Bridging Artificial Intelligence and Plant Biology: Innovations in Phenotyping and Transcriptome Analysis

Joshua Colmer

Degree of Doctor of Philosophy

University of East Anglia

Earlham Institute

June 2023

Abstract

Machine learning has the potential to revolutionise plant biology by offering unprecedented opportunities for predicting, measuring, and understanding complex biological processes. This thesis presents three projects that utilise machine learning techniques to tackle diverse challenges, ranging from seed germination detection to circadian time prediction and the identification of diagnostic biomarkers.

In the first project, SeedGerm, I developed a novel approach to automatically detect seed germination, a critical physiological process that determines the success of plant establishment. My pipeline utilises computer vision techniques for image-based seed segmentation and machine learning for predicting seed germination, enabling automated seed phenotyping for plant breeders and researchers.

For the second project, Trans-Learn, I identified diagnostic biomarkers for plant viruses through novel applications of image analysis and machine learning techniques. Plant viruses pose a major threat to global crop production, and biomarkers can facilitate the development of disease-resistant crop varieties. My supervised machine learning approach concentrates on transforming tabular datasets into tensors, intelligently arranging features, and interpreting a trained vision transformer to successfully isolate transcriptomic biomarkers in *Arabidopsis halleri* for turnip mosaic virus infection.

The third project, ChronoGauge, explored using transcriptomic biomarkers and multi-output regression models to predict the circadian clock, a fundamental biological mechanism that regulates the timing of processes in plants. Utilising circular regression techniques, statistical methods to quantify gene expression rhythmicity, and wrapper feature selection methods, I was able to predict the internal circadian time using transcriptomic data in *Arabidopsis thaliana* and wheat, surpassing the current state-of-the-art method in accuracy.

These projects collectively highlight machine learning's potential in addressing key challenges in plant biology. The open-source methods developed through these projects have applications to accelerate breeding practices and enable researchers to advance our understanding of plant biology. Overall, this thesis provides valuable insights into bridging the gap between computational techniques and biological research.

Acknowledgments

It is with a profound sense of appreciation and gratitude that I seize this opportunity to convey my heartfelt thanks to all those who have played a role in the completion of my PhD.

Foremost among these is my esteemed advisor, Prof. Anthony Hall, whose sagacious advice, steadfast guidance, and erudite scholarship have been invaluable to me throughout the course of my research. His patient tutelage and unwavering support have made this work and our future endeavours possible.

I also express my profound gratitude to my initial supervisor, Prof. Ji Zhou, who offered significant guidance and support during the early stages of my research journey. As a mentor, he has imbued me with motivation, confidence, and determination, enabling me to surmount challenges and obstacles.

My sincere thanks go to my secondary supervisor, Prof. Richard Morris, who has consistently provided support and encouragement throughout the highs and lows of my PhD journey.

I heartily appreciate Prof. Kate Kemsley, who played a crucial role in initiating my academic journey and inspiring my pursuit of a career in science. Without her guidance, I would not have recognised my potential or reached my current position.

To my colleagues and fellow researchers, who have generously shared their insights, knowledge, and expertise with me, I am deeply grateful. Their contributions have been instrumental in shaping the direction and focus of my research.

Finally, I would be remiss if I failed to convey my appreciation to my family and friends, whose love, encouragement, and unwavering support have sustained me throughout this challenging endeavour. To them, I offer my heartfelt thanks and deepest gratitude.

Table of Contents

List of Abbreviations

ABA – Abscisic Acid

AI – Artificial Intelligence

ANN – Artificial Neural Network

CAM – Class Activation Map

CNN – Convolutional Neural Network

CSV – Comma Separated Values

DE – Differential Expression

DEG – Differentially Expressed Gene

DFFS – Diversity Fixed Foundation Set

DL – Deep Learning

DNA - Deoxyribonucleic Acid

FCN – Fully Connected Network

FFT – Fast Fourier Transform

GA – Genetic Algorithm

GBM – Gradient Boosting Machine

GMM – Gaussian Mixture Model

GS – Genomic Selection

GWAS – Genome-Wide Association Study

GWHD – Global Wheat Head Detection

K-NN – K Nearest Neighbours

LGBM – Light Gradient Boosting Machine

miRNA – Micro RNA

mRNA – Messenger RNA

MAS – Marker Assisted Selection

MI – Mutual Information

ML – Machine Learning

MAE – Mean Absolute Error

MSE – Mean Squared Error

NN – Neural Network

NGS – Next Generation Sequencing

PCA – Principal Component Analysis

QTL – Quantitative Trait Locus

R-CNN – Region Convolutional Neural Network

ReLU – Rectified Linear Unit

RF – Random Forest

RGB – Red, Green, Blue

RNA – Ribonucleic Acid

RNN – Recurrent Neural Network

RPM – Reads Per Million

SFS – Sequential Feature Selection

SGD – Stochastic Gradient Descent

SVM – Support Vector Machine

TPM – Transcripts Per Million

TuMV – Turnip Mosaic Virus

ViT – Vision Transformer

YUV – Luminance and Chrominance Colour Space

List of Figures

List of Tables

List of Peer-Reviewed Publications

Peer-reviewed journal articles published during my PhD (2018-2023), not including conference proceedings.

Chapter 2. SeedGerm

- **Joshua Colmer**, Carmel M O'Neill, Rachel Wells, Aaron Bostrom, Daniel Reynolds, Danny Websdale, Gagan Shiralagi, Wei Lu, Qiaojun Lou, Thomas Le Cornu, Joshua Ball, Jim Renema, Gema Flores Andaluz, Rene Benjamins, Steven Penfield, Ji Zhou. SeedGerm: a cost-effective phenotyping platform for automated seed imaging and machine-learning based phenotypic analysis of crop seed germination. New Phytologist vol. 228 (2020)

Chapter 3. Trans-Learn (in preparation)

- **Joshua Colmer**, Mie N. Honjo, Hannah Rees, Jiawei Chen, Tomoaki Muranaka, Anthony Hall, Hiroshi Kudoh, Ji Zhou. Trans-Learn: a comprehensive machine learning platform for predicting host-disease interactions and multivariate biomarkers using the transcriptome.

Chapter 4. ChronoGauge

- Laura-Jayne Gardiner, Rachel Rusholme-Pilcher, **Josh Colmer**, Hannah Rees, Juan Manuel Crescente, Anna Paola Carrieri, Susan Duncan, Edward O Pyzer-Knapp, Ritesh Krishna, Anthony Hall. Interpreting machine learning models to investigate circadian regulation and facilitate exploration of clock function. Proceedings of the National Academy of Sciences vol. 118 (2021)
- Hannah Rees, Rachel Rusholme-Pilcher, Paul Bailey, **Joshua Colmer**, Benjamen White, Connor Reynolds, Sabrina Jaye Ward, Benedict Coombes, Calum A Graham, Luíza Lane de Barros Dantas, Antony N Dodd, Anthony

Hall. Circadian regulation of the transcriptome in a complex polyploid crop. PLOS Biology vol. 20 (2022)

- David Cuitun-Coronado, Hannah Rees, **Joshua Colmer**, Anthony Hall, Luíza L de Barros Dantas, Antony N Dodd. Circadian and diel regulation of photosynthesis in the bryophyte *Marchantia polymorpha*. Plant, Cell & Environment vol. 45 (2022)

Others:

- Yulei Zhu, Gang Sun, Guohui Ding, Jie Zhou, Mingxing Wen, Shichao Jin, Qiang Zhao, **Joshua Colmer**, Yanfeng Ding, Eric S Ober, Ji Zhou. Large-scale field phenotyping using backpack LiDAR and CropQuant-3D to measure structural variation in wheat. Plant Physiology vol. 187 (2021)
- Gang Sun, Hengyun Lu, Yan Zhao, Jie Zhou, Robert Jackson, Yongchun Wang, Ling-xiang Xu, Ahong Wang, **Joshua Colmer**, Eric Ober, Qiang Zhao, Bin Han, Ji Zhou. AirMeasurer: open-source software to quantify static and dynamic traits derived from multiseason aerial phenotyping to empower genetic mapping studies in rice. New Phytologist vol. 236 (2022)

Chapter 1: Introduction

Plant biology plays a critical role in addressing global challenges such as food security, climate change, and sustainable agriculture. As the world population continues to grow, innovative solutions are needed to increase crop yields, develop climate-resilient crop varieties, and combat plant diseases that threaten agricultural productivity. Recently, the emergence of advanced technologies such as computer vision, machine learning, and genomic sequencing has created new opportunities for interdisciplinary research. This thesis explores the potential of these cutting-edge technologies in solving complex problems in plant biology, with a focus on three distinct projects that each aim to provide new tools to the scientific community for advancing our understanding of plant systems.

The convergence of machine learning (ML) and plant biology has the potential to significantly enhance our ability to analyse and understand plant systems at a previously unattainable scale and resolution. By harnessing the power of ML algorithms, researchers have become empowered to process vast amounts of data from multiple sources, such as gene expression, genomic, proteomic, images, and environmental measurements, in order to gain valuable insights into plant processes, growth, and development. This fusion of disciplines enables the development of novel tools and methodologies that can accelerate and enhance plant research whilst also having an impact on global food security and sustainability.

There is high potential for novel tools and methodologies to yield significant impacts in several key domains, including precision agriculture, such as remote sensing and image analysis to provide real-time insights for crops; automated phenotyping to enable rapid assessment of favourable traits; crop trait prediction that incorporates environmental variation and measurements; and the identification of candidate gene targets for genome editing. This thesis explores research closely related to these domains, and I hope that the findings will significantly contribute to their advancement.

In this introductory chapter, I review the literature relevant to my PhD research, establishing a comprehensive background that encompasses a variety of machine learning applications in plant biology. Following this, I explain the motivation and the scope of my research, highlighting the interdisciplinary nature of the undertaking and the critical challenges that each project aims to surmount. Next, I detail the significance of my research, emphasising the potential impact of my results on the field of plant biology and their broader implications for agricultural productivity.

1.1 Background and Literature Review

In the following section, I will provide an in-depth review of the essential background knowledge and relevant literature associated with the three projects of this thesis. A comprehensive understanding of the relevant areas of plant biology, machine learning, and computer vision is critical for appreciating the context and significance of the research presented. This section is organised into five main subsections, each focusing on a specific topic, from plant biology and sequencing technologies to machine learning, biomarker detection, and computer vision. This review will explain the state of research in these areas, identify knowledge gaps, and underscore the potential impact of the novel findings within this thesis in the broader fields of plant biology and agriculture.

1.1.1 The Field of Plant Biology

The field of plant biology is an area of research that encompasses a wide range of topics, including but not limited to plant physiology, plant genetics, and the interactions between plants and their environments. Research in this field varies from investigating the role of plant hormones in regulating plant growth and development[1], the genetic basis of plant responses to biotic and abiotic stress[2], and the mechanisms underlying plant adaptations to changing environmental conditions[3]. As the global population continues to grow, there is an urgent need to improve crop production and ensure food security; research in the field of plant biology will become increasingly crucial in playing a critical role in achieving these goals.

Plant physiology is a fundamental aspect of plant biology, covering topics such as photosynthesis, metabolism, and plant responses to environmental stimuli. At the cellular and molecular level, plant physiology examines a range of processes that are crucial for understanding plant growth and development[4]. One of the critical processes investigated is photosynthesis, which involves the conversion of light energy into chemical energy and is fundamental to the functioning of plants[5]. Plant physiology studies have also been conducted to investigate the role of plant signalling pathways in mediating plant responses to biotic and abiotic stress[6]. These

studies highlight the importance of plant physiology research in understanding plant function, which can be exploited to develop strategies for improving crop production. Recent advancements in plant physiology research have deepened our understanding of plant function at the molecular and cellular levels, which can be exploited to improve crop production in the face of global environmental change[7].

In recent years due to increased capabilities of sequencing technologies and data analysis methods, plant genetics has emerged as a critical area of focus, including the study of gene expression, genetic variation, and genetic engineering[8]. A notable advancement in this domain is the investigation of the genetic basis of plant-pathogen interactions, which has implications for crop protection and disease resistance. For example, Oliva et al. utilised CRISPR-Cas9 genome editing to mutations in the promoters of three SWEET gene promoters to induce disease resistance in rice [9]. The mutations created in rice lines Kitaake, IR64, and Ciherang-Sub1 conferred broad-spectrum resistance to the pathogen, suggesting that this approach could be an effective strategy for crop protection. Furthermore, researchers have made strides in identifying genes implicated in plant growth and development, with several studies reporting on genome-wide association studies (GWAS) that have uncovered numerous candidate genes associated with plant architecture and flowering time, providing insights into the genetic regulation of these traits[10,11]. In addition to these studies, advances in genome editing technologies have facilitated crop trait improvement. Zaidi et al. reviewed the application of CRISPR/Cas9 and other genome editing tools in generating targeted mutations to improve crop traits, such as quality, yield, and resistance to biotic and abiotic stressors[12].

The interactions between plants and their environment are intricate, with plants needing to react and adjust to alterations in their surroundings. These interactions encompass a broad range of factors, including abiotic and biotic stresses, nutrient availability, and changes in temperature and precipitation patterns. A study by Iqbal et al. reviewed the impacts of drought stress on plant growth and development, discussing the importance of hormone signalling pathways in regulating plant responses to drought[13]. Temperature, as a critical environmental factor, influences plant growth and development, leading plants to evolve various strategies, such as alterations in gene expression, to cope with temperature fluctuations.[14]. The study of

complex regulatory networks governing temperature-dependent growth and development in plants has immense potential for informing breeding programs aimed at enhancing crop resilience in the face of global climate change[15].

The leading model organism in the field of plant biology is *Arabidopsis thaliana,* a plant within the Brassicaceae family, primarily due to its relatively simple genome and rapid lifecycle[16]. With a compact genome size of approximately 135 Mb, *Arabidopsis thaliana* offers researchers the advantage of a sequenced and annotated reference genome[17], facilitating the identification and functional analysis of genes underlying biological processes. Furthermore, its short generation time of 6-8 weeks and abundant seed production enable rapid progress in genetic and molecular studies[18]. While *Arabidopsis thaliana* has served as a valuable model organism for understanding fundamental plant biology, translating these findings to economically important crops presents several challenges. One primary obstacle is the inherent difference in genome structure and complexity between *Arabidopsis thaliana* and many crop plants. For instance, polyploidy, the presence of multiple sets of chromosomes, is common in crop species, complicating the extrapolation of findings from *Arabidopsis thaliana's* diploid genome[19]. Furthermore, some essential agronomic traits, such as perenniality, are not exhibited by *Arabidopsis thaliana*, which is an annual plant[20].

1.1.1.1 The Importance of Crop Traits

Crop traits are the heritable characteristics of plants that hold agricultural significance and have been the subject of extensive research, covering a diverse set of features and morphological, physiological, and biochemical properties[21]. Examples of crop traits include resistance to pests, diseases, and environmental stressors, such as drought, salinity, and temperature fluctuations.

The study and manipulation of crop traits have been crucial in advancing plant breeding efforts, with the ultimate goals of improving agricultural productivity, enhancing crop quality, and promoting the sustainability of food production systems. The recent advent of genomic tools and high-throughput phenotyping have further

enabled the characterisation of crop traits at a molecular level[22]. Quantitative Trait Loci (QTL) mapping and GWAS have been instrumental in the identification of genetic loci associated with complex traits, allowing for more targeted breeding approaches[23]. Additionally, recent advancements in genome editing technologies, for example, systems such as CRISPR/Cas9 system, have supplied plant breeders with an efficient method to introduce desirable traits into crops and accelerate breeding programmes[24]. By investigating the genetic basis of these traits, studies have shown it is possible to identify and select for desirable characteristics, facilitating the development of improved crop varieties[21]. With an increasingly unpredictable climate, the strategy of utilising our understanding of the genetics of crop traits is being used to develop climate-resilient crop varieties. By understanding the genetic mechanisms underlying stress tolerance, researchers can facilitate the breeding of crops capable of withstanding the challenges posed by a changing environment, such as droughts[25].

Traits concerning seed size, composition, and nutrient content have been shown to play a crucial role in determining the nutritional value of a crop, as well as its suitability for human consumption or animal feed[26]. Investigations by White and Broadley reveal that modifications in seed micronutrient content and bioavailability can result in crops with enhanced nutritional properties, contributing to improved dietary quality and mitigating micronutrient deficiencies in human and animal populations[27]. Consequently, the targeted manipulation of specific crop traits offers a promising avenue for addressing food security and nutritional challenges in the face of a growing global population, as demonstrated by Ortiz-Monasterio et al.[28]. Crop performance and stability are significantly influenced by the capacity to withstand pests, diseases, and environmental stressors[29]. Therefore traits that confer resistance to both biotic (pests and pathogens) and abiotic (drought, salinity, temperature extremes) stresses have been identified as vital factors in minimising yield losses and promoting more sustainable agricultural practices[30].

Due to an increase in the quantity and quality of genomic data being generated, significant progress has been made in understanding the genetic components of crop traits, which has facilitated the development of molecular markers and genomic tools for plant breeding[31]. Molecular markers have been instrumental in the identification

and selection of desirable traits, including enhanced yield potential, stress tolerance, and resistance to diseases[32]. For example, QTL mapping has been used to identify loci associated with important agricultural traits, such as yield, plant height, and disease resistance[33]. Molecular markers have been used to improve the efficiency of traditional breeding programs through a technique known as marker-assisted selection (MAS)[34]. MAS allows for the early selection of superior genotypes carrying desirable alleles, reducing the number of breeding cycles required to develop improved crop varieties[35]. Furthermore, genomic selection (GS) enables breeders to estimate genomic estimated breeding values (GEBVs), significantly accelerating the breeding process[36]. By integrating phenotypic data with genotypic markers, GS has been shown to enhance genetic gain per unit time compared to traditional breeding approaches[37].

1.1.1.2 Seed Germination

Seed germination is a critical stage in the plant life cycle, marking the transition from a quiescent, desiccated state to an actively growing seedling[38]. Germination is initiated by the absorption of water by the seed, a process known as imbibition, which triggers the activation of metabolic processes, mobilisation of stored nutrients, and initiation of cell division and elongation[39]. Moreover, the process of germination is tightly regulated by a complex interplay of hormones, such as abscisic acid (ABA) and gibberellins, which modulate the balance between dormancy and germination[40]. ABA is primarily responsible for maintaining seed dormancy, while gibberellins promote germination by stimulating the synthesis of hydrolytic enzymes and the breakdown of storage compounds[41,42].

Germination is an essential determinant of crop establishment and yield potential as rapid, uniform germination, and the emergence of seedlings are essential for maximising crop productivity and ensuring efficient use of resources, such as water, nutrients, and light[43]. Moreover, seed germination is influenced by various biotic and abiotic factors, including seed quality, dormancy status, soil conditions, and the presence of pathogens or pests[44]. Understanding the mechanisms underlying seed germination and their modulation by environmental factors is critical for optimising

agronomic practices and developing strategies to enhance crop performance[45]. Further research is needed to translate this knowledge into improved crop management practices and the development of crop varieties with enhanced germination characteristics under diverse environmental conditions.

When breeding for better-performing crops, essential traits for breeders to consider are germination rate, uniformity, and speed. Germination rate refers to the proportion of seeds that successfully germinate within a given time frame, typically expressed as a percentage. A high germination rate is vital for crop establishment and efficient use of resources, as it ensures that a larger proportion of seeds planted will produce viable seedlings, ultimately leading to higher crop yields[45]. Germination rate is influenced by factors such as seed quality, seed treatments, and environmental conditions[46].

Germination uniformity is the degree to which seeds germinate simultaneously under similar conditions. Uniform germination is crucial for crop productivity, as it enables plants to compete more effectively for resources, such as light, water, and nutrients, and reduces the potential for shading and competition among plants of different developmental stages[47]. Another advantage of having uniform germination is that it facilitates more predictable and efficient crop management practices, such as irrigation, fertilisation, and pest control.

Germination speed, or the time it takes for seeds to germinate, is an important trait in crop production as rapid germination can provide competitive advantages for plants, enabling them to establish more quickly and outcompete weeds for resources[48]. Faster germinating crops also help to reduce the time that seeds are exposed to potential pathogens and pests in the soil, thus lowering the risk of seedling damage or loss[49]. Furthermore, rapid germination is particularly important in regions with short growing seasons or variable weather conditions, as it allows crops to make better use of available growing time and increases the chances of successful crop production[50].

1.1.1.3 Plant-Pathogen Transcriptional Interactions

Gene expression, which is the process through which genetic information is used to produce proteins and perform other vital functions, and transcriptional responses play a crucial role in the complex and multifaceted molecular and cellular events that occur during plant-pathogen interactions[51]. These interactions, involving bacteria, fungi, or viruses, are essential in determining the outcomes of infections, as they involve dynamic interplay between the host plant and the invading pathogen[52]. Plants have evolved a diverse array of defence mechanisms that involve transcriptional regulation to counteract pathogen invasion, while pathogens continually develop strategies to evade or suppress host defences at the gene expression level[53,54]. Over time, coevolution between plants and pathogens has driven an ongoing arms race, with both parties continuously developing novel gene expression strategies to outmanoeuvre the other[55]. For example, pathogens have evolved to suppress host defences through various means, such as the secretion of small RNAs that target and silence host defence genes[56]. In response, plants have developed mechanisms such as through specific Argonaute proteins to recognise and degrade these pathogen-derived small RNAs, further bolstering their defence against invading pathogens at the transcriptional level[57].

Plants possess an innate immune system that recognises and reacts to molecules derived from pathogens, leading to changes in gene expression[58]. Various factors influence plant-pathogen interactions at the transcriptional level, such as the genetic makeup of both the host and the pathogen, environmental conditions, and the presence of other biotic agents[59,60]. Recent advancements in sequencing technologies and functional genomics have facilitated the identification of numerous genes and pathways implicated in plant immunity, as well as pathogen virulence factors that modulate host defences at the transcriptional level[61]. Understanding the molecular basis of gene expression and transcriptional responses in plant-pathogen interactions is essential for devising strategies to improve plant disease resistance and ensure sustainable crop production[62].

The study of gene expression and transcriptional responses in plant-pathogen interactions holds significant importance for crop development and agriculture. By understanding these interactions, researchers can develop new strategies to enhance crop resistance to pathogens and reduce yield loss. Advances in plant breeding techniques, such as MAS and genetic engineering, have facilitated the integration of novel resistance genes identified through the study of plant-pathogen interactions into crop varieties[63]. As a result, these improved crops are more resistant to diseases, reducing the need for chemical interventions and contributing to more sustainable agricultural practices. Additionally, understanding the molecular mechanisms of plant-pathogen interactions can inform the development of targeted approaches to manage plant diseases, such as the use of biological control agents or the application of specific elicitors that enhance plant defences[64].

1.1.1.4 The Circadian Clock

The circadian clock is an endogenous, self-sustaining timekeeping system that plays a critical role in enabling organisms to both anticipate and adapt to critical daily environmental fluctuations, such as light and temperature[65]. In plants, the circadian clock has been observed to regulate a multitude of physiological processes, including photosynthesis[66], stomatal movement[67], and flowering time[68], thereby optimising growth and development across the diurnal cycle[69].

The circadian clock in organisms, including plants, is entrained by external cues known as zeitgebers, which help synchronise the endogenous rhythms with environmental fluctuations[70]. Among these zeitgebers, light and temperature play prominent roles in regulating the clock's phase and period, ensuring that the organism's internal rhythms remain in sync with the external day-night cycle[71]. In addition to its role in developmental transitions, the circadian clock influences plant responses to both biotic and abiotic stressors, modulating defence mechanisms and stress tolerance pathways in a time-of-day-dependent manner[72]. This allows plants to anticipate and prepare for predictable daily changes in environmental conditions, such as changes in temperature, precipitation, or pathogen activity, and mount more effective responses[73,74]. Another example of the circadian clock's influence on a trait

is flowering time, which is crucial for successful pollination and seed development, ensuring proper reproductive timing in response to environmental cues[75]. Each of the aforementioned processes and traits are important for crop development, highlighting the need for a deeper understanding of the circadian clock's role in crop development and its potential applications in agriculture[76] that could lead to sustainable improvements in crop production, ultimately contributing to global food security.

The plant circadian clock is comprised of a network of interconnected transcriptional-translational feedback loops, generating gene expression oscillations with a period of approximately 24 hours[77]. The core loop of the plant circadian clock involves a set of key transcription factors, including CIRCADIAN CLOCK ASSOCIATED 1 (CCA1), LATE ELONGATED HYPOCOTYL (LHY), and TIMING OF CAB EXPRESSION 1 (TOC1)[78]. These factors form a negative feedback loop that drives rhythmic gene expression. Specifically, CCA1 and LHY act as transcriptional repressors, inhibiting the expression of TOC1 during the day, while TOC1 accumulates during the night and in turn represses CCA1 and LHY expression[79].

1.1.2. RNA-Sequencing in Plant Biology

Over the past two decades, sequencing technologies have brought about a paradigm shift in plant biology, enabling unparalleled comprehension of plant genomes, their structure, function, and evolutionary processes[80]. In this section, I will concentrate on RNA-sequencing (RNA-Seq), a robust and widely-used application of next-generation sequencing (NGS) technologies. RNA-Seq has been instrumental in enhancing our knowledge of plant transcriptomes and the intricate regulatory mechanisms governing them[81]. RNA-Seq has emerged as the standard method for quantifying gene expression, as it offers significant advantages over previous techniques, such as microarrays[82]. The capacity of RNA-Seq to provide strand-specific and quantitative data on transcript abundance, as well as the ability to identify novel transcripts, has made it an indispensable tool in plant biology research[83].

Compared to microarrays, RNA-Seq provides a broader dynamic range and higher sensitivity, enabling the detection of low-abundance transcripts and novel splice variants[84]. Furthermore, RNA-Seq is not limited by the need for prior knowledge of the target sequences, which allows for the identification of novel transcripts and isoforms[85]. These advantages have led to the widespread adoption of RNA-Seq in plant biology research, with applications ranging from gene expression profiling and alternative splicing analysis to the identification of non-coding RNAs and small RNAs[86]. Moreover, RNA-Seq has proven to be a valuable tool for comparative transcriptomics, enabling researchers to investigate the conservation and divergence of gene expression patterns across multiple plant species[87]. This has resulted in a deeper understanding of plant evolution, speciation, and adaptation to various environmental conditions[88].

One of the significant advantages of RNA-Seq is that the dynamic range of RNA-Seq is significantly higher than that of microarrays, enabling the quantification of transcripts across a broader range of expression levels. Another advantage is that RNA-Seq offers high reproducibility and quantitative accuracy, making it suitable for differential gene expression analysis and comparative transcriptomics. This capability allows researchers to investigate the molecular mechanisms underlying important crop traits and to identify candidate genes for breeding and genetic engineering.

In plant biology, RNA-Seq has been particularly useful for studying abiotic and biotic stress responses, developmental processes, and the regulation of metabolic pathways[89]. For instance, RNA-Seq has facilitated the analysis of complex transcriptional networks involved in plant responses to environmental challenges, such as drought, salt stress, and pathogen attack[90]. This has allowed researchers to uncover key regulatory genes and pathways that could be targeted for crop improvement and stress tolerance[91]. In addition, RNA-Seq has been employed to investigate developmental processes in plants, such as the molecular mechanisms controlling fruit ripening and flowering time[92].

1.1.3 Machine Learning: An Overview

Machine learning (ML), a prominent subfield of artificial intelligence (AI), emphasises the creation of algorithms and models that enable computers to learn from and make predictions or decisions rooted in data. In recent years, ML has garnered considerable attention due to its capacity to analyse extensive datasets and uncover intricate patterns that often prove challenging or infeasible to discern using conventional statistical methods[93].

In this literature review, I explore the core concepts and techniques in machine learning, focusing on their relevance to the projects discussed in this thesis. Supervised learning involves training models to make predictions based on labelled input-output pairs. This approach has led to numerous successful applications, including image classification[94], natural language processing[95], and medical diagnostics[96]. Generally, supervised learning encompasses two types – classification, which predicts a finite number of classes, and regression, involving the prediction of a continuous target. Unsupervised learning, on the other hand, aims to extract patterns from unlabelled data by leveraging various techniques, such as clustering and dimensionality reduction[97].

Deep learning, a subset of ML, employs artificial neural networks with multiple layers to model high-level abstractions and representations in data[98]. The advent of deep learning has significantly advanced the field of ML, enabling the development of more accurate and sophisticated models in various domains[99]. Machine learning algorithms have been extensively employed across various domains, including plant biology[100]. The rest of this section examines and explains some of the essential algorithms and methods in the field of ML:

Linear regression is an easily interpretable method used for modelling the relationship between a continuous target variable and one or more input features. Linear regression assumes a linear relationship between the input features and the target variable, which allows for predictions based on a linear combination of the input features[101].

Logistic regression (LR) is a variant of linear regression tailored for binary classification tasks where the target variable has two possible outcomes but can also be used in multiclass classification scenarios. Logistic regression employs the logistic function to model the probability of the target variable belonging to a specific class, thus facilitating probabilistic predictions[102].

Decision trees are a tree-like hierarchical structure that recursively partitions the input space based on the values of input features, ultimately leading to a decision or prediction. Decision trees can be applied to both regression and classification problems, and they are easily interpretable due to their intuitive, rule-based structure[103].

Random Forest (RF) is an ensemble learning method that constructs multiple decision trees during training and combines their predictions using majority voting or averaging. RFs are robust to noise, capable of handling high-dimensional datasets, and can model non-linear relationships, but are more computationally expensive than simpler algorithms[104].

Gradient Boosting Machines (GBM) are another ensemble learning method that sequentially builds and combines weak learners, typically shallow decision trees, by iteratively fitting them to the residuals of the previous learners[105]. GBMs can capture complex relationships, are robust to noise, and handle a variety of data types, but can be prone to overfitting and require careful tuning of hyperparameters, such as learning rate and tree depth.

K-Nearest Neighbors (K-NN) is an instance-based, non-parametric learning algorithm used for both classification and regression tasks. It relies on the assumption that similar inputs have similar outputs and makes predictions based on the majority class or the average target value of the k closest training samples in the feature space[106]. K-NN is easy to implement and can model complex relationships, but its performance can be sensitive to the choice of the distance metric, the value of k, the dimensionality of the input space, and the presence of irrelevant features.

Support Vector Machines (SVM) are a family of algorithms that can be applied to both regression and classification tasks. They aim to find the optimal hyperplane or decision boundary that maximises the margin between classes or between the regression line and the training samples[107]. SVMs can model non-linear relationships through the use of kernel functions such as the radial basis function kernel but can be sensitive to hyperparameter choices and may require more computational resources for large datasets[108].

Artificial neural networks (ANN) are inspired by the structure and function of biological neural networks, consisting of interconnected nodes (neurons) that process and transmit information[109]. ANNs are a type of deep learning (DL) algorithm and are capable of utilising non-linear multivariate relationships to solve complex problems.

Ensemble methods have emerged as a powerful tool in supervised machine learning, offering significant improvements in predictive performance by combining multiple base learners[110]. These methods operate under the principle that the collective knowledge of multiple models leads to more accurate and robust predictions than a single model alone. A wide range of ensemble techniques have been developed, including bagging, boosting, and stacking, each with their unique strengths and limitations[111,112]. Ensemble methods have been successfully applied across various domains, such as computer vision[113] and bioinformatics[114]. Through leveraging the diversity and strengths of multiple models, ensemble methods provide a robust and efficient approach to address the challenges faced in supervised machine learning tasks.

Feature selection, a crucial step in the majority of machine learning pipelines, is the process of identifying and selecting the most relevant input variables (features) that contribute significantly to the prediction of the target variable[115]. The primary goal of feature selection is to reduce the dimensionality of the dataset, thereby mitigating the curse of dimensionality and improving the generalisation of the model. Moreover, feature selection techniques can enhance the interpretability of the model, reduce training time, and minimise the risk of overfitting[116]. Various feature

selection methods have been proposed in the literature, which can be broadly classified into three categories: filter, wrapper, and embedded methods[117].

Filter methods evaluate the relevance of features independently of the learning algorithm, relying on intrinsic properties of the data, such as correlation, mutual information, or statistical tests[118]. These methods are computationally efficient and less prone to overfitting; however, they do not account for the interaction between features and the specific learning algorithm[119]. Wrapper methods, on the other hand, assess the quality of feature subsets based on the performance of a given learning algorithm[120]. Although wrapper methods can yield better performance, they are computationally expensive and more susceptible to overfitting due to the exhaustive search in the feature space. Embedded methods incorporate feature selection as part of the learning algorithm itself, often using regularisation techniques or in-built criteria to select the most relevant features[121]. These methods strike a balance between filter and wrapper approaches, offering improved performance and reduced computational cost.

Model testing and validation are essential steps in the machine learning pipeline to assess the performance and generalisation capability of a trained model on unseen data[122]. A common approach is to partition the available dataset into two or three disjoint subsets: training, validation, and testing sets. The training set is utilised for model fitting, the validation set for hyperparameter tuning and model selection, and the testing set for evaluating the model's performance[123]. However, this straightforward approach can lead to a high variance in the performance estimate, particularly when the dataset is limited in size or imbalanced. To obtain more reliable and robust evaluations of machine learning models, cross-validation techniques have been developed, which address these limitations by providing multiple performance estimates based on various data partitions[124].

K-fold cross-validation is a widely-used technique in which the dataset is divided into k equally-sized folds. For each iteration, one fold is held out as the validation set, while the remaining $k - 1$ folds are used for training. This process is repeated k times, yielding k performance estimates, which are then averaged to provide a more

accurate and stable evaluation of the model's performance[125]. A special case of k-fold cross-validation is leave-one-out cross-validation (LOOCV), where k is equal to the number of samples in the dataset, resulting in a single observation being used as the validation set in each iteration. While LOOCV can provide a low-biased performance estimate, it is computationally expensive for large datasets[124]. Stratified k-fold cross-validation is another variation, particularly useful for imbalanced datasets, where each fold is constructed to maintain the same class distribution as the original dataset[126].

Hyperparameters are parameters of machine learning algorithms that are not learned during model training but are set a priori. They control various aspects of the learning process, such as the architecture of a neural network, the regularisation strength in linear regression, or the depth of a decision tree. The choice of hyperparameters can significantly impact the performance of a machine learning model, and finding the optimal configuration is a critical task known as hyperparameter tuning[127]. This process involves searching the hyperparameter space for a configuration that yields the best generalisation performance, typically measured by cross-validation or held-out validation data.

Various strategies have been proposed for hyperparameter tuning, ranging from simple grid search and random search to more advanced techniques like Bayesian optimisation, genetic algorithms, and gradient-based optimisation [128]. Grid search involves an exhaustive search over a pre-defined set of hyperparameter values, while random search samples hyperparameter configurations from a specified distribution. Although these methods are easy to implement, they can be computationally expensive and inefficient, especially for high-dimensional hyperparameter spaces. Bayesian optimisation, on the other hand, employs a probabilistic model to efficiently explore the hyperparameter space by balancing the trade-off between exploration and exploitation[129]. This method has been shown to outperform traditional search techniques in various settings, albeit with an increased computational overhead for model fitting. Other optimisation techniques, such as genetic algorithms and gradient-based methods, have also been applied to hyperparameter tuning, offering alternative approaches to navigate complex search spaces.

Evaluation metrics play a vital role in the development and assessment of machine learning models, as they are the method of quantifying the performance of the models on specific tasks. Depending on the problem domain and the type of learning algorithm used, different evaluation metrics may be used to measure various aspects of a model's performance, such as accuracy, precision, recall, F1-score, area under the receiver operating characteristic curve (AUC-ROC)[130], and mean squared error (MSE). The choice of an appropriate evaluation metric is crucial, as it can significantly influence the model selection process and the interpretation of the results[131]. Moreover, evaluation metrics should take into account the specific characteristics of the problem, such as class imbalance or the relative importance of different types of errors, to ensure a meaningful assessment of the model's performance[132].

Loss functions, also known as cost or objective functions, serve as a measure of the discrepancy between the predicted outputs of a machine learning model and the actual target values. During model training, models aim to minimise the loss function by optimising parameters until training is complete. Various loss functions have been proposed in the literature, with the choice of an appropriate function dependent on the specific problem domain and the type of learning algorithm employed[133].

For regression tasks, commonly used loss functions include mean squared error (MSE), mean absolute error (MAE), and Huber loss, which measure the average deviation of predicted values from the true targets raised to different powers. However, each does so differently as MSE squares the deviations, MAE considers the absolute differences, and Huber loss, being less sensitive to outliers, uses a combination of both. In classification tasks, cross-entropy loss (log loss) and hinge loss are frequently utilised, as they provide a measure of the model's ability to assign correct class probabilities or large margin separation, respectively. The selection of an appropriate loss function is critical, as it influences the model's learning dynamics and generalisation performance.

Optimisation methods, particularly gradient-based techniques, play a critical role in machine learning and deep learning by enabling efficient training of models to minimise a given loss function. Stochastic Gradient Descent (SGD) is a widely-used optimisation algorithm that updates model parameters by estimating the gradient of the loss function using a random subset or a single instance of the training data at each step[134]. This approach offers computational efficiency, faster convergence, and the ability to escape local optima in non-convex optimisation problems, which are common in deep learning. However, SGD can be sensitive to the choice of learning rate and may suffer from oscillations or slow convergence near the optimal solution[135]. To address these challenges, several adaptive gradient-based optimisation algorithms have been proposed, such as AdaGrad[136] and Adam[137], which adapt the learning rate for each parameter based on historical gradients, resulting in improved convergence properties and reduced sensitivity to hyperparameter choices. The selection of appropriate optimisation methods is crucial for the effective training of machine learning and deep learning models, directly impacting their performance and generalisation capabilities.

Supervised machine learning research faces several ongoing challenges that need to be addressed to advance the field further and unlock new possibilities. One critical challenge is developing models with improved interpretability, as the increasing complexity of machine learning models, particularly deep learning architectures, has led to a trade-off between model performance and interpretability[138]. Another challenge is the requirement of large amounts of labelled data for training supervised models, which can be time-consuming and expensive to obtain. Developing methods for effective learning with limited labelled data, such as few-shot learning and semi-supervised learning, has the potential to expand the applicability of machine learning in real-world scenarios greatly [139].

Interpretability of machine learning algorithms has gained increasing attention, as understanding the rationale behind model predictions is crucial for building trust, ensuring fairness, and facilitating the adoption of these models in sensitive domains[140]. A variety of methods have been developed to provide insights into the internal workings of machine learning models, particularly for deep learning architectures such as Convolutional Neural Networks (CNNs). Grad-CAM

(Gradient-weighted Class Activation Mapping) is one such method, which generates visual explanations of CNN-based models by highlighting the regions in the input image that contribute the most to the model's prediction. This technique computes the gradients of the predicted class with respect to the feature maps of the last convolutional layer, which are then used to produce an approximate localisation map highlighting important regions.

In the context of this thesis, machine learning techniques play a crucial role in all three research projects, from computer vision-based seed germination detection to circadian time prediction using gene expression data and disease diagnostic biomarker detection. The following sections delve into the specific applications of machine learning in plant biology, as well as related areas such as biomarker detection, computer vision, and crop trait prediction.

1.1.3.1 Machine Learning in Plant Biology

Machine learning (ML) has emerged as a powerful tool in advancing plant biology research by offering insights into complex biological processes and addressing multiple challenges related to crop productivity, stress tolerance, and disease resistance[141]. Here, I compare statistical and machine learning techniques in plant biology as well as the potential of applying supervised machine learning to plant biology, emphasising its relevance to high-throughput crop phenotyping and trait prediction using gene expression.

Statistical and machine learning approaches have both been extensively used to analyse data in plant biology, with each methodology having its unique advantages and applications. Statistical approaches, such as analysis of variance (ANOVA) and principal component analysis (PCA), have been employed for decades to analyse plant phenotypic and genotypic data[142]. PCA is an unsupervised technique that aims to reduce the dimensionality of high-dimensional data by transforming the original variables into a smaller set of new variables that capture the maximum amount of variance in the data, ultimately facilitating the identification of patterns or clusters associated with specific phenotypes or conditions, as well as detecting potential

outliers. These methods have been particularly useful in QTL mapping and GWAS to identify genetic markers associated with important agronomic traits[143]. Statistical models provide a simple and interpretable framework for hypothesis testing and understanding the relationships between variables.

Supervised machine learning approaches have gained increasing attention in plant biology due to their ability to handle large and complex datasets. Methods such as random forests, support vector machines, and deep learning algorithms can identify nonlinear relationships and high-order interactions between variables, which are often difficult to capture with traditional statistical analyses and models. However, their weaknesses include limited interpretability, reliance on data quality and potential for overfitting, and the need for careful model selection and hyperparameter tuning. On the other hand, statistical approaches boast strong inferential and interpretative capabilities, well-established methodologies, and explicit assumptions that aid in result interpretation. These attributes make statistical methods suitable for applications focused on interpretability such as identifying differentially expressed genes[144] using DESeq2[145] or identifying gene regulatory relationships[146] through WGCNA[147]. However, statistical approaches can struggle with handling large, complex datasets, and are sensitive to assumptions and data quality issues such as noise, outliers, and missing data. Supervised machine learning methods, due to their capability of detecting intricate, multivariate patterns among diverse datatypes, emerge as a strong choice for predictive modelling in plant biology.

While both statistical and machine learning approaches have their respective advantages, the choice between them depends on the specific goals and the nature of the dataset. Statistical methods are generally more suitable for smaller datasets and hypothesis-driven research, whereas machine learning is better suited for large datasets with complex interactions and patterns. With the cost of remote sensing and sequencing decreasing, the quantity of data being generated is at unprecedented levels, so the integration of machine learning approaches into the field of plant biology is essential for developing our understanding of biological processes and making predictions for crop improvement[148]. Also, as machine learning continues to evolve, its applications in plant biology are anticipated to expand and become increasingly sophisticated, contributing to more efficient and sustainable agricultural

practices. Furthermore, the fusion of multiple data sources, such as remote sensing, genomics, and phenomics data, can enable the development of powerful models capable of understanding plant processes and their interactions with the environment.

Despite the potential advantages, several challenges remain in applying machine learning to plant biology. One significant challenge is the availability and quality of labelled data for model training, as the generation of high-quality, annotated datasets can be time-consuming and expensive. Additionally, data heterogeneity arising from diverse data sources, such as multispectral and hyperspectral images, and genomics and phenomics data, may complicate model development and require sophisticated pre-processing techniques to ensure compatibility. Furthermore, the interpretability and transparency of complex machine learning models, particularly deep learning approaches, may be limited, posing challenges for model validation and adoption by domain experts[149].

Machine learning algorithms have been increasingly employed to analyse high-throughput phenotyping data, enabling the identification of QTLs and the prediction of complex, polygenic traits[150]. These techniques facilitate the selection of desirable genotypes in breeding programs and improve our understanding of the genetic architecture of important crop traits. High-throughput phenotyping platforms with advanced machine learning algorithms have allowed researchers and the agricultural sector to identify and quantify plant traits more accurately.

One case study demonstrating the potential of supervised machine learning for high-throughput plant phenotyping is the work of Lu et al.[151], who applied CNNs for the automated identification of rice diseases from over 5,000 leaf images containing four diseases. Their combination of pre-processing techniques and supervised CNNs achieved an accuracy of over 95%, demonstrating the potential of ML-based disease detection for improving crop phenotyping and reducing yield loss. A similar application of ML to plant biology is the development of software to quantify yield-related phenotypes in lettuces using a customised pipeline of computer vision algorithms and a deep learning classifier[152]. Using the developed pipeline, phenotypes for millions of in-field lettuces were predicted with an accuracy of over 98%, allowing growers to make informed decisions about agricultural processes

related to harvesting. These case studies highlight the potential for how machine learning methods can automate and accelerate crop phenotyping to allow breeders and growers to make more informed decisions and improve agricultural productivity.

In addition to applying machine learning to remote sensing datasets for high-throughput plant phenotyping, supervised machine learning techniques have been applied to high-dimensional tabular gene expression datasets. For example, in Wang et al.[153], the authors present a novel approach for identifying photosynthesis-related genes in maize where they utilised an ensemble-based machine learning method with a majority voting scheme. The effectiveness of incorporating RNA-Seq data from multiple photosynthetic mutants to improve prediction accuracy was demonstrated as their approach successfully predicted 716 photosynthesis-related genes in maize that were previously lacking a known gene function.

Another example would be the application of supervised machine learning to predict mature plant complex traits in maize (*Zea mays*) at different developmental stages[154]. In this study, the findings indicate that transcript levels are comparable to genetic marker models in terms of predictive performance and that when the most important transcripts and genetic markers were combined into one model, the overall predictive ability was enhanced. This study also revealed a key finding that genetic markers crucial for predictions were not necessarily linked to the regulation of important transcripts, suggesting that the predictive power of transcript levels is not solely attributed to genetic variation within transcribed genomic regions. The authors emphasise that transcriptome data may offer valuable insights into the relationship between traits and genetic variation that are difficult to discern at the sequence level, as demonstrated by the identification of a larger number of benchmark flowering-time genes in transcript models compared to genetic marker models. This study highlights the utility of transcriptome data in the prediction and understanding of complex traits.

In summary, machine learning techniques have been instrumental in addressing various challenges in plant biology research, contributing to our understanding of complex biological processes and enhancing crop productivity. The projects described in this thesis showcase the potential of machine learning to further advance

plant biology research by addressing specific problems in seed germination detection, circadian time prediction, and disease diagnostic biomarker detection.

1.1.4 Biomarker Detection

The study of biomarkers in plants has gained significant attention in recent years due to the capability of generating the requisite datasets and their potential to provide valuable insights into various plant processes and traits[155]. These biomarkers can be classified into distinct categories, including transcriptomic, genomic, and proteomic markers. Here, I introduce different types of plant biomarkers, focusing on transcriptomic biomarkers and their associations with specific phenotypes, as well as the techniques used to identify these biomarkers from biological datasets.

Transcriptomic biomarkers focus on differentially expressed genes or transcripts, reflecting the dynamic nature of gene expression in response to various stimuli[156]. Genomic biomarkers, on the other hand, are specific DNA sequences or variations that can provide insights into the genetic makeup of plants and how these sequences relate to different traits and biological processes[157]. Proteomic biomarkers encompass differentially expressed proteins or protein modifications, highlighting the functional outcomes of gene expression and regulation on the cellular level[158]. With the emergence of high-throughput technologies like next-generation sequencing as well as protein microarrays, these respective fields have experienced significant advancements. All three of these forms of biomarkers can help us understand the gene expression patterns, DNA sequences, and protein expression patterns that are associated and responsible for plant development, stress response, and other important traits and processes.

These forms of biomarkers have been utilised to monitor physiological processes, including growth and development, photosynthesis, and nutrient uptake[159,160]. Furthermore, they have been employed as indicators of plant stress responses, such as drought[161], salinity[162], and pathogen exposure, thereby providing crucial information on plant adaptability and resilience. Moreover, plant biomarkers have

also played a significant role in understanding disease states and facilitating early detection of plant pathogens, which is vital for effective disease management[163].

A notable transcriptomic biomarker study is of WRKY transcription factors, which are involved in regulating a variety of plant processes, including responses to biotic and abiotic stresses and senescence[164]. Several studies have investigated the role of these transcription factors in different plant species, such as Arabidopsis[165] and rice[166], to unravel their function in plant defence mechanisms and tolerance to environmental challenges. By identifying and characterising these transcriptomic biomarkers, researchers have gained a better understanding of the complex gene regulatory networks that dictate plant responses to biotic and abiotic stresses, assisting with the development of stress-resistant crop varieties.

Another area of interest in transcriptomic biomarker research in plant biology is the study of gene expression changes in response to various environmental factors, such as temperature, drought, and nutrient availability. In a study conducted on wheat, gene expression changes in response to drought stress were investigated using high-throughput RNA sequencing, leading to the identification of numerous differentially expressed genes related to stress response and adaptation[167]. Similarly, transcriptome analysis of the model plant *Arabidopsis thaliana* exposed to cold stress revealed a complex network of cold-responsive genes that are regulated by various transcription factors, contributing to cold acclimation and freezing tolerance[168]. These studies demonstrate the importance of transcriptomic biomarkers in understanding plant responses to environmental stressors, which is critical for developing crops that can withstand changing climatic conditions and ensure global food security.

Statistical methods, such as differential expression analysis, network analysis, and pathway enrichment analysis, have been extensively employed to mine large-scale omics datasets for potential biomarkers. Differential expression analysis using methods such as DESeq2[145] or edgeR[169] allows for the identification of genes, proteins, or metabolites exhibiting significant changes in expression levels under different experimental conditions, thereby pinpointing potential biomarker candidates. Network analysis techniques, such as co-expression and protein-protein interaction networks, facilitate the identification of key regulatory molecules and

their functional partners, providing valuable insights into the complex relationships between biomarkers and their target pathways[170]. Pathway enrichment analysis enables the determination of overrepresented biological processes or pathways among the identified biomarkers, contributing to the understanding of their functional roles and molecular mechanisms[171].

Machine learning approaches, including supervised and unsupervised learning algorithms, have emerged as powerful tools for biomarker discovery and predictive modelling[172]. Supervised learning techniques, such as support vector machines, random forests, and neural networks, can be employed to build predictive models based on known biomarkers, enabling the classification of samples according to specific phenotypes or experimental conditions[173]. Supervised learning algorithms require labelled data, typically consisting of samples with known phenotypes or conditions, and learn a mapping between input features, such as gene expression levels, and output labels, such as stress response or disease. Once trained, these models can be used to predict the phenotype or condition of new, unlabelled samples, providing valuable insights into the underlying molecular mechanisms and the potential role of specific biomarkers in plant biology. A notable supervised machine learning network analysis method, GENIE3, combines pre-processing methods and a random forest to analyse expression datasets with the aim of predicting regulatory relationships between genes[174].

Unsupervised machine learning techniques have gained considerable attention in the analysis of high-throughput omics data, as they enable the exploration of the structure of large-scale datasets without prior knowledge of sample labels. These techniques, including clustering and dimensionality reduction algorithms, are useful for the identification of novel biomarker signatures and the discovery of previously unknown coexpression relationships between biomarkers and biological processes[175]. Clustering methods have been widely employed in plant biomarker research to reveal distinct subgroups or patterns within the data that may be associated with specific phenotypes or conditions, such as heat stress[176].

Univariate statistical methods, such as t-tests, analysis of variance (ANOVA), and the R package limma[177], have been widely employed for the identification of

differentially expressed genes, proteins, or metabolites between distinct experimental conditions. While univariate methods have proven valuable for biomarker discovery, they may be limited in their ability to model complex interactions between variables and can be sensitive to multiple testing issues, which can result in increased false discovery rates[178].

Multivariate methods can also be highly effective for biomarker discovery and data exploration, however, they also have certain limitations. For instance, multivariate statistical methods typically assume that the data follow a linear model, which may not always be the case in complex biological systems. Additionally, they can be sensitive to noise and multicollinearity in the data, necessitating careful pre-processing and model validation to ensure reliable results[179].

Feature selection techniques have become increasingly important in the analysis of high-throughput omics data, as they aim to identify a subset of relevant features, such as genes or proteins, that contribute the most information to the classification or prediction of a phenotype or condition[180]. In the context of biomarker discovery, feature selection methods have the potential to reduce the complexity of large-scale datasets, improve the interpretability of results, and ultimately facilitate the identification of robust and biologically meaningful biomarkers[181].

1.1.5 Computer Vision: An Overview

Computer vision, a subset of computer science that focuses on algorithms analysing and extracting information from images and visual data, has experienced rapid advancements in recent years, demonstrating immense potential in a wide range of applications across various domains. This progress can be attributed to the development of deep learning algorithms and the availability of large, annotated datasets, which have significantly enhanced the extraction of meaningful information from visual data, such as images and videos[182]. In this subsection, I aim to explore the potential and recent advances in computer vision approaches, with a focus on different computer vision tasks, associated methods, and their applications to plant biology.

A core task in computer vision, image classification, entails assigning a label to an image according to its content. Deep learning methods, particularly CNNs[183], and more recently, vision transformers (ViTs)[184], have revolutionised image classification by achieving unparalleled performance on benchmark datasets like ImageNet[185]. ImageNet is a large-scale dataset consisting of over 14 million images typically cropped to 224x224 pixels with 20,000 classes that has become the standard benchmark dataset for image classification tasks.

CNNs and ViTs directly learn hierarchical feature representations from raw image data, thereby eliminating the need for manual feature engineering, which was a significant bottleneck in conventional computer vision techniques[186]. CNNs are a type of neural network specifically designed for processing data with spatial relationships, such as images or time series. They are constructed with a hierarchy of convolutional, batch normalisation, pooling, and fully connected layers. The convolutional layers apply sets of learnable filters or kernels onto the input data, capturing local patterns and spatial features, including edges, textures, or even more complex shapes when deeper into the network. The batch normalisation process standardises the distributions of these feature responses, improving the network's learning speed and overall performance. Pooling or subsampling layers, typically applied after convolutional layers, reduce the spatial dimensions of the feature maps, which reduces computational complexity and enhances model generalisation. The fully connected layers act as an integrator of the learned feature maps, typically placed near the end of the network, converting the spatial features into flattened form, to output predictions. Through this hierarchical structure, CNNs can learn increasingly complex and abstract features from the input data, enabling them to achieve breakthrough performances in image classification tasks.

Vision transformers (ViTs), on the other hand, utilise the transformer architecture, originally developed for natural language processing tasks, to process images. ViTs divide an image into non-overlapping patches, which are then linearly embedded into a sequence of fixed-size vectors. Positional encoding is added to the sequence to provide spatial information, and the sequence is then fed into a transformer model. The transformer uses self-attention mechanisms to capture global dependencies and

interactions among the input patches, allowing the model to learn contextual information and high-level features. ViTs have demonstrated impressive performance on image classification tasks, rivalling or surpassing CNNs in some cases[187].

Cutting-edge neural network architectures have emerged in recent years, pushing the boundaries of performance and efficiency of model parameters. One of the key advances is ResNet[188] (Residual Network), a deep CNN architecture that introduces skip connections, or residual connections, to enable the efficient training of much deeper networks without suffering from vanishing gradients. Other advances include EfficientNet[189], a family of CNN models that use compound scaling to balance model depth, width, and resolution, which have achieved state-of-the-art performance despite reduced computational complexity.

Object detection, another critical computer vision task, seeks to identify and localise multiple objects within an image and assign appropriate labels to them. Advances in object detection have been driven by the development of region-based CNNs (R-CNNs)[190] and single-stage detectors, such as the YOLO (You Only Look Once)[191] models. These methods have demonstrated remarkable success in detecting and localising objects in complex scenes while maintaining real-time processing capabilities. In recent years, these methods have continued to improve object detection performance. For example, two-stage detectors, like Faster R-CNN[192], combine region proposal networks (RPNs) with R-CNNs to create a more efficient and accurate object detection pipeline. For single-stage detectors, neural networks such as YOLOv4[193] and YOLOv5[194] have further improved upon the original YOLO framework with better accuracy and speed. The performance of these methods is typically benchmarked using the intersection over union (IoU) metric that calculates the ratio between the area of overlap and the area of the union of the predicted bounding box and the ground truth bounding box.

Semantic segmentation is an image analysis task that aims to assign a class label to each pixel in an image, providing a comprehensive understanding of the image at the pixel level. This task has greatly benefited from recent developments in neural network architectures, specifically fully convolutional networks[195] (FCNs) and

encoder-decoder architectures. FCNs have replaced the fully connected layers in traditional CNNs with convolutional layers, allowing them to produce dense predictions for each pixel in an input image. Encoder-decoder architectures, such as U-Net[196] and SegNet[197], combine a downsampling path that captures the context of the input image and an upsampling path that produces a dense output of pixel-wise class labels. These methods have found applications in various domains, including autonomous driving, medical imaging[196], and crop phenotyping[198]. As well as FCNs, cutting-edge methods such as deep feature fusion have been developed to exploit the complementary strengths of different architectures to improve segmentation accuracy by training and combining multiple encoder-decoder architectures. Another category includes attention-based models[199], which strategically focus on the most relevant regions of the input image for segmentation, enabling them to handle complex scenes with multiple objects and varying scales.

An important step in computer vision pipelines is image processing, as it helps to improve the quality of visual data and enhance the performance of subsequent analysis tasks. Standard pre-processing techniques include noise reduction, edge detection, and feature extraction methods, which help to enhance image quality and facilitate the subsequent analysis tasks[200]. Noise reduction techniques, such as Gaussian smoothing and median filtering, can be employed to remove noise and artefacts from images while preserving the essential details and structures[201]. Edge detection methods, such as the Canny, Sobel, and Laplacian of Gaussian (LoG) operators, are used to identify boundaries between different regions within an image, which can be useful for feature extraction and object recognition tasks[202].

Image augmentation is a critical pre-processing technique used in deep learning for computer vision tasks. Its purpose is to increase the diversity and size of the training dataset by applying various transformations, such as rotation, scaling, flipping, and changes in brightness or contrast. These transformations simulate different variations of the input data, thereby expanding the available data and improving the generalisation capabilities of deep learning models[203]. One of the main benefits of image augmentation is its ability to mitigate the risk of overfitting, which is particularly important in scenarios where the available data is limited or imbalanced. Overfitting occurs when a model becomes too specialised to the training data and

performs poorly on new, unseen data. Image augmentation helps to prevent overfitting by introducing additional variations to the training data, making the model more robust to unseen data. Several studies have shown the effectiveness of image augmentation in improving the performance of deep learning models. For example, Wang et al.[203] applied random scaling, rotation, and colour jittering to the CIFAR-10 and CIFAR-100 datasets and demonstrated significant improvements in model performance.

An area of computer vision that has recently garnered significant attention is self-supervised learning, a type of unsupervised learning that involves training models to solve tasks on unlabelled data, with the aim of learning representations that can be transferred to downstream tasks. An example would be learning to maximise the agreement between the representations of different views of the same image. The learned representations can then be fine-tuned on a smaller amount of labelled data to achieve state-of-the-art performance on downstream tasks, such as object recognition, semantic segmentation, and image classification. One of the leading architectures in self-supervised learning is the DINO (Emerging Properties in Self-Supervised Vision Transformer)[204] method proposed by Caron et al., which along with other self-supervised learning methods such as SWAV[205] and SimCLR[206], has been shown to outperform pre-trained supervised ImageNet architectures, especially when trained on smaller datasets[207].

As well as supervised and unsupervised learning, semi-supervised learning lies in between, as it involves training models on a combination of labelled and unlabelled data, with the aim of leveraging the unlabelled data to improve model performance. This approach is particularly useful in scenarios where labelled data is limited or expensive to obtain. One variation of semi-supervised learning in computer vision is the use of generative models, such as generative adversarial networks[208] (GANs) and variational autoencoders[209] (VAEs), to generate additional unlabelled data. This approach has been used to improve model performance in a range of tasks, including image classification, object detection, and semantic segmentation[210]. Overall, the use of semi-supervised and self-supervised learning methods is expected to play an increasingly important role in advancing computer vision research by exploiting unlabelled data to improve the performance of models for a variety of tasks.

In the coming years, computer vision will benefit from integrating knowledge from fields like natural language processing, robotics, and neuroscience, leading to more efficient and robust systems. Advances in these areas, combined with novel research directions in self-supervised learning and the development of interpretable models using techniques such as attention mechanisms and saliency maps, will deepen the understanding of visual perception and optimise computer vision algorithms. Addressing the interpretability challenge, researchers are exploring explainable AI techniques to build trust in model predictions[211], particularly in safety-critical applications. The result of these developments will enable computer vision systems to effectively process visual data and facilitate a wide range of research and applications.

1.1.5.1 Computer Vision in Plant Biology

Computer vision techniques have significantly impacted plant biology research, allowing for the automated analysis of plant images and videos. These non-invasive methods provide high-throughput and quantitative data extraction from plant phenotypes, furthering the understanding of plant growth, development, and response to environmental factors[212], and playing a critical role in plant breeding and precision agriculture[213].

Image-based plant phenotyping, a crucial application of computer vision in plant biology, allows for the quantification of various phenotypic traits, such as growth, leaf area, and biomass accumulation[214]. Researchers have employed several image processing techniques, including thresholding, edge detection, and feature extraction, to segment plant structures and extract relevant phenotypic traits and features from images[215]. Deep learning methods, particularly CNNs, have been utilised to improve the accuracy and robustness of plant phenotyping tasks, such as leaf counting, organ segmentation, and disease detection[216].

A notable image-based plant phenotyping case study is the work of Mohanty et al.[217], which demonstrated the effectiveness of CNNs in classifying plant diseases. In

this research, the team assembled a dataset of over 50,000 images representing 14 crop species and 26 diseases. These images were sourced from various repositories, agricultural websites, and expert contributions to ensure a diverse dataset. The dataset was then partitioned into training, validation, and testing subsets to facilitate the development and evaluation of the CNN model. CNN architectures based on AlexNet and GoogleNet were applied, each being comprised of multiple convolutional layers, pooling layers, and fully connected layers. When evaluated using the hold-out test set, the GoogleNet CNN demonstrated the highest overall accuracy of 99.35% in identifying crop species and diseases. A key point to consider is the fact that training the supervised CNN to achieve 99.35% accuracy necessitates hours of GPU-intensive processing. However, making a prediction on a new image requires only a small amount of CPU power; enabling models like these to be implemented on smartphones, democratising the use of AI for plant phenotyping in regions with limited access to powerful computing facilities.

Furthermore, computer vision techniques can offer valuable insights into plant growth analysis. By tracking plant growth over time, quantifying growth rates, and identifying growth patterns, these approaches have proved to be essential tools[218]. Time-lapse analysis, tracking algorithms, and machine learning techniques, including CNNs and unsupervised learning methods, have been employed for this purpose. These automated and high-throughput approaches have been applied to analyse plant growth image datasets to provide detailed information on plant development and key growth stages such as flowering time[219].

In 2020, the Global Wheat Head Detection (GWHD) dataset was generated, comprising 193,364 labelled wheat heads from 4,700 RGB images sourced from various countries and institutions[220]. After being made available through a supervised machine learning competition on the website Kaggle, a more extensive dataset that contained an additional 1,722 images from 5 countries with over 80,000 wheat heads was published in 2021 and made available to the data science community[221].

In a study that utilised the GWHD, Wen et al.[222] present a highly efficient RetinaNet[223], named SpikeRetinaNet, to detect and count wheat spikes, a crucial

agronomic trait associated with wheat yield. The accurate detection and counting of wheat spikes have long been challenging due to the complex field conditions in wheat cultivation. The authors of this study addressed these challenges by introducing several optimisations to the RetinaNet. The first improvement involved the integration of a weighted bidirectional feature pyramid network (BiFPN) into RetinaNet's feature pyramid network (FPN). This enhancement allowed the network to fuse multiscale features and recognise wheat spikes across different varieties and complex environments. Secondly, the researchers added focal loss and attention modules to enhance the detection efficiency of objects. Lastly, they employed soft non-maximum suppression (Soft-NMS) to resolve occlusion issues. The proposed SpikeRetinaNet achieved a mean average precision of 0.9262, outperforming the state-of-the-art RetinaNet, You Only Look Once version 4 (Yolov4), and Faster-R-CNN models.

Semantic segmentation methods have been widely used in plant phenotyping for detecting and segmenting different plant organs with high accuracy. Yang et al.[224] proposed a method using Mask R-CNN and VGG16 for leaf segmentation when multiple leaves overlap and images have complicated. More than 2,500 leaf images with complicated backgrounds were collected and artificially labelled with target pixels (representing leaves) and background pixels. Out of these images, 2,000 were used to train a Mask R-CNN model for leaf segmentation. Following the segmentation step, a training set containing over 1,500 images of 15 different species was fed into the VGG16 network to train a model for leaf classification. To optimise the performance of both models, the authors compared various parameter combinations to identify the best hyperparameters. The results revealed that the average misclassification error (ME) of 80 test images using Mask R-CNN for leaf segmentation was 1.15%, and the average accuracy for leaf classification on 150 test images using VGG16 reached 91.5%. This paper demonstrates the potential for combining deep learning models such as Mask R-CNN and VGG16 to address challenges related to plant image analysis, segmentation, and classification. Compared to the size of the dataset, the use of 80 and 150 test images may be insufficient to confidently evaluate the generalisation error of the trained computer vision models.

In conclusion, computer vision approaches have shown great potential and recent advances in plant biology research. They offer automated, high-throughput, and quantitative means for analysing plant images and videos, enabling researchers to measure and extract information from plant phenotypes.

1.2 Motivation for the Research

The general motivation for my research concerns several factors that emphasise the need for interdisciplinary approaches in plant biology to address global challenges. These factors, which collectively drive my application of machine learning and computer vision methods in plant biology, include:

1. **Food security and increasing global population**: With the world population projected to reach nearly 10 billion by 2050 [225], there is an urgent need to increase agricultural productivity to meet the rising demand for food. Solutions that can optimise crop growth and mitigate crop diseases early are essential for ensuring food security.
2. **Climate change and environmental pressures**: An increasingly volatile climate poses significant threats to agricultural productivity through extreme weather events and increased temperatures[226]. Computational methods that can use machine learning to accelerate the development of climate-resilient crop varieties and our understanding of plant responses to environmental stressors are desperately needed.
3. **Advancements in technology and data generation**: The rapid growth of high-throughput technologies, such as next-generation sequencing and high-resolution imaging, has resulted in the creation of massive amounts of data in plant biology[227,228]. To keep up with the unprecedented rate at which data is being generated, machine learning and computer vision methods must be developed to analyse and interpret these vast datasets.
4. **Potential for discovery in complex systems**: There remains a knowledge gap in understanding the complex processes that control physiological processes and responses to environmental stimuli due to complicated regulatory pathways and systems. It might be possible to further our understanding of these processes by utilising recently developed machine learning methods.

As well as these overarching motivations for my research, I will now describe the specific motivations for each of the three projects.

1.2.1 Seed Germination

Due to the importance of seed germination in agricultural research and crop production, accurate and efficient scoring of seed germination and germination-related traits is essential for monitoring and breeding new crop varieties[229]. Traits such as germination uniformity, germination under stress, germination rate, and speed of germination are of high importance to breeders for developing new crop varieties[230].

