Heterogeneous Ensembles and Time Series Classification Techniques for the Non-Invasive Authentication of Spirits

James Large

School of Computing Sciences

University of East Anglia

This dissertation is submitted for the degree of

Doctor of Philosophy

April 2022

Acknowledgements

I'd like to thank my supervisor, Dr. Anthony Bagnall, for his continued mentorship both through my PhD and previously through my undergraduate degree. I certainly wouldn't be in this position today if he hadn't taken me under his wing. Thanks go to the Scotch Whisky Research Institute for funding of the project, and in particular to Dr. Ian Goodall and Dr. James Brosnan for their support and for graciously hosting my time in Edinburgh. I would also like to thank the Norwich Research Park and the BBSRC DTP scheme for their training and support.

The entirety of the time series classification group have been invaluable and a pleasure to work with. We are lucky to have a welcoming, friendly and supportive student community at UEA. Thanks to all the people that shared overly long lunches. I'd also like to thank the computing services and HPC team at UEA for tolerating my use of the cluster, sorry for bringing it all down that one time.

A special thanks goes to Dr. Aaron Bostrom, Amy Fellows, and Nacho for keeping me alive during the dark COVID lockdown days. Further thanks to everyone else who kept me going along the way, particularly Josh, Catherine, and my family.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements.

James Large

April 2022

Publications

As First Author

- Large, J., Lines, J., and Bagnall, A. (2019b). A probabilistic classifier ensemble weighting scheme based on cross-validated accuracy estimates. Data mining and knowledge discovery, 33(6):1674–1709.
- Large, J. and Bagnall, A. (2019). Mixing hetero-and homogeneous models in weighted ensembles. In International Conference on Intelligent Data Engineering and Automated Learning, pages 129–136. Springer, Cham.
- Large, J., Kemsley, E. K., Wellner, N., Goodall, I., and Bagnall, A. (2018). Detecting forged alcohol non-invasively through vibrational spectroscopy and machine learning. In Pacific-Asia conference on knowledge discovery and data mining, pages 298–309. Springer, Cham.
- Large, J., Bagnall, A., Malinowski, S., and Tavenard, R.(2019a). On time series classification with dictionary-based classifiers. Intelligent Data Analysis, 23(5):1073–1089.
- Large, J., Southam, P., and Bagnall, A. (2019c). Can automated smoothing significantly improve benchmark time series classification algorithms? In International Conference on Hybrid Artificial Intelligence Systems, pages 50–60. Springer, Cham.

As Second Author

- Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. Data mining and knowledge discovery, 31(3):606–660.

- Bagnall, A., Flynn, M., Large, J., Lines, J., and Middlehurst, M. (2020). On the usage and performance of the hierarchical vote collective of transformation-based ensembles version 1.0 (hive-cote v1. 0). In International Workshop on Advanced Analytics and Learning on Temporal Data, pages 3–18. Springer, Cham.

- Middlehurst, M., Large, J., Flynn, M., Lines, J., Bostrom, A., and Bagnall, A. (2021b). Hive-cote 2.0: a new meta ensemble for time series classification. arXiv preprint arXiv:2104.07551.

- Flynn, M., Large, J., and Bagnall, A. (2019). The contract random interval spectral ensemble (c-RISE): the effect of contracting a classifier on accuracy. In International Conference on Hybrid Artificial Intelligence Systems, pages 381–392. Springer, Cham.

- Middlehurst, M., Large, J., and Bagnall, A. (2020). The canonical interval forest (CIF) classifier for time series classification. arXiv preprint arXiv:2008.09172.

- Middlehurst, M., Large, J., Cawley, G., and Bagnall, A. (2021a). The temporal dictionary ensemble (TDE) classifier for time series classification. arXiv preprint arXiv:2105.03841.

- Bagnall, A., Flynn, M., Large, J., Line, J., Bostrom, A., and Cawley, G. (2018b). Is rotation forest the best classifier for problems with continuous features? arXiv preprint arXiv:1809.06705.

- Bagnall, A., Dau, H. A., Lines, J., Flynn, M., Large, J., Bostrom, A., Southam, P., and Keogh, E. (2018a). The uea multivariate timeseries classification archive, 2018. arXiv preprint arXiv:1811.00075.

Abstract

Spirits are a prime target for fraudulent activity. Particular brands, production processes, and other factors such as age can carry high value, and leave space for mimicry. Further, the improper production of spirits, either maliciously or through negligence, can result in harmful substances being sold for consumption. Lastly, genuine spirits producers themselves must ensure the quality and standardisation of their products before sale. Authenticating spirits can be a time consuming and destructive process, requiring sealed bottles to be opened for access to the product.

It is therefore desirable to have a fast, non-invasive means of indicating the authenticity, safety, and correctness of spirits. We advance and prototype such a system based on near infrared spectroscopy, and generate datasets for the detection of correct alcohol concentrations in synthesised spirits, for the presence of methanol in genuine spirits, and for the distinction of particular genuine products in a given bottle.

The standard chemometric pipelines for the analysis of spectra involve smoothing of the signal, standardising for global intensity, possible dimensionality reduction, and some form of least squares regression. This has decades of proof behind it, and works under the assumptions of clean signal gathering, potentially the separation of sample and particular substance of interest, and the generally linear relationship of light received/blocked and the analyte's contents. In the proposed system, at least one of these assumptions must be violated.

We therefore investigate the use of modern classification techniques to overcome these challenges. In particular, we investigate and develop ensemble methods and time series classification algorithms. Our first hypothesis is that algorithms which consider the ordered nature of the wavelength features, as opposed to treating the spectra effectively as tabular data, can better handle the structural changes brought about by different bottle and environmental characteristics. The second is that ensembling heterogeneous classifiers is the best initial technique for a new data science problem, but should in particular be helpful for the spirit authentication problem, where different classifiers may be able to correct for different defects in the data.

In initial investigations on datasets of synthesised alcohol solutions and different products, we prove the feasibility of the authentication system to make at least indicative predictions of authenticity, but find that it lacks the precision and accuracy needed for anything more than indicative results. Following this, we propose a novel heterogeneous ensembling scheme, CAWPE, and perform a large scale evaluation on public archives to prove its efficacy. We then outline improvements in the time series classification space that lead to the state of the art meta-ensemble HIVE-COTE 2.0, which makes use of CAWPE. We lastly apply the developed techniques to a final dataset on methanol concentration detection. We find that the proposed system can classify methanol concentration in arbitrary spirits and bottles from ten possible values, containing as little as 0.25%, to an accuracy of 0.921. We further conclude that while heterogeneously ensembling tabular classifiers does improve the authentication of spirits from spectra, time series classification methods confer no particular advantage beyond tabular methods.

Table of contents

List of figures

List of tables

Chapter 1

Introduction

Counterfeit alcohol poses potentially fatal health risks to the consumer where illegally and poorly produced spirits may contain harmful contaminants such as methanol, a large economic risk in most markets due to the avoidance of taxes, and a risk to brand integrity in cases where the fakes are being sold as named brands.

In a series of Trading Standards raids in 2010, up to 25% of licensed premises in some parts of the UK were found to have counterfeit alcohol for sale*, while a third of rare and auctioned whiskies have also been discovered as forged†. Brown-Forman, the company that makes Jack Daniels, estimates that around 30% of all alcohol in China is fake‡.

Forgeries can sometimes be detected through external appearance such as inconsistent labelling or bottling relative to a known standard, but currently there is no way to conclusively tell whether spirits are forged without opening the bottle to gain direct contact with the sample. Breaking the seal and taking samples from a bottle can be effectively a destructive process, because even if authenticity is confirmed the bottle cannot later be sold on store shelves or at auction, and

*https://www.bbc.co.uk/news/uk-12456360

†https://www.bbc.co.uk/news/uk-scotland-scotland-business-46566703

‡https://www.theguardian.com/sustainable-business/2015/sep/16/china-fake-alcohol-industry-counterfeit-bathtub-booze-whisky

collectors' whisky will be greatly devalued. Also, testing of samples can be an expensive and time consuming process that is not suitable for mass screening. No matter what process is used it will require one or likely more of: transport of the sample to a centralised lab; expert knowledge and handling; consumable materials used in the analysis; and analysis time for methods such as chromatography. It is therefore desirable to develop a system that can non-invasively determine the authenticity of a suspect bottle on-site in a cheap, simple and fast manner.

Near infrared spectroscopy (NIRS) in combination with modern machine learning techniques provides a promising potential solution to these problems. Ever improving and more affordable computing power and spectroscopy equipment, as well as continual advancements in data mining and machine learning methods, mean that on-site classification using cost effective equipment is becoming evermore feasible. Such setups are already used in a variety of food and drink authentication scenarios. We present three main sets of data collected for this thesis, in Chapters 3 and 5, using a prototype through-bottle near infrared spectroscopy system. We use these datasets to work towards a functioning system for the alcohol authenticity problem itself, but also as a test bed for the algorithms compared and evaluated throughout the thesis.

We investigate two broad strands of machine learning and their application to the forged alcohol problem. These are the construction and use of ensembles to improve classification performance and resultant human decision power and confidence, and the use of time series classification methods for spectroscopy data to leverage the ordered nature of the wavelength attributes.

Classification is the supervised task of assigning one or more predefined labels to instances of some problem. Learning algorithms are trained on a set of training data with known labels to learn a mapping from the data to their labels. The output of a learning algorithm applied to a dataset is a classifier that can make predictions

on new unlabeled data by applying the learned mapping. In the context of this thesis, we are interested in training classifiers to determine whether a suspect spirit is forged, has been adulterated, or contains harmful substances.

Some individual experiments in later chapters relate to predicting the concentrations of individual substances in samples, as a proxy for authenticity. Namely, the concentration of ethanol being within tolerable levels of that indicated on the label, and the absence of methanol, which is toxic. These sub-problems are more innately ordinal regression. However, we are still more interested in reducing these to classification problems of authentic or not. This is achieved by discretising the output space from continuous concentrations to buckets of acceptable versus not concentrations. We still compare to industry-standard regression methodologies to throughout these experiments.

Ensemble methods train and combine the predictions of multiple classifiers with the aim of improving some factor of performance over using any single classifier. They leverage the combination of a diverse set of experts that have learned different aspects of a problem, or learned in different ways (via different random initialisations, learning parameters, or entirely different learning algorithms) to make predictions that are (hopefully) more informed than a single classifier. At the cost of naturally requiring more computational resources than any of their singular components, ensembles in practice can generally provide improved predictive performance. Otherwise, they also improve robustness to anomalous data where members of the ensemble are more specialised to handle them, and improve probabilistic classification performance by averaging over the predictions of many classifiers. Inspired by previous work with ensembles and by the utility they demonstrate for our problem in Chapter 3, we propose a new ensembling scheme in Chapter 4, the Cross-validation Accuracy Weighted Probabilistic Ensemble (CAWPE). We evaluate it in terms of its general purpose classification performance on a wide range of datasets from two

public archives, as well as its application to our practical problem of forged spirits in Chapter 5.

Time series classification is the same supervised classification problem statement, but as applied to time series data in particular. Time series are ordered series of continuous values, and there is typically some information contained within the ordering of the time points themselves to be leveraged in mapping the series to its label. Typically the series are some measurement taken over time with a label attached, and can arise in many domains: sensor data, medical monitoring, weather, motion capture, econometrics, and computational biology to name a few. The measurements need not be strictly taken over time, however. The only requirement is that the readings are ordered. This means we can phrase spectroscopy data, which measures light received (or blocked) over wavelengths, as a time series problem.

Learning algorithms adapted to time series data, as opposed to e.g. tabular data, are typically concerned either with whole-series distances between existing data, or with finding and leveraging sequential patterns within the series. We develop a number of time series classification methods and evaluate their efficacy on the forged spirit problem relative to traditional chemometric and tabular-view machine learning approaches. The state of the art approaches of the time are evaluated on our initial datasets in Chapter 3, while in Chapter 5 we detail advances made towards a new state of the art for time series classification and its application to a dataset for determining methanol concentration in genuine spirits.

1.1 Contributions

The contributions described and support throughout this thesis can be summarised as follows:

- The description and prototyping of a non-invasive spirit analysis methodology based on near infrared spectroscopy and classification algorithms. Ultimately, we show in Chapter 5 that methanol concentration out of ten possible values in arbitrary spirits and bottles can be classified with an accuracy of 0.921.

- We generate three distinct datasets for future and public use in the literature. In Chapter 3, we describe datasets for the prediction of alcohol concentrations in synthesised alcohol-water solutions, which includes predicting correct ethanol concentrations and detecting the presence of methanol. We also present in Chapter 3 a dataset for distinguishing two particular whiskys in a given bottle. Lastly, in Chapter 5 we present a dataset of genuine spirits in their original bottles, progressively spiked with methanol. The dataset EthanolConcentration, derived from experiments of Chapter 3, already appears in the UCR archive of time series classification datasets and has been used as part of evaluations across the archive in numerous articles.

- In Chapter 4 we describe a novel heterogeneous ensemble scheme, CAWPE, and evaluate it across two public archives containing over 200 datasets. We show that on average for arbitrary datasets it outperforms alternative weighting, stacking, and ensemble selection schemes across identical base classifier sets, as well as homogeneous ensembles and heavily tuned classifiers. We show that using the CAWPE scheme to combine the different representations of HIVE-COTE results in a significantly improved ensemble, and forms part of the new state of the art for time series classification.

- In Chapter 5, we summarise improvements made to time series representations which also feed into the newest instantiation and current state of the art time series classifier, the HIVE-COTE 2.0 meta-ensemble. Aspects of this contribution have been in collaboration with other authors. This thesis contributes towards two members of the HIVE-COTE 2.0 ensemble (TDE and

DrCIF), as well as experimentation towards the selection and use of CAWPE as the meta-ensembling scheme for it. We give overviews of expansions to the BOSS algorithm to form SBOSS and ultimately TDE which forms the dictionary-based classifier within HIVE-COTE 2.0, and the improvements of TSF into CIF and ultimately DrCIF, the interval-based component. TDE and DrCIF now form the state of the art dictionary and interval representations respectively, as shown by experimentation conducted in part through this thesis. SBOSS is a contribution of the authors of this thesis alone, while work towards TDE, DrCIF, and ultimately HIVE-COTE 2.0 are led by others in collaboration with us. HIVE-COTE 2.0 is shown to be significantly more accurate on standard archive datasets than the previous state of the art, again as shown by experimentation conducted in part through this thesis. The aspects of HIVE-COTE 2.0 that are entirely not attributed to work undertaken within this thesis are the 'Shapelet Transform' and 'Arsenal' base classifiers of the ensemble.

1.2 Thesis Structure

In brief, we first cover the necessary background information in Chapter 2. For the contributions of this thesis, we detail in Chapter 3 the data collection and anaylsis of *synthesised* alcohol solutions using a non-invasive near infrared spectroscopy system as well as subsequent classification algorithm benchmarking. In Chapter 4, we present and evaluate a novel ensemble scheme with the aims of maximising predictive and probabilistic accuracy. In Chapter 5, we detail the collection and analysis of a further dataset of *genuine* spirits spiked with methanol under field-like conditions, and leverage the findings of the previous chapters. Here, we evaluate the methods developed in Chapter 4 as well as state of the art time series classification techniques developed in tandem with this thesis.

In more detail, each chapter is structured as follows:

Chapter 2 provides a technical background of vibrational spectroscopy, chemometrics, and the machine learning topics relevant to this thesis. We cover the necessary knowledge of the physical process that underlies the spectroscopy data being modelled, and focus in particular on spectroscopic methods that can collect non-invasive measurements and the data mining challenges that these pose. We provide an overview of existing and traditional chemometric approaches for the problem, and lay the groundwork for ensemble and time series classification approaches that we shall investigate. We also cover the experimental and evaluation methods used throughout the thesis.

Chapter 3 details the collection and evaluation of early through-bottle, non-invasive, near infrared spectroscopy datasets. Two problem statements are considered. First, whether alcohol concentrations (ethanol and methanol) can be determined regardless of the containing bottle. Second, whether two similar products from the same brand be distinguished. These are two factors that, if answered in the positive, demonstrate the feasibility of a portable, non-invasive system for detecting a variety of forged or adulterated spirits. It would suggest that poorly produced and potentially harmful spirits could be screened to within some limit of detection, and that a particular known brand, given example data of it, could be verified using such a system. We use the findings of this chapter to direct future algorithmic development in later chapters.

Chapter 4 develops an ensemble methodology aimed towards maximising probabilistic output over candidate classifier sets of an arbitrary problem/set of problems. Our hypothesis is that building ensembles of small sets of strong classifiers constructed with different learning algorithms is, on average, the best approach to classification for real-world problems. We propose a simple mechanism for building small heterogeneous ensembles based on exponentially weighting the

probability estimates of the base classifiers with an estimate of the accuracy formed through cross-validation on the train data. We demonstrate through extensive experimentation that, given the same small set of base classifiers, this method has measurable benefits over commonly used alternative weighting, selection or meta-classifier approaches to heterogeneous ensembles. We further extend the approach to include all models trained during the cross validation evaluation procedure of the base classifiers for improved robustness in predictions.

Chapter 5 presents the developments in the TSC space since Chapter 3, and their application to a larger-scale and more thorough methanol-spiking dataset. We first present improvements to the individual representations of the HIVE-COTE meta-ensemble, which when combined with the CAWPE ensembling scheme, are found to make significant improvements and constitute a new state of the art for time series classification. We then look to apply the developed techniques to a third new dataset. Methanol contamination poses a generalised problem that can persist in any product or market, and in Chapter 3 was discovered to difficult to detect accurately. We take a range of real spirits in their original bottles and progressively spike them with methanol, analysing them with the non-invasive near infrared spectroscopy setup. We show that the standard chemometric pipeline is unable to handle this data. We then thoroughly evaluate the developed TSC algorithms and the CAWPE ensemble from Chapter 4, among other competing algorithms, and show that while time series classification methods do not provide the benefit hypothesised, the CAWPE ensemble is the best approach of those evaluated.

Chapter 6 concludes the thesis, discusses the contributions put forward, and introduces possible future directions in the spirit authentication space.

Chapter 2

Background

This chapter introduces the relevant background materials for this thesis. We first give a brief descriptive background of vibrational spectroscopy methods, and focus on their application to authentication tasks in contrast to alternatives such as chromatography. We review previous spirit authentication and analysis in the literature, and motivate our investigations for fast, non-invasive authentication. We secondly summarise the chemometric techniques commonly covered based on the first section, and introduce the machine learning contexts, time series and ensemble classification techniques, that we shall advance and apply to the spectroscopic authentication problem. We finally describe the experimental procedure for classification experiments followed through the thesis, namely data sampling methods and frameworks for classifier evaluation and comparison.

2.1 Vibrational Spectroscopy

Vibrational spectroscopy (VS) is composed mainly of the two complementary analytical techniques, Infrared (IRS) and Raman (RS) spectroscopy. These are non–destructive, non–invasive tools that provide information about the molecular

composition, structure and interactions of a sample. A light source of some target wavelengths is shined towards the spectroscope either directly or via reflection. The intermediary sample interacts with the light, changing what is measured by the spectroscope. In particular, VS methods detect electronic changes in the internal vibrational energy levels of molecules, which are associated with the physical structure of a sample. The produced spectrum acts like a fingerprint signifying the contents, and can be used qualitatively and quantitatively for identification, characterisation, quality control and assurance.

In Infrared spectroscopy the sample is irradiated with polychromatic light, and a photon of light is absorbed when the frequency (energy) of the absorbed light matches the energy required for a particular bond to vibrate within the sample. The spectra produced measures the amount of light absorbed at each sampled wavelength.

Raman spectroscopy is based on an inelastic scattering effect. The sample is irradiated with monochromatic light and the photons are either elastically (the vast majority) or inelastically (<1%) scattered. The inelastically scattered light, known as Raman scatter, has lost or gained energy during this interaction and the emitted photon contains information about the molecular structure of the sample. The measured effect is the intensity of the Raman scattered light versus the energy difference, which is referred to as Raman Shift. Because potentially only one in a million photons will scatter, the overall intensity of Raman as opposed to Infrared is much weaker.

2.1.1 Infrared Spectroscopy

While there is no strict, physically defined boundary, the IR spectrum is typically split into three sections by convention; the near (NIR), mid (MIR) and far (FIR), defined by their position relative to the visible spectrum. The regions encompass

different wavelengths and energies, and therefore interact with the same molecules in different ways.

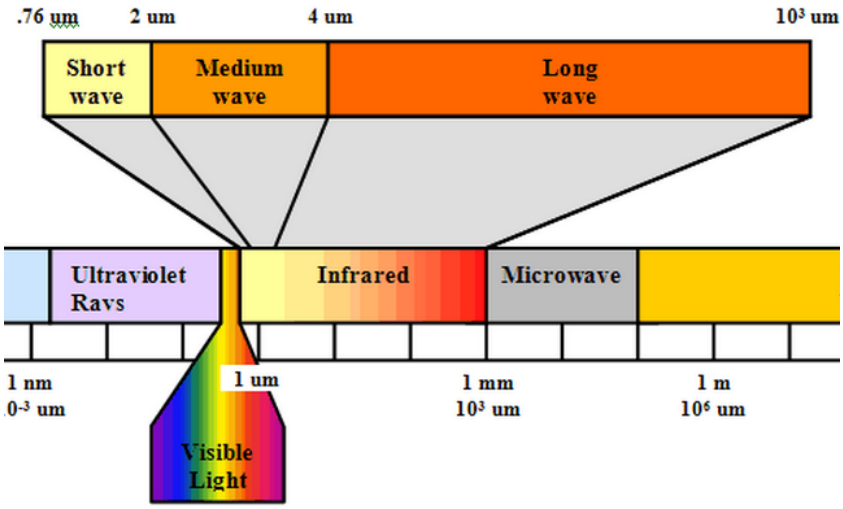


Fig. 2.1 The electromagnetic spectrum, with visible and infrared regions highlighted. Short, medium and long waves are synonymous with the near, mid and far regions. Image from [12].

Spectral bands in the NIR region are overtone and combination bands originating from fundamental bands in the MIR region. Both provide vibrational information, however, each has independent advantages and disadvantages that need to be considered for analysis [27]. NIR regions, being overtone and combination bands, are broad and have relatively low sensitivity and separation between components. A particular compound of interest may interact with many NIR wavelengths, and require inspection/modelling of all of them to detect it against other compounds. In contrast, spectral bands in the MIR are fundamental bands and typically the peaks indicating a compound are specific, sharp and sensitive. Most materials are strongly absorbing in the MIR region, however, making the likelihood of retrieving a clear signal through containers quite low. In contrast, glass, for example, is transparent to NIR radiation. This is a property that has been capitalised on by a number of industries, and is important of course for its choice in the alcohol authentication problem.

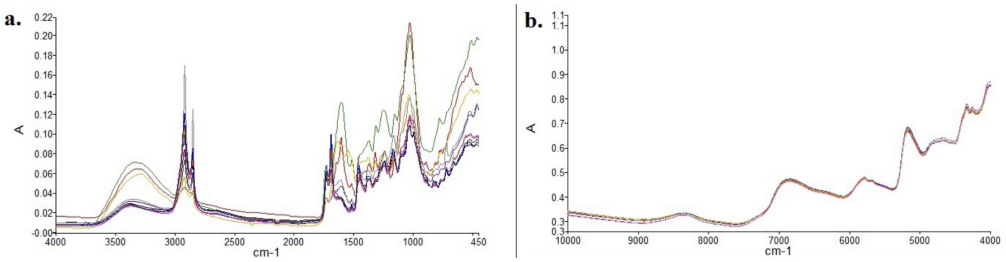


Fig. 2.2 Exemplars of near (a, left) and mid (b, right) infrared spectra characteristics. While the MIR region exhibits more isolated spikes corresponding to particular molecules, the NIR region presents molecules in a broader fashion with higher degrees of overlap and greater consistency between samples. Image from [79].

There are no fixed and globally accepted definitions of the extents of the different regions. However, the NIR band is typically defined as the range 700nm – 2500nm (~14000cm^{-1} - ~4000cm^{-1}), while the MIR band is 2500nm – 15000nm (~4000cm^{-1} - ~700cm^{-1}). The use of both is much researched within the food and drink research sector, due to IR spectroscopy's non-invasive and low operating-cost nature. IR suffers from high initial instrumental costs, however, and in many cases a lack of reliable and stable supplementary chemometrics for analysis [62]. Developments to introduce cheaper, more portable and more stable hardware have been consistent, in order to meet the growing demand for in-line and portable quality controls and verification needs [139].

Classical univariate spectroscopy methods in the UV and visible regions require physical separation of the substance of interest, usually by dissolution in a solvent. NIR spectroscopy combined with chemometrics/machine learning offers the potential for fast spectra collection and simplicity in sample presentation by learning to discover the underlying signal and ignore noisy artefacts [29].

2.1.2 Raman Spectroscopy

Raman scattering as a phenomenon has been known about since the 1930s. However, because the chances of inelastic scattering occurring are so low, practical use of it

has only been possible in the last few decades with the invention and increasing power of lasers. Most recently, further advances have led to the possibility of portable Raman devices and decreases in operational costs. Relative to IR however, it is still an expensive analytical technique, especially in terms of the initial costs of the instrumentation but also per sample to a larger degree than IRS.

In modern-day Raman spectroscopy, the main hurdles to overcome are the signal's susceptibility to be completely masked by fluorescence [22] and its extremely shallow penetration depth meaning that only the surface can be investigated. Many forms and extensions of Raman spectroscopy have appeared in the last two decades as it has become more practical. However, many are at most only only tangentially related to potential solutions to through-glass analysis, such as Surface Enhanced Raman Spectroscopy (SERS) [103] and Wide Area Illumination (WAI) Raman [67]. Of more interest is spatially offset Raman spectroscopy.

2.1.3 Spatially Offset Raman Spectroscopy

Standard Raman spectra can only give information about the surface of a sample or layer. Spatially Offset Raman Spectroscopy (SORS) is a technique for generating subsurface Raman spectra, giving information about deeper layers than conventional Raman can analyse. SORS involves the acquisition of two Raman spectra, one effectively representing the container and the second the subsurface photon, followed by a scaled subtraction which equates to the subsurface Raman spectrum. Figure 2.3 gives an example.