Manual scoring of seed germination is a tedious and time-consuming process that requires trained personnel to observe and count each individual seed that has germinated. This process is prone to errors and inconsistencies, as the scorers can make mistakes in counting due to numerous factors, including a subjective interpretation of germination criteria, eye strain and physical discomfort from prolonged periods of work, and biases towards certain seeds or conditions. Furthermore, the traditional methods of manual seed germination scoring are unsuitable for large-scale experiments, where thousands of seeds need to be scored within a short period of time. Given these challenges, there is a growing need for consistent, automated, and efficient methods of seed germination scoring.

1.2.2 Plant Pathogens Biomarkers

The impact of plant pathogens on agriculture necessitates the development of tools to diagnose infections and identify associated biomarkers. Plant diseases caused by pathogens can significantly reduce crop yield and quality, leading to food shortages and economic losses. Moreover, as climate change alters temperature and precipitation patterns, it may also influence the prevalence and distribution of plant pathogens[231]. Therefore, it is essential to predict their potential impacts to prevent devastating consequences. As well as the importance of our capability to detect pathogens, identifying biomarkers associated with plant pathogen infections is crucial for developing crop varieties that have disease-resistant traits.

The development of tools to diagnose plant pathogen infections and identify associated biomarkers is essential for minimising the impact of plant diseases on agriculture. In addition to developing new tools, understanding the underlying molecular mechanisms involved in pathogen-host interactions is crucial to generating new methods for controlling plant diseases and breeding disease resistant crops.

1.2.3 The Circadian Clock

Studying the circadian clock in plants is motivated by several factors that are important for agriculture and our overall understanding of biological processes. For example, using our understanding of the circadian clock to optimise the timing of processes such as photosynthesis[232], it might be possible to optimise growth conditions leading to improved water usage, nutrient absorption, crop yields, and other agriculturally important traits.

The circadian clock also regulates stress responses in plants[233], helping them survive various environmental challenges such as drought, extreme temperatures, and pathogen attacks. Knowledge of these mechanisms could aid in breeding or engineering plants with improved stress tolerance, contributing to a more resilient agricultural system. Moreover, with an increasingly volatile climate, it is important that we understand how circadian rhythms assist plants in adapting to their environment and synchronising their activities with the day-night cycle. A deeper understanding of these processes could inform strategies for adapting plants to new environments.

Studying the circadian clock comes with significant challenges due to the inherent complexities and costs associated with generating time-course datasets, particularly RNA-Seq time-course datasets. The circadian rhythm operates on a roughly 24-hour cycle, necessitating the collection of numerous samples at consistent intervals over an extended period to capture the dynamics of gene expression. Moreover, the need for replicates to ensure statistical power and the high variability between individuals further exacerbate the costs and resources required for these studies.

1.3 Scope of the Research

The scope of this thesis encompasses three separate projects, each with a distinct focus but with the common goal of leveraging machine learning and computer vision techniques to address key challenges in plant biology:

SeedGerm: The goal of the SeedGerm project was to develop a robust and accurate computer vision software system in Python that is capable of detecting seed germination and measuring morphological seed traits. The software should generalise to multiple crop species, so the datasets provided by my iCASE sponsor, Syngenta, to train and benchmark the model contained over 1000+ images of five crops: maize, tomato, pepper, barley, and Brassica. The system's germination predictions were to be compared with germination scoring performed by specialist seed technicians to benchmark its efficacy. A strong correlation between germination predictions from the SeedGerm system and the specialist seed technicians' scores would indicate that it is possible to accurately detect seed germination using computer vision and machine learning techniques. By developing an open-source software system, my aim was to enable researchers to perform high throughput germination phenotyping for studies that could identify environmental and genetic factors affecting germination traits.

Trans-Learn: In the Trans-Learn project, I aimed to explore the potential of applying image analysis models to tabular gene expression datasets in a case study to predict turnip mosaic virus (TuMV) infections in wild *Arabidopsis halleri* samples. By transforming the gene expression tabular data into an image format, I planned to test whether image analysis methods could outperform standard supervised ML models for tabular data and whether the arrangement of the features in the image affected the performance of the model. I set out to analyse thousands of gene expression samples while developing the pipeline and benchmark many supervised machine learning models by evaluating the performance of predictions made on hold-out test datasets. By reverse engineering the image analysis models, I attempted to identify sets of multivariate biomarkers for turnip mosaic virus and other pathogens. If the Trans-Learn software can effectively identify biomarkers and

accurately diagnose TuMV, this open-source research tool would have the potential to be applied to search for biomarkers associated with other traits and develop models to predict these traits.

ChronoGauge: To test the hypothesis of whether it was possible to accurately predict the internal circadian time of plants using gene expression data, I aimed to develop a pipeline that combines statistical methods for identifying rhythmically expressed genes and a supervised machine learning model for generating predictions. Five publicly available *Arabidopsis thaliana* gene expression datasets and two internal *Triticum aestivum* (wheat) gene expression datasets were selected to develop the pipeline. The accuracy of the model was to be assessed by comparing the predictions with the sampling times of the datapoints. If the model was able to predict the time of sampling accurately using gene expression data, then it would be possible to estimate the internal circadian time of a plant, assuming the plant's internal clock is synchronised with the external times. The pipeline that I aim to develop would be open-source and enable researchers to screen genetically different plants for variation in circadian rhythms, enabling the identification of genetic and environmental factors affecting the circadian clock.

This thesis aims to showcase the potential of interdisciplinary research in tackling critical challenges within plant biology by integrating the latest advancements in computer vision, machine learning, and genomics across three distinct projects. The overarching objective of this research is to further the progress of plant biology by offering insights into the potential applications of these methodologies in the field and providing open-source software tools for the analysis of large biological datasets.

1.4 Significance of the Research

The significance of this research lies in its potential to enhance plant biology research through the development of multiple machine learning and computer vision tools. By addressing various challenges in the field, the outcomes of this research are expected to have wide-ranging implications, including:

My development of machine learning and computer vision tools can assist in providing novel insights into complex plant processes and analysing vast amounts of data. Through scientific researchers applying the outputs of my research, future discoveries in plant biology relating to plant development and phenotypic traits can be accelerated. By leveraging machine learning and computer vision techniques, this research can contribute to a better understanding of the factors affecting seed germination. These insights can help optimise growth conditions, improve seed quality, and lead to increased crop productivity.

The ability to accurately predict complex crop traits, estimate seed germination, and identify biomarkers can significantly accelerate crop breeding and the development of new plant varieties. These advancements can lead to the creation of crops with improved yield, stress tolerance, nutrient-use efficiency, and resistance to diseases, ultimately contributing to increased agricultural productivity and global food security. For example, the early detection of plant diseases is crucial for mitigating their impact on crop yields and food security. My research could contribute to the development of novel diagnostic tools that enable the early identification of diseases, allowing for timely interventions and reduced losses in agricultural productivity.

The research in this thesis highlights the immense potential of interdisciplinary collaboration between plant biology, machine learning, and computer vision. By bridging the gaps between these fields, this work can inspire future research and innovation, leading to the development of new methodologies and approaches that can address key challenges in plant biology.

In summary, the significance of this research lies in its potential to accelerate plant biology research and agricultural productivity by leveraging the power of machine learning and computer vision. Therefore, the insights gained from this research can ultimately help address global challenges related to food security, climate change, and sustainable agriculture.

Chapter 2: SeedGerm – Machine Learning Based Phenotypic Analysis of Seed Germination

2.1 Introduction

In this chapter, I present the development and implementation of a machine learning-based computational pipeline, named SeedGerm, for the automatic detection of seed germination in images. This project was a collaborative effort between the Earlham Institute, where the software and hardware were developed, Syngenta, my iCASE industry sponsor that generously provided some of the image datasets, and the John Innes Centre, who provided additional datasets and conducted manual seed scoring and biological interpretation of the phenotypic predictions.

My contribution to this project involved the development of the software, including image pre-processing, panel and seed segmentation, morphological feature extraction, morphological trait analysis, hyperparameter tuning of the classification algorithm, and development of the user interface. The SeedGerm system was codeveloped with colleagues at the Earlham Institute who contributed their expertise in software engineering and designed and constructed SeedGerm's hardware. Collaborators at the John Innes Centre and Syngenta generated and labelled the datasets, as well as performed the genome-wide association study on the Brassica lines that were phenotyped.

To provide the necessary context for this chapter, I will first provide a focused literature review on seed germination traits and existing methods of seed phenotyping. Parts of this chapter have been adapted and modified with permission from my first author publication in New Phytologist: Colmer et al. (2020) "SeedGerm: A Cost-Effective Phenotyping Platform for Automated Seed Imaging and Machine-Learning Based Phenotypic Analysis of Crop Seed Germination."

2.1.1 Abstract

Efficient germination and establishment of seeds are crucial traits for crops grown in both fields and glasshouses. Large-scale germination experiments can be labour-intensive and susceptible to human errors, necessitating automated techniques. A method for comprehensive germination scoring was developed in this research, which involved five crop species, including tomato, pepper, Brassica, barley, and maize.

Here, I introduce the SeedGerm system, which integrates cost-effective hardware with open-source software for conducting seed germination experiments, automating seed imaging, and performing machine learning based phenotypic analysis. Multiple image series can be processed simultaneously by the software, and accurate analyses of germination- and establishment-related traits can be recorded in both comma-separated values (CSV) and processed images (PNG) formats.

In this chapter, a detailed description of the computational pipeline is provided. The ability of SeedGerm to match expert scoring of radicle emergence with Pearson's correlation coefficients exceeding 0.99 across multiple crops is also demonstrated. Germination curves were generated based on individual seed germination timing and rates rather than relying on a fitted curve, with morphological traits also tracked. A gene crucial in abscisic acid (ABA) signalling in seeds was identified using phenotypic predictions generated by SeedGerm through a GWAS across a diverse set of *Brassica napus* varieties.

Upon comparison with existing techniques, SeedGerm holds potential for widespread use in large-scale seed phenotyping and testing, suitable for both research and routine seed technology applications.

2.1.2 Seed Germination and Vigour

Seeds play a vital role in human life, serving not only as significant sources of nutrition but also as the foundation for effective crop production. Seed germination

is a highly regulated process dictated by a complex interplay of genetic and environmental factors such as temperature, moisture, light, and pathogens. The vigour of seeds, characterised by superior germination and seedling emergence rates, is crucial for ensuring consistent emergence across diverse agricultural environments, ultimately contributing to a crop's yield potential and uniformity[234]. Seed vigour is therefore an essential quality to evaluate when aiming for optimal agricultural outcomes, and understanding its genetic basis and interaction with environmental stimuli is important for breeding crop varieties with improve germination traits.

One widely employed method for gauging seed germination is the examination of radicle protrusion, a metric that evaluates both the rapidity and frequency with which seeds germinate[235]. Radicle protrusion is an early and visible indicator of seed germination, as it involves the emergence of the primary root from the seed coat. Assessing this parameter allows for a more in-depth understanding of a seed's germination potential and vigour. Historically, the task of evaluating seed germination through radicle protrusion was carried out by seed technologists, who conducted visual inspections of colour and morphological changes during the various physiological stages of seed germination[236]. These professionals relied on their expertise and experience to identify and assess the germination process as it unfolded, making judgments based on the observed characteristics of the seeds.

However, this traditional approach to assessing seed germination has its limitations, as it can be both labour-intensive and subjective. The reliance on human observation and judgment introduces a degree of variability and potential bias, which could impact the accuracy and consistency of the results. Additionally, the manual nature of this method can be time-consuming and requires a high level of expertise, making it less efficient and potentially less accessible to those involved in seed evaluation and agricultural production.

In summary, seed germination and vigour are critical aspects of crop production, with high-vigour seeds demonstrating better germination and seedling emergence rates, which ultimately contribute to improved yield potential and uniformity. The traditional method of assessing seed germination through radicle protrusion, though

valuable, has been limited by its labour-intensive and subjective nature. As such, there is a growing need for more objective, efficient, and accurate methods to evaluate seed germination and vigour in order to optimise agricultural outcomes.

2.1.3 Seed Phenotyping Methods

Seed phenotyping is the process of quantitatively characterising the physical and physiological traits of seeds to gain insights into their quality, performance, and potential for crop production. The process of routine seed germination phenotyping often depends on human observation, which practically limits the frequency, scale, and precision of such tests. This limitation has prompted numerous efforts to automate seed imaging and related phenotypic evaluations, giving rise to research-based solutions like GERMINATOR[237], the phenoSeeder[238] system, and the MultiSense tool[239]. In more recent times, sophisticated computer-vision and ML approaches have been utilised for germination assays, which include the Rice Seed Germination Evaluation System for determining the germination status of Thai rice species using an artificial neural network (ANN) classifier[240]; machine-vision-based examination of visible and X-ray images for appraising soybean seed quality based on physical purity, viability, and vigour[241]; deep learning algorithms such as U-Net[196] and ResNet[188] for segmenting and categorising rice seed germination status[242]; linear discriminant analysis and multispectral imaging combined for classifying cowpea seeds into categories of aging, germination, and normality[243]; and a high-throughput micro-CT-RGB phenotyping system for dissecting the rice genetic architecture from seedling[244].

These solutions encompass customised hardware devices (e.g., unique germination trays, image sensors, and seed management systems) and specialised analytic software based on Matlab Toolbox, ImageJ/Fiji, Microsoft Excel macros, image analysis libraries (e.g., VideometerLab3 and OpenCV), and ML/DL libraries (e.g., PyTorch). While not entirely automated, they have been effectively employed to extract germination traits from the captured seed images, such as morphological attributes (e.g., size and shape), cumulative germination rates (e.g., time to 50% germination, T50, and the percentage of seeds germinated at the end of an

experiment, G_{max}), and quality attributes like viability and vigour[245,246]. However, the throughput, degree of automation, and the variety of traits of the aforementioned solutions remain restricted, meaning that seed imaging and related germination-linked trait analyses still necessitate human intervention.

The rise of plant phenomics in recent years has introduced new outlooks to seed science research[247]. By integrating cost-effective digital imaging and environmental sensors, organ-level plant growth and development can be documented with detailed images at an extremely high frequency[248]. Specifically, numerous analytic techniques have been developed to enable the automation of organ-level phenotypic evaluation, including leaves, roots, and reproductive organs[249]. By collating colour, texture, morphology, and growth patterns information, seed germination can be assessed in a dynamic and objective way, allowing for the generation of large-scale and reproducible evidence that can facilitate new biological discoveries in seed physiology[250]. Moreover, automating seed germination scoring offers an excellent opportunity to initiate the standardisation of seed science research. This not only enables digital evaluation of seed quality and vigour, but also allows for the quantitative cross-referencing of biological experiments under various conditions, enhancing the reliability of research findings.

Alongside the progress seen in automated phenotyping, advances in molecular techniques, such as DNA and RNA sequencing, as well as metabolomics, have greatly impacted seed trait analysis at the molecular level in recent years. These methods provide crucial insights into gene expression, metabolism, and genetic variation linked to seed quality and performance, enabling the identification of key genes and their functions.

For instance, Baud et al.[251] discovered the WRINKLED1 (WRI1) gene as a crucial regulator of oil accumulation in Arabidopsis seeds, contributing to a better understanding of seed oil content regulation in plants and offering potential applications for improving oilseed crops. Tang et al.[252] identified genes controlling seed size and weight in rice using GWAS, finding 16 loci linked to these traits, which proved valuable for breeding programs aiming to increase rice yield. In legumes, Liu et al.[253] combined QTL mapping and RNA-Seq to identify the

GmSALT3 gene, responsible for seed yield and size in soybean under saline conditions, assisting in the development of salt-tolerant soybean varieties.

These methods allow for the quantitative characterisation of various seed traits, offering valuable information on seed quality, performance, and crop production potential. Technological advancements are expected to continue enhancing seed phenotyping's accuracy, efficiency, and scalability, contributing to improved crop breeding and production practices. Computer vision methods, in particular, present promising opportunities for seed phenotyping in germination studies, enabling automated and objective quantification of traits related to germination potential. This significantly accelerates and enhances seed phenotyping, providing valuable insights into the molecular mechanisms and markers controlling agronomic traits.

2.2 Aims and Objectives

The aim of the SeedGerm project was to develop a comprehensive phenotyping system for seed germination using computer vision and machine learning methods to extract morphological features from segmented seeds and produce germination predictions. The primary objectives of this project are as follows:

The first objective of this project was to develop open-source software using the Python programming language, which incorporates various computer vision and machine learning techniques for seed phenotyping. The software should provide a user-friendly interface for inputting seed images, segmenting seeds from the background, extracting relevant morphological features, and generating germination predictions based on machine learning models.

The next objective was to develop robust and accurate seed image segmentation algorithms as part of the software to automatically separate seeds from the background in the input images. This will involve implementing image processing techniques such as thresholding, edge detection, and machine learning-based approaches to accurately and efficiently segment seeds, even in the presence of variations in seed appearance, background, and lighting conditions.

Once the seeds were segmented, my objective was to develop feature extraction methods to quantify relevant morphological features from the segmented seed images. This involved designing and implementing algorithms to extract features such as seed size, shape, colour, and statistical image measurements, which could be indicative of seed germination potential. These features would serve as inputs for machine learning models for germination prediction.

After extracting morphological features, the next objective was to develop machine learning models for germination prediction using these features as input variables. Using the morphological features of ungerminated seeds, I explored the application of unsupervised methods to classify seeds based on their morphological features. The developed models were trained and tested using a dataset of annotated seed images

with corresponding germination outcomes, and the predictive accuracy and robustness of the models was evaluated.

My final objective was to thoroughly evaluate and validate the developed phenotyping system. This involved conducting experiments with a diverse set of seed species to assess the performance of the system in different scenarios. The system's predictions were compared with ground truth labels recorded by seed screening experts and evaluated in terms of its accuracy and usability.

In addition to evaluating and validating the SeedGerm system, there was an objective for my collaborators at the John Innes Centre to use the system to screen varieties of *Brassica napus* for germination-related traits. Once germination traits had been predicted across Brassica lines, they aimed to use a GWAS to identify DNA markers associated with seed germination traits.

Overall, the SeedGerm project aims to develop an automated phenotyping system for seed germination that is accessible to the seed phenotyping community as open-source Python software. The proposed system has the potential to significantly enhance and accelerate the process of seed phenotyping, providing valuable insights into seed germination behaviour and contributing to crop breeding and production practices.

Although the primary goal of this project is not an in-depth biological exploration of a specific phenomenon, an appropriate hypothesis for this study could be: "The developed phenotyping system, utilising computer vision and machine learning methods for extracting morphological features from segmented seeds, will accurately predict seed germination, demonstrating its potential as an efficient and reliable tool for seed germination phenotyping in agricultural crops."

This hypothesis assumes that the phenotyping system being developed will be able to accurately predict the germination outcomes of seeds based on the extracted morphological features. The hypothesis suggests that the combination of computer vision and machine learning methods will result in a robust and efficient system that can provide reliable germination predictions, contributing to the field of seed

phenotyping in agricultural crops. The hypothesis will be tested through experiments and validation processes using numerous seed species with diverse morphologies to assess the accuracy and reliability of the system’s predictions.

2.3 Methods

In this section, I present the methods that I developed and applied to investigate seed germination and morphology across various crop species. Detailed procedures for seed production, storage conditions, experimental setups, and automated seed imaging are outlined to provide a comprehensive understanding of the techniques used. Additionally, specific germination conditions and evaluation criteria for each crop species are described, along with the experimental design implemented to ensure reliable and reproducible results. This section serves as a guide for interpreting and replicating the findings of this study. The primary components of the SeedGerm methodology include the segmentation of the background panels, the segmentation of seeds, morphological feature extraction from segmented seeds, germination classification, and the quantification of seed germination traits over the progression of an image series (Figure 1).

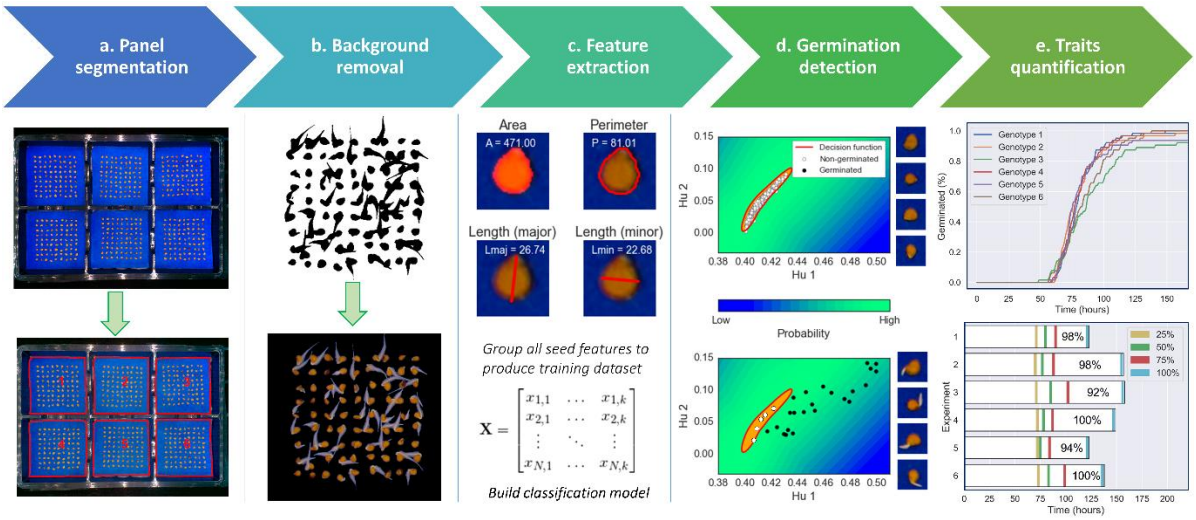

Figure 1.

The workflow of the SeedGerm analysis software, enabling automated seed germination scoring and phenotypic analysis. (a) YUV colour-space panel segmentation. (b) Seed-related objects are identified and retained using trained background removal models. (c) Seed morphological features are measured and combined into training matrices. (d) A one-class SVM model, trained on these vectors, classifies each seed's germination status in every image. (e) Germination scoring and morphological traits are aggregated to generate cumulative germination curves and timing plots.

2.3.1 Seed Lot Production and Storage

Seed batches were created in industrial manufacturing and preserved at 12°C and 35% relative humidity until needed. The 88 B. napus Diversity Fixed Foundation Set (DFFS) lines in this research were utilised to generate seeds by vernalising plants (8-hour photoperiod, 5°C) for six weeks at the four-leaf stage and cultivating them in a polytunnel. Within three months of being collected, seeds were used. Biological replicates consisted of seed groups from distinct parent plants. Top-quality seed collections of tomato and Brassica were used to produce lower quality sets, with a portion being heat-treated at 70°C for three days.

2.3.2 Seed Germination Conditions

A standard experimental arrangement involves A3-sized filter paper, dark blue seed examination paper (Grade 194, Bärenstein Germany) provided by Munktell Ahlstrom, placed in germination chambers. This setup accommodates six groups of 64 individual seeds (384 seeds altogether, in six germination panels) for tomato and Brassica seeds. For barley, experiments were conducted using three expanded germination panels, with 40 seeds per panel and 120 seeds in total. Due to maize seed size, the entire germination container accommodated 35 seeds per trial. For pepper seeds, 81 seeds were placed in a single panel, totalling 486. To enable accurate germination categorisation, a minimum of A4-sized filter paper is suggested to provide adequate space between seeds. Additionally, further divisions can be made to separate different genotypes.

Automated seed imaging typically occurred at one-hour intervals and usually lasted between 5 and 10 days, depending on the plant species. For instance, *Brassica napus* seeds germinated on saturated filter paper in SeedGerm containers under continuous white light at 10°C (either in a cold room or a growth chamber). Standard seed testing took 7-14 days, with germination frequency and seed vigour being the two primary traits regularly assessed by skilled seed technologists. To examine the 88 *Brassica napus* DFFS lines, seeds were arranged in 50-seed grids with six panels per

germination container and five replicates per line, following a completely randomised experimental design. In a typical experiment, each SeedGerm container contained two layers of white filter (Grade 3644, Hahnemuehle Germany) and one sheet of blue seed germination paper on top. A set volume of water (e.g., sterile de-ionized water, 350ml) was added to the filter paper stack before the experiment began. To ensure uniform absorption throughout the filter paper, the moistened paper was allowed to rest for 2 hours after water addition (i.e., another 30ml), prior to positioning the seeds and initiating the experiment.

2.3.3 Image Processing and Panel Segmentation

The pipeline starts by loading a series of RGB images obtained from a seed germination experiment, with these images serving as the primary data for further analysis. Due to the setup of the germination experiments that consist of multiple pieces of germination paper with seeds arranged on top, computer vision methods must first be employed to identify each germination panel and separate them from the background of the image.

To facilitate the differentiation of the germination paper from the background of the images, and the seeds from the germination paper, the colour space was transformed from RGB to YUV. This transformation assists with making distinct and linearly separable pixel groups for the following three different objects in the image: background, germination paper (panels), and seeds. The YUV colour space is a representation of colours in terms of luminance (Y) and chrominance (UV) components, which separates brightness information (Y) from colour information (U and V). The YUV colour space is related to the RGB (Red, Green, Blue) colour space, which is the standard representation of colours in digital images. The YUV colour space can be derived from the RGB colour space through a linear transformation, which involves specific weighting factors for each colour component. The conversion from RGB to YUV is given by the following equations:

$$Y = 0.299R + 0.587G + 0.114B$$

$$U = -0.147R - 0.289G + 0.436B$$

$$V = 0.615R - 0.515G - 0.100B$$

In the context of image segmentation and enhancing contrast between objects, the YUV colour space can be advantageous compared to the RGB colour space. As the Y component represents luminance, it can help emphasise the brightness differences between objects, making it easier to distinguish them based on their brightness levels. The U and V components contain colour information, which can be useful in identifying and segmenting objects with distinct chrominance values.

After transforming the colour space of the RGB images to YUV, the user then defines three threshold values, one for each channel, to separate panels from the background and seeds from panels. Here, the luminance channel (Y) is particularly effective at separating the background and seeds from panels. To guide the selection of an optimal set of threshold values, the first, last, and middle images of the series are shown to the user on a screen. Images from the first, last, and middle images were chosen as variation in imaging conditions across the course of the germination experiment as well as necessity to segment the emerging radicles both need to be considered when selecting the three YUV threshold values. Pixels with YUV values all above the three thresholds are assigned as 1 and pixels with any YUV values below the three thresholds are assigned as 0. This thresholding process generates a binary mask that is initially used to separate the panels containing seeds from the background.

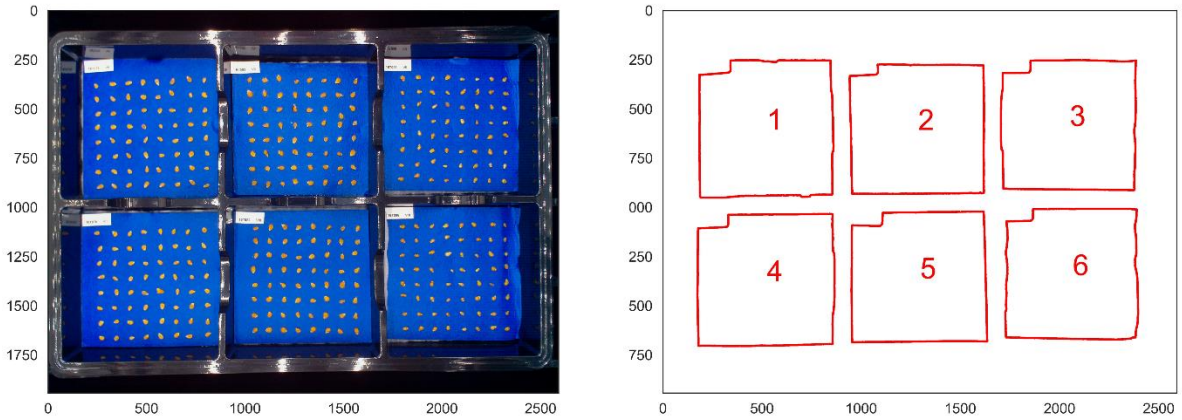

Figure 2.

(Left) An RGB image showcasing a seed germination experiment, depicting silver and black background, blue germination filter paper panels, and tomato seeds organised in 8x8 grids. (Right) Segmented panels, arranged sequentially from left to right, followed by top to bottom.

Using the generated binary mask, the panels of seeds can be segmented and are then ordered (Figure 2). If more objects were segmented than the expected number of panels, objects that were smaller than one sixth of the expected size of a panel were removed. Ordering was performed by considering the centroid of the y-coordinate of each panel first, followed by the x-coordinate of each panel's centroid. Meaning that the top left panel is assigned as the first panel, and the bottom right panel is assigned as the last. It was important to order the panels as if each panel contained a different genotype or treatment was applied, the recorded germination traits would correspond to the specific genotype or treatment. As a final step to validate the panels, the pipeline verifies that the identified panels contain seeds by counting the number of segmented objects that lie within the segmented panel area.

2.3.4 Seed Segmentation

The computational pipeline for seed segmentation employs a combination of machine learning models and image processing techniques to accurately identify and separate seeds in germination experiment images. The binary masks that were generated using the YUV threshold was used to generate pseudo-labelled images that were used to train a machine learning algorithm to predict whether a pixel should be

assigned to the foreground (seed) or background (germination paper panel) classes. To create the training datasets that were used to train the models, I chose to include a representative set of labelled training examples that comprised of the first 10 images (before imbibition), the middle image (after imbibition), and the last 5 images (after germination) (Figure 3). As mentioned in the previous section, this was done to ensure that variation in imaging conditions as well as seed radicle emergence was incorporated.

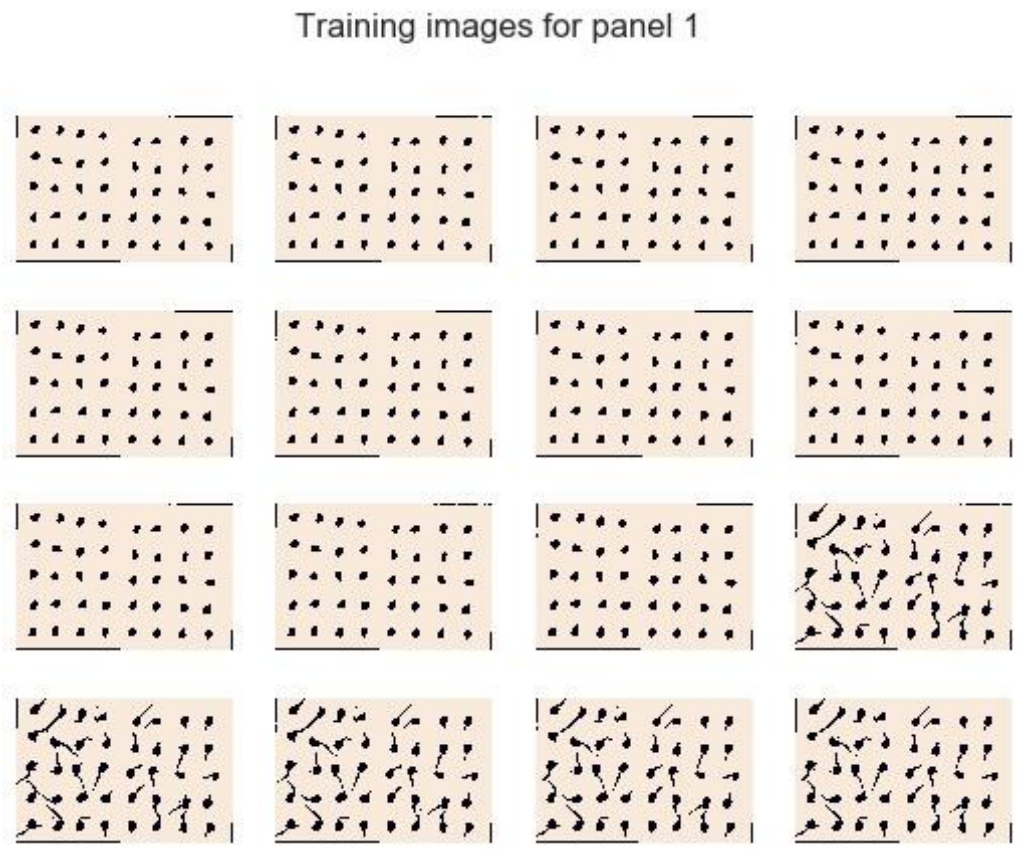

Figure 3.

16 training binary masks from a maize seed germination dataset. Black pixels indicate pixels belonging to the seed class, with the remaining pixels belonging to the blue germination filter paper class. The 10 images from the start, middle image, and final 5 images, are ordered sequentially from left to right and then from top to bottom.

For training a model to predict pixels for seed segmentation, I chose three methods that the user could select: 1. Gaussian Mixture Model, 2. Stochastic Gradient Descent Classifier, 3. U-Net. For all three models, I split the dataset of pixels into

80% training pixels and 20% test pixels for validation to determine when to stop model training and for hyperparameter tuning. Also, it is critical to note that for seed segmentation within panels, I trained and optimised a model for each panel as there can be significant variance in imaging conditions between panels that could negatively affect results if the model didn't generalise well between panels.

Gaussian Mixture Models (GMM) are an unsupervised learning technique used for clustering. I chose to apply a GMM in the context of image segmentation to separate seeds from blue backgrounds by modelling the distribution of YUV pixel intensities as a mixture of multiple Gaussian distributions. Each Gaussian distribution represents a distinct group of pixel intensities, corresponding to either the seeds or the germination paper background. The following steps outline how the GMM was employed for pixel segmentation.

The YUV pixel intensities and binary masks of the pseudo-labelled seeds and background from the chosen images were collated into training matrices, where the input data for the GMM is the YUV pixel intensities and the target is the binary mask. The GMM was then fitted to the pixel intensities by iteratively adjusting the parameters of the Gaussian components (mean, covariance, and mixing weights) using the expectation-maximisation (EM) algorithm. The EM algorithm refines the model parameters to maximise the likelihood of the observed data given the Gaussian mixture model. After converging to a solution, the GMM assigns each pixel to the Gaussian component with the highest probability. This step effectively separates the pixels into two clusters: one for seeds and the other for the blue germination paper background. After the GMM has converged, it is applied to all images in the image series to create binary masks from the cluster assignments, where one value (e.g., 1) represents seed pixels and the other value (e.g., 0) represents background pixels.

Stochastic Gradient Descent (SGD) classifier is a linear classification model frequently utilised in large-scale machine learning problems due to its computational efficiency, making it an efficient choice of model to classify millions of pixels. The core concept behind SGD is to optimise the model parameters iteratively using a stochastic approximation of the gradient of the loss function, which in this case was

chosen to be the hinge loss. The algorithm updates the model parameters in small steps, guided by a randomly selected subset of the training data (mini-batch) at each iteration. The randomness in selecting the mini-batch introduces noise into the optimisation process, which helps prevent the algorithm from getting stuck in local minima and improves the convergence speed.

The hinge loss function is defined by the following equation where L denotes the loss, Y denotes the ground truth class, W denotes the model weights, X denotes the input variables, and b denotes the model bias:

$$L(W, b) = \max\big(0, 1 - Y \cdot (W \cdot X + b)\big)$$

The hinge loss function quantifies the difference between the predicted output $\hat{Y}$ and the actual target values Y. If the prediction is correct and has a margin greater than 1 (i.e., the prediction is sufficiently far from the decision boundary), the hinge loss is 0. However, if the prediction is incorrect or has a margin less than 1, the hinge loss is positive and proportional to the distance from the correct side of the decision boundary. The hinge loss was selected so that the SGD classifier weighted the quantitative error more towards pixels whose predictions were close to the decision boundary as the hinge loss focuses on the most challenging samples. This should mean that the resulting predicted binary mask should be more precise around the pixels on the edges of seeds that may have otherwise been incorrectly classified as background if for example, log loss was chosen instead.

I implemented SGD to classify pixels in a similar way to the GMM implementation in that a large training dataset was prepared comprising two equally sized matrices that contained the YUV pixel intensities and the binary mask values from the selected training images. Following this, I trained the SGD model and optimised its hyperparameters using a grid search to identify the value of the alpha parameter that resulted in the lowest validation loss. Using the early stopping hyperparameter, I stopped training the SGD classifier at the epoch that resulted in the lowest validation loss after 10 iterations had passed with the loss not decreasing. Upon training the optimised model, it could be applied to the entire image series, segmenting all seeds within all panels.

The final model that I chose to use for seed segmentation was U-Net, a CNN architecture specifically designed for semantic segmentation tasks. Initially developed for biomedical applications, U-Net has since proven effective in various image segmentation domains. The U-Net architecture features a U-shape, consisting of an encoding path and a decoding path, which together give the architecture a U-shape. The encoding path consists of a series of convolutional layers with increasing feature channels, starting with 32 filters in the first convolutional layer and doubling this number at each subsequent layer in the encoding path, reaching 512 filters before the decoding path. followed by max-pooling layers for down-sampling. The decoding path mirrors the encoding path with a series of up-sampling layers and convolutional layers with decreasing feature channels. Skip connections from the encoding path to the decoding path are used to merge high-resolution features with up-sampled outputs, in order to capture both local and global contextual information.

The exact architecture of the implemented U-Net model is provided below:

Encoding Path

1. **Input Layer**: Takes images of specified shape.
2. **Convolutional Block 1**:
    - Convolutional Layer (32 filters, 3x3 kernel, ReLU, padding=same)
    - Convolutional Layer (32 filters, 3x3 kernel, ReLU, padding=same)
    - Max Pooling (2x2)
3. **Convolutional Block 2**:
    - Convolutional Layer (64 filters, 3x3 kernel, ReLU, padding=same)
    - Convolutional Layer (64 filters, 3x3 kernel, ReLU, padding=same)
    - Max Pooling (2x2)
4. **Convolutional Block 3**:
    - Convolutional Layer (128 filters, 3x3 kernel, ReLU, padding=same)
    - Convolutional Layer (128 filters, 3x3 kernel, ReLU, padding=same)
    - Max Pooling (2x2)
5. **Convolutional Block 4**:
    - Convolutional Layer (256 filters, 3x3 kernel, ReLU, padding=same)
    - Convolutional Layer (256 filters, 3x3 kernel, ReLU, padding=same)

- Max Pooling (2x2)

Bottleneck

- **Convolutional Block 5**:
  - Convolutional Layer (512 filters, 3x3 kernel, ReLU, padding=same)
  - Convolutional Layer (512 filters, 3x3 kernel, ReLU, padding=same)

Decoding Path

1. **UpSampling Block 1**:
   - UpSampling (2x2)
   - Concatenate with corresponding cropped output from Convolutional Block 4
   - Convolutional Layer (256 filters, 3x3 kernel, ReLU, padding=same)
   - Convolutional Layer (256 filters, 3x3 kernel, ReLU, padding=same)
2. **UpSampling Block 2**:
   - UpSampling (2x2)
   - Concatenate with corresponding cropped output from Convolutional Block 3
   - Convolutional Layer (128 filters, 3x3 kernel, ReLU, padding=same)
   - Convolutional Layer (128 filters, 3x3 kernel, ReLU, padding=same)
3. **UpSampling Block 3**:
   - UpSampling (2x2)
   - Concatenate with corresponding cropped output from Convolutional Block 2
   - Convolutional Layer (64 filters, 3x3 kernel, ReLU, padding=same)
   - Convolutional Layer (64 filters, 3x3 kernel, ReLU, padding=same)
4. **UpSampling Block 4**:
   - UpSampling (2x2)
   - Concatenate with corresponding cropped output from Convolutional Block 1
   - Convolutional Layer (32 filters, 3x3 kernel, ReLU, padding=same)
   - Convolutional Layer (32 filters, 3x3 kernel, ReLU, padding=same)
   - ZeroPadding2D (to adjust the final layer's output size to match the input)

Output Layer

- Convolutional Layer (number of classes, 1x1 kernel, softmax) for pixel-wise classification.

Training this model was different to training the unsupervised GMM and supervised SGD classifier and the implementation of U-Net is exploratory, with the aim of using deep learning techniques to improve analysis. Unlike the aforementioned models, which expected unordered matrices of pixels as input, the U-Net model requires images of the panels containing seeds as it utilises convolutional layers to encode spatial relationships between neighbouring pixels. To prepare the training dataset for U-Net, I generated an array of 16 input images as well as an array of the 16 binary mask targets. Using this data, I applied 8-fold cross-validation for hyperparameter optimisation, focusing on the learning rate and choice of optimiser. I used a grid search to attempt to identify the set of hyperparameters that minimised the validation loss. I also implemented an early stopping mechanism to determine the optimal number of epochs for training and a mechanism to reduce the learning rate once it plateaued to better optimise the fit. This learning rate mechanism was used as although a large learning rate can result in rapid convergence, an excessively large learning rate may cause the model to update its weights too drastically so that it struggles to converge. In contrast, a smaller learning rate can help fine-tune the model once it approaches convergence.

The U-Net model's output takes the form of a binary mask, in which pixels associated with seeds are assigned a value of 1, while those corresponding to the background receive a value of 0. This binary mask segments the seeds from the background, providing a clear distinction between the two classes. With the U-Net model in place, it can be applied to the remaining images in the series, accurately classifying each pixel and successfully segmenting the seeds. This approach enables precise seed identification and analysis, demonstrating the U-Net model's robustness and effectiveness in image segmentation tasks.

Building on the strengths of Gaussian Mixture Models (GMM), Stochastic Gradient Descent (SGD) classifiers, and the U-Net convolutional neural network architecture, I also employed an ensemble method to generate a robust binary segmentation mask for seeds and the germination filter paper background. The distinct advantages of

each method are effectively leveraged to enhance the accuracy and reliability of the binary mask.

The GMM approach excels at identifying underlying patterns in the distribution of YUV pixel intensities, effectively distinguishing between the seeds and the blue germination paper background. However, it's an unsupervised method and, thus, might have some limitations in accuracy. Complementing the GMM, the SGD classifier focuses on quantitative errors, with its hinge loss function allowing the classifier to more precisely handle pixels close to the decision boundary, an area where the GMM could falter. These two methods together can ensure a strong foundation for pixel classification, taking advantage of both unsupervised clustering and supervised classification approaches.

The U-Net CNN is also employed, encoding spatial relationships between pixels through its unique architecture. It improves upon the GMM and SGD classifier by incorporating the context of neighbouring pixels in the segmentation, which the other two models, treating each pixel individually, do not provide. The U-Net's capability to capture both local and global contextual information provides a level of robustness that a single model could not achieve.

This ensemble method effectively combines the strengths of these three techniques while also compensating for their individual limitations. The result is a highly robust, accurate segmentation binary mask. This combination benefits from the computational efficiency of SGD, the unsupervised pattern recognition of GMM, and the spatial contextual understanding of U-Net. It improves the overall segmentation performance and delivers a more reliable and precise binary mask that can better differentiate between seed and background pixels.

While the ensemble approach offers benefits in terms of robustness and accuracy, it also comes with certain challenges. One significant drawback is the increased computational cost and processing time. Each individual model in the ensemble, namely GMM, SGD, and U-Net, requires its own computational resources and time for training and inference. When combined, these requirements accumulate, making

the ensemble method more computationally expensive and time-consuming than any single model.

2.3.5 Feature Extraction

Once the seeds had been segmented, I next extracted morphological features that could be used for classifying seeds as having germinated or not. When considering this step, I planned on extracting features where the values of a non-germinated and germinated seed would differ most – for example, circularity as a non-germinated seed is likely to be more circular in morphology compared to a seed that has germinated with an emerged radicle. To achieve this, I used the scikit-image Python library, specifically the **regionprops** function within the **skimage.measure** module to extract various properties from the distinct segmented objects within each panel. The result allowed me to compile a list of features for each seed across the image series in each panel, with the features being:

Centroid coordinates – The centroid coordinates refer to the x and y pixel locations of the central point of a segmented seed in an image. During germination, seeds may move or roll due to various factors, such as the growth of roots and shoots, external forces, or interactions with other seeds. Tracking the centroid coordinates of each seed enables the identification and monitoring of individual seeds even if they change their positions during the experiment. By maintaining accurate tracking of seeds, I could ensure that the measurements and properties of a single seed remained consistent across different stages of the experiment.

Bounding box – The bounding box is the rectangular outline that encloses a segmented seed, providing a compact representation of the seed's size and position in the image. Seed orientation can therefore be assessed by examining the bounding box's aspect ratio (width to height ratio). In certain cases, this might contain information relating to the angle of a seed's position, or elongation in a certain direction for example in barley seeds that are less circular. Furthermore, bounding boxes can be instrumental in detecting overlapping seeds within an image. When the bounding boxes of multiple seeds intersect or partially overlap, this may indicate that

the seeds are in close proximity or have physically overlapped, potentially affecting the accuracy of seed segmentation and the subsequent germination analysis.

Hu moments – Hu moments[254] are a set of seven moment-based features that are invariant to translation, rotation, and scaling, allowing them to robustly characterise seed shapes. These shape descriptors can serve as useful features in differentiating between germinated and non-germinated seeds, as germinated seeds may display distinct shapes owing to the emergence of a radicle that the model can identify and use.

Area – The area refers to the total number of pixels occupied by a seed within the segmented region of the image. The area is useful for determining the size of a seed and detecting changes that occur during the germination process as seeds undergo several transformations, including imbibition (water absorption) and the emergence of a radicle. Imbibition causes the seed to swell as its internal tissues expand to accommodate the absorbed water. Consequently, this swelling leads to an increase in the seed's size, which is reflected in its area. A significant change in the seed area could serve as an indicator that the germination process is underway. Furthermore, as the seed germinates, a radicle emerges, modifying the seed's shape and potentially increasing its area. Incorporating patterns associated with a seed's area could enable the model to distinguish germinated and non-germinated seeds more accurately.

Perimeter – The perimeter quantifies the length of the seed's outer boundary and can be an effective metric for detecting alterations in seed shape or surface texture during germination. As seeds germinate, they undergo various morphological changes that can impact their shape and surface texture. For instance, the emergence of a radicle can cause the seed's overall shape to become more elongated or irregular which can be captured by monitoring the seed's perimeter over time. Fluctuations in perimeter may therefore indicate that the seed is undergoing significant transformations associated with germination, while a relatively stable perimeter could suggest that the seed remains unchanged. By recognising these patterns, the model can more accurately distinguish between germinated and non-germinated seeds, leading to improved predictions of germination status.

Eccentricity – Eccentricity quantifies the elongation of a region and can be instrumental in identifying germinated seeds that exhibit elongated shapes as a result of an emerging radicle. Eccentricity is a value that ranges from 0 to 1, where 0 represents a perfect circle and values closer to 1 indicate increasingly elongated shapes. During germination, seeds undergo various morphological changes, which can cause the seed's overall shape to become elongated. Therefore, monitoring the seed's eccentricity allows for the detection of these shape alterations, potentially serving as an indicator of germination progress for a machine learning model.

Minor and major axis length – These features characterise the dimensions of an ellipse fitted to the seed's contour, providing information about the seed's size and shape, both of which undergo changes during germination. The major axis length corresponds to the length of the longest axis (or the major diameter) of the fitted ellipse, while the minor axis length refers to the length of the shortest axis (or the minor diameter). These two metrics offer insights into the seed's overall dimensions and can help identify changes in size and shape that occur during the germination process. I considered that in cases where ungerminated seeds are round, calculating the ratio of the major and minor axis lengths could be an effective predictor of germination as the emergence of a radicle would rapidly change this ratio.

Convex area – The convex area refers to the area of the smallest convex polygon that fully encloses the seed. It is an important shape descriptor that helps capture the general outline of the seed, accounting for its size and any potential concavities or irregularities.

Solidity – Solidity denotes the ratio of the seed's area to its convex area, quantifying the compactness of the seed's shape. A solidity value of 1 indicates a completely convex shape, while values closer to 0 suggest a more concave or irregular shape. As seeds germinate, they undergo various morphological changes, such as radicle emergence, which can affect the seed's overall shape, potentially altering its solidity value. Incorporating solidity as a feature in a machine learning model might enhance the model's ability to recognise patterns related to seed shape changes during germination.

Extent – Extent calculates the proportion of a seed's area relative to its bounding box area, offering insights into the seed's shape and size. The extent metric is derived by dividing the seed's area by the area of its bounding box, which is the smallest rectangle that fully encloses the seed. Similar to solidity, extent values range from 0 to 1, with values closer to 1 indicating that the seed's shape closely aligns with the shape of its bounding box, while lower values suggest that the seed occupies a smaller portion of the bounding box, potentially indicating a more irregular or concave shape.

Mean and variance of RGB values – The average RGB pixel values of seeds, can be utilised to detect changes in seed colour during germination. For instance, germinated seeds may exhibit different colour characteristics due to the emergence of radicles. Additionally, certain colour changes may be associated with the seed's health or vigour, which can further contribute to the assessment of germination status.

In summary, I chose a range of morphological features to train a machine learning model for predicting seed germination and to track through the experiment as additional phenotypes that researchers could use to identify molecular markers associated with germination. I selected these features based on their ability to capture various aspects of the seed's development, shape, size, and colour changes that may occur during the germination process. Due to the comprehensive profile these morphological features provide the machine learning model, I think it will enhance the model's ability to recognise patterns related to seed germination and allow for more accurate classification of germinated and non-germinated seeds.

2.3.6 Quality Control and Seed Ordering

After extracting morphological features for all objects within each panel across the entire image series, I calculated the number of segmented objects and compared it with the expected number of seeds in each panel, as defined by the user at the beginning of the analysis. If there were more objects than expected, I implemented a filtering process to remove objects smaller than 0.6 times the 10^{th} percentile or larger

than 1.4 times the 95th percentile of the other objects' areas. This approach ensures that abnormally small or large objects, such as debris or filter paper fragments that might have been inaccurately classified as seeds, are eliminated.

Once the outlier objects are removed, I proceed to order the seeds by their y-coordinate first, followed by their x-coordinate. This ordering means that the top-left seed is considered the first, and the bottom-right seed is the last. To accomplish this, I utilise the centroid coordinates of each seed across the image series and use a histogram to create bins for both x- and y- coordinates. Starting with the first bin of y-coordinates, I label the seeds whose centroids lie within this bin, counting them from the lowest to the highest x-coordinate (Figure 4). This process is then repeated for all bins of y-coordinates. This step is crucial for maintaining consistency when measuring statistics for each seed in each image, ensuring that the data for each seed in the current image corresponds to the same seeds in the previous image.

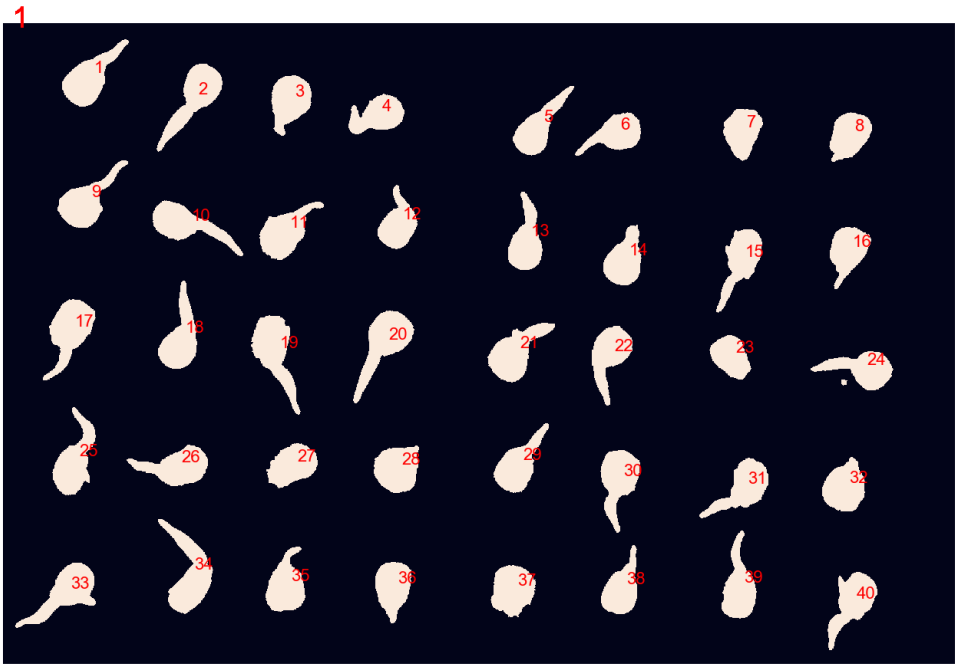

Figure 4.

Segmented and ordered maize seeds, where the red number next to each seed indicates the seed number and the larger red number in the top left corner indicates the panel number.

After organising and measuring the seeds in each image, I compiled the measured morphological features of each seed into structured tables using **pandas** dataframes, a data manipulation tool from the **pandas** library in Python, for different purposes and to facilitate saving them as CSV files at the end of the analysis. First, I created a main dataframe containing data from all images and panels, encompassing the entire morphological feature set. It was used to track overall morphological features across the entire dataset, providing a comprehensive view of seed development and germination trends. At the end of the analysis, the user can export the master dataframe as a CSV file for further examination, or to share with collaborators. In addition to the main dataframe, I created separate dataframes for each panel, containing data from all images related to a single panel. These specific dataframes enabled me to train the machine learning model separately for each panel due to potential variations in imaging conditions, seed placement, or other factors that could influence the model's performance.

2.3.7 Germination Classification

Due to having no labelled data except for knowing that all seeds at the start of the experiment would not have germinated, I needed to use an unsupervised learning model. Therefore, I used a one-class support vector machine (SVM) as this task could be considered novelty detection where non-germinated seeds are considered as the 'normal' class and germinated seeds are considered as outliers. A one-class support vector machine is a variation of the standard SVM algorithm specifically designed to handle unsupervised learning problems, particularly in novelty detection or outlier detection tasks. Unlike traditional SVMs that classify data points into two or more classes based on a supervised learning approach, a one-class SVM focuses on learning the characteristics of a single class, without the need for labelled data.

The one-class SVM works by mapping the input data points into a higher-dimensional space using a kernel function, such as the Radial Basis Function (RBF) or linear kernel. In this higher-dimensional space, the algorithm aims to find the best hyperplane that separates the data points from the origin. This hyperplane acts as a decision boundary, maximising the margin between the 'normal' data points and the

origin. Data points that lie on the same side of the hyperplane as the ‘normal’ class are considered to be part of this class, while those on the other side are considered outliers or novel instances.

To train the one-class SVM, I extracted the morphological features from the first 20% of images in the series, assuming that these images wouldn’t contain any germinated seeds. This assumption allowed me to treat their features as representative of the non-germinated seed class. I combined these features to form a training matrix and used it to train the one-class SVM, capturing the ‘normal’ feature patterns characterising non-germinated seeds. The trained one-class SVM was then applied to the remaining dataset, which included both germinated and non-germinated seeds. Based on the training data, the model would generate a decision function to enclose pre-germination feature vectors in the resulting p-dimensional space where p is the number of features. As a germination experiment progresses, feature vectors would be recomputed, and when a seed begins to germinate, its feature vector would gradually leave the boundary of the initial observation region, indicating a higher probability of germination.

The novelty detection model scored germination for all detected seeds, resulting in cumulative germination rates for each seed lot in a given germination panel. Seeds with morphological features that deviated significantly from the ‘normal’ distribution were considered outliers or ‘novel’ instances, representing germinated seeds. A threshold of 0.5 was applied to the predicted probabilities to distinguish between germinated and non-germinated seeds. Since the novelty detection model is reinitialised and retrained for each experiment using the first 20% of pre-germination images from the selected image series as training data, the detection model is dynamic, reducing the risk of overfitting. This approach is a key advantage over other supervised methods as it enables the model to adapt to varying imaging conditions and account for differences between seed batches.

2.3.8 GUI-based Analysis Software

Together with my colleagues Prof. Ji Zhou, Dr Aaron Bostrom, and Dr Danny Websdale, we created an ML-based phenotypic analysis module with workflows that are fundamentally identical for both GUI and command-line approaches. We utilised the more accessible GUI software to introduce the analysis procedure to researchers wishing to analyse their own seed germination experiments, designed to execute on either Windows (i.e., the .exe executable, tested on Windows 10) or macOS (i.e., the .app file, version 10 onwards). The analysis software packages can be downloaded from the following GitHub repository (https://github.com/Crop-Phenomics-Group/SeedGerm). The initial GUI contains an empty window with a menu bar, allowing users to add experiments via the "Add experiment" window. Here, users can enter a given experiment’s name, select an image series for processing, and choose a crop species such as Brassica, maize, pepper, tomato or cereals. New plant species can be trained and added to the software through the Modules directory, an approach independent of the core analysis algorithm.

Users need to briefly define the germination experiment associated with the selected image series, including the number of panels in a given SeedGerm device, as well as the rows and columns of seeds in each panel. Importantly, users can define the start and end image IDs to initiate and terminate the phenotypic analysis. This is because the background in early images can be oversaturated due to excess water absorbed by the filter paper, while late images may contain too many overgrown seedlings and roots. Default values for the start and end images are the first and last images of the selected series.

To accommodate varied image quality and features arising from factors such as lighting, crop species, and different establishment phases, I have implemented a range of ML-based algorithms in the software. Users can choose the ML technique from the "BG remover" dropdown menu to eliminate background pixels. The available options include U-Net, Gaussian mixture model (GMM), and stochastic gradient descent (SGD). Once an experiment has been added, users need to set YUV colour-space ranges (with Y representing brightness and U and V denoting colour

components) to delineate the background (i.e., filter paper) in the first image of the selected series. By adjusting the sliding bars in the "Set YUV ranges" window, various backgrounds can be retained to account for different types of filter paper used in diverse experiments.

After defining YUV values, users can click the "Process images" item to initiate the phenotypic analysis. The analysis software also employs parallel computing to process multiple experiments simultaneously. Up to 12 image series can be analysed concurrently on an average computer (Intel Core i5, 8GB RAM), and over 120 series on an HPC. This implementation enables multi-threading analysis to run on HPC clusters, facilitating greater throughput.

Upon completion of the analysis, various germination traits (e.g. germination timing curves for each panel, G_{max}, T25, T50, T75) and morphological traits (e.g. extent, area, width and length, convex area, and eccentricity for each seed) are generated, along with a collection of processed images that depict the germination process and label individual seeds. Users can click "View results" in the shortcut menu to display the analysis outputs. Additionally, they can download an assortment of processed images and the analysis results in CSV files. These files provide phenotypic analysis data at the image (overall results), panel (i.e. a given genotype), and seed levels.

2.4 Results

In this results section, I provide an analysis of the performance and outputs of the SeedGerm system. I evaluate the comprehensive performance of the system using relevant evaluation metrics, comparing SeedGerm's predictions with traditional methods of manual scoring across a variety of crop species. This comparative analysis demonstrates the benefits and constraints of the approach, underscoring its proficiency in handling intricate seed germination data analysis tasks.

Moreover, I showcase the SeedGerm system as a research tool in molecular biology experiments performed by collaborators at the John Innes Centre, further benchmarking its effectiveness in distinguishing genetic differences in seed germination. The germination parameters of various varieties analysed by SeedGerm is discussed, offering insights into their germination behaviours and potential biological significance. Through presenting these results, my goal is to underscore the system's capacity to discern valuable patterns and relationships in seed germination data, ultimately leading to a richer understanding of the underlying biological and agricultural processes.

2.4.1 Germination and Morphological Trait Quantification

A series of germination experiments were conducted to evaluate and enhance the SeedGerm platform. Figure 5 presents the analysis results of an experiment involving 384 tomato seeds (six genotypes) arranged on six panels in a customised germination box, with one genotype per panel (64 seeds). The imaging interval was 60 minutes, and a total of 186 images were captured over eight days. The analysis outputs encompass two categories of traits: (1) germination traits, quantified using the 1st to 186th images, which include cumulative germination curves, T50 germination rates to evaluate germination uniformity, and G_{max} to determine the proportion of seeds germinated by the end of the experiment; and (2) morphological traits, quantified using the 1st to 160th images, comprising seed area, width-to-length (W/L) ratio, and circularity. By integrating both trait types, it is possible to identify morphological changes in the six genotypes at the pre-germination stage

(before the 106th image). As the germination process commenced, the cumulative germination curves and corresponding morphological features diverged between genotypes. A strong correlation between germination curves and seed area curves is evident, aligning with the developmental stage when radicles emerge from seeds, significantly increasing the width/length ratio. The presence of more roots results in a lower W/L ratio and circularity. This quantification demonstrates the utility of combining both germination and morphological traits to validate and enhance the accuracy of seed germination scoring.

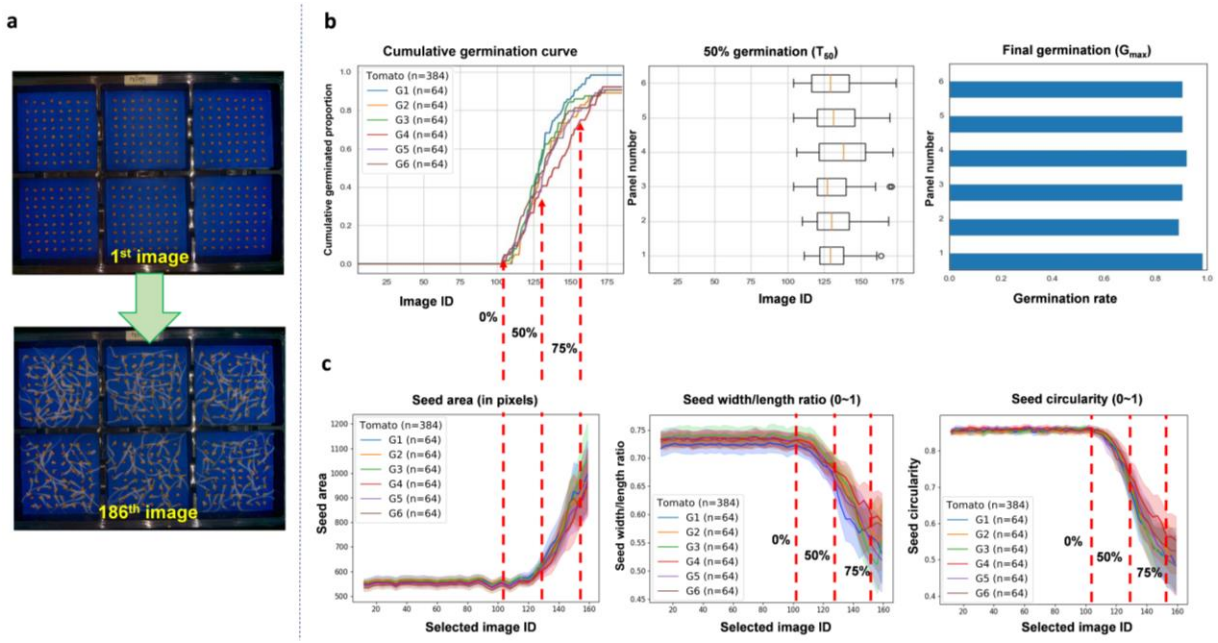

Figure 5.

Tomato experiment germination and morphological results
(a) The time-lapse image series of six tomato genotypes (384 seeds) acquired in an eight-day experiment. (b) Germination-related traits quantified, including cumulative germination curves, T_{50} germination rates, and G_{max}, final germination rate. (c) Morphological traits quantified, including seed area, width and length ratio, and circularity.

Moreover, I employed the analysis outputs to assess germination uniformity or variability, a crucial trait that previously required complex calculations to compute. For instance, box-and-whisker plots are included among the result files to illustrate the statistical dispersion of T50 germination rates highlighting the difference between the 25th and 75th percentiles for each genotype, as well as the median time to 50% germination for all genotypes. For example, genotype 6 seeds exhibit lower

germination variability and better germination uniformity, as confirmed by the narrower percentile ranges and consistent median values across the tested seed batches. I excluded several late images (post-T75) when presenting the morphological traits due to the significant measurement variations caused by overlapping roots at the later stages.

2.4.2 Germination Analysis for Different Crop Seeds

To demonstrate the robustness and generalisability of the SeedGerm system, I applied it to score germination for a variety of crop seeds. The germination analyses for four selected crop species include tomato, pepper, maize, and barley (Figure 6). Seed images at three distinct experimental stages can be observed on the left side of Figure 6. After conducting time-series seed imaging, I utilised SeedGerm software to measure germination and morphological traits. Each germination panel (enclosed by red-coloured dotted rectangles in Figure 6) contains a single genotype. Seeds within the panel were monitored continuously, with varying durations due to different research objectives, such as 165 hours (7 days) for tomato, 180 hours (8 days) for pepper, 138 hours (6-7 days) for maize, and 138 hours (6-7 days) for barley. These experiments were also assessed by specialists on a daily basis, allowing for a comparison and verification of manual and SeedGerm scores.

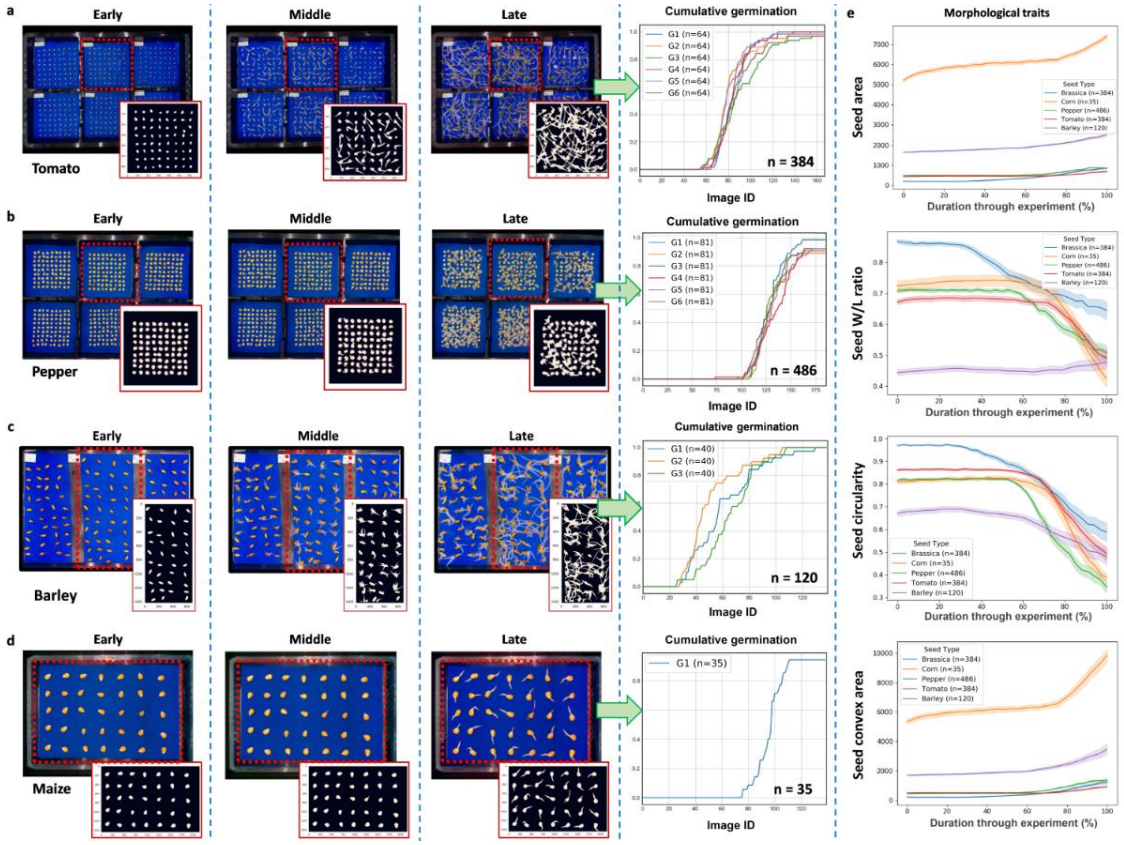

Figure 6.