The SORS technique is effectively able to bypass, to a depth of several millimetres, any fluorescence or Raman signals originating from the surface layers [22], and relatively speaking enhances the Raman signal from the sub-layer. Therefore, the Raman spectra of individual sub-layers within a complex multilayer system can be isolated with a considerably small and simple experimental approach [94]. In

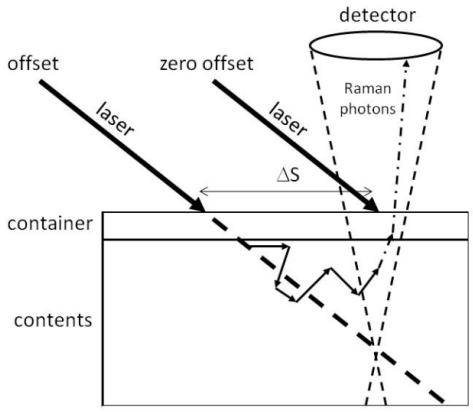


Fig. 2.3 SORS with an offset of ΔS, image from [14]

absolute terms however, the already relatively low intensity Raman scatter affect is in effect weakened once more by probing deeper into the surface.

Over the last decade since its invention, SORS has been the subject of much research in the pharmaceutical and security sectors in particular, for its ability to ascertain the contents of packaged medicine and pharmaceutical materials [14, 13], and concealed narcotics [109, 40] or explosives [63]. Developments are also in progress for the introduction of properly portable SORS instruments, particularly for use in airport security scenarios [117].

The potential problem for SORS in the context of this project however is that while its greatly diminished sensitivity is less of a problem for qualitative analysis in relatively simple samples, or more precisely where the target being searched for is a significant fraction of the sample, it may not be suitable for the discrimination of spirits within a relatively small distribution. Ethanol content could realistically be determined and discriminated upon. It is difficult to imagine though that the less abundant chemicals such as methanol and metals, which would be the focus of work coming from a public health standpoint, would be able to be reliably picked up when present.

If a potentially workable and cost-permissible piece of SORS hardware were to become available, experimentation with it would certainly be of interest, however.

2.2 Spectroscopy For Authentication

There is a wealth of research on predictive chemometrics applied to spectroscopy data for authentication applications, a selection of which are described below. In the context of food/drink research, wine is by far the most commonly research application. In comparison, whisky and most spirits appear to have had relatively little focus on them, despite distilled spirits being considered a larger industry worldwide [*]. The space of all distilled spirits, including home-made, is much larger than the commercial market, however.

Wine

Because the authentication of wine is such a similar problem, many indications of promising research avenues can be transferred from that existing body of research to the spirits domain. An overview of the practical application of VS to the analysis of wine is introduced in [29]. Included is a mention towards though-bottle analysis for the monitoring of of unwanted changes in the composition of the contents such as oxidisation. They believe however that such analysis can only be indicative rather than truly quantitative, because of the limits of NIR and its analytical accuracy.

A series of work on Australian wine, using MIRS to discriminate between varieties [11], a sample's organic status [30], and the use of NIRS to determine the concentration of various elements [31] provides further support to the idea of using spectroscopy in the discriminatory/predictive analysis of wine samples. This work culminates in a review by Cozzolino, which suggests the use of combined Visible and NIR spectroscopy for non-invasive and rapid indicative analysis of alcohol samples as a promising line of future research. This came with the warning though, that while NIR can tolerate longer path lengths, its low spectral resolution

[*] https://www.alliedmarketresearch.com/alcoholic-beverages-market

and therefore limited analytical accuracy means that it can likely be use only for qualitative analysis rather than quantitative.

Food and Agriculture

There is a huge array of work on the more general topic of food authenticity and quality control in terms of agricultural crops. Danezis et al. [32] reviews current and potential future techniques for food authentication. In their review, VS research is mainly focused on drinks and foods in liquid forms, such as oils and fats, dairy and wine, in comparison to other analytical techniques such as chromatographic and molecular methods.

Lohumi et al. [90] gives a review focused on VS techniques in the food industry. They note that its use has gained popularity over the last decade due to its suitability as a non-invasive, sensitive, and rapid analytical method for quality control and assurance. However, chemometric analytical methods for these techniques are still developing, and to some degree need to catch up. RS is also seeing a rise in prevalence and use in certain industrial scenarios. However, it cannot see much more use outside of lab conditions until costs go down and more complex variants of Raman spectra collection become feasible in a portable scenario.

Examples of particular applications include distinguishing the geographic origin of extra virgin olive oils [129], detecting the adulteration of strawberry purees [61], and the discrimination of coffee of different origins in instant coffee [19].

Drug detection

Eliasson et al. [40] tested the feasibility of detecting illegal drugs dissolved in alcohol for the purpose of smuggling using Raman spectroscopy, cocaine in rum in particular. A very difficult to detect method of drug smuggling is to dissolve the substance in a carrier solution, and then separate them once more at the target

destination. The authors tested a non-invasive SORS setup to detect 300g of 75% pure cocaine dissolved in 40% alcohol rum within its original bottle, and found that when comparing pure rum and rum-plus-cocaine samples, the scaled difference in the two spectra was identifiable as the signature of cocaine.

The authors noted that using the NIR band helped to combat intense fluorescence which can potentially swamp the Raman signal. Arbitrary bottles with different characteristics, e.g bottle shape, colour, and glass width, as well as the solution colour and viscosity, with much more fluorescence could be combated with specialised methods added on to the signal collection system such as Kerr gated Raman spectroscopy [89].

Later work by Burnett et al. [23] tested the detection of cocaine in a wider range of alcoholic solutions and bottles, and further collected data using a range of different instruments, one laboratory-based and two portable. They found that a general limit of detection in this scenario was roughly 8% w/v of cocaine in rum, in an arbitrary bottle. In this context, cocaine is often dissolved at much higher concentrations in order to reduce wasted materials during the smuggling process, and therefore the experiments were very promising.

In terms of transferring those methods over to the context of this project, however, this work may show the limitations of Raman and its low power and resulting low resolution. While we cannot directly transfer those results over to the fraudulent alcohol case, since this study was looking for the presence of one substance in another, if the limit of reliable detection was 8%, that suggests that current methods cannot be relied of for the level of precision we need.

Security and Explosives Detection

More general security applications of Raman spectroscopy and non-invasive substance detection has seen much attention in recent years. Izake [63] provides a

high-level overview of the strengths and limitations of various kinds of Raman spectroscopy in the general context of forensic and homeland security, while [91] reviews Infrared and Raman spectroscopy in various practical applications in the area.

A thorough investigation into optimizing SORS for the detection of substances concealed within opaque plastic and glass containers is given in [14]. As well as illustrating SORS's ability to bypass containers over conventional backscatter Raman, number of practical conclusions are drawn and guidance given. It is found that in most cases there exists a well-defined spatial offset that maximises the signal to noise ratio of the resultant spectrum, which is variable dependant on the container and content materials. However clear liquids or potentially other non-diffusely-scattering substances exhibit a wider optimal band, suggesting insensitivity to the offset. Further, container thickness seemed to have little effect on the optimal offset value. These points together provide compelling practical arguments for applying SORS to the fraudulent alcohol problem, however the low resolution of Raman as a general technique is still a problem to overcome.

While not as strictly relevant to the proposed use cases of this project, Izake et al. [64] demonstrates promising results using non-invasive 'standoff' Raman spectroscopy to detect explosives precursors/ingredients in highly fluorescing packaging from as far as 15m. Given that Raman's often quoted weaknesses are its relatively low resolution and susceptibility to fluorescence, the fact that technology able to leverage Raman's strengths at such distances does at the very least demonstrate the potential for Raman to develop as a technology and for hardware improvements to make it usable in a wider array of applications over the coming decades. Perhaps further applications arising from the work in this project can lead to distant detection of fraudulent alcohol as a larger scale system, for example alongside a conveyor belt at customs and imports.

Pharmaceutical Supply

Spectroscopic techniques for analytical chemistry are commonplace in the pharmaceutical sector. In particular non-invasive techniques have received much intention when looking for ways to automatically and rapidly verify the contents of ingredients for pharmaceuticals at the production stage [13, 67] and at the consumer distribution stage [82]. Such a depth and breadth of research into non-invasive means of analytical chemistry is promising. However, the caveats must be made that the substances being verified are typically much simpler in the pharmaceutical context, that is, having fewer constituent compounds, and that the typical problem in the pharmaceutical context (is the correct drug about to be given to this person?) is once more qualitative rather than quantitative.

2.3 Spirit Authentication

We now turn to literature on the authentication of spirits in particular. While the authentication of wine is such a similar topic, a limited amount of literature appears for the authentication of spirits.

2.3.1 Non-Spectroscopic Alcohol Authentication Methods

We have seen that VS methods have unique advantages that are desirable in authentication applications, however it can be limited in analytical power for situations where precision is required to make a quantitative decision. More time consuming and destructive techniques such as gas [54] or liquid [104] chromatography, are able to chemically fingerprint spirits with great accuracy. Other attempted analytical techniques include nuclear magnetic resonance spectroscopy [102], capillary electrophoresis [59], and artificial tongues [93], to name a few. All of these have

various strengths and weaknesses in terms of analytical power, setup costs, practicalities of usage, and familiarity, and all these factors naturally change over time as understanding and hardware production processes advance. We study VS, NIRS in particular, in this work largely to leverage one of its main attractive strengths within the context of detecting forged alcohol; the ability to measure samples within-bottle, which the others above cannot.

2.3.2 Vibrational Spectroscopy Applications to Alcohol Authentication

We separate VS-specific literature into those performing analysis with direct contact to the sample first, and those performing through-bottle analysis second.

2.3.2.1 Direct Contact with Sample Required

Numata et al. [107] performed experiments to determine the feasibility and practical implications of quantitative analysis of binary alcohol-water solutions using Raman spectroscopy. Acetonitrile was used as a reference sample, with a reading of it being taken before each reading of a binary solution. To determine the alcohol concentration, the ratio between the band of alcohol in the binary solution and the reference sample is calculated. The authors posit that the use of reference readings is a requirement for quantitative analysis, otherwise the intensity of the band may not be directly proportional to the alcohol concentration. As well as the sample itself, the Raman intensity depends on several instrumental conditions such as laser power and ambient environmental conditions. The authors demonstrated in earlier work the dependency on laser power of the absolute Raman intensity, however the band ratio is a constant, independent of the laser power [108]. The authors found excellent correlation between the band ratios and the mass fraction (concentration) of ethanol and methanol in water, $R^2 = 0.9996$ in methanol–water, and $R^2 =$

1.000 in ethanol–water. In ethanol-methanol solutions, calibration curves with $R^2 = 0.9992$ and $R^2 = 0.9999$ for ethanol and methanol respectively were found, suggesting that, even though they appear in close spectral bands, discrimination between them and determination of their levels is still accurate. This is important for the measurement of full samples, which would contain many more confusive compounds albeit in (hopefully) lower concentrations. Lastly, the authors found that even if the acetonitrile standard was measured a single time, as opposed to before every reading, the calibration curve formed from the band intensity ratio shows good linearity but still with a reduction in R^2. Therefore, the standard should be measured and used for each sample.

MIR spectrometry with an attenuated total reflectance (ATR) probe, with direct contact to the sample, is used to detect counterfeit Scotch whisky samples in McIntyre et al. [95]. MIR data were processed using a Savitzky–Golay first derivative filter, which employed a width of 7 data points and a second order polynomial. Regions in the data that would only contribute noise to the measurements were removed, however the exact process to achieve this was not specified. In one set of experiments, ethanol concentration was measured using univariate and multivariate PLS models. The authors found that it was possible to predict the concentration of ethanol in the whisky samples with an average relative error of 1.2% and 0.8%, respectively. In a second set of experiments authenticity of seventeen whisky samples was determined using a combination of predicted ethanol level and principal component analysis (PCA) of the spectra to investigate the colorant added. It was found that neither method alone was enough to determine authenticity, but with both methods combined the seventeen samples were correctly discriminated as legitimate or illegitimate.

Li et al. [83] tests the use of Visible-NIR spectroscopy for the discrimination of Chinese Liquors based on the factors of brand, age, flavours and alcohol levels. These experiments required direct access to the sample, however the authors intend

to do further work involving spectra collection through bottle. Support vector machine (SVM), Soft Independent Modeling of Class Analogy (SIMCA), and Linear Discriminate Analysis based on Principal Component Analysis (PCA-LDA) were tested on 730 samples of 22 kinds, ten brands, and six flavors. Contrary to many other studies, raw, unfiltered spectra led to the highest classification accuracy. Derivative processing, which was expected to correct additive and multiplicative effects in the spectra sharply decreased accuracy of models. PCA-LDA achieved the best results with a mean accuracy of 98.94% in the training set and 95.70% in the test set. The percent correctly classified were all in the range of 95.65–100% in the discrimination of different brands, alcohol levels, ages, and flavours.

Wu et al. [142] considers the use of MIR-ATR spectroscopy to monitor the levels of the main chemical parameters involved in the fermentation process of Chinese rice wines. The MIR spectra are once again Savitzky–Golay filtered, using the first derivative for all readings after a short comparison between different parameters of the filter on the raw data. PLS, SVM with a Radial Basis Function kernal, and versions of the two on optimised intervals found through i-PLS are tested. The SVM on the optimised interval (i-SVM), which reduced the number of attributes from 1660 to 83, was found to be the most accurate, with root mean squared errors of prediction (RMSEP) of 6.92, 3.32, 3.24, and 6.33, for total sugar, ethanol, total acid, and amino nitrogen respectively.

2.3.2.2 Through-Bottle

NIR and Raman spectroscopies are compared for their suitability to determine of alcohol content in whisky and vodka contained within clear and coloured glass bottles in Nordon et al. [106]. Univariate regression models for each type of drink were calibrated for the Raman data using the signal at $873cm^{-1}$ in the first derivative spectrum, while a multivariate PLS was calibrated for the NIR data.

The latter calibration procedure involved some optimisations on the test data, and therefore the results specifically should be treated with caution. However, the higher level conclusions in terms of the relative difficulty of different aspects of the experiments are still insightful.

Differences between bottles accounted for the greatest variation and difficulty in the analysis, relative to differences in bottle positioning and time of measurement. Coloured glass made analysis particularly difficult, due to the effect of large amounts of fluorescence on the spectra. Using the doubly-transmitted NIR method, a signal could not be collected from the widest part (70mm path length) of the largest bottles, whereas comparable signals to that of the smallest bottles could be found by measuring through the neck of the bottle (40mm path length).

Kiefer et al. [66] study the ability for Raman spectroscopy to discriminate between certain Scotch Whisky production factors from within their original containers is tested. 44 whisky samples were measured directly through the glass walls using an Avantes Raman instrument. The authors suggest that the location of measurement (from the neck, base or centre) had no influence on the quality of the readings, in contrast to previous literature. The stability of their sampling suggested excellent reproducibility, with normalised spectra being 'virtually identical'. Principal Component Analysis (PCA) could distinguish between the type of cask each whisky was matured in, but otherwise had limited separability.

In experiments with PLS Regression (PLSR) through leave-one-out cross validation, a quantitative analysis of important factors related to authentication was described: age; ethanol concentration; and the presence of artificial colourings. The age of samples between 3-22 years could be estimated to within 0.42 years. On average ethanol concentration could be estimated to within 0.44%, which is only just outside the regulatory limits of Scotch Whisky (0.3%).

Two outliers were found with reported ethanol concentrations differing greatly from their label. These were explained by the evaporation of alcohol due to a failed seal, and repeated opening of the bottle, respectively, over longer periods of time. These are very strong results, suggesting the feasibility of quantitatively determining key factors to whisky authentication. Ethanol level determination is perhaps less surprising, as it does form 40-55.8% by vol of the samples, but determining unintuitive properties like age within such an accuracy is a positive result.

Successful studies towards the same goal using handheld SORS are described in [41]. SORS enables the use of Raman spectroscopy through thin surfaces, glass bottles in this case. Experiments with spirits in identical, 2.3mL clear glass vials and a thirty second acquisition time with the SORS device demonstrated the strength of the process by determining the concentrations of ethanol, methanol and other less abundant compounds present in real or adulterated spirits. Further experiments with three off-the-shelf, branded, and variously-coloured 50mL bottles revealed seemingly less reliable, but still well within tolerable levels, detection of methanol concentration over a ninety second acquisition time.

Our experiments described later take measurements in full-size bottles (typically 700mL) and have a one second acquisition time. The results of these methods are therefore are not directly comparable. However, that such results are achievable with non-invasive spectroscopic methods is promising.

Continuing on from these works, our own investigation into this problem focuses on portability, simplicity, and speed in all aspects of the analysis of a sample. The final aim is to allow a non-expert to determine with sufficient confidence the authenticity of an arbitrary spirit on-site and within seconds. The previous works listed here provide a solid basis on which to continue, and to apply and evaluate

more modern and expansive machine learning techniques in the context of higher-throughput data collection procedures.

2.3.3 Chemometrics

The underlying theme in essentially all of the authentication works covered in the previous section is the pipeline of: smoothing and/or filtering the signal in some form; performing some form of interval selection and/or non-linear dimensionality reduction (potentially included internally in the following modelling process); ultimately predicting through the use of PLS regression (PLSR) for the most part. Figure 2.4 summarises this process.

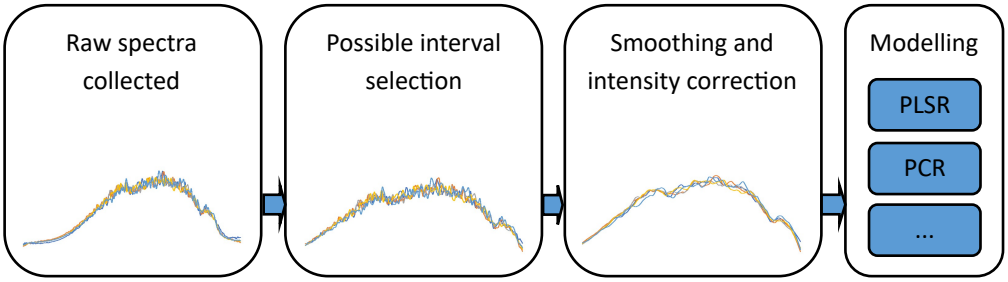


Fig. 2.4 An overview of a standard chemometric pipeline, applied to example spectra.

Savitzky–Golay (SG) is most frequently used for smoothing, and sometimes also for simultaneously taking the first or second order derivatives to remove the effects of total illumination. Suitable parameters for the filter would need to be found for any individual dataset, as an optimal amount of smoothing would be dependant on the signal to noise ratio. The amount of noise present, of the type best handled by smoothers such as SG, is generally a function of the variability of the total light source, the integration time for each spectra, and the averaging of sub-spectra to take each reading. Light source stability comes down to the engineering quality of the intended light source and the (mostly imperceptible to humans) variance of ambient lighting brought about through e.g. atmospheric

effects. The integration time is simply how long light is collected for, similar to a camera. Lastly, many spectroscopic software packages will average over readings automatically behind the scenes. A one second reading may be five averaged readings of 0.2 second integration times each, for example. These factors will play into the amount of smoothing that may or may not be needed.

As demonstrated by Wu et al. [142], but also intuitively, selection of the most informative interval (wavelength band) significantly improves accuracy. Much of the time, where particular substances are being search for, wavelength bands are selected through domain knowledge of known resonant frequencies. For more complex or multiple substances, or for indicative readings of higher level properties (e.g. age or cask, from above), automated feature selection could be used. In the context of our spirit authentication problem, while methanol level is a more concrete marker for safety and legitimacy, ethanol level alone will likely not provide enough discriminatory power to discover real world forgeries. Using colour, via the visible spectrum, in combination with the NIR bands where ethanol and methanol appear, may well prove to be strong discriminators for general legitimacy. The problem with leveraging colorants in the context of this project is that, depending on the particular use case pursued, the nature of the bottle can vary wildly from sample to sample. Bottle colour, shape, reflectiveness, and ambient lighting all affect the visible spectrum to a far greater degree than they affect the NIR.

PLS regression appears to be the classical yet accurate method that most authors fall back on, at least as an initial model. However some authors are using more recent and complex algorithms such as SVMs [83, 142] and Random Forests [124, 81] where factors such as container variability come into play. Simple (by modern standards) neural networks have been considered in some older works [120, 105], but deeper, more complex networks are generally limited by the smaller data scales available to individual labs. More recent works such as [145] look to incorporate modern convolutional architectures, in this case Inception modules [127]. However,

in practice, we have found the use and reproduction of a number of particular architectures troublesome. In our experiments later throughout Chapters 3 and 5, we adopt convolutional networks originally designed for time series classification due to their proven general utility and available implementations. These are discussed later, in Section 2.4.2.7.

Linear systems on reduced attribute spaces work satisfactorily for clean spectra collected under professional and standardised conditions. However, they may be unable to handle any non-linear structural changes in the data described. Further, spirits and especially whiskies are particularly complex in their chemical compositions, to their benefit as a final product. Depending on the manner of forgery, adulteration, or contamination, discriminatory information may be relatively easy or very difficult to extract.

2.3.4 Calibration Transfer

An important factor to consider in a fielded system is calibration transfer between different spectral hardware and, more generally, ensuring that the data collected by two or more different spectroscope devices are mutually useful in modelling the global problem posed. For the most part, calibration transfer comes down to the standardisation of instrumental responses through the comparison of spectra collected of representative sample sets, called transfer samples. Standardisation adjusts the response of one instrument to the other, and once adjustments have been learned, can be applied as a preprocessing step to spectra of future genuine samples. A typical scenario would be that a system is developed on bench-top, lab-based equipment (the 'primary' instrument). Later, portable devices for use in the field ('secondary' devices) need to be calibrated to produce equivalent responses such that models trained on data from the primary instrument can maintain their performance.

This topic is not covered in further depth throughout this thesis, largely due to the local practical consideration of equipment costs. For future use of as developed system in the field, however, it is an important aspect to consider.

2.4 Classification Methods

When considering particular aspects of the spirit authentication problem such as the concentrations of alcohols to inform decision of authenticity, the use of regression models makes sense. However, through consultation with industry, the ultimate use case designed to aid field use is a traffic light classification scheme; green (genuine), yellow (suspect), and red (forged). The confidence thresholds for each class can be set by the user in response to factors such as the costs of verification and screening. Typical regression models can of course still be used though, since regression can be reduced to classification through discretisation of the output space. In the case of alcohol concentration, for example, the output space can be discretised to correct concentration (according to that reported on the label) and not, to within some acceptable limit.

The unique challenges of the domain in question suggest more powerful modelling methods may be required than those described as being typically used previously. Non-standard containers, variable environmental conditions, portable devices with lower power than bench-top solutions, and the relative homogeneity of the sample properties being distinguished under our experimental conditions all work to make the problem more difficult. Here, we shall first define the standard notation used throughout the thesis when working with data and classifiers, and then give overviews of two broad classification methods we shall develop and apply to the domain: ensembles and time series classification.

We use the following notation. A dataset $\boldsymbol{D}$ of size n is a set of attribute vectors with an associated observation of a class variable (the response), $\boldsymbol{D} = \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n)\}$, where the class variable has c possible values, $y \in \{1, \ldots, c\}$ and we assume there are m attributes, $\boldsymbol{x_i} = \{x_{i,1}, \ldots, x_{i,m}\}$. A learning algorithm L, takes a training dataset $\boldsymbol{D}_r$ and constructs a classifier or model M, which is evaluated on a test dataset $\boldsymbol{D}_e$. A classifier M is a mapping from the space of possible attribute vectors to the space of possible probability distributions over the c valid values of the class variable, $M(\boldsymbol{x}) = \hat{\boldsymbol{p}}$, where $\hat{\boldsymbol{p}} = \{\hat{p}(y = 1|M, \boldsymbol{x}), \ldots, \hat{p}(y = c|M, \boldsymbol{x})\}$. Given $\hat{\mathbf{p}}$, the estimate of the response is simply the value with the maximum probability.

$$\hat{y} = \underset{i \in \{1,\ldots,c\}}{\arg\max}\, \hat{p}(y = i|M, \boldsymbol{x}).$$

2.4.1 Ensembles

An ensemble of classifiers is a collection of classifiers trained to solve the same overall problem whose predictions for new data are combined to produce a single prediction for the whole ensemble. Many factors are at play when selecting and designing a classifier to use for a decisioning system, and these are multiplied when considering ensembles. Five key factors can be defined when considering ensemble systems [119, 73], the way in which to combine the outputs of base classifiers, how to train base classifiers both in the learning algorithm used and whether each classifier is dependent or independent of the others, how to ensure diversity between base classifiers, how large the ensemble should be and whether this is dynamic or fixed, and whether any specific base classifier learning algorithm is required or the choice is arbitrary.

The two key concepts in ensemble design which we are most interested in for the experiments of Chapter 4 are the necessity for classifiers in the ensemble to

produce diverse predictions [38, 110, 52, 57] and how to combine the outputs of the models.

For the former, an ensemble needs to have classifiers that are good at estimating the response in areas of the attribute space that do not overlap too much. That being said, there is no single precise definition or measure of diversity accepted throughout the literature, with dozens of different candidates having been proposed [72, 128]. Further, it has been argued that diversity is a necessary but not itself sufficient condition of a strong ensemble [74], with conditions of minimal performance of the base classifiers and suitable combination methods playing a role. Base classifiers that each produce entirely random predictions will likely be very diverse, but not very useful. In this vein, diversity may be broken down into 'good' and 'bad' diversity for ultimate predictive performance, and be related to the choice of performance metric and combination scheme [20]. Regardless of whether diversity itself can be optimised for, its presence is a requirement in any ensemble that is to be better than its base classifiers. In the opposite case of the previous hyperbolic example, ensembling many classifiers that all predict the same thing is no better than using just one of them. Broadly speaking, diversity can be engineered by changing the training data or parameters given to the same learning algorithm to form a homogeneous ensemble, or by employing different learning algorithms to train each base classifier, forming a heterogeneous ensemble.

On combining models, a top-level taxonomy of non-trainable, trainable, and meta-classifier combination methods can be defined [73]. Non-trainable combination methods would include taking the means of the outputs of the base classifiers, or taking the single most confident prediction in a probabilistic setting. Trainable combination methods include weighting schemes where weights need to be learned and assigned, or even classifier selection approaches where the single classifier to use needs to be picked. Meta-classifiers, also described as stacking, go further

by taking the outputs of the base classifiers and using them as inputs to a further classifier (-system).