Germination and morphological measurements for different crop species
(a) 384 tomato seeds (six genotypes) used for seed germination experiments, producing six cumulative germination curves on hourly measures during a seven-day period. (b) 486 pepper seeds (six genotypes) used for germination experiments, producing six cumulative germination curves on hourly measures during an eight-day period. (c) 120 barley seeds (three genotypes) used for germination experiments, producing three cumulative germination curves on hourly measures during a six-day period. (d) 35 maize seeds (one genotype) used for a germination experiment, producing a cumulative germination curve on hourly measures during a six-day period. (e) Morphological measurements produced by plotting hourly changes against the duration of experiments, so that all experiments can be compared on similar bases, including a number of traits such as seed area (in pixels), W/L ratio (0~1), seed circularity (0~1), and convex area (in pixels).

The tomato seed germination experiments were conducted across six panels (i.e. six genotypes), with 64 seeds per panel and a total of 384 seeds monitored (Figure 6A). Six cumulative germination curves were produced based on hourly measurements over a seven-day period. I could clearly identify minor differences among these genotypes between T50 and T75, when germination rates diverged. Similarly,

germination variances could also be quantified for pepper and barley experiments (Figures 6B,C). The three barley genotypes monitored displayed a wide range of cumulative germination, consistent with previous reports. Due to the size of maize seeds, I conducted one experiment per germination box (35 seeds per box, Figure 6D). Nevertheless, the SeedGerm software can perform accurate measurements even when the number of germination experiments varies. The above panel- and seed-level germination measures were exported and saved in multiple CSV files.

More complex morphological traits incorporated in the SeedGerm analysis include seed convex area, seed extent, and seed solidity, which were used to quantify the dynamics of germination for different crop seeds as they were challenging to assess using traditional approaches. For example, using the seed convex area trait, I found that maize had the fastest establishment rate after T50, while other crop seeds were quite similar (Figure 6E). Due to significant variations caused by the excessive overlap of radicals at the late germination stage, end image IDs for the above analysis differ. Similarly, the panel- and seed-level morphological measures are saved in CSV file format.

2.4.3 Validation of the SeedGerm Platform

To validate the analysis produced by SeedGerm, a variety of metrics were generated (Table 1), including Pearson's correlation coefficient (r) to measure the strength of the linear relationship between SeedGerm and manual scoring for cumulative germination rates. For all tested crop species, SeedGerm's cumulative predictions yield a Pearson's correlation greater than 0.98 (column two in Table 1), indicating a very strong linear correlation. Pearson's correlation (r) was also employed to evaluate the linear relationship between SeedGerm's true positive germination timings and their respective timings scored by seed scientists (column three in Table 1).

In addition to the correlation metrics, the mean absolute error (MAE, column four in Table 1) was calculated to interpret the average error in hours of the predicted germination time compared with manual scores for each germinated seed. The MAE

measures forecast error in SeedGerm's prediction against human scoring (the true value), demonstrating a satisfactory error range. Lastly, the F1 score (1 indicates a perfect set of classifications and 0 means all false negatives or false positives), a classification metric similar to accuracy but more suitable for imbalanced datasets, was used to incorporate the number of true positives, false positives, and false negatives into a single score for evaluating the germination classifications made by SeedGerm. Based on F1 scores (column five in Table 1), it is apparent that the SeedGerm performed well across all tested crop species. The aforementioned metrics evaluate both SeedGerm's final germination scoring and the germination timing of each seed, covering germination rate, timing, and uniformity, respectively.

Table 1. Validation metrics used to compare between manual counting and SeedGerm scoring

r denotes Pearson's correlation, MAE denotes mean absolute error, F1 denotes F1 score.

	***r* Cumulative Rate**	***r* Image ID**	**MAE (hours)**	**F1**
Barley	0.981	0.804	13.275	0.961
Brassica	0.992	0.886	9.141	0.936
Maize	0.994	0.874	3.543	0.986
Pepper	0.999	0.952	6.025	0.994
Tomato	0.993	0.888	4.903	0.992

To visualise the correlation between SeedGerm scoring and seed specialists' counting, 19 time series (over 4,000 images in total) were used for the correlation analysis, comprising three series of maize (129 seeds in total), six series of tomato (384 seeds), six series of Brassica (384 seeds), one series of pepper (81 seeds), and three series of barley (120 seeds). Manual scoring was performed using the image series, where cumulative germinated seed counts for each image and the image ID for when each seed germinates were recorded. There is a strong correlation between SeedGerm's scoring and that of the manual observers, as illustrated in Figure 7. A predicted-equals-actual line (coloured red) is included (Figure 7A) to demonstrate

how SeedGerm's cumulative scores deviate from the manual scores. Furthermore, line plots contrasting cumulative seed-by-seed scoring between SeedGerm and specialists' counting are displayed in Figure 7B. SeedGerm's scoring is largely identical in comparison with manual counting, except for a tendency to overestimate the number of germinated seeds in crowded experiments, such as the later establishment stages for Brassica, pepper, and tomato experiments.

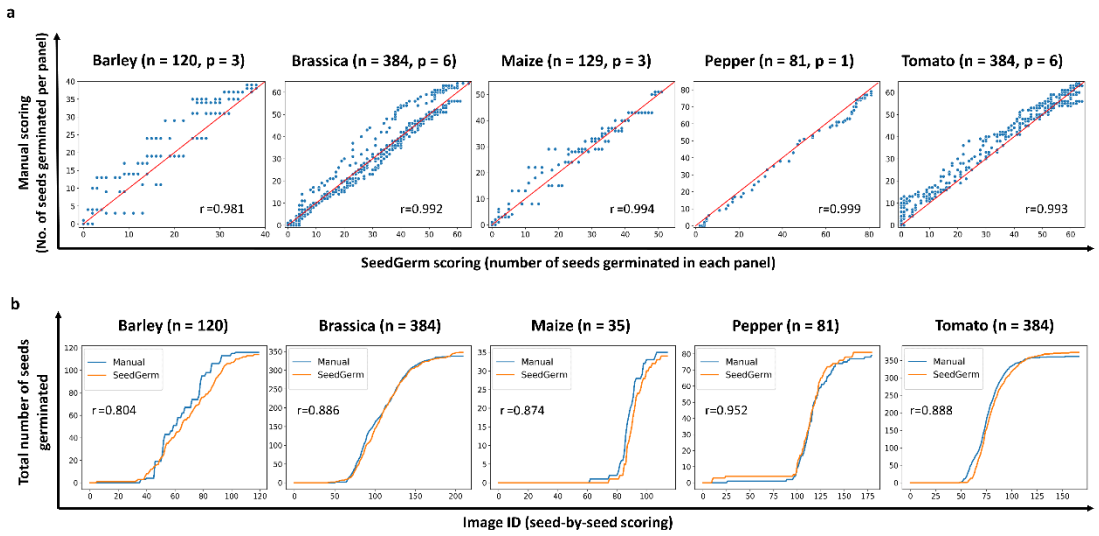

Figure 7.

Correlation between manual and SeedGerm seed scoring
(a) For all tested species, SeedGerm's predictions display a strong linear correlation and goodness of fit (r > 0.98) based on cumulative germination rates. Each point represents the number of seeds classified as germinated in a panel in an image, meaning multiple populations are plotted together. The red line displays SeedGerm's cumulative count equalling the cumulative manual count. The number of panels associated with each scatterplot is denoted as p. (b) Seed-by-seed scoring between SeedGerm and specialists' counting plotted to demonstrate the reliability of SeedGerm scoring as well as its tendency to predict additional germinated seeds at the end of crowded experiments.

2.4.4 Brassica Genome Wide Association Analysis

To test the ability of SeedGerm to be used as a research tool in routine biological experiments, together with collaborators at the John Innes Centre, we used the *B. napus* Diversity Fixed Foundation Set (Harper et al., 2012) to detect genetic differences in seed germination. After setting replicate seed batches of each variety, biological replicates of 50 seeds were sowed in SeedGerm boxes in a randomised design. SeedGerm scored the germination parameters of 88 varieties with a range of germination behaviours, with some showing strong dormancy, while most seed lots germinated to high levels, but with varying kinetics. SeedGerm scored the T_{10}, T_{50}, T_{90} and G_{max} after 8 days. To test the accuracy of the SeedGerm outputs, 60 seed lots were also scored by a manual observer. The correlation was high, except for T_{90} in varieties requiring the longest time to germination, where SeedGerm has a weak tendency to score seeds as germinated before the manual observation.

The SeedGerm outputs were then used for associative transcriptomic (AT) analysis[255]. The AT found no significant associations between T_{10}, T_{50} and T_{90} and polymorphisms in *B. napus*. However, a strong association between G_{max} and genotype on chromosome A5 was found, with both SNPs and gene expression markers [256] associated with the trait in this region (Figure 8). This is distinct from loci identified in previous studies [257,258], but significant, even after correcting for multiple testing. This region spans approximately 340kb and contains at least 69 known transcribed genes, one of which is a *B. napus* orthologue of the known germination regulator, protein phosphatase 2C known as *HIGH ABA INDUCED 3* (*HAI3*) [259,260], which has a role in seed sensitivity to abscisic acid. Although more work is needed to precisely identify the underlying gene of interest, it is evident that the SeedGerm platform is capable of automating phenotypic analysis of seed germination with sufficient accuracy to perform standard genetic analysis of seed dormancy and germination rate.

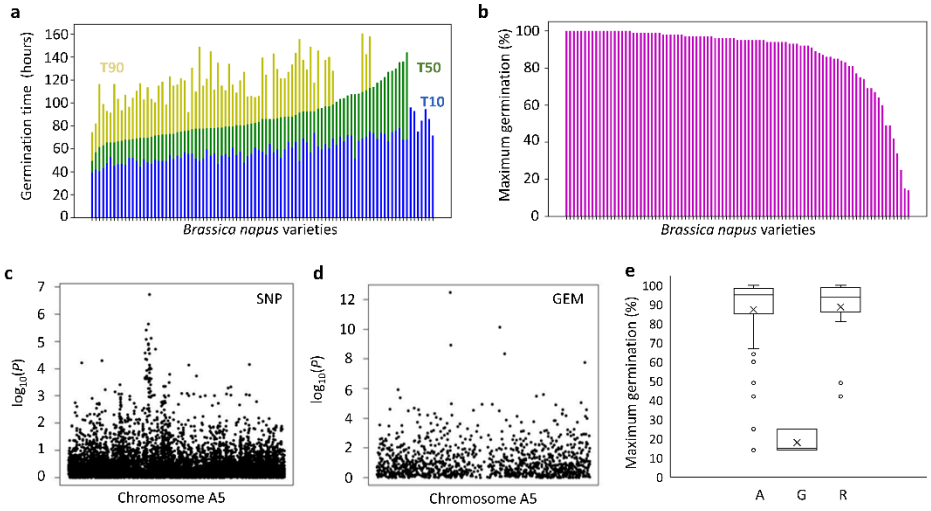

Figure 8.

SeedGerm's application in detecting genetic variances in 88 Brassica napus varieties exhibiting diverse germination behaviours, scored by T10, T50, T90, and G_{max} metrics after eight days. ***(a)*** *Germination times for T10, T50, and T90 rates across varieties.* ***(b)*** *G_{max} rates for the same set.* ***(c)*** *SNP association between G_{max} and genotype on chromosome A5, with a blue dashed line indicating significant associations (FDR of 0.1).* ***(d)*** *Box plot illustrating germination scores based on allele score for the key SNP on chromosome A5, with details on quartile values, range, and outliers.* ***(e)*** *Correlation analysis of gene expression markers highlighting the association of at least two genes in the region with germination, showcasing SeedGerm's precision in conducting genetic analysis of seed dormancy and germination rate (significant associations marked with a blue dashed line; FDR of 0.005).*

2.5 Discussion

2.5.1 Automated Seed Phenotyping

Plant phenomics is a fast-developing research area focusing on obtaining meaningful phenotypic information to enable scientists to address diverse biological questions, from cellular organisms to populations in the field[261–264]. Numerous academic and industrial efforts have been made to study seed germination and seedling vigour. They have utilised research-based tools like MultiSense, RSGES, and Germinator, as well as commercial solutions. These include the PhenoSeeder platform developed by Forschungszentrum Jülich in Germany, SeedAIXPERT, the Germination Scanalyzer, and the Seeds Automatic Germination Analyzer (SAGA) from France. These methods are capable of carrying out high-throughput seed imaging, advanced 3D seed morphological analysis (i.e. *pheno*Seeder), and germination related traits analyses; however, their applications are limited due to their costs, availability, automation level, analysis throughput, and the technical scalability.

In this chapter, I have presented the SeedGerm system, a platform that combines automated seed imaging and vision-based phenotypic analysis with cost-effective hardware to enable high-throughput analysis of seed germination experiments for a variety of crop species. Based on the experiments and system improvements, I believe that the system is easy-to-access and capable of carrying out scalable seed germination scoring for the following reasons: its low-cost and easy-to-build hardware design, its flexibility to incorporate different experiments, its open-source and modular software design, its scalability of traits analyses, and the availability of user-friendly GUI software, source code and design documents.

2.5.2 The SeedGerm Software Design

There is a growing need for standardising plant phenotyping in recent years, resulting in the ISA-Tab format[265], minimal Information About Plant Phenotyping Experiments (MIAPPE)[266], and ontology approaches to enable comparative

phenomics research[267]. Many studies[268–270] in seed phenotyping have employed bespoke data formats and data collection processes, limiting laboratories and external researchers to utilise and support these methods. Hence, when designing SeedGerm's software system, it was chosen to standardise the collection of image and sensor datasets following the ontological suggestions. Additionally, to calibrate images acquired by different SeedGerm devices, users are required to enter metadata to define their experiments, including experiment ID, biological replicates, genotype, experiment duration, and treatment.

To increase the scalability of the phenotypic analysis, I chose to implement my algorithms in Python instead of MATLAB as used in previously studies[271,272]. The reasons for this decision are that Python is easy-to-understand, cross-platform, and self-contained [273], which is supported by a wide range of open-source libraries such as Scikit-Image [274], OpenCV [275], Scikit-Learn [276], and Keras/TensorFlow [277]. Publicly available development kits have made upgrading the software relatively straightforward and this facilitates further development by the seed phenotyping community. For example, new crop species and traits can be added to the core analysis algorithm through new modules, where guideline seed morphological features can be predefined. Also, my colleagues and I followed a modular software design, so that modules developed for one species can be shared by other functions in analysis and parallel computing.

Recently, deep learning has become a powerful technique used by some seed germination analysis software [269,278,279], for which it was applied to extract features, segment seeds, and classify germination status. Although DL is relatively easy to implement through Python presently, the reasons I chose a combined computer vision (CV) and ML approach are: 1) DL requires a very large amount of training datasets to perform better than supervised ML and CV-based methods; for features that need to be engineered frequently such as varied seed germination experiments, DL might not be suitable. 2) normally one would need to build a dedicated DL model of each species; hence, it is time-consuming and ineffective to employ DL techniques for analysing a large number of crop species. 3) DL is likely to be overfitting for particular experiment settings and becomes problematic when conditions are changed. To allow the solution to be adopted by a broader research

community that has varied experimental settings, I chose supervised GMM, SGD and novelty detection learning techniques based on generalised feature selection. More importantly, by designing the ML models to reinitialise and retrain with background features at different establishment stages for each experimental setting, the learning models embedded in SeedGerm are dynamic and can be updated for each experiment, avoiding overfitting the learning models for a specific crop species or a particular experiment.

By employing CV algorithms, SeedGerm can also measure cumulative germination rates and seed morphologies such as size, width and length, extent and circularity to assess seed quality and seedling vigour, from germination to seedling. For example, I measured imbibition using the change of seed size, radical protrusion based on seed major/minor ratio, and germination speed through seed extent. If new biological questions are proposed, new traits and features could be designed jointly by biologists and computer scientists, instead of relying on DL techniques blindly. Because the SeedGerm software can be easily extended and accessed, I believe it is scalable and easy-to-access.

2.5.3 Applications of the SeedGerm Software

The results presented in this chapter demonstrate that SeedGerm is capable of scoring germination and measuring morphological changes automatically, for five major crop species and between different genotypes. The results show that SeedGerm could be employed to score germination frequency and seedling vigour, based on which the preformation of seed batches can be assessed. These traits were regularly checked by experienced seed engineers and scientists in order to provide certificates of seed germination and establishment performance in seed testing and seed insurance[280,281]. Hence, it is evident that SeedGerm has the potential to provide a replacement for manual assessment of germination frequency and radical emergence activities. Furthermore, as many traits measured by SeedGerm are highly correlated with seed performance and the effectiveness of post-harvest seed enhancement processes, SeedGerm could potentially contribute towards seeds certification, guidance on sowing density, or even seed insurance in the future.

Besides routine seed testing on germination frequency, the applications of SeedGerm could also be expanded to the seed vigour (i.e. how fast and uniform radical emergence) through monitoring morphological traits, which are important for estimating canopy closure, weed suppression, and crop yields through seed research[282–284]. Beyond existing traits analyses, the continuous phenotypic analysis can extend our insights into the entire physiological procedure of germination to understand phenotypic effects of individual seed and seed batches under dissimilar treatments. Furthermore, together with collaborators at the JIC, we also set up a range of experiments to score germination rates and timing across a diverse panel of *B. napus* varieties to demonstrate the biological relevance of SeedGerm as a research tool to measure the effect of genetics. It was shown that SeedGerm outputs can be used successfully for GWAS, identifying an association on *B. napus* chromosome A5 that explains the difference between high and low germinating varieties in the panel. Although the GWAS study identified associations over a 100kb region, this region does contain one gene BnaA05g27660D, a homologue of Arabidopsis *AHG3*, known to regulate ABA signalling during germination in Arabidopsis [259], which would be a strong candidate for further study. The low-germinating allele is only present in older spring varieties including Bronowski and Duplo, suggesting that it has been consistently selected against by modern oilseed rape breeders. Hence, I believe that SeedGerm has a great potential to have significant utilities in seed germination scoring and seed testing, for both research and routine seed technology applications.

2.5.4 SeedGerm's Challenges

It is also important to point out some edge cases where the system has struggled. Due to camera position and lighting problems, some image series were of poor quality. Although I added software calibration features to allow users to improve the classification accuracy on the low-quality datasets (e.g. colour features), the analysis could still suffer. For such datasets, only through manually selecting image IDs could I realistically reduce variance and improve the analysis accuracy. SeedGerm's scoring tends to overestimate the number of germinated seeds in crowded

experiments such as the later stages for Brassica and pepper experiments. To provide reproducible measures of the uniformity, timing and germination rates, I scored large numbers of seed samples and it was found that distancing the seeds from each other with at least 1 cm apart improved the analysis accuracy noticeably for crowded experiments. As different crop seeds have very diverse morphologies, some morphological measures cannot be easily transferred from one species to the other, which indicates the application of Online-Leaning or Transfer Learning mechanisms [285] could be potentially beneficial in future development. Although the learning models embedded in SeedGerm are dynamic for each experiment, the cost of such a design is that additional computational resources are required, demanding users to build a moderately powerful desktop computer (i7 CPU with 16GB memory) to perform analysis. Notably, to maintain the reliability of the parallel computing, I do not recommend more than eight tasks to be paralleled on an average computer, because processing multiple image series simultaneously requires a high demand of computing resources and some Python functions have been locked because they are not thread-safe during multi-thread processing.

2.5.5 Conclusion

In conclusion, the aim of my research was to address limitations in the current methodologies for seed imaging and scoring, especially regarding automated and scalable analyses of seed germination. To this end, the SeedGerm system was developed, demonstrating a synergistic combination of cost-effective hardware and user-friendly software. The system is adept at performing seed imaging and conducting machine learning-based analyses, successfully achieving the key objectives of this research.

The SeedGerm system was rigorously evaluated across numerous germination experiments, involving five distinct crop species. This extensive application of SeedGerm has proven its capability in meeting my objectives, particularly in offering a quantitative assessment of the performance of seed batches. Furthermore, the system’s proficiency in measuring vital morphological traits, including seed size,

width, length, extent, and circularity, aligns with the goals I originally set to provide insights into the physiological nuances of seed germination.

In testing the stated hypothesis, SeedGerm has demonstrated its alignment with seed specialists' observations in the accurate scoring of germination timing and rate. This not only validates the initial hypothesis but also affirms the system's biological relevance, evident from its role in identifying an ABA signalling gene in seeds, as revealed by associative transcriptomics by collaborators at the JIC.

The SeedGerm system, born out of this research's aims and objectives, has validated the original hypothesis about the potential of an integrated phenotyping system. The SeedGerm system stands as a testament to the possibilities of utilising computer vision and machine learning methods for seed germination phenotyping in agricultural crops. Given its wide-ranging applications, both for research purposes and industrial applications, SeedGerm paves the way towards a more automated and insightful future in seed testing and germination scoring.

While the SeedGerm system performed exceptonally, it is important to recognise the limitations inherent in my research. This study predominantly focused on image capture through an RGB camera and did not explore multispectral image analysis. As this area was unexplored, potential spectral features indicative of germination status or of seed vigour were not incorporated into the system.

Moreover, the challenges posed by radicles overlapping in later images were not fully mitigated. As the seeds began to germinate and radicles started to grow and intertwine, the complexity of the image increased, causing issues in accurately segmenting the individual objects. This overlapping of roots often resulted in seeds and their roots being segmented together, which impacted the precision of the extracted morphological features and the subsequent germination predictions – especially in the tomato and Brassica datasets.

2.6 Future Research

The field of seed germination phenotyping using computer vision and machine learning methods is rapidly evolving, and there are several exciting avenues for future research in this area. Here, I outline some potential areas of future research that could contribute to further advancements in the field.

A pivotal challenge encountered during the course of this study was the issue of seed overlap, particularly when radicles and roots came into contact. Addressing this problem is crucial for the accurate analysis of germination and seedling growth. Future research could focus on developing innovative solutions to tackle this issue, which may include the integration of advanced imaging techniques, machine learning algorithms, and novel hardware design. One potential avenue for exploration is the application of three-dimensional (3D) imaging technologies, such as structured light or laser scanning, to capture detailed spatial information of the seeds and their developing roots. By obtaining a more comprehensive representation of seed morphology and growth, researchers could more accurately differentiate between individual seeds and roots, even when they are in close proximity or overlapping.

Another approach could involve the development of advanced machine learning and computer vision algorithms, specifically designed to recognise and separate overlapping seeds and roots in image data. Deep learning techniques, such as CNNs, could be trained on large datasets of seed images with varying degrees of overlap, enabling the models to identify and disentangle seeds and roots in complex scenarios. Also, future research could explore the possibility of designing innovative hardware solutions, such as modified seed germination boxes or growth chambers, which incorporate features that minimise seed overlap. For instance, researchers could devise specialised seed positioning or spacing systems, or implement materials that discourage root entanglement, thereby reducing the likelihood of seed overlap and facilitating more accurate phenotypic analysis.

By addressing the challenge of seed overlap through a combination of these strategies, future research can pave the way for more accurate and reliable seed phenotyping systems, capable of handling complex germination scenarios and providing valuable insights into seed dormancy, germination rate, and overall plant growth.

Building upon the findings of this study, future research could utilise feature selection to identify the most significant attributes for predicting germination status accurately. This exploration would involve a systematic analysis of various morphological, physiological, and environmental features to determine their individual and combined impact on germination success. By identifying the most influential features, researchers can develop more efficient and precise predictive models for seed germination. One approach to feature selection could involve employing advanced statistical techniques, such as recursive feature elimination, principal component analysis, or correlation-based methods. These methods can help researchers identify the most critical features that contribute to accurate germination predictions, while simultaneously eliminating redundant or less relevant attributes. Once the most important features have been identified, they could be combined with deep learning methodologies, such as CNNs or recurrent neural networks (RNNs), to create powerful predictive models. By incorporating the most influential germination features into these deep learning architectures, researchers can improve the model's accuracy and generalisability across various seed types and experimental conditions.

Future research in seed germination prediction could benefit from the application of time series analysis methods, which focus on identifying and exploiting temporal patterns in morphological features to enhance prediction accuracy. Time series analysis offers a wealth of techniques that can capture the dynamic changes occurring during the seed germination process, allowing researchers to better understand the progression of germination events and their relationship to various factors. One potential avenue for investigation could be the application of time series modelling techniques, such as autoregressive integrated moving average (ARIMA) models, or more complex approaches like Long Short-Term Memory (LSTM) networks. By incorporating these methods, models that account for the temporal dependencies within the germination data could be developed. In conjunction with

time series modelling, deep learning time series image analysis approaches could also be employed to further enhance germination prediction accuracy. For example, researchers could utilise convolutional LSTM networks, which combine the strengths of CNNs for spatial feature extraction with LSTMs for modelling temporal dependencies. This hybrid approach can enable the model to learn both the spatial and temporal characteristics of the seed germination process.

Seed germination and phenotypic analysis could greatly benefit from the application of multispectral and hyperspectral imaging techniques as these advanced imaging methods have the potential to capture more detailed and comprehensive data, revealing new features that could be better predictors of germination and provide deeper insights into other seed phenotypic traits. Multispectral imaging involves capturing images at specific wavelengths across the electromagnetic spectrum, while hyperspectral imaging collects continuous spectral information for each pixel in an image. These techniques enable the acquisition of information beyond the visible spectrum, which could reveal new insights in the context of seed phenotyping. By utilising multispectral or hyperspectral imaging, researchers can identify previously unseen features and patterns that are directly related to germination and other important seed traits, such as seed vigour, disease resistance, or nutrient content. These features may include, but are not limited to, variations in seed coat colour or texture, the presence of specific biochemical compounds, or even the detection of early signs of germination invisible to the human eye. In addition to revealing new predictors of germination, these advanced imaging techniques could help uncover correlations between different seed phenotypic traits and germination performance.

Seed germination prediction accuracy could be improved by exploring the application of object detection deep learning models to automatically identify germinated seeds in imaging data. These models, which have already shown remarkable success in detecting and classifying objects in various fields, could potentially outperform traditional methods that rely on segmentation, feature extraction, and manual feature engineering. Object detection models, such as the region-based convolutional neural networks (R-CNNs) and You Only Look Once (YOLO), are capable of detecting multiple objects within an image, so by training

these models with annotated seed germination data, they could learn to recognise germinated seeds and distinguish them from non-germinated seeds.

The advantage of using object detection models over traditional segmentation and feature extraction methods lies in their ability to learn complex, spatial features and patterns in the data automatically. Instead of relying on manually engineered features, which might not always capture the intricacies of germination, object detection models can learn to recognise germinated seeds from various perspectives, scales, and lighting conditions. This adaptability could lead to more accurate and robust predictions of germination status. Additionally, object detection models can handle overlapping or touching seeds more effectively than segmentation-based approaches. These models are capable of identifying and classifying individual seeds even when they are in close proximity or partially obscured by neighbouring seeds, radicles, or roots. This feature can be particularly useful in situations where seeds overlap due to radicle and root growth, a common challenge faced in seed germination studies.

An exciting possibility for future research would be to investigate the potential of self-supervised learning techniques to detect seed germination from images. In the context of seed germination studies, self-supervised learning models could leverage the temporal structure of time-lapse image sequences to learn meaningful representations of seed development. One approach could be to train models to predict the next image in the sequence or to fill in missing frames, thus forcing the model to learn relevant features related to seed germination progress. Another possible approach would be to use contrastive learning, a self-supervised learning technique that encourages the model to learn representations that are similar for semantically similar images and dissimilar for semantically different images. In the case of seed germination, the model could learn to associate images of seeds at similar stages of germination while distinguishing them from images of seeds at different stages or non-germinated seeds without explicit labels.

After learning useful representations through self-supervised learning, these models could then be fine-tuned using a small set of annotated seed germination images to perform the actual germination detection task. By pre-training the model using self-

supervised learning, the need for large amounts of labelled data would be reduced and it could be possible to achieve better performance with less manual annotation effort, a key issue in seed germination phenotyping.

In conclusion, future research efforts in tackling seed overlap, the exploration of temporal methods, object detection models, and self-supervised learning, as well as novel hardware designs equipped with multispectral or hyperspectral capabilities could further advance the accuracy, robustness, and practical applicability of automated seed germination phenotyping systems. These advancements would accelerate seed phenotyping and therefore the identification of molecular markers that enable effective crop breeding strategies, therefore resulting in improved crop performance, yield, and sustainability.

Chapter 3: Trans-Learn – A Novel Approach to Exploit Spatial Gene Expression Interactions for Plant-Pathogen Diagnostics

3.1 Introduction

In this chapter, I present a novel pipeline, called Trans-Learn, for analysing gene expression datasets that utilises extensive feature selection methods, creative feature engineering, as well as deep learning methods. This project was a collaborative effort between the Earlham Institute and The Center for Ecological Research, Kyoto University, where my collaborators Dr Mie Honjo and Prof Hiroshi Kudoh generated the gene expression datasets and provided biological interpretation of the results.

My contribution to this project involved the solo development of the pipeline and software, including feature selection, feature engineering, model hyperparameter tuning, model selection, model ensembling, model interpretation, gene ontology analysis, gene network analysis, and development of the user interface. My collaborator Prof Ji Zhou provided ideas and support regarding method development.

To provide the necessary context for this chapter, I will first provide a focused literature review on plant-pathogen interactions and methods to detect transcriptomic biomarkers and associated genes.

3.1.1 Abstract

Machine learning and deep learning techniques have become increasingly popularly utilised in life sciences in recent years. However, their applications are still limited in addressing challenging problems that require exploiting multivariate and complex feature relationships to enable novel biological discoveries.

Here, I present Trans-Learn, a versatile analytic software platform that incorporates a range of supervised learning techniques. Amongst these techniques are CNN and ViT based gene identification approaches for predicting the presence of a given virus, virus abundance levels, and virus related genes using a variety of input data including transcriptomes, seasonal patterns, and virus levels.

The system has been applied to study turnip mosaic virus (TuMV), an important crop pathogen, and its associated defence genes using seasonal transcriptomes from *Arabidopsis halleri* subsp. *gemmifera*. To rigorously validate the effectiveness of the supervised ML methods and demonstrate the broader applicability of the system, Trans-Learn was further applied to various other RNA-Seq datasets. These datasets encompassed cancer subtype classification, COVID-19 detection, and wheat tissue type classification.

Following the training of Trans-Learn's models, a gene ontology analysis was conducted to examine highly relevant genes that exhibited a joint dependency on TuMV, as identified through the multivariate pattern analysis incorporated in Trans-Learn. This in-depth analysis facilitated the construction of custom gene dependency networks, providing valuable insights into host-virus interactions in dynamic natural environments.

Trans-Learn demonstrates the potential for supervised learning techniques in deciphering complex biological data, leading to a more profound understanding of various biological systems and offering promising applications across agriculture and diagnostics.

3.1.2 Plant Pathogens and Detection Methods

Crop diseases and viruses can have a significant impact on crop yield and development, posing a threat to global food security. These pathogens can infect crops at various stages of growth, from germination to maturity, causing a wide range of symptoms, including leaf wilting, stunting, discolouration, and fruit rot[286]. The severity of the impact depends on several factors, including the type of

pathogen, crop species, environmental conditions, and management practices. One primary way in which crop diseases and viruses affect yield is by reducing the photosynthetic capacity of infected plants[287]. Infected plants may have reduced leaf area, chlorophyll content, and photosynthetic efficiency, resulting in decreased production of sugars and ultimately reduced yield.

Additionally, some pathogens may directly attack reproductive structures, such as flowers or fruits, leading to reduced fruit set, poor seed quality, or even complete crop failure[288]. For example, fungal pathogens can infect flowers, causing them to abort, while viral infections can lead to malformed fruits and reduced seed production[289]. Viral diseases often cause serious damage to a wide range of crops and wild plants in both agricultural and natural ecosystems[290,291]. In order to safeguard against the spread of disease, considerable attention has been devoted to selecting virus-free plants and eliminating virus-infected plants [292]. Given that viruses are parasitic only to the living cells of the host where they propagate and proliferate[293], the quantity of viruses present in the hosts is often utilised to represent the potential risk of virus transmission to other healthy plants [294]. In both cases, the impact on crop yield can be substantial, and infected crops may need to be discarded, further exacerbating the impact on yield and food security. Consequently, to control viral diseases in ecosystems, it is vital to develop reliable methods to identify infected plants as well as measure the virus abundance level in plants.

The main biological focus of this chapter is on turnip mosaic virus (TuMV), a significant plant pathogen belonging to the *Potyvirus* genus, which causes considerable yield losses in various economically important crop species worldwide[295]. Infection by TuMV can result in a range of symptoms, including leaf mosaic patterns, stunting, malformation, and necrosis, ultimately leading to reduced crop quality and yield[296]. TuMV is prevalent in many regions of the world, particularly in temperate and subtropical climates where its primary host, Brassica crops, are grown. Besides Brassica species, such as turnip, cabbage, and cauliflower, TuMV is known to infect other plant species, including *Arabidopsis thaliana*, *Arabidopsis halleri*, and various ornamental plants, highlighting its wide host range[297].

Detecting plant pathogens using cutting-edge methods has therefore become a critical aspect of modern plant disease management. Advances in technology have enabled the development of innovative techniques that offer enhanced sensitivity, specificity, and speed for pathogen detection. These methods utilise various molecular, genomic, and bioinformatics approaches to identify and quantify plant pathogens with high accuracy and precision. One widely used technique for detecting plant pathogens is enzyme-linked immunosorbent assay (ELISA)[298]. ELISA is able to detect and quantify pathogens by specific binding of antibodies to their corresponding antigens present on the surface of pathogens and measuring a signal that is proportional to the amount of antibodies, which correlates with the pathogen quantity. However, ELISA is less relevant to my project, as the focus of the project is on utilising machine learning techniques to analyse quantitative gene expression data rather than antibodies.

One method for plant pathogen detection is polymerase chain reaction (PCR), a molecular technique that allows for the amplification and detection of specific DNA or RNA sequences of pathogens, offering high sensitivity and specificity, and enabling the detection of pathogens even at low levels[299]. Real-time PCR, also known as quantitative PCR (qPCR), is a variation of PCR that allows for the simultaneous amplification and quantification of pathogen DNA or RNA in real-time, providing rapid and quantitative results[300]. This method has been widely adopted for the detection of various plant pathogens, including viruses, bacteria, and fungi, owing to its high sensitivity, specificity, and rapid turnaround time[301]. Additionally, digital PCR (dPCR) is a newer technique that offers absolute quantification of pathogen DNA or RNA, providing precise quantification without the need for standard curves. This method partitions the PCR reaction into thousands of individual reactions, enabling the direct counting of target nucleic acid molecules present in the sample[302]. Due to its high precision and accuracy, dPCR has been utilised for the detection of various plant pathogens[303].

In a study by Robene et al.[304], the authors developed and validated new conventional and real-time quantitative PCR (qPCR) assays for the detection of *Xanthomonas citri pv. citri* (Xcc), the causative agent of Asiatic Citrus Canker. The newly developed assays targeted the Xcc XAC1051 gene and displayed enhanced analytical sensitivity

and specificity compared to previously published PCR and qPCR assays. The real-time PCR assay, XAC1051-2qPCR, included an internal plant control and demonstrated repeatability, reproducibility, and transferability between real-time devices. When tested with an extensive collection of target and non-target strains, both assays showed high analytical sensitivity and specificity. The XAC1051-2qPCR assay was also able to detect Xcc from herbarium citrus samples, indicating its potential utility for both applied and academic research on this bacterium.

More recent methods that utilise next-generation sequencing, and specifically RNA-Seq, has provided comprehensive, high-throughput sequencing of RNA samples, enabling the study of gene expression profiles and revealing the presence of both known and unknown pathogens. This approach offers several advantages over traditional methods, such as PCR, as it allows for the simultaneous detection, identification, and characterisation of multiple pathogens in a single experiment, thus enabling a more complete understanding of plant-pathogen interactions. The extensive data generated through RNA-Seq provides an opportunity to apply machine learning techniques to detect pathogens by identifying patterns in gene expression profiles that are associated with specific pathogens or disease states[305]. By integrating RNA-Seq data with machine learning algorithms, it is possible to develop predictive models that can accurately classify samples based on the presence or absence of pathogens, as well as estimate their abundance and distribution in the environment[306].

The application of machine learning to RNA-Seq data offers several benefits for plant pathogen detection. Machine learning can facilitate the discovery of novel biomarkers or genes associated with disease resistance or susceptibility, thus providing valuable insights into the molecular mechanisms underlying plant-pathogen interactions and informing the development of more effective disease management strategies[307]. Moreover, machine learning algorithms could be trained to recognise and adapt to the dynamic nature of pathogen populations, allowing for the detection of emerging pathogens and the monitoring of the evolution and spread of existing pathogens in response to changing environmental conditions and management practices. However, it is worth noting that considerably more research

has been performed in humans and mammals to develop diagnostic models that use gene expression data for important diseases and viruses[308].

An example of how RNA-Seq can enhance our understanding of plant pathogens is Chittem et al.'s study[309], which investigated the differential gene expression patterns of the fungal pathogen *Sclerotinia sclerotiorum*, responsible for Sclerotinia stem rot in canola (*Brassica napus*), during disease development on two canola lines with varying susceptibility. By sequencing RNA libraries from inoculated petioles and mycelium grown in liquid medium, the researchers identified genes differentially expressed during early and late infection stages on both susceptible and resistant lines. Gene ontology[310] (GO) categories associated with cell wall degradation, detoxification of host metabolites, and peroxisome-related activities were significantly enriched in up-regulated gene sets on both lines. The study highlighted the importance of peroxisome-related pathways, along with cell wall degradation and detoxification of host metabolites, as key mechanisms underlying the pathogenesis of S. sclerotiorum on *Brassica napus*.

Cutting-edge methods for plant pathogen detection, such as PCR and NGS, provide enhanced sensitivity, specificity, and speed for the detection and identification of plant pathogens, enabling early and accurate diagnosis for effective disease management. Continued advancements in technology are expected to further improve plant pathogen detection methods, leading to more efficient and sustainable plant disease management strategies.

Another approach that can detect viruses effectively is to measure plant responses using genome-wide gene expression analysis, i.e. transcriptome [311]. RNA sequencing (RNA-Seq) has become a quick and universal technique to obtain transcriptomic data from diverse plant species [312]. Because transcriptomes can represent the full range of messenger RNA (mRNA) molecules expressed from genes of an organism, useful information could be obtained as biomarkers to indicate dynamic and comprehensive plant responses to specific abiotic and biotic environmental factors. For example, gene expressions have been used as sensitive biomarkers to identify the status of nitrogen [313], water [314], and drought stress [315]. Furthermore, recent studies suggest that gene expressions could also signify the host plant's responses to virus

infection at early stages, prior to observable disease symptoms [316]. Hence, it is theoretically feasible to use transcriptome data as biomarkers for virus detection, based on which insights into virus-host interactions can be revealed and predicted. Nevertheless, in a natural environment where plant-virus interaction occurs, plants are often exposed to changeable environmental factors and the series of environmental stimuli are likely to modify the transcriptome of infected and uninfected plants in a complex manner [317]. Besides, natural plant populations consist of a diverse range of genotypes that can also affect transcriptome [318]. As a result, it is challenging to establish a plant-virus analysis platform based on transcriptome through traditional approaches, indicating the necessity of developing novel analytic solutions for such research objectives.

Digital gene expression technologies are not only making whole transcriptome analysis feasible, but also creating novel opportunities to analyse small RNAs for gene regulation, protein expression, and functional genomics [319]. A number of software packages have been widely used by the community to detect differentially expressed genes and target gene expression patterns from RNA-Seq data, for instance, edgeR [320], GENIE3 [321], DESeq2 [322], and SARTools [323]. These analysis tools include functions to normalise data, estimate dispersion, and perform differential expression analysis across genes together with systematic quality control to prevent analysis errors. Although these methods are capable of detecting candidate genes and expression patterns that are dependent on the specified treatment or condition (i.e. the target), the nature of univariate statistical approaches employed by some of these methods indicates that they would struggle to identify important multivariate relationships [324]. When subsets of genes (i.e. features) jointly have a strong dependence on the target but individually have a weak dependence, multivariate feature analysis is key to the discovery of these unknown relationships.

As a relative newcomer to life sciences, machine learning (ML) and deep learning (DL) techniques have brought new perspectives to many data-driven biological challenges [325]. ML/DL based techniques use statistics and sparse representations to build complex analytical procedures to progressively discover relationships between inputs and outputs with limited or even no human intervention [326]. Learning tasks such as classification, clustering, feature selection and regression have been

popularly applied to address a diverse range of biological problems, including the use of multiple cellular morphologies to classify varied cell cycle stages [327], organising gene products using DNA sequences [328], linking growth-related phenotypes to specific genotypes [329], and the prediction of yield traits to forecast crop production [330]. The key to successful ML/DL applications in life sciences is sufficiently sized input data, appropriately labelled training datasets, suitable learning algorithms, and well-defined targets, resulting in a generalisable and reproducible computational solution [331].

To date, ML and DL techniques have also been applied to classify gene expression and assess gene importance. For example, a genetic algorithm (GA) was used to select important genes based on their expression levels using k-means clustering [332], univariate feature selection was applied to classify highly relevant genes from omics data [333], Support Vector Machines (SVM) [334] and Artificial Neural Networks (ANN) [335] have been employed to select candidate genes from high dimensional biological datasets. More interestingly, by converting non-image samples (e.g. gene expression values) into image-based forms (e.g. feature matrices), DL models such as CNNs are being utilised to identify genes with high predictive power to classify tumour types [336], as well as to detect specific cancer trends through expression patterns (e.g. DeepInsight) [337]. The aforementioned studies show the power of deep learning in achieving novel biological discoveries using DNA sequences and RNA-seq datasets. Nevertheless, a recent review of univariate feature selection [338] indicates that systematically combining ML and DL is likely to produce more robust results from complex datasets that contain varying features, diverse distributions and multivariate patterns.

Multiple papers exploring the binary classification of infection using supervised machine learning algorithms on gene expression data exist [334,339], but none go on to link the amount of virus present to transcriptomics datasets. A common associated goal of classification is identifying the best biomarkers, which has resulted in literature concerning the gene selection process [340]. Statistical tests such as the Mann-Whitney U test [341] and dimensionality reduction through PCA [333] have been used to produce an effective subset of features for classification.

A study comparing the effectiveness of different feature selection methods on several datasets was performed by Li et al [342]. No individual method was found to be superior to the others as the best feature selection method depended greatly on the dataset as well as the chosen classification method. The authors address a key problem regarding feature selection – multicollinearity [343]. The subset of features that are selected as being relevant is likely to contain features that contain similar information. These features will appear relevant when looked at individually, however, when assessed together, the collection contains the same information as the individual.

Limited studies have been accomplished to measure multivariate patterns between genes of interest and targets such as infection status and virus severity; hence, the novelty of my work lies in the algorithmic development, the predictive modelling through a combined ML and DL approach, and the identification of genes that highly relevant to TuMV through feature selection and CNN/VIT-based host-virus association. The systematic application of ML and DL techniques is likely to open a new door for a broader plant research community to identify useful and new biomarkers that are challenging to obtain through traditional approaches.

3.1.3 Machine Learning Methods to Predict Diseases

Machine learning has emerged as a powerful approach for predicting disease infection using gene expression data, providing valuable insights into the molecular mechanisms underlying various conditions and enabling the identification of potential diagnostic biomarkers and therapeutic targets. In recent years, numerous studies have leveraged machine learning algorithms to analyse gene expression datasets and develop predictive models for a wide range of diseases[344].

In the area of applying machine learning methods to gene expression datasets, the biomedical field has performed significantly more studies than in plant research[345]. Notably, supervised machine learning methods have been extensively applied to cancer gene expression datasets, enabling researchers to develop predictive models for cancer classification, prognosis, and treatment response. These approaches have

provided valuable insights into the molecular mechanisms underlying various types of cancer and have contributed to the identification of potential diagnostic biomarkers and therapeutic targets[346].

A study by Golub et al.[347] utilised supervised machine learning to classify acute leukaemia subtypes based on gene expression data. The authors analysed gene expression profiles of bone marrow samples from patients with acute myeloid leukaemia (AML) and acute lymphoblastic leukaemia (ALL) using DNA microarrays. They then applied a weighted voting algorithm to classify the samples based on the expression patterns of a selected set of genes. The resulting classification was highly accurate, with only one misclassification in the original dataset and similarly high performance in an independent validation set. This study demonstrated the potential of machine learning methods for accurate cancer classification based on gene expression data, however the sample size was concerningly low so the generalisability of the model could be questionable.

Shi and Zhang[348] address the challenge of small sample sizes in gene expression profiling for cancer outcome prediction. Due to the limited availability of labelled data, traditional supervised learning techniques have faced difficulties in developing robust and accurate classifiers for cancer prognosis. As a result, a large amount of microarray data with insufficient follow-up information remains unused. To overcome this issue, Shi and Zhang utilised a semi-supervised learning technique called low density separation (LDS)[349]. They demonstrated the effectiveness of this method in predicting recurrence risk in colorectal cancer patients, with the results showing that semi-supervised classification using LDS improved prediction accuracy compared to the state-of-the-art supervised methods. As expected for semi-supervised approaches, the performance gain increased with the number of unlabelled samples, indicating that leveraging unlabelled data can significantly enhance the predictive capability of cancer prognostic models.

In the context of plant pathology, machine learning has rarely been employed to predict plant diseases and identify potential resistance genes using gene expression data. Therefore, I will focus on an example where supervised machine learning has been used to predict a different complex trait using gene expression data.

A significant recent study by Cheng et al.[350] tackles both the difficulty of predicting phenotypic traits based on gene expression data and confirming functional relevance of identified biomarker genes. They employed a machine learning technique informed by evolutionary knowledge to predict phenotypes, taking advantage of transcriptome responses to nitrogen treatments that are shared within and between species, specifically in Arabidopsis accessions and maize varieties. This strategy allowed the researchers to capitalise on the phenotypic diversity of nitrogen use efficiency (NUE) and transcriptome responses conserved through evolution. Concentrating on nitrogen-responsive genes conserved evolutionarily, they managed to decrease the feature dimensionality in machine learning, ultimately enhancing their gene-to-trait model's predictive capacity. In this study, the authors also functionally confirmed seven transcription factors in Arabidopsis and one in maize that exhibited predictive power for NUE outcomes. This demonstrated the efficacy of their approach informed by evolutionary knowledge in pinpointing genes with a significant influence on the phenotype under investigation. Moreover, the authors emphasised the potential of their evolutionarily informed pipeline for application in other species, including rice and mice models. This method has enormous potential as if biomarker genes that have been conserved evolutionary can be identified across additional crop species, a universal set of biomarker genes could be used to train a supervised machine learning model that could be used to assay global crops globally.

In conclusion, machine learning has shown great promise in predicting diseases and viruses in humans, and these techniques could be transferred to plants, creating valuable tools for early detection and management of plant diseases.

3.1.4 Methods to Detect Associated Genes

Statistical methods play a critical role in identifying genes associated with plant pathogens and viruses, providing valuable insights into the molecular mechanisms underlying plant-pathogen interactions. These methods utilise various statistical approaches, including differential gene expression analysis, gene ontology analysis, and machine learning algorithms, to identify genes that are differentially expressed or associated with plant-pathogen interactions.

Differential expression analysis is a widely used statistical approach in transcriptomics that aims to identify and quantify changes in gene expression levels between different experimental conditions or groups, such as healthy versus diseased plants, or plants exposed to different environmental stresses[351]. This method is essential for understanding the molecular basis of various biological processes, including responses to pathogen infection, adaptation to environmental changes, or the regulation of developmental processes[352]. Differential expression analysis begins with the generation of gene expression data that estimates the expression levels of individual genes using high-throughput techniques, such as RNA-Seq or microarrays. Once RNA-Seq data has been processed into a gene expression matrix, data is typically normalised to account for technical and biological variability, and statistical tests are employed using packages such as DESeq2[145] and edgeR[169] to identify genes that exhibit significant changes in expression between the groups of interest.

DESeq2 and edgeR are two of the most used Bioconductor packages in R for the analysis of differential gene expression from high-throughput RNA sequencing (RNA-Seq) data. Although these methods share some similarities, they differ in their underlying statistical models and normalisation techniques.

DESeq2 is based on a negative binomial generalised linear model (GLM) that models the read counts for each gene as a function of experimental conditions or factors, while accounting for biological variability between replicates. The negative binomial distribution is used to model count data because it allows for overdispersion, which is often observed in RNA-Seq data due to biological variability and technical noise[353]. The DESeq2 method estimates dispersion parameters for each gene by fitting a local regression to the mean-variance relationship, and then uses the Wald test to assess the significance of the differences in expression between conditions.

EdgeR also employs a negative binomial model, but instead of using a GLM, it relies on an empirical Bayes estimation of gene-specific dispersion parameters. The method uses a normalisation technique called the Trimmed Mean of M-values

(TMM) to account for differences in library size and composition between samples[354]. After normalisation, edgeR performs an exact test, based on the negative binomial distribution, to determine the significance of differences in expression between experimental conditions.

In summary, DESeq2 and edgeR are both popular methods for differential expression analysis of RNA-Seq data, employing negative binomial models to account for count data and overdispersion. While DESeq2 uses a generalised linear model and the Wald test, edgeR relies on an empirical Bayes estimation and an exact test based on the negative binomial distribution. Both methods provide robust and accurate identification of DEGs in RNA-Seq experiments. One of the key challenges in differential expression analysis is distinguishing true biological changes from random noise or experimental artefacts. To address this issue, researchers often use multiple testing correction methods, such as the Benjamini-Hochberg procedure [355] or the Bonferroni correction[356], to control the false discovery rate and minimise the risk of identifying false. When considering differential expression analysis, the choice of appropriate statistical models and tests, as well as careful experimental design, is critical to ensure accurate and reliable results.

After identifying differentially expressed genes, they can then be further analysed to reveal insights into the biological processes or pathways that are affected by the experimental conditions, using tools such as gene ontology (GO) analysis or pathway enrichment analysis. GO analysis is a powerful bioinformatics tool used to systematically annotate and categorise genes based on their molecular functions, biological processes, and cellular components. This comprehensive and structured vocabulary enables researchers to identify the roles and relationships of genes within an organism, and to compare gene function across different species. GO analysis is essential for interpreting high-throughput data generated by methods such as RNA-Seq or microarrays, as it helps to organise and make sense of the large number of differentially expressed genes identified in these experiments[357]. In the context of plant diseases, GO analysis can be employed to uncover genes associated with defence response and disease resistance. By analysing the functional categories enriched among differentially expressed genes in plants exposed to pathogens, researchers can identify key genes and pathways involved in the plant's response to

infection[358]. This information can further our understanding of the molecular mechanisms underlying plant-pathogen interactions and provide valuable insights for breeding resistant plant varieties or designing targeted agrochemical interventions[359].

As well as gene ontology analysis as a method to analyse differentially expressed genes or candidate genes, network analysis methods are becoming increasingly popular as tools for analysing gene interactions as well as inferring gene regulatory relationships in a systems biology context. These methods allow researchers to construct and visualise complex gene networks, providing insights into the organisation and function of biological systems[360]. By modelling gene interactions as networks, it becomes possible to identify key regulatory genes, predict their targets, and in the case of plant pathogens uncover the mechanisms underlying various cellular processes and disease states.

A commonly used network inference algorithm is GENIE3[361], which combines regression trees with ensemble learning to predict gene regulatory networks from gene expression data. GENIE3 identifies potential regulatory relationships by ranking the importance of each gene in predicting the expression levels of its potential target genes. This method has demonstrated high accuracy in the DREAM5 challenge, a community-based competition for gene network inference, and has been successfully applied to various biological systems, including yeast, bacteria, and mammalian cells[362].

Cytoscape, an open-source software platform, is another valuable tool for visualising and analysing complex gene networks[363]. Cytoscape enables researchers to integrate gene expression data with other sources of biological information, such as protein-protein interactions, to generate comprehensive network representations of biological systems. The platform also provides a wide range of plug-ins and apps for performing advanced network analysis tasks, including network clustering, functional enrichment analysis, and network comparison. Cytoscape has been extensively used for studying various biological processes, such as signal transduction, metabolic pathways, and gene regulatory networks, as well as for investigating the molecular basis of complex diseases[364].

While differential expression analysis involves identifying genes that exhibit significant changes in expression between experimental conditions, machine learning methods offer an alternative approach for the identification of biomarkers in gene expression datasets. Both strategies aim to reveal insights into the molecular mechanisms underlying biological processes or disease states, however, machine learning techniques can provide additional benefits by uncovering complex patterns and relationships that may be difficult to detect using traditional differential expression analysis alone. Machine learning algorithms can be trained on gene expression data to classify samples based on their phenotypic characteristics or disease status[365]. In contrast to differential expression analysis, which focuses on individual gene expression changes, machine learning models can capture higher-order interactions between genes and incorporate these into their predictions. By interpreting the trained models and examining feature importance, researchers can identify potential biomarkers that are functionally relevant or indicative of the underlying biological mechanisms[366].

Feature selection techniques can be employed alongside supervised machine learning methods to systematically identify a subset of predictive genes that contribute the most to the classification task[367]. These techniques help to eliminate redundant or irrelevant features, thus improving the interpretability and generalisability of the model, and facilitating the identification of potential biomarkers. This contrasts with differential expression analysis, which typically ranks genes based on their statistical significance, without considering the potential redundancy or combinatorial effects between genes. Unsupervised machine learning algorithms, such as clustering or dimensionality reduction techniques, can also be applied to gene expression datasets to identify groups of co-expressed genes or samples with similar expression profiles[368]. While differential expression analysis identifies individual genes with significant expression changes, unsupervised learning methods can reveal broader patterns in the data that may be indicative of coordinated gene regulation or shared biological functions.

Several studies have successfully employed machine learning, differential expression, or gene ontology analysis to identify biomarker genes. In a study by Alon et al.[369], unsupervised clustering analysis was used to uncover patterns of gene

expression in tumour and normal colon tissues, leading to the identification of genes associated with colon cancer. Similarly, van ‘t Veer et al.[370] utilised supervised machine learning algorithms to identify a 70-gene expression signature that could predict breast cancer outcomes, demonstrating the potential of these methods for prognostic applications. In another study, Subramanian et al.[371] introduced the Gene Set Enrichment Analysis (GSEA) method, which combines differential expression analysis with gene ontology information to identify functionally related gene sets associated with a specific phenotype. This approach has been widely adopted to uncover the biological processes underlying various diseases and to identify potential biomarker genes.

In conclusion, statistical and machine learning methods play a crucial role in identifying genes associated with plant pathogens and viruses, providing valuable insights into the molecular mechanisms underlying plant disease. Differential gene expression analysis, machine learning algorithms, network analysis, and gene ontology analysis are among the commonly used approaches for analysing transcriptomic datasets. These methods can enable the identification of genes that are differentially expressed or associated with plant-pathogen interactions, providing a deeper molecular understanding of these complex interactions.

3.1.5 The Application of Image Analysis Methods to Tabular Datasets

Gene expression datasets or gene expression matrices are represented in a tabular format, a commonly used format for storing and organising data. These datasets are typically organised in rows and columns, similar to a spreadsheet. Each row in a gene expression dataset represents an individual sample, while each column represents a specific gene. The elements within the matrix represent the level of expression of a gene for a given sample, typically normalised using the transcripts per million (TPM) method to account for differences in library size and gene length, making it possible to compare gene expression levels across samples[372].

Transforming tabular datasets into image formats has emerged as a novel approach that can offer new avenues for utilising supervised machine learning methods to

develop predictive models. The conversion of such tabular datasets into image format, commonly referred to as image representation or data encoding, can provide a unique perspective on the data and harness the potential of image-based machine learning techniques. The rationale behind this transformation is that image representation can effectively capture spatial relationships and patterns within the data, which might be overlooked when analysing tabular datasets using traditional machine learning algorithms[373]. By converting the data into images, it is also possible to leverage advanced image-based machine learning techniques, such as convolutional neural networks (CNNs) and vision transformers (ViTs), which have demonstrated superior performance in various image classification and recognition tasks[374].

One common approach for transforming tabular datasets into image format is by converting each sample into a grid by representing the tabular data as a matrix, with rows and columns forming a grid, and the values in the cells of the grid representing the data values. Once the tabular data has been converted into a grid-based representation, it can be further processed into an image format suitable for image-based supervised machine learning algorithms. This can be achieved by encoding the grid data as a greyscale or colour image, where the intensity or colour of each pixel corresponds to the value in the respective cell of the grid.

One example of the successful application of image representation is in the analysis of gene expression data. In this context, tabular datasets representing gene expression levels can be transformed into image formats, enabling the use of CNNs to identify patterns associated with specific phenotypes or disease states. This approach has been shown to improve the performance of predictive models by capturing complex relationships between genes and phenotypic traits that might not be apparent when using conventional machine learning methods. For example, in a study conducted by Rukhsar et al.[375], RNA-Seq data for five cancer types from the Mendeley data repository were analysed by converting the tabular RNA-Seq dataset into 2D images, applying normalisation and zero padding. Subsequently, relevant features were extracted and selected using convolutional neural networks and methods to interpret the trained weights. Lastly, classification was performed on the

test data using eight different state-of-the-art deep learning architectures, with the authors' custom architecture CNN achieving the highest level of accuracy.

A further study that served as an inspiration for my research was conducted by Lyu et al[376], who employed a convolutional neural network in the analysis of gene expression data for various cancer subtypes. In their investigation, they transformed high-dimensional RNA-Seq data into 2-D images and made use of a convolutional neural network to categorise 33 distinct tumour types. Their methodology achieved a remarkable final accuracy of 95.59%, outperforming another study utilising a genetic algorithm and k-nearest neighbours method on the same dataset. Moreover, Guided Grad-CAM (Gradient-weighted Class Activation Mapping) [377] was applied to create significance heatmaps for all genes within each category. A functional analysis of genes exhibiting high intensities in these heatmaps verified that the top genes were associated with tumour-specific pathways, and some had already been identified as biomarkers in other studies, validating the efficacy of their approach. Significantly, Lyu and Haque were the first to utilise a convolutional neural network to analyse the Pan-Cancer Atlas dataset for classification and to link classification significance with gene importance, illustrating the potential applicability of their method to other transcriptomics data.

In addition to the grid method of transforming tabular data into images, Sharma et al. developed a method called DeepInsight, which transforms features vectors into a two-dimensional matrix of Cartesian coordinates using unsupervised techniques such as t-SNE and kernel PCA. The resulting Cartesian coordinates represent the gene features in the data. A convex hull is constructed from the set of points in Cartesian space, and some final pre-processing is performed to prepare an image of the points located inside the convex hull for input in supervised image analysis models. By intelligently arranging similar genes closely into groups, multivariate patterns become more readily accessible to deep learning models that utilise spatial patterns, enabling the identification of hidden relationships as compared to analysing genes individually. When a CNN was trained on gene expression images generated by DeepInsight, the model scored a classification accuracy of 99% on a holdout test dataset, surpassing the best-performing random forest model. This method can be generalised to other tabular datasets not necessarily containing gene expression data

and could also be extended to cases involving multi-omics datasets to discover additional hidden multivariate patterns concerning protein expression, methylation, and other biological phenomena.

Expanding upon their research that produced DeepInsight[337], Sharma et al. subsequently developed a computational pipeline, DeepFeature[378], which transforms tabular omics data into an image format optimally suited for supervised neural networks that can exploit spatial relationships to accurately predict a target. In addition to optimising the transformation of the input data, the pipeline can reverse-engineer a trained CNN, that has the SqueezeNet architecture in their study, to identify the most important genes for biological interpretation using class activation maps (CAM)[379]. In contrast to DeepInsight, DeepFeature[378] applies the Snowfall compression algorithm that can be combined with the existing unsupervised t-SNE technique of transforming the tabular data. The Snowfall method aims to minimise the overlap of features within the image, thereby increasing the number of features exposed to the supervised image analysis model. The DeepFeature pipeline demonstrated a 98% accuracy in classifying 10 cancer subtypes within The Cancer Genome Atlas (TCGA) dataset[380], surpassing the performance of the most effective traditional supervised machine learning approaches with the combination of t-SNE and Snowfall pre-processing yielding the most accurate results. Moreover, the genes that were identified using cancer subtype specific class activation maps significantly overlapped with established biomarkers and cancer pathways, validating the biomarker identification aspect of the pipeline.

In conclusion, transforming tabular datasets into image format can offer a novel approach for using supervised machine learning methods to train predictive models. Grids that represent sample vectors as matrices, as well as unsupervised pipelines such as DeepInsight and DeepFeature are some of the methods that can be used to represent tabular data as images. Leveraging image-based representations with image-based supervised machine learning techniques, such as CNNs and ViTs, can provide new insights and opportunities for modelling complex patterns in tabular data, enhancing the predictive capabilities of the models.

3.2 Aims and Objectives

At the start of this project, the aim was to produce a machine learning model that could accurately predict whether a sample was infected with TuMV or not. After producing a model capable of doing this, the aims of the project shifted towards maximising the predictive performance of the Trans-Learn software, identifying groups of genes which jointly depend on TuMV, and comparing the performance of my method with DeepInsight.

The aim of the Trans-Learn project was to develop a novel method for analysing tabular transcriptomic datasets through novel feature encoding, feature selection methods, and image analysis techniques with a specific focus on identifying biomarkers associated with turnip mosaic virus (TuMV) in *Arabidopsis halleri*.

Secondly, my objective was to develop novel feature encoding methods to transform tabular datasets into an image format that is optimal for image analysis based supervised neural networks. To accomplish this, I intended to explore methods of arranging elements within a grid format to emphasise patterns between genes and provide spatial relationships to a CNN or ViT. It is important that this method is computationally-efficient and beneficial if applicable to a wide range of gene expression and other tabular datasets.

An important objective was to implement feature interpretation techniques on the encoded image data to identify relevant biomarkers associated with TuMV infection in A. helleri and other datasets. Using methods from the literature, I intended to use Grad-CAM or CAM to extract the most important features from the trained computer vision neural network. The aim is to reduce the dimensionality of the encoded image data while retaining the relevant information, thus improving the efficiency and effectiveness of the subsequent machine learning models.

A key objective was to develop supervised machine learning models using the tabular and encoded image data to predict TuMV infection as well as targets in other gene expression datasets. As well as exploring standard machine learning methods

such as gradient boosting machines and logistic regression, I planned to utilise convolutional neural networks (CNNs) and ViTs, widely used deep learning technique for image processing tasks. Once trained, I planned to compare the performance of the different models across multiple datasets to identify the best performing method for predicting targets using gene expression data.

Another objective was to analyse identified associated genes or biomarkers using gen ontology analysis as well as network analysis, such as inferring gene regulatory relationships. To accomplish this, I planned to use existing software and packages such as Cytoscape in order to gain biological insights into the results of my pipeline.

The final objective of this project was to develop open-source Python software for implementing the proposed methods, making them accessible and reproducible for the scientific community. The software will include modules for encoding tabular datasets into image format, implementing feature selection techniques, and training supervised machine learning models for biomarker identification. The software will be documented and made available on a public code repository, facilitating ease of use due to its user-friendly interface, customisation, and further development by other researchers and practitioners in the field of plant pathology.

In summary, this chapter aims to develop a pipeline based on computer vision and machine learning methods for analysing tabular gene expression datasets through novel feature encoding, selection, and interpretation methods, with a specific objective of identifying biomarkers associated with TuMV in *Arabidopsis halleri*. The development of open-source Python software will contribute to the scientific community by providing accessible and reproducible tools for plant pathology research, as well as enabling the identification of biomarkers and development of predictive models for other traits. The outcomes of this research can have potential applications in early detection and management of TuMV infection in *Arabidopsis halleri*, contributing to the field of plant pathology and advancing our understanding of host-virus interactions.

This chapter has some focus on a specific biological process the transcriptional response to TuMV in *Arabidopsis halleri*, however as this is just a case study with

my focus being on the method development, a suitable hypothesis for this study could be: "The developed computational pipeline, employing computer vision and machine learning approaches for encoding, selecting, and interpreting features from tabular gene expression datasets, will effectively identify significant multivariate patterns and relationships within the data, demonstrating its potential as a powerful and reliable predictive tool for diagnostic purposes and understanding gene expression dynamics in various biological contexts."

The underlying assumption of this hypothesis is that the developed pipeline will have the ability to exploit multivariate relationships effectively and precisely in gene expression datasets, relying on the features generated through the encoding method. The hypothesis suggests that combining computer vision and machine learning methodologies will create a robust system capable of offering understanding of gene expression dynamics, thus benefiting the domains of functional genomics and systems biology. To evaluate the system’s accuracy and trustworthiness in providing insights and interpretations, the hypothesis will be subjected to validation procedures using a range of gene expression datasets from various biological settings.

3.3 Methods

Here, I introduce my methods to analyse gene expression datasets through novel feature encoding, selection, and interpretation techniques. Comprehensive procedures for data acquisition, pre-processing, experimental configurations, and automated analysis are described to offer a thorough understanding of the methodologies applied. Moreover, specific criteria for feature encoding, selection, and interpretation for each gene expression dataset are detailed, along with the experimental design established to guarantee consistent and replicable outcomes. This portion acts as a blueprint for understanding and reproducing the results of this investigation. The primary components of the Trans-Learn methodology include the normalisation of gene expression data, the selection of highly predictive genes, feature encoding, model training, and the interpretation of genes deemed to have high multivariate importance for making predictions (Figure 9).

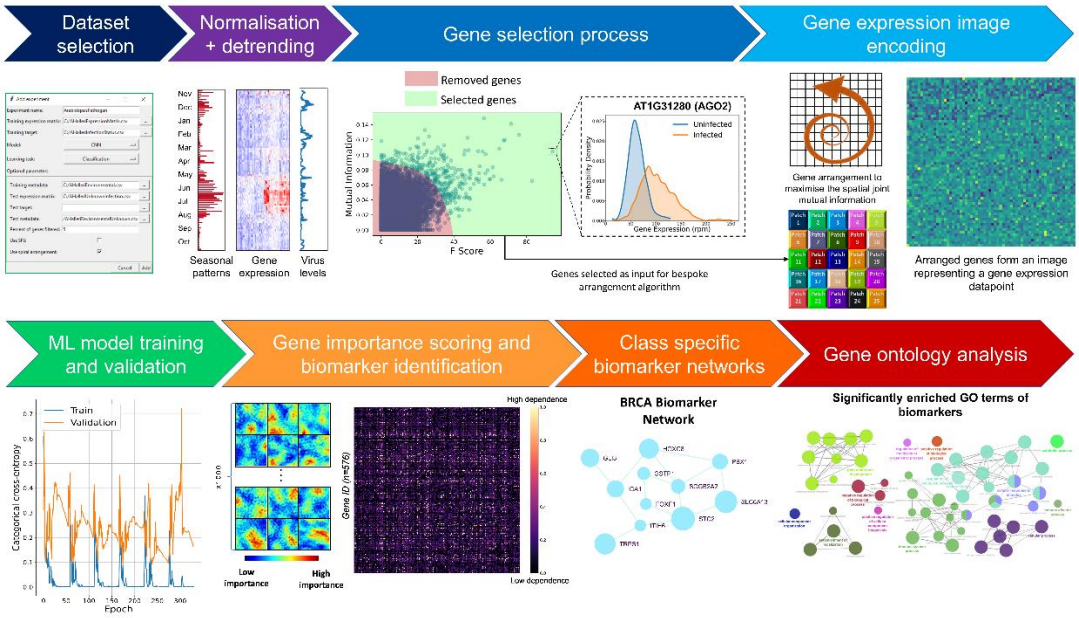

Figure 9.

The figure represents the workflow of the Trans-Learn software to identify multivariate biomarkers and produce highly accurate predictions. Read from top left to bottom right, 1) labelled gene expression dataset selection using the GUI, 2) expression data normalisation and seasonal detrending, 3) gene selection using filter feature selection, 4) gene expression image encoding, 5) model training and validation, 6) gene importance quantification to

identify multivariate biomarkers, 7) multivariate biomarker network creation, 8) gene ontology analysis of biomarkers

3.3.1 RNA-Seq Datasets

TuMV is one of the most well-characterised plant viruses and an important crop pathogen as a wide range of plant species (e.g. the family of Brassicaceae) can be infected by the virus. TuMV was chosen as the study system as well as *A. halleri*, a wild plant of Brassicaceae distributed throughout East Asia (including eastern China, Japan, Korea and far eastern Russia). My collaborators, Dr Mie Honjo, and Prof Hiroshi Kudoh, conducted in-field tissue sampling at Omoide gawa (35°06' N, 134°55' E, 190–230 m in altitude) and Monzen study sites (35°05' N, 134°54' E, 140–150 m in altitude) during all seasons of the year in Japan. Both sites are 3.5 km apart, with *A. halleri* growing in open spaces next to small streams running through secondary forests. To generate transcriptomic data, the host-virus system (i.e. *A. halleri – TuMV*) served as a natural model system due to perennial and ever-green habits of *A. halleri*. Conveniently, close relatedness of *A. halleri* to *A. thaliana* allowed for the annotation of gene functions using the molecular information of the model plant. The data used for this study was collected in my collaborators' long-term study sites, central Honshu, Japan.

The seasonal dataset was obtained weekly from July 2011 to September 2017, which consists of six weekly plant tissues sampled in the Omoide River site (sample ID 1-490) and four sets of 48-hour samples taken at intervals of two hours in the Monzen site (sample ID 491–873). The 48-hour sampling (starting from 16:00 on the first day) was performed at the spring equinox (i.e. 19th-21st March), the summer solstice (i.e. 26th-28th June), the autumn equinox (i.e. 24th-26th September) and the winter solstice (i.e. 12th-14th December). In each set, half were continuously collected from two clonal patches and the other half were collected from diverse clonal patches. TuMV abundance and transcriptome data (i.e. DRA005871, DRA005872, DRA005873, DRA005874, DRA005875) were obtained from the leaf samples by RNA-Seq. Gene expression and virus quantity were quantified in relation to the total reads of host mRNA[317]. Meteorological data (metadata associated to the

transcriptome) was obtained from the nearest weather station in Nishiwaki, Japan (Meteorological Agency ID 63331, 34° 59.9’ N, 134° 59.8’ E, 72 m in altitude), including ambient temperature (°C), precipitation (mm), daylight hours (h) and wind speed (m/s) measured on an hourly basis.

To cover a broad range of environmental fluctuation, I used both seasonal (n=874, m=32,648) and diurnal (n=554, m=32553) datasets obtained from previous studies by my collaborators[381]. The data came from a mixture of natural-growing, diverse *Arabidopsis halleri* plants. Because the two datasets were collected using slightly different techniques, they contain dissimilar sample sizes and gene numbers. These datasets were combined into one large expression matrix using the intersection of the two sets of genes, resulting in an expression matrix X that had dimensions [1428, 31571]. For the label that characterised the TuMV infection status, my collaborators quantified the presence of TuMV in the RNA-Seq samples in reads per million (RPM), providing a continuous target vector of size [1428]. The following equations describe the dimensions of X, the gene expression matrix, and Y, the target vector. In these equations, n denotes the number of samples, and m denotes the number of genes.

$$\boldsymbol{X} = \begin{pmatrix} X_{0,0} & \dots & X_{0,m} \\ \vdots & \ddots & \vdots \\ X_{n,0} & \dots & X_{n,m} \end{pmatrix}, \qquad \boldsymbol{Y} = \begin{pmatrix} Y_0 \\ \vdots \\ Y_n \end{pmatrix}$$

As well as using RNA-Seq data using the TuMV-*A.halleri* study system, I also identified datasets that I could use to benchmark models that I trained. First, I identified a popular dataset for benchmarking machine learning applications to gene expression datasets, the TCGA dataset that consists of thousands of expression samples, each labelled with different cancer subtypes. The selected dataset consisted of 10 cancer subtypes, including Breast Invasive Carcinoma (BRCA), Colon Adenocarcinoma (COAD), Head and Neck Squamous Cell Carcinoma (HNSC), Kidney Renal Clear Cell Carcinoma (KIRC), Brain Lower Grade Glioma (LGG), Lung Adenocarcinoma (LUAD), Lung Squamous Cell Carcinoma (LUSC), Prostate Adenocarcinoma (PRAD), Thyroid Carcinoma (THCA), and Uterine Corpus Endometrial Carcinoma (UCEC). These represent a diverse range of cancers, with

several being closely related, such as LUAD and LUSC meaning that the model will need to accurately identify subtle patterns in gene expression to separate these subtypes. This expression dataset comprised 6,280 samples and 60,483 genes, resulting in a gene expression feature matrix X with dimensions [6,280, 60,483] and a cancer subtype target vector Y with dimensions [6,280].

Another dataset that was used to benchmark Trans-Learn's models was a publicly available COVID-19 dataset from a study by Lieberman et al.[382] that contained gene expression data of humans that were infected and uninfected with the COVID-19 virus. As well as the gene expression data, metadata describing the age and gender of the person was available, potentially giving additional context and information to a supervised machine learning model. In this dataset, the class frequency distribution of infected and uninfected people was highly imbalanced, with 430 of the individuals having SARS-CoV-2, and 54 uninfected negative control samples. Therefore, this expression dataset comprised 484 samples and 35,784 genes, resulting in a gene expression feature matrix X with dimensions [484, 35,784] and a COVID-19 status target vector Y with dimensions [484].

Finally, I utilised a wheat RNA-Seq dataset generated by my colleagues at the Earlham Institute that contained six different tissue types: leaf at dusk, leaf at dawn, grain, spike, root, and flagleaf, across sixteen different wheat varieties. This dataset was generated as part of a different project to identify tissue specific expression differences between cultivars and identify clusters of varieties that behaved similarly. However, when the dataset was generated, a proportion of the samples were presumed to be mislabelled as after performing principal component analysis, the tissue types formed clear clusters, but some biological replicates were found to belong to clusters containing different tissue types. For each variety and tissue type combination, there were approximately three samples yielding 271 samples, each containing expression level values for 99,364 genes, resulting in a gene expression feature matrix X with dimensions [271, 99,364] and a tissue type feature vector Y with dimensions [271]. As well as the individual samples, a pooled expression dataset using pooled RNA was generated which when plotted after performing PCA, generated six separate clusters – one for each tissue type (dusk, dawn, grain, spike,

root, and flagleaf). A supervised model could be used to learn a relationship between the expression data in the pooled dataset to the respective tissue types and this mapping can then be applied to the individual samples to obtain tissue type predictions.

The majority of the analysis was performed on the TuMV-A. Halleri plant-pathogen model dataset to reveal molecular markers associated with pathogen defence. The purpose of the other cancer, COVID-19, and wheat tissue type gene expression datasets was to rigorously validate my proposed methodology (Figure 10).

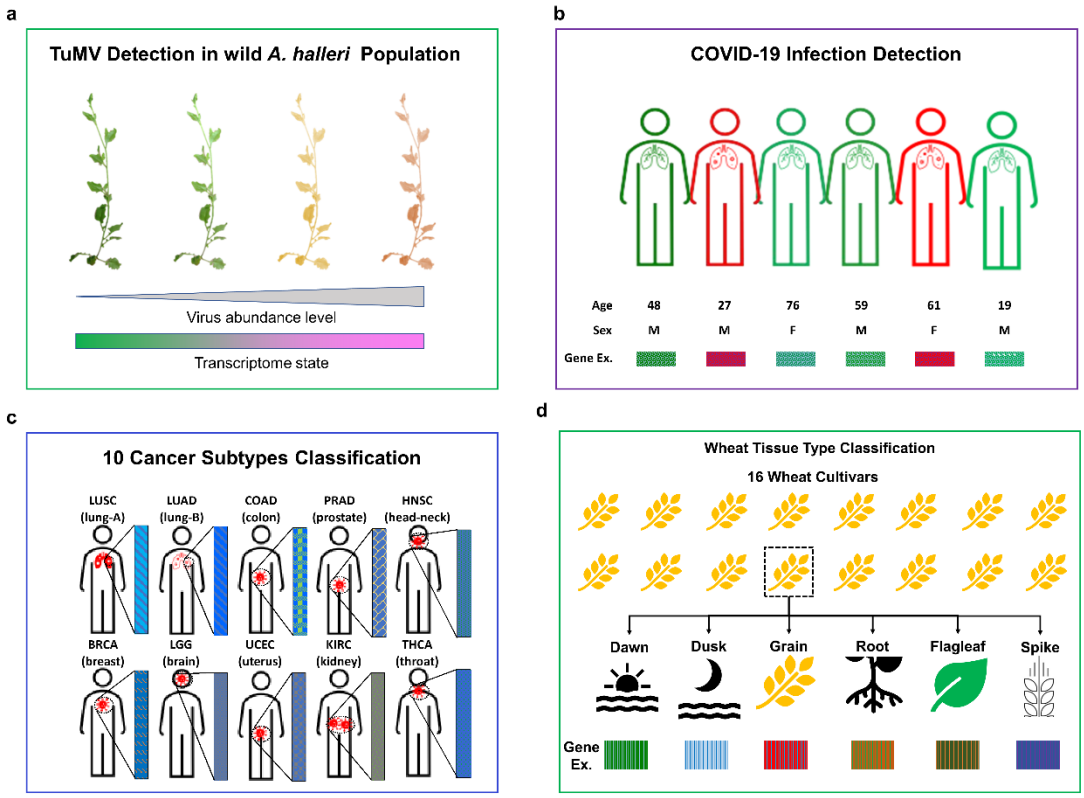

Figure 10.

The RNA-Seq datasets used to benchmark the Trans-Learn software, including (a) TuMV-A.halleri study system plant pathogen dataset, (b) COVID-19 diagnostic gene expression dataset, (c) TGCA cancer subtype classification gene expression dataset, (d) Sixteen wheat cultivars and six tissue types gene expression dataset

3.3.2 Method of Validation

In order to effectively evaluate the performance of the supervised machine learning models and prevent data leakage, a robust data splitting and cross-validation strategy was employed. Each labelled gene expression dataset was initially partitioned into 80% for training and 20% as a hold-out testing set (Figure 11). This allocation allowed for the training and validation of the models on a substantial portion of the data, whilst preserving an independent test set for evaluating their generalisability and performance on unseen data towards the end of the study.

The decision was made to use repeated stratified 8-fold cross-validation for the model selection and hyperparameter tuning process. This advanced cross-validation method ensures that the class frequency distribution is representative across all the splits, maintaining the original proportion of each class in both the training and validation sets. By using stratified k-fold cross-validation, the risk of biased performance estimates due to imbalanced class distribution was mitigated, which can arise when the dataset is split into random subsets without considering the class labels.

The repeated stratified 8-fold cross-validation process involves dividing the training set into 8 equally sized folds, with one fold serving as the validation set whilst the model is trained on the remaining 7 folds. This process is repeated 8 times, with each fold acting as the validation set once. To further enhance the reliability of the performance estimates, this entire 8-fold cross-validation procedure is performed multiple times, each with a different random seed. The average performance across all repetitions and validation sets is then used to estimate the model's performance.

By employing repeated stratified 8-fold cross-validation, a reasonable trade-off was achieved between the variance and bias of the performance estimates, as well as the computational time required for the cross-validation process. Critically, the risk of overfitting is reduced as cross-validation provides a more reliable estimate of the model's performance on unseen data, decreasing the risk of overfitting by ensuring that the model generalises well. Also, better hyperparameter tuning and model

selection are achieved. Repeated cross-validation helps to fine-tune model hyperparameters and select the best model by evaluating their performance across multiple repetitions and folds, leading to more stable and reliable results. Robustness to imbalanced class distribution is increased as stratified k-fold cross-validation maintains the class distribution across training and validation sets, reducing the risk of biased performance estimates caused by imbalanced classes.

When working with smaller datasets, the risk of overfitting increases due to the limited number of samples available for training and validation. By using repeated stratified 8-fold cross-validation, more unique training and validation set combinations are generated from the limited data, providing a more comprehensive evaluation of the model's performance. This approach helps to identify the best model and hyperparameters that generalise well, rather than overfitting to a specific subset of the data.

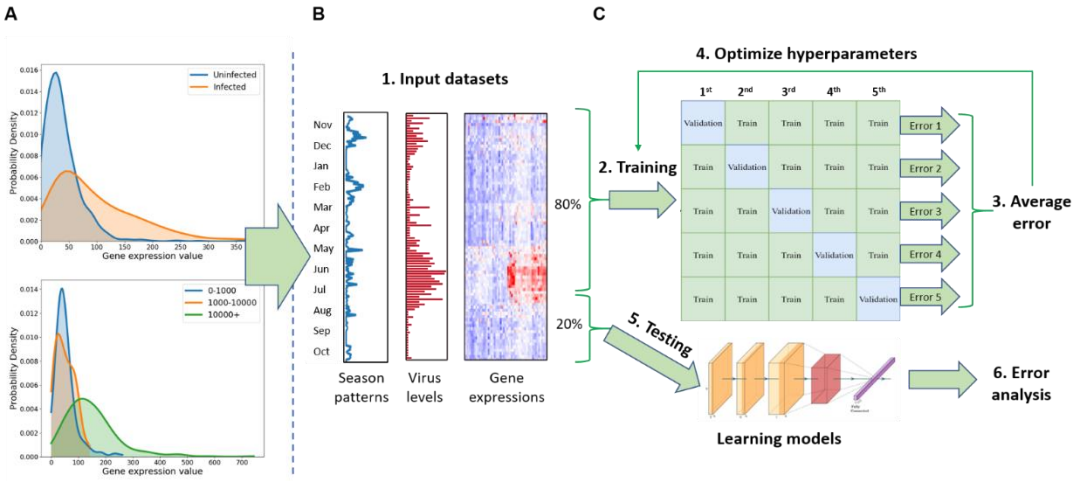

Figure 11.

Feature selection and cross-validation strategy

The training strategy designed for the machine learning models embedded in the Trans-Learn platform. (A) The probability density mapping between gene expression values and virus infection status and severity. (B) Gene expressions, virus levels and seasonal patterns combined as input datasets. (C) A legacy 5-fold cross-validation scheme used to train and optimise the learning models in the Trans-Learn platform.

3.3.3 Gene Expression Pre-processing and Target Encoding

In the pre-processing stage for the gene expression matrices, several methods were utilised to ensure the data was appropriately transformed and scaled for optimal model performance. These methods included the Box-Cox transformation, standard scaling, and log scaling, which were applied to different datasets depending on the specific requirements. In all cases, normalisation was treated as a hyperparameter so that the optimal normalisation method could be applied to the different expression datasets. This was done as the format of the expression datasets was different, with the A. helleri gene expression being normalised using RPM, the cancer subtype dataset being normalised using TPM, and the COVID-19 dataset being read counts.

The Box-Cox transformation is a power transformation technique that can only be applied to strictly positive data, such as gene expression data. It aims to adjust the data distribution to be more Gaussian-like, which can improve the performance of certain machine learning algorithms that assume normality. The transformation is particularly useful for stabilising the variance and making the data more symmetric. The transformation is defined by the following:

$$y(\lambda) = \begin{cases} \frac{y^{\lambda} - 1}{\lambda}, if\ \lambda \neq 0; \\ \log(y), if\ \lambda = 0. \end{cases}$$

where y(i) is the original value of the feature, y'(i) is the transformed value, and λ is the transformation parameter that determines the power to which the data is raised. The optimal value of λ can be determined by an optimisation process such as maximum likelihood estimation (MLE), which seeks to find the value of λ that maximises the log-likelihood function of the transformed data.

The standard scaling method, also known as z-score normalisation, is a pre-processing technique that transforms the data such that each feature has a mean of zero and a standard deviation of one. This scaling method ensures that all features contribute equally to the model, preventing any one feature from dominating the model due to differences in scale. Standard scaling can be particularly useful for algorithms that are sensitive to the scale of input features, such as distance-based

methods like k-Nearest Neighbours and gradient descent-based optimisation algorithms.

The equation for standard scaling is as follows:

$$z = \frac{x - \mu}{\sigma}$$

where y(i) is the original value of the feature, y'(i) is the standardized value, μ is the mean of the feature, and σ is the standard deviation of the feature.

The log scaling method is a pre-processing technique that can be particularly useful for reducing the impact of extreme values and compressing the data's range. This transformation is commonly applied when the data exhibits a skewed or heavy-tailed distribution, as it can help to make the data more symmetric and Gaussian-like.

The equation for log scaling is as follows:

$$y' = \log_2(y + 1)$$

where y is the original value of the feature, y' is the log-transformed value. 1 is added to the original value to handle zero values and ensure the logarithm is well-defined.

Log scaling is frequently used in gene expression data scaling due to the nature of the data. Gene expression values can span several orders of magnitude, and the distribution of these values is often positively skewed, with a long tail of highly expressed genes. By applying a logarithmic transformation, the data is compressed, reducing the impact of extreme values and improving the model's ability to capture patterns in the data. Additionally, log-transformed gene expression values are more interpretable, as fold changes in gene expression are directly related to the differences in log-transformed values.

In the case of the seasonal Arabidopsis halleri dataset samples, the expression of samples was normalised with respect to uninfected samples collected in the same

month of the year. This approach was adopted to minimise the effect of seasonality on gene expression that might otherwise negatively impact the model's performance. By accounting for the seasonal variation, the model can better discern the differences in gene expression patterns attributable to the condition of interest, rather than being confounded by seasonal fluctuations.

The scaling method can be defined as:

$$y_i' = \frac{\left(y_i - \overline{y_{,ui}}\right)}{\sigma_{u,i}}$$

Where is the month, and u denotes infected samples.

As well as pre-processing the expression data, it was essential for me to encode the target variable as in the COVID-19, cancer subtype, and wheat tissue type datasets, this variable was categorical as they represent distinct classes instead of continuous numerical values. To enable machine learning algorithms to work effectively with these targets, it is necessary to encode them using a suitable categorical encoding method. I used integer encoding as well as one-hot encoding to transform the target variables in these three datasets, with the encoding method changing depending on the supervised ML algorithm as different encoding methods have been shown to be more optimal for specific types of models, such as one hot encoding for neural networks.

One commonly used method for encoding categorical target variables is integer encoding or label encoding. In this method, each distinct category is assigned an integer value. For instance, if there are three COVID-19 status classes (e.g., negative, positive, and recovered), they could be encoded as 0, 1, and 2, respectively. Similarly, for cancer subtypes, each subtype would be assigned a unique integer value.

Another popular encoding method for categorical targets in multi-class classification problems is one-hot encoding (also known as dummy encoding). One-hot encoding transforms the target variable into a binary vector, where each element in the vector corresponds to a specific category. The vector has a length equal to the number of

distinct categories, and for each observation, the element corresponding to the category is set to 1, while all other elements are set to 0. To give an example of these encoding types, for cancer subtypes, each subtype would be assigned a unique integer value or one-hot encoded value as can be seen in Table 2.

Table 2. A table containing the integer encoded and one hot encoded values of the 10 cancer subtypes in the TGCA dataset.

Cancer Subtype	**Integer Encoding**	**One-Hot Encoded Value**
Breast Invasive Carcinoma (BRCA)	0	[1, 0, 0, 0, 0, 0, 0, 0, 0, 0]
Colon Adenocarcinoma (COAD)	1	[0, 1, 0, 0, 0, 0, 0, 0, 0, 0]
Head and Neck Squamous Cell Carcinoma (HNSC)	2	[0, 0, 1, 0, 0, 0, 0, 0, 0, 0]
Kidney Renal Clear Cell Carcinoma (KIRC)	3	[0, 0, 0, 1, 0, 0, 0, 0, 0, 0]
Brain Lower Grade Glioma (LGG)	4	[0, 0, 0, 0, 1, 0, 0, 0, 0, 0]
Lung Adenocarcinoma (LUAD)	5	[0, 0, 0, 0, 0, 1, 0, 0, 0, 0]
Lung Squamous Cell Carcinoma (LUSC)	6	[0, 0, 0, 0, 0, 0, 1, 0, 0, 0]
Prostate Adenocarcinoma (PRAD)	7	[0, 0, 0, 0, 0, 0, 0, 1, 0, 0]
Thyroid Carcinoma (THCA)	8	[0, 0, 0, 0, 0, 0, 0, 0, 1, 0]
Uterine Corpus Endometrial Carcinoma (UCEC)	9	[0, 0, 0, 0, 0, 0, 0, 0, 0, 1]

Despite the target variable for the TuMV-*A. halleri* dataset being reads per million to quantify TuMV, I decided to classify each sample as being uninfected or infected so

that all models would be for classification, and due to the extreme variance in the quantity of TuMV making it a challenging variable to predict through regression.

3.3.4 Feature Selection

Feature selection techniques are commonly used in supervised learning to pre-process raw input data. Before selecting suitable learning models, it is important to conduct feature selection such as filter, wrapper and embedded methods[338]. Filter methods assess features independently of a classifier by using statistics such as mutual information [383], which selects features based on the strength of their dependencies on the target. Wrapper methods [339] search the feature space for subsets and select features that can minimise the loss function in a cross-validation environment.

In my case, a combined filter and wrapper method was used to reduce the number of features (e.g. genes) in order to improve the interpretability and accuracy of the classifiers. By removing irrelevant and redundant genes in the input data through feature selection, the features used in the learning models should not only be better predictors, but also relevant to the actual biological process advised,

First, I focused on filtering out genes that exhibited low variance or were not expressed in a significant proportion of samples. This filtering step aimed to reduce noise in the data, decrease computational complexity of calculations later in the pipeline, and mitigate the risk of overfitting by eliminating irrelevant or uninformative features. To identify low-variance genes, I calculated the variance of each gene's expression values across all samples prior to normalisation. Then, I established a threshold by selecting the 20th percentile of variance values among all genes and removed those genes falling below this threshold. This filtering step should discard the most uninformative genes, as those with higher variance are more likely to be associated with significant biological differences between samples.

Additionally, I assessed the proportion of samples in which each gene exhibited no expression, i.e., an expression level of 0. This step aimed to identify genes with

sparse expression patterns, as these genes may be less relevant for predicting the target variable and could introduce noise to the model. By calculating the proportion of samples with 0 expression for each gene, I set a threshold of 15% and removed genes that exceeded this value. This approach ensured that the remaining genes were consistently expressed across a majority of the samples, increasing the likelihood that they contribute valuable information to the machine learning models and that they would be reliable biomarkers that generalise well to test data. It is important to note that these filtering steps were applied using the 80% training split of the dataset to prevent data leakage, and that the thresholds should be considered as a hyperparameter for the user to control.

In addition to filtering out genes with sparse expression patterns, I utilised additional filter feature selection methods to further refine the gene set and identify those most relevant for predicting the target variable. This was accomplished by using both mutual information and the one-way ANOVA F-value statistics to select two sets of highly scoring genes based on these univariate scores.

Mutual information is a measure that quantifies the degree of dependence between two variables, in this case, the expression level of a given gene and the target variable, such as disease status or cancer subtype. Higher mutual information values indicate a stronger relationship between the gene and the target variable, suggesting that the gene may be more informative for prediction. Given that the gene expression is considered a continuous distribution, whereas the target variables in my datasets are discrete random variables, the criterion for the mutual information method can be represented by the following equation:

$$I(G_i;T) = \sum_{t\epsilon T} \int_{G_i} p_{(G_i,T)}(g_i,t) \log\left(\frac{p_{(G_i,T)}(g_i,t)}{p_{G_i}(g_i)p_T(t)}\right) dg_i$$

where G_i denotes the i^{th} gene, T denotes the target, g_i denotes the expression level of the i^{th} gene, t denotes the target value, $p_{(G_i,T)}$ represents the joint probability density function of the i^{th} gene and the target variable, while p_{G_i} and p_T denote the marginal probability density functions of the i^{th} gene and the target variable, respectively.

The **f_classif** method utilises the one-way ANOVA F-value to assess the relationship between each gene and the target variable, ranking genes based on the strength of this relationship. The criterion for the f_classif method can be expressed as:

$$F(G_i) = \frac{SS_b \cdot df_w}{SS_w \cdot df_b}$$

Where:
$F(G_i)$ is the F-value for the ith gene
SS_b is the sum of squares between groups
df_b is the degrees of freedom between groups (number of groups - 1)
SS_w is the sum of squares within groups
df_w is the degrees of freedom within groups (total number of samples - number of groups)

The one-way ANOVA F-value measures the ratio of between-group variability to within-group variability. In the context of gene expression data, it identifies genes with different mean expression levels across target classes, which can be indicative of these genes being strong predictors for classification tasks and being associated with biological responses.

My approach to gene selection aims to capture both linear and non-linear relationships between gene expression levels and the target variable by employing two different feature selection metrics: F regression and mutual information. These two methods complement each other, as they emphasise different aspects of the relationship between gene expression and the target. To expand on this, by quantifying the amount of information shared between a gene's expression level and the target variable, mutual information could identify genes with complex, non-linear relationships to the target that could have been overlooked by the F-value metric. Therefore, by applying both mutual information and F-value filter methods, I could select two sets of highly informative genes that exhibit strong relationships with the target variable. I then combine these sets into a final set of genes that is taken forward for analysis. This union of genes possess a diverse range of relationships with the target variable, capturing both linear and non-linear dependencies. Consequently, the combined gene set provides a rich source of information for the

machine learning models to utilise, increasing the likelihood of identifying robust and reliable biomarkers that generalise well to test data.

After identifying highly predictive genes from the union of the two sets, I employed a Sequential Forward Floating Selection (SFFS) wrapper method to further refine the subset of genes. SFFS is a classifier-dependent, iterative algorithm that considers the interactions between features and the specific learning algorithm being used. This method is highly effective in selecting the most informative and relevant features for a given model, making it a suitable choice for refining the gene set after initial filtering.

SFFS begins by selecting the feature that contributes most significantly to reducing the validation loss. It then iteratively adds features to the subset, evaluating the impact of each addition on the model's performance. During this process, SFFS also allows for the removal of previously selected features if their inclusion is found to negatively impact the model's performance, thus ensuring an optimal and compact feature set. This flexibility enables SFFS to handle complex gene dependencies and redundant genes effectively.

Depending on the machine learning or deep learning models used, SFFS continues to add features until the optimal number of genes is reached. The optimal gene subset was determined by monitoring the performance of the model on a validation set, with the goal of maximising the model's accuracy while minimising the number of features used. Once the optimal subset of genes were found, they were provided as input data for the learning models integrated into the Trans-Learn platform.

An additional benefit of the SFFS approach is that it can be adapted to refine the number of features down to a specified number, depending on the desired application. This capability has practical implications, such as identifying biomarkers for use in techniques such as RT-qPCR. By refining the feature set to a specific number using SFFS, it is possible to identify a small panel of highly informative biomarkers suitable for RT-qPCR validation. This panel would allow for a focused investigation of gene expression patterns in the context of the studied

condition, potentially leading to the discovery of reliable biomarkers for diagnostic or prognostic purposes.

3.3.5 Supervised Learning Algorithms

A range of supervised ML techniques have been embedded in the Trans-Learn platform to perform the two-stage virus prediction. To identify virus presence and classify virus abundance levels, five ML techniques have been included in the pipeline as classifiers: K-nearest neighbours, LightGBM, Logistic Regression, ANN, and SVM. Here, I introduce these supervised machine learning models and justify why I chose to include them in the Trans-Learn software.

The K-nearest neighbours (K-NN) algorithm is an intuitive, supervised machine learning technique that operates by mapping each sample's features onto a p-dimensional feature space, where p corresponds to the number of features. In this space, the algorithm analyses the proximity of samples to one another, identifying their relationships based on the distances between them. To classify a new sample, K-NN determines its k-nearest neighbours within the feature space and assigns it to the majority class among these neighbours, considering their respective distances. The majority voting approach typically used in K-NN models makes it robust to outliers. As a non-parametric method, it makes no assumptions about the underlying distribution of the data, which is advantageous for complex gene expression patterns. The most important hyperparameter for this model is k, the number of nearest neighbours to consider when making a prediction so I will tune this using my cross-validation strategy.

However, K-NN has some weaknesses when applied to high-dimensional gene expression data. It has a high time complexity for large datasets and is sensitive to feature scaling, however, the scaling methods I have applied should tackle this. Moreover, K-NN can suffer from the "curse of dimensionality," leading to issues with model performance when too many features are included. In terms of interpretability, K-NN is less transparent than linear models but more interpretable than complex models like artificial neural networks. Feature importance can be

assessed through techniques like feature ablation. Compared to other algorithms, K-NN is generally faster to train but slower in making predictions. Its simplicity can be advantageous, but it may not perform as well as other algorithms for high-dimensional data or complex relationships between features and the target variable.

Light Gradient Boosting Machine (LGBM) trains an ensemble of decision trees in sequence; through learning from the negative gradients of the decision trees, LGBM uses gradient descent to minimise residual errors to find the optimal way to split the input features in each tree [384] , offering several advantages when dealing with high-dimensional gene expression data. One of its strengths lies in its ability to capture complex, non-linear patterns and relationships within the data, which may be challenging for linear models like logistic regression. Compared to other supervised machine learning methods in the domain of tabular data modelling, LightGBM and other gradient boosting machines excel, frequently winning machine learning competitions. As well as its strong predictive capabilities, LGBM is relatively fast to train, as it utilises a histogram-based algorithm that allows for faster tree-growing, making it more efficient compared to traditional gradient boosting methods. This training efficiency is compounded by the possibility of training on a GPU rather than CPU for faster matrix calculations.

However, LGBM does have some weaknesses. It can be prone to overfitting, especially when using a large number of boosting rounds or a deep tree structure. Regularisation techniques and careful hyperparameter tuning can help mitigate this issue, which I plan to implement when hyperparameter tuning. The interpretability of LGBM models is better than deep learning models, as it is possible to identify the key features that strongly influence the ensemble of decision trees. However, the interpretability of LGBM models is worse than simpler models like logistic regression.

With respect to the other listed algorithms, LGBM is a particularly well-suited method for tabular gene expression datasets as it offers the ability to capture complex relationships between genes, efficient model training, and strong predictive performance. However, the potential for overfitting necessitates careful model selection and hyperparameter tuning.

Logistic Regression (LR) is a linear classifier that models the probability of a sample belonging to a particular class based on its features. Its primary strengths lie in its simplicity, faster training times, and straightforward interpretation compared to more complex models like LGBM or Artificial Neural Networks. This interpretability is particularly valuable for extracting meaningful insights and performing statistical significance tests on the model, which can be important for testing biological hypotheses.

The logistic regression model uses the logistic function to model the probability of a sample belonging to a particular class. The logistic function is defined as:

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

The logistic regression model can be expressed as:

$$P(Y = 1|X) = \sigma(w^T \cdot X + b)$$

Here,
$P(Y = 1|X)$ is the probability of the sample belonging to class 1 given its features X.
$\sigma(x)$ is the logistic function, which maps any real-valued number to the range (0, 1).
w is a vector of weights for each feature.
X is a vector of the feature values for a sample.
$w^T \cdot X$ is the dot product of the weight vector w and the feature vector X.
b is the bias term, which shifts the decision boundary.

To fit the model, weights and bias are estimated by minimising the negative log-likelihood (cross-entropy loss) using optimisation techniques like gradient descent. The implementation I used applies the L-BFGS optimisation method, which approximates the Hessian for faster convergence.

In high-dimensional gene expression data, LR performs well when relationships between features and the target variable are predominantly linear. However, gene

expression datasets often contain a mix of linear and non-linear relationships. LR's inability to capture non-linear relationships may lead to suboptimal performance compared to algorithms like LGBM, Artificial Neural Networks, and Support Vector Machines with kernel functions, which can model non-linearities more effectively.

To control overfitting, LR utilises regularisation terms, such as L1 or L2 regularisation, along with a tuneable regularisation strength parameter which I intend to adjust during hyperparameter tuning. Both L1 and L2 regularisation help prevent the model from becoming overly complex by promoting weight sparsity in the case of L1 and penalising large coefficients in the case of L2, promoting simpler models with more generalisable performance.

LR presents a simple, interpretable, and computationally efficient option for classifying high-dimensional gene expression data. While it may be outperformed by non-linear algorithms when complex relationships exist within the data, its ease of interpretation and fast training time make it a valuable option for certain applications, especially in transcriptomic analysis as interpretability is essential.

Artificial Neural Networks (ANNs) represent a sophisticated class of machine learning models composed of interconnected neurons organised into layers. These networks incorporate at least one hidden layer equipped with non-linear activation functions, enabling ANNs to discern non-linear relationships in data. This feature renders them particularly adept at handling high-dimensional gene expression data, which often exhibit intricate patterns.

The key strengths of ANNs include their capacity to learn and model complex patterns, capture non-linear relationships, and generalise effectively to new data. Additionally, ANNs offer a high degree of adaptability, providing the flexibility to fine-tune various aspects, such as the number of hidden layers, neurons per layer, activation functions, and learning rate, to optimise performance for specific tasks. This customisability allows for tailoring the model architecture to achieve optimal results, a marked advantage over simpler models like K-NN, which rely on a single primary hyperparameter.

However, ANNs also possess certain drawbacks. First, they can be computationally intensive to train, particularly for larger networks and datasets, resulting in longer training times compared to more streamlined models like Logistic Regression. Second, ANNs are prone to overfitting, particularly in cases with limited training data or highly complex network architectures. Employing regularisation techniques, such as dropout or weight decay, can help mitigate this issue. Finally, ANNs exhibit the lowest interpretability among all algorithms, as the weights and connections within the network can be difficult to decipher, particularly for deep architectures, making it challenging to extract meaningful insights from the model.

In summary, Artificial Neural Networks constitute an effective choice for classifying high-dimensional gene expression data due to their ability to model complex patterns and non-linear relationships. While they may require increased computational resources and training time compared to more straightforward models, their adaptability and generalisation capabilities render them a valuable option across various applications. Nonetheless, careful attention must be given to address potential overfitting and interpretability challenges.

Support Vector Machines (SVMs) are versatile machine learning models that map the samples' features into a feature space in which different targets are separable. Unlike K-NN that relies on proximity to neighbouring samples for classification, SVMs construct a decision boundary, or hyperplane, that maximises the margin between different classes in the feature space. By using the radial basis function (RBF) kernel technique, which I chose to apply, SVMs are capable of tackling more intricate classification tasks, making them particularly effective for high-dimensional gene expression data. Their capacity to handle both linear and non-linear relationships, as well as their robustness to overfitting, contribute to their effectiveness in this context. Key hyperparameters for SVMs include the regularisation parameter (C) and kernel-specific parameters, such as the gamma parameter in the RBF kernel.

Strengths of SVMs include their ability to model complex, non-linear relationships using kernel functions, which can be especially valuable for gene expression data, where such relationships are common. Additionally, SVMs can show robustness to

overfitting due to the regularisation parameter, which helps control the model's complexity and margin size.

However, SVMs have certain weaknesses. One drawback is that they can be computationally intensive, particularly when dealing with large datasets, making them slower to train compared to simpler models like Logistic Regression. Moreover, the choice of kernel function and tuning of hyperparameters can significantly impact model performance, necessitating thorough exploration of the hyperparameter space to achieve optimal results. Additionally, while SVMs are more interpretable than deep learning models, their interpretability is not as straightforward as that of linear models such as LR or even tree models such as LGBM.

In summary, Support Vector Machines offer a powerful and flexible option for classifying high-dimensional gene expression data, adept at handling complex linear and non-linear relationships. Their robustness to overfitting and predictive performance makes them an attractive choice for various applications, however, the high computational demands and lack of interpretability need to be considered.

I have included a diverse set of supervised machine learning models, each possessing their own strengths and weaknesses, making them suitable for different scenarios depending on the user's needs and the characteristics of the data. I plan to explore and tune each of these models, adjusting their key hyperparameters to optimise their performance on the gene expression datasets. By benchmarking their performance on the four datasets, I aim to provide a comprehensive comparison that can guide users in selecting the appropriate model for their specific needs and objectives, ensuring the best possible results.

3.3.5.1 Model Ensemble Methods

In addition to using individual models, I decided to employ ensemble methods, specifically stacking[111] and blending[385], to potentially improve the performance of

the predictions at the expense of interpretability. Ensemble methods combine the predictions of multiple models to create a more accurate and robust final prediction.

Stacking is an ensemble technique that involves training multiple base models on the dataset and then using a second-level model, known as the meta-model or meta-learner, to make predictions based on the predictions of the base models. The base models are recommended to be diverse, consisting of different supervised ML algorithms or through varied hyperparameter settings. The meta-model is trained to make a final prediction using the predictions of the base models as features. Stacking can improve predictive performance by exploiting the strengths of the diversity in multiple models while reducing the impact of their individual weaknesses. Stacking can be computationally intensive especially when tuning hyperparameters or employing wrapper-based feature selection methods, as it requires training multiple models. Interpretability is sacrificed due to the complexity of the final model, which is harder to interpret than individual models since it combines the predictions of multiple models, making it challenging to understand the underlying relationships between the features and target variable.

In stacking, it is easy for data leakage to accidentally occur by information from the training sets being inadvertently used in the validation sets. This can happen if the base models and meta-model are trained on the same part of a dataset as even if cross-validation was performed on the meta-features that are created by the base models, the meta-model could learn to exploit noise in the meta-features and result in overfitting. Therefore, it is critical for the base models to be trained on the training set or folds, and the meta-model should be trained on the predictions generated by the base models on the validation set or folds. The performance of the entire stacked model should be evaluated on the testing set.

For my implementation of stacking, after splitting the training data using repeated stratified k-fold cross-validation, I trained each of the five base models on the training data and generated out-of-fold predictions for the validation set. These out-of-fold validation predictions were saved as they represent the features that are used to train the meta-model, logistic regression in the stacked ensemble I have designed (Figure 12). Once all out-of-fold predictions are generated, I would use them as input

features to train the logistic regression meta-model. Then I would train each of the five base models on the entire training set, then use the trained base models and the meta-model to generate predictions for the test set, and evaluate the stack's performance using evaluation metrics.

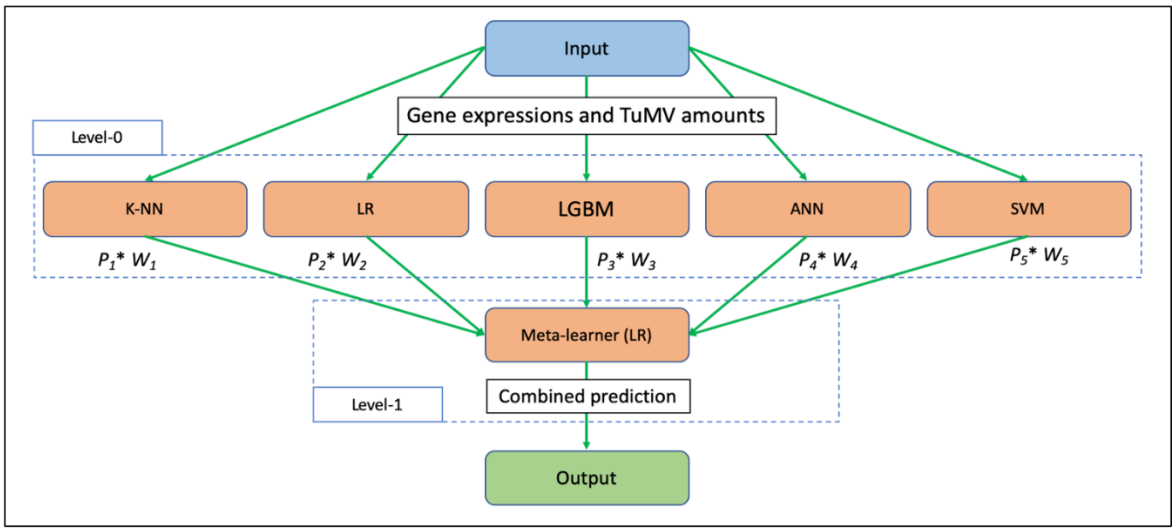

Figure 12.

Stacked ensemble model.

The architecture of the stacking model used to combine a range of machine learning models to increase the predictive power of Trans-Learn. The input data at the top is gene expression data, which is used to produce predictions from the five level-0 models. The predictions of the level-0 models are the input features for the meta-learner (level-0 model), and a final prediction is generated by the meta-learner.

Blending is similar to stacking but differs in the way the meta-model is trained. In blending, the dataset is split into two parts: one is used to train the base models, and the other is used to make predictions with the base models. These predictions are then used as features to train the meta-model. This approach can be simpler and faster than stacking, as it avoids the need for cross-validation to generate out-of-fold predictions. However, blending can be less robust than stacking, as it relies on a single holdout set to train the meta-model, which might not fully represent the diversity of the data.

A weighted average of the individual model predictions can be used as a blend, with weights assigned to each. For example, if LR, SVM, LGBM, K-NN, and ANN

models produce predictions of 0.70, 0.60, 0.85, 0.70, and 0.75 for a given sample, and weights of 0.2, 0.3, 0.1, 0.2, and 0.2 were assigned to each model respectively, the weighted average would be calculated to produce the final prediction of 0.67. The blending process involves training the base models on one part of the dataset and using the other part to generate predictions. These predictions are then combined using the assigned weights to produce the final prediction, which is evaluated on a separate test set.

The weighted average blend approach can be simpler and faster than stacking as it avoids the need for cross-validation to generate out-of-fold predictions. The weights assigned to the models can also be tuned using the validation set, allowing for greater control over the blend's performance. However, blending relies on a single holdout set to train the meta-model, which might not fully represent the diversity of the data, leading to overfitting or underfitting. Therefore, I intend to evaluate the blend's performance on a separate test set to ensure its generalisation to the hold-out test datasets.

For blending, I intend to weight different base models by their respective repeated stratified k-fold cross validation loss. To calculate the weights for each model based on their validation loss, I will use the following formula:

$$w_i = \frac{e^{-l_i}}{\sum_{i \in I} e^{-l_i}}$$

w_i is the normalized weight for the i-th model
l_i is the validation loss for the i-th model
I is the set of all models

Using this formula, the models that have a lower validation loss will receive a larger weight in the blend and the weights are normalised so that their sum is equal to one. This enables the weights to be multiplied by the base models' respective predictions to compute the weighted average which represents the final predictions.

Both stacking and blending have their advantages and disadvantages. Stacking has been shown to demonstrate better performance in supervised ML competitions as it can better exploit the strengths of multiple models at the cost of being more computationally intensive and less interpretable. By experimenting with both stacking and blending, I aim to identify the ensemble technique that provides the best performance in the context of my studies to predict targets using gene expression datasets.

3.3.6 Feature Representation

A key aspect of this project was to investigate and evaluate the application of image-based models to gene expression data and in order to accomplish this, it was necessary to encode the data into a suitable image format. This process involves arranging the genes and samples into a grid (2D matrix), where the rows represent the genes and the columns represent the samples. By transforming the gene expression data into an image-like structure, it is possible to take advantage of the spatial features that CNNs and ViTs are designed to detect. Specifically, the predictive power of a CNN model relies not only on the individual genes within the input image, but also on their relationships to neighbouring genes in the image as localised features in the input image can contain complex non-linear relationships between highly expressed genes, co-expressed genes, and virus status. For instance, if two genes have a strong dependency with the virus status, the CNN model can only link the two genes when they are close to each other in the 2D gene expression matrix.

This encoding method employs the strengths of CNNs and ViTs, which are well-suited for identifying patterns and relationships within image-like data. By transforming gene expression data into an image format, I can leverage the power of these models to identify features and relationships that might be missed by other machine learning algorithms.

3.3.7 CNN Architecture

A key objective in this study was to investigate and apply image analysis methods to gene expression datasets, so I chose to apply a CNN as a supervised ML model. During the development of this research, I observed that the relatively small input image size (as small as 12x12) rendered the use of existing deep Convolutional Neural Network (CNN) architectures impractical. Consequently, a customised CNN architecture was designed for the classification tasks. The choice of this tailored architecture was driven by the need to optimally process gene expression images, which were typically represented as 32x32 2D matrices.

The learning architecture (illustrated in Figure 13) accepts gene expression images as input. To efficiently process this data, the model comprises three initial convolutional layers, which contain 64, 128, and 256 filters, respectively. A series of experiments were conducted to determine the optimal kernel size and padding strategy, resulting in the selection of 3x3 kernels and a no-padding approach. This configuration causes the input matrix to shrink after passing through the three layers, effectively condensing the gene information.

Following the three convolutional layers, a max-pooling layer with a 4x4 input window is introduced to maintain spatial relationships while further reducing the size of the window. Subsequently, the output of the max-pooling layer is flattened into a vector of 4,096 elements, which is then processed by a 512-neuron dense layer. This is followed by either a *p*-neuron final layer of softmax to predict the *p* classes in the one-hot encoded target matrix.

The output of the max-pooling layer is then flattened into a vector of 4,096 elements and processed by a 512-neuron dense layer. This design choice facilitates the combination and abstraction of the extracted features to generate high-level representations. The final layers consist of either a 2-neuron softmax layer to predict virus presence or a 3-neuron softmax layer to classify virus abundance levels, allowing the model to make predictions based on the learned features.

All three convolutional layers employ the Rectified Linear Unit (ReLU) activation function, chosen for its ability to accelerate convergence during training and address the vanishing gradient problem. L2 regularisation is utilised to enhance the model's generalisation capabilities, mitigating overfitting risks. The Adam optimisation algorithm is used for model training, with cross-entropy serving as the loss function, as they are well-suited for classification tasks. Additionally, the EarlyStopping callback function is applied to prevent overfitting during CNN model training by estimating the optimal number of epochs.

In conclusion, the choice of a tailored CNN architecture for this research is well-founded, given the need to process the relatively small gene expression images and the unique characteristics of the dataset (Figure 13). The customised design, featuring three initial convolutional layers with increasing filter counts and the selection of 3x3 kernels with a no-padding strategy, ensures optimal feature extraction and efficient processing. This approach ultimately enhances the model's predictive performance and facilitates a deeper understanding of the relationships between genes and virus presence and abundance levels.

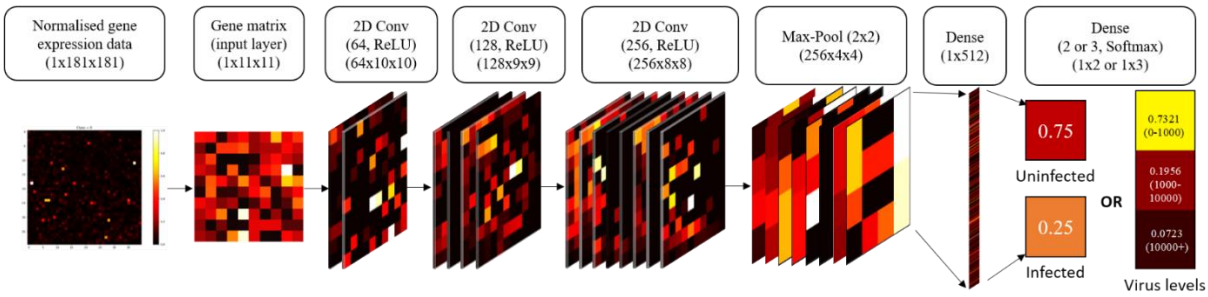

Figure 13.

CNN model architecture

The architecture of the CNN model trained for predicting virus presence and virus abundance levels, read from left to right.

3.3.7.1 Spatial Joint Mutual Information Spiral Arrangement

During the development of this research, there were no existing algorithms or approaches specifically designed to organise genes in a manner optimised for CNN models. To address this issue, I developed a novel method known as the spiral gene arrangement. The primary objective of this method was to arrange genes in an order that maximises the joint mutual information between neighbouring gene expression patterns and the given target, such as disease status.

The spiral method of arranging genes is defined as follows: to begin, an empty image matrix was created to serve as the foundation for the spiral gene arrangement. In the initial step, the central element of the empty matrix was populated by the gene that demonstrated the greatest mutual information with the target. Subsequently, the remaining elements of the matrix were filled in a clockwise spiral manner, moving outwards from the centre with genes that maximised the joint mutual information metric with previously selected genes within 3x3 neighbourhood or an *N*x*N* neighbourhood, where *N* represents the size of the kernel utilised in the CNN architecture. This spiral gene arrangement process is continued until all elements in the matrix are filled (Figure 14). Consequently, genes that maximise the joint mutual information between surrounding genes and the target are situated in a neighbourhood central to the new gene matrix. Meanwhile, genes positioned closer to the border maintain a weaker relationship but are in theory still placed near genes with higher joint mutual information compared to a random arrangement.

Image of genes

Order of gene arrangement

81	50	51	52	53	54	55	56	57
80	49	26	27	28	29	30	31	58
79	48	25	10	11	12	13	32	59
78	47	24	9	2	3	14	33	60
77	46	23	8	1	4	15	34	61
76	45	22	7	6	5	16	35	62
75	44	21	20	19	18	17	36	63
74	43	42	41	40	39	38	37	64
73	72	71	70	69	68	67	66	65

Figure 14.

(Left) An example gene expression input image (Right) The spiral method of arranging genes in order to maximise the joint mutual information at the centre of an input image. The first gene to be set is the centre gene, and genes are sequentially added spirally outwards clockwise from the centre.

The spiral gene arrangement method could offer significant potential for improving the predictive performance of a CNN model when applied to gene expression data. To appreciate the benefits of this approach, it is essential to understand the fundamental workings of convolutional layers within a CNN architecture. A key aspect of the success of CNNs lies in their convolutional layers, which employ local receptive fields and shared weights to capture spatial relationships and hierarchical features within the input data. By using small kernels, convolutional layers can identify local patterns and retain spatial information, which is subsequently combined and abstracted through deeper layers of the network to recognise higher-level features.

In the context of gene expression data, capturing and exploiting the intricate relationships between genes and their interactions with the target variable is crucial for achieving accurate predictions. Traditional approaches for organising gene expression data do not necessarily account for these complex relationships, which may hinder the CNN model's ability to discern relevant patterns and extract meaningful features. The spiral gene arrangement method addresses this challenge by organising genes in a manner that maximises the joint mutual information between neighbouring gene expression patterns and the target, such as disease status.

By arranging genes in a spiral configuration, the method effectively captures local dependencies and spatial correlations between genes, positioning those with the strongest mutual information with the target in the centre of the matrix. This arrangement ensures that genes with the most significant impact on the target are situated in close proximity, allowing the CNN model to better capture and exploit these relationships during the feature extraction process.

Furthermore, by arranging genes that maximise the joint mutual information within NxN neighbourhoods, the method aligns with the principles of convolutional layers, which utilise kernels of size NxN to extract local patterns. This compatibility enables the CNN model to more effectively identify and extract meaningful features from the gene expression data, potentially leading to enhanced predictive performance.

In summary, the spiral gene arrangement method offers a novel and intuitive approach for organising gene expression data in a manner that complements the inherent strengths of CNN architectures. By maximising the joint mutual information between neighbouring genes and the target, the method enhances the network's ability to capture and exploit the complex relationships that underpin the data. This compatibility, in turn, allows the CNN model to more effectively extract meaningful features and achieve improved predictive performance, ultimately contributing to a more nuanced understanding of gene interactions and their impact on disease status or other relevant targets.

3.3.8 Vision Transformer Architecture

During the development of this research, a key objective was to investigate and apply advanced image analysis methods to gene expression datasets. In pursuit of this goal, I chose to apply a Vision Transformer[184] (ViT) as a supervised machine learning model, as ViTs have been proven to excel in image recognition and classification tasks, particularly by exploiting spatial relationships, including long-distance ones, using patches.

A Vision Transformer (ViT) is a relatively recent development in the field of computer vision, which adapts the transformer architecture, originally designed for natural language processing, to handle image data. ViTs have shown great promise in image classification tasks, outperforming traditional CNNs in certain cases. A major advantage of ViTs is their ability to capture and exploit spatial relationships within an image, even those spanning long distances, by dividing the input image into non-overlapping patches and processing them as a sequence of tokens.

In this study, a customised ViT architecture was designed for the classification of gene expression data. The ViT begins by dividing the input gene expression image into non-overlapping patches, which are then linearly embedded and processed as a sequence of tokens. The positional encoding is added to each token to preserve the spatial information of the patches. The resulting sequence of embedded patches is then passed through the transformer layers, with each layer containing multiple heads, with both the number of transformer layers and heads being critical hyperparameters. These multi-head self-attention mechanisms enable the ViT to capture intricate relationships between patches, including long-range dependencies that are often challenging for traditional CNNs to detect.

The self-attention mechanism in Vision Transformers (ViTs) captures relationships between non-overlapping patches, including long-range dependencies, by adaptively determining the importance of each patch based on its relevance to the current patch being processed. This process computes attention weights for each token (patch) in the sequence and applies a weighted sum to generate the output. The multi-head self-attention allows ViTs to learn diverse features and relationships simultaneously, resulting in a comprehensive understanding of the input image.

After the patches have been processed by the transformer layers, the final feature representation is obtained by applying a layer normalisation followed by a fully connected output layer with softmax activation, which facilitates the classification of gene expression data into the relevant categories (Figure 15). The use of a ViT model enables the extraction of both local and global features from the gene expression data, providing a comprehensive understanding of the relationships between genes and their impact on various biological phenomena.

Applying a ViT to gene expression data offers several benefits, including improved predictive performance, discovery of diverse features and relationships, robustness to input size, and scalability. ViTs' ability to capture long-distance interactions between genes through its self-attention mechanism significantly enhances their predictive power, allowing the model to discover and exploit complex gene interactions, making them particularly well-suited for analysing gene expression data.

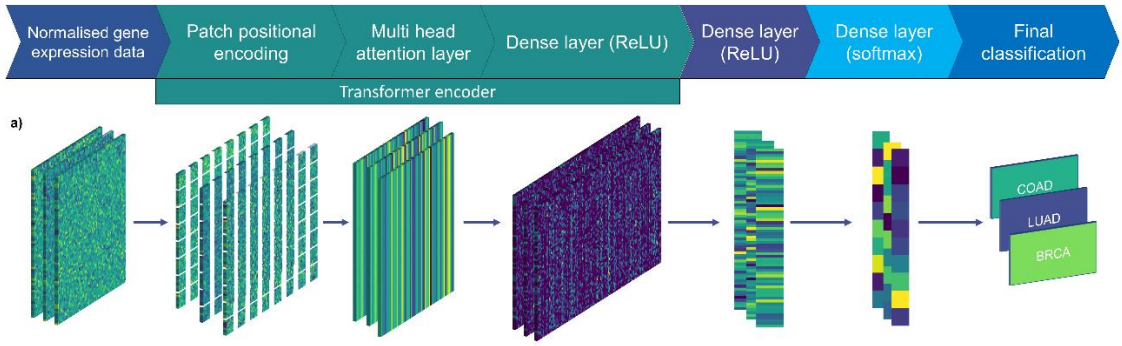

Figure 15.

The architecture of the ViT model trained for predicting cancer subtypes, read from left to right.

3.3.8.1 Spatial Joint Mutual Information Patch Arrangement

Vision transformers operate by dividing input images into non-overlapping patches of pixels, based on the assumption that each patch contributes unique information about the image content. In the context of gene expression data, if genes were randomly arranged within the encoded input image, it would be improbable for meaningful gene-gene interactions and coexpression patterns with predictive capabilities to be highly represented in the patches. Consequently, to optimise the performance of the ViT model and effectively exploit these relationships, an iterative algorithm was developed to arrange genes within the patches.

The objective of this algorithm is to maximise the joint mutual information of the genes within each patch, thereby capturing the essential gene-gene interactions and coexpression patterns that hold predictive power. By iteratively rearranging the

genes within the patches based on their relevance to the target variable, the algorithm generates an optimal arrangement that enhances the model's ability to discern spatial patterns and dependencies.

The algorithm operates in the following manner: it fills the patches sequentially from the top-left corner of the image to the bottom-right corner. The filling process commences with the top-left patch, where the initial gene assigned to this patch is the one whose expression level exhibits the highest mutual information with the target variable. Subsequently, the joint mutual information between the selected gene, each of the remaining genes, and the target variable is calculated. The gene that maximises the joint mutual information is then added to the patch. This iterative process continues until the patch is populated with the optimal arrangement of genes. Once the first patch is complete, the procedure is repeated for all remaining patches in the image. This sequential approach ensures that each patch captures the most informative gene interactions and coexpression patterns, contributing to an enhanced representation of the underlying biological phenomena.

In the case of multiclass classification, the algorithm allocates patches to each class, filling them with sets of multivariate biomarkers exhibiting high joint mutual information with the specific class. The process begins by selecting an appropriate number of patches and classes, ensuring that each class is allocated a proportionate number of patches. After completing the iterations for each class, the remaining patches are filled using a similar approach, but with a focus on the overall target variable instead of individual classes. This method allows for a comprehensive understanding of the relationships between genes and their impact on various biological phenomena, ultimately improving the predictive performance of the Vision Transformer model.

3.3.9 Hyperparameter Tuning

In the project, hyperparameter tuning was an essential step for improving the performance and generalisation of the machine learning models, including neural network models and standard supervised ML models. To achieve optimal

performance, distinct methods were utilised for tuning the hyperparameters of each model.

For the neural network models, Bayesian hyperparameter tuning was employed using the Hyperas library. Bayesian optimisation was chosen as it is an efficient and effective approach for optimising complex, high-dimensional functions, like those encountered in neural networks. It balances the exploration and exploitation trade-off, leading to more accurate and faster convergence compared to grid search or random search.

For all hyperparameter tuning, the function that was optimised was the categorical cross-entropy loss function. By minimising the categorical cross-entropy, the hyperparameters that maximise the probability of the prediction data belonging to the population distribution of the target given the gene expression can be identified. Categorical cross-entropy loss is a widely used loss function for multi-class classification problems, that measures the dissimilarity between the true class probabilities and the predicted class probabilities, and is defined by the following equation:

$$CCE = -\frac{1}{N}\sum_{i=0}^{N}\sum_{j=0}^{J} y_j \cdot \log(\hat{y}_j) + (1 - y_j) \cdot \log(1 - \hat{y}_j)$$

The parameters are defined as follows:

CCE: The categorical cross-entropy loss,

N: The total number of samples in the dataset.

i: The index for each individual sample, ranging from 0 to $N - 1$.

J: The total number of classes in the classification problem.

j: The index for each class, ranging from 0 to $J - 1$.

y_j: An indicator function for the true class, taking a value of 1 if the sample belongs to class j, and 0 otherwise.

$\hat{y}_j$: The predicted class probability for sample i belonging to class j, which is the output of the model for class j.

The equation computes the loss for each sample and class, and then averages the loss across all samples in the dataset to obtain the overall categorical cross-entropy loss value.

All hyperparameter optimisation was carried out using repeated stratified k-fold cross-validation to mitigate potential biases that could arise from imbalanced class distributions. Also, the repeated cross-validation enhances the reliability of performance estimates by averaging the results across multiple random data partitions.

For standard supervised ML models, the **hyperopt** library was employed to perform Bayesian hyperparameter tuning as this library facilitates the fine-tuning of model-specific hyperparameters. Here, I describe my strategy to tuning hyperparameters of each model.

The choice of k in the K-NN algorithm significantly affects its generalisation ability and overall performance. Optimising k aims to balance overfitting and underfitting. Overfitting arises when a small k value causes the model to capture noise rather than patterns, leading to poor generalisation. Conversely, underfitting occurs when a large k value makes the model insensitive to local patterns, resulting in suboptimal predictions based on broader neighbourhoods of instances.

The regularisation parameter, C, controls the strength of this penalty. A smaller value of C corresponds to a stronger regularisation, resulting in a higher constraint on the model coefficients and a simpler model, while a larger value of C leads to less constraint on the coefficients, allowing for a more complex model. The process of optimising C involves searching for the optimal value that balances the trade-off between model complexity and prediction accuracy. A model with a small value of C may be too simple and unable to capture complex patterns in the data, leading to underfitting. On the other hand, a model with a large value of C may overfit the training data by capturing noise and irrelevant details, resulting in poor generalisation to unseen data.

Optimising key hyperparameters in LightGBM, such as the number of estimators, tree depth, learning rate, and other parameters, is crucial for achieving the best model performance. The number of estimators, or boosting iterations, impacts performance

and overfitting risk. Techniques like early stopping or cross-validation help determine the optimal number. Tree depth affects the model's complexity, and finding the right depth balances underfitting and overfitting. Grid search, random search, or Bayesian optimisation can be used to explore different depths. The learning rate determines each tree's contribution to the final prediction. A smaller rate results in a more robust model, while a larger rate may cause overfitting. Searching over a range of possible values using grid search, random search, or Bayesian optimisation helps optimise the learning rate. Other hyperparameters, such as **min_data_in_leaf**, **feature_fraction**, and **bagging_fraction**, can be tuned to control complexity, improve generalisation, and reduce overfitting.

To achieve optimal SVM performance, it is essential to fine-tune key hyperparameters, such as the regularisation parameter (C) and kernel coefficient (gamma). C controls the trade-off between margin and classification error. A small C creates a larger margin but may underfit, while a large C enforces stricter separation, potentially overfitting. Optimising C involves using techniques like grid search, random search, or Bayesian optimisation and assessing the model's performance on a validation dataset. Gamma, specific to the Radial Basis Function (RBF) kernel, determines the decision boundary shape and the influence of individual training instances. A small gamma captures complex patterns, while a large gamma leads to a smoother boundary, potentially underfitting. Optimising gamma follows a similar process to C, exploring possible values and evaluating model performance on a validation dataset. The choice of the RBF kernel was determined by minimising the validation loss, ensuring the best possible model configuration.

The hyperparameter tuning process was executed after applying the SFFS wrapper algorithm. This order was chosen to guarantee the completion of feature selection prior to the model optimisation, generating a more stable and informative input space for the models. The SFFS algorithm not only reduces dimensionality but also eliminates irrelevant features. This enables a more computationally efficient hyperparameter optimisation process, which can concentrate on the most informative features and their relationships with the target variable. In this particular application, feature selection has a more significant impact on the loss than hyperparameter

tuning. Therefore, optimising a model for a set of features that, when refined, will substantially affect the error would be less effective, as it would necessitate re-optimisation.

In order to optimise the CNN and ViT models, Bayesian optimisation techniques were employed using the **hyperas** library, alongside other Python packages. The objective was to fine-tune key hyperparameters, including architecture, learning rate, regularisation, number of filter maps, attention heads, and neurons, to achieve optimal model performance. Bayesian optimisation is an effective and efficient approach to hyperparameter tuning, as it leverages a probabilistic model to explore the hyperparameter space intelligently, reducing the number of evaluations needed to identify the optimal configuration. The **hyperas** library, which is built upon the Keras and **hyperopt** libraries, simplifies the implementation of Bayesian optimisation for deep learning models.

For the CNN model, the following hyperparameters were optimised:

1. Convolutional layers: The number of convolutional layers was varied to determine the optimal depth of the CNN architecture. A deeper architecture can capture more complex spatial features and hierarchical representations, whereas a shallower architecture may be more computationally efficient but may not capture high-level abstractions as effectively.

2. Dense layers: The number of dense (fully connected) layers was optimised to strike a balance between model capacity and overfitting. More dense layers can improve the model's ability to learn complex relationships among features, while fewer layers may prevent overfitting by reducing model complexity.