Extending the notation defined above, an ensemble E is a collection of classifiers $\boldsymbol{E} = \{M_1, \ldots, M_k\}$ built by a set of (possibly identical) learning algorithms $\boldsymbol{L} = \{L_1, \ldots, L_k\}$ which train on (possibly different) sets of train data. An ensemble algorithm involves defining the learning algorithms $\boldsymbol{L}$, the data $\boldsymbol{D}$ used by each learning algorithm to produce the models $\boldsymbol{E}$ and a mechanism for combining the output of the k models for a new case into a single probability distribution or a single prediction.

2.4.1.1 Heterogeneous Ensembles

Heterogeneous ensemble design relies more heavily on diversity through the use of different learning algorithms, and focuses on how to use the output of the base classifiers to form a prediction for a new case. i.e., given k predictions $\{\hat{y_1}, \ldots, \hat{y_k}\}$ or k probability distributions $\{\hat{\boldsymbol{p}}_{\boldsymbol{1}}, \ldots, \hat{\boldsymbol{p}}_{\boldsymbol{k}}\}$, how to produce a single prediction $\hat{y}$ or probability distribution $\hat{\boldsymbol{p}}$. There are three core approaches: define a weighting function on the model output (weighting schemes); select a subset of the models and ignore other output (ensemble selection schemes); or build a model on the training output of the models (stacking) [116].

Weighting Schemes

Weighted combination schemes estimate a weight w_j for each base classifier and then apply it to their predictions. Base classifier predictions multiplied by some weight are summed,

$$s_i = \sum_{j=1}^{k} w_j \cdot d(i, \hat{y_j})$$

where

$$d(a,b) = \begin{cases} 1, & \text{if } a == b \\ 0, & \text{otherwise} \end{cases}$$

then the class with the highest weighted prediction is chosen

$$\hat{y} = \underset{i \in \{1,\ldots,c\}}{\arg\max}\, s_i.$$

Based on the framework described in Kuncheva and Rodríguez [71], we concentrate on four weighting schemes, which are described as following on from one another when relaxing assumptions about base classifiers' performance.

1. Majority vote (MV): $w_j = 1$ for all base classifiers.

2. Weighted majority vote (WMV): w_j is set as an estimate of the accuracy of the base classifier found on the train data.

3. Recall (RC): Rather than a single weight w_j, a separate weight is assigned to each class $w_{i,j}$. This weight is set to be the proportion of cases correct for that class on the training data (the true positive rate/recall/sensitivity).

4. Naive Bayes Combiner (NBC). The Naive Bayes combiner uses the conditional distributions to form an overall distribution, assuming conditional independence.

   $$\hat{p}(y = i|\{\hat{y_1},\ldots,\hat{y_k}\}) = \hat{p}(y = i|\hat{y_1}) \cdot \hat{p}(y = i|\hat{y_2}),\ldots,\hat{p}(y = i|\hat{y_k})$$

   where the probability estimates are derived directly from the cross-validation confusion matrix of the train data. The final prediction is the index of the maximum probability.

Ensemble Selection

A popular approach is to use a heuristic to select a subset of classifiers. Also referred to as an overproduce and select strategy or ensemble pruning, it was initially proposed for ensembles of diverse neural networks [113], but later became generalised to other classifier types [53]. The approach became known to a wider audience after the landmark paper by Caruana et al. [25], which describes the algorithm we compare to in Chapter 4 and call ensemble selection (ES).

Given a set of base classifiers, ES uses forward selection to progressively build the ensemble, selecting the classifier at each stage that gives the largest improvement to the ensemble's performance, or stopping when no improvement can be made. This process has a large potential for overfitting, and so this is mitigated through three strategies: selecting with replacement allows for the incorporation of good models multiple times, instead of being forced to select poor models sooner that may by chance improve ensemble performance on the current set; initialising the ensemble with a subset of the best classifiers in the pool gives a strong and reasonable start to the process; and lastly, repeating the selection process multiple times on bagged subsamples of the set of base classifiers before aggregating into a final ensemble gives the inter-relationships between different sets of models more chances to be recognised.

Stacking

The third popular approach to building heterogeneous ensembles is stacking [141]. This involves taking the output of the base classifiers on the train data, then applying another learning algorithm to determine how to best combine the outputs to predict the class value. Thus the cross-validation on the train data produces a set of predictions or probabilities for each case from all ensemble members and a further classifier is then trained on this output. New cases are classified by first producing

the output of the base classifiers, then passing these outputs to the meta-classifier to form a prediction. The first stacking algorithm to gain widespread usage was stacking with multi-response linear regression (SMLR) [131]. Two extensions to SMLR were proposed in [39]. These were stacking with multi-response linear regression on extended features (SMLRE) and stacking with multi-response model trees (SMM5).

HESCA

Lines et al. [87] describes a particular instantiation of an ensemble (base classifier set and method of combination). The Shapelet Transform (discussed later, Section 2.4.2.3) for time series classification, generates a dataset that is transformed from time series to tabular data. A general purpose classifier was desired to generate predictions from the transform, which was accurate on test data but also could generate a reliable estimate of performance on the train data. The Heterogeneous Ensemble of Standard Classification Algorithms (HESCA) was designed to fill this role. HESCA includes eight constituent classifiers, two of which themselves are ensembles: k Nearest Neighbour; Naive Bayes; C4.5 decision tree; Support Vector Machines with linear and quadratic basis function kernels; Random Forest (with 500 trees); Rotation Forest (with 50 trees); and a Bayesian network. These classifiers were chosen to give a balance between probabilistic, instance-based, and tree-based classifiers. The intention behind this was to create a simple, untuned, ensemble that was immediately diversified by the nature of its base classifiers. An evaluation on 72 datasets from the UCI archive (Section 2.5.1.1) confirmed its utility against its own base classifiers, although Rotation Forest itself was close in performance. The benefit of HESCA, however, was shown when used to classify time series data from the Shapelet Transform, where it was clearly superior [87].

HESCA, although proven experimentally prior to use, was conceptualised relatively simplistically. The selected base classifiers had thought put into them for their diversity in theory, but were not optimised empirically for it. Likewise, most of the classifiers are known to be generally strong on arbitrary tabular data, but the particular classifiers selected for HESCA were not optimised for accuracy. Put simply, a generically strong classifier was required to form predictions from the transform, which was designed to do the heavy lifting of time series feature extraction. HESCA formed the early basis of our interest in ensembles over small heterogeneous classifier sets, and eventually leads to the development of a new ensembling scheme in Chapter 4.

2.4.1.2 Homogeneous Ensembles

Homogeneous ensemble design focuses more on how to diversify the base classifiers than on how to combine outputs. Popular homogeneous ensemble algorithms based on sampling cases or attributes include: Bagging decision trees [17]; Random Committee, a technique that creates diversity through randomising the base classifiers, which are a form of random tree; Dagging [130], which trains base classifiers on disjoint stratified folds of the data; Random Forest [18], which combines bootstrap sampling with random attribute selection to construct a collection of unpruned trees; and Rotation Forest [118], which involves partitioning the attribute space then transforming in to the principal components space. Of these, we think it fair to say Random Forest is by far the most popular. These methods combine outputs through a majority vote scheme, which assigns an equal weight to the output of each model.

Boosting ensemble algorithms seek diversity through iteratively re-weighting the training cases and are also very popular. These include AdaBoost (Adaptive Boosting) [45], which iteratively re-weights based on the training error of the

base classifier; Multiboost [138], a combination of a boosting strategy (similar to AdaBoost) and Wagging, a Poisson weighted form of Bagging; LogitBoost [47] which employs a form of additive logistic regression; and gradient boosting algorithms [46], which have become popular through the performance of recent incarnations such as XGBoost [26]. Boosting algorithms also produce a weighting for each classifier in addition to iteratively re-weighting instances. This weight is usually derived from the the training process of the base classifier, which may involve regularisation if cross-validation is not used.

Random Forest

A Random Forest is composed of k trees, with each tree being trained on a random subset of the instances of the training set. At each node in the tree, a random subset of attributes is selected, and the best attribute in the sample is selected for partitioning the data. The Random Forest therefore generates diversity between its constituent classifiers (trees) through random sampling of both instances and attributes of the dataset. The hyperparameters controlling the sample sizes create a trade-off between the inter-dependence of trees, with higher diversity driven by lower values, and the strength of individual trees, with individual prediction performance generally increased via access to more data. In practice, Random Forests are often viewed as robust to these parameters when sufficient trees are built (typically 500) [18, 44].

Rotation Forest

Rotation Forest is similarly an ensemble of trees, however its mechanisms for generating diversity via the instances and attributes differs. For each tree, all attributes are randomly partitioned into r distinct groups. For each group, instances are randomly sampled with replacement from a random subset of the classes. A

principle component analysis is performed on each group, and the coefficients learned from the instance subset used to transform the full instance space. All r transformed groups of attributes are recombined to form the new dataset, which a standard tree is trained on.

Gradient Boosting

Gradient Boosting, for which we make use of the XGBoost package for practical implementation, is an approach where each new model is created to predict the residual errors of previous models, and concatenated to make the final prediction. Each tree is an intentionally simple and weak learner, limited primarily by its max depth to typically between two to ten, which is treated as a tune-able hyperparameter. Gradient descent is used to minimise loss and controls the tree addition process. XGBoost as an implementation largely introduces regularisation techniques to the loss, tree pruning strategies, and implementation optimisations to leverage hardware resources and parellelisation.

2.4.2 Time Series Classification

Time series classification (TSC) is concerned with learning techniques for data recorded consistently over some variable (typically time) in particular. A time series is a set of ordered and numeric attributes. Beyond the regular tabular data description of $\boldsymbol{D}$ previously, in a time series context there can be discriminatory information embedded within the ordering of attributes itself in the form of shape and autocorrelation. The description of $\boldsymbol{D}$ holds for univariate series, where each instance $\{\boldsymbol{x_i}, y_i\}$ has one time series $\boldsymbol{x_i}$ associated with each label y_i. This can be generalised to the multivariate time series classification (MTSC) case, where multiple time series are associated to one label. In MTSC, each instance is a list of vectors over d dimensions and m observations, $\boldsymbol{X} =< \boldsymbol{x_1}, \ldots \boldsymbol{x_d} >$, where

$\boldsymbol{x_k} = (x_{1,k}, x_{2,k}, \ldots, x_{m,k})$. We denote the j^{th} time step of the i^{th} instance of dimension k as the scalar $x_{i,j,k}$.

The has been a large increase in the prevalence of TSC literature over the last decade. This is due to the ubiquity of time series data in practical applications providing motivation to improve modelling methods, as well as the practical experimental benefits brought through the introduction and continued expansion of the UCR univariate (Section 2.5.1.1) and UEA multivariate [3] TSC dataset archives. For a time, the received wisdom was that whole-series distance comparison using dynamic time warping (DTW) [115] with a nearest neighbour classifier was the gold standard and difficult to beat on average. In 2017, a large-scale evaluative comparison of proposed classification techniques on a wider set of datasets found that a number of algorithms could significantly outperform the DTW benchmark [6]. It also defined a taxonomy of time series representations, which leverage different features of time series through transformation into alternative domains. This taxonomy has morphed and expanded since that study. We describe and group algorithms in a similar way to aid contextual understanding of how different algorithms fundamentally operate.

2.4.2.1 Algorithms based on raw series

Techniques based on raw series compare two series either as a vector (as with traditional tabular classification) or by a distance measure that uses all data points. In the latter case, measures are typically combined with one-nearest-neighbour (1-NN) classifiers and the simplest variant is to compare series using Euclidean Distance. However, this baseline is easily beaten in practice on all but the simplest and most carefully aligned datasets, and most research effort has been directed toward finding techniques that can compensate for small misalignments between series using specialised elastic distance measures. The almost universal benchmark for

whole-series measures is Dynamic Time Warping (DTW) but numerous alternatives have been proposed. The most accurate whole series approach as of the bakeoff comparison was the Elastic Ensemble (EE) [85], an ensemble of 1-NN classifiers using various elastic measures, including DTW, combined through a proportional voting scheme.

2.4.2.2 Interval-based algorithms

Rather than use the whole raw series, the interval class of algorithm selects one or more phase-dependent intervals of the series. At its simplest, this involves a feature selection of a contiguous subset of attributes. However, the three most effective techniques generate multiple intervals, each of which is processed and forms the basis of a member of an ensemble classifier [36, 9, 8]. There is no significant difference in accuracy between these approaches, but the simplest and most widely adopted is the Time Series Forest (TSF) [36].

TSF aims to capture basic summary features from intervals of a time series. For any given time series of length m there are $m(m-1)/2$ possible intervals that can be extracted. TSF takes a random forest-like approach to sampling these intervals. For each tree, k intervals are randomly selected, each with a random start position and length. Each interval is summarised by the mean, standard deviation and slope, and the summaries of each interval are concatenated into a single feature vector of length $3k$ for each time series. A decision tree is built on this concatenated feature vector. New cases are classified using a majority vote of all trees in the forest.

CIF [97], the Canonical Interval Forest, and the subsequent **DrCIF** [99], the Diverse Representation Canonical Interval Forest, expand on TSF and shall be discussed in Chapter 5.

2.4.2.3 Shapelet-based algorithms

Shapelets were first introduced by Ye et al. [143]. Shapelet approaches are a family of algorithms that focus on finding short patterns (shapelets) that can appear anywhere in the series, whose presence indicate class membership. Likelihood of presence is typically determined by the so-called sDist, which slides the shapelet along the time series of interest, and finds the location and computed distance of the location from which the subsequence of equal length to the shapelet has the minimum Euclidean distance. This is illustrated in Figure 2.5. Low sDist implies that the pattern described by the shapelet is present in the series, and vice versa. The main difficulties are then finding informative shapelets that discriminate between classes well, and the means of leveraging them for classification.

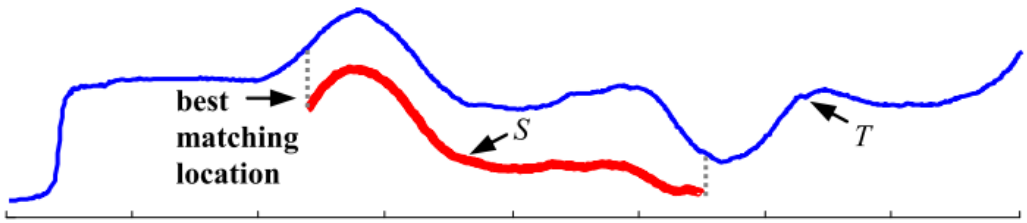


Fig. 2.5 Illustration of a shapelet S, being compared to time series T. A start location in T is found where S has minimal Euclidean distance to the subsequence of the same length. Image from [143].

The two leading ways of finding shapelets are through enumerating the candidate shapelets in the training set [86, 60] or searching the space of all possible shapelets with a form of gradient descent [56]. The bakeoff found that the shapelet transform algorithm used in conjunction with a heterogeneous classifier ensemble (ST-HESCA) is the most accurate approach on average. The transform searches for shapelets, and then computes the distance between each series and each shapelet to create a new (tabular) dataset for the training of a standard classifier. In more recent works, the classifier trained on the transformed data has been a modified and contracted rotation forest [1] instead of HESCA [4] for greater usability. As

is done there, we refer to this version as the Shapelet Transform Classifier (STC) henceforth.

2.4.2.4 Dictionary-based algorithms

Shapelet algorithms look for subseries patterns that identify a class through presence or absence. However, if a class is defined by the relative frequency of a pattern, shapelet approaches will be poor. Dictionary approaches address this by forming frequency counts of repeated patterns. They approximate and reduce the dimensionality of series by transforming into representative words, then compute similarity by comparing the distributions of words between series. Figure 2.6 illustrates the type of data dictionary-based algorithms should particularly be strong on. Correctly parameterised (length of pattern, and severity of smoothing/discretisation) transforms will be able to capture the patterns and form counts of each, separating the classes.

As of the bakeoff, three of the approaches had been published in the data mining literature are: Bag of Patterns (BOP) [84]; the Symbolic Aggregate Approximation Vector Space Model (SAXVSM) [125]; and the Bag of Symbolic Fourier Approximation Symbols (BOSS) [121]. BOSS was among the most accurate single-representation classifiers, and the only of the dictionary approaches to significantly beat DTW.

BOSS uses Symbolic Fourier Approximation (SFA) [122] to discretise sliding windows into words. SFA first finds the Fourier transform of the window, then discretises the first l Fourier terms into α symbols to form a word using a bespoke supervised discretisation algorithm. Histograms of the words in each series are formed, and predictions made using nearest neighbour between the historgrams with a bespoke distance function that considers only words contained in the test instance's histogram (i.e. the word count is above zero). Otherwise, it is Euclidean

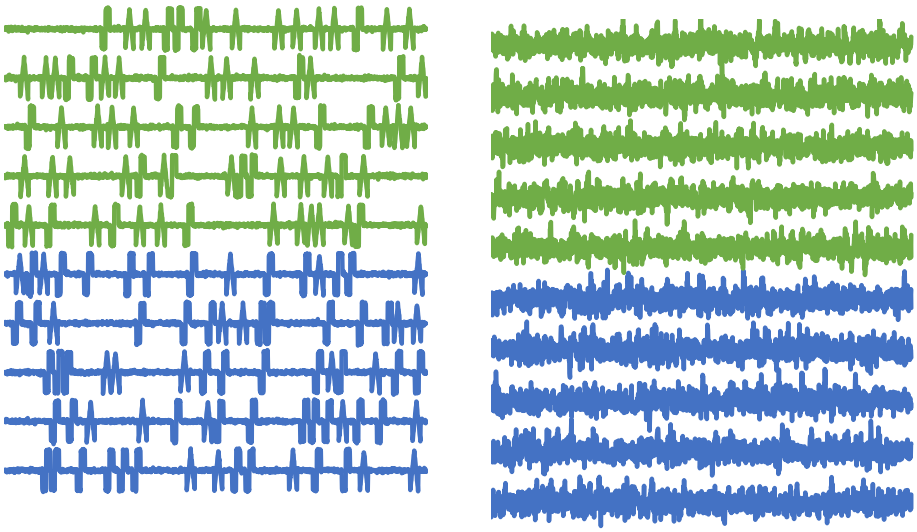


Fig. 2.6 Examples of simulated dictionary data for a two class problem. Class is defined by colour, the top five cases are of one class, the bottom five of another. Both classes contain examples of two distinct shapes, but each shape occurs more commonly in one class than the other. The first class contains more spike shapes than step shapes. The right contains far more random noise than the left, but the underlying patterns are still present.

.

Distance. Since BOSS, a number of extensions to improve different aspects have been proposed.

cBOSS [100] made BOSS contractable, and introduced optimisations and randomisation to reduce the work done in BOSS's parameter search.

WEASEL [123], Word Extraction for Time Series Classification, introduced expanded histograms through the incorporation of bigrams of words, and adopted a process of over-producing bigrams before utilising feature selection to keep the most informative ones.

SBOSS [77], Spatial-BOSS, incorporated the temporal information back into the by-default global histograms of BOSS. While dictionary classifiers, BOSS included, are concerned with the frequency of patterns in a series regrdless of location, SBOSS expands the histograms to include locational information which

is naturally leveraged by the distance functions used for final prediciton. This is discussed further in Chapter 5.

TDE [98] is the Temporal Dictionary Ensemble, which is a culmination and unification of BOSS's three improvements described previously. This is discussed in Chapter 5.

2.4.2.5 Spectral-based algorithms

The frequency domain will often contain discriminatory information that is hard to detect in the time domain. Methods include constructing an autoregressive model ([28, 5]) or combinations of autocorrelation, partial autocorrelation and autoregressive features ([7]). An interval-based spectral ensemble called Random Interval Spectral Ensemble (RISE) was proposed Lines et al. [87] and shown to be more accurate on average than whole series spectral approaches.

RISE draws on the ideas of random forests and TSF. Like TSF, RISE builds trees on random intervals from the data to construct a random forest-like classifier. The difference being that instead of summary statistics, RISE extracts spectral features over each random interval instead. RISE uses several forms of spectral features: the power spectrum, the autocorrelation function, the partial autocorrelation and the autoregressive model. New classes are classified using a simple majority vote.

2.4.2.6 Combining Representations

Two or more of the above approaches can be combined into a single classifier. For example, an approach that concatenates different feature spaces is described by Kate [65], forward selection of features for a linear classifier is the method adopted by Fulcher and Jones [48]) and transformation into a feature space that represents each group above and ensembling classifiers together formed the basis of the Collective of Transformations Ensemble (Flat-COTE) classifier [7]. Time series

data, at least those in the public archives, appear to benefit greatly from the use of particular representations. A model selection process can be used to select one. However, selection processes from the train data will not always generalise and, regardless, in some cases multiple respresentations are beneficial within a single dataset also. It is for that reason that the combinations of representations has been state of the art, and was so stand-alone until recently.

HIVE-COTE [87], the Hierarchical Vote Collective of Transformation-based Ensembles (later dubbed HIVE-COTE alpha), succeeded Flat-COTE to become the state of the art. It is a modular meta-ensemble of classifiers from each class of algorithms: EE, TSF, BOSS, ST-HESCA and RISE. Each module is encapsulated and built on the train data independently of the others. For new data, each module passes an estimate of class probabilities to the control unit, which combines them to form a single prediction. It does this by weighting the probabilities of each module by an estimate of its testing accuracy formed from the training data. The key principle behind HIVE-COTE is that TSC problems are best approached by careful consideration of the data representation, and that with no expert knowledge to the contrary, the most accurate algorithm design is to ensemble classifiers built on different representations.

HIVE-COTE 1.0 [4], was introduced to improve HIVE-COTE alpha's utility and scalability. The goal of HIVE-COTE alpha was to achieve the highest level of accuracy without concern for computational resources. Version 1.0 dropped the distance based EE due to the high computational overhead without significant loss of accuracy. STC introduced binary shapelets [15] and a randomised search controlled by a time parameter. HIVE-COTE 1.0 uses the Cross-validation Accuracy Weighted Probabilistic Ensemble (CAWPE) [78] ensemble structure (introduced and evaluated in Chapter 4 of this thesis). CAWPE uses an accuracy estimate of each classifier formed on the train data to weight the probabilities of each component. It constructs a tilted distribution through exponentiation using a parameter α to

extenuate differences in classifiers. Each component's weight is found through an internal estimate for each classifier if capable, else a ten fold cross-validation of the training data is performed.

HIVE-COTE 2.0 updates each of the representational components to the most recent state of the art, generalises to the MTSC scenario (from univariate only previously), and introduces several usability updates. It is described in Chapter 5.

TS-CHIEF [126], the Time Series Combination of Heterogeneous and Integrated Embedding Forest, is the classifier most comparable to HIVE-COTE. It too combines different representations. However, representations are embedded into the nodes of trees instead of modularly combined. TS-CHIEF is made up of an ensemble of trees which embed distance, dictionary and spectral base features. A number of splitting criteria from each representation with randomly initialised parameters are considered at each node. The different types of split criteria are dictionary based splits based on BOSS, similarity based splits based on EE and interval based splits based on RISE. Across the UCR archive, there is no significant difference in accuracy to HIVE-COTE 1.0, and so TS-CHIEF was among the state of the art for some time. HIVE-COTE 2.0, as shown in Chapter 5, improves over it.

2.4.2.7 Deep Learning

Instead of selecting a single data representation, or combining over multiple, deep learning can be utilised to learn bespoke representations per dataset. Despite their strength and popularity in handling 2D image data, a result of AlexNet's performance on the ImageNet dataset [70], deep learning approaches have only more recently been heavily studied in the (notionally easier) 1D time series domain. While the UCR archive contains numerous datasets, many of these would be considered tiny by deep learning standards, and generalisable learning on individual datasets can prove difficult. Regardless, knowledge of training methods and archi-

tectures gained from the former can be utilised on the latter, and progress in deep learning approaches has been rapid.

While Wang et al. [137] started with a smaller comparison of originally proposed architectures, Fawaz et al. [42] provided the first standardised large-scale comparative study of deep learning approaches for TSC. Nine architectures were evaluated on 85 datasets of the univariate UCR archive and 13 datasets of the Baydogan multivariate archive[†]. The Residual Network, ResNet [137] was found to be significantly better than all other approaches on the univariate datasets, and on all univariate and multivariate datasets combined. For the multivariate datasets in isolation, no significant difference was found between all approaches, mainly due the small sample size, but also due to a conservative adjustment for multiple testing. The Fully Convolutional Neural Network, FCN [137] had a slightly better overall rank, however no definitive conclusions of superiority could be drawn. We use this comparative study to take ResNet as a baseline deep learning approach. Currently the state-of-the-art deep learning approach for TSC is InceptionTime [43], which builds on ResNet.

ResNet was first applied to TSC by Wang et al. [137]. It is a network of three consecutive blocks, each comprised of three convolutional layers, which are connected by residual 'shortcut' connections that add the input of each block to its output. Residual connections allow the flow of gradient directly through the network, combating the vanishing gradient effect [58]. The residual blocks are followed by global average pooling and softmax layers to form features and subsequent predictions.

InceptionTime achieves high accuracy through a combination of building on ResNet to incorporate Inception modules [127] and ensembling over five multiple random-initial-weight instantiations of the network for greater stability [43]. A

[†]http://www.mustafabaydogan.com/

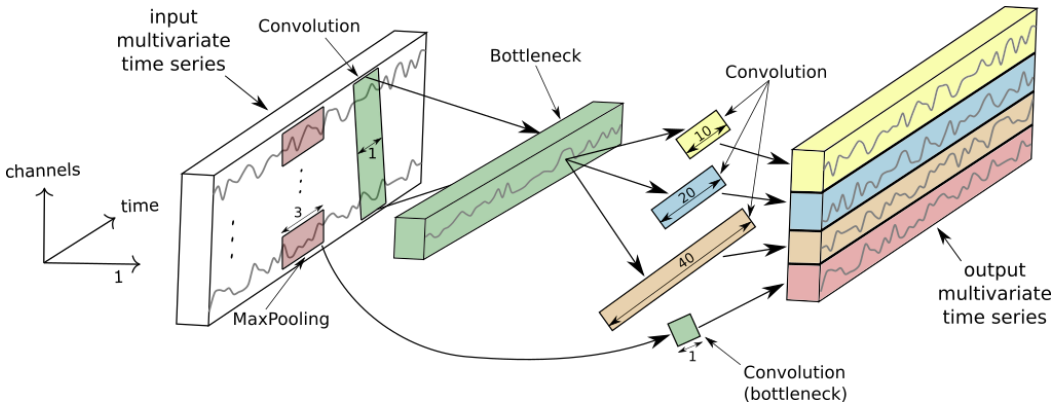


Fig. 2.7 An Inception module with example parameters, figure from [43]. Three of these are concatenated to form a block in InceptionTime.

single network out of the ensemble is composed of two blocks of three Inception modules each, as opposed to the three blocks of three traditional convolutional layers in ResNet. These blocks maintain residual connections, and are followed by global average pooling and softmax layers as before.

An Inception module is summarised in Figure 2.7. It takes an input multivariate series of length m, dimensionality d, and first uses a bottleneck layer with length and stride 1 to reduce the dimensionality to $d' < d$ while maintaining output length m. This greatly reduces the number of parameters to later learn. Convolutions of different lengths are applied to the output of the bottleneck layer to find patterns of different sizes. The outputs of these convolutions are combined with an additional source of diversity, a Max Pooling followed by bottleneck (with the same value of d') applied to the original time series, and all stacked to form the dimensions of the output multivariate time series to be fed into the next layer.

Developing approaches

ResNet and InceptionTime detailed above are two of the most experimentally verified models within the available TSC literature at the time of writing, InceptionTime being the state of the state of the art for arbitrary datasets. That being said rate of knowledge transfer from other data tasks to time series is rapid, and new models are proposed frequently. One architecture paradigm, in the same conceptual vein as

convolutional and recurrent architectures, yet to be practically utilised within TSC is the Self-attention based Transformer network [133]. These have been used to great effect in natural language processing [37, 21] and vision [111] domains within the last few years. Transformers can be framed as being able to compute *context-aware* arbitrary sequence-to-sequence functions on a compact domain [144], as opposed to fixed word embeddings as in something like word2vec [101].

The lack (so far) of Transformer networks performing well within the generalised TSC literature is likely because of a lack of millions-scale public datasets for effective pretraining within the field. Throughout this thesis, we focus on ResNet and InceptionTime as proven classifiers that we can access and implement for experimentation.

2.4.2.8 ROCKET

A newer approach which defies the categorisations above is the Random Convolutional Kernel Transform (ROCKET) [34]. ROCKET uses a large number of randomly parameterised convolution kernels applied to each instance. As each kernel is applied to a series, the max value and proportion of positive values are recorded and concatenated into a feature vector. These features are then used to build a linear ridge regression classifier with built in cross-validation to select the alpha parameter. The key motivation here is to overproduce many features that are weak by themselves, and allow the simple linear classifier to sort their relative importance, even if only relatively few of them turn out to be useful.

For each kernel generated, the parameters are selected from the following spaces: The length, l, is selected such that, $l \in \{7,9,11\}$; the value of each weight, w_i, is randomly sampled from a normal distribution $\sim \mathscr{N}(0,1)$, and are then mean centered; bias b is sampled from a uniform distribution $\sim \mathscr{U}(-1,1)$; dilation, a, is sampled from an exponential scale up to series length; the binary decision to pad

the series p is chosen with equal probability, if true the series is zero padded at the start and end equally such that middle element of the kernel is applied to every point in the input series. Stride is always set to 1. For multivariate datasets, each kernel is assigned a random number of randomly selected dimensions. The kernel for the multivariate case is still one dimensional, but with weighting being different for each dimension. The max and proportion of positive values is calculated across all selected dimensions.

2.4.2.9 Application to spectra

The classification of spectra can be phrased as a TSC problem [6]. Instead of continuous values being measured over time, they are measured over wavelength. Our hypothesis in approaching this problem for this thesis is that TSC methods that consider overall shape may be able to correct for structural defects in the spectra brought about by the many sources and differing effects of noise involved with non-invasive spectra collection. Relative to many other time series datasets that may be encountered, spectra have some simplifying advantages. First, instances are typically or can be trivially made equal-length. If the same individual type of spectroscope is used for data collection, this is automatic. Otherwise, wavelengths can simply be truncated and down/up sampled to match wavelength sampling frequencies. Second, spectra will be automatically phase-aligned. Wavelengths having a defined physical meaning means that the jth observation of one spectra refers to precisely the same physical concept as the jth observation of another. In this sense, spectra can be informally viewed as partway between tabular and time series data. Through the use of standard chemometric approaches, spectra are treated entirely as tabular data. Clearly this has been successful enough in a wide variety of domains, discussed previously. We investigate whether augmenting the modelling process to include order information is of benefit to our alcohol authentication problem.

Seven spectral datasets were included in the bakeoff evaluation [6]. These were of various foods and drinks, with direct contact to the sample, collected with Fourier transform infrared (FTIR) spectroscopy with either diffuse reflectance (DRIFT) or attenuated total reflectance (ATR) sampling. These data collection scenarios fall under the same conditions of direct contact with the sample described previously in Section 2.3.2.1. True to conventional wisdom, Bagnall et al. [6] found that on these seven datasets standard tabular classifiers (referred to as vector classifiers), albeit generally more complex than (partial-) least squares regression, were marginally stronger.

2.5 Classifier Evaluation and Comparison

In order to claim one classification method as better than another for some problem, the data structure and sampling, the exact benefit being sought, and the statistical methods used to test for difference must all be sound.

2.5.1 Data and Resampling

Throughout this thesis, we make use of datasets from three main sources. For generalised classifier comparison across multiple domains, mainly throughout Chapter 4, we use the public UCI (tabular data) and UCR (univariate time series) dataset archives. Otherwise, spectroscope data of spirit samples has been manually collected in procedures that shall be described in detail in their respective Chapters, 3 and 5.

2.5.1.1 Public archives

The University of California, Irvine (UCI) machine learning archive [‡] is widely used in the machine learning and data mining literature. An extensive evaluation of 179 classifiers on 121 datasets from the UCI archive, including different implementations of notionally the same classifier, was performed by Delgado et al. [44]. It is worth mentioning there have been several problems identified with the experimental procedure used in this study (see Wainberg et al. [134] for a critique). Firstly, some algorithms were tuned, others were used with the built in default parameters, which are often poor. For example, random forest in Weka defaults to 10 trees. Secondly, for some of the tuned algorithms, there was an overlap between validation and test datasets, which will have introduced bias. Thirdly, the data were formatted to contain only real valued attributes, with the categorical attributes in some data sets being naively converted to real values. We retain this formatting in order to maintain consistency with previous research but this may bias against certain types of classifier. Comparisons between, for example, different heterogeneous ensembles in Chapter 4 should be entirely unaffected, since they are all built on the same base classifier prediction information. We have no prior belief as to the impact of the formatting on other base classifiers and in order to avoid any suggestion of *a priori* bias, we use the exact same 121 datasets. A summary of the data is provided in Table 6.2 in the Appendix.

The University of California, Riverside (UCR) archive [33] is a continually growing collection of real valued TSC datasets[§]. Datasets come from various domains such as image outlines, audio, motion sensor readings, electrocardiograms, and spectroscopy data, as mentioned previously. A study [6] implemented 18 state-of-the-art TSC classifiers within a common framework and evaluated them on 85 datasets in the archive. The best performing algorithm, the Collective

[‡] http://archive.ics.uci.edu/ml/index.php
[§] http://www.timeseriesclassification.com

of Transformation-based Ensembles (COTE), was a heterogeneous ensemble of strong classifiers. These results were our primary motivation for further exploring heterogeneous ensembles for classification problems in general. The UCR datasets are summarised in Table 6.3, in the Appendix once more.

2.5.1.2 Random Stratified Resampling

When evaluating a classifier on a dataset, we want to understand the performance of the learning algorithm on the conceptual problem being presented. To reduce the effects of particularly favourable or harmful data distributions, it is generally wise to train and test the learning algorithm multiple times using distinct subsets of the data. Multiple approaches exist to sample a dataset, most notably cross validation, bootstrapping, and random stratified resampling.

It should be clarified that when talking about time series datasets throughout this thesis, or the usage of time series classifiers, the instances of data are either innately independent or are assumed to have been made so for the datasets in public archives. In, for example, econometrics, in cases where we consider a streamed series and predict the future values of it, it is vital that models are trained on past data to make future predictions. In our scenario, however, independent time series with their own labels are considered, and this factor can be ignored when resampling data.

For evaluation over arbitrary datasets, we take 30 random stratified resamples as the default method. When tuning hyperparameters or performing model selection on the train set of a resample, we use a nested 10-fold cross validation. All learning of parameters and hyperparameters is done on the train set of a resample, and the resulting classifier is evaluated on the corresponding test set only.

For the UCI data, 50% of the data is taken for training, 50% for testing. There is no overlap in train or test data as previously observed by Wainberg et al. [134]

and the data can be used in a similar manner to Wainer and Cawley [135] without introducing bias.

The UCR archive provides a default train/test split defined by the datasets' respective sources, and the first 'resample' is always this default split for ease comparison to other results. Later resamples maintain the class and size distributions defined in these default splits. We always compare classifiers on the same resamples, and these can be exactly reproduced with our published code.

2.5.1.3 Leave One Category Out

Where evaluating performance in the average case, we also use random stratified resampling for the evaluation of classifiers on our alcohol data. Sometimes, however, we want to evaluate performance under particular data restrictions, or assess the difficulty of particular aspects of the data. For these cases, we use a data sampling strategy where a secondary categorical attribute, that is not the class label or one that is learned from, is switched on to have data reserved for the test set of a sample, with the remainder taken for training.

In our alcohol authentication experiments, this takes the form of a leave-one-bottle-out (LOBO) cross-validation. In this scheme, all samples contained within a particular bottle or bottle type are reserved for the test set, with the remainder forming the training set. By evaluating in this manner, classifiers predicting on unseen test cases should not be able to leverage any discriminatory features caused by the bottle itself, focusing on the contents as the only commonly varying factor. This data sampling strategy is particularly useful when we want to evaluate the capability of classifiers to 'ignore' the bottle. Being able to do so suggests that suspect samples in future unseen bottles can still be reliably analysed, instead of requiring extensive training data for every bottle bottle shape, colour, etc.

2.5.2 Performance Measures

Classifiers produce a list of predictions on the unseen test data after each resample. The performance of these predictions can be summarised using a range of different statistics, dependent on the factors being selected for.

Our primary concern over many arbitrary datasets is generally error (or accuracy) because of its ease of motivation and interpretability. In particular cases where class imbalance poses a challenge, balanced error (or accuracy) can also be considered. However, in applications such as ours the costs of measurement, verification, and misclassification externally influence the ways in which decisions need to be made. For example, if the costs of confirming the legitimacy of a suspect bottle are high, relative to the resources available to the analyst, then the decision boundary may be skewed to favour the 'genuine' label. As a result, only samples that the device is more confident are fake will be seized or sent for further analysis.

Error cannot entirely capture these factors. Therefore measures that assess the quality of the classifiers' probabilistic outputs are also reported; the Negative Log-Likelihood and the Area Under the Receiver Operating Characteristic Curve.

We now formally define these metrics of interest. Recall that a data set $\boldsymbol{D}$ of size n is a set of attribute vectors with an associated observation of a class variable (the response), $\boldsymbol{D} = \{(\boldsymbol{x_1}, y_1), \ldots, (\boldsymbol{x_n}, y_n)\}$, where the class variable has c possible values, $y \in \{1, \ldots, c\}$.

A classifier M is a mapping from the space of possible attribute vectors to the space of possible probability distributions over the c valid values of the class variable, $M(\boldsymbol{x}) = \hat{\boldsymbol{p}}$, where $\hat{\boldsymbol{p}} = \{\hat{p}(y = 1|\boldsymbol{x}), \ldots, \hat{p}(y = c|\boldsymbol{x})\}$. Given $\hat{\mathbf{p}}$, the estimate of the response is simply the value with the maximum probability, i.e.

$$\hat{y} = \underset{j=1,\ldots,c}{\arg\max}\, \hat{p}(j).$$

A correctness function $f(y,\hat{y})$ returns 1 if the prediction is correct, zero otherwise,

$$f(y,\hat{y}) = \begin{cases} 1, & \text{if } y = \hat{y} \\ 0, & \text{otherwise} \end{cases}$$

The test set error is simply the proportion of incorrect predictions

$$e(D_e|M,D_r) = 1 - \frac{\sum_{y_i \in D_e} f(y_i,\hat{y}_i)}{|D_e|}. \tag{2.1}$$

On some occasions we refer to the accuracy (one minus the error) for clarity. To compensate for class imbalance, we also examine the balanced error rate. If we define the proportion correct in the test set for each class j as

$$s_j = \frac{\sum_{y_i \in D_e, y_i = j} f(y_i,\hat{y}_i)}{\sum_{y_i \in D_e} f(y_i, j)},$$

and denote r_j as the proportion of class j in the train data, then the balanced error is

$$e_b(D_e|M,D_r) = \sum_{j=1}^{c} r_j \cdot s_j. \tag{2.2}$$

The likelihood is the probability of having observed the test data given our classifier, i.e.

$$L(D_e|M,D_r) = \prod_{\boldsymbol{x_i} \in D_e} \hat{p}(y_i|\boldsymbol{x}_i,M).$$

The likelihood will be zero if the classifier predicts zero probability for the true class for any test instance. This limits the usefulness of the statistic, as it can significantly skew the results. For this reason we normalise all probability estimates when calculating the likelihood so that the minimum probability for any one class is 0.01. To make comparison with error more meaningful, we assess classifiers with

the negative log likelihood (NLL),

$$l(D_e|M,D_r) = \sum_{x_i \in D_e} \log_2(\hat{p}(y_i|\boldsymbol{x}_i,M)). \tag{2.3}$$

The fourth statistic is the area under the receiver operator characteristic curve (AUROC). AUROC is best defined where one class is considered a 'success'. Suppose we designate $y = 1$ a success and all other outcomes a failure. The classifier predictions of the probability of a success for the n instances in D_e as $\hat{p} = \{\hat{p}_1, \ldots, \hat{p}_n\}$. Observed values of the response are $\{y_1, \ldots, y_n\}$. The AUROC is based on the order statistics. We let $\hat{p}_{(i)}$ denote the i^{th} order statistic (in descending order) and $y_{(i)}$ the observed value of the response associated with probability estimate $\hat{p}_{(i)}$. These values are then used as classification functions $d(i,j)$, where 1 is a success and 0 a failure,

$$\hat{y}_{(j)} = d(i,j) = \begin{cases} 1, & \text{if } j \leq i \\ 0, & \text{otherwise} \end{cases}$$

The ROC curve is a series of n points representing the false positive rate (the proportion of failures classified as a success) on the x-axis and the true positive rate (proportion of actual successes classified as a success) on the y-axis each associated with a decision boundary. So, for example, if there are a positive cases and b negative ($a+b=n$), then, for any point i, the decision boundary is to classify as positive only those with probability greater than or equal to $\hat{p}_{(i)}$. The true positive rate is given by

$$tpr_i = \frac{\sum_{j=1}^{i} f(y_{(j)}, d(i,j))}{a},$$

and the false positive rate is

$$fpr_i = \frac{\sum_{j=1}^{i} (1 - f(y_{(j)}, d(i,j)))}{b}.$$

Given a list of n points

$$t =< (fpr_1, tpr_1), \ldots, (fpr_n, tpr_n) >$$

from the n decision boundaries, the ROC curve is a subset of this list consisting of pairs with unique point fpr values. If there are duplicate fpr values in t, the one with the maximum tpr is selected for the *ROC*. (0,0) is inserted at the beginning and(1,1) at the end. Given then a ROC curve

$$ROC =< (a_1, b_1), \ldots, (a_k, b_k) >$$

If class s is judged success, AUROC is defined as

$$AUROC_s(D_e|M, D_r) = \sum_{i=2}^{k} a_i \cdot (b_{i+1} - b_i)$$

For problems with two classes, we treat the minority class as a success. For multiclass problems, we calculate the AUROC for each class and weight it by the class frequency in the train data, as recommended by Provost and Domingos [114],

$$AUROC(D_e|M, D_r) = \sum_{i=1}^{c} w_i \cdot AUROC_i(D_e|M, D_r) \tag{2.4}$$

2.5.3 Classifier Comparison

When simply reporting a classifier's e.g. accuracy on a given dataset, we give the average accuracy over the resamples. When comparing classifiers on a dataset, because we evaluate over many resamples we can then compare two classifiers on a particular dataset with paired two sample tests, such as Wilcoxon signed-rank test. For comparing two classifiers on multiple datasets we can compare either the

number of datasets where there is a significant difference over resamples, or we can do a pairwise comparison of the average errors over all resamples.

For comparing multiple classifiers on multiple datasets, we follow the recommendation of Demšar [35] and use the Friedmann test to determine if there are any statistically significant differences in the rankings of the classifiers. However, following recent recommendations [10, 50], we have abandoned the Nemenyi post-hoc test originally used by Demsar [35] to form cliques (groups of classifiers within which there is no significant difference in ranks). Instead, we compare all classifiers with pairwise Wilcoxon signed-rank tests, and form cliques using the Holm correction (which adjusts family-wise error less conservatively than a Bonferonni adjustment).

All statistical tests of all types are performed with $\alpha = 0.05$ throughout this thesis.

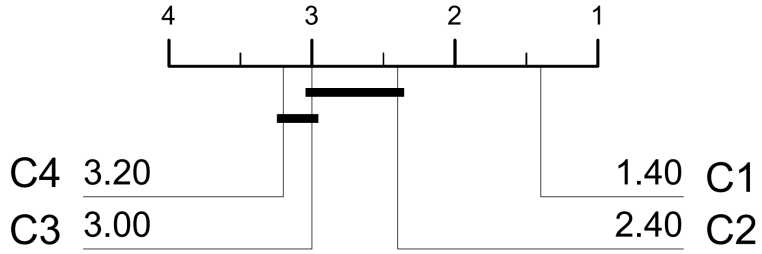


Fig. 2.8 An illustrative example of a critical difference diagram.

We can present the results of statistical tests of multiple classifiers over multiple datasets using critical difference diagrams, demonstrated in Figure 2.8. Classifiers are compared and ordered by their average performance ranks. A lower rank, and further to the right on the order line, is better. Classifiers with a thick bar connecting their lines are considered not significantly different from one another. In this example, four classifiers are compared. C1 is significantly better than the rest. C2 and C3 cannot be significantly separated. C3 and C4 cannot be separated,

but we can say that C2 is better than C4, for this performance metric over these datasets.

Forming cliques with pairwise tests is the best procedure [10], but it can be deceptive when presenting many classifiers that are very similar. A clique contains classifiers with no pairwise difference between them. However, that does not mean there is always a significant difference between all combinations of classifiers in different cliques. A pairwise test between two classifiers in isolation may draw the conclusion of no significant difference, while intermediary classifiers in terms of overall ranking *are* concluded to be significantly different. Where such a situation occurs, we clarify in the text or caption.

These methodologies for classifier evaluation and comparison provide a solid framework for the experiments developing classification algorithms and evaluating non-invasive alcohol authentication methods throughout this thesis.

Chapter 3

Classification Methods for the Prediction of Spirit Authenticity

Contributing Publications

- Large, J., Kemsley, E.K., Wellner, N., Goodall, I. and Bagnall, A., 2018, June. Detecting forged alcohol non-invasively through vibrational spectroscopy and machine learning. In *Pacific-Asia conference on knowledge discovery and data mining* (pp. 298-309). Springer, Cham.

3.1 Introduction

Chapter 2 introduced the background for this thesis, including the problem of forged spirits and spectroscopic means of analysis. Counterfeit alcohol poses potentially fatal health risks to the consumer, as illegally and poorly produced spirits may contain harmful contaminants such as methanol, a large economic risk in most markets due to the avoidance of taxes, and a risk to brand integrity in cases where the fakes are being sold as named brands.

Forgeries can sometimes be detected through external appearance such as inconsistent labelling or bottling relative to a known standard, but currently there is no way to conclusively tell whether spirits are forged without opening the bottle to gain direct contact with the sample. Breaking the seal and taking samples from a bottle can be effectively a destructive process, because even if authenticity is confirmed the bottle cannot later be sold on store shelves or at auction, and collectors' whisky will be greatly devalued. Also, testing of samples can be an expensive and time consuming process that is not suitable for mass screening. No matter what process is used it will require one or more of: transport of the sample to a centralised lab; expert knowledge and handling; consumable materials used in the analysis; and time for methods such as chromatography. It is therefore desirable to develop a system that can non-invasively determine authenticity of a suspect bottle on-site in a cheap, simple and fast manner.

Near infrared spectroscopy (NIRS) in combination with modern chemometric and machine learning techniques provides a promising potential solution to these problems. Ever improving and more affordable computing power and spectroscopy equipment as well as continual advancements in machine learning methods mean that on-site classification using cost effective equipment is becoming evermore feasible. Such setups are already used in a variety of food and drink authentication scenarios, as discussed in Section 2.2, of Chapter 2.

The alcohol concentration of genuine spirits in the UK is tightly controlled. For example, Scotch Whisky must by law contain the level stated on the bottle to within 0.3% (v/v), although the majority of commercial producers maintain stricter bounds than that for quality control reasons. Forgeries typically do not have this level of quality control, with the alcohol content often being lower than reported. Alternatively, methanol and higher alcohols have regulations prohibiting or restricting their presence in many spirits to within certain maximal concentrations to ensure safe consumption [76], and are also tightly controlled. Ethanol and methanol

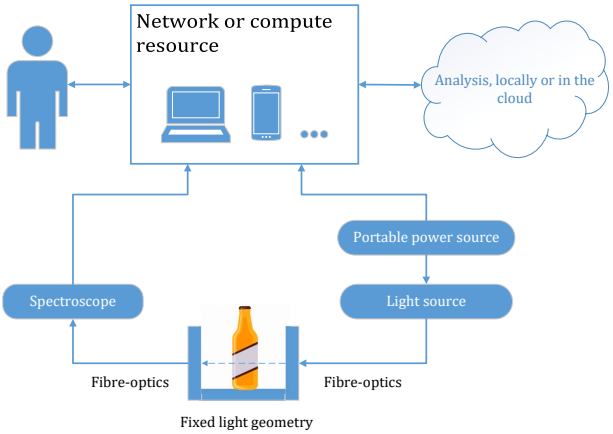


Fig. 3.1 A high level view of the proposed non-invasive forged alcohol detection system. In our experiments, a laptop is attached onto which data is saved for later analysis. In a finished system, however, utilising trained models on-site and real-time predictions are easily possible.

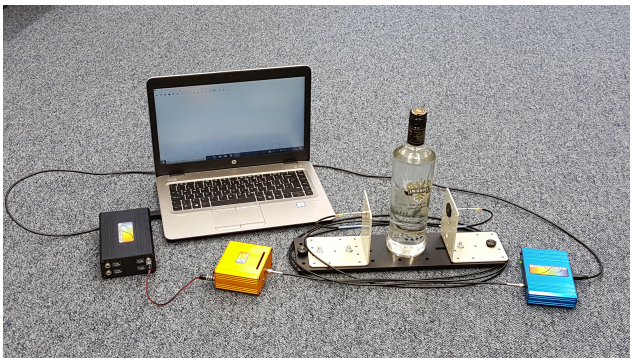


Fig. 3.2 The physical prototype equipment used through this study. A fixed light path accessed via fiber optics allows the suspect bottle to be placed consistently.

concentrations can both in principle be determined via vibrational spectroscopy and chemometric analysis methods [96, 75, 66].

While correct alcohol concentration is a necessary condition for legitimate alcohol, it is not sufficient. Carefully produced forgeries, or even the substitution of lower-priced but legitimate spirits pose a different challenge. Recently, there have been several cases of individual very high-value, often very old, spirits found to have been swapped with younger, cheaper spirits. However, this can also happen on a larger scale with low-valued spirits being fraudulently sold as medium- to high-valued.

We wish to evaluate to what extent these problems can be solved by using a non-invasive NIRS system summarised by Figure 3.1 and exemplified in Figure 3.2, where the spectra can be collected, data processed, and prediction of authenticity made within seconds. We describe two sets of experiments on collected spectra, classified using a wide variety of benchmark machine learning algorithms into 'genuine' and 'forged' categories. In the first, experiments are performed on synthesised alcohol-water solutions in real and arbitrary sealed bottles, analysed through-bottle using NIRS, and classified based on their ethanol and methanol concentrations. In the second, we evaluate whether two genuine products from the same brand, one more expensive than the other, can be distinguished within the same real bottle using the same NIRS setup.

As well as evaluating the feasibility of the problem itself, we equally test a range of different algorithms for their suitability to the data. We evaluate classical chemometrics methods, strong general purpose classifiers, TSC algorithms, and ensembles of the latter two to compare their strengths and weaknesses for the domain.

We first clarify the particular evaluation methods used for the classification experiments in Section 3.2. We separate the presentation and discussion of the alcohol detection and brand authentication experiments into their own sections for readability. Data collection, analysis, and results are presented for the alcohol concentration experiments in Section 3.3, and the brand determination experiments in Section 3.4. Overall conclusions for the general feasibility of the proposed system for forged alcohol detection are drawn in Section 3.5.

3.2 Experimental Setup

We outline here the classification algorithms and methods of evaluation using the spirit authentication datasets that have been formed. Data collection methods and analyses for each of the two sets of experiments are described in their respective sections. We perform benchmark and exploratory evaluations with a wide variety of classification schemes.

The standard classifiers evaluated are: Partial Least Squares Regression (PLSR); 1-Nearest-Neighbour with Euclidean Distance (ED); and quadratic SVM (SVMQ), while the generally stronger but more computationally expensive ensembles considered are the Heterogeneous Ensemble of Standard Classification Algorithms (HESCA); Random Forest (RandF); and eXtreme Gradient Boosting (XGBoost), all introduced in Section 2.4.1 of Chapter 2.

The TSC-specific classifiers are: Residual Network (ResNet), Random Interval Spectral Ensemble (RISE); Bag of Symbolic Fourier Approximation Symbols (BOSS); Shapelet Transform with HESCA as the classifier (ST-HESCA), Time Series Forest (TSF), and the Heirarchical-Vote Collective of Transformation Ensembles (HIVE-COTE), ensembling over the previous four classifiers, all discussed in Section 2.4.2 of Chapter 2.

The exact setup and parameterisation of each classifier (default to the literature in all cases), along with data and other supplementary material links for this work, can be found in our codebase *.

Recall from Chapter 2 Section 2.5.1, that we evaluate classifiers on many resamples/folds of the data, and average over them. The method of sampling for any particular experiment is of course dependent on the aspect of the data distribution that we are trying to reduce the variance of or correct for in an experimental setting.