3. Batch normalisation layers: Batch normalisation layers were incorporated into the architecture to accelerate training and improve model stability. These layers normalise the input to each layer, reducing the risk of vanishing or exploding gradients during backpropagation.

4. Kernel sizes: Kernel sizes in the convolutional layers were explored to identify the most appropriate size for capturing spatial features from the input

data. Smaller kernel sizes enable the model to learn fine-grained local features, while larger kernel sizes can capture more global contextual information. The chosen kernel sizes were coordinated with the kernel used in the spiral method, ensuring that the model is optimally suited to process the organised gene data.

5. Learning rate: The learning rate is a critical hyperparameter that influences the convergence speed and stability of the training process. Different learning rates were tested to identify a value that allows the model to converge to an optimal solution without oscillating or overshooting. A smaller learning rate can lead to more stable convergence but may require more iterations, while a larger learning rate can accelerate convergence but may risk instability.

6. Regularisation: Regularisation techniques, such as L1, L2, or combined L1 and L2, were assessed to prevent overfitting and improve the model's generalisation performance. These techniques add a penalty term to the loss function, encouraging the model to learn simpler, more robust representations. Additionally, dropout was employed as a regularisation method, which involves randomly deactivating a subset of neurons during training, forcing the model to learn more robust, redundant representations.

7. Number of filter maps: The optimal number of filter maps in each convolutional layer was evaluated to effectively capture the most informative spatial features from the input data. Having more filter maps increases the model's capacity to learn diverse feature representations but may also increase the risk of overfitting and computational complexity. A balance between the number of filter maps and model complexity was sought during optimisation.

8. Neurons: The number of neurons in the fully connected layers was optimised to ensure sufficient model capacity without causing overfitting. A larger number of neurons increases the model's ability to learn complex relationships among features, while a smaller number may prevent overfitting by reducing model complexity. The optimisation process aimed to identify the appropriate number of neurons to achieve a balance between model capacity and generalisation performance.

For the ViT model, largely the same hyperparameters as the CNN were optimised as the architecture was explored to determine the most optimal configuration. This included testing various numbers of transformer layers and multi-head attention layers, as well as different patch sizes and numbers. Adjusting these parameters affects the model's capacity to capture complex feature representations and spatial relationships across the input data. The patch hyperparameters, specifically the number and size of patches, were particularly important to investigate in this context. Given the novel method of optimising the patterns within each patch, it was crucial to identify an optimal configuration that effectively captured the underlying structure and relationships of the data while maintaining computational efficiency.

3.3.10 Model Training

In the training process of iterative models, I employed the **EarlyStopping** callback to halt training when the validation loss ceases to decrease for a specified number of epochs, known as "patience." The primary motivation for incorporating **EarlyStopping** is to mitigate overfitting by stopping training when the model starts demonstrating reduced performance on the validation data. This approach ensures that the model does not over-adapt to the training data and consequently performs better on unseen data. **EarlyStopping** was implemented in the LightGBM, CNN, and ViT models.

For each model type, I obtained 24 trained models, resulting from 8 folds and 3 repeats, all optimised on their individual validation datasets. This approach facilitated an assessment of the model's performance and stability across different data partitions. To generate predictions for the test datasets, I employed an ensemble technique, averaging the predictions from all 24 models. This method leverages the strengths of each individual model, leading to a more robust and stable output that is less susceptible to the peculiarities of any single model.

On the other hand, for models that did not utilise **EarlyStopping**, I optimised their hyperparameters using the 24 validation folds. This approach allowed me to explore

various hyperparameter combinations and identify the optimal configuration that yields the best performance on the validation data. Once the optimal hyperparameters were identified, I retrained the models on the entire training dataset to take full advantage of the available data and maximise their generalisation capabilities. Finally, I generated test dataset predictions using these optimally tuned and trained models, ensuring that the final models exhibit the best possible performance on unseen test data.

In the case of the ViT and CNN models, I also applied a technique known as snapshot ensembling. Snapshot ensembling is an approach designed to improve model performance and robustness by combining the predictions of multiple instances of the same model, trained using a cyclic learning rate schedule. This method is particularly useful for deep learning models, which often involve complex architectures and large parameter spaces as the ensemble can leverage diversity in the parameter space to achieve more accurate and reliable predictions.

The cyclic learning rate schedule involves splitting the training into distinct cycles, in this case, five cycles. During each cycle, the learning rate starts high and gradually decreases, allowing the model to converge towards a solution as the cycle progresses. As the learning rate becomes smaller towards the end of the cycle, the model is more likely to settle into a local minimum. To prevent the optimiser from getting trapped in these local minima, the learning rate is increased significantly at the end of the cycle, effectively restarting the optimisation process and encouraging exploration of other regions in the parameter space.

The primary benefit of employing snapshot ensembling is the increased model diversity that results from training with multiple learning rate cycles. Each cycle allows the model to explore different regions of the parameter space, potentially converging to diverse local minima (Figure 16). By averaging the predictions of these models, snapshot ensembling effectively combines their strengths and mitigates individual weaknesses, resulting in a more robust and accurate ensemble model.

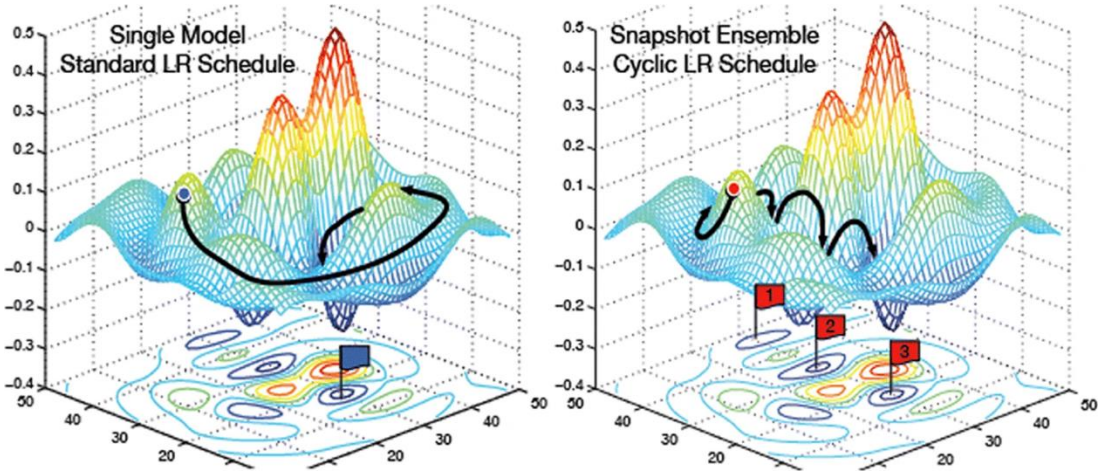

Figure 16.

Three-dimensional visualisation demonstrating the efficacy of snapshot ensembling. The graph shows multiple 'snapshots' represented by numbered markers, each indicating the weights saved at different local minima during the training process. This method enables the capture of diverse solutions and mitigates the risk of getting stuck in poor local optima. The diversity of solutions is then leveraged in an ensemble to achieve robust and superior predictive performance. Figure adopted with permission from Annavarapu et al. (2021).

By using snapshot ensembling with five cycles, I obtained 24 models per cycle, amounting to a total of 120 models for the ViT and CNN architectures. The predictions of these 120 models were averaged to produce the final test predictions. The ensemble approach, in combination with the cyclic learning rate schedule, enhances the generalisation capabilities of the models, leading to improved performance on unseen test data and increased robustness against various data distributions.

3.3.11 Feature Extraction Methods

A common criticism of deep learning (DL) techniques is their lack of interpretability, which can make it challenging to understand the relationships between input features and model predictions. In this study, I aim to identify relevant genes and decode the gene-target relationships using traditional supervised ML methods, as well as CNN and VIT models. To enhance the interpretability of the

models, I employed two approaches to uncover the relationships between input features (genes) and output targets.

The first approach focused on identifying multivariate relationships between features and the target using Gradient-weighted Class Activation Mapping[386] (Grad-CAM++). Grad-CAM++ is a powerful tool capable of reverse-engineering feature patterns in vision-based models, providing valuable insights into the model's decision-making process. It generates a saliency map that highlights the contribution of each pixel in the input image to the predicted classes. This is achieved by fusing pixel space gradient visualisation (i.e., gene expressions) with the class discriminative property, allowing the calculation of partial derivatives for each output class (i.e., virus abundance levels) with respect to the feature maps derived from the final layer before the dense layers of the CNN and ViT models.

The guided backpropagation algorithm in Grad-CAM++ selectively passes positive gradients to activated regions (interconnected genes), enabling the coarse localisation and visualisation of pixel importance based on the weighted average of the gradients for each pixel. By leveraging this method, I aimed to locate multivariate feature patterns between relevant genes and virus status, thereby highlighting the dynamic relationships between host and pathogen.

To thoroughly explore the potential multivariate patterns among genes in the feature set, I employed a strategy of randomly arranging the genes 1000 times and training 1000 corresponding models (Figure 17). This approach was adopted to ensure that genes had the opportunity to be in close proximity to other genes, thereby increasing the likelihood of capturing any relevant interactions within the data. Without implementing this strategy, I would only be able to quantify a limited number of multivariate patterns based on a single arrangement of genes. In such a scenario, patterns between distant genes in the CNN or genes situated in different patches would be less likely to be detected. By randomising the gene arrangements and training multiple models, I increased the chances of detecting significant gene-gene relationships that might have been overlooked in a single arrangement.

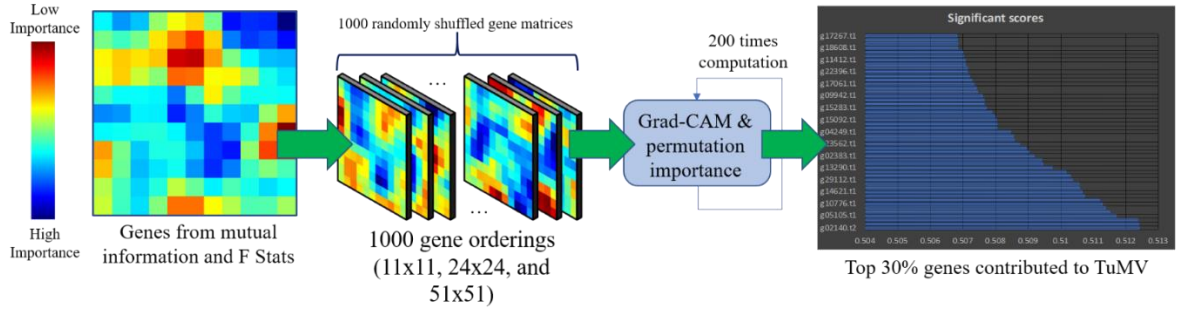

Figure 17.

Gradient-based importance score

Grad-CAM embedded with the CNN model to identify genes with multivariate patterns, and permutation importance for detecting individual genes in connections with virus abundance.

In conclusion, the application of Grad-CAM++ and related techniques allows enhanced interpretability of the CNN model by identifying multivariate relationships between genes and viruses. This improved understanding of gene-virus relationships can provide valuable insights into the complex interactions between host and pathogen, paving the way for more targeted and effective treatment strategies.

The second approach I employed focuses on identifying individual genes that are crucial in relation to the target. For this purpose, I utilised the permutation feature importance method[387] to gauge the significance of a feature (i.e., a gene) by calculating the increase in the model's prediction error after permuting the feature. A feature is considered important if shuffling its values leads to an increase in the model error, indicating that the model relies on this specific feature for its predictions. Conversely, a feature is deemed unimportant if rearranging its values does not affect the model error.

Although the permutation feature importance method offers a highly compressed and global insight into the model's behaviour, it is worth noting that this approach may weaken the measurement of interactions between features. This limitation arises because the method focuses on the importance of individual features rather than considering the complex relationships between multiple features. Nevertheless, by employing this method, I was able to identify individual genes that played a crucial role in predicting the interactions between the host and the virus.

In summary, the second approach, using the permutation feature importance method, enabled pinpointing individual genes that were essential for predicting host-virus interactions. While this method provides valuable insights into the significance of individual features, it may not fully capture the intricate relationships between multiple features. By combining this approach with techniques that reveal multivariate relationships, such as Grad-CAM++, it is possible to obtain a more comprehensive understanding of the complex gene-virus interactions, ultimately contributing to more effective treatment strategies.

Interpreting the features from trained LightGBM models enabled identifying the most predictive genes, which are crucial for understanding gene-virus interactions and potentially developing targeted treatment strategies. To interpret the features and identify the most predictive genes, I analysed the feature importances derived from the trained LightGBM models. Feature importance in LightGBM is typically measured using two approaches: gain and split. Gain refers to the improvement in the training loss that results from splitting a feature, while split denotes the number of times a feature is used to split the data. I used the split method to analyse feature importances, hoping to provide insight into which genes contribute the most to the model's performance and the prediction of virus presence or abundance levels. I then ranked the genes based on their feature importances in the LightGBM models and selected the most relevant ones for further analysis.

3.3.12 Model Evaluation Criteria

In this study, some of the target distributions are slightly imbalanced, rendering accuracy an inadequate evaluation metric for assessing a model's performance. Accuracy, defined as the proportion of correct predictions to total predictions, may be misleading in imbalanced datasets, as it does not account for the unequal distribution of classes. Consequently, I selected the F1-score, a more proficient evaluation metric for imbalanced datasets, alongside AUROC and accuracy, to provide a comprehensive assessment of the model's performance.

It is defined as the harmonic mean of precision and recall:

$$F1 = \left(\frac{p^{-1} + r^{-1}}{2}\right)^{-1}$$

where *p* and *r* are precision and recall that are defined as:

$$p = \frac{t_p}{t_p + f_p}, r = \frac{t_p}{t_p + f_n}$$

with t_p, f_p and f_n denoting true positive, false positive and false negative predictions respectively. F1-score is weighted to distribute the importance of each class more evenly and used to represent a model's performance.

In addition to the F1-score, I used the Area Under the Receiver Operating Characteristic (AUROC) as an evaluation metric. AUROC measures the model's ability to distinguish between different classes, illustrating the trade-off between true positive rate (sensitivity) and false positive rate (1-specificity) across varying decision thresholds. An AUROC of 0.5 indicates random classification, whereas a value of 1 signifies perfect classification. AUROC is particularly useful when the class distribution is imbalanced or when the cost of false positives and false negatives varies. Accuracy was still included as it is easy to interpret due to it directly reflecting the percentage of correct predictions made by the model.

By employing F1-score, AUROC, and accuracy as evaluation metrics, I ensured a comprehensive assessment of the model's performance, accounting for the inherent challenges posed by imbalanced datasets and the varying costs of false positives and false negatives. Users can choose to evaluate a model using a specific metric depending on the distribution of their target variable and the associated penalty with type one or type two errors.

3.3.13 Joint Gene Importance Network

Rather than interpreting the gradient-based importance scores individually, I thought they should be analysed in a multivariate manner as they are dependent on the gene arrangement of the input matrix. To do this, I first applied local thresholding to each heatmap to segment and extract clusters of highly important genes. Once these

clusters were extracted across the 1,000 models, a metric that quantifies the joint dependence between all genes with each other was applied to the extracted clusters to generate a matrix of joint dependency scores, shown in the following equation.

$$S_{i,j} = \frac{\sum_{k}^{n} \frac{g_{i,k} \times g_{j,k}}{\sqrt{\left(x_{i,k} - x_{j,k}\right)^2 + \left(y_{i,k} - y_{j,k}\right)^2}}}{n}$$

where $S_{i,j}$ denotes the joint dependency score between gene i and j, k denotes the iteration number, $g_{i,k}$ denotes the Grad-CAM++ important score for gene i in iteration k, n denotes the total number of iterations, and $x_{i,k}$ and $y_{i,k}$ denote the x and y coordinates for gene i in the input matrix for iteration k.

To effectively visualise the joint dependencies between genes captured in the matrix of joint gene dependency scores, I devised a Python script utilising the py2cytoscape package. This script imports the data into the Cytoscape software and generates an informative gene network. In this network, the strength of the joint dependence is represented by the edge width, while the Grad-CAM++ importance score is depicted by the node size. Additionally, the colour of the nodes corresponds to various gene functions identified using the ClueGO app within Cytoscape. The Trans-Learn software provides users with the flexibility to adjust thresholds for joint gene dependencies, enabling them to tailor the analysis to their specific needs. Visualising genes and their interactions in a 2D graph not only simplifies the identification of patterns but also facilitates clustering of genes based on gene ontology IDs. This approach helps pinpoint key genes that act as connectors between clusters while providing relevant statistical information (Figure 18).

By adopting this visualisation method, researchers can gain valuable insights into gene interactions, better understand the functional connections between genes, and potentially unveil novel relationships and mechanisms that drive the host-virus dynamics.

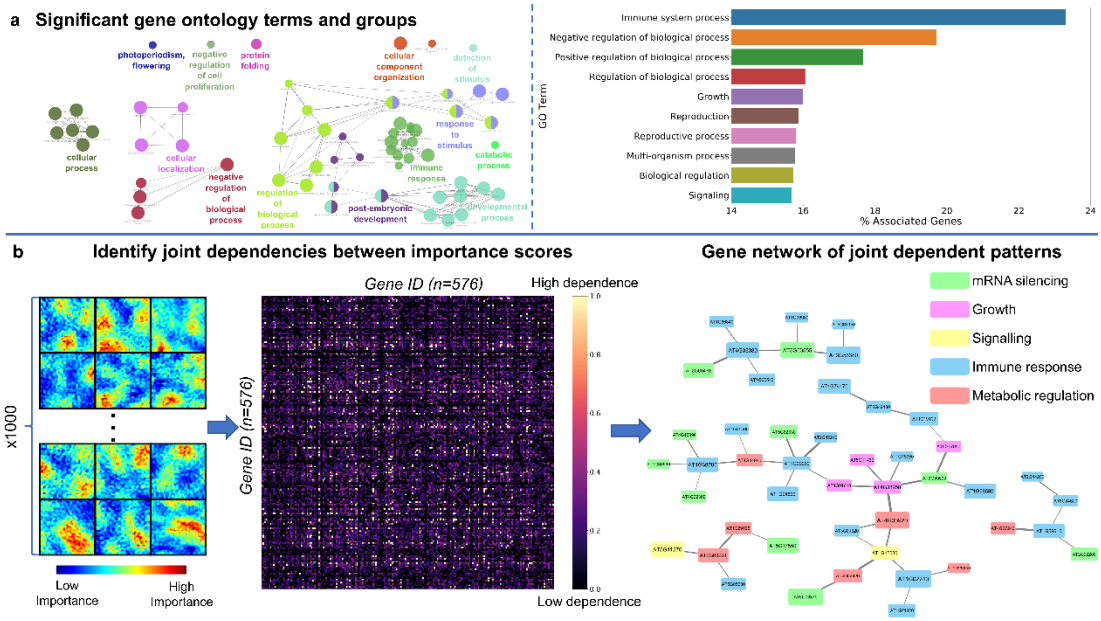

Figure 18.

Joint Gene Importance and Network analysis

(a) *Significant gene ontology terms and network associated with the identified multivariate biomarkers*

(b) *Joint dependency calculated between genes using 1000 Grad-CAM++heatmaps to produce a network of multivariate biomarker dependencies*

3.3.14 Gene Ontology Analysis

In order to biologically interpret the sets of genes identified by Trans-Learn, I applied Gene Ontology (GO) analysis. GO provides a consistent and logical description of the biological processes, molecular functions, and cellular components of gene products. I utilised this technique to categorise the genes located through Grad-CAM++ and permutation importance into bins according to their functional characteristics.

By creating a functional profile of the gene set, I aimed to better correlate the identified genes with the underlying biological processes. I then employed a statistical test to rank each bin, evaluating whether it is enriched for the input genes at a P-value of 0.05 after applying the Holm-Bonferroni correction. For the GO analysis, I used the ClueGO package within Cytoscape. ClueGO utilises Cohen's kappa coefficient to measure the similarity between GO terms, allowing for the

generation of GO groups. Each group consists of similar GO terms, which enhances the interpretability of the GO results when visualised in a network.

In addition to utilising the ClueGO app within Cytoscape, I coded a gene ontology analysis module within the Trans-Learn software. To accomplish this, I obtained Biological Process ontology terms for a number of plant species as well as homosapiens. I then pre-processed the data to match gene identifiers in the GO terms with those in the gene identifier list. This involved converting gene identifiers to strings and using regular expressions to extract relevant identifiers.

Next, I created a copy of the filtered GO terms dataset and formatted the Ensembl gene names by converting them to uppercase and removing gene isoforms. I then filtered the gene list to include only those with the "ENS" prefix. To identify the genes associated with GO terms, I iterated through the gene list and checked if they were present in the GO term dataset. This resulted in a list of genes that had at least one annotation in the GO Biological Processes.

To evaluate the enrichment of the gene list in specific GO terms, I performed a hypergeometric test, a statistical method that calculates the probability of obtaining a specific number of "successes" (genes associated with a specific GO term) in a fixed number of "draws" (genes in the list) without replacement, given the total number of "successes" in the population (genes associated with the specific biological process). The hypergeometric test is based on the following probability mass function:

$$P(X = k) = \frac{\binom{m}{k} \cdot \binom{N - m}{n - k}}{\binom{N}{n}}$$

where:

- N is the total number of genes considered (gene universe).
- n is the number of identified genes.
- m is the number of genes in the gene ontology category.
- k is the number of identified genes in the gene ontology category.

- $\binom{a}{b}$ denotes the number of combinations of choosing b items from a set of size a.

For each GO term, I calculated the p-value using the hypergeometric distribution function from the SciPy library (**stats.hypergeom**). The p-value represents the probability of observing the number of genes in the list that are associated with the specific GO term by chance, considering the background distribution of all gene annotations in GO Biological Processes. A lower p-value indicates a stronger association between the gene list and the specific GO term, suggesting that the enrichment is unlikely to have occurred by chance.

In the custom analysis, the hypergeometric test is performed for each GO term, and the resulting p-values are adjusted for multiple testing using the Bonferroni correction. The final list of significantly enriched GO terms is determined by filtering the results to include only those with adjusted p-values less than 0.05. Finally, I exported the list of enriched genes to a CSV file for further analysis.

3.3.15 GUI Pipeline Manager

I developed a GUI (Graphical User Interface) Pipeline Manager in Python using the Tkinter library, which offers an intuitive and user-friendly interface for managing the entire analysis workflow. This GUI Pipeline Manager allows users of varying programming skills to interact with the underlying methods and algorithms without directly manipulating the code, making it more accessible to a diverse range of users.

The GUI Pipeline Manager is designed to facilitate the selection of various parameters and the execution of methods in an organized and sequential manner. The pipeline is structured as follows:

1. **Data Loading**: Users can upload their dataset in a compatible format, such as CSV or TXT files. The GUI provides an option to browse and select the file from directories on their local system.
2. **Data Splitting**: Users can customise the proportions to split their datasets into train, validation, and test sets, allowing them to control the balance

between model training and evaluation. The GUI offers flexibility in choosing various data splitting strategies, such as random splits or stratified splits based on class labels. Additionally, users can opt for a cross-validation approach by specifying the number of repeats and folds. This feature enables users to fine-tune the model evaluation process and achieve more robust performance estimates.

3. **Data Preprocessing**: Users can define parameters for data filtering, such as the proportion of genes to filter and the variance threshold for filtering. The pipeline applies these filters to the dataset, effectively removing low-variance genes and retaining only the most informative ones. This step helps reduce noise and improve model performance by focusing on relevant features. The GUI also offers a choice of normalisation methods, such as standard scaling, box-cox transformation, or log transformation, to ensure that data is appropriately scaled and distributed. If seasonal normalisation is selected, the pipeline normalises the data based on the underlying seasonal trends, further enhancing data quality and model interpretability.

4. **Model Selection:** Users can select the model or ensemble of models to train on their pre-processed data. The GUI allows users to choose from a comprehensive list of available models, including traditional machine learning models such as decision trees and support vector machines, as well as advanced deep learning models like convolutional neural networks. If a vision-based model is selected, the data is transformed into an image grid representation, and users have the option to specify the arrangement method, either spiral or patch, for organising the grid. This customisation allows users to tailor the model selection process to best suit their dataset and analysis objectives.

5. **Model Hyperparameter Tuning:** If users desire, they can apply automated hyperparameter tuning to the selected model using the **hyperopt** and **hyperas** Python packages. This step involves exploring different combinations of hyperparameters to find the optimal configuration that yields the best performance. Although this process can significantly increase the

time taken for the pipeline to complete, it can lead to improved performance and generalisation capabilities of the selected model(s).

6. **Model Training:** After the model or ensemble selection, users can specify additional training parameters to fine-tune their models further. Options such as snapshot ensembling or ensembling models optimised on validation folds are available for enhancing model robustness and accuracy. The pipeline then proceeds to train the chosen model(s) on the pre-processed data, taking into account the specified settings.

7. **Model Interpretation:** Users have the flexibility to select a preferred method for interpreting the trained models, such as Grad-CAM++ or permutation importance. The pipeline applies the selected interpretation technique to the trained model(s), allowing users to identify critical genes and their interactions that contribute to the model's predictions. This step is crucial for understanding the biological significance and relevance of the model's results.

8. **Gene Ontology Analysis:** The pipeline conducts a gene ontology (GO) analysis on the identified genes, categorising them into bins based on their functional characteristics. Users can specify the desired p-value threshold for the enrichment analysis to control the stringency of the results. If Cytoscape is installed on the user's system, it automatically opens and performs a gene ontology analysis using the ClueGO app, as well as visualises jointly dependent gene networks. Alternatively, users can opt for the built-in module to conduct the gene ontology analysis if preferred.

9. **Visualisation and Reporting:** At the final stage, the pipeline generates comprehensive visualisations and reports that summarise the results. These include gene networks, model performance metrics, enriched GO terms, and key genes connecting clusters. Users can interact with these visualisations to delve deeper into the data, explore relationships between genes, and gain valuable insights into the underlying biological processes driving the observed patterns. This step empowers users to make informed decisions based on the analysis results and fosters a deeper understanding of the biological systems under study.

The GUI pipeline manager serves as a powerful tool that not only simplifies the process of interacting with the analysis workflow but also empowers researchers to explore gene expression datasets in innovative ways across various areas of biology. By providing a user-friendly interface and a well-organised pipeline, the tool integrates every step of the analysis, from data loading to visualisation and reporting, making it more accessible to users with varying levels of programming expertise.

Within the field of plant biology, the GUI Pipeline Manager has potential applications in crop improvement, plant-environment interactions, understanding plant development, and plant-microbe interactions. For example, researchers can use the tool to identify genes associated with desirable traits such as drought tolerance, disease resistance, or increased yield, potentially leading to the development of novel crop varieties or precision breeding strategies to enhance agricultural productivity and sustainability. Additionally, the tool can facilitate the study of plant adaptation to changing environmental conditions, informing strategies to mitigate the effects of climate change on agriculture. It can also be employed to uncover the molecular mechanisms underlying various plant developmental processes and identify plant-specific molecular profiles that modulate plant-microbe interactions.

The generalisability of the pipeline extends its applicability to gene expression data from other areas of biology, such as animal and human genomics, microbiology, and ecology. Moreover, the pipeline can handle any tabular dataset, allowing researchers to analyse diverse types of data, including those generated from different high-throughput omics technologies. This flexibility enables the pipeline to be employed in the analysis of protein expression data, providing insights into protein abundance, interactions, and their functional roles in biological systems.

3.4 Results

In this results section, I will present a comprehensive analysis of the performance and outputs of the supervised machine-learning models within the Trans-Learn system. I will discuss the overall performance of the pipeline in terms of accuracy, specificity, sensitivity, and other relevant metrics, comparing it to existing methodologies such as DeepFeature and differential expression linear models. This comparison will shed light on the advantages and limitations of my approach, highlighting its effectiveness in addressing complex gene expression data analysis tasks.

Additionally, I shall investigate jointly dependent genes detected by the pipeline and explore their potential biological significance. The gene ontology analysis performed on these genes will be discussed, providing insights into their functional roles and interactions within the biological systems under investigation. By presenting these results, I aim to demonstrate the pipeline's capacity to uncover meaningful patterns and relationships in gene expression data, ultimately contributing to a deeper understanding of the underlying biological processes.

3.4.1 Accuracy and Error of Predictions

The trained machine learning models were tested on the three test datasets and one validation datasets: (1) TuMV detection and severity classification, (2) COVID-19 diagnosis, (3) cancer subtype classification, (4) wheat tissue type classification. The accuracy, F1, and AUROC were recorded for each of the models across the four datasets to compare performance.

3.4.1.1 Turnip Mosaic Virus

This subsection details the predictive performance of different machine learning models applied to the Turnip Mosaic Virus (TuMV) detection task. The model results are summarised in Table 3, highlighting the accuracy and F1-score for each model with and without seasonal normalisation.

Table 2 presents the results of eight different models, namely K-Nearest Neighbors (K-NN), Convolutional Neural Networks (CNN), Logistic Regression (LR), Light Gradient Boosting Machine (LGBM), Artificial Neural Networks (ANN), Support Vector Machine (SVM), and a Stacking ensemble (Stack) excluding CNN. The table compares the results based on two metrics: Accuracy and F1-score, each under two different conditions - without and with seasonal normalisation.

Table 3. TuMV virus presence model results with and without seasonal normalisation applied. Best individual and ensemble model performances scores are bolded.

Model	**Accuracy** ***(without seas. norm.)***	**Accuracy** ***(with seas. norm.)***	**F1** ***(without seas. norm.)***	**F1** ***(with seas. norm.)***	AUROC ***(without seas. norm.)***	AUROC ***(with seas. norm.)***
K-NN	0.815	0.806	0.802	0.791	0.822	0.813
CNN	0.836	0.900	0.822	0.893	0.831	0.904
LR	0.822	0.815	0.817	0.808	0.828	0.822
LGBM	0.858	0.879	0.853	0.866	0.861	0.881
ANN	0.808	0.822	0.799	0.817	0.813	0.827
SVM	0.822	0.822	0.817	0.817	0.828	0.828
ViT	**0.865**	**0.921**	**0.863**	**0.918**	**0.873**	**0.928**
Stack	0.886	**0.935**	0.879	**0.931**	0.889	**0.942**
Blend	**0.892**	**0.935**	**0.885**	**0.931**	**0.896**	**0.942**

The models included in this analysis showed a range of performance, highlighting the distinct strengths of various machine learning approaches. The K-NN model, while simple and intuitive, underperformed compared to other models across all metrics, with accuracy scores ranging from 0.806 to 0.815 and F1 scores from 0.791 to 0.802. The CNN model showed marked improvement with seasonal normalisation, boosting its accuracy from 0.836 to 0.900 and F1 score from 0.822 to 0.893. This suggests that the CNN model was able to leverage the patterns and structure in the seasonally adjusted data to make more accurate predictions.

Light Gradient Boosting Machine (LGBM) model and Artificial Neural Networks (ANN) both demonstrated higher performance than the traditional models, with LGBM achieving an accuracy of 0.879 and F1 score of 0.866 with seasonal normalisation, reflecting the power of ensemble learning and advanced neural architectures in capturing complex patterns in the data.

The Vision Transformer (ViT) model delivers the best individual model performance, with accuracy, F1-score, and AUROC reaching 0.921, 0.918, and 0.928 respectively when seasonal normalisation is applied. However, the stacked ensemble model and the blended model both show even higher performance, indicating the benefits of combining the strengths of diverse models. The efficacy of the embeddings created by the ViT model to distinguish uninfected and infected samples can be seen in Figure 19.

The application of seasonal normalisation generally led to performance improvements across most models, particularly for complex models like CNN and ViT. Seasonal normalisation essentially removes cyclical patterns attributable to the seasonality in the data, allowing the models to better focus on the underlying trend. This process seems to provide particularly advantageous for models capable of capturing complex non-linear patterns in the data, as they can better exploit the cleaner, trend-focused data resulting from normalisation. However, as seen with LR and SVM, not all models benefit equally from this pre-processing step, emphasising the need for tailored strategies based on the chosen model’s characteristics and assumptions.

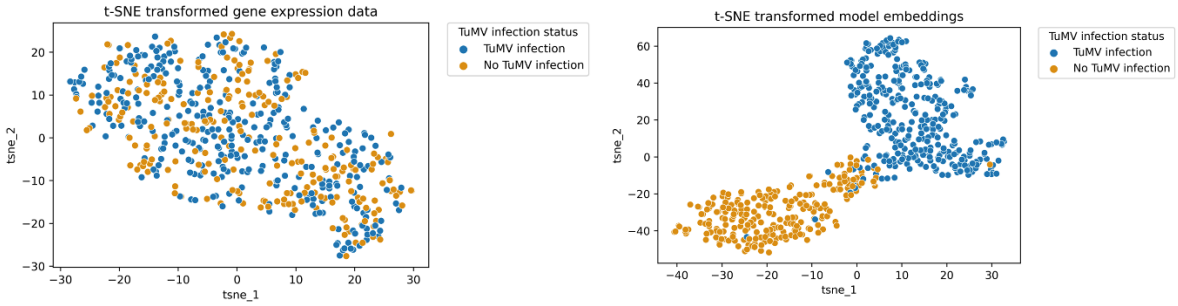

Figure 19.

(Left) tSNE transformed gene expression data for the TuMV dataset where orange datapoints are uninfected samples, and blue datapoints are infected samples. (Right) tSNE transformed ViT model embeddings for the TuMV dataset.

Continuing from the binary classification problem of predicting the presence of TuMV infection, the next task was to predict the severity of the infection, which is a multiclass classification problem. This task is inherently more complex due to the additional classes and the inherent order in the severity levels.

The results for this task, using the same models as before, are presented in Table 4. Similar to the previous task, the performance metrics used are accuracy and F1-score, and the results are presented both with and without seasonal normalisation.

Table 4. TuMV virus severity model results with and without seasonal normalisation applied. Best individual and ensemble model performances scores are bolded.

Model	**Accuracy** ***(without seas. norm.)***	**Accuracy** ***(with seas. norm.)***	**F1** ***(without seas. norm.)***	**F1** ***(with seas. norm.)***	AUROC ***(without seas. norm.)***	AUROC ***(with seas. norm.)***
K-NN	0.737	0.812	0.721	0.798	0.744	0.819
CNN	0.812	0.900	0.802	0.892	**0.817**	**0.906**
LR	0.843	0.872	0.836	0.866	0.848	0.877
LGBM	0.843	0.900	0.834	0.892	0.847	0.903
ANN	0.836	0.865	0.827	0.857	0.841	0.870
SVM	0.772	0.844	0.763	0.835	0.779	0.851
ViT	**0.857**	**0.907**	**0.849**	**0.899**	**0.864**	**0.913**
Stack	**0.871**	**0.914**	**0.864**	**0.907**	**0.878**	**0.921**
Blend	0.857	0.907	0.849	0.899	0.864	0.913

This time, the K-NN model benefitted from the seasonal normalisation with a sharp increase in accuracy from 0.737 to 0.812, and a comparable rise in the F1-score, from 0.721 to 0.798. Likewise, the CNN model continued to exhibit substantial improvements with seasonal normalisation. However, it's worth noting that while it showed marked improvements with normalisation in both tasks, the relative gains appeared to be more substantial in this multiclass classification scenario. This could suggest that CNNs are particularly effective at leveraging normalised data for more complex classification tasks.

The ViT model was a top performer in the binary classification task, and it maintained this strong performance in the multiclass classification scenario, showing the effectiveness of this model across different task complexities. It's worth noting that the ViT model scores are with the spatial JMI patch arrangement method. As with the binary classification task, seasonal normalisation generally boosted model performance, particularly for models like CNN and ViT, that can exploit complex non-linear patterns in data.

Finally, the stacked ensemble model once again outperformed all individual models, achieving the highest accuracy of 0.914 and an F1-score of 0.907 with seasonal normalisation. This reinforces the value of ensemble methods, which leverage the strengths of multiple individual models to achieve superior performance.

3.4.1.2 TCGA Cancer Subtype Classification

The results of the supervised machine learning models applied to the Cancer Genome Atlas (TCGA) for cancer subtype classification are presented in Table 5.

Table 5. TCGA cancer subtype classification model results with and without seasonal normalisation applied. Best individual and ensemble model performances scores are bolded.

Model	Accuracy	F1	AUROC
K-NN	0.949	0.947	**0.952**
CNN	0.978	0.977	0.979
LR	0.962	0.961	0.964
LGBM	0.978	0.977	0.979
ANN	0.973	0.971	0.975
SVM	0.963	0.961	0.965
ViT	**0.995**	**0.994**	**0.997**
Stack	**0.995**	**0.994**	**0.997**
Blend	**0.995**	**0.994**	**0.997**

The K-NN model again underperforms compared to other models but performs well in that all metrics surpass 0.94. This performance intimates the potential clustering characteristics of the TCGA dataset, wherein each cancer subtype may inhabit a distinct region in the gene expression (input feature) space. Both CNN and LGBM models yield identical results, with an accuracy and F1 score of 0.978 and an AUROC of 0.979. CNN, with its capability to recognise spatial hierarchies, and LGBM, employing a gradient boosting framework, appear to manage to capture the

intricacies of the classification task effectively. Conventional models such as LR and SVM maintain impressive performance, surpassing an accuracy and F1 score of 0.96. Unsurprisingly, the ANN model improves upon LR and SVM performances by achieving accuracy and F1 metrics above 0.97. This underlines the capacity of neural networks to encapsulate complex, non-linear relationships within the dataset. The Vision Transformer (ViT) model emerges as the best performing individual model for this task, yielding near-perfect accuracy, F1, and AUROC values (0.995, 0.994, and 0.997 respectively).

Visual evidence of the ViT model's remarkable performance can be seen in Figure 20, which provides t-SNE transformations of the ViT model embeddings for the TCGA dataset. Clearly delineated clusters, each corresponding to a distinct cancer subtype, demonstrate the ViT model's proficiency in distinguishing among different cancer subtypes in the gene expression data. Mirroring the performance of the ViT model, both the stacked ensemble model and the blended ensemble model achieve identical accuracy, F1, and AUROC scores. Notably, the ViT outperforms the best DeepFeature model (DeepFeature t-SNE with Snowfall) that scored 97.9% accuracy on the same test dataset.

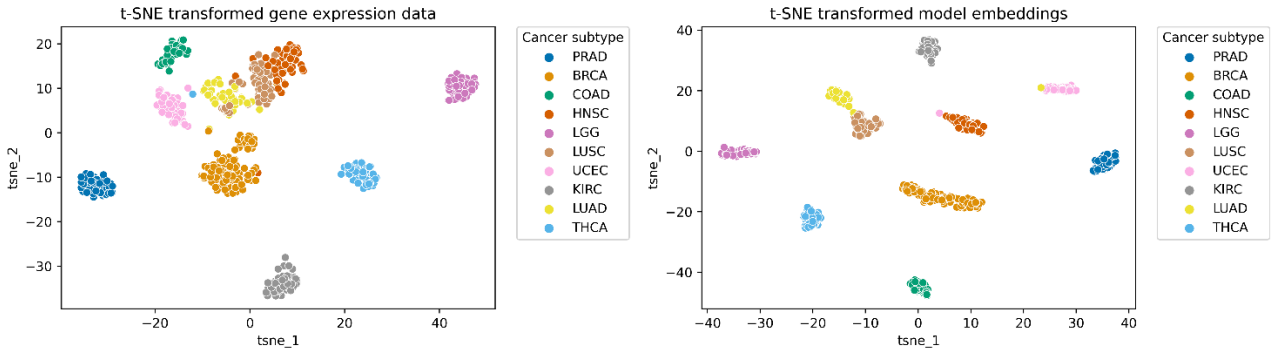

Figure 20.

(Left) tSNE transformed gene expression data for the TCGA dataset where the colour of datapoints correspond to the cancer subtype associated with the sample. (Right) tSNE transformed ViT model embeddings for the TCGA dataset.

3.4.1.3 COVID-19 Detection

The results of the supervised machine learning models applied to the COVID-19 dataset for diagnostic purposes are presented in Table 6.

Table 6. COVID-19 binary classification model results

Model	Accuracy	F1	AUROC
K-NN	0.958	0.956	0.959
CNN	0.979	0.976	0.981
LR	0.958	0.958	0.959
LGBM	0.979	0.978	0.981
ANN	0.969	0.967	0.972
SVM	0.948	0.947	0.950
ViT	**1.000**	**1.000**	**1.000**
Stack	**1.000**	**1.000**	**1.000**
Blend	**1.000**	**1.000**	**1.000**

The performance across the models is notably high, suggesting highly distinguishable gene expression patterns between infected and uninfected samples. K-NN, CNN, LR, LGBM, ANN, and SVM all surpass an accuracy and F1 score of 0.94, with CNN and LGBM again delivering an identical performance, achieving accuracy and AUROC above 0.97.

The ViT model, along with the Stack and Blend ensemble models, demonstrates flawless performance, with all metrics reaching the perfect score of 1.00. This suggests an exemplary ability of these models to differentiate between gene expression profiles of COVID-19 infected and uninfected samples. Figure 21 solidifies the strong performance of the ViT model as the t-SNE visualisation of the ViT model embeddings for the COVID-19 dataset shows clear separation of infected and uninfected samples.

Overall, these findings reflect the considerable potential of advanced machine learning models, especially transformer-based models and ensemble strategies, in effective, precise, and robust COVID-19 diagnosis using gene expression data.

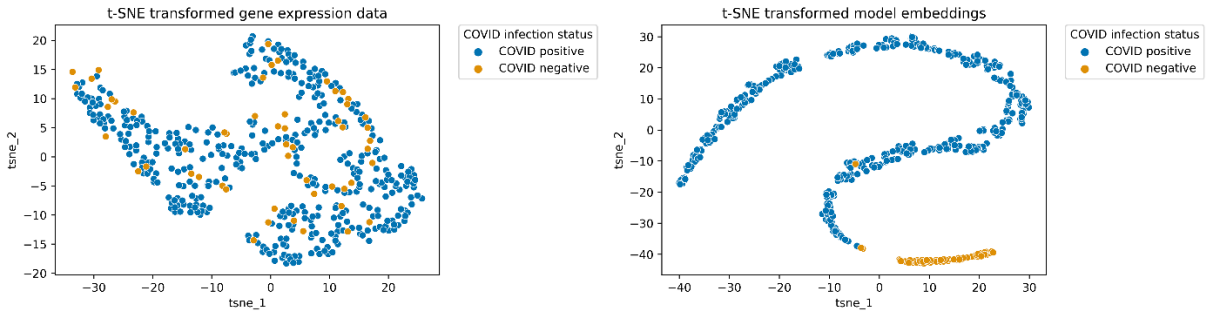

Figure 21.

(Left) tSNE transformed gene expression data for the COVID-19 dataset where orange datapoints are uninfected samples, and blue datapoints are infected samples. (Right) tSNE transformed ViT model embeddings for the COVID-19 dataset.

3.4.1.4 Wheat Tissue Type Classification

The results of the supervised machine learning models applied to the wheat tissue type dataset are presented in Table 7. Due to the test samples not having labels, these results are cross validation scores from the pooled dataset. However, they still provide an estimate of the performance of each model if it was to predict on unseen data.

Table 7. Wheat tissue type classification cross validation model results

Model	Accuracy	F1	AUROC
K-NN	1.000	1.000	1.000
CNN	1.000	1.000	1.000
LR	1.000	1.000	1.000
LGBM	1.000	1.000	1.000
ANN	1.000	1.000	1.000
SVM	1.000	1.000	1.000
ViT	**1.000**	**1.000**	**1.000**
Stack	**1.000**	**1.000**	**1.000**

While these results initially seem promising, the uniform perfection across all models does not provide detailed insights into the comparative performance of the different models. The perfect scores suggest that the pooled wheat tissue type dataset, perhaps due to the feature selection applied or inherent characteristics of the data, is a problem well-suited to all the applied models. Therefore, these findings do not offer a nuanced understanding of the performance trade-offs between different types of models.

Figure 22 displays scatterplots of pooled and raw wheat transcriptomic datasets after the application of principal component analysis. The diagrams indicate improved separation between tissue types after feature selection, further suggesting that the data may be inherently well-structured for accurate classification. The perfect performance of the models may hence be a testament to the power of the data pre-processing and feature selection rather than the individual models' complexity or capability.

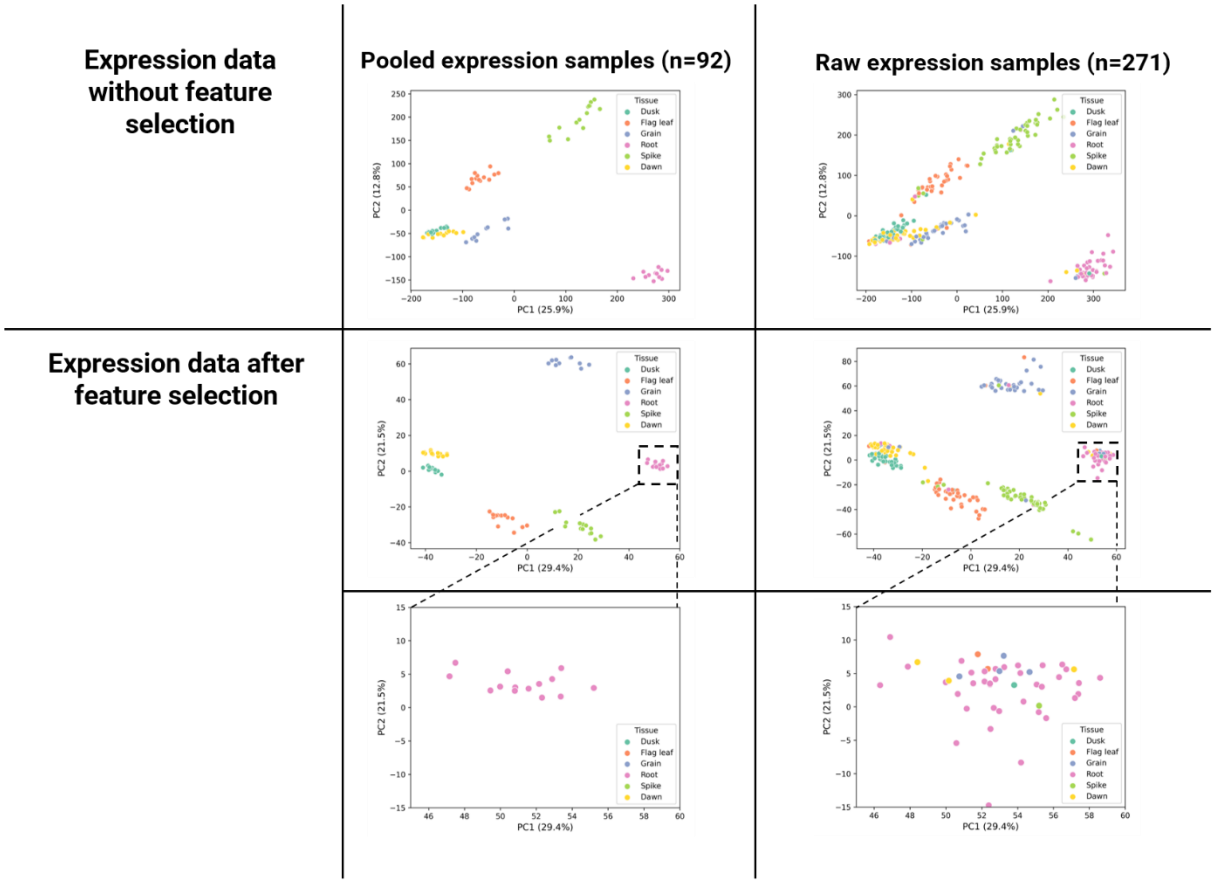

Figure 22.

Pooled and raw wheat transcriptomic datasets after applying principal component analysis.

Top row (left to right) – Scatterplot of pooled samples showing separation between most tissue types except for dusk and dawn, scatterplot of raw samples showing lots of overlap due to mislabelled samples.

Bottom rows (left to right) – Improved separation between tissue types after feature selection, scatterplot of pooled samples showing separation between all and no overlapping of samples in the root cluster, scatterplot of raw samples still showing overlap with the root cluster containing samples from four other tissue types.

Figure 23 provides a visualisation of the decision boundaries for the Support Vector Machine (SVM) model when predictions were generated for the test samples. Using these predictions, Upon generating the predictions with the SVM model, 42 test samples were predicted as having incorrect tissue type labels. This identification process not only underscores the proficiency of Trans-Learn as a quality control tool but also demonstrates how machine learning models can effectively identify potential anomalies or errors in large and complex biological datasets. Consequently,

these 42 samples were relabelled according to the SVM's predictions, enhancing the dataset's integrity and ensuring accurate downstream analyses.

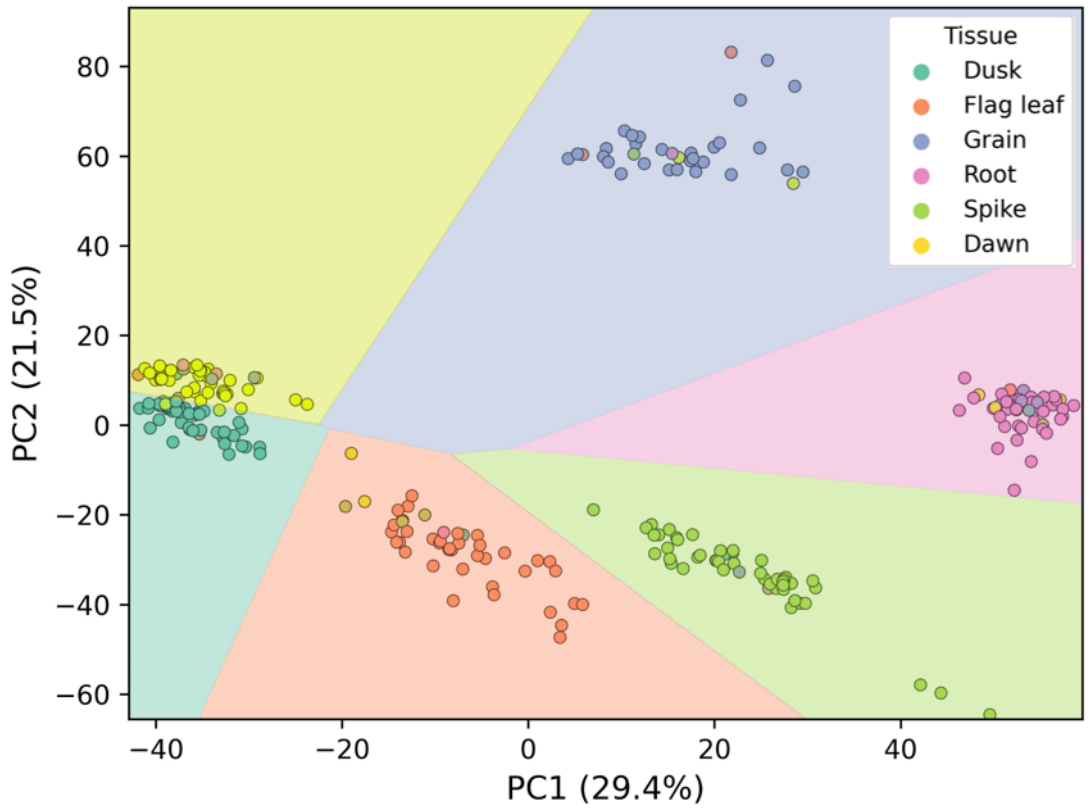

Figure 23.

Two dimensional representation of decision boundaries classifying tissue type for the raw samples (n=271)

In summary, the results demonstrate the successful classification of wheat tissue types by all models in the pooled samples. Still, the homogeneity in the perfect scores across the models limits the inference about the strengths and weaknesses of individual models in this context.

3.4.2 Spatial Gene Arrangement Performance

The application of Joint Mutual Information (JMI) to arrange patches for the ViT model and genes for the CNN model in the four tasks demonstrates significant potential for improving model performance. To investigate the efficacy of this

approach, I calculated the JMI of patches, comparing these results to the average JMI when genes were arranged randomly.

I observed that the JMI of the patch arrangement consistently outperformed the average of the random gene arrangements (Figure 24). This suggests that the spatial arrangement of genes plays a crucial role in predictive models, and that the JMI of the patch arrangement provides a more effective method for gene organisation.

The efficacy of the JMI patch arrangement method is visually represented in the heatmap of JMI, which shows a distinct gradient of JMI values, decreasing from the top right to the bottom left. This pattern indicates that the algorithm is functioning as expected, as the spatial organisation of genes in the patches systematically affects the JMI.

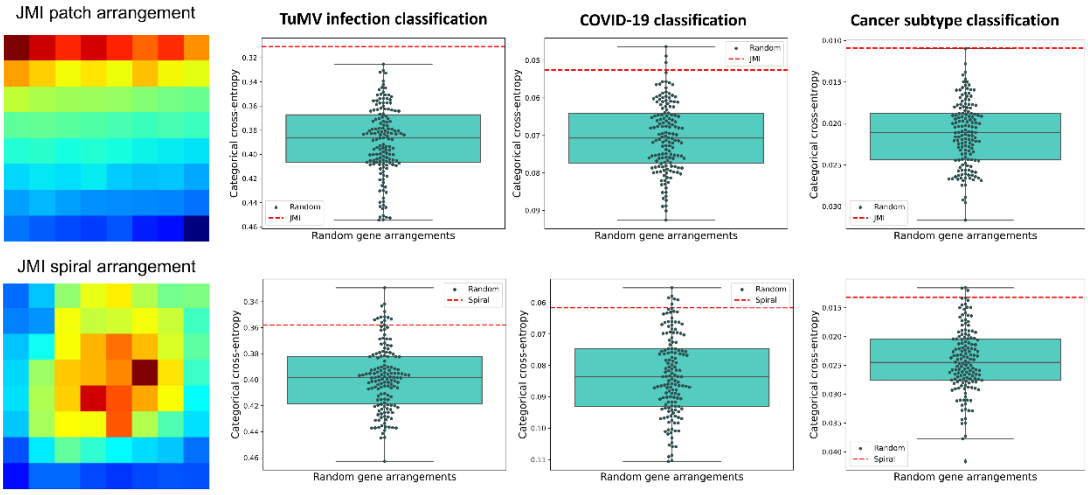

Figure 24.

(Top Row) – Heatmap displaying the joint mutual information (JMI) of the patches, boxplots with datapoints overlayed to display the distribution of errors when genes were randomly arranged in patches compared to the dashed red line which indicates the error of the JMI patch arrangement method.
(Bottom Row) - Heatmap displaying the joint mutual information (JMI) of the genes in a 3x3 kernel neighbourhood, boxplots with datapoints overlayed to display the distribution of errors when genes were randomly arranged compared to the dashed red line which indicates the error of the JMI spiral arrangement method.

In the task of classifying Turnip Mosaic Virus (TuMV) infection, the JMI patch arrangement method notably outperformed all 150 models based on random gene arrangements. Specifically, the JMI patch arrangement method achieved a cross-entropy error lower than 0.32, while the median cross-entropy error for models with random gene arrangements was 0.39. This significant reduction in error underscores the importance of gene arrangement for spatial models and highlights the superior performance of the JMI patch arrangement method in this task.

The COVID-19 diagnosis task also demonstrated the strength of the JMI patch arrangement method. When compared to 150 models with random gene arrangements, the JMI patch arrangement method scored in the top 3 percentile of errors. This result further attests to the efficacy of the JMI patch arrangement method in achieving lower prediction error rates, even in the complex task of COVID-19 diagnosis.

Lastly, in the TCGA cancer subtype classification task, the JMI patch arrangement method again outperformed all 150 models with random gene arrangements. The method achieved a median entropy error of 0.011, significantly lower than the median entropy error of 0.022 for models with random gene arrangements. This result reiterates the power of the JMI patch arrangement method in handling high-dimensional, complex classification tasks such as cancer subtype classification.

In conclusion, these results demonstrate the superiority of the JMI patch arrangement method over random gene arrangements in various tasks. The method consistently achieves lower error rates, indicating its potential for improving the accuracy of spatial models with diverse applications, from diagnosing diseases to predicting other complex traits. Future work may explore the utility of this approach in other domains and refine its implementation for even greater predictive performance.

My investigation extended to the application of the spiral method for arranging genes in Convolutional Neural Network (CNN) models. The efficacy of this method was validated using heatmaps to display the Joint Mutual Information (JMI) in a neighbouring 3x3 area. The heatmaps provided a clear illustration of the impact of

the spiral arrangement method on the JMI, offering visual confirmation of its efficacy.

Across the diverse classification tasks of Turnip Mosaic Virus (TuMV) infection, The Cancer Genome Atlas (TCGA) cancer subtype, and COVID-19 diagnosis, the CNN model trained on spirally arranged genes consistently outperformed the upper quartile of the CNN models using random gene arrangements. This finding highlights the utility of the spiral arrangement method in improving the performance of CNN models on tabular datasets.

However, while the spiral arrangement method consistently yielded performance within the top 10 percentile, it never surpassed the best model based on randomly arranged genes. This observation suggests that while the spiral method offers a reliable heuristic for improving CNN model performance, there may be other factors, potentially specific to each task, that can lead to even better model performance.

When compared to the JMI patch method applied to the Vision Transformer (ViT) model, the spiral method for the CNN was found to be less effective. Despite its consistent performance in the top quartile, the spiral method never outperformed the best performing model using the JMI patch arrangement in the ViT model. This finding suggests that while both methods provide valuable heuristics for improving model performance, the JMI patch method may be more effective for certain tasks or models, specifically for ViT models.

Both the JMI patch method for ViT and the spiral method for CNN highlight the importance of intelligently ordering genes when building predictive models. Both methods demonstrated their potential to enhance model performance across a variety of tasks, confirming their utility as heuristics for tabular dataset analysis. These findings suggest that the thoughtful arrangement of genes is a critical factor in maximising the predictive performance of machine learning models. Furthermore, the generalisability of both methods to any tabular dataset opens avenues for their application in a wide range of contexts and tasks. Future research may focus on further refining these methods and exploring their effectiveness in other models and tasks.

3.4.3 Jointly Dependent Gene Interactions and Gene Ontology Analysis

Three sets of genes of high relevance were extracted from an initial pool of 3,000 genes, selected based on mutual information and ANOVA's F statistic. These subsets contained 121, 576, and 2,601 genes respectively. These genes were subsequently transformed into three 2D gene matrices of dimensions 11x11, 24x24, and 51x51. To ensure the robustness of the results obtained from Grad-CAM++ and permutation importance, all genes within the three matrix sets were randomly shuffled 1,000 times, utilising the **random.shuffle** function from the NumPy library.

The Grad-CAM technique was deployed to pinpoint multivariate gene patterns correlating with virus abundance. The intention was to identify groups of A. halleri genes collectively contributing to TuMV abundance. The top 30% of genes identified through Grad-CAM were selected for gene network analysis and Gene Ontology (GO) analysis. Included within this selection were key genes, such as Argonaute proteins 1 and 2 (AGO1, AGO2), which were accorded high importance scores. The chosen genes were then subjected to enrichment analysis, as a concentrated set of high-scoring genes is necessary to derive statistically significant results.

For individual genes linked to virus abundance, permutation importance was the method of choice. A positive permutation importance score indicates a positive contribution of a given gene to the model's performance. Genes identified via permutation importance were also included in the GO analysis, to explain individual gene-pathogen relationships. This analysis allows for the identification of important genes that could serve as biomarkers, reflecting the dynamic interplay between host and virus.

Upon applying GO analysis to the top 25 genes deemed important by the CNN model, two GOs emerged as enriched: the ribonucleoprotein complex (GO:0030529) and siRNA binding (GO:003519). No GO enrichment was detected for other learning models. Four genes from the top 25 were assigned to these enriched GOs. Two of these, Ahg473977 and Ahg473645, encode homologues of AGO1 and

AGO2, respectively, which are known to play significant roles in antiviral RNA silencing, a virus-specific defence mechanism in plants. In A. thaliana, AGO2 is particularly critical for antiviral RNA silencing in leaves against TuMV infection. Previous research has shown that upregulated genes involved in RNA silencing, including AGO1 and AGO2, corresponded with elevated within-host virus accumulation.

An expanded GO analysis was applied to the top 48 genes identified by the CNN model, leading to the enrichment of one GO, the ribonucleoprotein complex (GO:0030529). The other two genes, Ahg322062 and Ahg915450, encode homologs of La-related proteins (LARPs) in A. thaliana, specifically, LARP6B and LARP6A, and were included in this GO. LARPs have a crucial role in RNA metabolism, and LARP6B, in particular, possesses a PABP-interacting motif 2 (PAM2) that facilitates its targeting to mRNA 3’ UTRs. Despite the functions of LARPs being largely unknown in plants, the fact that TuMV is characterised by a poly(A) tail akin to mRNA leads to speculation about possible, yet unidentified, interactions between these proteins and TuMV. Earlier research suggests that A. thaliana class II poly(A)-binding proteins are required for efficient TuMV multiplication.

3.5 Discussion

3.5.1 Supervised Learning Methods for Gene Expression

The application of supervised machine learning methods to gene expression datasets has emerged as a promising approach in bioinformatics and computational biology, enabling researchers to extract valuable insights from large and complex biological datasets. In this discussion section, I will critically evaluate the application of supervised machine learning methods to gene expression datasets, highlighting their strengths, limitations, and potential implications for biological research.

One of the major strengths of supervised machine learning methods in gene expression analysis is their ability to handle high-dimensional data. Gene expression datasets typically consist of thousands of genes with expression values measured across multiple conditions or time points. Supervised machine learning methods, such as support vector machines, decision trees, and random forests, are capable of handling such high-dimensional data by automatically identifying relevant features or genes that are important for classification or prediction tasks. This allows researchers to identify genes or gene sets that are differentially expressed between different conditions or groups, which can provide valuable insights into biological processes and mechanisms.

Another advantage of supervised machine learning methods in gene expression analysis is their ability to model complex non-linear relationships between genes and conditions. Gene expression data often exhibit non-linear patterns due to the complex regulatory mechanisms involved in gene expression. Traditional statistical methods may struggle to capture these non-linear relationships, whereas supervised machine learning methods, such as kernel methods or neural networks, can model these complex interactions and provide more accurate predictions. This enables researchers to uncover hidden patterns or interactions among genes that may not be apparent using traditional statistical methods.

Furthermore, supervised machine learning methods in gene expression analysis are highly adaptable and can be applied to a wide range of biological questions and experimental designs. They can be used for diverse tasks such as classification of different disease subtypes, prediction of patient outcomes, identification of biomarkers, and drug discovery. Moreover, supervised machine learning methods can be applied to different types of gene expression data, including microarray data, RNA-seq data, and single-cell RNA-Seq data. This flexibility makes supervised machine learning methods a versatile tool for analysing gene expression datasets across various biological contexts.

However, there are several limitations and challenges associated with the application of supervised machine learning methods to gene expression datasets. One key challenge is the issue of overfitting, where the model may learn to perform well on the training data but fails to generalise to new, unseen data. Gene expression datasets are often noisy and subject to various technical and biological confounders, which can impact the performance and interpretability of the models. Proper pre-processing, feature selection, and model validation techniques, such as cross-validation, should be carefully applied to mitigate the risk of overfitting and ensure the robustness of the results.

Another limitation of supervised machine learning methods in gene expression analysis is the need for large sample sizes. Supervised machine learning methods typically require a large number of samples to effectively learn the underlying patterns in the data. However, gene expression datasets with large sample sizes are not always available, especially in certain rare diseases or experimental conditions. This can limit the applicability and generalisability of supervised machine learning methods in some scenarios, and alternative approaches such as transfer learning or data augmentation techniques may need to be considered.

Moreover, the interpretability and biological interpretability of supervised machine learning models in gene expression analysis can be challenging. Many supervised machine learning methods, such as deep neural networks, are often considered as "black box" models, where the relationship between input features and output predictions may not be easily interpretable. This can hinder the ability of researchers

to gain meaningful biological insights or generate hypotheses based on the model predictions. Therefore, developing interpretable and biologically meaningful models that provide insights into the underlying biology remains an important challenge in the field.

Despite these challenges, the application of supervised machine learning methods to gene expression datasets has the potential to significantly impact biological research.

3.5.2 Comparison with Differential Expression Analysis

Genes that are highly relevant to viral infection can be located either individually or collectively through Trans-Learn. I have utilised a reliable computational solution to reverse engineer (i.e. Grad-CAM++) the trained neural network model to identify genes according to their contribution to the models as well as their multivariate properties linking to virus abundance levels. For example, if the plant is experiencing multiple stresses, the potential combination of up- and down-regulation for a specific subset of genes could lead to false positives or negatives. Since the samples were collected from plants in a natural environment, the potential for additional and unrelated factors that could influence gene expression can affect the results. Although these external factors did not affect the prediction accuracy of Trans-Learn (90%+) in this study, it is worth pointing out that incorrect predictions could be treated as given samples are be more likely to be affected by external factors, as the gene expression mapping does not generalise to the incorrect classification.

Additionally, the incorporation of GO analysis in the interpretation could add biological relevance to the data-driven analysis approach embedded in Trans-Learn. GO weights all genes included in a specific GO term equally, based on which enrichment is tested for and the number of genes that are selected. By doing so, GO analysis together with the concluded terms for the clusters of genes will be able to guide users with the selection of candidate genes more consistently. The understanding of a joint and collective biomarker network can be expanded by the exploitation of Trans-Learn’s results, which could lead to the allocation of these

biomarkers to a regulatory network, as well as to understand how different modules in existing regulatory networks are represented.

Trans-Learn exhibits a distinctive advantage over traditional differential expression or univariate methods, especially in scenarios involving complex biological systems. Traditional methods often employ an approach where genes are independently assessed, considering each gene in isolation. This limits the ability of such methods to discern complex relationships or dependencies among genes, which is a hallmark of biological systems. In contrast, Trans-Learn recognises and accommodates the interdependence among genes, enabling the identification of clusters of jointly dependent genes. The approach is more attuned to the reality of biological systems, where gene interactions often play crucial roles in influencing phenotypic outcomes.