*https://github.com/uea-machine-learning/tsml/tree/paper/alcohol

In the first set of experiments on determining alcohol concentration regardless of bottle (Section 3.3), we evaluate each classifier using the leave-one-bottle-out (LOBO) cross-validation scheme describe in Section 2.5.1.3. Otherwise in the second set of experiments, classifying whisky brand within a particular bottle type (Section 3.4), our resampling strategy is to simply take 30 random stratified resamples of the full data, reserving half the data for each train set, and the remaining half for the test set.

3.3 Determination of Alcohol Concentration

In this first set of experiments we wish to determine the feasibility of non-invasively classifying the alcohol concentration of a sample contained in an arbitrary bottle. Successful classification of ethanol and methanol concentrations allow for the easy detection of 'low-effort' fakes, which comprises the largest volume of fraudulent activity compared to individually high-valued spirits.

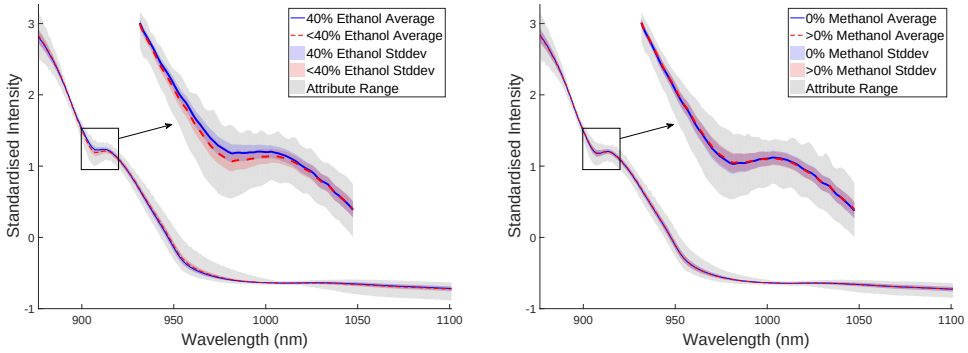
Concentrations of alcohols can be determined accurately by vibrational spectroscopy methods in standardised lab conditions [96, 75]. However, many factors could confound a fielded non-invasive alcohol classification system: variation within-product or within-batch of suspect but genuine samples; ambient light and environmental conditions; variation in spectral hardware; statistical variance in the trained classifier; and the measurement habits of different users may all cause variation in the resulting spectra. However, we believe one of the largest sources of variation which needs to be accounted for arises from the properties of the bottle a sample is contained in. Bottle shape and size, glass thickness and colour, and interfering labeling and embossing can all work to frustrate the collection of consistent, reliable spectra. Therefore, with these experiments, we primarily wish to determine the difficulty of measuring and classifying the alcohol content of samples in arbitrary bottles.

3.3.1 Data

We have conducted experiments using 44 different examples of real, non-standardised bottles. While most of the bottles are transparent and cylindrical, some are coloured, rectangular or skewed. Using a single StellarNet BLACK-Comet-SR spectrometer, transmission near-infrared spectra over a one second integration time of ethanol, methanol and water solutions within each bottle were collected to form two datasets. For the ethanol concentration experiments, 40% ethanol (with the remainder being water) is taken to be the 'genuine' case, while concentrations of 35% and 38% ethanol are taken to be 'forgeries'. The second dataset is detecting the presence of methanol. With 40% total alcohol concentration being maintained, solutions with 1%, 2% and 5% methanol (v/v) form the forged class, while 0% methanol (i.e 40% ethanol) constitutes not forged. The two classification problems are therefore to determine from a spectra whether or not a solution within an arbitrary sealed bottle 1) has less than 40% alcohol or 2) contains dangerous levels of methanol.

Three batches of each alcohol concentration were produced, and for each solution in each bottle three repeat readings are taken, resulting in a total of over 2000 readings. Bottles were positioned such that the light travels through the widest part of the bottle while avoiding labelling, embossing and seals as much as possible. However, to mimic future conditions a precise recreation of the exact path on each placement was intentionally not attempted. For simplicity, and to mimic a possible portable sampling station, the geometry of the light source and receiver was fixed at 15cm; enough to accommodate the widest bottles tested. Spectra are presented in the wavelength range 876.5nm - 1101nm, sampled every 0.5nm, and each spectrum has a dark reading subtracted and is standardised.

To help give an intuition of the classification problem, Figure 3.3 shows the average series of each class to demonstrate their differences. The progressively shaded regions show the overall standard deviation and range of intensities at each

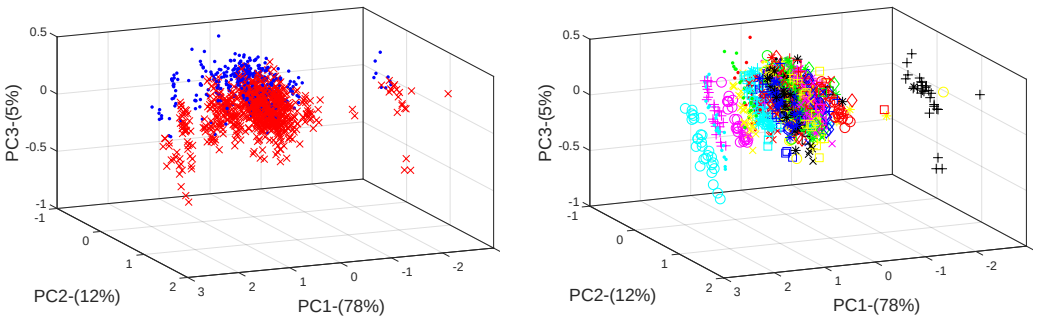


(a) Ethanol (b) Methanol

Fig. 3.3 Graphs showing the average series of each class, overall standard deviation and range for the ethanol and methanol concentration datasets. For each image, the main discriminatory region is zoomed.

wavelength. The overall variance in the dataset is very low, and the inter-class variance a fraction of that.

The zoomed regions show the wavelength ranges where alcohols are known to have a strong resonance. A clear separation between classes can be seen within the ethanol problem. However, for methanol the classes appear to be indistinguishable. Ethanol and methanol have overlapping resonances, and therefore the fact that the overall concentration of alcohol (ethanol plus methanol) has been maintained at 40% means that any difference between the class values in the resulting spectra is drastically reduced.



(a) Categorised by concentration (b) Categorised by bottle

Fig. 3.4 Graphs of the top three PCs of the PCA-transformed ethanol forgery dataset, with samples categorised by (a) ‘genuine’ (blue dot) and ‘forgery’ (red cross) based on ethanol concentrations, and (b) by bottle.

Relative to the apparent differences in the average class spectra, individual series are greatly affected by noise introduced by a variety of means through the nature of the experiment, further increasing classification difficulty. For example, an individual series may be skewed by the lensing effects of a uniquely shaped bottle.

This is evidenced by Figure 3.4. It shows the first three principle components (PCs) of the transformed ethanol dataset, which explain 95% of the total variance. In (a), the instances are categorised by their ethanol concentrations. While some separation is found between the two classes, this is observed mostly in the second and third PCs, which account for only 17% of the total variation. The first PC, as (b) shows, for the most part explains variance due to the bottles. This is in line with our expectations that bottle variation would be one of the larger obstacles to overcome for the final use case of an authentication system. While many bottles are clustered close together, there are some that form clear and separate clusters of their own. As might be expected, these are bottles that have some particularly non-standard bottle property, such as irregular shape or colour. The black + for example is a Bernheim Original bottle, which is compared to one of the more standard cylindrical bottles, Smokehead, in Figure 3.5.

a) Bernheim Original b) Smokehead

Fig. 3.5 Pictures comparing an ‘irregular’ bottle, the Bernheim Original (a), with a ‘regular’ bottle, the Smokehead (b).

Promisingly, the PCA transform does suggest a good separation between ethanol concentrations within a particular type of bottle, as best illustrated by the outlying bottle clusters when compared between figures. The equivalent figures for the methanol dataset are not included in this paper for the sake of readability and space, however, they (and the source ethanol images including keys) are available online [†]. What they show is analogous to Figure 3.3(b); that the PCA is almost entirely unable to distinguish between the alcohol concentrations. However, trends by bottle type are largely the same in that they dominate the first PC, and bottles form bands across it.

A simple statistical summary of the data used for all datasets in this Chapter and for Chapter 5 can be found in Table 6.1 in the Appendix.

3.3.2 Results: Determination of Alcohol Concentration

3.3.2.1 Leave-one-bottle-out Cross Validation

Table 3.1 summarises accuracy, area under the curve, and negative log likelihood scores of the classifiers for the LOBO experiments on the original (time series form) data. Figure 3.6 displays the ROC curves for the five classifiers with the best AUC score. Two trends are immediately apparent from these: ethanol concentration, with the correct models, can be classified with high accuracy; determining methanol concentration in a constant overall alcohol level is much more difficult. Only some of the classifiers tested achieve much higher than the minimum expected accuracy of 0.75, the proportion of the majority class.

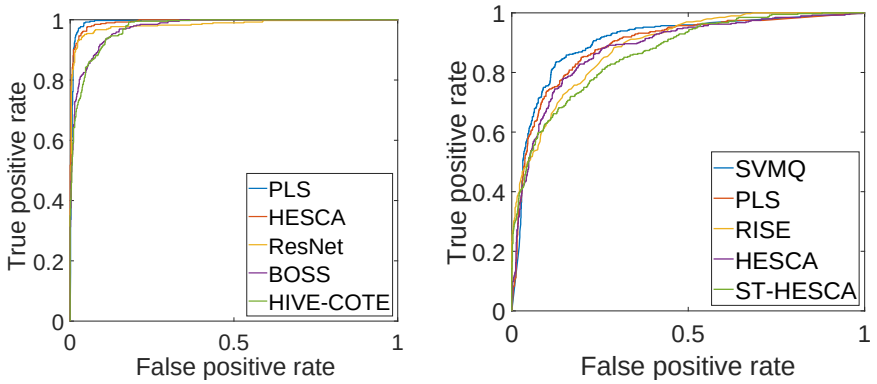
Across both problems the TSC-specific approaches appear to add little value over the more standard, vector-based approaches and ensembles. The de-facto standard PLS is consistently at or close to the top performances across each evalu-

[†]https://github.com/uea-machine-learning/tsml/tree/paper/alcohol

Table 3.1 Average accuracies over all folds of the leave-one-bottle-out-sampled alcohol concentration datasets. The best scores in each column are bold. Classifiers grouped by being considered as standard, ensemble, or TSC-bespoke classifiers.

	Ethanol			Methanol		
Classifier	ACC ↑	AUC ↑	NLL ↓	ACC ↑	AUC ↑	NLL ↓
ED	0.866	0.851	0.891	0.672	0.564	2.177
PLS	**0.965**	0.994	0.271	0.860	0.913	0.543
SVMQ	0.959	0.981	0.300	**0.864**	**0.920**	0.827
HESCA	**0.965**	**0.995**	**0.170**	0.843	0.898	**0.522**
RandF	0.888	0.972	0.399	0.758	0.727	0.737
XGBoost	0.923	0.980	0.315	0.794	0.815	0.776
ResNet	0.958	0.991	0.322	0.815	0.859	2.055
BOSS	0.913	0.981	0.299	0.786	0.820	0.642
RISE	0.817	0.962	0.647	0.793	0.908	0.636
ST	0.919	0.981	0.271	0.836	0.878	0.543
TSF	0.878	0.974	0.409	0.769	0.817	0.695
HIVE-COTE	0.915	0.986	0.346	0.802	0.870	0.618

ation metric. The heterogeneous ensemble HESCA, which ensembles over eight relatively standard classifiers, and SVMQ perform the best on the ethanol and methanol problem formulations respectively. HESCA is a generally strong ensemble, however the performance of SVMQ is somewhat surprising considering its parameters were not tuned.



(a) Ethanol (b) Methanol

Fig. 3.6 ROC curves for the classifiers on the LOBO-sampled alcohol concentration problems. Predictions are concatenated over all folds. For clarity, only the five best classifiers in terms of AUC are displayed.

Because each test fold represents a single bottle, the accuracy on a fold gives an indication of the difficulty that a particular bottle adds to the classification problem.

We took the top classifiers on the ethanol problem, PLS and HESCA, and looked at which bottles were preventing perfect classification. Of 44 bottles, 19 had the alcohol concentration of their contents classified perfectly by both PLS and HESCA. Of the rest, most of the errors are split to one or two per bottle, with a couple of exceptions among the irregular bottles. The worst average fold accuracy represents the Bernheim Original Kentucky Straight wheat whiskey bottle, where HESCA made 9 errors, and PLS 3.

Similarly, if we split the computation of AUC scores between the bottle characteristics for the methanol problem, there is a clear drop in performance from the standard bottles (best AUC scores of around 0.93) to the irregular bottles (best AUC scores around 0.85). This does lend credence to the idea that the determination of alcohol concentration cannot be done *entirely* irrespective of bottle. However, the fact that there is clearly some transferability (evidenced by better-than-guessing performance in this LOBO format) is promising.

In previous through-bottle studies on alcohol concentrations [66, 106], coloured glass posed challenges for the collection of Raman spectra, which particularly struggles to handle fluorescence, but also for NIRS in the latter. Our experiments included three green-glass bottles, however on these no significant drop in predictive accuracy was observed in the same analysis of the top four classifiers. These three bottles also showed no clear separation from the largest central cluster in the PCA transform presented in Figure 3.4b.

3.3.2.2 Classifying the bottle

The PCA transform of the ethanol dataset, Figure 3.4b, indicated that the majority of the variance corresponded with differences in the containing bottle's properties. Further, most of the first PC was caused by a small number of irregularly shaped bottles. The majority of bottles otherwise formed a dense cluster. To further

Table 3.2 Results of classifying the containing bottle regardless of contents, 44 class problem. SVMQ exhibits a surprisingly wide gain in performance over the other algorithms. The best scores in each column are bold.

Classifier	ACC ↑	AUC ↑	NLL ↓
ED	0.400	0.717	3.982
PLS	0.056	0.521	6.101
SVMQ	**0.551**	**0.947**	2.660
HESCA	0.512	0.938	2.713
RandF	0.431	0.917	2.875
XGBoost	0.403	0.898	3.351
ResNet	0.420	0.924	6.087
BOSS	0.463	0.884	2.674
RISE	0.503	0.930	3.049
ST	0.499	0.932	**2.464**
TSF	0.468	0.922	2.704
HIVE-COTE	0.509	0.940	2.550

investigate the extent to which features of the bottle are detectable in the spectra, we ran experiments with the same set of classifiers but with the containing bottle as the class label, instead of alcohol concentration. We would expect the outlying bottles on the PCA transform to be the easiest to classify, with the standard bottles being guessed at.

The dataset was split 30 times using random stratified sampling with a 70/30 train/test split. Table 3.2 summarises these results. The best accuracies achieved were up to 0.551 (SVMQ), on the 44 class problem.

In the interest of finding where the classifiers were making their errors, we grouped bottles by whether they could be described as being standard (clear glass and cylindrical, 28 bottles) or irregular (coloured glass and/or non-cylindrical, 16 bottles). Considering the SVMQ's predictions, Figure 3.7 depicts a confusion matrix with the bottles (classes) grouped by whether they are standard or irregular. Classifying a standard bottle as a different standard bottle accounts for 67% of the total number of errors, while the remaining three account for 12, 12, and 9% each.

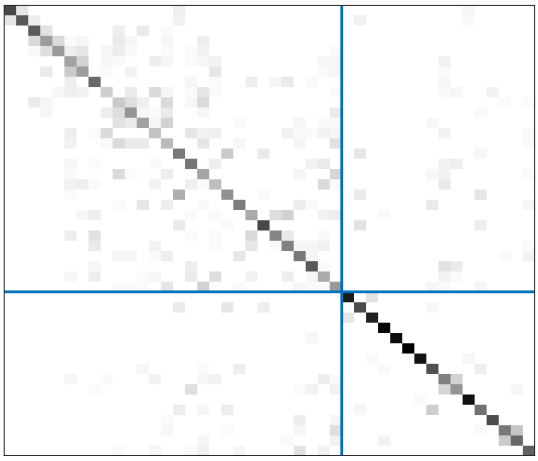


Fig. 3.7 A confusion-matrix of SVMQ's predictions the bottle classification problem, aggregated over folds. It is much more likely to mistake a standard bottle for another standard bottle than anything else.

These results have positive implications for the original goal of generic non-invasive alcohol level determination. It suggests that a classifier could be reliably trained under the assumption that the test sample bottle has certain properties matching those in the train set. In terms of the practical use and production costs of a device, the worst case is that each individual type of bottle requires its own adequately populated training data for a model to learn on. While this may still be needed for each of the irregular bottles, a device that can effectively classify the contents of many different bottles within some particular range of properties is still a worthwhile improvement over the worst case.

3.3.2.3 PCA Transforms

Lastly for alcohol concentration, we repeated the LOBO classification experiments again with PCA-transformed versions of the datasets (calculated and applied to each resample individually), maintaining components that explain 95% of the variance. Analysis of spectral data in the literature often involves a dimensionality-reducing transformation such as PCA, both to highlight discriminatory variance and reduce

Table 3.3 Performances of non-TSC algorithms averaged across folds on the PCA transformed alcohol concentration problems. Performances are greatly diminished in relation to classification using the full spectra. The best scores in each column are bold.

	Ethanol			Methanol		
	ACC ↑	AUC ↑	NLL ↓	ACC ↑	AUC ↑	NLL ↓
HESCA	**0.818**	**0.935**	**0.544**	**0.750**	0.616	**0.800**
ED	0.779	0.747	1.468	0.627	0.506	2.475
PLS	0.801	0.927	0.593	0.745	0.622	**0.800**
RandF	0.817	0.925	0.572	0.714	0.530	0.885
SVMQ	0.803	0.927	0.581	**0.750**	0.537	0.810
XGBoost	0.809	0.911	0.781	0.703	0.539	1.016

the computation time of analysis. However, in this case it appears to reduce accuracy relative to classification performed on the time series, in agreement with Kiefer et al. [66].

The methanol PCA transform seemingly cannot discriminate between concentrations at all, with all classifiers simply picking the majority class. For ethanol, all classifiers except ED achieve very similar accuracies. Referring to Figure 3.4a, it would seem that most of the classifiers are forming almost identical decision boundaries, the same that a human naively would by eye.

3.4 Authentication of Reported Brand

Our first set of experiments considered cases where alcohol concentrations may not align with expected values. This represented the verification of a necessary, but not sufficient, test for authenticity and safety.

We now present experiments that look into a harder problem within this domain - distinguishing between different genuine spirits. These represent attempts to confirm or deny that the spirit contained within a suspect bottle align with that reported on the label. Cheaper, but still genuine in their own right, spirits can be sold under the guise of rarer and more expensive examples for profit. Depending

on their familiarity with the expected product, the average consumer would likely not recognise any difference. Alcohol concentration, which accounts for a high proportion of the volume of the samples in question, cannot be relied on in this case. The discriminatory information must reside in the less abundant compounds of the sample.

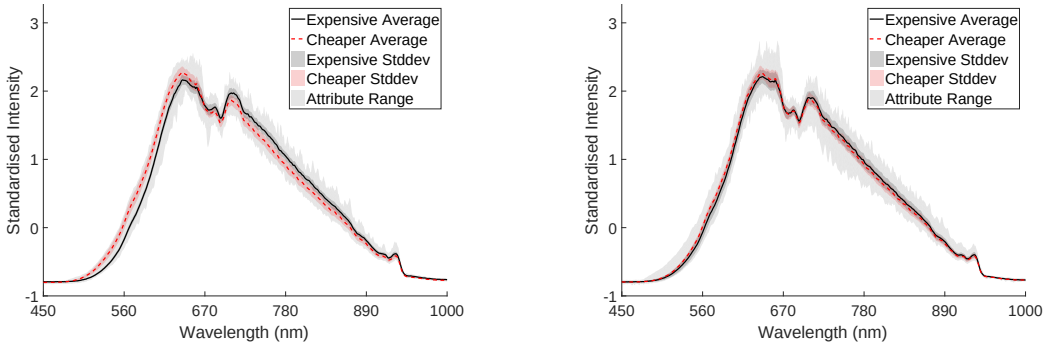
3.4.1 Data

We experiment with two different spirits from the same producer, one more expensive than the other. We collect data using the same high-level setup as in the previous set of experiments of eight cheaper, and eight more expensive real whiskies. We swap the contents of four pairs of bottles, such that we have four examples each of cheaper and more expensive whiskies in their original bottles, and cheaper and more expensive whiskies in incorrect bottles. We use these to form two sets of classification problems: in each of the bottle designs, do the contents align with what's reported on the label? An ideal system would of course discriminate between more than just two particular whiskies. For obvious practical reasons, however, we limit the scope of our experiments to these samples as a feasibility test.

In total, four batches of data involving whiskies have been collected. We define a batch in this context as ten readings of each bottle for 160 total readings, taken in a randomised order in a single continuous session. The four batches consist of two taken by an expert user, and one each taken by informed but non-expert users. In total our full dataset therefore consists of 640 spectra. The data collection process mirrors the previous experiments, however a BLUE-Wave spectrometer was used instead of the BLACK-Comet-SR. Data was also collected by multiple people, one expert and two non-expert users. The potential effects of different users shall be investigated at the end of Section 3.4.2.3, but otherwise all users' data are combined to simulate datasets composed of readings from many sources. Spectra

are presented in the wavelength range 450nm - 1000nm, sampled every 0.5nm, and each spectrum is standardised.

A reason for separating these experiments into two problems based on the containing bottles is that there is and interesting variation in the bottle design based on the time period of acquisition. While the expensive bottles are all identical, there is minor variation in the cheaper bottles due to the time of purchase and having been sold via the European Union, as opposed to solely within the UK. This has had an impact on the quality of the spectra and ultimately differentiability of the contents between the expert and non-expert users, where the former was correcting for this fact but latter were unaware of it. The differences are much finer than the structural differences that were the focus of the bottles tested in the alcohol concentration experiments previously. However, as shall be seen, the differences were enough to significantly affect performance of the models on this more difficult classification problem.

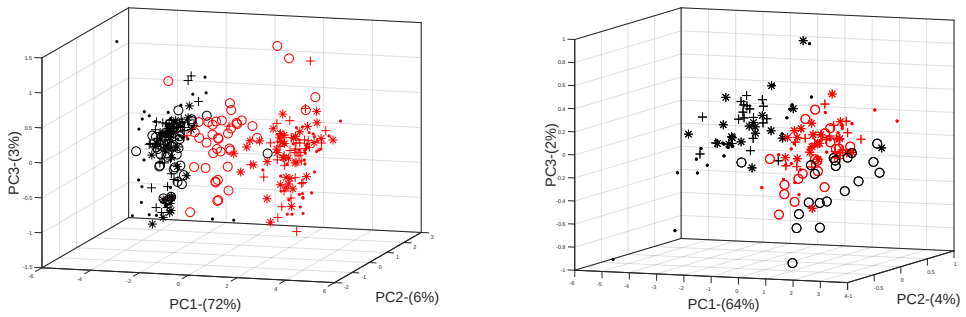


(a) Contained in the more expensive bottles (b) Contained in the less expensive bottles

Fig. 3.8 Graphs displaying the averages (lines), standard deviations (dark shaded regions), and overall range (light shaded region) of the more expensive and cheaper whiskies while in the more expensive (a) and cheaper (b) bottles.

Figure 3.8(a) displays summarised spectra of the two whiskies inside the more expensive and consistent bottles. A surprisingly clear degree of separation is observed, especially around the wavelength interval 450 to 700nm. Meanwhile for the whiskies presented in cheaper bottles in (b), which have the differences noted

above, far less distinction between the classes can be found, and a far larger range of intensities are observed.



(a) Contained in the more expensive bottles

(b) Contained in the less expensive bottles

Fig. 3.9 Graphs of the top three PCs of the PCA-transformed brand authentication dataset, with samples categorised by cheaper (red symbols) and more expensive whisky (black symbols), whilst in the more expensive (a) and cheaper (b) bottles.

Figure 3.9 displays the first three PCs of the PCA-transformed datasets of Figure 3.8. Different individual bottles are denoted by different symbol types ($\circ, +, *, \bullet$). In Figure 3.4(b), it was seen that a very large proportion of the variance was explained by the containing bottle. This was to be expected of that experiment. Here, however, the bottles are supposed to be effectively identical within certain parameters of precision of the bottle and whisky production process. It can be seen however that the '$\circ$' bottles of the forgery in each figure are somewhat outliers, especially in the case of the cheaper bottles on the right. That the outlier bottle in both images is represented by $\circ$ in both cases is coincidental; there is no connection between the physical bottles they refer to.

3.4.2 Results: Authentication of Reported Brand

3.4.2.1 Stratified Random Resample

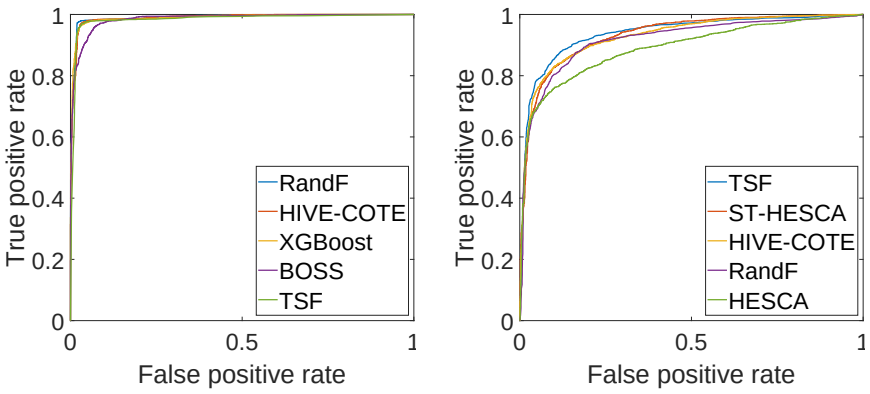
We first present results for the general problem statement; pooling all the data together and performing a 30 stratified random resample evaluation. Table 3.4

Table 3.4 Performances of each classifier averaged across folds on the brand authentication problem. The best scores in each column are bold.

	Expensive Bottle			Cheaper Bottle		
Classifier	ACC ↑	AUC ↑	NLL ↓	ACC ↑	AUC ↑	NLL ↓
ED	0.965	0.93	0.235	0.753	0.562	1.64
PLS	0.879	0.931	0.470	0.603	0.590	1.690
SVMQ	0.967	0.961	0.246	0.734	0.649	2.045
HESCA	**0.976**	0.983	0.160	0.825	0.891	0.616
RandF	**0.976**	**0.993**	**0.145**	0.849	0.922	0.525
XGBoost	0.965	0.991	0.178	0.827	0.889	0.735
ResNet	0.923	0.974	0.573	0.672	0.738	2.158
BOSS	0.942	0.987	0.233	0.733	0.812	0.755
RISE	0.941	0.979	0.391	0.761	0.817	0.807
ST	0.958	0.980	0.200	0.861	0.935	0.484
TSF	0.966	0.984	0.181	**0.879**	**0.942**	**0.453**
HIVE-COTE	0.968	0.991	0.189	0.864	0.931	0.535

displays the scores of each classifier across the three performance metrics, while Figure 3.10 displays the ROC curves.

In the alcohol concentration problem, the simpler classifiers out-performed the more complex and TSC-based ones. The classification problem was fundamentally linear in nature - the alcohol concentration. The ethanol problems were easier than methanol due to the larger absolute differences in concentration and lack of overlapping contributions to the spectra from the two different alcohols. This brand authentication problem must classify on non-linear factors, and as a result we find that the more complex classifiers and TSC-based classifiers make a relative gain on this problem. Standard ensembles still win on the easier formulation of determining contents in the more constistent expensive bottles, however TSF in particular (and HIVE-COTE, an ensemble of which TSF is a member) become relatively stronger on the less consistent cheaper bottle formulation.



(a) More expensive bottles (b) Cheaper bottles

Fig. 3.10 ROC curves for the classifiers on the brand authentication problems. Predictions are concatenated over all folds. For clarity, only the five best classifiers in terms of AUC are displayed.

3.4.2.2 PCA Transforms

Even when classifying on the full series, predictive performance has been shown for a number of classifiers to be strong in the optimal case of identical bottle designs. Dimensionality reduction in the form of techniques such as a simple attribute or interval selection and transforms such as PCA may not have much room to improve on this. If predictive performance could be maintained after their application, however, they may provide a means of reducing the model training and more importantly prediction time, as well as perhaps lead to better ways to inform the user of the reasons for any prediction a system makes.

Figure 3.9 displayed strong separation between the classes of the brand authentication problem within the top three PCs, but also showed that these PCs only accounted for 81% and 70% of the variance of each dataset respectively. Closer inspection revealed that beyond the fourth or fifth PC, the remaining components each explain similarly diminishingly small amounts. We experimented with PCA transforms maintaining only the top three components, i.e. the exact information displayed in Figure 3.9. While not presented for brevity, maintaining further components did not help performance.

Table 3.5 Performances of considered non-TSC classifiers on PCA transforms of the brand authentication problem in the two sets of bottles. Due to the restricted number of attributes (first three principle components) and therefore frequent mathematical problems in computing matrices being encountered, PLS is not included. The best scores in each column are bold.

	Expensive Bottle			Cheaper Bottle		
Classifier	ACC ↑	AUC ↑	NLL ↓	ACC ↑	AUC ↑	NLL ↓
HESCA	0.969	**0.990**	**0.166**	0.822	0.889	0.610
ED	**0.971**	0.943	0.193	**0.838**	0.702	1.073
RandF	0.960	0.979	0.187	0.830	**0.904**	**0.591**
SVMQ	0.967	0.986	0.217	0.767	0.740	0.872
XGBoost	0.968	0.986	0.201	0.823	0.891	0.753

Table 3.5 shows that performance drops only slightly from usage of the full spectra, for both bottle sets. The classifiers considered in this study vary wildly in terms of their training and testing time and complexities, and different models types appear to work to differing degrees of success depending on the particular problem formulation. The usage of a PCA transform as both a visualisation and classification tool to allow simpler and faster predictions in this problem aids the end goal greatly.

3.4.2.3 Testing on different user's data

Taking a reading with the equipment used for these experiments involves placing a bottle upright between the spectroscope and light source and saving the spectra on an attached laptop. While we maintain efforts to avoid labelling, embossing, etc., we do not impose any precise restrictions on recreating the exact same path geometries relative to the bottle positioning, as this would not be possible to recreate in a field-scenario. Given that for these experiments the samples are contained within supposedly identical bottles, we wish to investigate another potential form of confounding information: whether any performance-impacting differences arise in the spectra collected by different users. Instructions and brief demonstrations were provided to two non-expert users to collect batches of readings, as described

Table 3.6 Performances of each classifier on the brand authentication problem sampled such that data collected by one user are used for the train data, and those collected by others are reserved for testing. Scores are expressed as the difference to the average scores across 30 stratified random resamples as reported in Table 3.4. ACC and AUC aim to be maximised, while NLL aims to be minimised. Performance degrades everywhere, except from the accuracy of RISE (likely by chance).

	Expensive Bottle			Cheaper Bottle		
Classifier	ACC ↑	AUC ↑	NLL ↓	ACC ↑	AUC ↑	NLL ↓
ED	-0.009	-0.017	0.057	-0.105	-0.180	0.699
PLS	-0.112	-0.050	0.223	-0.131	-0.223	0.936
SVMQ	-0.036	-0.048	0.136	-0.105	-0.126	0.856
HESCA	-0.039	-0.015	0.099	-0.127	-0.038	0.282
RandF	-0.008	-0.017	0.110	-0.114	-0.062	0.233
XGBoost	-0.003	-0.008	0.067	-0.123	-0.112	0.797
ResNet	-0.068	-0.063	0.960	-0.068	-0.063	3.639
BOSS	-0.118	-0.012	0.230	-0.129	-0.110	0.186
RISE	0.015	-0.011	0.099	-0.126	-0.062	0.090
ST	-0.153	-0.025	0.391	-0.201	-0.144	0.468
TSF	-0.173	-0.052	0.469	-0.105	-0.058	0.228
HIVE-COTE	-0.012	-0.009	0.177	-0.153	-0.090	0.222

previously. However, they were then left to collect the readings used in the final dataset with minimal supervision. If we train using only the data of the expert user, and test on the data of the others, is there any degradation in prediction performance?

Table 3.6 shows that, relative to the performance scores on the 30 stratified random resamples of the same data, there is a minimal but consistent drop in performance when separating data based on the collector. The degree of performance loss varies by classifier, and also by the containing bottle type. The more expensive bottle shows anywhere from negligible to severe losses depending on the classifier, while the cheaper bottle with minor structural differences shows a consistent and large drop of 10 to 20% accuracy, with the exception of ResNet which was already scoring poorly (Table 3.4).

We hypothesise that the expert collector, whose data formed the train set, was correcting for slight differences in the label size and positioning, resulting in

different refractive properties for the light path through the bottle. This is perhaps particularly important in the case of the particular bottles tested because they are square-faced rather than cylindrical bottles, and as such the light received would be more sensitive to differences in bottle placement. As a result, the training of future users may be a required step for collecting useful data to fit models or predict on, as opposed to being a completely accessible device off-the-shelf.

In any case, that such a drastic difference in predictive performance and visualisation arises from such small differences suggests a hurdle to be overcome if a generalised database and classification system for many brands in many bottle types is to be engineered. We believe these problems to be largely solvable via scaling the data with more samples, analysts, and in a similar vein, more spectroscopes.

3.5 Conclusions

We have demonstrated the feasibility of a system to accurately detect forged alcohol in two important ways, using near infrared spectroscopy and machine learning to generate non-invasive, non-destructive predictions of authenticity within seconds.

When determining the alcohol concentration of sealed bottles of arbitrary spirits, ethanol level can be classified with high accuracy. Ethanol concentration aligning with that reported is a necessary condition of authenticity, which is often not met by low-effort fakes. Dangerous levels of methanol within a consistent total alcohol concentration were more difficult to accurately detect. However, results significantly better than random guessing were demonstrated, suggesting that the discriminatory features are not entirely lost at the physical hardware level. There is likely room for improvement with different optical geometries, more tailored data processing and model selection.

Bottles with particularly unusual properties introduce extra difficulty, but the minor differences between more standard, cylindrical bottles do not confound the classifiers. This suggests that a combined dataset can be made to train models for easy bottles, with perhaps a two-stage classification pipeline required only for particularly unique bottle designs.

When concerned with the authenticity of a particular brand as a opposed to a general spirit, factors other than alcohol may need to be used. We have found that a particular brand of spirit contained within a bottle could also be classified with high accuracy. For this problem though, a practical system clearly needs to discriminate between more than just two products, and performance fell with variation in the suspect bottle structure. A more extensive database, including different brands, bottle types and production batches of the same brand, would shed clearer light on the practical generalisation of the method to many different possible spirits and their associated bottles. We have, however, shown the feasibility of detecting the presence of a particular brand as marketed on the bottle.

The traditional method used in chemometrics of Partial Least Squares regression performed well when detecting alcohol concentration. However, it struggled on the distinctly non-linear brand determination problem. A quadratic support vector machine performed well, and a larger computational investment for thorough tuning would likely lead to improved results for this. Ensembles scored highly on all problems throughout. The deep learning approach ResNet achieved strong but never the best evaluation scores and would probably benefit relatively more from larger datasets. Algorithms bespoke to TSC saw gains when classifying the more complex brand authentication formulations, however it is possible they were overcoming difficulties in the data presentation rather than the underlying problem of interest, and that human effort spent there may allow for the cheaper standard methods to work just as well or better.

Along with generally increasing the amount of useful data available to the learners, a combination of tuning and ensembling standard classification methods seems to be the most promising route to follow for this problem. Tighter wavelength interval selections in preprocessing, be that done manually or automatically, are also worth investigation. This is especially true because the two problem formulations evaluated here displayed discriminatory information at different places in the spectra.

Chapter 4

CAWPE: An Ensemble Method for General Purpose Classification

Contributing Publications

- Large, J., Lines, J. and Bagnall, A., 2019. A probabilistic classifier ensemble weighting scheme based on cross-validated accuracy estimates. *Data mining and knowledge discovery*, 33(6), pp.1674-1709.
- Large, J. and Bagnall, A., 2019, November. Mixing hetero-and homogeneous models in weighted ensembles. In *International Conference on Intelligent Data Engineering and Automated Learning* (pp. 129-136). Springer, Cham.
- Bagnall, A., Flynn, M., Large, J., Lines, J. and Middlehurst, M., 2020. A tale of two toolkits, report the third: on the usage and performance of HIVE-COTE v1.0. *arXiv e-prints*, pp.arXiv-2004.

4.1 Introduction

Chapter 3 introduced the alcohol authentication problem space and initial benchmarking attempts in the context of synthesised alcohol solutions. One of the findings was that ensemble methods, in particular HESCA, were among the best performing. In the initial problem statement, the reliability of probability distributions was a factor identified as a key requirement to aid in actual decision making by end users in the field. In this Chapter, we develop and evaluate an augmented weighted vote ensemble scheme with the express aims of improving probability estimates above and beyond general predictive performance. We evaluate in this Chapter on many datasets from multiple domains for the benefit of general purpose classification and application to many different use case, and the resulting ensemble is evaluated more specifically on this thesis' particular application of alcohol authentication in Chapter 5.

Investigation into the properties and characteristics of classification algorithms forms a significant component of all research in machine learning. Broadly speaking, there are three families of algorithms that could claim to be state of the art for classification: support vector machines; multilayer perceptrons/deep learning; and tree based ensembles. Nevertheless, there are still good reasons, such as scalability and interpretability, to use simpler classifiers such as decision trees. Thousands of publications have considered variants of these algorithms on a huge range of problems and scenarios. Sophisticated theories into performance under idealised conditions have been developed and tailored models for specific domains have achieved impressive results. However, data mining is an intrinsically practical exercise and our interest is in answering the following question: if we have a new classification problem or set of problems, what family of models should we use given our computational constraints? Large-scale comparative studies of classifiers

attempt to give some indication (for example Fernández-Delgado et al. [44]), but most people make the decision for pragmatic or dogmatic reasons.

Our first hypothesis is that, in the absence of specific domain knowledge, it is in fact better to ensemble classifiers from different families rather than intensify computational efforts into selecting and optimising a specific type. Our second hypothesis is that the best way of combining a small number of effective classifiers is to combine their probability outputs, weighted by an accuracy estimate derived through cross-validation on the training data, raised to the power four to magnify differences in competence. We call this weighting scheme the Cross-validation Accuracy Weighted Probabilistic Ensemble (CAWPE). The algorithm has the benefit of being very simple and easy to implement, trivially parallelisable, incremental (in that new classifiers can be added to the ensemble in constant time) and, on average, provides state-of-the-art performance. We support the last claim with a series of experiments on two data archives containing over 200 datasets using over twenty different classification algorithms. We compare classifiers on unseen data based on the quality of the decision rule (using classification error and balanced classification error to account for class imbalance), the ability to rank cases (with the area under the receiver operator characteristic curve) and the probability estimates (using negative log likelihood).

The algorithms we compare against can be grouped into three classes: heterogeneous ensembles; homogeneous ensembles; and tuned classifiers. The first of these classes is in direct competition with our approach, while the latter two are examples of attempts to improve individual types of classifiers.

The heterogeneous ensemble algorithms most similar to our approach involve alternative weighting schemes, ensemble selection algorithms and stacking techniques, introduced in Section 2.4.1.1 of Chapter 2. We compare CAWPE to nine variants of these heterogeneous ensembles that all use the same base classifiers

and the same estimate of accuracy found through train set cross-validation. We demonstrate that CAWPE provides a small, but significant, improvement on all of them.

To put the performance of CAWPE in a wider context we also compare it to homogeneous ensembles and tuned single classifiers. We choose classifiers to compare against from among those often considered to be state of the art: random forest; support vector machines; neural networks; and boosting forests. Using data derived from the UCI archive, we find that a small ensemble of five untuned simple classifiers (logistic regression, C4.5, linear support vector machine, nearest neighbour classifier and a single hidden layer perceptron) combined using CAWPE is not significantly worse than either state-of-the-art untuned homogeneous ensembles, nor tuned random forest, support vector machine, multilayer perceptron and gradient boosting classifiers.

To avoid and correct for any danger of dataset bias, we repeat the core experiments on a completely separate repository, the UCR archive of TSC problemss, and draw the same conclusions. We show that the CAWPE scheme can provide a small, but significant, improvement to the current state-of-the-art TSC algorithm.

We then address the question as to why CAWPE does so well. We compare CAWPE to choosing the best classifier and find that the CAWPE approach is significantly better. It is most effective for data with small train set size. CAWPE consists of four key design components: using heterogeneous classifiers; combining probability estimates instead of predictions; weighting these probabilities by an estimate of the quality of the classifier found on the train data; and increasing the differences of these weights by raising them to the power α, the single parameter of the classifier. On their own, none of these components are novel. Our contribution is to demonstrate that when used together, the whole is greater than the sum of the parts. To demonstrate this we perform an ablation study for the last three design

components of CAWPE and show that each element contributes to the improved performance. We perform a sensitivity analysis for the parameter α and show that CAWPE is robust to changes to this parameter, but that the default value of $\alpha = 4$ we decided on *a priori* and use in all experiments may be improved with tuning. The exponentiation through the parameter α allows for the amplification of small differences in accuracy estimates. This facilitates base classifiers that show a clear affinity to a given problem to provide a larger contribution to the ensemble while still allowing it to be overruled when enough of the other base classifiers disagree. It provides a mechanism to balance exploiting information found from the train data (through high α) and mitigating for potential variance in the accuracy estimate (through lower α).

In summary, the remainder of this chapter is structured as follows. Section 4.2 describes the CAWPE classifier and motivates the design decisions made in its definition. Section 4.3 contains our assessment of the CAWPE classifier. We compare CAWPE to its components (4.3.1), other heterogeneous ensemble schemes (4.3.2), homogeneous ensemble schemes (4.3.3), and tuned state-of-the-art classifiers (4.3.4) on 121 UCI datasets. We also present a reproduction study of the performance gain between CAWPE and its base classifiers on the UCR TSC datasets (4.3.5), and compares its performance to the standard benchmark classifier in that domain. Section 4.4 provides a deeper analysis into the CAWPE scheme. We explore the differences in performance between combining a set of classifiers with CAWPE and picking the best of them based on the train set of any given dataset (4.4.1). To better understand the nature of the improvements, we also carry out an ablation study that builds up from simple majority voting to CAWPE (4.4.2), and perform a sensitivity analysis of CAWPE's parameter, α (4.4.3). We also investigate mechanisms to maintain models created through the cross validation process, to create hybrid homogeneous and heterogeneous ensembles and assess their relative performance and stability (4.4.4). Finally, we conclude in Section 4.5.

Our conclusion is that it is, on average, better to ensemble the probability estimates of strong classifiers with a weighting scheme based on cross-validated estimates of accuracy than expend resources on a large amount of tuning of a single classifier and that the CAWPE scheme means that classifiers can be incrementally added to the ensemble with very little extra computational cost.

4.2 The Cross-validation accuracy weighted probabilistic ensemble (CAWPE)

The key features that define the weighting scheme we propose in the context of other commonly used weighting schemes such as those described in Chapter 2 are that, firstly, we weight with accuracy estimated through cross-validation instead of a single hold-out validation set, secondly, we extenuate differences in accuracy estimates by raising each estimate to the power of α and thirdly, we weight the probability outputs of the base classifiers instead of the predictions. To clarify, prediction weighting takes just the prediction from each member classifier,

$$\hat{p}(y=i|\boldsymbol{E},\boldsymbol{x}) \propto \sum_{j=1}^{k} w_j d(i,\hat{y_j})$$

whereas probability weighting weights the distribution each classifier produces,

$$\hat{p}(y=i|\boldsymbol{E},\boldsymbol{x}) \propto \sum_{j=1}^{k} w_j \hat{p}_j(y=i|M_j,\boldsymbol{x}). \tag{4.1}$$

Figure 4.1 gives an overview of the components of CAWPE that make it different to majority voting.

Our approach is based on the idea of building a smaller number of effective classifiers and combining the output rather than learning a huge number of weak classifiers. The rationale for using the probability estimates rather than the predic-

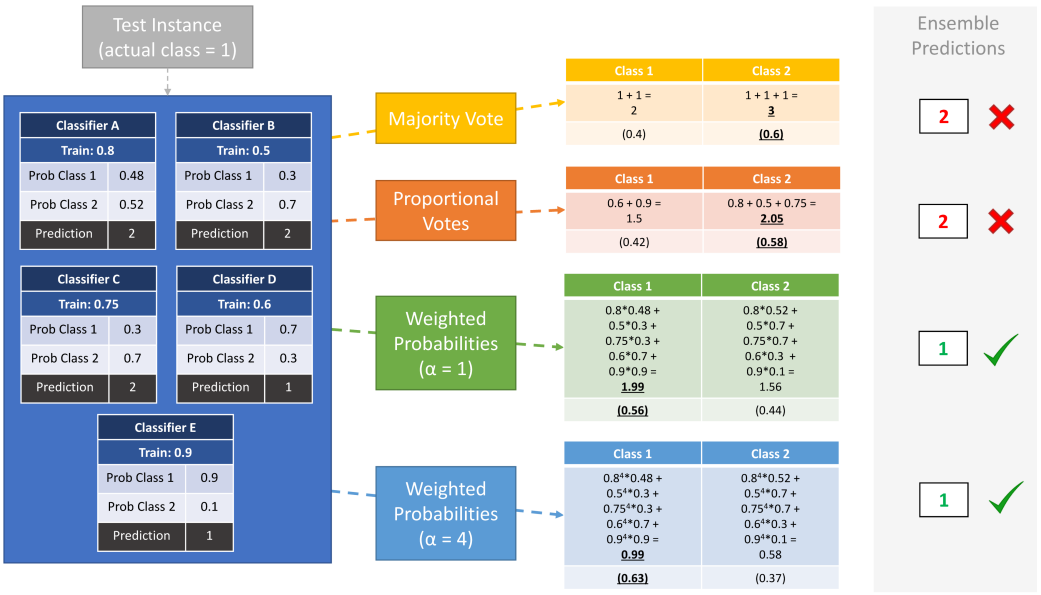


Fig. 4.1 Illustration of the different effects of combination and weighting schemes on a toy instance classification. Each stage progressively pushes the predicted class probabilities further in the correct direction for this prediction.

tions is that they will contain more information than a point estimate, and with fewer classifiers we need to capture all information available. With 500 base classifiers the voting mechanism is less important than with 5 classifiers, since averaging over 500 votes is likely to have lower variance than averaging over 5 votes.

The construction of the CAWPE ensemble involves estimating the classification accuracy of each base classifier on the train data through a ten-fold cross-validation, then constructing a model of each base classifier on the whole train data. Classifying a new case, described in Algorithm 1 and Equation 4.1, requires obtaining a probability estimate of each class from all the base classifiers, weighting these by the cross-validation accuracy raised to the power α (the only parameter of the approach), then either normalising if probability estimates are required or returning the index of the maximum probability if a prediction is needed.

As α increases, the weightings of classifiers found to be stronger on the training data relative to the rest are increased, until the ensemble becomes functionally identical to the single best classifier in training. Conversely, when alpha is zero all

Algorithm 1 CAWPE classify(A test case $\boldsymbol{x}$)

Given a set of classifiers $< M_1, \ldots, M_k >$, an exponent α, a set of weights w_i, and the number of classes c
$\{\hat{p}_1, \ldots, \hat{p}_c\} = \{0, \ldots, 0\}$
for $i \leftarrow 1$ to k **do**
  for $j \leftarrow 1$ to c **do**
    $\hat{q}_j \leftarrow \hat{p}((y = j|M_i, \boldsymbol{x})$
    $\hat{p}_j \leftarrow \hat{p}_j + w_i^{\alpha} \cdot \hat{q}_j$

return $\arg\max_{j=1\ldots c} \hat{p}_j$

members will be equally weighted. Therefore, on a high level, the α parameter defines the degree to which the base classifiers' error estimates should be trusted in guiding the ensemble's output. Set α too high, and all but the best classifier's outputs are diminished. Set α too low, and the competitive advantage that the best individual is estimating it has is potentially wasted. The quality of the error estimate is key to this process, of course, thus the use of cross-validation as opposed to a single validation set as used in a number of previous works [69].

The optimal value of α will therefore allow the strongest classifiers to steer the ensemble, but enable them to be overruled when sufficiently outvoted. This value will be dependent on the relative performances and distribution of probabilistic outputs of the base classifiers on the given dataset. To keep in line with the general ethos of simplicity, we remove the need to tune α and potentially overfit it by fixing α to 4 for all experiments and all component structures presented. We chose the value 4 fairly arbitrarily as a sensible starting point before running any experiments. In Section 4.4 we revisit the importance of the α parameter and whether it could benefit from tuning, as well other design decisions we have made.

4.3 Results

We perform experiments across the UCI and UCR dataset archives as described in Chapter 2, Section 2.5.1. Each classifier on each dataset is evaluated over 30 resamples. The initial experimentation and bulk of the analysis shall be conducted on the UCI archive for its wider applicability and for ease of comparison to other results following Delgado et al. [44]. However, transference and generality of the findings to the UCR timeseries archive is a key additional step and motivated by Bagnall et al. [6], which implemented 18 state-of-the-art TSC classifiers within a common framework and evaluated them on 85 datasets in the archive. The best performing algorithm, the Collective of Transformation-based Ensembles (COTE), was a heterogeneous ensemble of strong classifiers. These results were our primary motivation for further exploring heterogeneous ensembles for classification problems in general.

We demonstrate the benefits of the CAWPE scheme through a sequence of experiments to address the following questions:

- Does CAWPE improve heterogeneous base classifiers (Section 4.3.1)?
- Is CAWPE better on average than alternative heterogeneous ensemble schemes all using the same base classifiers and error estimates (Section 4.3.2)?
- Is CAWPE better on average than homogeneous ensembles (Section 4.3.3)?
- How does CAWPE compare to tuned versions of classifiers commonly considered state of the art (Section 4.3.4)?
- Do the results generalise to other data (Section 4.3.5)?

Throughout, we make the associated point that CAWPE is significantly better than its components when they are approximately equivalent. CAWPE has a single

parameter, α, which is set to the default value of 4 for all experiments. We stress that we perform no tuning of CAWPE's parameter α: it simply combines classifier output using the algorithm described in Algorithm 1. We investigate the sensitivity of CAWPE to α in Section 4.4.3.

We present results in this section through critical difference diagrams which display average rankings. A full list of the average scores for each classifier is provided in Table 6.4 in the Appendix, while further spreadsheets are available on the accompanying website.

4.3.1 Does CAWPE improve heterogeneous base classifiers?

Ensembling multiple classifiers inherently involves more work than using any single one of them. As a basic sanity check, we assess whether applying CAWPE to a random set of classifiers improves performance. We randomly sampled 5 out of 22 classifiers available in Weka and constructed CAWPE on top of them. Over 200 random configurations, CAWPE was significantly more accurate than the individual component with the best average rank on 143 (71.5%), and insignificantly more accurate on a further 34 (17%), over the 121 UCI datasets. CAWPE was never significantly worse than the best individual component. Note that many of these sets contain components that are significantly different, with average accuracies across the archive ranging between 81.4% and 62.7%.

To avoid confusion as to the components of any CAWPE instantiation, we continue the evaluation with two sets of base classifiers. The first, simpler set contains well known classifiers that are fast to build. These are: logistic regression (Logistic); C4.5 decision tree (C4.5); linear support vector machine (SVML); nearest neighbour classifier (NN); and a multilayer perceptron with a single hidden layer (MLP1). These classifiers are each distinct in their method of modelling the data, and are roughly equivalent in performance. We call this version CAWPE-S.

The second set of five classifiers are more complex, and generally considered more accurate than the previous set. These are: random forest (RandF); rotation forest (RotF); a quadratic support vector machine (SVMQ); a multi layer perceptron implementation with two hidden layers (MLP2); and extreme gradient boosting (XGBoost). We call CAWPE built on this second set of advanced classifiers CAWPE-A.

In Figure 4.2 we compare CAWPE-A and CAWPE-S against their respective base classifiers in terms of accuracy. In both cases, CAWPE is significantly better than all components. CAWPE also significantly improves of all the base components in terms of balanced accuracy, AUROC, and log likelihood.