Another fundamental limitation of univariate methods is their assumption of linearity in relationships between genes and the target variable. This simplification often fails to capture the complexity and non-linearity inherent in biological systems. In contrast, Trans-Learn, particularly with its incorporation of CNN and VIT methods, can model and understand these non-linear relationships. The multivariate property of the gene importance metric, coupled with the CNN's ability to capture complex, non-linear interdependencies, renders Trans-Learn an exceptionally powerful tool for biomarker identification. Furthermore, differential expression analysis tends to be reactive rather than predictive, identifying changes in gene expression after the fact. It may also struggle with false positives due to multiple hypothesis testing. Trans-Learn, on the other hand, is inherently predictive, trained to recognise patterns and make accurate predictions on unseen data. This makes it a potentially valuable tool for early detection and prevention efforts in a variety of applications.

In particular, the multivariate property of my proposed gene importance metric has enabled me to identify genes that are dependent on a given target virus with varying levels of significance, which also have certain degrees of dependence between themselves. The primary methodical advantage is that it can reflect the model's non-linear multivariate relationships between gene and the target, because the gradients after the convolutions with neighbouring genes in the CNN-based model can be included in the calculation. This approach has enabled the detection of clusters of

jointly dependent genes. In this case, the models show an increase in predictive power in comparison to generalised linear models when tested on the same data. Furthermore, automatic feature extraction in convolutional and non-linear activation layers in the CNN-based model can create feature maps that contain gene-virus patterns, which cannot be revealed in traditional hypothesis testing or linear methods. From this perspective, my method is capable of resolving both classification and regression problems in order to identify multivariate biomarkers in molecular biology, whereas traditional methods are typically defined for studying the univariate relationships in a controlled against treated environment.

3.5.3 Trans-Learn's Challenges

Whilst Trans-Learn provides a powerful platform for predicting host-virus interactions, and advancing our understanding of underlying gene dependencies, several challenges and limitations remain to be addressed.

Firstly, the adaptive threshold utilised for discerning important genes is mathematically robust but may be biologically arbitrary. Its basis in joint dependency scores may not universally align with all hypotheses in life sciences, requiring careful adaptation for differing biological contexts.

Secondly, the predetermined ordering of genes through the spiral gene arrangement before input into the CNN-based model may unintentionally influence Grad-CAM++ results. This factor may distort interpretations by skewing results towards the arrangement of genes rather than the gradient-based method intrinsic to CNNs. A further challenge lies in the inherent trade-off between matrix size and computational efficiency in the implementation of Grad-CAM++. As the input matrix expands, Grad-CAM++ must be applied more frequently with varying gene arrangements to assure a diverse range of gene neighbours are sampled from the gene pool. This escalating demand introduces substantial computational complexity and time costs.

Moreover, a potential avenue for future development is the integration of the two selection methods, Grad-CAM++ and permutation importance, to coherently signify

the dynamic relationships between host and virus. Currently, this joint approach remains unexplored.

On a broader scale, the translatability of feature encoding and selection methodologies to diverse gene expression datasets or plant-virus interactions should be carefully evaluated. The performance of these techniques may vary with the specific characteristics of the datasets, such as the type of virus, host plant species, and experimental conditions. Therefore, ongoing validation and optimisation of these methods across a range of datasets and interactions are essential to gauge their generalisability and reliability.

The challenges extend beyond the methodological scope, encompassing aspects of software adoption and maintenance. Therefore, I have produced a comprehensive instructions for use file that is available for users to read through before using the software. As an open-source Python software, the success of its dissemination and application relies on factors like ease of use, comprehensive documentation, and readily available support. Ensuring that the software remains accessible, well-documented, and user-friendly is pivotal to its broad adoption, and fosters an environment for ongoing development and improvement.

Lastly, maintaining software sustainability necessitates regular updates to address potential bugs, compatibility issues with new software libraries, and to meet evolving research requirements. If adopted by the community, this continual upkeep represents a significant challenge in resource allocation, time commitment, and fostering an engaged user community, all critical for the long-term viability and utility of the software.

3.5.4 Interpretation of Trans-Learn’s Results

The gene dependency matrix produced during the Trans-Learn pipeline is founded upon a weighted combination of the positive partial derivatives of the CNN-based model’s final convolutional layer. This is with respect to the categorisation of the various levels of viral abundance. It is important to recognise that these gene

dependency scores should not be construed as a statistic for testing the impacts of viral infections. Instead, they should be interpreted as suggestive indicators, inviting more comprehensive exploration between certain subsets of genes and the targeted virus. Despite a dataset of over 500 samples being adequate to evaluate the performance of my method, an anticipation for a greater predictive power of the neural networks is held. This expectation is justified given their demonstrated proficiency with considerably larger datasets.

In terms of the generated network's interpretation, a specific focus should be given to those genes interconnected by thicker edges. The substantial evidence suggests that CNNs and ViTs can effectively utilise patterns within gene expression matrices (the input) and correlate them with the output, for example, TuMV severity. This correlation can either be unlinked or associated with different components of the variance. My findings indicate that genes with a higher connectivity in the dependency network are able to pinpoint more neighbouring genes. This occurs when these genes are convolved and undergo non-linear transformations collectively via the neural network. As such, highly connected genes appear to possess distinctive patterns not found in the genes they are linked with. Conversely, genes with fewer connections can be considered to contain a pattern that becomes effective only when coupled with a small subset of genes. The potency of this effectiveness is mirrored in the network by the size of the node representing that specific gene.

3.5.5 Applications of the Trans-Learn Software

The Trans-Learn software has wide-ranging potential applications across various fields in molecular biology and genomics, which extend beyond its initial usage for identifying multivariate biomarkers or training predictive models. As a tool, Trans-Learn presents a novel and cutting-edge approach that combines machine learning techniques with biological insights, providing a powerful platform to address complex biological questions.

Firstly, Trans-Learn could play a critical role in the field of diagnostic medicine. It can be used to train predictive models on datasets derived from human gene

expression studies related to specific diseases. These models can then assist clinicians in identifying a panel of genes that collectively contribute to the diagnosis of a particular disease, leading to more accurate and efficient diagnoses. Given the multivariate nature of many diseases, the application of Trans-Learn could assist diagnostic practices by considering the combined effects of multiple gene expressions, rather than focusing solely on the expression of individual genes. This could lead to the development of comprehensive diagnostic panels that account for the complexity of disease processes.

Additionally, the capabilities of Trans-Learn could be utilised for trait prediction in plant and animal breeding programmes. By training predictive models on datasets generated from the genetic profiling of various phenotypes, breeders could identify sets of genes that contribute to the expression of desirable traits. Consequently, this can enable breeders to make informed selection decisions and effectively enhance breeding strategies. Furthermore, the multivariate approach of Trans-Learn could allow breeders to better understand the complex genetic networks that underpin phenotypic traits, which is often a challenge in traditional breeding programmes.

In the field of plant genomics, Trans-Learn could be used to identify multivariate biomarkers for various agronomically important traits. By training predictive models on datasets generated from high-throughput phenotyping and genotyping studies, agricultural scientists could gain insights into the complex gene interactions underlying traits like yield, stress resistance, and nutrient use efficiency. This could help in the development of precision breeding strategies to improve crop performance under varying environmental conditions.

While these potential applications highlight the promise of Trans-Learn, it is important to note that each would require careful validation and optimisation of the platform for the specific context. Nevertheless, the capabilities of Trans-Learn provide a promising avenue for harnessing the power of machine learning in various areas of biological research and practice.

3.5.6 Conclusion

In this chapter, I introduced the Trans-Learn platform, a pioneering approach that I developed with the initial aim of identifying multivariate biomarkers linked to TuMV infection in *Arabidopsis halleri*. By harnessing machine learning and deep learning methodologies, I succeeded in transforming tabular transcriptomic datasets into an image-like format conducive to the deployment of techniques such as CNNs and ViTs.

A key accomplishment was the creation of effective feature encoding methods that allowed me to emphasise patterns and relationships between genes, thereby providing my models with the critical spatial awareness they required. It is worth noting that the efficacy and applicability of these methods to a wide range of gene expression and tabular datasets make them an invaluable tool for the research community.

As part of my objectives, I developed feature interpretation techniques to identify relevant biomarkers linked to TuMV infection in *A. halleri*, and anticipate their applicability to other datasets. Utilising Grad-CAM, I successfully extracted gene importance and created a metric to quantify the multivariate nature of the dependencies between biomarkers.

A focal objective involved utilising both standard and advanced machine learning techniques like gradient boosting machines, logistic regression, and deep learning techniques such as CNNs and ViTs. By implementing these on the tabular and encoded image data, I endeavoured to predict TuMV infection and other targets within gene expression datasets. I compared the performance of various models across multiple datasets, revealing the most effective methods for prediction utilising gene expression data to be my specific patch arrangement ViT model as well as the stack and blend ensembles.

Following the identification of associated genes or biomarkers, I utilised gene ontology and network analysis methodologies to gain biological insights, fulfilling

another objective. Tools such as Cytoscape facilitated this process, allowing me to infer gene regulatory relationships. However, my limited biological background limited my interpretation of the biomarkers and so my focus remained on designing the software to maximise predictive performance.

I also developed open-source Python software to make the Trans-Learn platform accessible and reproducible for the wider scientific community. This software includes modules for encoding tabular datasets into image format, implementing feature selection techniques, and training supervised machine learning models for biomarker identification.

Overall, I established a robust pipeline rooted in computer vision and machine learning techniques that demonstrates significant potential in early detection and management of TuMV infection in *Arabidopsis halleri*. While this work primarily focused on this specific case, the methods and tools I've developed hold promise for broader applications, including analysing gene expression dynamics in other biological contexts and predicting different traits.

However, there were key challenges facing the Trans-Learn project, including biological interpretation of identified genes, integration of multi-omics data and questionable performance on smaller datasets. These challenges present exciting opportunities for further refining and expanding the capabilities of Trans-Learn. As I continue to make strides in these areas, I am confident that Trans-Learn will stand as a testament to my hypothesis - that a computational pipeline, which melds computer vision and machine learning for encoding, selecting, and interpreting features from tabular gene expression datasets, can indeed surface significant multivariate patterns and relationships within the data. I believe Trans-Learn can serve as a powerful tool for predictive purposes and for deepening our understanding of gene expression dynamics in diverse biological contexts.

3.6 Future Research

There are several promising directions for future research to further develop machine learning methods for the analysis of gene expression data in crops and the diagnosis of plant diseases through biomarkers. One potential avenue is the integration of multi-omics data, encompassing transcriptomics, proteomics, and metabolomics, to capture a more comprehensive and dynamic view of gene expression patterns and molecular interactions in plants. The amalgamation of multiple omics datasets could offer a more holistic understanding of the complex biological processes underpinning crucial traits in crops, such as yield, stress tolerance, and disease resistance. Machine learning methods such as Trans-Learn's pipeline that can effectively incorporate and analyse multi-omics data could facilitate the identification of novel biomarkers and predictive models with enhanced accuracy and robustness.

Future research could concentrate on developing interpretable machine learning methods for gene expression analysis in crops. Interpretability is a critical aspect of practical machine learning applications, allowing researchers and practitioners to understand and validate the underlying mechanisms and biological relevance of predictive models. The development of machine learning methods that offer interpretable insights into regulatory networks, functional annotations, and biological pathways associated with gene expression patterns in crops can expedite the discovery of meaningful biomarkers and enhance the utility of predictive models for crop improvement and disease diagnosis.

In addition, there is a requirement for further validation and optimisation of machine learning methods across diverse crops, plant species, and environmental conditions. Different crops may exhibit unique gene expression patterns, regulatory networks, and molecular interactions, which can impact the performance and generalisability of machine learning models. Carrying out extensive validation studies in various crops and plant species, as well as under different environmental conditions, can help to assess the robustness, reliability, and applicability of machine learning methods for gene expression analysis in varied agricultural contexts.

Finally, future research could focus on the development of user-friendly and accessible software tools for applying machine learning methods to gene expression data in crops. Software tools with intuitive interfaces, detailed documentation, and extensive support can empower researchers and practitioners in the field of plant pathology to easily apply and customise machine learning methods to their specific research questions and experimental data. Encouraging open-source software development, community engagement, and collaboration among researchers can facilitate the dissemination and adoption of machine learning methods in the field of crop research and plant disease diagnosis.

In conclusion, there are exciting opportunities for future research in developing machine learning methods for analysing gene expression data in crops and diagnosing diseases in plants using biomarkers. The integration of multi-omics data, the utilisation of deep learning techniques, the development of interpretable models, validation in diverse crops and environmental conditions, and the development of user-friendly software tools are important areas of focus for advancing the field of gene expression analysis in crops and plant pathology research. Continued research and innovation in these areas can contribute to the development of more accurate, robust, and interpretable predictive models for crop improvement and disease diagnosis, ultimately benefiting agriculture and food security.

Chapter 4: ChronoGauge – An Open-Source Machine Learning Method for Circadian Gene Expression Analysis and Time Prediction

4.1 Introduction

In this chapter, I introduce ChronoGauge, a machine learning-based method designed to identify biomarkers of the circadian clock and predict an organism's internal circadian time. This method and associated software, now employed by Mr Connor Reynolds for his PhD at the Earlham Institute and utilised during his Alan Turing Fellowship position, were independently developed by myself. My colleagues at the Earlham Institute took on the responsibility of processing the RNA-Seq datasets.

My contribution to this project was extensive and comprised the development of the entire pipeline and software. This included the selection of rhythmic genes, the coding of rhythmic metrics, the implementation of custom sequential feature selection, model selection, and the formulation of a circular linear regression model. Small parts of this chapter have been adapted and modified with permission from my publication in PNAS: "Interpreting machine learning models to investigate circadian regulation and facilitate exploration of clock function".

To provide the necessary context for this chapter, I will initially provide a concentrated literature review on the circadian clock. This review will focus on methods used to detect rhythmically expressed gene expression markers, as well as existing strategies for predicting an organism's internal time using gene expression.

4.1.1 Abstract

In this chapter, I introduce ChronoGauge, an innovative open-source machine learning method I developed to predict the internal circadian time of organisms. This

is achieved by using transcriptional biomarkers derived from a single transcriptomic time point. I applied this method to both wheat and Arabidopsis datasets and prioritised the selection of biomarkers across circadian phases to enhance accuracy and robustness. The technique involves an innovative use of custom sequential feature selection and a circular linear regression model, implemented via stochastic gradient descent, resulting in highly accurate predictions on test datasets.

I evaluated the predictive capabilities of ChronoGauge on clock mutants and datasets with various environmental conditions, achieving an impressive mean absolute error (MAE) of 46 minutes in a test dataset. Interestingly, my best-performing model, comprised of 15 carefully selected transcripts, surpassed the performance of the state-of-the-art ZeitZeiger method, which achieved an MAE of 143 minutes. This underscores my hypothesis that a more evenly distributed selection of biomarker transcripts across rhythmic phases would yield a more robust and generalisable mapping from expression data to internal circadian time.

The final subset of 15 transcripts, which interestingly included no core clock genes, effectively served as a small, yet sufficient group of biomarker transcripts for accurately predicting circadian time. I observed no discernible relationship between circadian time and prediction error, barring slight discrepancies in the training dataset. In essence, the innovative application of ChronoGauge to gene expression data not only provides a valuable tool for accurately predicting circadian time but also lays the groundwork for further advancements in the field.

4.1.2 Crops Synchronise Processes with Environment

The circadian clock is a fundamental timekeeping mechanism that regulates the biological rhythms of various organisms, including plants. It is an endogenous, self-sustained timing system that enables plants to anticipate and adapt to daily environmental changes, such as light-dark cycles and temperature fluctuations. The circadian clock in plants is critical for coordinating various physiological and molecular processes, including growth, photosynthesis, stomatal conductance,

hormone signalling, and stress responses, which ultimately influence plant performance and fitness.

The circadian clock is a critical trait in agricultural crops as it plays a pivotal role in regulating various aspects of plant growth, development, and stress responses, which ultimately impact crop productivity and yield. The circadian clock allows plants to anticipate and adapt to daily environmental changes, such as light-dark cycles, temperature fluctuations, and water availability, thus optimising their physiological and metabolic processes.

One key aspect of the circadian clock in agricultural crops is its regulation of plant growth and development. The circadian clock controls the timing of key developmental processes, such as flowering, stem elongation, leaf expansion, and root growth, which are crucial for crop yield and quality. For instance, proper timing of flowering is essential for reproductive success and seed production in many crop species, influencing crop yield and seed quality. The circadian clock also regulates stem elongation and leaf expansion, affecting plant architecture and canopy development, which impact light capture, nutrient uptake, and overall plant growth. Furthermore, the circadian clock controls root growth and nutrient uptake, influencing nutrient acquisition efficiency and water use efficiency, which are important for crop performance under varying environmental conditions.

In addition to growth and development, the circadian clock also regulates plant responses to abiotic and biotic stresses, which are major challenges in agriculture. The circadian clock modulates plant responses to various stresses, such as drought, salinity, heat, cold, and pathogens, by regulating the expression of stress-related genes and the synthesis of stress-related metabolites. Proper circadian clock regulation is crucial for optimising stress responses and enhancing crop resilience to adverse environmental conditions. For example, the circadian clock regulates stomatal opening and closing, influencing plant water use efficiency and drought tolerance. The circadian clock also regulates the expression of defence-related genes, such as those involved in the synthesis of secondary metabolites and defence signalling pathways, which play a crucial role in plant defence against pathogens and pests.

Furthermore, the circadian clock also impacts crop quality traits, such as nutritional content, flavour, aroma, and shelf life, which are important for marketability and consumer preference. The circadian regulation of metabolic processes, such as photosynthesis, starch metabolism, and secondary metabolite synthesis, influences the accumulation of key nutritional and quality-related compounds in crops. Proper circadian clock regulation is essential for optimising the synthesis and accumulation of these compounds, which impact crop nutritional value, taste, aroma, and overall quality. However, our understanding of such complex transcriptional regulatory systems is limited by our ability to assay them, requiring the generation of long high-resolution time-series datasets.

In conclusion, the circadian clock is an important trait in agricultural crops as it regulates plant growth, development, stress responses, and quality traits, which collectively impact crop productivity, yield, and marketability. Understanding the role of the circadian clock in crop biology has important implications for crop improvement, stress resilience, and sustainable agriculture, and can aid in the development of crop management strategies that optimise plant performance in changing environmental conditions.

4.1.3 Methods to Identify Circadian Regulated Genes

Circadian rhythms are endogenous biological rhythms that follow a roughly 24-hour cycle and regulate various physiological processes in living organisms, including gene expression. Identifying circadian-regulated genes is an important area of research in the field of chronobiology, as it provides insights into the molecular mechanisms underlying circadian clock regulation. Several statistical methods have been developed and utilised to identify circadian-regulated genes from high-throughput transcriptomic data. Here, I will discuss some of the commonly used statistical methods for identifying circadian-regulated genes.

Time series analysis serves as a robust approach for detecting circadian-regulated genes from transcriptomic data. This technique involves studying and analysing gene

expression profiles over several time points, typically spanning at least one 24-hour cycle. As circadian rhythms follow a cyclic pattern, this analysis enables us to capture the inherent periodicity of the data and analyse the rhythmic oscillations of gene expression.

One of the methods employed for time series analysis is Fourier analysis, which is particularly effective in handling periodic data such as circadian rhythms. Fourier analysis works by decomposing a time series into a set of sinusoidal components with different frequencies, which can be used to identify rhythmic patterns in the gene expression data. The outcome is a spectrum that displays the strength of these rhythmic components at different frequencies, thus revealing the periodic nature of the gene expression and the primary circadian rhythms present within the biological system.

Developed by Hughes et al. (2010), JTK_CYCLE is a widely used, nonparametric algorithm designed to detect and characterise rhythms in large-scale time-series data. The method, based on the Jonckheere-Terpstra-Kendall (JTK) test, is known for its robustness and reliable results, boasting superior sensitivity and specificity compared to alternative methods. JTK-Cycle utilises a combined approach of the Jonckheere-Terpstra test for detecting orderings in value groups and Kendall's tau as a rank correlation measure to quantify the dependence between two quantities. This method fits a cosine curve to temporal expression data, calculating the phase and period-lengths that minimise Kendall's tau p-value for each transcript, thereby quantifying their rhythmicity. Despite its resistance to outliers and computational efficiency, JTK-Cycle might produce false negatives due to the potential difficulty of calculating parameters in low-resolution datasets. In application, JTK-Cycle has unveiled a novel cluster of circadian-regulated RNA-interacting genes in previously published datasets.

ARSER (Autoregressive spectral estimation rhythm), devised by Yang and Su (2010), stands as an effective tool for detecting and characterising circadian rhythms in extensive time-series gene expression data. By applying spectral analysis to an autoregressive model that describes time-series data, ARSER successfully handles common challenges in circadian rhythm research such as noise, short time-series,

and irregular sampling. The process starts with linearly detrending the time series, followed by smoothing and mapping into the autoregressive spectrum to identify genes with peaks in the 20-28 hour range, with the peaks serving as period estimates. Comparable to JTK-Cycle, ARSER estimates signal phase, period, amplitude, and mean level, using these parameters to fit the harmonic regression model. An F-test then quantifies the rhythmicity of the transcript or gene by testing for significance in the covariates.

Autocorrelation and cross-correlation are traditional statistical methods that have been applied to the analysis of rhythmically expressed genes. Autocorrelation measures the similarity between observations as a function of time lag, enabling the identification of repeating patterns in time-series data, such as rhythmic gene expression. Cross-correlation, on the other hand, measures the similarity between two time-series as a function of the time lag applied to one of them, which can be used to assess the phase relationship between two rhythmic processes.

MetaCycle, developed by Wu et al. (2016), is an integrated interface that provides several algorithms for detecting rhythmic signals from time-series datasets. It includes both ARSER and JTK_CYCLE, along with the Lomb-Scargle method. MetaCycle has been designed to handle both evenly and unevenly sampled time-series data, offering great flexibility for the analysis of rhythmic gene expression. The component of MetaCycle that should be used is dependent on the dataset e.g. number of timepoints, replicates, missing values etc. as no single method is best, however as MetaCycle works as an ensemble of these methods, the authors advise that their method can be used on a variety of datasets.

A popular metric for quantifying rhythmicity not used in MetaCycle is the relative amplitude error (RAE) [388] that utilises FFT-NLLS (fast Fourier transform – non-linear least squares) to fit a cosine curve to the signal and the goodness of fit estimates the rhythmicity of the signal. Similar to ARSER, a fast Fourier transformation is applied to the signal and properties of the transformed signal are used as initial estimates of the cosine curve's phase, period and amplitude parameters. BioDare, an online system for the sharing, processing, and analysis of circadian datasets has integrated the FFT-NLLS method for the purpose of

identifying rhythmic genes[389]. An advantage of calculating the RAE is that signals that dampen or change in amplitude will be punished as the cosine fit will result in a larger error.

In conclusion, identifying circadian-regulated genes from transcriptomic data requires the use of statistical methods that can model rhythmic patterns in gene expression. These methods can provide valuable insights into the molecular mechanisms of circadian regulation and its impact on gene expression in living organisms.

4.1.4 Predicting the Circadian Time Using Biomarkers

Predicting the circadian time in plants, which refers to estimating the time of day based on gene expression patterns, is an active area of research in plant chronobiology. Several algorithms and tools have been developed specifically to predict circadian time using high throughput transcriptomic data. Some notable examples include Molecular Timetable[390], ZeitZeiger[391], and TimeSignatR[392].

The Molecular Timetable method, first described by Ueda et al., utilises gene expression data from a single time point to estimate the circadian time in an organism. The premise of this method is that different genes are expressed at different times throughout the circadian cycle, and so the expression of a given set of genes provides a "timetable" or reference to predict the circadian time. However, this method is often limited by the requirement for large numbers of rhythmically expressed genes to accurately predict the circadian time, which may not always be feasible or accurate in some biological contexts.

ZeitZeiger, proposed by Hughey et al., is a robust method for predicting circadian time using high-dimensional regression. It employs principal component analysis (PCA) to reduce the complexity of gene expression data, followed by sparse regression to identify a select few genes most indicative of circadian time. This method proves efficient even with noisy data or samples collected at less than optimal times, although its performance may diminish when there are limited

rhythmically expressed genes available. ZeitZeiger's strength lies in its ability to generate a sparse representation of variation in gene expression in relation to circadian time through sparse PCA, opting for maximum likelihood over standard supervised machine learning for faster, more accurate results. Importantly, its application extends to predicting circadian time using a single time point from a set of marker genes in human blood, enabling detection of changes in clock function due to disease or environmental conditions.

TimeSignatR, developed by Troup et al., is a machine learning-based tool that was designed to generate "time signatures" to predict the time of day from gene expression data in humans. The algorithm was trained on a dataset that includes thousands of samples from over 50 tissues, collected at different times of the day. TimeSignatR stands out due to its ability to predict the time of day from transcriptomic data, not just from blood but from multiple tissues. This method's success underscores the value of machine learning methods in circadian time prediction.

All these methods highlight the importance of time-point specific gene expression in predicting circadian time and have made significant contributions to the field. However, it's crucial to note that these methods need to be adapted and validated for different organisms and various environmental conditions to ensure their broad applicability in circadian research.

In conclusion, predicting circadian time in plants from gene expression data requires the use of statistical and machine learning methods that can model the periodic patterns in gene expression.

4.2 Aims and Objectives

The aim of the ChronoGauge project was to create a predictive model for circadian time in plants, leveraging both machine learning and statistical methods on gene expression data. In particular, my project objectives were:

1. My main objective was to create an efficient, user-friendly, and comprehensive Python software tool for circadian time prediction in plants. This tool would need to accommodate high-throughput gene expression data from various plant species and datasets, and I planned to make it freely available as an open-source resource for the scientific community.
2. My second objective was identifying predictive circadian biomarkers. These are genes or gene sets that consistently demonstrate rhythmic expression patterns across different datasets and plant species. By integrating and analysing multiple gene expression datasets from different plant species with known sampling time labels, I aimed to identify potential circadian biomarkers using feature selection techniques like sequential feature selection and ARSER, JTKCycle, and MetaCycle. This would help highlight the most informative genes for circadian time prediction. My hypothesis was that having a diverse set of biomarkers with different peak phases would perform better than having a feature set comprised of many biomarkers with the same phase.
3. Next, I planned to develop and optimise machine learning and statistical models for circadian time prediction using these identified circadian biomarkers. I intended to use various methods like neural networks as well as circular regression. I intended to evaluate their performance using cross-validation and metrics like mean absolute error (minutes) to identify the best-performing models for circadian time prediction.
4. I also aimed to test the developed predictive circadian time models across different datasets, involving various plant species and experimental conditions. This involved applying the trained models to independent datasets with known circadian time information and assessing their performance in terms of prediction accuracy, robustness, and generalisability. This would help understand the applicability and transferability of the

predictive circadian time models across different biological contexts and datasets.

Overall, this project aimed to develop a robust and accurate predictive model for circadian time in plants by applying machine learning and statistical methods to gene expression data. The project also sought to identify predictive circadian biomarkers and assess the performance of the developed model(s) across different datasets, contributing to the advancement of our understanding of plant circadian clocks and providing a valuable tool for circadian research in plants. The development of an open-source Python software tool also served the scientific community by offering a freely accessible resource for circadian time prediction in plants.

4.3 Methods

4.3.1 RNA-Seq Circadian Datasets

Initially, three transcriptomic *Arabidopsis thaliana* datasets were chosen to develop a method for predicting the endogenous circadian time and identifying a subset of marker genes (Table 8). These datasets came from published studies that reveal differences in transcriptome regulation [393–395]. The datasets were all generated by growing *Arabidopsis thaliana* seedlings in light-dark conditions (12 hourrs, 12 hours) and then put in constant light to trigger circadian regulation without light receptive genes being mistaken as false positives.

Table 8. Information relating to the three circadian experiment datasets used for most of the ChronoGauge project.

Source	**Ecotype**	**Material**	**Biological Replicates**	**Number of timepoints**	**Sampling frequency (hours)**
Romanowski et al. (2020)	Col-0	Areal	2	12	4
Yang et al. (2020)	Col-0	Seedlings	2	8	3
Graf et al. (2017)	Col-0 and WS	Whole rosettes	2-3	2	12

During later stages of the ChronoGauge project, my colleague Mr Connor Reynolds processed additional circadian RNA-Seq datasets from studies by Locke et al., Yavonsky et al., Mas et al., and Blair et al.. The compiled information relating to the combined seven publicly available datasets is presented in Figure 25.

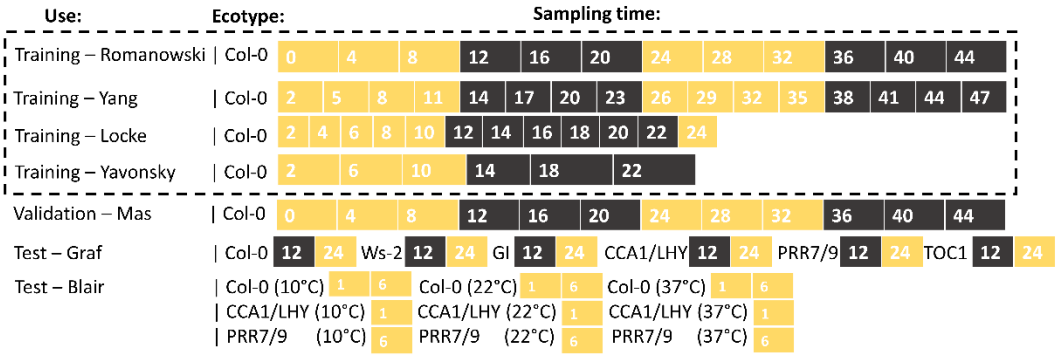

Figure 25.

All of the circadian experiment gene expression datasets used in this study. All datasets were labelled with the label being the sampling time which acts as a proxy label for the internal time of the plant. The number in each rectangle indicates the sampling time.

4.3.2 Method of Validation

In order to assess and improve the performance of my models, I employed a technique called time series cross validation[124]. Time series cross validation is a model validation technique for assessing how the results of a statistical analysis will generalise to an independent data set. Unlike traditional cross-validation methods, time series cross validation respects chronological ordering of data, and "future" data is never used to predict "past" data. This is particularly important for the time-dependent nature of circadian datasets.

I initially assigned the Romanowski dataset as the training dataset due to its strong rhythms and numerous time points, while the Yang and Graf datasets were designated as the validation and test datasets, respectively. The rationale behind this validation scheme was to establish a robust expression-to-time mapping using the training dataset, then refine the optimal hyperparameters that would minimise the error on the Yang dataset. After identifying the optimal set of hyperparameters and rhythmic genes, I proceeded to make predictions on the Graf dataset to obtain a final error value, which would serve as an estimation of the model's error on unseen data. The integration of time series cross validation for optimising the hyperparameters of my models eventually involved combining the Yang and Romanowski datasets.

Once additional datasets were processed, I incorporated the Romanowski, Yang, Locke, and Yavonsky datasets into the training process. This allowed me to have extra samples for time series cross validation, thereby increasing the robustness and performance of my predictive models.

4.3.3 Expression Matrix Pre-processing

The initial stage of my feature selection approach involved the exclusion of genes with a variance less than 5. Genes falling under this category are typically either lowly expressed, thus of minimal interest, or lack rhythmic variance that is dependent on endogenous time.

After removing the low variance genes, I standardised the expression data, a vital step to ensure the comparability and reliability of the results. This was accomplished using the standard scaling method which involved subtracting the mean expression of each gene from the expression matrix, then dividing by each gene's standard deviation. Mathematically, this is represented as:

$$X_{scaled} = \frac{X - \mu}{\sigma}$$

This scaling method effectively centres each gene's expression at zero, eliminating linear trends that could cause variations in amplitudes across time points.

Due to having multiple datasets from different sources involved in the ChronoGauge project, my colleague Connor Reynolds and I implemented a method known as ComBat[396] to minimise batch effects among the datasets. ComBat is a statistical method used to correct batch effects in RNA-Seq studies when combining multiple datasets. It leverages a negative binomial regression model and empirical Bayesian framework to estimate and adjust for batch effects, while attempting to preserve biological variance. After defining a regression model for each gene, it estimates a 'batch-free' distribution that allows for the alignment of data across different batches to a common reference. By adopting this approach, I was able to effectively

standardise data from different datasets, thereby improving the robustness and validity of my subsequent analyses.

4.3.4 Rhythmic Gene Selection

A recurring theme in machine learning applications to expression datasets is the significance of feature selection, and it holds particular importance in this example, where genes exhibiting a rhythmic pattern are crucial for accurately predicting the circadian time of the plant. I employed the following set of metrics to quantify rhythmicity:

- Autocorrelation – This metric measures the degree of correlation between a gene's temporal expression pattern and a delayed copy of itself. A high autocorrelation suggests rhythmic gene expression, making it an effective feature for predicting the plant's endogenous time.
- JTK-Cycle[397] – This published metric for rhythmicity utilizes a combination of a non-parametric test to establish data orderings and a measure of rank correlation to determine the optimal phase and period of an expression pattern.
- ARSER[398] – Another published method for quantifying rhythmicity, ARSER, involves transforming the data using an autoregressive spectrum and identifying peaks in this spectrum within the range of [20-28]. These peaks are then used as estimates for the period in a harmonic regression model.

By applying these metrics to evaluate each gene, I obtained a matrix of size [n x 3] containing scores relating to the rhythmicity of each gene. I calculated the mean of the three features for each gene, resulting in a single metric that was used to rank and select genes. Later, the methodology was refined by using MetaCycle's single metric for gene ranking, which is a specialised procedure for assessing gene rhythmicity.

Next, I selected the top-ranking n genes and calculated the relative amplitude error (RAE) for each of them. RAE is a metric that measures the goodness of fit for an

FFT-NLLS[389] curve-fitting algorithm. Due to there not existing open source code to calculate the RAE, I developed open-source Python code that follows these steps:

1. Remove linear trends in the temporal expression pattern by subtracting the linear least squares fit from the dataset.
2. Compute the fast Fourier transform of the detrended expression data.
3. Sequentially initialize the FFT-NLLS frequency, phase, and amplitude parameters $(\tau_i, \varphi_i, \alpha_i)$ by identifying the FFT peak frequencies. The frequency should be initialised as the FFT peak frequency, the phase as the angle of the FFT-transformed detrended expression data, and the amplitude as the mean of the absolute value of the amplitude.
4. Employ the **curve_fit** package from scipy to fit the FFT-NLLS model to each gene, optimizing the three parameters to minimise the mean squared error between the learned cosine curve and the expression data.
5. Utilise the **pcov** method of **curve_fit** to obtain a covariance matrix for the estimated parameters, allowing the generation of 95% confidence intervals.
6. To compute the relative amplitude error score, divide the amplitude parameter by the range of the 95% confidence interval of the amplitude error. A smaller value indicates a more reliable rhythmic gene.

Once RAE has been calculated, I select the top-ranking n genes to proceed with further scaling between 0 and 1. This scaling enables easy identification of the ratio between the maximum expression in the training and validation datasets. Any genes with a maximum expression in the validation dataset exceeding 1.4 times the maximum expression in the training dataset are subsequently removed. Consequently, a subset of highly rhythmic genes with consistent amplitude between the training and validation datasets is obtained.

4.3.5 Circular Loss Function for Time Prediction

When dealing with cyclic variables, such as time measurements that repeat periodically, calculating the squared distance between predictions and targets using a traditional mean squared error loss function can lead to inflated distances in certain

cases. Consider an example where the target variable, denoted as y, is cyclic with a period of 24 hours. Suppose the predicted value $\hat{y}$ is 22, while the target value y is 2. The squared distance between y and $\hat{y}$ is $20^2 = 400$ when using the standard regression loss function of mean squared error (Figure 27).

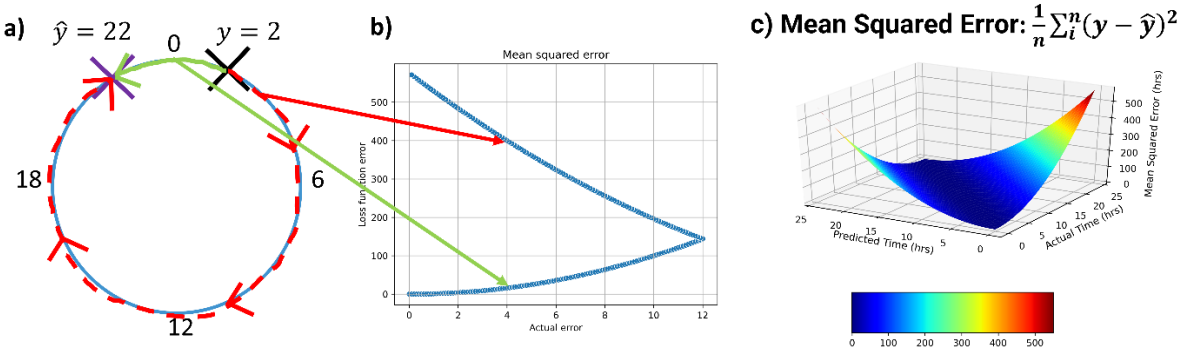

Figure 26.

Mean squared error loss function when applied to a cyclic variable. ***(a)*** *The two possible distances between a target value of 2, and a predicted value of 22.* ***(b)*** *Graph displaying the corresponding squared errors for the two possible distances.* ***(c)*** *Loss curve in the case of a cyclic variable with period 24.*

However, if one considers that $y = 2 mod(24)$ meaning that y can take the values $2, 26, 50$, then the appropriate squared distance to measure would be $(26 - 22)^2 = 16$ (Figure 28). This example demonstrates why using the mean squared distance is not appropriate when the target is cyclic. A more appropriate measure of distances in a modular setting is to map the points onto a circle that wraps around the modulus – 24 in the aforementioned example. To address this, I transform the dependent variable using two new variables defined as the sine and cosine of the variable:

$$Y_{cos} = -\cos\left(\frac{2\pi Y}{p}\right)$$

$$Y_{sin} = sin\left(\frac{2\pi Y}{p}\right)$$

Here, p represents the period of the cyclic variable, and Y denotes the value of the cyclic variable itself. By plotting the set of points that these transformed values can take, I observe that the range of the dependent variable becomes the unit circle. Training a model to predict values within this range enables both the dependent variable and the prediction to be mapped onto the unit circle, where the distance

between the points corresponds to the angle of the arc formed by connecting the coordinates to the circle's centre.

The distance that I aim to minimise is defined as the angle between the prediction and the dependent variable. I denote this distance as L and calculate it using the following formula:

$$L = \cos^{-1}\left(\frac{[Y_{cos} \quad Y_{sin}] \cdot \begin{bmatrix}\hat{Y}_{cos} \\ \hat{Y}_{sin}\end{bmatrix}}{||[Y_{cos} \quad Y_{sin}]||[\hat{Y}_{cos} \quad \hat{Y}_{sin}]||}\right)$$

$$= \cos^{-1}\left(\frac{Y_{cos}\hat{Y}_{cos} + Y_{sin}\hat{Y}_{sin}}{||[\hat{Y}_{cos} \quad \hat{Y}_{sin}]||}\right)$$

$$= \cos^{-1}\left(\frac{Y_{cos}\hat{Y}_{cos} + Y_{sin}\hat{Y}_{sin}}{\sqrt{\hat{Y}_{cos}^2 + \hat{Y}_{sin}^2}}\right)$$

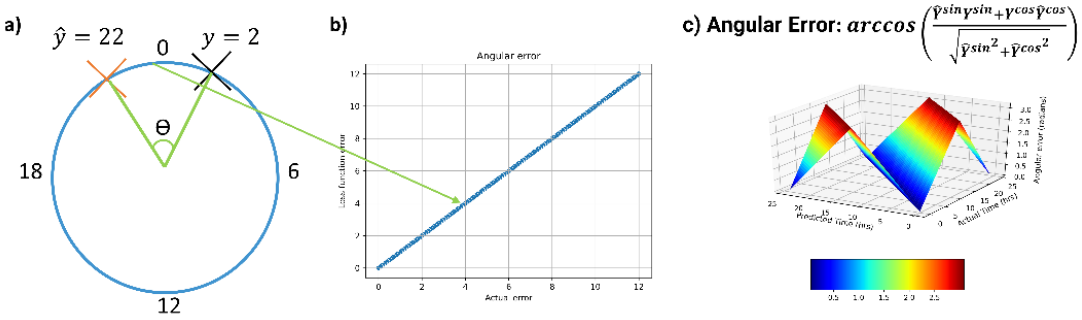

Figure 27.

Angular error loss function when applied to a cyclic variable. ***(a)*** *The angle θ between a target value of 2, and a predicted value of 22.* ***(b)*** *Graph displaying the corresponding error for the angle of θ.* ***(c)*** *The angular loss curve in the case of a cyclic variable with period 24.*

4.3.6 Custom Multi-Output Linear Regression

Now that I have defined the circular loss function, I proceed with selecting a suitable model. In this section, I present a multi-output circular linear regression model that

utilises stochastic gradient descent in combination with the circular loss function. This model enables accurate learning of the relationship between a set of covariates (such as gene expression) and a transformed cyclic response variable (such as time).

Typically, linear regression uses least-squares estimation, first calculating the mean squared distance between the predictions and the target $MSE = \frac{1}{n}\sum_i^n (y_i - \hat{y}_i)^2$ which is the loss function to minimise. The simple linear regression model can be written as:

$$Y_i = \beta_0 + \beta_1 X_{i,1} + \beta_2 X_{i,2} + \cdots + \beta_p X_{i,p} + \epsilon_i$$

Where β_0 is the model intercept, $\beta_{\{1,\ldots,p\}}$ are covariates for the p independent variables, and ε is the error term.

Stochastic gradient descent forms the basis of my planned approach and an iterative optimisation algorithm commonly used for training machine learning models. It efficiently handles large datasets by updating the model parameters incrementally based on randomly selected subsets of the training data, known as mini-batches.

The goal of stochastic gradient descent is to minimise the loss function by iteratively adjusting the model parameters in the direction that leads to the steepest descent. This iterative process involves the following steps:

1. Randomly initialise the model parameters, including the coefficients $\beta_0, \beta_1, \ldots, \beta_p$.
2. Shuffle the training dataset.
3. Divide the shuffled dataset into mini-batches.
4. For each mini-batch:
    a. Compute the predicted values, $\hat{y}$, based on the current parameter values and the corresponding covariate values.
    b. Calculate the gradient of the loss function with respect to the model parameters using the circular loss function.
    c. Update the model parameters by taking a step in the direction opposite to the gradient, multiplied by a learning rate. Mathematically, the parameter update can be written as:

$\beta_j = \beta_j - \eta \frac{\partial L}{\partial \beta_j}$, where β_j is the jth model parameter η is the learning rate, and L is the loss function.

5. Repeat steps 4a-4c for a predefined number of epochs or until convergence is achieved.

Through this iterative process, stochastic gradient descent gradually adjusts the model parameters to find an optimal solution that minimises the circular loss function. By updating the parameters based on mini-batches, stochastic gradient descent efficiently traverses the training data and avoids getting stuck in local minima.

Applying stochastic gradient descent with the circular loss function to my multi-output linear regression model allows me to effectively capture the relationships between the covariates and the transformed cyclic response variable. The iterative nature of stochastic gradient descent helps in progressively refining the model's predictions and improving its accuracy. Here, I will outline the necessary mathematics that I performed to program my circular linear regression model in Python using SGD.

In the case of predicting a cyclic target,

$$X \sim n\ samples,\ p\ genes$$

$$\begin{bmatrix} x_1^1 & \dots & x_1^p \\ \vdots & \ddots & \vdots \\ x_n^1 & \dots & x_n^p \end{bmatrix}$$

$$Y = \{y_0, y_1, \dots, y_n\}$$

Where $Y = mod(t, p)$ where t is the measured value and p is the period of the cyclic variable.

Therefore, $Y^{sin} \sim n\ labels = \left[y_1^{sin},\ y_2^{sin},\ \dots,\ y_n^{sin}\right]$ and

$Y^{cos} \sim n\ labels = [y_1^{cos},\ y_2^{cos},\ \dots,\ y_n^{cos}]$.

Since there are two targets, the model will need two sets of weights:

$W^{sin} \sim p\ weights = \left[w_1^{sin}, w_2^{sin},\ \dots, w_p^{sin}\right]$

$W^{cos} \sim p\ weights\ = [w_1^{cos}, w_2^{cos}, \ \ldots, w_p^{cos}]$

Resulting in two sets of predictions:

$\hat{Y}^{sin} \sim n\ predictions = [\hat{y}_1^{sin}, \hat{y}_2^{sin}, \ \ldots, \hat{y}_n^{sin}]$

$\hat{Y}^{cos} \sim n\ predictions = [\hat{y}_1^{cos}, \hat{y}_2^{cos}, \ \ldots, \hat{y}_n^{cos}]$

The targets and predictions will lie on a unit circle, and the error is the angle between the two vectors which can be calculated using the following formula:

$$\theta = \cos^{-1}\left(\frac{a \cdot b}{|a||b|}\right)$$

Applying that to this specific case, where a = (Y^{cos}, Y^{sin}) and $b = \left(\hat{Y}^{cos}, \hat{Y}^{sin}\right)$, the loss function is:

$$L\ =\ \cos^{-1}\left(\frac{\hat{Y}^{sin}Y^{sin} + Y^{cos}\hat{Y}^{cos}}{\sqrt{\hat{Y}^{sin^2} + \hat{Y}^{cos^2}}}\right)$$

Meaning that the values needed to update the weights, W^{sin} and W^{cos}, are:

$$w_i^{sin'} = w_i^{sin} - \alpha\frac{\partial L}{\partial w_i^{sin}}\ (1)$$

$$w_i^{cos'} = w_i^{cos} - \alpha\frac{\partial L}{\partial w_i^{cos}}\ (2)$$

Where α is the learning rate. To calculate $\frac{\partial L}{\partial w_i^{sin}}$ and $\frac{\partial L}{\partial w_i^{cos}}$, I can use the chain rule:

$$\frac{\partial L}{\partial w_i^{sin}} = \frac{\partial L}{\partial \hat{y}^{sin}} \times \frac{\partial \hat{y}^{sin}}{\partial w_i^{sin}}\ (3)$$

$$\frac{\partial L}{\partial w_i^{cos}} = \frac{\partial L}{\partial \hat{y}^{cos}} \times \frac{\partial \hat{y}^{cos}}{\partial w_i^{cos}}(4)$$

The simple part is calculating $\frac{\partial \hat{y}^{sin}}{\partial w_i^{sin}}$ and $\frac{\partial \hat{y}^{cos}}{\partial w_i^{cos}}$ as $\hat{y}^{sin}$ and $\hat{y}^{cos}$ are simple equations:

$$\hat{y}^{sin} = x^1\, w_1^{sin} + x^2\, w_2^{sin}\ (5)$$

$$\hat{y}^{sin} = x^1\, w_1^{cos} + x^2\, w_2^{cos}\ (6)$$

$$\frac{\partial \hat{y}^{sin}}{\partial w_i^{sin}} = x^i \ (7)$$

$$\frac{\partial \hat{y}^{cos}}{\partial w_i^{cos}} = x^i \ (8)$$

Where α is the learning rate. To calculate $\frac{\partial L}{\partial w_i^{sin}}$ and $\frac{\partial L}{\partial w_i^{cos}}$, use the chain rule:

$$\frac{\partial L}{\partial w_i^{sin}} = \frac{\partial L}{\partial \hat{y}^{sin}} \times \frac{\partial \hat{y}^{sin}}{\partial w_i^{sin}} = \frac{\partial L}{\partial \hat{y}^{sin}} \times x^i \ (3)$$

$$\frac{\partial L}{\partial w_i^{cos}} = \frac{\partial L}{\partial \hat{y}^{cos}} \times \frac{\partial \hat{y}^{cos}}{\partial w_i^{cos}} = \frac{\partial L}{\partial \hat{y}^{cos}} \times x^i \ (4)$$

The more complex part is calculating $\frac{\partial L}{\partial \hat{y}^{sin}}$ and $\frac{\partial L}{\partial \hat{y}^{cos}}$ as I start with the more complicated equation:

$$L = arccos\left(\frac{\hat{y}^{sin} y^{sin} + y^{cos} \hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2} + \hat{y}^{cos^2}}}\right) (9)$$

$$\frac{\partial L}{\partial \hat{y}^{sin}} = \frac{\partial}{\partial \hat{y}^{sin}}\left(arccos\left(\frac{\hat{y}^{sin} y^{sin} + y^{cos} \hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2} + \hat{y}^{cos^2}}}\right)\right) (10)$$

$$Apply\ chain\ rule\ \frac{df(u)}{dx} = \frac{df}{du} \times \frac{du}{dx}$$

$$f = arccos(u),\ u = \frac{\hat{y}^{sin} y^{sin} + y^{cos} \hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2} + \hat{y}^{cos^2}}} \ (11)$$

$$\frac{\partial L}{\partial \hat{y}^{sin}} = \frac{\partial}{\partial u}(\arccos(u)) \frac{\partial}{\partial \hat{y}^{sin}}\left(\frac{\hat{y}^{sin} y^{sin} + y^{cos} \hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2} + \hat{y}^{cos^2}}}\right) (12)$$

$$\frac{\partial}{\partial u}\arccos(u) = -\frac{1}{\sqrt{1-u^2}} \ (13)$$

Substitute using (11), (12), and (13) to produce:

$$\frac{\partial L}{\partial \hat{y}^{sin}} = -\frac{1}{\sqrt{1-\left(\frac{\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\right)}}\frac{\partial}{\partial \hat{y}^{sin}}\left(\frac{\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\right) (14)$$

Next stage is to calculate $\frac{\partial}{\partial \hat{y}^{sin}}\left(\frac{\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\right)$, then I will be able to calculate $\frac{\partial L}{\partial \hat{y}^{sin}}$

$$Apply\ quotient\ rule\colon \left(\frac{f}{g}\right)' = \frac{f'g-g'f}{g^2}$$

$$\frac{\partial}{\partial \hat{y}^{sin}}\left(\frac{\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\right) = \frac{\frac{\partial}{\partial \hat{y}^{sin}}\left(\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}\right)\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}-\frac{\partial}{\partial \hat{y}^{sin}}\left(\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}\right)\left(\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}\right)}{\left(\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}\right)^2} (15)$$

$$\frac{\partial}{\partial \hat{y}^{sin}}\left(\frac{\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\right) = \frac{y^{sin}\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}-\frac{\hat{y}^{sin}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\left(\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}\right)}{\left(\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}\right)^2} (16)$$

Simplify:

$$\frac{\partial}{\partial \hat{y}^{sin}}\left(\frac{\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\right) = \frac{-y^{cos}\hat{y}^{cos}\hat{y}^{sin}+y^{sin}\hat{y}^{cos^2}}{\left(\hat{y}^{sin^2}+\hat{y}^{cos^2}\right)\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}} (17)$$

(14) and (17) contain the terms required.

I now have everything needed to calculate $\frac{\partial L}{\partial \hat{y}^{sin}}$

$$\frac{\partial L}{\partial \hat{y}^{sin}} = -\frac{1}{\sqrt{1-\left(\frac{\hat{y}^{sin}y^{sin}+y^{cos}\hat{y}^{cos}}{\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}\right)}}\frac{-y^{cos}\hat{y}^{cos}\hat{y}^{sin}+y^{sin}\hat{y}^{cos^2}}{\left(\hat{y}^{sin^2}+\hat{y}^{cos^2}\right)\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}}}$$

$$= -\frac{-y^{cos}\hat{y}^{cos}\hat{y}^{sin}+y^{sin}\hat{y}^{cos^2}}{\left(\hat{y}^{sin^2}+\hat{y}^{cos^2}\right)\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}-y^{sin^2}\hat{y}^{sin^2}-2y^{sin}\hat{y}^{sin}y^{cos}\hat{y}^{cos}-y^{cos^2}\hat{y}^{cos^2}}}$$

Returning to the original equation to calculate the gradient for w_i^{sin}:

$$\frac{\partial L}{\partial w_i^{sin}} = \frac{\partial L}{\partial \hat{y}^{sin}} \times \frac{\partial \hat{y}^{sin}}{\partial w_i^{sin}}\frac{\partial L}{\partial w_i^{sin}}$$

$$= -\frac{-y^{cos}\hat{y}^{cos}\hat{y}^{sin}+y^{sin}\hat{y}^{cos^2}}{\left(\hat{y}^{sin^2}+\hat{y}^{cos^2}\right)\sqrt{\hat{y}^{sin^2}+\hat{y}^{cos^2}-y^{sin^2}\hat{y}^{sin^2}-2y^{sin}\hat{y}^{sin}y^{cos}\hat{y}^{cos}-y^{cos^2}\hat{y}^{cos^2}}} \times x^i$$

Solving these equations results in the following loss function and gradients for backpropagation:

Angular error:

$$L = arccos\left(\frac{\hat{Y}^{sin}Y^{sin}+Y^{cos}\hat{Y}^{cos}}{\sqrt{\hat{Y}^{sin^2}+\hat{Y}^{cos^2}}}\right)$$

Sine gradient

$$\frac{\partial L}{\partial w_i^{sin}}$$

$$= -\frac{-Y^{cos}\hat{Y}^{cos}\hat{Y}^{sin}+Y^{sin}\hat{Y}^{cos^2}}{\left(\hat{Y}^{sin^2}+\hat{Y}^{cos^2}\right)\sqrt{\hat{Y}^{sin^2}+\hat{Y}^{cos^2}-Y^{sin^2}\hat{Y}^{sin^2}-2Y^{sin}\hat{Y}^{sin}Y^{cos}\hat{Y}^{cos}-Y^{cos^2}\hat{Y}^{cos^2}}} \times x^i$$

Cosine gradient:

$$\frac{\partial L}{\partial w_i^{cos}}$$

$$= -\frac{-Y^{sin}\hat{Y}^{cos}\hat{Y}^{sin}+Y^{cos}\hat{Y}^{sin^2}}{\left(\hat{Y}^{sin^2}+\hat{Y}^{cos^2}\right)\sqrt{\hat{Y}^{sin^2}+\hat{Y}^{cos^2}-Y^{sin^2}\hat{Y}^{sin^2}-2Y^{sin}\hat{Y}^{sin}Y^{cos}\hat{Y}^{cos}-Y^{cos^2}\hat{Y}^{cos^2}}} \times x^i$$

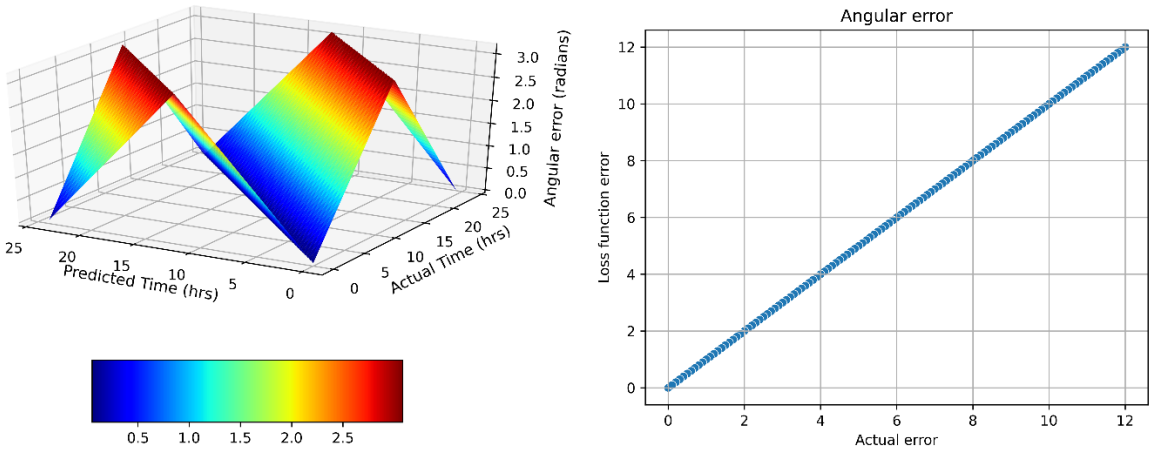

Figure 28.

Left – Loss curve for the angular error function over the input space given a period of 24.
Right – Loss curve for the angular error between the actual vector and the predicted vector.

In the case that I would want to square the angular error in order to get a smoother convergence due to steeper gradient further from target and shallower gradient closer to the target, the equations would become:

Squared angular error:

$$L = \left(arccos\left(\frac{\hat{Y}^{sin} Y^{sin} + Y^{cos} \hat{Y}^{cos}}{\sqrt{\hat{Y}^{sin^2} + \hat{Y}^{cos^2}}} \right) \right)^2$$

Sine gradient:

$$\frac{\partial L}{\partial w_i^{sin}}$$

$$= -\frac{2\arccos\left(\frac{Y^{sin} \hat{Y}^{sin} + Y^{cos} \hat{Y}^{cos}}{\sqrt{\hat{Y}^{sin^2} + \hat{Y}^{cos^2}}} \right) \left(-Y^{cos} \hat{Y}^{cos} \hat{Y}^{sin} + Y^{sin} \hat{Y}^{cos^2} \right)}{\left(\hat{Y}^{sin^2} + \hat{Y}^{cos^2} \right) \sqrt{\hat{Y}^{sin^2} + \hat{Y}^{cos^2} - Y^{sin^2} \hat{Y}^{sin^2} - 2Y^{sin} \hat{Y}^{sin} Y^{cos} \hat{Y}^{cos} - Y^{cos^2} \hat{Y}^{cos^2}}} \times x^i$$

Cosine gradient:

$$\frac{\partial L}{\partial w_i^{cos}}$$

$$= -\frac{2\arccos\left(\frac{Y^{sin}\hat{Y}^{sin} + Y^{cos}\hat{Y}^{cos}}{\sqrt{\hat{Y}^{sin^2} + \hat{Y}^{cos^2}}}\right)\left(-Y^{sin}\hat{Y}^{cos}\hat{Y}^{sin} + Y^{cos}\hat{Y}^{sin^2}\right)}{\left(\hat{Y}^{sin^2} + \hat{Y}^{cos^2}\right)\sqrt{\hat{Y}^{sin^2} + \hat{Y}^{cos^2} - Y^{sin^2}\hat{Y}^{sin^2} - 2Y^{sin}\hat{Y}^{sin}Y^{cos}\hat{Y}^{cos} - Y^{cos^2}\hat{Y}^{cos^2}}} \times x^i$$

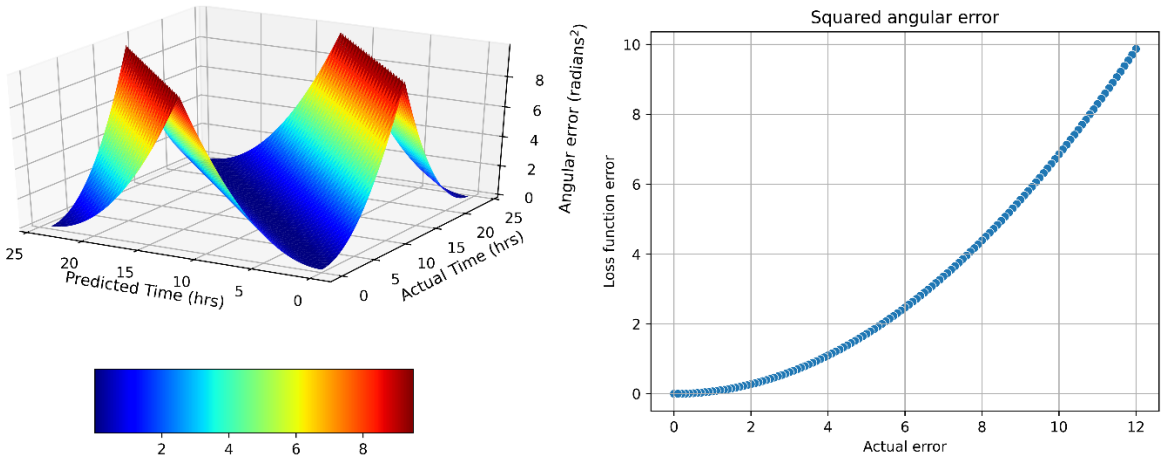

Figure 29.

Left – Loss curve for the squared angular error function over the input space given a period of 24.
Right – Loss curve for the squared angular error between the actual vector and the predicted vector.

These are all of the terms necessary for my multi-output (sine and cosine transformed target) circular linear regression model that converges using stochastic gradient descent. Using the described mathematics, I programmed this SGD-based model in Python and the code is available on GitHub.

4.3.7 Multi-Output Neural Network

I developed a machine learning (ML) pipeline to predict the circadian time (phase) at any given transcriptomic sampling timepoint using gene expression data from a set of marker genes. In this study, I employed TensorFlow (v2.0.0), a Python package

containing functions for developing deep learning and neural network models, to construct an artificial neural network. The advantage of using TensorFlow was its capability to easily implement a custom multi-output cyclical loss function, which accurately represented the cyclical nature of time. This approach was preferred over traditional ML methods, such as those offered by the scikit-learn library, which were limited to one-dimensional, linear representations of time using standard loss functions like mean squared error. Since my model required a more comprehensive representation of the cyclical nature of time, I needed to develop custom code. The code for my approach can be found in a Jupyter Notebook, along with instructions for running it: https://github.com/AHallLab/PredictingCircadianTime.

To create my model, I designed a shallow neural network using TensorFlow (v2.0.0). The network consisted of three dense layers with ReLU activation functions and varying numbers of neurons: 32, 128, and 512, respectively. This was followed by a softmax layer with two neurons – for the sine and cosine predictions.

Given the cyclical nature of the target variable, which ranged from 0 to 24 hours, traditional regression loss functions were not appropriate for this task. Instead, I programmed the aforementioned squared angular loss function to quantify the error in the predictions. This loss function measured the squared angle between the actual circadian time and the predicted circadian time, after transforming both onto a unit circle. This approach provided a more suitable and meaningful measure of error for my model, with the squared loss was chosen to achieve a smoother convergence.

By implementing this ML pipeline and leveraging TensorFlow’s capabilities, I was able to accurately predict the circadian time using gene expression data. The use of a custom cyclical loss function and careful optimisation of the neural network’s architecture contributed to the success of my approach.

4.3.8 Bayesian Hyperparameter Tuning

Reducing the error within the validation dataset was an objective so I invested time fine-tuning various hyperparameters. These included the number of genes and the

artificial neural network architecture, amongst others. In my pursuit of optimising these hyperparameters, I employed Bayesian optimisation via the **hyperas** package. The least variance for a gene to be incorporated, as well as the total number of genes included, were optimised to minimise the validation error. The number of genes parameter significantly influences the model's performance. Its sensitivity escalates particularly at lower numbers, presumably due to the potential for either drastically degrading or enhancing prediction quality should a gene with significant variant amplitude across datasets be added or removed.

4.3.9 Custom Forward Floating Feature Selection

Since machine learning algorithms are likely to perform better and generalise better if the features are not highly correlated with each other, I wanted to ensure that the distribution of phase for the subset of genes was uniform across the potential phases. This concept should be intuitive as if two genes share the same rhythmic pattern and phase, their expression patterns will very highly correlated and therefore a model will identify approximately the same pattern whether one or both of the genes is used for training. By including genes that share different phase patterns, the amount of information encoded in the input data is much greater, and this could be proved by calculating the joint mutual information between gene A, gene B, and the time. My colleague, Dr Rachel Rusholme-Pilcher, used WGCNA to cluster the genes' temporal expression patterns and the 8 produced clusters were mainly differentiated by phase.

In this project, I used a combination of filter and wrapper feature selection methods to select *n* circadian genes for model training, prioritising weighted representation of genes from each of the 8 expression sub-clusters generated by WGCNA gene co-expression network analysis performed by my colleague. This was done with the intention of improving generalisation and robustness of the model as the similarity between features would be reduced and the diversity of features should enable the neural network model to engineer more intricate embeddings of the expression data compared to if all features were highly correlated and belonged to the same phase cluster.

My initial method of incorporating genes that had different phases, was to use the ranked list of genes and select the highest ranking $\frac{n}{8}$ genes for each cluster, giving me a final subset of genes that possessed a uniform phase distribution. I also created a modified sequential feature selection method (Figure 31) that iteratively selects genes depending on which clusters are underrepresented in the selected gene subset (Figure 32). Initially, the model is trained and evaluated using one feature with each gene being used individually and the gene resulting in the lowest angular loss is selected. All genes not belonging to the now overrepresented cluster are tried in combination with the already selected gene and the combination of two genes that minimises the error is selected.

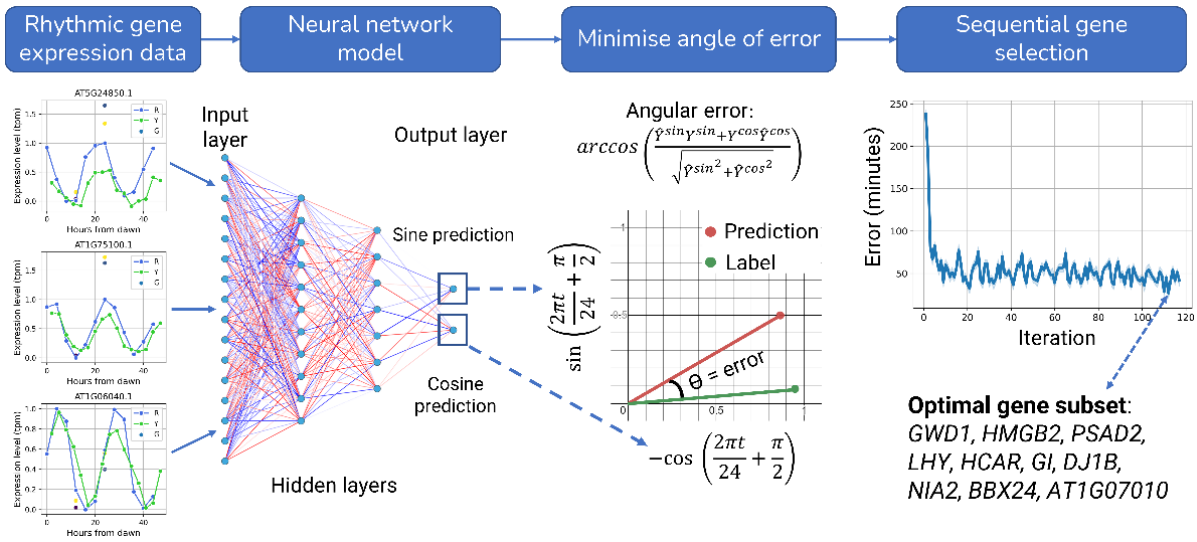

Figure 30.