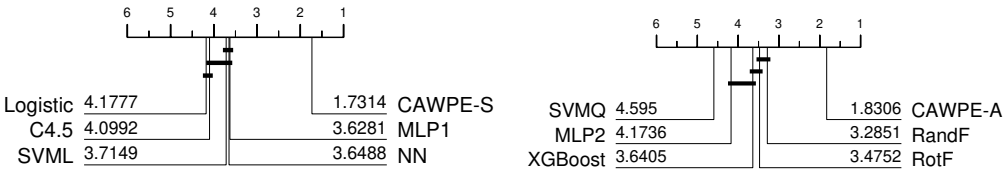


Fig. 4.2 Critical difference diagrams CAWPE-S with its base classifiers (left), and CAWPE-A with its base classifiers (right). Ranks formed on test set accuracy averaged over 30 resamples.

The improvement is not particularly surprising for CAWPE-S, since the benefits of ensembling weaker learners are well known. It is perhaps more noteworthy, however, that learners often considered state-of-the-art on arbitrary tabular data such as random forest, rotation forest and XGBoost, are improved by inclusion in the CAWPE-A ensemble. This improvement is achieved at a computational cost. The CAWPE scheme will require more computation than using a single classifier, since a cross-validation procedure is required for each base classifier. If a ten-fold cross-validation is used, as we do in all our experiments, CAWPE requires approximately 50 times longer to train than the average training time of its five base classifiers. In terms of time taken to predict a new test case, CAWPE simply needs five times the average prediction time of the base classifiers. We have experimentally verified this

is the case, but exclude results for brevity. This constant time overhead is easy to mitigate against: it is trivial to distribute CAWPE's base classifiers and even the cross-validation folds for each classifier can easily be parallelised.

4.3.2 Is CAWPE better on average than alternative heterogeneous ensemble schemes?

We compare the particular weighting scheme used in CAWPE, over the -S and -A base classifier sets, to well known alternative weighting, selection and stacking approaches described in Chapter 2, Section 2.4.1.1. For ease of reference, the weighted ensembles are: Majority Vote (MV); Weighted Majority Vote (WMV); Recall (RC); Naive Bayes (NBC) and our scheme (CAWPE). The selection ensembles are: Pick Best (PB); and Ensemble Selection (ES). The stacking schemes are: stacking with multi-response linear regression (SMLR); stacking with multi-response linear regression on extended features (SMLRE); and stacking with multi-response model trees (SMM5). Recall that HESCA, evaluated throughout Chapter 3, is effectively a WMC but with a particular defined base classifier set. Here, we generalise away from this.

It should be noted that of course this is not an exhaustive list of possible alternative heterogeneous ensembles. Rather, we believe that these constitute a fair representation of the different types of classifier combination schemes, provided a consistent and relatively small base classifier set.

In each comparison, all ensembles use the same set of base classifiers, so the only source of variation is the ensemble scheme. Algorithms such as ensemble selection were originally described as using a single validation set to assess models. However, cross-validation will on average give a better estimate of the true error than a single hold-out validation set [69]. Given that CAWPE uses cross-validation error estimates and that these estimates are already available to us, we also use

these for all ensembles. Hence, we are purely testing the ability of the ensembles to combine predictions with exactly the same meta-information available.

Figure 4.3 shows the summary ranks of ten heterogeneous ensembles built on the simpler classifier set on the 121 UCI datasets using four performance metrics. CAWPE-S is highest ranked for error and in the top clique for both error and balanced error. It is significantly better than all other approaches for AUC and NLL. It has significantly lower error than all but SMLR, and significantly lower balanced error than all but NBC.

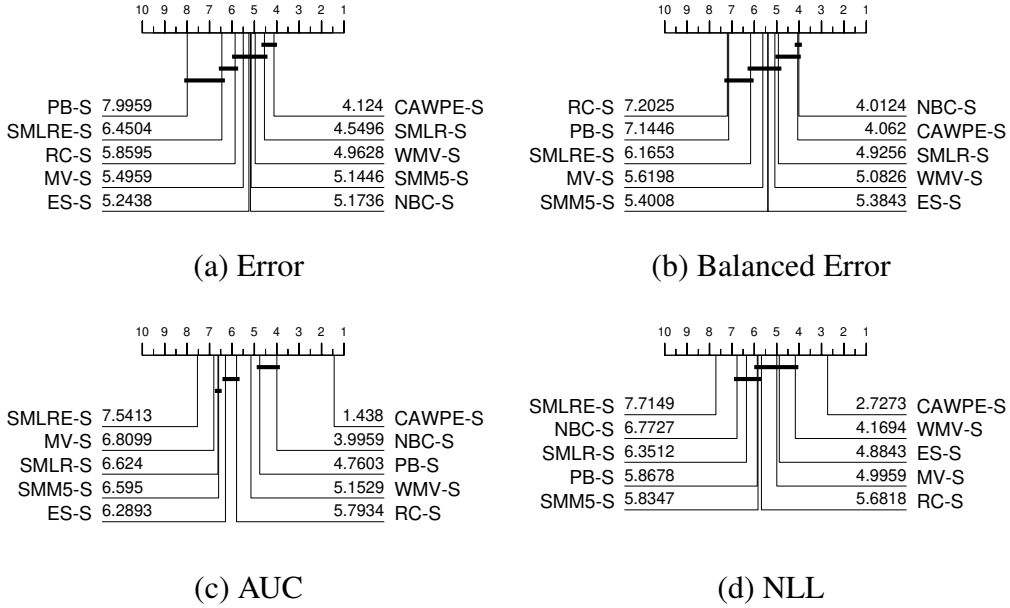
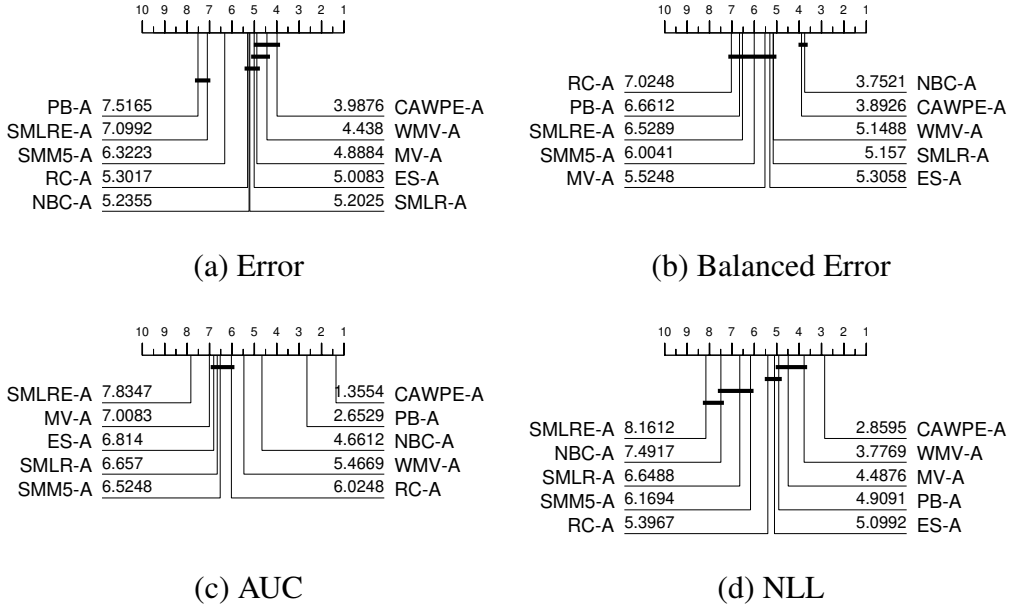


Fig. 4.3 Critical difference diagrams for ten heterogeneous ensemble classifiers on 121 UCI data built using logistic, C4.5, SVML, NN and MLP1 base classifiers.

Figure 4.4 shows the summary ranks of the same ten heterogeneous ensembles on the 121 UCI datasets using the more advanced classifiers. The pattern of results is very similar to those for the simple classifiers. CAWPE-A is top ranked for error and in a clique with majority vote and weighted majority vote. For balanced error, it is not significantly different to NBC and is significantly better than the others. For both AUC and NLL, it is significantly better than all the other methods. Considering the results for both CAWPE-S and CAWPE-A, it is apparent that the CAWPE scheme is more consistent than other approaches, since it is the only



(a) Error (b) Balanced Error

(c) AUC (d) NLL

Fig. 4.4 Critical difference diagrams for ten heterogeneous ensemble classifiers on 121 UCI data built using Random Forest (RandF), Rotation Forest (RotF), Support Vector Machine with a quadratic kernel (SVMQ), a two layer multilayer perceptron (MLP2) and extreme gradient boosting (XGBoost) base classifiers.

algorithm in the top clique for all measures for both sets of classifiers. We think this suggests that the CAWPE scheme on this data is the best heterogeneous ensemble technique, at least for the simple and advanced component sets studied.

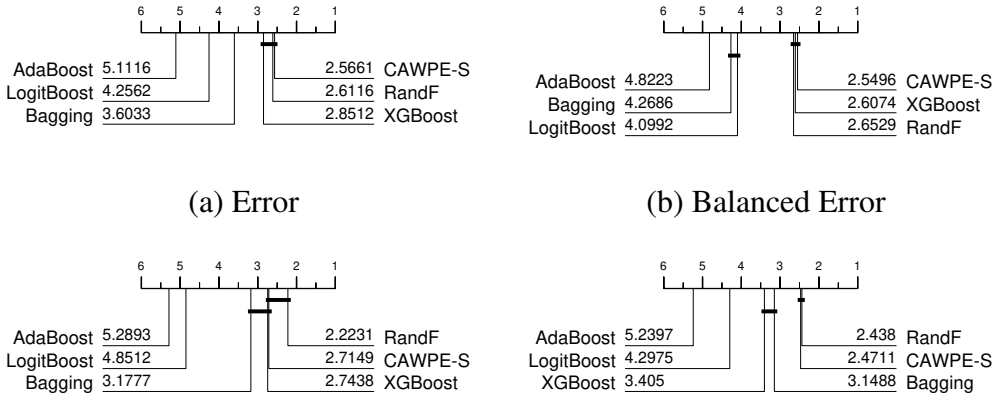
Given the ensembles are using the same base classifiers and accompanying error estimates, and these are all good classifiers in their own right, we would expect the actual differences in average error to be small, and this is indeed the case (see Table 6.4 in Appendix). Nevertheless, the weighting scheme used in CAWPE is significantly better than nearly all the other methods using the four metrics.

In conclusion, CAWPE makes sets of approximately equivalent classifiers significantly better, and is competitive with or generally better than commonly used weighting, selection and stacking schemes when the number of classifiers is small. Given how simple CAWPE is, we believe it is a sensible starting point for any attempt at combining small numbers of base classifiers on an arbitrary problem.

The question then is, should you heterogeneously ensemble at all, or rather should you focus efforts into improving a single model?

4.3.3 Is CAWPE better on average than homogeneous ensembles?

We examine how CAWPE-S compares to five homogeneous ensembles that each employ 500 duplicates of the same base classifier. CAWPE-A, which includes RandF and XGBoost in its base classifier set, is significantly better on all four performance metrics than both them and all the homogeneous ensembles evaluated here (see Figure 4.2, the results are available on the accompanying website). However, this improvement requires roughly 50 times the computational effort of XGBoost or Random Forest alone. We are more interested in assessing how the simpler and faster CAWPE-S compares with homogeneous ensembles.



(a) Error (b) Balanced Error

(c) AUC (d) NLL

Fig. 4.5 Critical difference diagrams for CAWPE (built using logistic, C4.5, SVML, NN and MLP1 base classifiers) against 5 homogeneous ensemble classifiers on 121 UCI data.

Figure 4.5 shows the results of five ensembles each with 500 base classifiers and CAWPE-S. We observe that CAWPE-S is significantly more accurate than AdaBoost, LogitBoost and Bagging, and not significantly worse than Random Forest and XGBoost. With minimal effort using standard classifiers we have pro-

Table 4.1 Summaries of train times for CAWPE-S and the homogeneous ensembles. All times are in seconds, and are averaged across the 121 UCI data.

Classifier	CAWPE-S	LogitBoost	RandF	XGBoost	Bagging	AdaBoost
Mean	524.9	302.2	111.9	46.8	22.7	7.8
Median	13.7	8.9	6.9	2.1	0.7	0.06

duced an ensemble that is not significantly worse than state-of-the-art homogeneous ensembles.

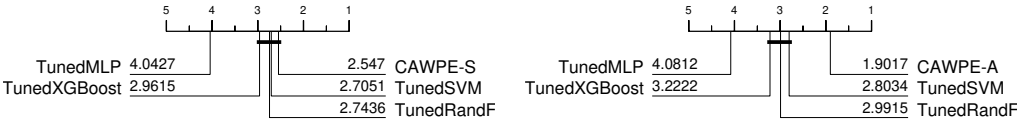
Table 4.1 summarises the train times of CAWPE-S and the homogeneous ensembles in seconds. CAWPE on this simpler component set has a much larger mean train time than RandF and XGBoost. This largely comes down to the logistic regression component, which takes a relatively much longer amount of time on datasets with larger numbers of classes. The median times are closer, however XGBoost especially still achieves predictive performance not significantly different to that of CAWPE-S in much shorter times on average.

These timings should be interpreted with the understanding that XGBoost is a highly optimised library, while the logistic and MLP1 implementations in particular are relatively straight forward and unoptimised implementations in Java. The fact that CAWPE-S has a median train time within the same order of magnitude as XGBoost while not being significantly less accurate is, we think, a positive result.

Note that due to human error on the part of the authors, Rotation Forest is missing from this comparison versus competing homogeneous ensembles. Based on the author's experience and the previous experiments of this Chapter, we would expect Rotation Forest to be in the same clique as RandF and CAWPE, possibly insignificantly higher-ranked in the same manner as RandF.

4.3.4 How does CAWPE compare to tuned classifiers?

In Section 4.3.1 we showed the ensemble scheme outperforms its set of base classifiers. However, finding the weights requires an order of magnitude more work than building a single classifier because of the ten fold cross-validation across the different components. Given it is widely accepted that tuning parameters on the train data can significantly improve classifier accuracy [2], perhaps a carefully tuned classifier will do as well as or better than CAWPE built on untuned classifiers. To investigate whether this is the case, we tune an SVM with a radial basis function kernel (SVMRBF), XGBoost, MLP and a random forest and compare the results to CAWPE-S and CAWPE-A. We tune by performing a ten-fold cross-validation on each train resample for a large number of possible parameter values, described in Table 4.2. This requires a huge computational effort. We can distribute resamples and parameter combinations over a reasonably sized cluster. Even so, considerable computation is required; we were unable to complete a full parameter search for 4 datasets (within a 7 day limit): adult; chess-kvrk; miniboone; and magic. To avoid bias, we perform this analysis without these results.



(a) (b)

Fig. 4.6 Average ranked errors for (a) CAWPE-S and (b) CAWPE-A against four tuned classifiers on 117 datasets in the UCI archive. The datasets adult, chess-krvk, miniboone and magic are omitted due to computational restraints.

Figure 4.6 compares CAWPE-S and CAWPE-A to tuned versions of MLP, XGBoost, RandF and SVM. On average, CAWPE-S, containing the five simpler untuned base classifiers (Logistic, C4.5, SVML, NN and MLP1), is significantly better than the tuned MLP and not significantly worse than tuned versions of XGBoost, SVMRBF and Random Forest (Figure 4.6(a)). The highest ranked tuned

Table 4.2 Tuning parameter ranges for SVMRBF, Random forest, MLP and XGBoost. c is the number of classes and m the number of attributes

Classifier	Total	Parameter	Range
SVMRBF	1089	Regularisation C (33 values)	$\{2^{-16}, 2^{-15}, \ldots, 2^{16}\}$
		variance γ (33 values)	$\{2^{-16}, 2^{-15}, \ldots, 2^{16}\}$
Random Forest	1000	number of trees (10 values)	$\{10, 100, 200, \ldots, 900\}$
		feature subset size (10 values)	$\{\sqrt{m}, (\log_2 m + 1), \frac{m}{10}, \ldots, \frac{m}{3}\}$
		max tree depth (10 values)	$\{0, \frac{m}{9}, \frac{m}{8} \ldots, m\}$
MLP	1024	hidden layers (2 values)	$\{1, 2\}$
		nodes per layer (4 values)	$\{c, m, m+c, \frac{m+c}{2}\}$
		learning rate (8 values)	$\{1, \frac{1}{2}, \frac{1}{4}, \ldots, 1/(2^7)\}$
		momentum (8 values)	$\{0, \frac{1}{8}, \frac{2}{8}, \ldots, \frac{7}{8}\}$
		decay (2 values)	$\{true, false\}$
XGBoost	625	number of trees (5 values)	$\{50, 100, 250, 500, 1000\}$
		learning rate (5 values)	$\{0.01, 0.05, 0.1, 0.2, 0.3\}$
		max tree depth (5 values)	$\{2, 4, 6, 8, 10\}$
		min child weight (5 values)	$\{1, 3, 5, 7, 9\}$

classifier is SVM, but it is still ranked lower than CAWPE-S. This despite the fact that CAWPE-S is two orders of magnitude faster than the tuned SVM and at least one order of magnitude faster than tuned Random Forest, MLP and XGBoost. Sequential execution of CAWPE-S for miniboone (including all internal cross-validation to find the weights) is 5 hours. For TunedSVM, ten-fold cross-validation on 1089 different parameter combinations gives 10890 models trained for each resample of each dataset. For the slowest dataset (miniboone), sequential execution would have taken more than 6 months. Of course, such extensive tuning may not be necessary. However, the amount and exact method of tuning to perform is in itself very hard to determine. Our observation is that using simple approach such as CAWPE-S avoids the problem of guessing how much to tune completely.

If we use CAWPE-A, containing the more advanced components (RandF, RotF, SVMQ, MLP2 and XGBoost), we get a classifier that is significantly more accurate

than any of the individuals (Figure 4.6(b)). CAWPE-A takes significantly longer to train than CAWPE-S, but it is still not slower on average than the tuned classifiers. We are not claiming that CAWPE-A is significantly faster than tuning a base classifier in the general case, because this is obviously dependent on the tuning strategy. CAWPE-A involves a ten fold cross-validation of five classifiers, so it is going to be comparable in run time to one of these single classifiers tuned over 50 parameter settings. However, our experiments demonstrate that tuning a single base learner over a much larger parameter space does not result in as strong of a model, on average.

Our goal is not to propose a particular set of classifiers that should be used with CAWPE. Rather, we maintain that if one has some set of classifiers they wish to apply to problem, ensembling them using CAWPE is generally at least as strong as other heterogeneous ensemble schemes when we have a relatively small number of base classifiers, that it significantly improves base classifiers that are approximately equally strong, and that the degree of improvement is such that state-of-the-art level results can be achieved with minimal effort. Once a classifier is trained and the results are stored, ensembling is very quick. To perhaps belabour the point, we ensembled the four tuned classifiers using the parameter ranges given in Table 4.2 and the resulting classifier was significantly better than the components in a manner reflecting the patterns observed in Section 4.3.1.

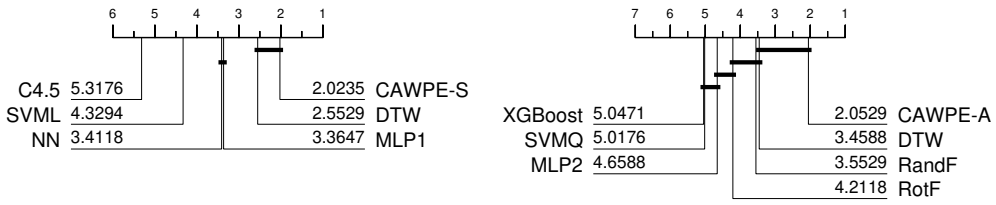
4.3.5 Does the CAWPE performance generalise to other datasets?

Our interest in heterogeneous ensembles originated in TSC problems, where we ensemble over different representations of the data in a style similar to CAWPE [88]. TSC involves problems where the attributes are ordered (not necessarily in time) and all real valued. The UCR repository for TSC contains problems from a wide range of domains such as classifying image outlines, EEG and spectrographs. There

are currently 85 datasets, with diverse data characteristics. A full list of the 85 datasets is listed in the Appendix in Table 6.3.

Traditionally, dynamic time warping distance (with window size set through cross-validation) [115] with a 1-nearest neighbour classifier (referred to as just DTW henceforth) has been considered the benchmark algorithm for this type of problem. In recent years, a range of bespoke algorithms have been proposed in high impact journals and conferences. A large-scale experimental evaluation [6] found that of 18 such algorithms, only 13 were significantly better (in terms of accuracy) than DTW.

Our goal is to test how well the results observed for CAWPE on the UCI data generalise to other data, by testing whether CAWPE significantly improves over its components on the UCR archive also. To do so, we ignore the ordering of the series and treat each time step in the series as a feature for traditional vector-based classification. The UCR datasets generally have many more features than the UCI data. This has meant we have had to make one change to CAWPE-S: we remove logistic regression because it cannot feasibly be built on many of the data. Since DTW is a 1-nearest neighbour classifier, it always produces 0/1 probability estimates. Because of this, we omit a probabilistic evaluation using AUC and NLL, as it has little meaning for DTW.



(a) (b)

Fig. 4.7 Average ranked errors for DTW against (a) CAWPE-S and its components and (b) CAWPE-A and its components on the 85 datasets in the UCR archive.

Figure 4.7 shows the critical difference diagrams for accuracy of CAWPE-S, CAWPE-A, their respective constituents, and DTW. Both sets of base classifiers are

significantly improved by CAWPE once more. These results closely mirror those on the UCI datasets presented above. Furthermore, neither of the CAWPE versions are significantly worse than DTW and both have higher average rank. This should be considered in the context that neither classifier takes advantage of any information in the ordering of attributes. Despite this, CAWPE-A has a higher average rank than 9 of the 18 bespoke TSC algorithms evaluated by Bagnall et al. [6], and is not significantly worse than 11 of them. CAWPE, a simple ensemble using off the shelf components and a simple weighting scheme, has been made as accurate as complex algorithms that use a range of complicated techniques such as forming bags of patterns, using edit distance based similarity, differential based distances, compression techniques and decision trees based on short subseries features.

Using standard classifiers for TSC is unlikely to be the best approach. The best performing TSC algorithm in the study [6], significantly more accurate than all the others, was the Collective of Transformation-based Ensembles (COTE) [7]. It has components built on different representations of the data. COTE uses an ensemble structure that is the progenitor of CAWPE. The latest version of COTE, HIVE-COTE [88] uses weighted majority voting for five modularised classifier components defined on shapelet, elastic distance, power spectrum, bag-of-words and interval based representations, and is significantly more accurate than the previous version, flat-COTE, and all of the competing algorithms. HIVE-COTE exploits the diversity of the representations through an ensemble scheme. We address the question of whether CAWPE is the best ensemble scheme for HIVE-COTE.

Figure 4.8 shows how HIVE-COTE performs when we incrementally add in the CAWPE combination scheme methods. The left most version, weighted majority vote, is the classifier used by the original HIVE-COTE [88]. Raising the weight to the power of four significantly reduces error. Switching to using probabilities is significantly better than either weighted voting scheme. Using CAWPE (probs, a=4 in Figure 4.8) is significantly better than all variants. It is not just a matter of

tiny improvements in accuracy improving the ranks. The overall mean accuracy over all problems for HIVE-COTE using CAWPE is 87.16%, whereas the accuracy reported originally [88] using WMV is 85.97%. An overall improvement of over 1% for such a simple change is hugely valuable. For context, the average accuracy of DTW is 77.7%.

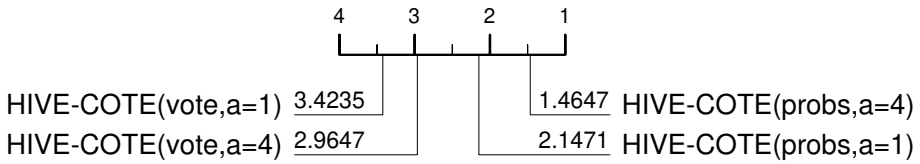


Fig. 4.8 Average ranked errors for HIVE-COTE using four variants of the combination schemes on the UCR datasets.

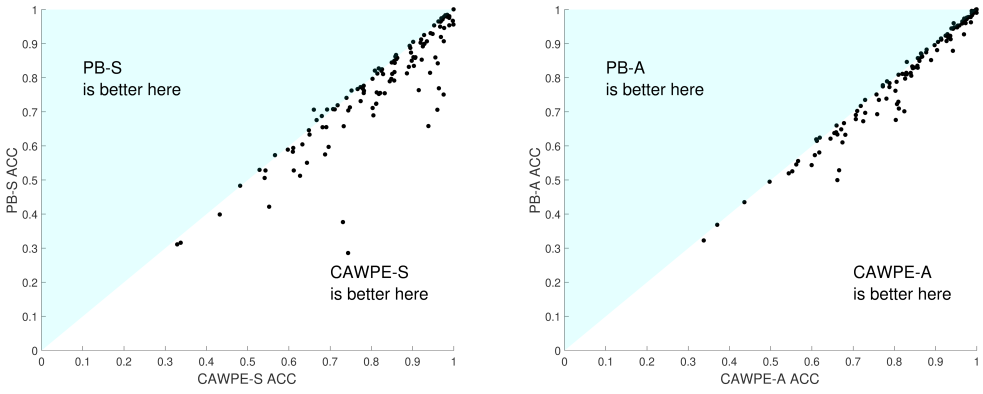
4.4 Analysis

We perform a more in-depth analysis of results to determine whether there are any patterns in the results that indicate when and why CAWPE performs well. We compare various facets of performance against choosing the best component on any given dataset (Section 4.4.1). We then perform an ablative study of CAWPE (Section 4.4.2), and a sensitivity study of its parameter, α (Section 4.4.3).