The feature selection and model training methodology. From left to right, the rhythmically expressed genes that are filtered enter the neural network or linear regression model as input. The model generates a sine and cosine prediction and weights are optimised to minimise the angular loss. At each iteration of the SFFS, the gene that minimises the angular loss is added – this continues until convergence.

This iterative procedure continues until the desired number of genes is reached. Based on cross-validation scores, both of these methods outperformed using the most rhythmic genes independent of cluster, with the iterative method being best.

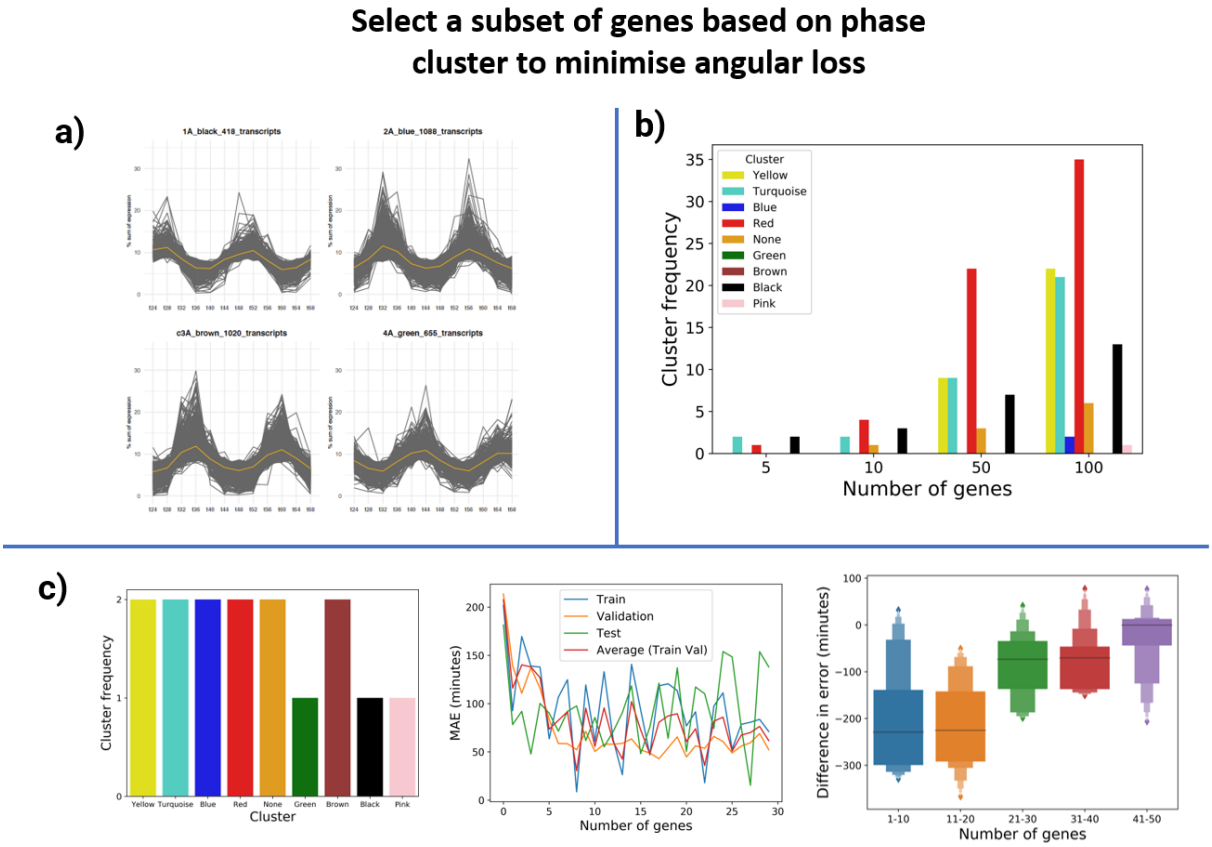

Figure 31.

Diverse marker genes outperform using most rhythmic genes

Using a subset of circadian marker genes that represented all clusters of expression outperformed only selecting the most rhythmic genes, especially when few genes were used. (A) Weighted gene coexpression analysis performed to generate clusters of rhythmic genes based on phase. (B) The frequency distribution of each cluster when the top ranked rhythmic genes were selected. (C) Left – The frequency distribution after applying the iterative feature selection algorithm I used. Middle – The error in minutes for the different numbers of genes selected when using the iterative feature selection algorithm. Right – Boxenplots showing the reduction in error for different numbers of genes when the feature selection algorithm was applied compared to selecting the most rhythmic genes independent of cluster.

4.3.10 Methods to Improve Generalisation

It was observed that the amplitude of the expression patterns was not consistent between datasets for many genes, causing problems for the fitted model as a gene in the validation dataset may have a trough higher than the peak in the training dataset, meaning all samples in the validation dataset would be predicted as the time of the peak in the training dataset. To tackle this problem, I implemented several methods such as maximum expression thresholds between datasets as well as using the Kolmogorov-Smirnov test[399] to measure whether the training and validation dataset were generated from the same distribution. Additional ideas aimed at improving the robustness of the supervised neural network model focussed on normalising genes with respect to each other so that if the amplitudes changed across different datasets, as long as a partner gene's amplitude changed in a similar way, that pair of genes can still be used to make accurate predictions. As well as these ideas, I investigated the effectiveness of using expression values that lie within a range of percentiles e.g. values between the 35^{th} and 65^{th} percentile for each sample. This would likely remove the genes that have a greatly different distribution of expression levels across two datasets, improving the robustness of the model. As well as applying this concept to the features, it could also be applied to the ensemble predictions that are typically averaged to generate a final prediction. By removing the largest and smallest predictions, outlier predictions would hopefully be removed.

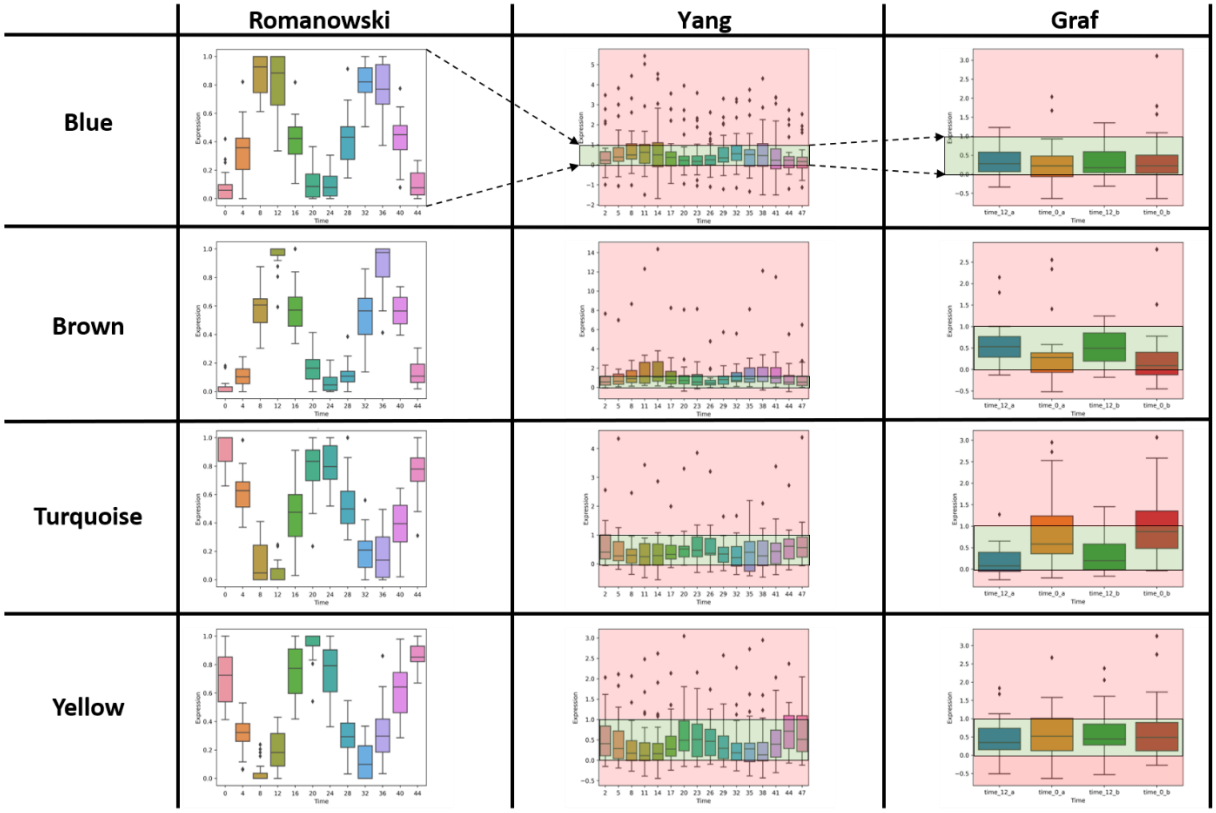

Figure 32.

Expression amplitude changing between datasets, values between quartiles mostly consistent

Boxplots showing the patterns of genes belonging to the respective row's cluster and the column's dataset. In most of the clusters, the values between the upper and lower quartile mostly stay within the [0, 1] range so might be more generalisable features in comparison to genes.

4.3.11 Cross Correlation

In separate research with my colleague, Dr Hannah Rees', I developed several Python functions. These functions enabled one to compute the relative amplitude error and cross-correlation, aiming to identify temporal expression patterns that exhibit a phase difference. In this context, cross-correlation was evaluated to yield a correlation score across varying quantities of time point shifts. The number of time points, or 'lag', that optimises the cross-correlation score represents the necessary shift along the time axis for a gene's expression pattern to most closely align with another gene's expression pattern.

I employed cross-correlation analysis to detect the time shift (lag) that produced the highest correlation between two rhythms. This methodology was crucial in identifying modules that peaked either synchronously (displaying a peak lag of 0 hours) or asynchronously (demonstrating a peak lag of 4, 8 or 12 hours). This was accomplished by correlating the 'eigengenes' for each module. Moreover, my colleagues utilised my cross-correlation function to discover unbalanced phases within rhythmic wheat triads.

Prior to calculating the cross-correlation between two expression rhythms, I initially normalised both expression patterns using their respective means and standard deviations. This ensured that the output mirrored a time-dependent Pearson correlation coefficient, which could range between -1 and 1:

$$Z_A = \frac{X_A - \bar{X}_A}{s_A}, Z_B = \frac{X_B - \bar{X}_B}{s_B}$$

Where $Z_i, X_i, \bar{X}_i$, and s_i represent the standardised expression level, tpm expression level, mean expression level, and standard deviation of gene A and B respectively.

Once both expression patterns had been normalised, the discrete cross-correlation between the two was computed using the **np.correlate** function, subsequently divided by the number of time points in the expression signal. This returns the Pearson correlation coefficient at different lags. The index of the array boasting the highest Pearson correlation coefficient score corresponds to the lag that maximises the resemblance between the two temporal expression patterns.

4.4 Results

In this results section, I will present the performance and outputs of the supervised machine-learning methods that ChronoGauge comprises of. I will discuss the overall performance of the pipeline in terms of mean absolute error and these results will be compared to established 'gold standard' methodologies, such as ZeitZeiger[391], that have a proven track record in predicting endogenous time. Furthermore, I will present and provide interpretations for the predictions generated from the clock mutant and different ecotype datasets.

4.4.1 Accuracy and Error of Predictions

4.4.1.1 Arabidopsis Thaliana Circadian Datasets

Table 9 highlights the mean absolute errors (MAE) of the predictions of circadian time without hyperparameter optimisation on the three temporal transcriptomic datasets, using different sized subsets of the highest ranked rhythmic genes. The lowest MAE, based on the test dataset, was 104 minutes and was observed with a selected subset of 50 transcripts (Table 9).

Table 9. ***A table containing the mean absolute errors of the ChronoGauge neural network model on the training, validation, and test datasets using different numbers of input genes.***

Error (MAE)	**1000 genes**	**100 genes**	**50 genes**	**10 genes**	**5 genes**
Training	9 mins	**4 mins**	44 mins	11 mins	10 mins
Validation	125 mins	**81 mins**	119 mins	84 mins	125 mins
Test	180 mins	289 mins	**104 mins**	146 mins	125 mins

Using confidence of rhythmicity for transcript prioritisation, I noted that the representation of the subsets of transcripts across the 8 co-expression modules generated by the WGCNA gene co-expression network analysis was not uniform.

This reflects an uneven representation across the phases of rhythmic expression. Therefore, I prioritised selection of transcripts using model interpretation in the form of feature selection to make the frequency distribution across the modules more uniform. Optimising performance using this method feature selection based on the validation dataset, the best performing model overall used a final subset of 15 transcripts and had a MAE of 21 minutes on the Romanowski et al. training data, 56 minutes on the Yang et al. validation dataset and 46 minutes on the test data from Graf et al.

4.4.1.2 Arabidopsis Thaliana Shoot Dataset

Despite the Mas et al. dataset deriving from the shoot apex as opposed to leaf tissue, ChronoGauge consistently yielded accurate predictions across various time points. The training was performed on the Romanowski, Yang, Locke, and Yavonsky datasets, subsequently treating the Mas dataset as a validation set. This approach ensured that the genes selected through my tailored sequential feature selection algorithm were chosen with an intent to minimise the error on the Mas dataset. The mean absolute error during training across the four datasets for the neural network was 30.4 minutes, while the validation predictions resulted in a mean absolute error of 53.4 minutes (Figure 34). For the linear regression model, the training error was 42.1 minutes, and the validation error was 56.6 minutes, suggesting that the neural network achieved a better fit. These results underscore the precision of ChronoGauge’s predictions, maintaining their accuracy even when applied to samples from disparate tissue types.

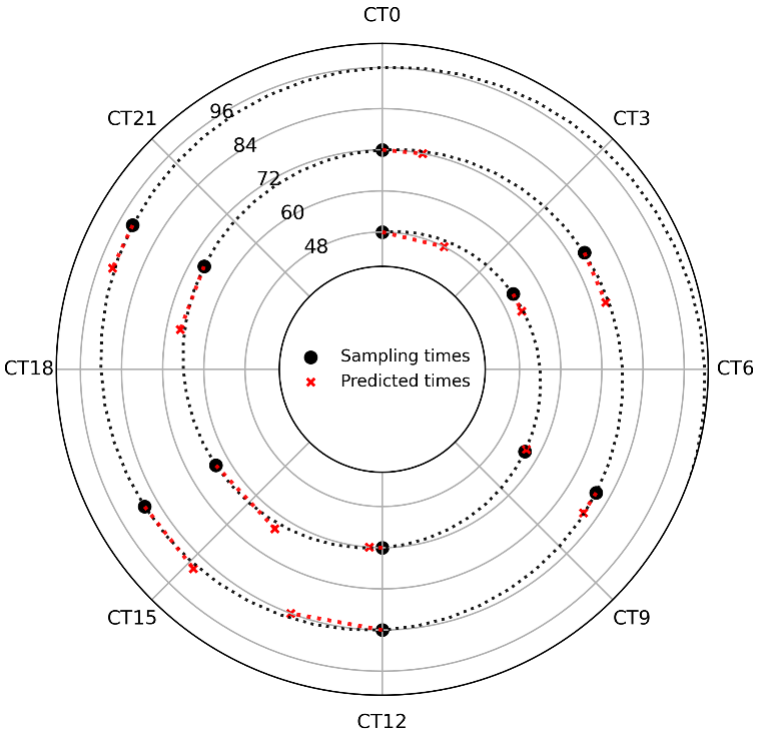

Figure 33.

This polar plot demonstrates ChronoGauge's predictions on the Mas dataset. The radial axis signifies the time elapsed since the plant was subjected to constant light, whereas the polar axis represents time within a 24-hour cycle. Ground truth values are denoted by black points, whilst predictions are represented by red crosses.

4.4.1.3 Mutant and Temperature Variation Dataset

The Blair dataset encompasses samples that have been subjected to fluctuating temperatures and genetic modifications, specifically the knockout of either the CCA1 gene or the PRR7/9 genes. These alterations are known to cause disturbances to the internal circadian clock, particularly noticeable in instances where the CCA1 gene has been knocked out due to its crucial role in circadian clock regulation. Following the approach employed with the Mas dataset, the Romanowski, Locke, Yang, and Yavonsky datasets were utilised for model training, with the Mas dataset serving as the validation set. Hold-out test predictions were executed on the Blair dataset to examine if these influencing factors impacted the predictions made by ChronoGauge. The predictions would logically be out of sync with sampling times if the internal circadian clock had been perturbed.

Table 10. Mean absolute error (minutes) across various Arabidopsis genotypes at different temperatures.

Mutation	**10ºC**	**22ºC**	**37ºC**
WT	87.3	54.8	128.3
CCA1	301.1	316.2	322.0
PRR7/9	80.8	92.4	5.5

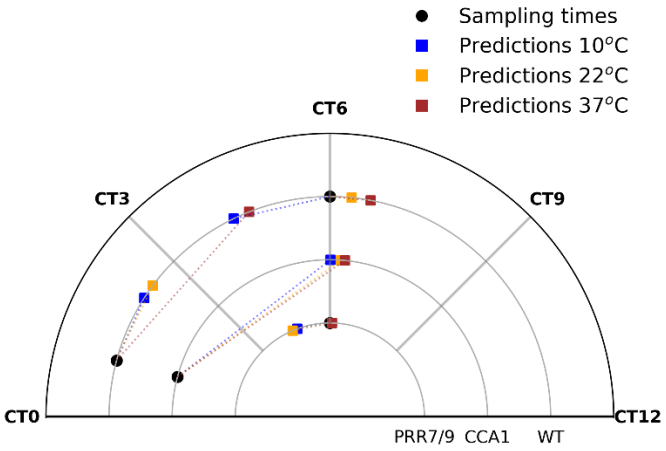

Figure 34.

This polar plot demonstrates ChronoGauge's predictions on the Blair dataset. The radial axis signifies the genotype of the plant, whereas the polar axis represents time within 12 hours of a 24-hour cycle. Ground truth values are denoted by black points, whilst predictions are represented by coloured squares corresponding to the temperature associated with the sample.

Table 10 and Figure 35 present the predictions and errors generated by the neural network model. It's noteworthy that errors for the CCA1 mutation samples are considerably larger than for other genotypes, implying that the internal clock of the CCA1 mutants is significantly disrupted. Interestingly, the errors for the PRR7/9 mutants resemble those of the wild type, despite the fact that the internal clock of the PRR7/9 mutant is known to be affected. The predictions for the wild type proved most accurate at 22 degrees Celsius, an expected outcome given that a majority of the training dataset samples were grown at a comparable temperature.

4.4.1.3 Cadenza and Wheat Pantranscriptome Datasets

After successfully establishing the capability to accurately predicted the internal circadian phase using gene expression markers in Arabidopsis, I extended my approach by applying the ChronoGauge tool to the wheat dataset previously analysed using the Trans-Learn software. The samples collected at the start and end of the day, dusk and dawn respectively, formed the central point of my research; my primary objective was to determine if any of these samples were inaccurately labelled.

To achieve this, I trained ChronoGauge using a wheat (Cadenza) circadian dataset, kindly provided by my colleague, Dr Hannah Rees. This dataset was composed of 16 samples, each collected at 4-hour intervals, culminating in a 64-hour timeframe. Each sample was replicated three times, ensuring the data's robustness. In terms of prediction performance, both the linear regression and neural network methods yielded cross-validation errors of less than 60 minutes, as shown in Table 11. To prevent data contamination, I ensured that all replicates were grouped together in the same fold.

Table 11. Training and cross validation errors on the Cadenza samples

Mean Absolute Error	**Training Folds**	**Validation Folds**
Neural Network	**9.6 minutes**	53.0 minutes
Linear Regression	14.1 minutes	**49.8 minutes**

After training and tuning the linear regression and neural network models with the aid of the Cadenza dataset, I proceeded to apply these models to the pantranscriptome dataset previously analysed in the Trans-Learn project. The objective of this was two-fold. Firstly, I aimed to make informed predictions on the dusk and dawn samples to validate the accuracy of their labels, thus improving the integrity of the data. Secondly, I sought to investigate the possibility of circadian phenotype variations across diverse wheat varieties, analysing the broad genetic diversity and its functional implications in this crop species.

Upon running predictions for all the wheat varieties, one particular sample, MAC_D3 (Mace wheat variety), presented itself as an anomaly. Contrary to its given label as a dusk sample, the model's prediction confidently suggested that it was sampled at dawn, as depicted in Figure 36. To validate this surprising outcome, I performed an additional analysis correlating the expression levels of the circadian biomarkers with the dusk (D) and dawn (V) MAC samples. The correlation patterns strongly supported the model's prediction, indicating that D3 indeed belonged to the V class.

	ARI	CAD	CLA	JAG	JUL	LAN	LER	MAC	MAT	NOR	PAR	ROB	STA	WEE
D1	16.15097	15.8109	15.32436	15.29045	15.76423	13.47662	11.14245	15.22633	15.86745	15.42942	14.15065	15.84583	13.81609	15.30165
D2	15.08964	14.97681	14.1453	15.38068	15.27862	13.42146	11.67714	14.88135	15.69022	15.44726	14.43477	15.78317	13.33836	14.43322
D3	16.1748	14.60517	14.69478	14.94933	15.07039	13.18643	14.11429	4.317986	14.69979	15.22206		15.45499	13.16234	14.71243
V1	5.326805	9.188091	4.873953	4.274358	2.315832	2.978872	4.546143	5.034788	3.31661	4.294131	8.732118	8.291213	2.979447	5.843838
V2	4.759813	6.235737	5.31181	1.779288	6.716242	6.172904	4.42047	5.826336	2.281831	1.493312	8.004608	7.12818	5.597784	4.203879
V3	8.070469	8.514313	5.542817	5.107335	7.410691	7.179209			3.246998	8.648324	4.432756	4.92875		6.546995

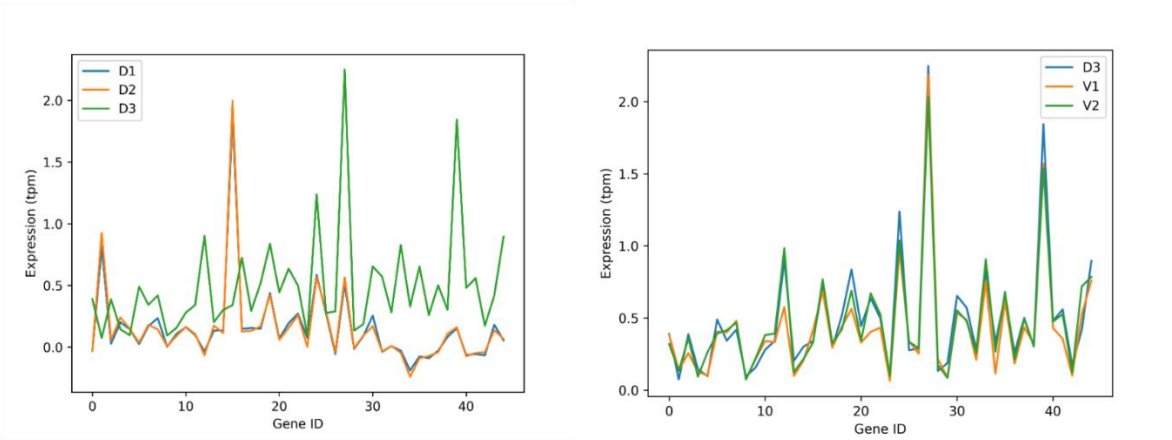

Figure 35.

Predictions on the wheat pantranscriptome dataset across different wheat varieties and dusk (D) and dawn (V) tissue samples. Top – Colour coded prediction table for the wheat varieties and tissue types. Bottom left – Biomarker expression for the MAC D samples, with D3 clearly not aligning with D1 and D2. Bottom right – Biomarker expression for the MAC V samples and D3, with D3 correlating highly with V1 and V2.

This empirical exploration demonstrates the potential of tools like ChronoGauge, serving as methods for analysing and ensuring quality control of biological datasets, resulting in more reliable downstream analyses. These findings have generated phenotypic estimations for this pantranscriptome wheat dataset that is currently under comprehensive analysis. These findings fit into a broader context of wheat

genomics and have contributed valuable information to my research group's ongoing research. This extensive analysis performed by my colleagues, inclusive of ChronoGauge's predictive modelling, is currently being prepared for an upcoming publication, set to enrich our understanding of diversity in the wheat transcriptome as well as the underlying mechanisms that control circadian rhythms.

4.4.2 Comparison of ChronoGauge with ZeitZeiger

My most proficient model was a neural network configuration, which utilised 15 transcripts to achieve a mean absolute error of 46 minutes on the test dataset. To gauge the efficacy of ChronoGauge directly, I contrasted it with ZeitZeiger, a methodology widely recognised as the benchmark for circadian time prediction. I employed the same datasets in my comparison as I did for my initial modelling to maintain consistency in the approach. Initially, I utilised the Romanowski dataset to fit ZeitZeiger, before generating predictions on the Yang dataset to optimise ZeitZeiger's hyperparameters, serving as the validation process. Subsequently, I used the Graf test dataset to compare the predictions generated by ZeitZeiger with those produced by ChronoGauge.

ChronoGauge significantly outclassed ZeitZeiger on the test dataset, as demonstrated by the MAE of 46 minutes versus 143 minutes (Figure 37). This is a testament to the precision with which ChronoGauge can generate circadian time predictions. It is worth noting that ZeitZeiger demonstrated a considerable discrepancy in training, validation, and test errors (MAE of 6 minutes on training, 119 on validation, and 143 on testing), which suggests overfitting.

I theorised that my method of selecting biomarker transcripts, which ensures an even representation across the different phases of rhythmic expression through my custom sequential feature selection approach, would lead to a more robust or generalisable mapping from expression data to internal circadian time. In other words, I hoped to minimise the risk of overfitting, and this comparative analysis supports my hypothesis. This effectively showcases ChronoGauge's adaptability and reliability in

generating accurate predictions, enabling research that could lead to a more robust understanding of circadian rhythms and their influence on biological processes.

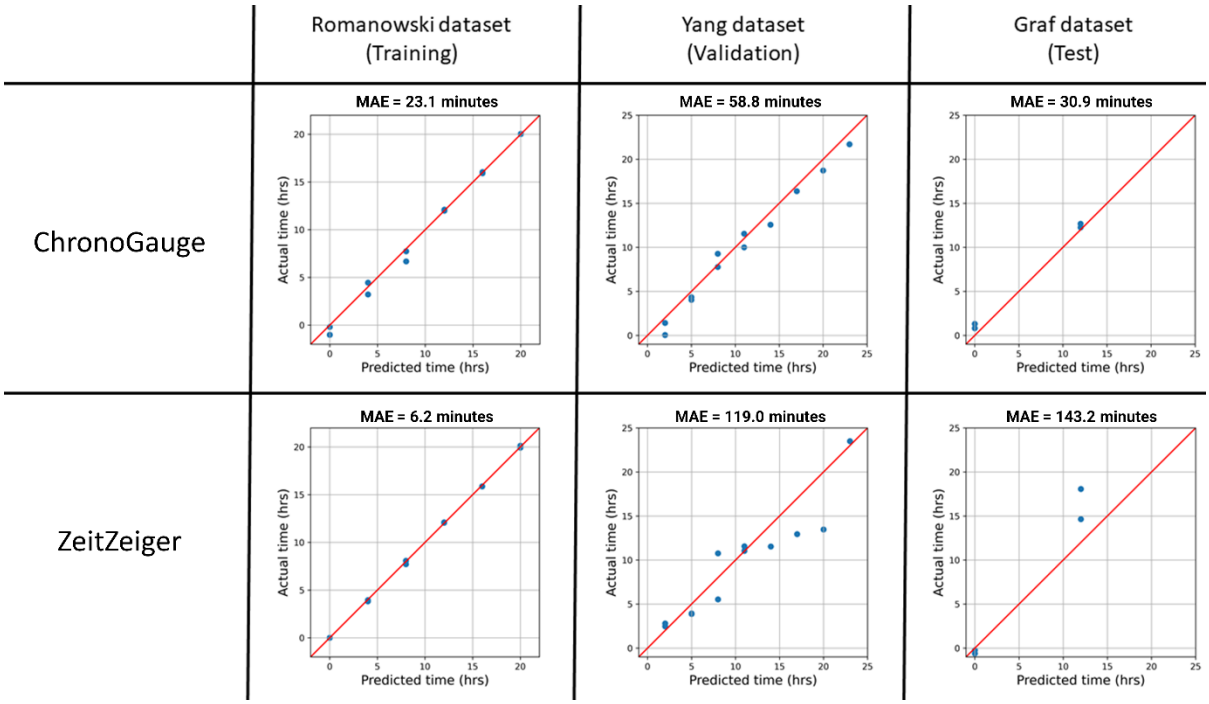

Figure 36.

Comparison of ChronoGauge with ZeitZeiger

Six scatterplots displaying the predictive performance of ChronoGauge compared to the performance of ZeitZeiger on the same three datasets. Each blue point is a sample and the red lines represent where the actual time is equal to the predicted time. The title for each scatterplot is the mean absolute error (MAE) of the predictions of the given model and dataset combination, where the MAE is the average difference between the actual time and the predicted time in minutes.

4.4.3 Analysis of Arabidopsis Circadian Biomarkers

The final subset of 15 transcripts serves as a compact and efficient selection of biomarker transcripts, optimised for the accurate prediction of circadian time by the bespoke sequential feature selection method (Table 12). Surprisingly, none of the core clock genes were included among these 15 transcripts, however, they also displayed strong rhythmic expression patterns (Figure 38).

Table 12. List of 15 selected biomarker transcripts for the prediction of circadian time using ChronoGauge's neural network model.

Gene	**Name**
AT1G13650.1	
AT3G55450.1	PBL1
AT1G02930.2	GSTF6
AT1G79500.3	AtkdsA1
AT5G24850.1	CRY3
AT5G06870.1	PGIP2
AT5G01820.1	SR1
AT4G08870.1	ARGAH2
AT1G75100.1	JAC1
AT2G29650.2	PHT4
AT5G06690.1	WCRKC1
AT3G17609.2	HYH
AT4G15690.1	GRXS5
AT5G41460.1	
AT1G06040.1	STO

This analysis was carried out using the *Arabidopsis thaliana* ecotype Col-0. Nevertheless, when applying the model to the Ws-2 data, I observed an impressive mean absolute error (MAE) of just 53 minutes on this ecotype - a reduction of 5 minutes compared to the Col-0 ecotype upon which the model was initially trained. Overall, there didn't appear to be a substantial correlation between circadian time and prediction error, with one notable exception within the training dataset. Here, errors recorded at the 20-hour timepoint were markedly larger than those from other times. However, the range of error across the timepoints predominantly remained below 90 minutes. This level of resolution is well within acceptable limits for circadian time prediction, considering that typical sampling strategies involve intervals ranging from 2 to 4 hours. These facts highlight the generalisability of the biomarker transcripts that were selected by my feature selection methods and the models that were optimised through cross validation.

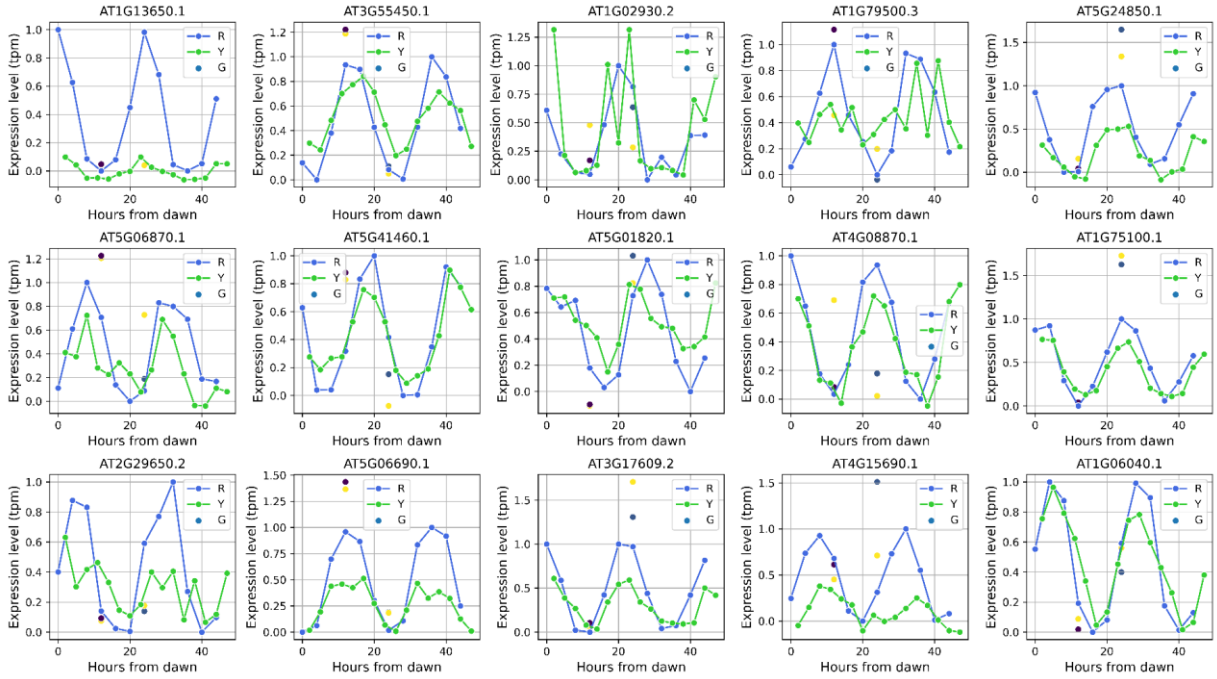

Figure 37.

Expression patterns of the 15 selected biomarker transcripts over the course of the 48 hours of the circadian experiment. The blue lines and datapoints denote the Romanowski samples, the green the Yang samples, and the points are from the Graf dataset.

The absence of core clock genes within the identified biomarker subset could be explained by a number of factors. It is possible that these genes, while critical to the functioning of the circadian clock, may not exhibit sufficient variability in their expression patterns to act as effective biomarkers for predicting specific circadian times. Additionally, while core clock genes set the pace of the biological clock, their expression levels might be buffered or influenced by various other biological processes, making them less distinct and therefore less useful as markers for specific times. Finally, the absence of core clock genes among the biomarkers might also highlight the complexity of the circadian system and suggest that many genes outside the core clock can contribute to and influence the timing of biological rhythms.

4.5 Discussion

4.5.1 Methods for Predicting the Circadian Clock

The prediction of the internal circadian time in organisms is a key component to developing our understanding of the interplay between genetic and environmental influences on biological rhythms. ChronoGauge has demonstrated its efficacy in this research area, setting a new benchmark in the accuracy and reliability of such predictions.

In this study, the power of ChronoGauge's feature selection methods has been unequivocally demonstrated. By selecting a compact and efficient subset of 15 biomarker transcripts, ChronoGauge was able to generate highly accurate predictions of circadian time. This lean approach to feature selection offers a balance between model complexity and accuracy, and helps to avoid overfitting, a problem often faced when dealing with high-dimensional data such as gene expression profiles. The selection of biomarkers also displayed an interesting characteristic - the absence of core clock genes. This discovery raises questions regarding the complexity of the circadian system.

Comparatively, while ZeitZeiger requires lower computational resources, its performance was significantly worse than ChronoGauge in my benchmark comparison. Despite being widely recognised as a gold standard in the field, ZeitZeiger's efficiency in resource use did not translate into superior predictive accuracy on the test dataset. This raises a significant point of consideration when choosing between these methods: the trade-off between computational efficiency and prediction accuracy. Here, ChronoGauge, despite being more computationally demanding, exploited a leaner subset of biomarkers and demonstrated that a carefully curated, efficient selection can result in a more accurate model.

Nonetheless, it's crucial to remember that this study primarily utilised smaller datasets. In future investigations, it would be valuable to assess the performance of

ChronoGauge on larger, more complex datasets, including those derived from different organisms or under varying environmental conditions.

Other methods, such as the Molecular Timetable method, also offer alternative approaches to predicting circadian time. This method, which utilises the time of peak expression of different genes to infer internal time, might offer different advantages or limitations compared to ChronoGauge and ZeitZeiger. A comprehensive comparison of these methods, as well as TimeSignatR, taking into account not only prediction accuracy but also considerations such as computational efficiency, scalability, and ease of interpretation, would be highly beneficial for the field.

In summary, the development and optimisation of methods for predicting the internal circadian time, such as ChronoGauge, hold great promise for our understanding of biological rhythms. However, further investigations are necessary to fully explore the potential of these methods, their limitations, and their implications for the field of chronobiology.

4.5.2 ChronoGauge's Challenges and Gene Expression Variability

Models developed to generate predictions across a multitude of independent gene expression datasets necessitate robustness to minor fluctuations in gene expression. However, the three datasets employed in my study exhibited major variations, with some rhythmic genes showing amplitudes over 10 and, in extreme cases, even 50 times higher in one dataset compared to another. This degree of variation presents a formidable challenge to any machine learning model, as it renders the task of making accurate predictions from data drawn from significantly divergent distributions near impossible.

Consequently, despite ChronoGauge's accurate performance on the validation and test datasets, there are concerns regarding its application to future datasets. Specifically, if the range of expression levels in the marker genes deviates substantially from what the model has been trained on, its performance is likely to be compromised. Therefore, the robustness of my model, while impressive under

current conditions, may be challenged when faced with substantially dissimilar data distributions.

This variation between different gene expression datasets will pose challenges to machine learning models, as it can introduce issues such as covariate shift. Covariate shift refers to differences in the distribution of input features (i.e., gene expression data) between training and testing datasets, which can lead to reduced model performance and generalisation. In the context of gene expression data, covariate shift can arise from various sources, including differences in experimental protocols, sample preparation techniques, data normalisation methods, and even batch effects.

One common scenario where covariate shift can occur is when gene expression datasets are generated by different research groups, despite similar experimental conditions. Research groups may have their own laboratory protocols, techniques, and equipment, which can result in subtle differences in the gene expression profiles obtained. For example, variations in RNA extraction methods, sample storage conditions, RNA sequencing platforms, and data pre-processing pipelines can all contribute to differences in gene expression datasets, even if the experiments were conducted under seemingly similar conditions. These differences in data characteristics can introduce bias and heterogeneity, which can negatively impact the performance and generalisability of machine learning models trained on these datasets.

The presence of covariate shift in gene expression datasets can lead to several issues in machine learning model development and deployment. First, it can affect the accuracy and reliability of predictive models, as the models may not effectively capture the underlying patterns in the data due to the differences in data distributions. For example, a model trained on one dataset may not perform well when applied to a different dataset with distinct data characteristics, resulting in poor generalisation performance. This can limit the model’s ability to provide accurate predictions and hinder its applicability across different datasets or experimental conditions.

Second, covariate shift can also result in biased model predictions, as the model may learn to rely on features that are specific to a particular dataset rather than capturing

the true biological signals of interest. This can lead to overfitting or misleading results, as the model may not accurately reflect the underlying biology of the system being studied. For instance, if a model is trained on a dataset with a specific batch effect, it may learn to rely on those batch-specific features, which may not be relevant or informative in other datasets.

To mitigate the impact of covariate shift in gene expression datasets, several strategies can be employed. One approach I took was to carefully pre-process and normalise the data to minimise the effects of technical variation, such as batch effects, before training machine learning models. I did this by applying ComBat, a batch correction method, to adjust for batch effects and harmonise the data across different datasets.

In order to mitigate the issue of covariate shift within gene expression datasets, I consider it imperative to standardise protocols and experimental conditions associated with RNA-Seq. Inconsistent procedures between research groups can introduce substantial variation in gene expression data, which poses challenges for machine learning models that thrive on consistency for accurate predictions. Standardisation would involve maintaining uniformity across a multitude of factors, including sample collection methods, sample handling, sequencing protocols, and even data analysis techniques. This unified approach could significantly reduce the variability across datasets, ensuring the expression levels of marker genes remain consistent and within a range the models have been trained on. As a result, this would enhance the robustness of machine learning models like ChronoGauge, making them more resilient when faced with new datasets and consequently improving the accuracy and reliability of their predictions. Another benefit of this would be that more datasets could be combined for meta-analysis as well as creating larger training datasets for machine learning and deep learning models to facilitate biological discoveries.

In summation, while ChronoGauge has demonstrated considerable potential in its ability to accurately predict across varying gene expression datasets, it is not devoid of challenges. The substantial variability in marker gene expression levels across different datasets, coupled with the problem of covariate shifts, may potentially

undermine the model's performance on future datasets that significantly deviate from its training data. However, these challenges can be mitigated. By carefully pre-processing and normalising data, it is possible to minimise the impact of technical variations. Furthermore, standardising RNA-Seq protocols—from sample collection and handling to sequencing procedures and data analysis—can greatly reduce dataset variability, thereby enhancing the robustness and reliability of machine learning models like ChronoGauge. Despite these hurdles, the potential of ChronoGauge remains clear. With continued methodological refinement and application of robust strategies to manage dataset variability, ChronoGauge is poised to facilitate advancements in our understanding of biological processes.

4.5.3 Identification of Circadian Clock Biomarkers

The identification of predictive circadian time biomarkers is a critical facet of ChronoGauge's predictive capabilities, one that distinguishes it from other methodologies such as ZeitZeiger. It is noteworthy that my approach employs a combination of filter and wrapper feature selection methods, which arguably offers a comprehensive analysis of rhythmically expressed genes and their combined predictive power with respect to circadian phase.

Comparatively, traditional methods have largely relied on statistical strategies, such as cosine fitting, to unearth rhythmically expressed genes. This approach, whilst effective, may inadvertently overlook potential circadian biomarkers that do not adhere to the expected sinusoidal pattern of gene expression. A prime example of an alternate strategy is encapsulated by ZeitZeiger, which instead employs a dimensionality reduction approach using sparse PCA. While this harnesses the power of unsupervised learning to identify potentially relevant features, it is subject to some limitations. For instance, it may struggle to identify relevant markers if the dominant patterns of variation in the data do not directly relate to the circadian rhythm.

ChronoGauge's feature selection strategy, on the other hand, incorporates an element of informed univariate curation in the filter stage, followed by a multivariate wrapper

approach. The filter stage allows for an initial broad selection of potential biomarkers, while the wrapper stage sequentially optimises this selection whilst maintaining diversity in phase of the biomarkers, facilitating the reduction of dimensionality without sacrificing the interpretability of the selected features.

A fascinating observation from this study is that ChronoGauge's most effective biomarkers were not the canonical core clock genes, traditionally regarded as the primary drivers of circadian rhythms. This suggests that the wrapper-based sequential feature selection strategy may have the capacity to identify novel genes associated with circadian regulation, potentially enriching our understanding of circadian biology. These newly identified genes could represent additional layers of complexity in the regulation of the circadian clock, which might have been previously overlooked when focusing solely on core clock genes. Furthermore, these genes could also potentially be involved in other biological processes that exhibit circadian variation, indicating an avenue for future research.

In conclusion, ChronoGauge's method of identifying circadian time biomarkers, which relies on a combination of filter and wrapper feature selection methods, offers a distinct and potentially more encompassing approach to capturing the complexity of circadian biology. The capability to identify non-traditional biomarkers opens the door to a broader understanding of circadian processes and, potentially, to the discovery of novel targets for circadian-based interventions.

4.5.4 Applications of ChronoGauge

ChronoGauge is highly proficient in accurately predicting the circadian time, yielding it considerable promise across a broad spectrum of applications ranging from basic research to practical, real-world uses. For example, in the realm of plant research, ChronoGauge could facilitate and accelerate the study of plant circadian rhythms. The ability to predict the internal circadian phase of a plant using single transcriptomic timepoint data offers a significant advancement over traditional methods, which generally rely on extensive time-series data. These factors together could reduce the need for exhaustive time-series experiments, saving time and

resources, as well as unlock novel insights into how plant behaviour and physiology are shaped by their internal clocks.

By understanding the circadian rhythms of crops, vertical farms could optimise their practices, such as watering, light exposure, and temperature to align with the plants' natural cycles. This could potentially improve yields and efficiency, making vertical farming agriculture more productive. ChronoGauge's ability to detect anomalies in circadian rhythms could also serve as an early warning system for diseases or stress conditions that disrupt these rhythms in crops, allowing for earlier interventions.

In the domain of chronotherapy, therapeutics, and medicine, ChronoGauge could help tailor treatments to patients' individual circadian rhythms, enhancing the effectiveness of medication and minimising side effects. This would be a significant step towards personalised medicine, as the timing of medication delivery can significantly impact its efficacy.

Finally, the application of ChronoGauge could extend to genome-wide association studies (GWAS). By predicting circadian phenotypes across a diversity panel of plants, it could reveal links between genetic variations and circadian traits. This information could then be leveraged to breed plants with desirable circadian traits, such as those that can adapt to changing climate conditions or exhibit enhanced productivity.

4.5.5 Conclusion

Reflecting on the ChronoGauge project, I am pleased to conclude that it has largely achieved its aims and objectives, despite encountering some limitations. Notably, I succeeded in developing an efficient and user-friendly predictive model for circadian time in plants, utilising both machine learning and statistical methodologies applied to gene expression data.

Fulfilling my primary objective, I developed a comprehensive Python software tool for circadian time prediction in plants. This tool, designed to process high-

throughput gene expression data from a variety of plant species and datasets, has been made freely available as an open-source resource. I believe this represents a valuable contribution to the toolkit of the scientific community.

My second objective was also realised, as I managed to identify predictive circadian biomarkers. By employing feature selection techniques, such as sequential feature selection, and rhythmicity analysis algorithms like ARSER, JTKCycle, and MetaCycle on multiple gene expression datasets, I identified genes and gene sets that consistently demonstrate rhythmic expression patterns. This work underscored the most informative genes for circadian time prediction and I think will serve as a useful basis for future research.

In relation to my third objective, I successfully developed and optimised machine learning and statistical models for circadian time prediction. My use of advanced techniques like neural networks and circular regression, and my evaluation of their performance through cross-validation and metrics such as mean absolute error, enabled the identification of powerful models for circadian time prediction.

The fourth objective, to test these predictive models across various datasets and plant species under different experimental conditions, was met with encouraging results as the analysis of the Blair dataset showed that the CCA1 mutation has a perturbed circadian rhythm. My developed models demonstrated robustness, accuracy, and generalisability across different biological contexts and datasets, as shown when making predictions on different tissue types and ecotypes such as Ws-2, which is evidence of their broad applicability and potential for transferability.

I am also aware of the limitations of this study. The relatively small datasets I utilised may pose challenges to the robustness and generalisability of the findings, as I cannot have reasonable confidence that ChronoGauge would outperform ZeitZeiger or other methods across a larger sample size. Additionally, the absence of readily available open-source packages for multioutput circular models necessitated the development of my own in Python. While this was ultimately successful, it introduced an additional layer of complexity to the project.

In conclusion, despite these limitations, the ChronoGauge project has successfully developed a robust and accurate predictive model for circadian time. This work not only advances our understanding of plant circadian clocks, but also provides a valuable tool for circadian research in plants. I am confident that, with further refinement and by addressing these limitations, ChronoGauge holds significant promise for a wide range of applications, from basic research in plant biology to practical uses in agriculture and chronotherapy.

4.6 Future Research

Methods of predicting the circadian time using gene expression biomarkers are still evolving, and there are several exciting avenues for future research that can build upon the work I have done to train models for predicting the circadian time in plants.

While research may have focused on a specific plant species, extending the validation of the predictive models to other plant species can provide valuable insights into the conservation and diversity of circadian regulation. This can help in identifying common circadian biomarkers that can be used across different plant species and elucidating species-specific differences in circadian regulation.

Gene regulatory networks play a crucial role in circadian regulation, and understanding the interactions among different genes and their regulatory elements can provide deeper insights into the circadian clock. Future research could involve the integration of gene expression data with other types of omics data, such as transcriptomics, proteomics, and epigenomics, to construct comprehensive gene regulatory networks that capture the complex dynamics of circadian regulation. Machine learning algorithms, such as network inference methods, can be used to identify key regulatory interactions and biomarkers that can accurately predict circadian time.

Circadian regulation is known to be influenced by various environmental and developmental factors, such as light, temperature, and age. Future research could investigate how these factors modulate the circadian clock and affect the predictive accuracy of circadian biomarkers. For example, studying the effects of different light regimes or temperature conditions on the circadian clock and incorporating these factors into machine learning models can help in developing more robust and adaptable predictive models for circadian time.

While my research may have focused on plants, the ChronoGauge can potentially be applied to other organisms as well, such as animals, fungi, and bacteria, where circadian regulation also plays a crucial role. Future research can involve translating

the predictive models to other organisms and investigating their applicability in diverse biological systems. Additionally, the predictive models can be further extended to other applications beyond circadian time prediction, such as drug discovery, disease diagnosis, and precision medicine, where circadian regulation has been implicated.

In my research, I have developed open-source Python software for circadian time prediction. Future research can focus on further refining and optimising the software, making it more user-friendly and accessible to the broader scientific community. This can include developing graphical user interfaces, providing documentation and tutorials, and incorporating feedback from users to continuously improve the software.

In conclusion, the research area of predicting circadian time using biomarkers offers significant potential for future research. Building upon the work I have done, further research can involve validation across different plant species, investigation of gene regulatory networks, exploration of environmental and developmental factors, application of advanced machine learning techniques, translation to other organisms and applications, and development of user-friendly software. These research directions can contribute to a deeper understanding of circadian regulation and enable the development of practical applications in various fields. I am satisfied knowing that Mr Connor Reynolds will be exploring these research areas and developing the ChronoGauge system as part of his PhD at the Earlham Institute.

Chapter 5: Discussion and Perspectives

The pursuit of using machine learning and computer vision to understand complex biological systems and their interaction with a changing environment has driven my research. Motivated by the urgency of global challenges, my work aims to address the need for increased and sustainable agricultural productivity in the face of a growing population and climate change. Additionally, the rapid advancement of high-throughput technologies has generated vast quantities of complex data, necessitating the development of cutting-edge computational methods to efficiently decode this information. With these needs and opportunities clear in my mind, my research takes an interdisciplinary approach, applying machine learning and computer vision methods to the field of plant biology.

At the core of my interdisciplinary research are three distinct projects, each tailored to tackle specific challenges within plant biology. The SeedGerm project focuses on harnessing computer vision to automate seed germination detection and facilitate the analysis of additional seed phenotypic traits. By optimising seed quality and growth conditions, this project has the potential to significantly enhance crop productivity, a critical factor in ensuring global food security.

The second project, Trans-Learn, explores the potential of transforming tabular gene expression datasets into image format to exploit vision-based methods capable of predicting turnip mosaic virus (TuMV) infections and other complex traits. Through innovative approaches, I seek to improve disease prediction, accelerating the development of plant varieties resistant to diseases and climate change. The ability to identify biomarkers associated with plant pathogen infections could help mitigate the impact of these diseases on agricultural productivity. Moreover, this method can be applied to predict a wide range of complex traits, offering opportunities to accelerate the development of improved crop varieties.

Lastly, the ChronoGauge project probes the possibility of predicting a plant’s internal circadian time using gene expression data. Through this, it could be possible to harness our understanding of the circadian clock to improve crop yields and stress

tolerance, thereby promoting a more resilient agricultural system. Similar to the Trans-Learn project, the prediction of the circadian clock represents an example of how complex traits can be harnessed to improve crop varieties.

In each of these projects, I have strived not just to address individual challenges, but also to create open-source tools that can facilitate further research. By intersecting plant biology, machine learning, and computer vision, my research is positioned to accelerate discoveries and innovation in plant biology, offering novel methodologies and insights. I hope that the tools and knowledge derived from my work will contribute to addressing critical global issues related to food security, climate change, and sustainable agriculture.

To realise the full potential of my research tools and methods, I have plans to create impact through an agritech spinout company. By establishing this company, I aim to translate the findings into practical applications and solutions for the agricultural industry. I will focus on developing innovative services that leverage the advancements that I have made during my thesis in biomarker identification and complex trait prediction. By doing so, I seek to make a tangible and meaningful contribution to agriculture, benefiting breeders and the broader agricultural community.

From the start of my PhD, the objective of the SeedGerm project was to address limitations in the traditional methodologies of seed imaging and scoring, focusing on a scalable and automated approach to seed germination analysis. I developed the software component of the SeedGerm system that serves as a fusion of cost-effective hardware and user-friendly software, capable of performing automated seed imaging and machine learning-based analyses.

To validate the system, SeedGerm was applied across a variety of germination experiments involving five distinct crop species. The results of these extensive tests demonstrated the system’s efficiency in quantitatively assessing seed batches and measuring both basic and complex morphological traits. By providing insights into seed size, width, length, as well as more complex phenotypes like extent and

circularity, SeedGerm enables a deeper understanding of the physiological intricacies of seed germination, which was a fundamental objective of the study.

A pivotal accomplishment of the SeedGerm project was the system's alignment with seed specialists' observations in scoring germination timing and rate. By matching the accuracy of seed technicians, SeedGerm validated the original hypothesis of the project. Additionally, SeedGerm played a crucial role in a biological discovery by identifying an ABA signalling gene in seeds through associative transcriptomics, in collaboration with the JIC.

The implications of the SeedGerm project are far-reaching, particularly for plant biology research and studies focusing on environmental and genetic factors affecting germination traits. The system's proficiency in providing reliable quantitative assessments of seed batches can expedite the process of evaluating the impact of different genetic and environmental factors on seed germination. By removing the human error factor and increasing the analysis speed, SeedGerm can contribute significantly to the efficiency of research projects and experiments for seed testing labs and plant breeding programs, allowing more accurate and informed decisions about seed viability and quality.

Although SeedGerm has demonstrated its robustness and utility, there are potential avenues for improvement and expansion that can be explored to enhance the system's effectiveness. One area for future development lies in integrating multispectral image analysis into the system. The project focused on RGB image capture, leaving potential spectral features indicative of germination status or seed vigour unexplored. This integration could enhance SeedGerm's predictive capabilities and offer a more comprehensive understanding of seed germination.

Another area that requires further exploration is the overlapping of radicles in later stages of seed germination. The increased complexity of images as germination progresses currently hampers the system's ability to accurately segment individual seeds and their roots. Advanced algorithms and techniques could potentially resolve this challenge, improving the precision of extracted morphological features and germination predictions.

Lastly, expanding the system's applicability to other species and traits presents a promising avenue for the future. While SeedGerm has proven its efficacy across five distinct crop species, exploring its application in other plant species, and possibly even in assessing root traits, could broaden SeedGerm's utility and relevance in plant biology research and agricultural practices.

Transitioning to the Trans-Learn project , a collaborative endeavour that developed during the course of my research. Here, I aimed to identify sets of multivariate biomarkers related to TuMV infection in *Arabidopsis halleri* by harnessing the capabilities of spatial arrangement of genes as well as image-based deep learning methodologies. The novelty of this approach lies in its transformation of tabular transcriptomic datasets into an image-like format, ideal for techniques such as CNNs and ViTs.

A significant breakthrough was the development of effective feature encoding methods that emphasised complex patterns and relationships between genes, providing the deep learning models with highly predictive spatial features. These methods, due to their efficacy, outperforming similar published methods, and wide applicability across gene expression and tabular datasets, stand as an essential asset for the scientific community. Also, by implementing feature interpretation techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM) and developing my own metric to quantify the multivariate nature of dependencies between biomarkers, I was able to identify sets of jointly dependent marker genes for TuMV and other pathogens.

The success of the Trans-Learn project has strong implications for the potential of image analysis methods in the realm of gene expression datasets. The ability to transform tabular data into image-like formats opens up opportunities for deploying sophisticated image analysis techniques on these datasets, which could lead to the discovery of novel patterns and relationships. It leads us towards a future where we can leverage the power of machine learning techniques to identify multivariate biomarkers, leading to advancements in the understanding and prediction of complex biological phenomena.

The potential future applications of Trans-Learn are significant, with opportunities for its expansion to identify biomarkers associated with other traits. The encoding and feature interpretation techniques developed in this project can be adapted and applied to other datasets, enabling researchers to identify critical biomarkers linked to a wide array of traits and conditions. Furthermore, the development of open-source Python software during this project means the Trans-Learn platform is accessible and reproducible, ready to be employed by the wider scientific community. Researchers across diverse biological contexts can utilise this platform to analyse gene expression dynamics and predict different traits, thereby expanding the horizons of its application.

Nevertheless, challenges such as the biological interpretation of identified genes, integration of multi-omics data, and performance on smaller datasets offer exciting avenues for future research. Addressing these challenges can enhance the capabilities of Trans-Learn, providing a more comprehensive and efficient tool for the scientific community. As I continue refining and expanding Trans-Learn, it can potentially serve as a powerful tool for predictive purposes and for deepening our understanding of gene expression dynamics in a myriad of biological contexts.

Finally, the ChronoGauge project's primary achievement was the successful creation of a predictive model for circadian time in plants, relying on both machine learning and statistical methodologies applied to gene expression data. My initial goal was to construct a comprehensive Python software tool for circadian time prediction in plants, and this has been achieved. The tool has undergone rigorous validation across various plant species gene expression datasets and is now available as an open-source toolkit available to the scientific community.

I managed to identify predictive circadian biomarkers, fulfilling another of my objectives. Utilising feature selection techniques such as sequential feature selection, coupled with rhythmicity analysis algorithms like ARSER, JTKCycle, MetaCycle, and relative amplitude error, I identified genes and gene sets consistently demonstrating rhythmic expression patterns.

I also successfully developed and optimised machine learning and statistical models for circadian time prediction, employing advanced techniques like neural networks in **tensorflow** and coding my own multi-output SGD circular regression. Performance evaluation through cross-validation and metrics such as mean absolute error led to the identification of robust models for circadian time prediction. Analysis of the Blair dataset substantiated the robustness, accuracy, and generalisability of these models across different biological contexts and datasets, as shown by their successful application on various tissue types and ecotypes such as Ws-2.

The ability to predict the circadian time of plants accurately can have far-reaching implications for research in plant biology. It aids in better understanding the functioning of plant circadian clocks, which play a critical role in regulating plant physiological processes. This can potentially provide new insights into how environmental factors and genetic mutations impact clock function. Moreover, having a reliable predictive model for circadian time can be useful for studies aiming to optimise plant growth and productivity in different environmental conditions. It could be leveraged to determine the optimal time for application of agricultural practices and therapeutics, leading to improvements in yield and disease management. In a broader sense, due to ChronoGauge's generalisable method, this research can offer potential insights for chronobiology research beyond the realm of plant science.

While the results of the ChronoGauge project are promising, there is ample room for refinement and expansion. The datasets used were relatively small, which may impact the generalisability of the findings. As such, future work should focus on testing and refining the tool using larger and more diverse datasets to improve the robustness and accuracy of the predictive models. A possibility for future development would be to explore the integration of additional layers of omics data to the existing gene expression data. This could help unveil even more precise predictors of circadian time and provide more comprehensive insights into the underlying biological processes. Addressing the project's current limitations and expanding its scope of application, ChronoGauge can potentially evolve into a powerful tool for plant biology research and its practical applications in agriculture and chronotherapy. I am confident that my colleague, Mr Connor Reynolds will

continue to develop ChronoGauge, improving its generalisability and broadening its applications through his PhD research.

The collective impact of the SeedGerm, Trans-Learn, and ChronoGauge projects is substantial, as they collectively advance the interdisciplinary approach of integrating computer vision, machine learning, and genomics in plant biology. These projects demonstrate the immense potential of leveraging cutting-edge technologies and methodologies to address critical challenges and open up new avenues of research in the field.

The analysis of large biological datasets poses unique challenges, and the open-source tools developed in this thesis provide valuable solutions. They offer user-friendly interfaces, comprehensive functionalities, and efficient data processing capabilities, making them accessible to researchers with varying levels of computational expertise.

The successful development of these tools in three distinct projects has enhanced knowledge exchange and collaboration within the scientific community. By enabling worldwide access and adaptability to these resources, it facilitates progress in plant biology research and fosters a culture of open science. Furthermore, the open-source nature of these tools underpins a framework that promotes collaboration, transparency, and reproducibility in scientific research. This communal approach enhances both the efficiency and accuracy of analyses, cultivating innovation through the integration of diverse perspectives and expertise.

This work underscores the vast potential of interdisciplinary research in tackling key challenges within plant biology. By synergising computer vision, machine learning, and genomics, I have demonstrated the potential to enhance our understanding of seed germination, gene expression dynamics, and circadian clocks in plants. The generalisability of the Trans-Learn and ChronoGauge methods harbours a vast potential for biomarker discovery and predictive modelling for complex traits.

In essence, the research conducted in this thesis carries considerable implications for agriculture, specifically in addressing complex traits and enhancing crop

productivity. As I venture into the next stage of my career, I intend to unlock the potential of biomarker discovery and predictive modelling, ultimately translating these theoretical explorations into tangible applications for the agricultural industry. This pursuit holds promise to significantly transform the agricultural sector and contribute towards achieving global food security.

Chapter 6: References

1. Kumar, R., Khurana, A. & Sharma, A. K. Role of plant hormones and their interplay in development and ripening of fleshy fruits. *J Exp Bot* **65**, 4561–4575 (2014).
2. Atkinson, N. J., Lilley, C. J. & Urwin, P. E. Identification of Genes Involved in the Response of Arabidopsis to Simultaneous Biotic and Abiotic Stresses 1[C][W][OPEN]. doi:10.1104/pp.113.222372.
3. Ruberti, I. *et al.* Plant adaptation to dynamically changing environment: The shade avoidance response. *Biotechnol Adv* **30**, 1047–1058 (2012).
4. Taiz, L., Zeiger Eduardo & Murph, A. S. *Plant Physiology And Development*. (Sinauer Associates, 2017).
5. Mu, X. & Chen, Y. The physiological response of photosynthesis to nitrogen deficiency. *Plant Physiology and Biochemistry* **158**, 76–82 (2021).
6. Suzuki, N., Rivero, R. M., Shulaev, V., Blumwald, E. & Mittler, R. Abiotic and biotic stress combinations. *New Phytologist* **203**, 32–43 (2014).
7. Vara Prasad, P. V *et al.* Crop Production under Drought and Heat Stress: Plant Responses and Management Options. *Front. Plant Sci* **8**, 1147 (2017).
8. Weigel, D. & Mott, R. The 1001 genomes project for Arabidopsis thaliana. *Genome Biol* **10**, 107 (2009).
9. Oliva, R., Ji, C., Atienza-Grande, G., Huguet-Tapia, J. C. & Perez-Quintero, A. Broad-spectrum resistance to bacterial blight in rice using genome editing. *Nat Biotechnol* doi:10.1038/s41587-019-0267-z.

10. Xu, L. *et al.* Genome-wide association study reveals the genetic architecture of flowering time in rapeseed (Brassica napus L.). *DNA Res* **23**, 43 (2016).
11. Atwell, S. *et al.* Genome-wide association study of 107 phenotypes in Arabidopsis thaliana inbred lines. *Nature* (2010) doi:10.1038/nature08800.
12. Shan-e-Ali Zaidi, S., Mahas, A., Vanderschuren, H. & Mahfouz, M. M. Engineering crops of the future: CRISPR approaches to develop climate-resilient and disease-resistant plants. doi:10.1186/s13059-020-02204-y.
13. Iqbal, S. *et al.* Phytohormones Trigger Drought Tolerance in Crop Plants: Outlook and Future Perspectives. *Front Plant Sci* **12**, 3378 (2022).
14. Yao, W. *et al.* VpPUB24, a novel gene from Chinese grapevine, Vitis pseudoreticulata, targets VpICE1 to enhance cold tolerance. *J Exp Bot* **68**, 2933–2949 (2017).
15. Janni, M. *et al.* Molecular and genetic bases of heat stress responses in crop plants and breeding for increased resilience and productivity. *J Exp Bot* **71**, 3780–3802 (2020).
16. Somerville, C. & Koornneef, M. A fortunate choice: the history of Arabidopsis as a model plant. *Nat Rev Genet* **3**, 883–889 (2002).
17. Kaul, S. *et al.* Analysis of the genome sequence of the flowering plant Arabidopsis thaliana. *Nature 2000 408:6814* **408**, 796–815 (2000).
18. Koornneef, M. & Meinke, D. The development of Arabidopsis as a model plant. *Plant J* **61**, 909–921 (2010).
19. Soltis, P. S. & Soltis, D. E. The role of genetic and genomic attributes in the success of polyploids. *Proc Natl Acad Sci U S A* **97**, 7051 (2000).
20. Honjo, M. N. & Kudoh, H. Arabidopsis halleri: a perennial model system for studying population differentiation and local adaptation. *AoB Plants* **11**, (2019).
21. Tester, M. & Langridge, P. Breeding technologies to increase crop production in a changing world. *Science (1979)* **327**, 818–822 (2010).

22. Hake, S. & Ross-Ibarra, J. Genetic, evolutionary and plant breeding insights from the domestication of maize. *Elife* doi:10.7554/eLife.05861.001.
23. He, Y. *et al.* GWAS, QTL mapping and gene expression analyses in Brassica napus reveal genetic control of branching morphogenesis OPEN. doi:10.1038/s41598-017-15976-4.
24. Jiang, W. *et al.* Demonstration of CRISPR/Cas9/sgRNA-mediated targeted gene modification in Arabidopsis, tobacco, sorghum and rice. doi:10.1093/nar/gkt780.
25. Wu, J. *et al.* Expression of trehalose-6-phosphate phosphatase in maize ears improves yield in well-watered and drought conditions. *Nat Biotechnol* **33**, (2015).
26. Bouis, H. E., Hotz, C., McClafferty, B., Meenakshi, J. V & Pfeiffer, W. H. Biofortification: A new tool to reduce micronutrient malnutrition. *Food Nutr. Bull.* **32**, S31–S40 (2011).
27. White, P. J. & Broadley, M. R. Biofortification of crops with seven mineral elements often lacking in human diets--iron, zinc, copper, calcium, magnesium, selenium and iodine. *New Phytol* **182**, 49–84 (2009).
28. Ortiz-Monasterio, J. I. *et al.* Enhancing the mineral and vitamin content of wheat and maize through plant breeding. *J Cereal Sci* **46**, 293–307 (2007).
29. Strange, R. N. & Scott, P. R. Plant Disease: A Threat to Global Food Security. *Annu. Rev. Phytopathol* **43**, 83–116 (2005).
30. Cabello, J. V., Lodeyro, A. F. & Zurbriggen, M. D. Novel perspectives for the engineering of abiotic stress tolerance in plants. *Curr Opin Biotechnol* **26**, 62–70 (2014).
31. Collard, B. C. Y. & Mackill, D. J. Marker-assisted selection: an approach for precision plant breeding in the twenty-first century. *Philosophical Transactions of the Royal Society B: Biological Sciences* **363**, 557–572 (2007).

32. Varshney, R. K., Hoisington, D. A. & Tyagi, A. K. Advances in cereal genomics and applications in crop breeding. *Trends Biotechnol* **24**, 490–499 (2006).

33. Xu, Y. & Crouch, J. H. Marker-Assisted Selection in Plant Breeding: From Publications to Practice. *Crop Sci* **48**, 391–407 (2008).

34. Collard, B. C. Y., Jahufer, M. Z. Z., Brouwer, J. B. & Pang, E. C. K. An introduction to markers, quantitative trait loci (QTL) mapping and marker-assisted selection for crop improvement: The basic concepts. *Euphytica 2005 142:1* **142**, 169–196 (2005).

35. Hospital, F. Challenges for effective marker-assisted selection in plants. *Genetica* **136**, 303–310 (2009).

36. Meuwissen, T. H. E., Hayes, B. J. & Goddard, M. E. Prediction of total genetic value using genome-wide dense marker maps. *Genetics* **157**, 1819–1829 (2001).

37. Wang, X., Xu, Y., Hu, Z. & Xu, C. Genomic selection methods for crop improvement: Current status and prospects. *Crop J* **6**, 330–340 (2018).

38. Bewley, J. D. Seed Germination and Dormancy. *Plant Cell* **9**, 1055–1066 (1997).

39. Finch-Savage, W. E. & Leubner-Metzger, G. Seed dormancy and the control of germination. *New Phytologist* **171**, 501–523 (2006).

40. Finkelstein, R., Reeves, W., Ariizumi, T. & Steber, C. Molecular aspects of seed dormancy. *Annu Rev Plant Biol* **59**, 387–415 (2008).

41. Penfield, S. & MacGregor, D. R. Effects of environmental variation during seed production on seed dormancy and germination. *J Exp Bot* **68**, erw436 (2017).

42. Shu, K., Liu, X. D., Xie, Q. & He, Z. H. Two Faces of One Seed: Hormonal Regulation of Dormancy and Germination. *Mol Plant* **9**, 34–45 (2016).

43. Gibson, L. R. & Mullen, R. E. Soybean Seed Quality Reductions by High Day and Night Temperature. *Crop Sci* **36**, 1615–1619 (1996).

44. Marcos-Filho, J. Seed vigor testing: an overview of the past, present and future perspective. *Sci Agric* **72**, 363–374 (2015).

45. Rajjou, L. *et al.* Seed Germination and Vigor. *https://doi.org/10.1146/annurev-arplant-042811-105550* **63**, 507–533 (2012).

46. Jacobsen, S. E. & Bach, A. P. The influence of temperature on seed germination rate in quinoa (Chenopodium quinoa Willd. *Seed Science and Technology (Switzerland)* **26**, 515–523 (1998).

47. Finch-Savage, W. E. & Bassel, G. W. Seed vigour and crop establishment: extending performance beyond adaptation. *J Exp Bot* **67**, 567–591 (2016).

48. Dürr, C., Dickie, J. B., Yang, X. Y. & Pritchard, H. W. Ranges of critical temperature and water potential values for the germination of species worldwide: Contribution to a seed trait database. *Agric For Meteorol* **200**, 222–232 (2015).

49. Ram Lamichhane, J. *et al.* Abiotic and biotic factors affecting crop seed germination and seedling emergence: a conceptual framework. *Plant and Soil 2018 432:1* **432**, 1–28 (2018).

50. Mwale, S. S., Hamusimbi, C. & Mwansa, K. Germination, emergence and growth of sunflower (Helianthus annuus L.) in response to osmotic seed priming. *Seed Science and Technology* **31**, 199–206 (2003).

51. Jones, J. D. G. & Dangl, J. L. The plant immune system. *Nature 2006 444:7117* **444**, 323–329 (2006).

52. Dodds, P. N. & Rathjen, J. P. Plant immunity: towards an integrated view of plant–pathogen interactions. *Nature Reviews Genetics 2010 11:8* **11**, 539–548 (2010).

53. Zipfel, C. Plant pattern-recognition receptors. *Trends Immunol* **35**, 345–351 (2014).

54. Chisholm, S. T., Coaker, G., Day, B. & Staskawicz, B. J. Host-microbe interactions: shaping the evolution of the plant immune response. *Cell* **124**, 803–814 (2006).
55. Thrall, P. H. *et al.* Rapid genetic change underpins antagonistic coevolution in a natural host-pathogen metapopulation. *Ecol Lett* **15**, 425–435 (2012).
56. Weiberg, A. *et al.* Fungal small RNAs suppress plant immunity by hijacking host RNA interference pathways. *Science (1979)* **342**, 118–123 (2013).
57. Zhang, X. *et al.* Arabidopsis Argonaute 2 regulates innate immunity via miRNA393(∗)-mediated silencing of a Golgi-localized SNARE gene, MEMB12. *Mol Cell* **42**, 356–366 (2011).
58. Zhou, J. M. & Zhang, Y. Plant Immunity: Danger Perception and Signaling. *Cell* **181**, 978–989 (2020).
59. Poland, J. A., Balint-Kurti, P. J., Wisser, R. J., Pratt, R. C. & Nelson, R. J. Shades of gray: the world of quantitative disease resistance. *Trends Plant Sci* **14**, 21–29 (2009).
60. Alcázar, R. & Parker, J. E. The impact of temperature on balancing immune responsiveness and growth in Arabidopsis. *Trends Plant Sci* **16**, 666–675 (2011).
61. Boller, T. & Felix, G. A renaissance of elicitors: perception of microbe-associated molecular patterns and danger signals by pattern-recognition receptors. *Annu Rev Plant Biol* **60**, 379–407 (2009).
62. Dangl, J. L., Horvath, D. M. & Staskawicz, B. J. Pivoting the Plant Immune System from Dissection to Deployment. *Science* **341**, 746–751 (2013).
63. Brown, J. K. M., Chartrain, L., Lasserre-Zuber, P. & Saintenac, C. Genetics of resistance to Zymoseptoria tritici and applications to wheat breeding. *Fungal Genetics and Biology* **79**, 33–41 (2015).

64. Wiesel, L. *et al.* Molecular effects of resistance elicitors from biological origin and their potential for crop protection. *Front Plant Sci* **5**, 655 (2014).
65. McClung, C. R. Plant circadian rhythms. *Plant Cell* **18**, 792–803 (2006).
66. Dodd, A. N., Belbin, F. E., Frank, A. & Webb, A. A. R. Interactions between circadian clocks and photosynthesis for the temporal and spatial coordination of metabolism. *Front Plant Sci* **6**, 1–7 (2015).
67. Hassidim, M. *et al.* CIRCADIAN CLOCK ASSOCIATED1 (CCA1) and the Circadian Control of Stomatal Aperture. *Plant Physiol* **175**, 1864 (2017).
68. Song, Y. H., Shim, J. S., Kinmonth-Schultz, H. A. & Imaizumi, T. Photoperiodic flowering: time measurement mechanisms in leaves. *Annu Rev Plant Biol* **66**, 441–464 (2015).
69. Harmer, S. L. *et al.* Orchestrated transcription of key pathways in Arabidopsis by the circadian clock. *Science* **290**, 2110–2113 (2000).
70. Greenham, K. & McClung, C. R. Integrating circadian dynamics with physiological processes in plants. *Nat Rev Genet* **16**, 598–610 (2015).
71. Gould, P. D. *et al.* The molecular basis of temperature compensation in the Arabidopsis circadian clock. *Plant Cell* **18**, 1177–1187 (2006).
72. Covington, M. F., Maloof, J. N., Straume, M., Kay, S. A. & Harmer, S. L. Global transcriptome analysis reveals circadian regulation of key pathways in plant growth and development. *Genome Biol* **9**, 1–18 (2008).
73. Bhardwaj, V., Meier, S., Petersen, L. N., Ingle, R. A. & Roden, L. C. Defence Responses of Arabidopsis thaliana to Infection by Pseudomonas syringae Are Regulated by the Circadian Clock. *PLoS One* **6**, e26968 (2011).

74. Wilkins, O., Bräutigam, K. & Campbell, M. M. Time of day shapes Arabidopsis drought transcriptomes. *The Plant Journal* **63**, 715–727 (2010).
75. Johansson, M. & Staiger, D. Time to flower: interplay between photoperiod and the circadian clock. *J Exp Bot* **66**, 719–730 (2015).
76. Dodd, A. N. *et al.* Plant circadian clocks increase photosynthesis, growth, survival, and competitive advantage. *Science* **309**, 630–633 (2005).
77. Nohales, M. A. & Kay, S. A. Molecular mechanisms at the core of the plant circadian oscillator. *Nat Struct Mol Biol* **23**, 1061–1069 (2016).
78. Wang, Z. Y. & Tobin, E. M. Constitutive expression of the CIRCADIAN CLOCK ASSOCIATED 1 (CCA1) gene disrupts circadian rhythms and suppresses its own expression. *Cell* **93**, 1207–1217 (1998).
79. Alabadí, D. *et al.* Reciprocal regulation between TOC1 and LHY/CCA1 within the Arabidopsis circadian clock. *Science* **293**, 880–883 (2001).
80. Varshney, R. K., Nayak, S. N., May, G. D. & Jackson, S. A. Next-generation sequencing technologies and their implications for crop genetics and breeding. *Trends Biotechnol* **27**, 522–530 (2009).
81. Wang, Z., Gerstein, M. & Snyder, M. RNA-Seq: a revolutionary tool for transcriptomics. *Nat Rev Genet* **10**, 57–63 (2009).
82. Marioni, J. C., Mason, C. E., Mane, S. M., Stephens, M. & Gilad, Y. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res* **18**, 1509–1517 (2008).
83. Ozsolak, F. & Milos, P. M. RNA sequencing: advances, challenges and opportunities. *Nature Reviews Genetics 2010 12:2* **12**, 87–98 (2010).
84. Marioni, J. C., Mason, C. E., Mane, S. M., Stephens, M. & Gilad, Y. RNA-seq: an assessment of technical reproducibility and comparison with gene expression arrays. *Genome Res* **18**, 1509–1517 (2008).

85. Trapnell, C. *et al.* Transcript assembly and quantification by RNA-Seq reveals unannotated transcripts and isoform switching during cell differentiation. *Nature Biotechnology 2010 28:5* **28**, 511–515 (2010).
86. Filichkin, S. A. *et al.* Genome-wide mapping of alternative splicing in Arabidopsis thaliana. *Genome Res* **20**, 45–58 (2010).
87. Yeaman, S. *et al.* Conservation and divergence of gene expression plasticity following c. 140 million years of evolution in lodgepole pine (Pinus contorta) and interior spruce (Picea glauca × Picea engelmannii). *New Phytologist* **203**, 578–591 (2014).
88. Phule, A. S. *et al.* RNA-seq reveals the involvement of key genes for aerobic adaptation in rice. *Sci Rep* **9**, (2019).
89. Dugas, D. V. & Bartel, B. Sucrose induction of Arabidopsis miR398 represses two Cu/Zn superoxide dismutases. *Plant Mol Biol* **67**, 403–417 (2008).
90. Zhang, H., Zhu, J., Gong, Z. & Zhu, J. K. Abiotic stress responses in plants. *Nat Rev Genet* **23**, 104–119 (2022).
91. Todaka, D., Nakashima, K., Shinozaki, K. & Yamaguchi-Shinozaki, K. Toward understanding transcriptional regulatory networks in abiotic stress responses and tolerance in rice. *Rice* **5**, 1–9 (2012).
92. Wils, C. R. & Kaufmann, K. Gene-regulatory networks controlling inflorescence and flower development in Arabidopsis thaliana. *Biochim Biophys Acta Gene Regul Mech* **1860**, 95–105 (2017).
93. Jordan, M. I. & Mitchell, T. M. Machine learning: Trends, perspectives, and prospects. *Science (1979)* **349**, 255–260 (2015).
94. Krizhevsky, A., Sutskever, I. & Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks.
95. Devlin, J., Chang, M. W., Lee, K. & Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference* **1**, 4171–4186 (2018).
96. Esteva, A. *et al.* Dermatologist-level classification of skin cancer with deep neural

networks. *Nature 2017 542:7639* **542**, 115–118 (2017).

97. Bengio, Y., Courville, A. & Vincent, P. Representation Learning: A Review and New Perspectives. *IEEE Trans Pattern Anal Mach Intell* **35**, 1798–1828 (2012).
98. Lecun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature 2015 521:7553* **521**, 436–444 (2015).
99. Schmidhuber, J. Deep Learning in Neural Networks: An Overview. *Neural Networks* **61**, 85–117 (2014).
100. Kamilaris, A. & Prenafeta-Boldu, F. X. Deep learning in agriculture: A survey. *Comput Electron Agric* **147**, 70–90 (2018).
101. Schneider, A., Hommel, G. & Blettner, M. Linear Regression Analysis: Part 14 of a Series on Evaluation of Scientific Publications. *Dtsch Arztebl Int* **107**, 776 (2010).
102. Sperandei, S. Understanding logistic regression analysis. *Biochem Med (Zagreb)* **24**, 12 (2014).
103. Breiman, L., Friedman, J. H., Olshen, R. A. & Stone, C. J. Classification and regression trees. *Classification and Regression Trees* 1–358 (2017) doi:10.1201/9781315139470/CLASSIFICATION-REGRESSION-TREES-LEO-BREIMAN.
104. Liaw, A. & Wiener, M. Classification and Regression by randomForest. **2**, (2002).
105. Friedman, J. H. Greedy function approximation: A gradient boosting machine. *https://doi.org/10.1214/aos/1013203451* **29**, 1189–1232 (2001).
106. Cover, T. M. & Hart, P. E. Nearest Neighbor Pattern Classification. *IEEE Trans Inf Theory* **13**, 21–27 (1967).
107. Cortes, C., Vapnik, V. & Saitta, L. Support-vector networks. *Machine Learning 1995 20:3* **20**, 273–297 (1995).
108. Schölkopf, B. SVMs - A practical consequence of learning theory. *IEEE Intelligent Systems and Their Applications* **13**, 18–21 (1998).

109. Uhrig, R. E. Introduction to artificial neural networks. *Proceedings of IECON '95 - 21st Annual Conference on IEEE Industrial Electronics* **1**, 33–37.
110. Dietterich, T. G. Ensemble methods in machine learning. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **1857 LNCS**, 1–15 (2000).
111. Wolpert, D. H. Stacked generalization. *Neural Networks* **5**, 241–259 (1992).
112. Freund, Y. & Schapire, R. E. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J Comput Syst Sci* **55**, 119–139 (1997).
113. Simonyan, K. & Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings* (2014).
114. Yang, P., Yang, Y. H., Zhou, B. B. & Zomaya, A. Y. A review of ensemble methods in bioinformatics: * Including stability of feature selection and ensemble feature selection methods (updated on 28 Sep. 2016).
115. Guyon, I. & De, A. M. An Introduction to Variable and Feature Selection André Elisseeff. *Journal of Machine Learning Research* **3**, 1157–1182 (2003).
116. Chandrashekar, G. & Sahin, F. A survey on feature selection methods. *Computers & Electrical Engineering* **40**, 16–28 (2014).
117. Saeys, Y., Inza, I. & Larrañaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* **23**, 2507–2517 (2007).
118. Bommert, A., Sun, X., Bischl, B., Rahnenführer, J. & Lang, M. Benchmark for filter methods for feature selection in high-dimensional classification data. *Comput Stat Data Anal* **143**, 106839 (2020).
119. Dash ', M. & Liu, H. Feature Selection for Classification. *IDA ELSEVIER Intelligent Data Analysis* **1**, 131–156 (1997).
120. Kohavi, R. & John, G. H. Wrappers for feature subset selection. *Artif Intell* **97**, 273–324 (1997).

121. Guyon, I., Weston, J., Barnhill, S. & Vapnik, V. Gene selection for cancer classification using support vector machines. *Mach Learn* **46**, 389–422 (2002).
122. Hastie, T., Tibshirani, R. & Friedman, J. The Elements of Statistical Learning. (2009) doi:10.1007/978-0-387-84858-7.
123. Cawley, G. C. & Talbot, N. L. C. On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation. *Journal of Machine Learning Research* **11**, 2079–2107 (2010).
124. Arlot, S. & Celisse, A. A survey of cross-validation procedures for model selection. *https://doi.org/10.1214/09-SS054* **4**, 40–79 (2010).
125. Varma, S. & Simon, R. Bias in error estimation when using cross-validation for model selection. *BMC Bioinformatics* **7**, 1–8 (2006).
126. Sechidis, K., Tsoumakas, G. & Vlahavas, I. On the stratification of multi-label data. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **6913 LNAI**, 145–158 (2011).
127. Bergstra, J., Ca, J. B. & Ca, Y. B. Random Search for Hyper-Parameter Optimization Yoshua Bengio. *Journal of Machine Learning Research* **13**, 281–305 (2012).
128. Bergstra, J., Bardenet, R., Bengio, Y. & Kégl, B. Algorithms for Hyper-Parameter Optimization. *Adv Neural Inf Process Syst* **24**, (2011).
129. Snoek, J., Larochelle, H. & Adams, R. P. Practical Bayesian Optimization of Machine Learning Algorithms. *Adv Neural Inf Process Syst* **4**, 2951–2959 (2012).
130. Fawcett, T. An introduction to ROC analysis. *Pattern Recognit Lett* **27**, 861–874 (2006).
131. Japkowicz, N. & Shah, M. Evaluating Learning Algorithms: A Classification Perspective. *Evaluating Learning Algorithms: A Classification Perspective* **9780521196000**, 1–406 (2011).
132. He, H. & Garcia, E. A. Learning from imbalanced data. *IEEE Trans Knowl Data Eng* **21**, 1263–1284 (2009).
133. Janocha, K. & Czarnecki, W. M. On Loss Functions for Deep Neural Networks in Classification. *Schedae Informaticae* **25**, 49–59 (2017).

134. Ruder, S. An overview of gradient descent optimization algorithms *.

135. Sutskever, I., Martens, J., Dahl, G. & Hinton, G. On the importance of initialization and momentum in deep learning. 1139–1147 Preprint at https://proceedings.mlr.press/v28/sutskever13.html (2013).

136. Duchi JDUCHI, J. & Singer, Y. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization * Elad Hazan. *Journal of Machine Learning Research* **12**, 2121–2159 (2011).

137. Kingma, D. P. & Ba, J. L. Adam: A Method for Stochastic Optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings* (2014).

138. Dosilovic, F. K., Brcic, M. & Hlupic, N. Explainable artificial intelligence: A survey. *2018 41st International Convention on Information and Communication Technology, Electronics and Microelectronics, MIPRO 2018 - Proceedings* 210–215 (2018) doi:10.23919/MIPRO.2018.8400040.

139. Ren, M. *et al.* Meta-Learning For Semi-Supervised Few-Shot Classification.

140. Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R. & Yu, B. Definitions, methods, and applications in interpretable machine learning. *Proc Natl Acad Sci U S A* **116**, 22071–22080 (2019).

141. Chen, D. *et al.* Dissecting the phenotypic components of crop plant growth and drought responses based on high-throughput image analysis. *Plant Cell* **26**, 4636–4655 (2014).

142. Compton, M. E. Statistical methods suitable for the analysis of plant tissue culture data. *Plant Cell Tissue Organ Cult* **37**, 217–242 (1994).

143. Korte, A. & Farlow, A. The advantages and limitations of trait analysis with GWAS: A review. *Plant Methods* **9**, 1–9 (2013).

144. Herranz, R. *et al.* RNAseq Analysis of the Response of Arabidopsis thaliana to Fractional Gravity Under Blue-Light Stimulation During Spaceflight. *Front Plant Sci* **10**, 1529 (2019).

145. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol* **15**, 1–21 (2014).

146. Burks, D. J., Sengupta, S., De, R., Mittler, R. & Azad, R. K. The Arabidopsis gene co-expression network. *Plant Direct* **6**, e396 (2022).
147. Langfelder, P. & Horvath, S. WGCNA: An R package for weighted correlation network analysis. *BMC Bioinformatics* **9**, 1–13 (2008).
148. Rasheed, A. *et al.* Crop Breeding Chips and Genotyping Platforms: Progress, Challenges, and Perspectives. *Mol Plant* **10**, 1047–1064 (2017).
149. Rudin, C. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence 2019 1:5* **1**, 206–215 (2019).
150. Lin, F., Fan, J. & Rhee, S. Y. QTG-Finder: A Machine-Learning Based Algorithm To Prioritize Causal Genes of Quantitative Trait Loci in Arabidopsis and Rice. *G3: Genes|Genomes|Genetics* **9**, 3129 (2019).
151. Lu, Y., Yi, S., Zeng, N., Liu, Y. & Zhang, Y. Identification of rice diseases using deep convolutional neural networks. *Neurocomputing* **267**, 378–384 (2017).
152. Bauer, A. *et al.* Combining computer vision and deep learning to enable ultra-scale aerial phenotyping and precision agriculture: A case study of lettuce production. *Hortic Res* **6**, 70 (2019).
153. Wang, Y., Dai, X., Fu, D., Li, P. & Du, B. PGD: a machine learning-based photosynthetic-related gene detection approach. *BMC Bioinformatics* **23**, 1–11 (2022).
154. Azodi, C. B., Pardo, J., VanBuren, R., de Los Campos, G. & Shiu, S. H. Transcriptome-Based Prediction of Complex Traits in Maize. *Plant Cell* **32**, 139 (2020).
155. Marchand, G. *et al.* A biomarker based on gene expression indicates plant water status in controlled and natural environments. *Plant Cell Environ* **36**, 2175–2189 (2013).
156. Ye, Z. *et al.* Biomarker Categorization in Transcriptomic Meta-Analysis by Concordant Patterns With Application to Pan-Cancer Studies. *Front Genet* **12**, 1079 (2021).

157. Nadeem, M. A. *et al.* DNA molecular markers in plant breeding: current status and recent advancements in genomic selection and genome editing. *http://mc.manuscriptcentral.com/tbeq* **32**, 261–285 (2017).
158. Liu, Y. *et al.* Proteomics: a powerful tool to study plant responses to biotic stress. *Plant Methods 2019 15:1* **15**, 1–20 (2019).
159. Rohr, A. D. *et al.* Identification and validation of early genetic biomarkers for apple replant disease. *PLoS One* **15**, e0238876 (2020).
160. Yan, H. *et al.* Research on Biomarkers of Different Growth Periods and Different Drying Processes of Citrus wilsonii Tanaka Based on Plant Metabolomics. *Front Plant Sci* **12**, 1443 (2021).
161. Melandri, G. *et al.* Biomarkers for grain yield stability in rice under drought stress. *J Exp Bot* **71**, 669–683 (2020).
162. Soltabayeva, A., Ongaltay, A., Omondi, J. O. & Srivastava, S. Morphological, Physiological and Molecular Markers for Salt-Stressed Plants. *Plants* **10**, 1–18 (2021).
163. Chitarrini, G. *et al.* Identification of biomarkers for defense response to Plasmopara viticola in a resistant grape variety. *Front Plant Sci* **8**, (2017).
164. Rushton, P. J., Somssich, I. E., Ringler, P. & Shen, Q. J. WRKY transcription factors. *Trends Plant Sci* **15**, 247–258 (2010).
165. Chen, L. *et al.* The role of WRKY transcription factors in plant abiotic stresses. *Biochim Biophys Acta* **1819**, 120–128 (2012).
166. Liu, X., Bai, X., Wang, X. & Chu, C. OsWRKY71, a rice transcription factor, is involved in rice defense response. *J Plant Physiol* **164**, 969–979 (2007).
167. Kakumanu, A. *et al.* Effects of Drought on Gene Expression in Maize Reproductive and Leaf Meristem Tissue Revealed by RNA-Seq. *Plant Physiol* **160**, 846–867 (2012).
168. Park, S. *et al.* Regulation of the Arabidopsis CBF regulon by a complex low-temperature regulatory network. *Plant J* **82**, 193–207 (2015).
169. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139 (2010).

170. van Dam, S., Võsa, U., van der Graaf, A., Franke, L. & de Magalhães, J. P. Gene co-expression analysis for functional classification and gene–disease predictions. *Brief Bioinform* **19**, 575–592 (2018).
171. Khatri, P., Sirota, M. & Butte, A. J. Ten Years of Pathway Analysis: Current Approaches and Outstanding Challenges. *PLoS Comput Biol* **8**, e1002375 (2012).
172. Singh, A. *et al.* DIABLO: an integrative approach for identifying key molecular drivers from multi-omics assays. *Bioinformatics* **35**, 3055–3062 (2019).
173. Cruz, J. A. & Wishart, D. S. Applications of Machine Learning in Cancer Prediction and Prognosis. *Cancer Inform* **2**, 59 (2006).
174. Huynh-Thu, V. A., Irrthum, A., Wehenkel, L. & Geurts, P. Inferring Regulatory Networks from Expression Data Using Tree-Based Methods. *PLoS One* **5**, e12776 (2010).
175. Eisen, M. B., Spellman, P. T., Brown, P. O. & Botstein, D. Cluster analysis and display of genome-wide expression patterns. *Proc Natl Acad Sci U S A* **95**, 14863–14868 (1998).
176. Tang, B. *et al.* Transcriptome data reveal gene clusters and key genes in pepper response to heat shock. *Front Plant Sci* **13**, 3142 (2022).
177. Ritchie, M. E. *et al.* limma powers differential expression analyses for RNA-sequencing and microarray studies. *Nucleic Acids Res* **43**, e47 (2015).
178. Noble, W. S. How does multiple testing correction work? *Nature Biotechnology 2009 27:12* **27**, 1135–1137 (2009).
179. Saccenti, E., Hoefsloot, H. C. J., Smilde, A. K., Westerhuis, J. A. & Hendriks, M. M. W. B. Reflections on univariate and multivariate analysis of metabolomics data. *Metabolomics* **10**, 361–374 (2014).
180. Saeys, Y., Inza, I. & Larrañaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* **23**, 2507–2517 (2007).
181. Xing, E. P., Jordan, M. I. & Karp, R. M. Feature Selection for High-Dimensional Genomic Microarray Data.

182. Lecun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature 2015 521:7553* **521**, 436–444 (2015).
183. O'Shea, K. & Nash, R. An Introduction to Convolutional Neural Networks. *Int J Res Appl Sci Eng Technol* **10**, 943–947 (2015).
184. Dosovitskiy, A. *et al.* An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. (2020).
185. Russakovsky, O. *et al.* ImageNet Large Scale Visual Recognition Challenge. *Int J Comput Vis* **115**, 211–252 (2015).
186. Lowe, D. G. Distinctive image features from scale-invariant keypoints. *Int J Comput Vis* **60**, 91–110 (2004).
187. Chen, C.-F., Fan, Q. & Panda, R. CrossViT: Cross-Attention Multi-Scale Vision Transformer for Image Classification.
188. He, K., Zhang, X., Ren, S. & Sun, J. Deep Residual Learning for Image Recognition. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **2016-December**, 770–778 (2015).
189. Tan, M. & Le, Q. V. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks. *36th International Conference on Machine Learning, ICML 2019* **2019-June**, 10691–10700 (2019).
190. Girshick, R., Donahue, J., Darrell, T. & Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* 580–587 (2013) doi:10.1109/CVPR.2014.81.
191. Redmon, J., Divvala, S., Girshick, R. & Farhadi, A. You Only Look Once: Unified, Real-Time Object Detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **2016-December**, 779–788 (2015).
192. Ren, S., He, K., Girshick, R. & Sun, J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *IEEE Trans Pattern Anal Mach Intell* **39**, 1137–1149 (2015).

193. Bochkovskiy, A., Wang, C.-Y. & Liao, H.-Y. M. YOLOv4: Optimal Speed and Accuracy of Object Detection. (2020).
194. Zhu, X., Lyu, S., Wang, X. & Zhao, Q. TPH-YOLOv5: Improved YOLOv5 Based on Transformer Prediction Head for Object Detection on Drone-captured Scenarios. *Proceedings of the IEEE International Conference on Computer Vision* **2021-October**, 2778–2788 (2021).
195. Shelhamer, E., Long, J. & Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans Pattern Anal Mach Intell* **39**, 640–651 (2014).
196. Weng, W. & Zhu, X. U-Net: Convolutional Networks for Biomedical Image Segmentation. *IEEE Access* **9**, 16591–16603 (2015).
197. Badrinarayanan, V., Kendall, A. & Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans Pattern Anal Mach Intell* **39**, 2481–2495 (2015).
198. Jiang, Y. & Li, C. Review Article Convolutional Neural Networks for Image-Based High-Throughput Plant Phenotyping: A Review. (2020) doi:10.34133/2020/4152816.
199. Dai, Y., Gieseke, F., Oehmcke, S., Wu, Y. & Barnard, K. Attentional Feature Fusion.
200. Canny, J. A Computational Approach to Edge Detection. *IEEE Trans Pattern Anal Mach Intell* **PAMI-8**, 679–698 (1986).
201. Tomasi, C. & Manduchi, R. Bilateral filtering for gray and color images. *Proceedings of the IEEE International Conference on Computer Vision* 839–846 (1998) doi:10.1109/ICCV.1998.710815.
202. Marr A N, Y. D. & Hildreth, D. E. Theory of edge detection. *Proc. R. Boc. Lond. B* **207**, 187–217 (1980).
203. Perez, L. & Wang, J. The Effectiveness of Data Augmentation in Image Classification using Deep Learning. (2017).
204. Caron, M. *et al.* Emerging Properties in Self-Supervised Vision Transformers. *Proceedings of the IEEE International Conference on Computer Vision* 9630–9640 (2021) doi:10.1109/ICCV48922.2021.00951.

205. Caron, M. *et al.* Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. *Adv Neural Inf Process Syst* **2020-December**, (2020).
206. Chen, T., Kornblith, S., Norouzi, M. & Hinton, G. A Simple Framework for Contrastive Learning of Visual Representations. *37th International Conference on Machine Learning, ICML 2020* **PartF168147-3**, 1575–1585 (2020).
207. Chen, T., Kornblith, S., Swersky, K., Norouzi, M. & Hinton, G. Big Self-Supervised Models are Strong Semi-Supervised Learners. *Adv Neural Inf Process Syst* **2020-December**, (2020).
208. Goodfellow, I. *et al.* Generative Adversarial Networks. *Commun ACM* **63**, 139–144 (2014).
209. Kingma, D. P. & Welling, M. Auto-Encoding Variational Bayes. *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings* (2013).
210. Odena, A. Semi-Supervised Learning with Generative Adversarial Networks. (2016).
211. Montavon, G., Samek, W. & Müller, K. R. Methods for interpreting and understanding deep neural networks. *Digit Signal Process* **73**, 1–15 (2018).
212. Minervini, M., Giuffrida, M. V., Perata, P. & Tsaftaris, S. A. Phenotiki: an open software and hardware platform for affordable and easy image-based phenotyping of rosette-shaped plants. *The Plant Journal* **90**, 204–216 (2017).
213. Araus, J. L. & Cairns, J. E. Field high-throughput phenotyping: the new crop breeding frontier. *Trends Plant Sci* **19**, 52–61 (2014).
214. Fahlgren, N. *et al.* A Versatile Phenotyping System and Analytics Platform Reveals Diverse Temporal Responses to Water Availability in Setaria. *Mol Plant* **8**, 1520–1535 (2015).
215. Zhou, J. *et al.* Leaf-GP: An open and automated software application for measuring growth phenotypes for arabidopsis and wheat. *Plant Methods* **13**, 1–17 (2017).

216. Dobrescu, A., Giuffrida, M. V. & Tsaftaris, S. A. Doing More With Less: A Multitask Deep Learning Approach in Plant Phenotyping. *Front Plant Sci* **11**, (2020).
217. Mohanty, S. P., Hughes, D. P. & Salathé, M. Using deep learning for image-based plant disease detection. *Front Plant Sci* **7**, 1419 (2016).
218. Paulus, S., Schumann, H., Kuhlmann, H. & Léon, J. High-precision laser scanning system for capturing 3D plant architecture and analysing growth of cereal plants. *Biosyst Eng* **121**, 1–11 (2014).
219. Yang, H. *et al.* FlowerPhenoNet: Automated Flower Detection from Multi-View Image Sequences Using Deep Neural Networks for Temporal Plant Phenotyping Analysis. *Remote Sensing 2022, Vol. 14, Page 6252* **14**, 6252 (2022).
220. David, E. *et al.* Global Wheat Head Detection (GWHD) Dataset: A Large and Diverse Dataset of High-Resolution RGB-Labelled Images to Develop and Benchmark Wheat Head Detection Methods. *Plant Phenomics* **2020**, 12 (2020).
221. David, E. *et al.* Global Wheat Head Dataset 2021: more diversity to improve the benchmarking of wheat head localization methods. (2021).
222. Wen, C. *et al.* Wheat Spike Detection and Counting in the Field Based on SpikeRetinaNet. *Front Plant Sci* **13**, (2022).
223. Lin, T. Y., Goyal, P., Girshick, R., He, K. & Dollar, P. Focal Loss for Dense Object Detection. *IEEE Trans Pattern Anal Mach Intell* **42**, 318–327 (2017).
224. Yang, K., Zhong, W. & Li, F. Leaf Segmentation and Classification with a Complicated Background Using Deep Learning. *Agronomy 2020, Vol. 10, Page 1721* **10**, 1721 (2020).
225. Van Dijk, M., Morley, T., Rau, M. L. & Saghai, Y. A meta-analysis of projected global food demand and population at risk of hunger for the period 2010-2050. doi:10.1038/s43016-021-00322-9.
226. Challinor, A. J. *et al.* A meta-analysis of crop yield under climate change and adaptation. *Nat. Clim. Chang.* **4**, 287–291 (2014).
227. Furbank, R. T. & Tester, M. Phenomics – technologies to relieve the phenotyping bottleneck. *Trends Plant Sci* **16**, 635–644 (2011).

228. Metzker, M. L. Sequencing technologies — the next generation. (2009) doi:10.1038/nrg2626.

229. Finch-Savage, W. E. & Bassel, G. W. Seed vigour and crop establishment: extending performance beyond adaptation. (2015) doi:10.1093/jxb/erv490.

230. Penfield, S., Macgregor, D. R. & Bassel, G. Effects of environmental variation during seed production on seed dormancy and germination. (2016) doi:10.1093/jxb/erw436.

231. Chakraborty, S. & Newton, A. C. Climate change, plant diseases and food security: An overview. *Plant Pathol* **60**, 2–14 (2011).

232. Haydon, M. J., Mielczarek, O., Robertson, F. C., Hubbard, K. E. & Webb, A. A. R. Photosynthetic entrainment of the Arabidopsis thaliana circadian clock. *Nature* (2013) doi:10.1038/nature12603.

233. Sanchez, A., Shin, J. & Davis, S. J. Plant Signaling & Behavior Abiotic stress and the plant circadian clock. (2011) doi:10.4161/psb.6.2.14893.

234. TeKrony, D. M. & Egli, D. B. Relationship of Seed Vigor to Crop Yield: A Review. *Crop Sci* **31**, 816–822 (1991).

235. Finch-Savage, W. E. & Bassel, G. W. Seed vigour and crop establishment: extending performance beyond adaptation. *J Exp Bot* **67**, 567–591 (2016).

236. Lin, Y., Sun, L., Nguyen, L. V., Rachubinski, R. A. & Goodman, H. M. The Pex16p Homolog SSE1 and Storage Organelle Formation in Arabidopsis Seeds. *Science (1979)* **284**, 328–330 (1999).

237. Joosen, R. V. L. *et al.* germinator: a software package for high-throughput scoring and curve fitting of Arabidopsis seed germination. *The Plant Journal* **62**, 148–159 (2010).

238. Jahnke, S. *et al.* phenoSeeder - A Robot System for Automated Handling and Phenotyping of Individual Seeds. *Plant Physiol* **172**, 1358–1370 (2016).
239. Keil, P., Liebsch, G., Borisjuk, L. & Rolletschek, H. MultiSense: A multimodal sensor tool enabling the high-throughput analysis of respiration. *Methods in Molecular Biology* **1670**, 47–56 (2017).
240. Lurstwut, B. & Pornpanomchai, C. Image analysis based on color, shape and texture for rice seed (Oryza sativa L.) germination evaluation. *Agriculture and Natural Resources* **51**, 383–389 (2017).
241. Mahajan, S., Mittal, S. K. & Das, A. Machine vision based alternative testing approach for physical purity, viability and vigour testing of soybean seeds (Glycine max). *J Food Sci Technol* **55**, 3949–3959 (2018).
242. Nguyen, T. T., Hoang, V. N., Le, T. L., Tran, T. H. & Vu, H. A vision based method for automatic evaluation of germination rate of rice seeds. *2018 1st International Conference on Multimedia Analysis and Pattern Recognition, MAPR 2018 - Proceedings* **2018-January**, 1–6 (2018).
243. Elmasry, G. *et al.* Utilization of computer vision and multispectral imaging techniques for classification of cowpea (Vigna unguiculata) seeds. *Plant Methods* **15**, 1–16 (2019).
244. Wu, D. *et al.* Combining high-throughput micro-CT-RGB phenotyping and genome-wide association study to dissect the genetic architecture of tiller growth in rice. *J Exp Bot* **70**, 545–561 (2019).
245. Ducournau, S. *et al.* Using computer vision to monitor germination time course of sunflower (Helianthus annuus L.) seeds. *Seed Science and Technology* **33**, 329–340 (2005).
246. Ligterink, W. & Hilhorst, H. W. M. High-Throughput Scoring of Seed Germination. *Methods Mol Biol* **1497**, 57–72 (2017).
247. Watson, A. *et al.* Speed breeding is a powerful tool to accelerate crop research and breeding. *Nature Plants 2018 4:1* **4**, 23–29 (2018).

248. Reynolds, D. *et al.* What is cost-efficient phenotyping? Optimizing costs for different scenarios. *Plant Science* **282**, 14–22 (2019).
249. Pound, M. P. *et al.* Deep machine learning provides state-of-the-art performance in image-based plant phenotyping. *Gigascience* **6**, 1 (2017).
250. Elmasry, G., Mandour, N., Al-Rejaie, S., Belin, E. & Rousseau, D. Recent Applications of Multispectral Imaging in Seed Phenotyping and Quality Monitoring—An Overview. *Sensors 2019, Vol. 19, Page 1090* **19**, 1090 (2019).
251. Baud, S., Dubreucq, B., Miquel, M., Rochat, C. & Lepiniec, L. Storage reserve accumulation in Arabidopsis: metabolic and developmental control of seed filling. *Arabidopsis Book* **6**, e0113 (2008).
252. Tang, L. *et al.* Genome-Wide Association Analysis Dissects the Genetic Basis of the Grain Carbon and Nitrogen Contents in Milled Rice. *Rice* **12**, 1–16 (2019).
253. Liu, Y. *et al.* GmSALT3, Which Confers Improved Soybean Salt Tolerance in the Field, Increases Leaf Cl- Exclusion Prior to Na+ Exclusion But Does Not Improve Early Vigor under Salinity. *Front Plant Sci* **7**, (2016).
254. Hu, M. K. Visual Pattern Recognition by Moment Invariants. *IRE Transactions on Information Theory* **8**, 179–187 (1962).
255. Harper, A. L. *et al.* Associative transcriptomics of traits in the polyploid crop species Brassica napus. *Nat Biotechnol* **30**, 798–802 (2012).
256. Harper, A. L. *et al.* Associative transcriptomics of traits in the polyploid crop species Brassica napus. *Nat Biotechnol* **30**, 798–802 (2012).
257. Hatzig, S. *et al.* Hidden Effects of Seed Quality Breeding on Germination in Oilseed Rape (Brassica napus L.). *Front Plant Sci* **9**, 1–14 (2018).
258. Hatzig, S. V. *et al.* Genome-wide association mapping unravels the genetic control of seed germination and vigor in Brassica napus. *Front Plant Sci* **6**, 1–13 (2015).
259. Yoshida, T. *et al.* ABA-hypersensitive germination3 encodes a protein phosphatase 2C (AtPP2CA) that strongly regulates abscisic acid

signaling during germination among Arabidopsis protein phosphatase 2Cs. *Plant Physiol* **140**, 115–126 (2006).

260. Bhaskara, G. B., Nguyen, T. T. & Verslues, P. E. Unique Drought Resistance Functions of the Highly ABA-Induced Clade A Protein Phosphatase 2Cs. *Plant Physiol* **160**, 379–395 (2012).
261. Furbank, R. T., Jimenez-Berni, J. A., George-Jaeggli, B., Potgieter, A. B. & Deery, D. M. Field crop phenomics: enabling breeding for radiation use efficiency and biomass in cereal crops. *New Phytologist* **223**, 1714–1727 (2019).
262. Tardieu, F., Cabrera-Bosquet, L., Pridmore, T. & Bennett, M. Plant Phenomics, From Sensors to Knowledge. *Current Biology* **27**, R770–R783 (2017).
263. Zhou, J. *et al.* Plant phenomics: history, present status and challenges. *Journal of Nanjing Agricultural University* **41**, 580–588 (2018).
264. Yang, W. *et al.* Crop Phenomics and High-Throughput Phenotyping: Past Decades, Current Challenges, and Future Perspectives. *Mol Plant* **13**, 187–214 (2020).
265. Sansone, S. A. S. A. *et al.* Toward interoperable bioscience data. *Nat Genet* **44**, 121 (2012).
266. Ćwiek-Kupczyńska, H. *et al.* Measures for interoperability of phenotypic data: minimum information requirements and formatting. *Plant Methods* **12**, 44 (2016).
267. Oellrich, A. *et al.* An ontology approach to comparative phenomics in plants. *Plant Methods* **11**, 10 (2015).
268. Demilly, D., Ducournau, S. & Wagner, M.-H. Digital imaging of seed germination. in *Plant Image Analysis Fundamentals and Applications* (eds. Gupta, S. D. & Ibaraki, Y.) 147–165 (CRC Press, Boca Raton, 2015). doi:10.1201/b17441.
269. Nguyen, T. T., Hoang, V. N., Le, T. L., Tran, T. H. & Vu, H. A vision based method for automatic evaluation of germination rate of rice seeds. *2018 1st International Conference on Multimedia Analysis and Pattern Recognition, MAPR 2018 - Proceedings* **2018-Janua**, 1–6 (2018).

270. Wu, D. *et al.* Combining high-throughput micro-CT-RGB phenotyping and genome-wide association study to dissect the genetic architecture of tiller growth in rice. *J Exp Bot* **70**, 545–561 (2019).
271. Jahnke, S. *et al.* pheno Seeder - A Robot System for Automated Handling and Phenotyping of Individual Seeds . *Plant Physiol* **172**, 1358–1370 (2016).
272. Elmasry, G. *et al.* Utilization of computer vision and multispectral imaging techniques for classification of cowpea (Vigna unguiculata) seeds. *Plant Methods* **15**, 1–16 (2019).
273. Millman, K. J. & Aivazis, M. Python for scientists and engineers. *Comput Sci Eng* **13**, 9–12 (2011).
274. van der Walt, S. *et al.* Scikit-image: image processing in Python. *PeerJ* **2**, 1–18 (2014).
275. Howse, J. *OpenCV Computer Vision with Python*. (Packt Publishing Ltd., Birmingham, UK, 2013).
276. Pedregosa, F. *et al.* Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* **12**, 2825–2830 (2011).
277. Rampasek, L. & Goldenberg, A. TensorFlow: Biology's Gateway to Deep Learning? *Cell Syst* **2**, 12–14 (2016).
278. Mahajan, S., Mittal, S. K. & Das, A. Machine vision based alternative testing approach for physical purity, viability and vigour testing of soybean seeds (Glycine max). *J Food Sci Technol* **55**, 3949–3959 (2018).
279. Halcro, K. *et al.* The BELT and phenoSEED platforms: shape and colour phenotyping of seed samples. *Plant Methods* **16**, 1–13 (2020).
280. Khurana, E. & Singh, J. S. Ecology of seed and seedling growth for conservation and restoration of tropical dry forest : A review. *Environ Conserv* **28**, 39–52 (2001).
281. Dell'Aquila, A. New Perspectives for Seed Germination Testing Through Digital Imaging Technology. *Open Agric J* **3**, 37–42 (2009).

282. Attree, S. M., Pomeroy, M. K. & Fowke, L. C. Manipulation of conditions for the culture of somatic embryos of white spruce for improved triacylglycerol biosynthesis and desiccation tolerance. *Planta* **187**, 395–404 (1992).
283. Paparella, S. *et al.* Seed priming: state of the art and new perspectives. *Plant Cell Rep* **34**, 1281–1293 (2015).
284. Nelson, D. C., Flematti, G. R., Ghisalberti, E. L., Dixon, K. W. & Smith, S. M. Regulation of Seed Germination and Seedling Growth by Chemical Signals from Burning Vegetation. *Annu Rev Plant Biol* **63**, 107–130 (2012).
285. Wen, L., Gao, L. & Li., X. A new deep transfer learning based on sparse auto-encoder for fault diagnosis. *IEEE Trans Syst Man Cybern Syst* **49**, 136–144 (2017).
286. Agrios, G. Plant pathology: Fifth edition. *Plant Pathology: Fifth Edition* **9780080473789**, 1–922 (2004).
287. Baker, N. R. & Ort, D. R. Light and crop photosynthetic performance. *Topics in photosynthesis* (1992) doi:10.3/JQUERY-UI.JS.
288. Strange, R. N. & Scott, P. R. Plant disease: a threat to global food security. *Annu Rev Phytopathol* **43**, 83–116 (2005).
289. Jones, R. A. C. Plant virus ecology and epidemiology: historical perspectives, recent progress and future prospects. *Annals of Applied Biology* **164**, 320–347 (2014).
290. Waterworth, H. E. & Hadidi., A. Economic losses due to plant viruses. in *Plant Virus Disease Control* (ed. Hadidi et al.) 684 (American Phytopathological Society Press, St. Paul, Minn., 1998).
291. Vincent, S. J., Coutts, B. A. & Jones, R. A. C. Effects of introduced and indigenous viruses on native plants: Exploring their disease causing potential at the agro-ecological interface. *PLoS One* **9**, 6–9 (2014).
292. Wang, M. R. *et al.* In vitro thermotherapy-based methods for plant virus eradication. *Plant Methods* **14**, 1–18 (2018).
293. Hull, R. *Plant Virology*. (Elsevier, Academic Press, London, UK, 2014).
294. Froissart, R., Doumayrou, J., Vuillaume, F., Alizon, S. & Michalakis, Y. The virulence-transmission trade-off in vector-borne plant viruses: A

review of (non-)existing studies. *Philosophical Transactions of the Royal Society B: Biological Sciences* **365**, 1907–1918 (2010).

295. Walsh, J. A. & Jenner, C. E. Turnip mosaic virus and the quest for durable resistance. *Mol Plant Pathol* **3**, 289–300 (2002).
296. Tomlinson, J. A. Epidemiology and control of virus diseases of vegetables. *Annals of Applied Biology* **110**, 661–681 (1987).
297. Gibbs, A. & Ohshima, K. Potyviruses and the digital revolution. *Annu Rev Phytopathol* **48**, 205–223 (2010).
298. Clark, M. F. & Adams, A. N. Characteristics of the microplate method of enzyme-linked immunosorbent assay for the detection of plant viruses. *J Gen Virol* **34**, 475–483 (1977).
299. Mullis, K. *et al.* Specific enzymatic amplification of DNA in vitro: the polymerase chain reaction. *Cold Spring Harb Symp Quant Biol* **51 Pt 1**, 263–273 (1986).
300. Bustin, S. A. *et al.* The MIQE Guidelines: Minimum Information for Publication of Quantitative Real-Time PCR Experiments. *Clin Chem* **55**, 611–622 (2009).
301. Boonham, N. *et al.* Methods in virus diagnostics: from ELISA to next generation sequencing. *Virus Res* **186**, 20–31 (2014).
302. Hindson, B. J. *et al.* High-throughput droplet digital PCR system for absolute quantitation of DNA copy number. *Anal Chem* **83**, 8604–8610 (2011).
303. Gutiérrez-Aguirre, I., Rački, N., Dreo, T. & Ravnikar, M. Droplet digital PCR for absolute quantification of pathogens. *Methods Mol Biol* **1302**, 331–347 (2015).
304. Robène, I. *et al.* Development and comparative validation of genomic-driven PCR-based assays to detect Xanthomonas citri pv. citri in citrus plants. *BMC Microbiol* **20**, (2020).
305. Westermann, A. J., Gorski, S. A. & Vogel, J. Dual RNA-seq of pathogen and host. *Nature Reviews Microbiology 2012 10:9* **10**, 618–630 (2012).

306. Sperschneider, J. Machine learning in plant–pathogen interactions: empowering biological predictions from field scale to genome scale. *New Phytologist* **228**, 35–41 (2020).
307. McCarthy, J. F. *et al.* Applications of Machine Learning and High-Dimensional Visualization in Cancer Detection, Diagnosis, and Management. *Ann N Y Acad Sci* **1020**, 239–262 (2004).
308. Shipp, M. A. *et al.* Diffuse large B-cell lymphoma outcome prediction by gene-expression profiling and supervised machine learning. *Nature Medicine 2002 8:1* **8**, 68–74 (2002).
309. Chittem, K., Yajima, W. R., Goswami, R. S. & del Río Mendoza, L. E. Transcriptome analysis of the plant pathogen Sclerotinia sclerotiorum interaction with resistant and susceptible canola (Brassica napus) lines. *PLoS One* **15**, e0229844 (2020).
310. Ashburner, M. *et al.* Gene Ontology: tool for the unification of biology. *Nat Genet* **25**, 25 (2000).
311. Fang, Y. & Ramasamy, R. P. Current and prospective methods for plant disease detection. *Biosensors (Basel)* **5**, 537–561 (2015).
312. Richards, C. L., Rosas, U., Banta, J., Bhambhra, N. & Purugganan, M. D. Genome-wide patterns of Arabidopsis gene expression in nature. *PLoS Genet* **8**, 1–14 (2012).
313. Yang, Y., Yu, X., Song, L. & An, C. ABI4 activates DGAT1 expression in Arabidopsis seedlings during nitrogen deficiency. *Plant Physiol* **156**, 873–83 (2011).
314. Marchand, G. *et al.* A biomarker based on gene expression indicates plant water status in controlled and natural environments. *Plant Cell Environ* **36**, 2175–2189 (2013).
315. Phelix, C. F. & Feltus, F. A. Plant stress biomarkers from biosimulations: The Transcriptome-To-Metabolome™ (TTM™) technology - effects of drought stress on rice. *Plant Biol* **17**, 63–73 (2015).

316. Lu, J. *et al.* Transcriptome Analysis of Nicotiana tabacum Infected by Cucumber mosaic virus during Systemic Symptom Development. *PLoS One* **7**, (2012).
317. Honjo, M. N. *et al.* Seasonality of interactions between a plant virus and its host during persistent infection in a natural environment. *ISME Journal* **14**, 506–518 (2020).
318. Jimenez-Gomez, J. M., Corwin, J. A., Joseph, B., Maloof, J. N. & Kliebenstein, D. J. Genomic analysis of QTLs and genes altering natural variation in stochastic noise. *PLoS Genet* **7**, (2011).
319. Linsen, S. E. V. *et al.* Limitations and possibilities of small RNA digital gene expression profiling. *Nat Methods* **6**, 474–476 (2009).
320. Robinson, M. D., McCarthy, D. J. & Smyth, G. K. edgeR: A Bioconductor package for differential expression analysis of digital gene expression data. *Bioinformatics* **26**, 139–140 (2009).
321. Huynh-Thu, V. A., Irrthum, A., Wehenkel, L. & Geurts, P. Inferring regulatory networks from expression data using tree-based methods. *PLoS One* **5**, 1–10 (2010).
322. Love, M. I., Huber, W. & Anders, S. Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. *Genome Biol* **15**, 550 (2014).
323. Varet, H., Brillet-Guéguen, L., Coppée, J. Y. & Dillies, M. A. SARTools: A DESeq2- and edgeR-based R pipeline for comprehensive differential analysis of RNA-Seq data. *PLoS One* **11**, 1–8 (2016).
324. Anderson, M. J. Permutation tests for univariate or multivariate analysis of variance and regression. *Canadian Journal of Fisheries and Aquatic Sciences* **58**, 626–639 (2001).
325. Cios, K. J., Kurgan, L. A. & Reformat, M. Machine Learning in the Life Sciences. *IEEE Engineering in Medicine and Biology Magazine* **26**, 27–36 (2007).
326. Divya, K. S., Bhargavi, P. & Jyothi, S. Machine Learning Algorithms in Big data Analytics. *International Journal of Computer Sciences and Engineering* **6**, 63–70 (2018).

327. Sommer, C. & Gerlich, D. W. Machine learning in cell biology - teaching computers to recognize phenotypes. *J Cell Sci* **126**, 5529–39 (2013).
328. Libbrecht, M. W. & Noble, W. S. Machine learning applications in genetics and genomics. *Nat Rev Genet* **16**, 321–332 (2015).
329. Tsaftaris, S. A., Minervini, M. & Scharr, H. Machine Learning for Plant Phenotyping Needs Image Processing. *Trends Plant Sci* **21**, 989–991 (2016).
330. Bauer, A. *et al.* Combining computer vision and deep learning to enable ultra-scale aerial phenotyping and precision agriculture: A case study of lettuce production. *Hortic Res* **6**, 70 (2019).
331. Angermueller, C., Pärnamaa, T., Parts, L. & Stegle, O. Deep learning for computational biology. *Mol Syst Biol* **12**, 878 (2016).
332. Li, Y. *et al.* A comprehensive genomic pan-cancer classification using The Cancer Genome Atlas gene expression data. *BMC Genomics* **18**, 1–13 (2017).
333. Perez-Riverol, Y., Kuhn, M., Vizcaíno, J. A., Hitz, M. P. & Audain, E. Accurate and fast feature selection workflow for high-dimensional omics data. *PLoS One* **12**, 1–14 (2017).
334. Devi Arockia Vanitha, C., Devaraj, D. & Venkatesulu, M. Gene expression data classification using Support Vector Machine and mutual information-based gene selection. *Procedia Comput Sci* **47**, 13–21 (2014).
335. Urda, D., Montes-Torres, J., Moreno, F., Franco, L. & Jerez, J. M. Deep learning to analyze RNA-Seq gene expression data. in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* vol. 10306 LNCS 50–59 (2017).
336. Lyu, B. & Haque, A. Deep Learning Based Tumor Type Classification Using Gene Expression Data. in *Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics - BCB '18* 89–96 (2018). doi:10.1145/3233547.3233588.

337. Sharma, A., Vans, E., Shigemizu, D., Boroevich, K. A. & Tsunoda, T. DeepInsight: A methodology to transform a non-image data to an image for convolution neural network architecture. *Sci Rep* **9**, 1–7 (2019).
338. Hira, Z. M. & Gillies, D. F. A Review of Feature Selection and Feature Extraction Methods Applied on Microarray Data. (2015) doi:10.1155/2015/198363.
339. Piao, Y., Piao, M., Park, K. & Ryu, K. H. H. An ensemble correlation-based gene selection algorithm for cancer classification with gene expression data. *Bioinformatics* **28**, 3306–3315 (2012).
340. Liu, S. *et al.* Feature selection of gene expression data for Cancer classification using double RBF-kernels. *BMC Bioinformatics* **19**, 1–14 (2018).
341. Sakhinia, E. *et al.* Comparison of gene-expression profiles in parallel bone marrow and peripheral blood samples in acute myeloid leukaemia by real-time polymerase chain reaction. *J Clin Pathol* **59**, 1059–1065 (2006).
342. Li, T., Zhang, C. & Ogihara, M. A comparative study of feature selection and multiclass classfication methods for tissue classification based on gene expression. *Bioinformatics* **20**, 2429–2437 (2004).
343. Gheyas, I. A. & Smith, L. S. Feature subset selection in large dimensionality domains. *Pattern Recognit* **43**, 5–13 (2010).
344. Libbrecht, M. W. & Noble, W. S. Machine learning applications in genetics and genomics. *Nat Rev Genet* **16**, 321–332 (2015).
345. Ching, T. *et al.* Opportunities and obstacles for deep learning in biology and medicine. *J R Soc Interface* **15**, (2018).
346. Yousefi, S. *et al.* Predicting clinical outcomes from large scale cancer genomic profiles with deep survival models. *Scientific Reports 2017 7:1* **7**, 1–11 (2017).
347. Golub, T. R. *et al.* Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* **286**, 531–527 (1999).

348. Shi, M. & Zhang, B. Semi-supervised learning improves gene expression-based prediction of cancer recurrence. *Bioinformatics* **27**, 3017–3023 (2011).
349. Chapelle, O. & Zien, A. Semi-Supervised Classification by Low Density Separation. 57–64 Preprint at https://proceedings.mlr.press/r5/chapelle05b.html (2005).
350. Cheng, C. Y. *et al.* Evolutionarily informed machine learning enhances the power of predictive gene-to-phenotype relationships. *Nature Communications 2021 12:1* **12**, 1–15 (2021).
351. Cui, X. & Churchill, G. A. Statistical tests for differential expression in cDNA microarray experiments. *Genome Biol* **4**, (2003).
352. Rapaport, F. *et al.* Comprehensive evaluation of differential gene expression analysis methods for RNA-seq data. *Genome Biol* **14**, 1–13 (2013).
353. Anders, S. & Huber, W. Differential expression analysis for sequence count data. *Genome Biol* **11**, 1–12 (2010).
354. Robinson, M. D. & Oshlack, A. A scaling normalization method for differential expression analysis of RNA-seq data. *Genome Biol* **11**, 1–9 (2010).
355. Benjamini, Y. & Hochberg, Y. Controlling the False Discovery Rate: A Practical and Powerful Approach to Multiple Testing. *Journal of the Royal Statistical Society: Series B (Methodological)* **57**, 289–300 (1995).
356. Rice, T. K., Schork, N. J. & Rao, D. C. Methods for Handling Multiple Testing. *Adv Genet* **60**, 293–308 (2008).
357. Conesa, A. *et al.* Blast2GO: a universal tool for annotation, visualization and analysis in functional genomics research. *Bioinformatics* **21**, 3674–3676 (2005).
358. Yon Rhee, S., Wood, V., Dolinski, K. & Draghici, S. Use and misuse of the gene ontology annotations. *Nat Rev Genet* **9**, 509–515 (2008).

359. Hammond-Kosack, K. E. & Jones, J. D. G. PLANT DISEASE RESISTANCE GENES. *Annu Rev Plant Physiol Plant Mol Biol* **48**, 575–607 (1997).
360. Barabási, A. L. & Oltvai, Z. N. Network biology: understanding the cell's functional organization. *Nature Reviews Genetics 2004 5:2* **5**, 101–113 (2004).
361. Huynh-Thu, V. A., Irrthum, A., Wehenkel, L. & Geurts, P. Inferring Regulatory Networks from Expression Data Using Tree-Based Methods. *PLoS One* **5**, e12776 (2010).
362. Marbach, D. *et al.* Wisdom of crowds for robust gene network inference. *Nat Methods* **9**, 796–804 (2012).
363. Shannon, P. *et al.* Cytoscape: A Software Environment for Integrated Models of Biomolecular Interaction Networks. *Genome Res* **13**, 2498 (2003).
364. Bauer-Mehren, A. *et al.* Gene-Disease Network Analysis Reveals Functional Modules in Mendelian, Complex and Environmental Diseases. *PLoS One* **6**, e20284 (2011).
365. Guyon, I., Weston, J., Barnhill, S. & Vapnik, V. Gene selection for cancer classification using support vector machines. *Mach Learn* **46**, 389–422 (2002).
366. Alon, U. *et al.* Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci U S A* **96**, 6745–6750 (1999).
367. Saeys, Y., Inza, I. & Larrañaga, P. A review of feature selection techniques in bioinformatics. *Bioinformatics* **23**, 2507–2517 (2007).
368. Maaten, L. van der & Hinton, G. Visualizing Data using t-SNE. *Journal of Machine Learning Research* **9**, 2579–2605 (2008).
369. Alon, U. *et al.* Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proc Natl Acad Sci U S A* **96**, 6745–6750 (1999).

370. Van't Veer, L. J. *et al.* Gene expression profiling predicts clinical outcome of breast cancer. *Nature 2002 415:6871* **415**, 530–536 (2002).
371. Subramanian, A. *et al.* Gene set enrichment analysis: A knowledge-based approach for interpreting genome-wide expression profiles. *Proc Natl Acad Sci U S A* **102**, 15545–15550 (2005).
372. Wagner, G. P., Kin, K. & Lynch, V. J. Measurement of mRNA abundance using RNA-seq data: RPKM measure is inconsistent among samples. *Theory Biosci* **131**, 281–285 (2012).
373. Zhu, Y. *et al.* Converting tabular data into images for deep learning with convolutional neural networks. *Scientific Reports 2021 11:1* **11**, 1–11 (2021).
374. Raghu, M., Unterthiner, T., Kornblith, S., Zhang, C. & Dosovitskiy, A. Do Vision Transformers See Like Convolutional Neural Networks? *Adv Neural Inf Process Syst* **15**, 12116–12128 (2021).
375. Bangyal, W. ; H. *et al.* Analyzing RNA-Seq Gene Expression Data Using Deep Learning Approaches for Cancer Classification. *Applied Sciences 2022, Vol. 12, Page 1850* **12**, 1850 (2022).
376. Lyu, B. & Haque, A. Deep Learning Based Tumor Type Classification Using Gene Expression Data. *bioRxiv* 364323 (2018) doi:10.1101/364323.
377. Selvaraju, R. R. *et al.* Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *Proceedings of the IEEE International Conference on Computer Vision* **2017-Octob**, 618–626 (2017).
378. Sharma, A., Lysenko, A., Boroevich, K. A., Vans, E. & Tsunoda, T. DeepFeature: feature selection in nonimage data using convolutional neural network. *Brief Bioinform* **22**, 1–12 (2021).
379. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A. & Torralba, A. Learning Deep Features for Discriminative Localization. *Proceedings of the*

IEEE Computer Society Conference on Computer Vision and Pattern Recognition **2016-December**, 2921–2929 (2015).

380. Tomczak, K., Czerwińska, P. & Wiznerowicz, M. The Cancer Genome Atlas (TCGA): an immeasurable source of knowledge. *Contemp Oncol* **19**, A68 (2015).
381. Nagano, A. J. *et al.* Annual transcriptome dynamics in natural environments reveals plant seasonal adaptation. *Nat Plants* **5**, 74–83 (2019).
382. Lieberman, N. A. P. *et al.* In vivo antiviral host transcriptional response to SARS-CoV-2 by viral load, sex, and age. *PLoS Biol* **18**, e3000849 (2020).
383. Kinney, J. B. & Atwal, G. S. Equitability, mutual information, and the maximal information coefficient. *Proc Natl Acad Sci U S A* **111**, 3354–9 (2014).
384. Mason, L., Baxter, J., Bartlett, P. & Frean, M. Boosting Algorithms as Gradient Descent. in *International Conference on Neural Information Processing Systems.* 512–518 (1999). doi:10.1103/PhysRevD.91.072004.
385. Zhai, B. & Chen, J. Development of a stacked ensemble model for forecasting and analyzing daily average PM2.5 concentrations in Beijing, China. *Science of The Total Environment* **635**, 644–658 (2018).
386. Chattopadhyay, A., Sarkar, A., Howlader, P. & Balasubramanian, V. N. Grad-CAM++: Improved Visual Explanations for Deep Convolutional Networks. (2017) doi:10.1109/WACV.2018.00097.
387. Altmann, A., Toloşi, L., Sander, O. & Lengauer, T. Permutation importance: A corrected feature importance measure. *Bioinformatics* **26**, 1340–1347 (2010).
388. Costa, M. J. *et al.* Inference on periodicity of circadian time series. *Biostatistics* **14**, 792–806 (2013).
389. Zielinski, T., Moore, A. M., Troup, E., Halliday, K. J. & Millar, A. J. Strengths and limitations of period estimation methods for circadian data. *PLoS One* **9**, e96462 (2014).

390. Ueda, H. R. *et al.* Molecular-timetable methods for detection of body time and rhythm disorders from single-time-point genome-wide expression profiles. *Proc Natl Acad Sci U S A* **101**, 11227–11232 (2004).
391. Hughey, J. J., Hastie, T. & Butte, A. J. ZeitZeiger: Supervised learning for high-dimensional data from an oscillatory system. *Nucleic Acids Res* **44**, e80 (2016).
392. Braun, R. *et al.* Universal method for robust detection of circadian state from gene expression. *Proc Natl Acad Sci U S A* **115**, E9247–E9256 (2018).
393. Romanowski, A., Schlaen, R. G., Perez-Santangelo, S., Mancini, E. & Yanovsky, M. J. Global transcriptome analysis reveals circadian control of splicing events in *Arabidopsis thaliana*. *The Plant Journal* **103**, 889–902 (2020).
394. Yang, Y., Li, Y., Sancar, A. & Oztas, O. The circadian clock shapes the Arabidopsis transcriptome by regulating alternative splicing and alternative polyadenylation. *Journal of Biological Chemistry* **295**, 7608–7619 (2020).
395. Graf, A. *et al.* Parallel analysis of arabidopsis circadian clock mutants reveals different scales of transcriptome and proteome regulation. *Open Biol* **7**, (2017).
396. Zhang, Y., Parmigiani, G. & Johnson, W. E. ComBat-seq: batch effect adjustment for RNA-seq count data. *NAR Genom Bioinform* **2**, (2020).
397. Hughes, M. E., Hogenesch, J. B. & Kornacker, K. JTK-CYCLE: An efficient nonparametric algorithm for detecting rhythmic components in genome-scale data sets. *J Biol Rhythms* **25**, 372–380 (2010).
398. Yang, R. & Su, Z. Analyzing circadian expression data by harmonic regression based on autoregressive spectral estimation. *Bioinformatics* **26**, 168–174 (2010).
399. Massey, F. J. The Kolmogorov-Smirnov Test for Goodness of Fit. *J Am Stat Assoc* **46**, 68 (1951).