4.4.1 CAWPE vs Pick Best Exploratory Analysis

Given CAWPE ensembles based on estimates of accuracy obtained from the train data and gives increasingly larger weights to the better classifiers, it seems reasonable to ask, why not just choose the single classifier with the highest estimate of accuracy? Figure 4.3 demonstrated that it is on average significantly worse choosing a single classifier than using the CAWPE ensembles. When comparing algorithms over entire archives, we get a good sense of those which are better for general purpose classification. However, differences in aggregated ranks do not

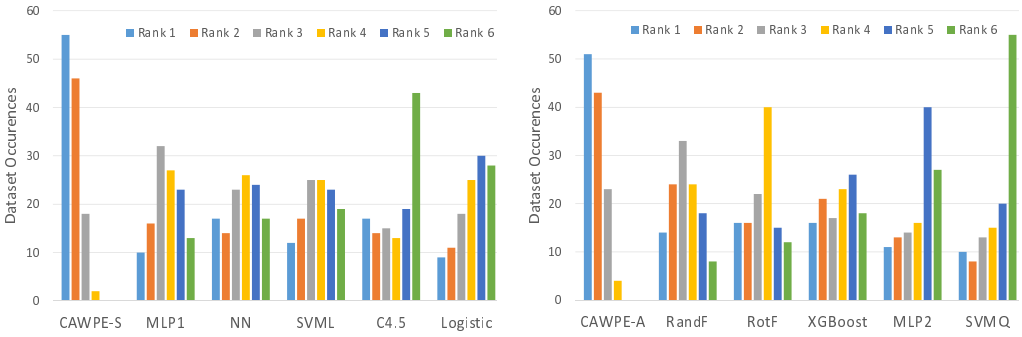
tell the whole story of differences between classifiers. It could be the case that CAWPE is just more consistent that its components: it could be a jack of all trades ensemble that achieves a high ranking most of the time, but is usually beaten by one or more of its components. A more interesting improvement is an ensemble that consistently achieves higher accuracy than all of its components. For this to happen, the act of ensembling needs to not only cover for the weaknesses of the classifiers when in their suboptimal domains, but accentuate their strengths when within their specialisation too. Figure 4.9 shows the scatter plots of accuracy for choosing the best base classifier from their respective component sets against using CAWPE. This demonstrates that CAWPE has higher accuracy than Pick Best on the majority of problems, and that the differences are not tiny.



(a) (b)

Fig. 4.9 Accuracy of (a) CAWPE-S and (b) CAWPE-A vs picking the best component.

Figure 4.10 shows the counts of the rankings achieved by CAWPE built on the simpler (a) and advanced (b) components, in terms of accuracy, over the 121 UCI datasets. CAWPE is the single best classifier far more often than any of its components, and is in fact more often the best classifier than second best. Both versions of CAWPE are never ranked fifth or sixth, and very rarely ranked fourth, demonstrating the consistency of the improvement. This suggests that the simple combination scheme used in CAWPE is able to actively enhance the predictions of its locally specialised members, rather than just achieve a consistently good rank.



(a) (b)

Fig. 4.10 Clustered histograms of accuracy rankings over the 121 UCI datasets for (a) CAWPE-S and (b) CAWPE-A and their respective components. For each classifier, the number of occurrences of each rank being achieved relative to the other classifiers is shown.

Table 4.3 CAWPE-S vs pick best split by train set size. The three datasets with the same average error have been removed (acute-inflammation, acute-nephritis and breast-cancer-wisc-diag). If there is a significant difference within a group (tested using a Wilcoxon sign rank test) the row is in bold.

#Train Cases	#Problems	#CAWPE-S WINS	Mean Error Difference
1-100	**28**	**21**	**1.49%**
101-500	**46**	**36**	**0.71%**
501-1000	**12**	**11**	**1.51%**
1001-5000	23	11	0.16%
>5001	9	2	0.02%

For clarity we restrict further analysis to the CAWPE-S results. Comparable results for CAWPE-A are available on the accompanying website.

Comparing overall performance of classifiers is obviously desirable; it addresses the general question: given no other information, what classifier should I use? However, we do have further information. We know the number of train cases, the number of attributes and the number of classes. Does any of this information indicate scenarios where CAWPE is gaining an advantage? The most obvious factor is train set size, since picking the best classifier based on train estimates is likely to be less reliable with small train sets.

Table 4.3 breaks down the results of CAWPE-S compared to Pick Best by train set size. With under 1000 train cases, CAWPE-S is clearly superior. With

1000-5000 cases, there is little difference. With over 5000 cases, CAWPE-S is better on just 2 of 9 problems, but there is only a tiny difference in error. This would indicate that if one has over 5000 cases then there may be little benefit in using CAWPE-S, although it is unlikely to be detrimental. Analysis shows there is no detectable significant effect of number of attributes. For the number of classes, there is a benefit for CAWPE-S on problems with more than 5 classes. CAWPE-S wins on 62% of problems with five or fewer classes (53 out of 85) and wins on 85% of problems with 6 or more (28 out of 33). This is not unexpected, as a large number of classes means fewer cases per class, which is likely to introduce more noise into the estimate of error.

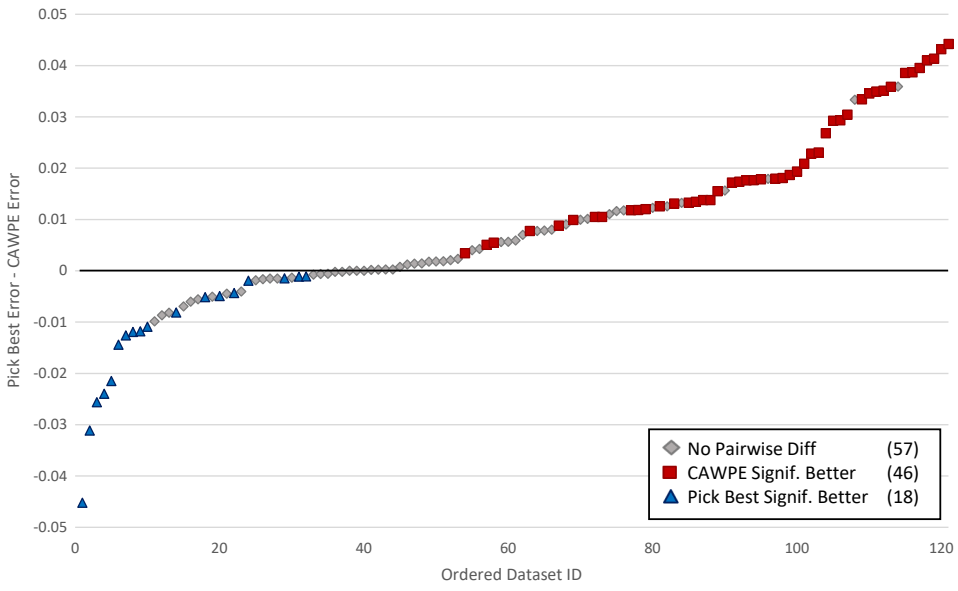


Fig. 4.11 The difference in average errors in increasing order between CAWPE-S and picking the best classifier on each dataset. Significant differences according to paired t-tests over folds are also reported. CAWPE-S is significantly more accurate on 46, the best individual classifier on 18, and there is no significant difference on 57.

Despite using the same classification algorithms, not all of the differences between pick best and CAWPE-S are small in magnitude. Figure 4.11 shows the ordered differences between the two approaches. The largest difference in favour of CAWPE-S (averaged over 30 folds) is 4.42% (on the arrhythmia dataset) and in

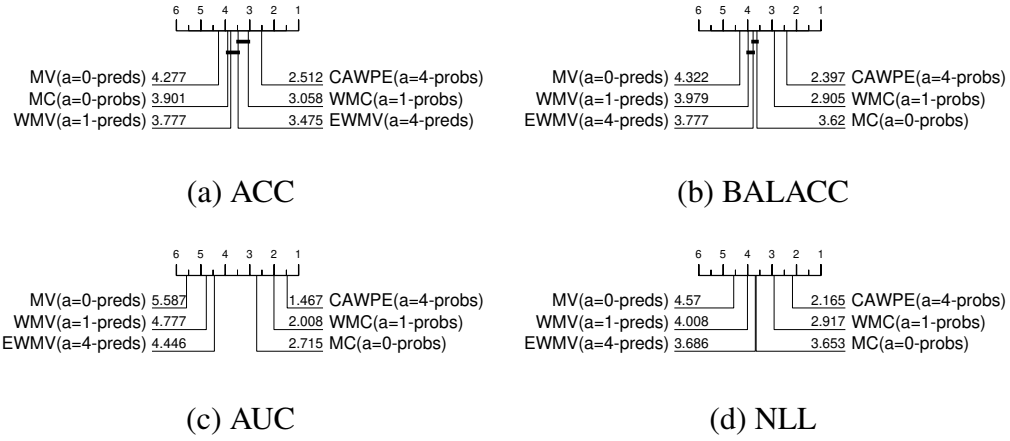
favour of pick best 4.5% (on energy-y1). This demonstrates the importance of the selection method for classifiers; it can cause large differences on unseen data.

This analysis indicates that CAWPE-S is likely to be a better approach than simply picking the best when there is not a large amount of training data, there are a large number of classes and/or the problem is hard. Overall, CAWPE requires almost no extra work beyond pick best and yet is more accurate.

4.4.2 CAWPE Ablative Study

CAWPE belongs to the family of ensemble schemes broadly categorised as weighted output combination. We found in Section 4.3 that both CAWPE-S and CAWPE-A are significantly better than the most common instantiations of this type of ensemble; majority vote and weighted majority vote. The major design components of CAWPE are the fact it uses the probabilistic outputs of its base classifiers and the emphasising of differences in weights by using α set to 4. Figure 4.8 has already shown that that both of these factors result in significant improvement of the TSC algorithm HIVE-COTE. Here we wish to delve further into the contribution that each factor of CAWPE has on its performance. For brevity, we perform all analysis using the CAWPE-S set of simpler classifiers.

We split CAWPE based on these two factors, building up from majority vote to CAWPE: the use of the base classifiers' probabilities (probs) or predictions (preds); and the extent to which we make use of the base classifiers' cross-validation accuracy to weight their contribution: none at all (a=0); standard weighting (a=1); and extenuated weighting (a=4). Figure 4.12 details the results of a comparison between all combinations of these factors. To better ground these results in the context of the previous comparison to other heterogeneous ensembles in general in Section 4.3.2, we reuse and define new labels relevant to combinations of these factors of weighted output combination. These are: Majority Vote (MV:



(a) ACC (b) BALACC

(c) AUC (d) NLL

Fig. 4.12 Critical difference diagrams of the stages of progression from a simple majority vote up to CAWPE, on the 121 datasets of the UCI archive using the CAWPE-S variant.

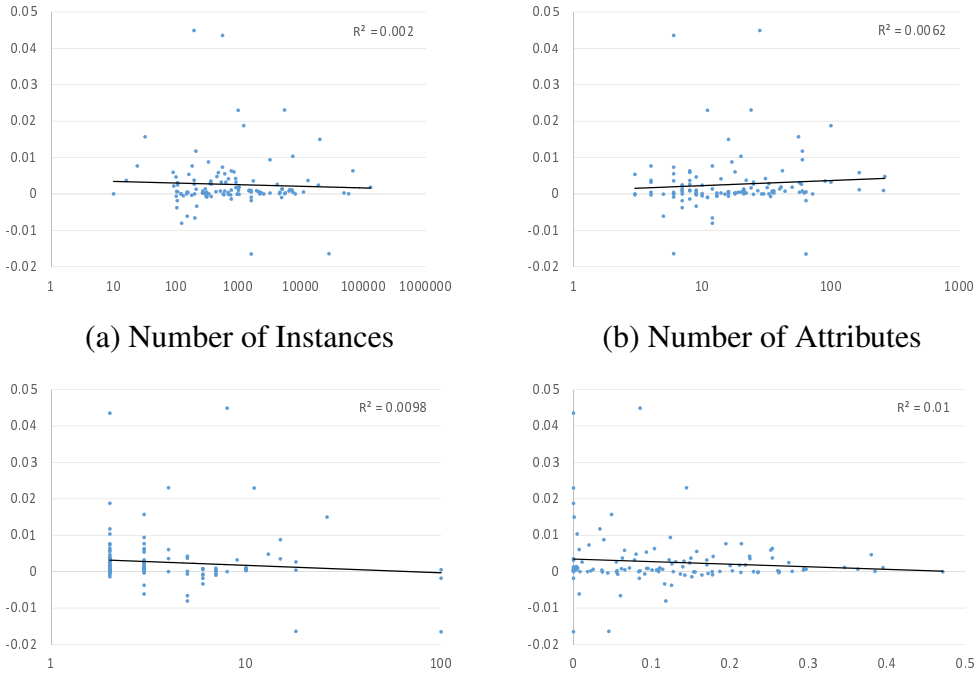
a=0,preds); Majority Confidence (MC: a=0,probs); Weighted Majority Vote (WMV: a=1,preds); Weighted Majority Confidence (WMC: a=1,probs); Exponentially Weighted Majority Vote (EWMV: a=4,preds); and finally Exponentially Weighted Majority Confidence (CAWPE: a=4, probs).

These diagrams confirm some suspicions. Firstly, for equal values of α, it is always better to use probabilities instead of predictions. For AUC and NLL, the performance metrics most relevant to probabilistic output, the use of probabilities is better even regardless of the value of α. Secondly, the use of a weighting scheme, and then further increasing the value of α to 4 also always provides improvement on average.

The improvement from increasing α to 4 is consistent, too, providing in some instances surprising improvements in absolute accuracy. When directly comparing CAWPE (α=4, probs) to WMC (α=1, probs), CAWPE wins on 86 datasets and loses on 28. The largest reduction in error was 4.49% on the flags dataset, with the largest increase in error being 1.65% on plant-shape.

Figure 4.13 displays scatter plots to demonstrate these findings. Against differences in error between CAWPE and WMC, it plots a four dataset characteristics: the

number of instances; number of attributes; number of classes; and class imbalance. For this purpose, the class imbalance of a dataset is informally calculated as the average difference between each class' actual proportional representation in the dataset, and its expected value, $1/c$. These confirm visually that there is no obvious relationship between the improvement α provides and any of these characteristics.



(a) Number of Instances

(b) Number of Attributes

(c) Number of Classes

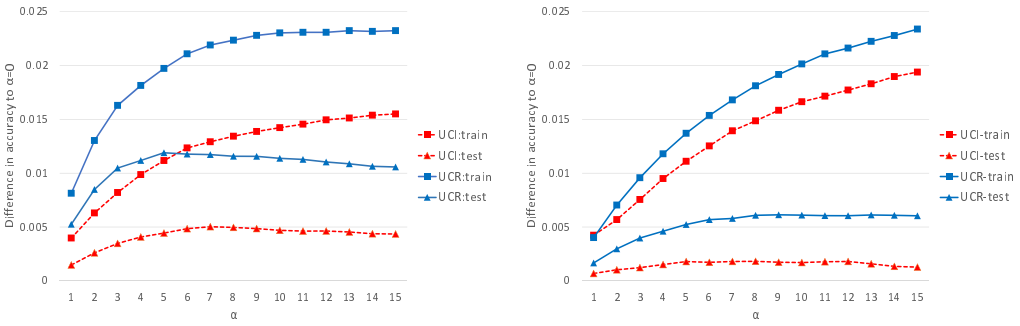
(d) Average Class Imbalance

Fig. 4.13 Four plots of the difference in error between CAWPE (α=4,probs) and WMC (α=1,probs), against different dataset characteristics. Above zero CAWPE wins, below zero WMC wins. Trend represented by solid black line, R^2 reported in top-right corner.

4.4.3 CAWPE Sensitivity Analysis

Section 4.4.2 has shown that exaggerating the weights of classifiers using α gives a significant increase in performance over standard weighted averaging of probabilities, even with all else being equal. As stated at the end of Section 4.2, the value of α was fixed to 4 for CAWPE for all experiments reported throughout the previous sections. This value was decided on while developing HIVE-COTE.

Having performed our experiments with $\alpha = 4$, we were interested to find out how sensitive the performance of CAWPE is to this single parameter.



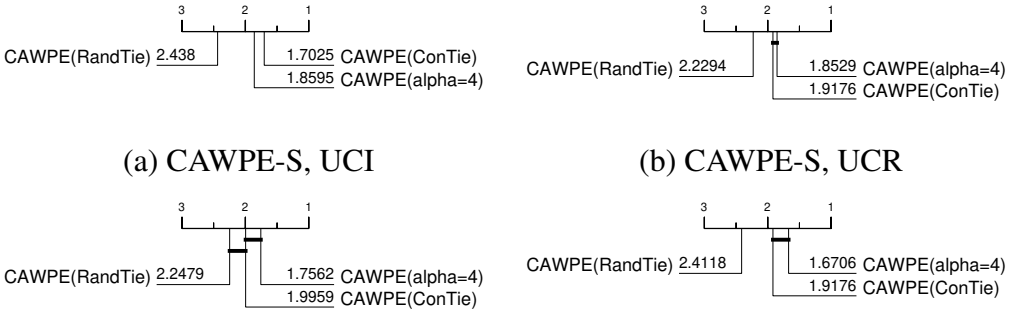
(a) CAWPE-S (b) CAWPE-A

Fig. 4.14 Mean train (squares) and test (triangles) accuracies over the 121 UCI (dashed line) and 85 UCR (solid line) datasets as the alpha parameter changes, expressed as the difference to equal weighting (α=0).

Figure 4.14 depicts what happens if we fix α to progressively higher values over both dataset archives and both base classifier sets used throughout, the basic set (Logistic, C4.5, SVML, NN and MLP1) and the advanced set (RandF, RotF, SVMQ, MLP2 and XGBoost). To keep everything on the same scale and to appropriately highlight the actual differences in accuracy, the average accuracy of each α value is expressed as the difference between itself and using $\alpha = 0$, i.e. no weighting of the base classifiers. Even across the two different archives and base classifier sets, the test performances of different values of α show a fairly consistent pattern, rising steadily until around five to seven before tapering off or eventually falling again. Ultimately as α tends to infinity, we know that the ensemble becomes equivalent to picking the best individual, at which point the line has fallen far below 0 on these graphs. While not included for the sake of space and clarity, the results for the other three test statistics (balanced error, AUC, and NLL) follow an effectively identical pattern.

These results give us an understanding of the surprisingly consistent properties of α overall. However, given some particular set of base classifiers, their relative

performances and ability to estimate their own performance on the training set could vary to different extents depending on the individual dataset provided. As such, the amount that we want to extenuate the differences between the classifier could change from dataset to dataset. It is therefore natural to wonder whether the alpha parameter could be tuned. To do this in a completely fair and unbiased way, we would need to perform a further nested level of cross-validation. However, we can find a much faster (but possibly biased) estimate of the ensemble's error by using exactly the same folds as the base classifiers once more, and simply recombining their predictions. Such a procedure has been shown before to sufficiently sound in practice [136] when performing model selection with nested hyperparameter tuning, and we adopt this here.



(a) CAWPE-S, UCI

(b) CAWPE-S, UCR

(c) CAWPE-A, UCI

(d) CAWPE-A, UCR

Fig. 4.15 Critical difference diagrams over test error of CAWPE on the UCI and UCR archives as it stands (alpha=4), and against two tuning schemes for the alpha parameter: resolving ties in error estimates randomly (RandTie); and conservatively picking the lowest alpha amongst the ties (ConTie).

However, as Figure 4.15 shows, tuning alpha over the range $\{0,1,\ldots 15,\infty\}$ appears to offer little to no benefit when doing so with simple and sensible tuning rules such as picking the α with the best accuracy estimate, and resolving ties (which can be quite common in this scenario) either randomly (RandTie) or conservatively, by choosing the smallest tied value of α (ConTie). ConTie tends towards more evenly averaging the base classifier's outputs, both to counteract any potential overfitting by the base classifiers and, as shown in Figure 4.14, the tendency for

higher values of α to increasingly lead to higher estimates of the ensemble's own performance incorrectly.

One could imagine many more complex tuning schemes potentially having a positive effect, such as sticking to the default value of 4, and only deviating if another value significantly improves accuracy over the cross-validation folds. However, considering both this analysis of α and the findings of the previous section, and remembering our initial guiding principle of simplicity, we believe we can reasonably fall back to fixing the value of α.

4.4.4 Incorporating homogeneity back into CAWPE

CAWPE cross-validates each member on the train data to generate error estimates for weightings. Models rebuilt on the full train data are used to form predictions for the ensemble. During the CV process, however, models are made on each fold which are then discarded. A natural question is whether these can be retained and leveraged to improve predictive performance.

We investigate whether retaining these models, in addition to the models retrained to the full training set, can improve classification performance. We also assess whether accuracy can be maintained while skipping the retraining step on the full data, saving time in the training phase. While maintaining these models incurs no additional training time cost, prediction time and space requirements clearly increase in proportion to the number of CV folds. We further analyse the variance of the maintained classifiers and their effects on the resulting ensemble's variance.

Explicitly building homogeneous (sub-)ensembles from heterogeneous base classifiers is not a new idea. Gashler et al. [51] build forests of trees from different tree building algorithms and shows that larger purely-homogeneous forests can be matched or beaten by smaller mixed forests. Ensemble selection [25] (or pruning) can similarly be applied to purely hetero- or homogeneously generated model sets,

or mixtures of the two [112]. Alongside these works, we specifically wish to study the effects of maintaining homogeneous models, with potentially lower-quality estimates of competency attached, on the CAWPE weighting scheme which relies heavily on the weightings applied.

We evaluate three ensemble configurations that retain the models evaluated on CV folds of the train data against the original CAWPE, which ensembles only over the models retrained on the entire train set. These are to a) (M)aintain all models trained on CV folds and add them to the ensemble alongside the fully trained models (CAWPE_M), b) (M)aintain all models once more, but systematically (D)own-(W)eight them relative to the fully trained models due to their potentially less reliable error estimates (CAWPE_M_DW) c) maintain *only* those models trained on the CV folds, and skip the retraining step on the full train data, (R)eplacing the original models (CAWPE_R).

We take the UCI archive once more to evaluate on, however use a subset of 39 datasets, following feedback received on the superset of these datasets used in the previous study [78]. This set of the datasets are summarised in Table 4.4, and are taken as specifically independent, non-toy, and relatively larger in size. All configurations of CAWPE tested here use the CAWPE-S set of base classifiers. Because all dataset resamples and CV folds of the respective train splits are aligned, each ensemble configuration is therefore being built from identical (meta-)information and we are only testing the configuration's ability to combine the predictions. For reference, we also compare once more these combined homogeneous and heterogeneous CAWPE variants to the homogeneous ensembles RandF and XGBoost, each with 500 trees.

Table 4.4 A full list of the 39 UCI datasets used in these sub-experiments. Full names saved for horizontal space: *[1] conn-bench-sonar-mines-rocks, *[2] conn-bench-vowel-deterding, *[3] vertebral-column-3clases.

Dataset	#Cases	#Atts	#Classes	Dataset	#Cases	#Atts	#Classes
bank	4521	16	2	page-blocks	5473	10	5
blood	748	4	2	parkinsons	195	22	2
breast-cancer-w-diag	569	30	2	pendigits	10992	16	10
breast-tissue	106	9	6	planning	182	12	2
cardio-10clases	2126	21	10	post-operative	90	8	3
sonar-mines-rocks*[1]	208	60	2	ringnorm	7400	20	2
vowel-deterding*[2]	990	11	11	seeds	210	7	3
ecoli	336	7	8	spambase	4601	57	2
glass	214	9	6	statlog-landsat	6435	36	6
hill-valley	1212	100	2	statlog-shuttle	58000	9	7
image-segmentation	2310	18	7	statlog-vehicle	846	18	4
ionosphere	351	33	2	steel-plates	1941	27	7
iris	150	4	3	synthetic-control	600	60	6
libras	360	90	15	twonorm	7400	20	2
magic	19020	10	2	vertebral-column*[3]	310	6	3
miniboone	130064	50	2	wall-following	5456	24	4
oocytes_m_nucleus_4d	1022	41	2	waveform-noise	5000	40	3
oocytes_t_states_5b	912	32	3	wine-quality-white	4898	11	7
optical	5620	62	10	yeast	1484	8	10
ozone	2536	72	2				

4.4.4.1 Results

We summarise comparative results succinctly here in three forms: Figure 4.16 displays CAWPE configurations and reference homogeneous ensembles ordered by average ranks in accuracy along with cliques of significance formed; Table 4.5 details the average scores of all four evaluation metrics; and Table 4.6 details pairwise wins, draws and losses between the original and proposed CAWPE configurations.

Maintaining the individual fold classifiers significantly improves over the original CAWPE. Within the three proposed configurations there is very little difference in performance. This is largely to be expected since they are working from the same meta-information, with the exception of CAWPE_R, which replaces the fully re-trained models only with those trained during CV. This does mean that training time can seemingly be saved by avoiding this final retraining step without a tangible reduction in predictive performance.

Note that while maintaining the fold classifiers improves performance with statistical significance, the average improvement in absolute terms is very small, roughly 0.3% in terms of accuracy, balanced accuracy, and area under the curve

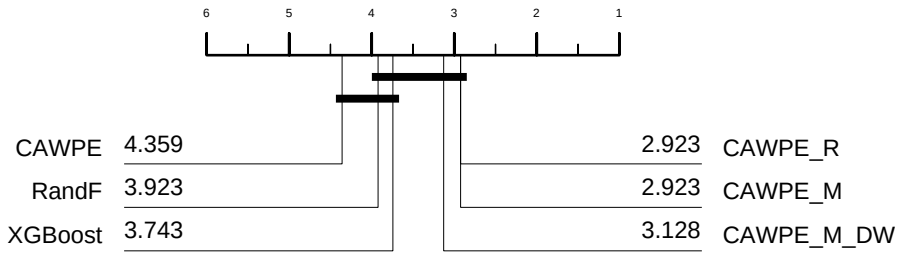


Fig. 4.16 Critical difference diagram displaying the average ranks of accuracy of the original CAWPE and three tested configurations and reference homogeneous ensembles. Classifiers connected by a solid bar are considered within the same clique and not significantly different from each other.

Table 4.5 Averages scores for four evaluation metrics of each of the CAWPE configurations and homogeneous ensembles tested.

Classifier	ACC ↑	BALACC ↑	AUC ↑	NLL ↓
CAWPE	0.861	0.787	0.915	0.53
CAWPE_M_DW	0.864	0.789	0.917	0.517
CAWPE_M	0.865	0.79	0.918	0.515
CAWPE_R	0.865	0.789	0.918	0.516
RandF	0.854	0.78	0.91	0.564
XGBoost	0.85	0.784	0.907	0.647

(Table 4.5). Meanwhile, XGBoost's average accuracy is a full 1.2% lower, but still significantly similar to the new CAWPE configurations. This is because the improvement found while being small, is very consistent (and, conversely, XGBoost is strong but has relatively high variance in performance across datasets). When looking at the paired wins, draws and losses between the configurations in Table 4.6, the contrast between the relatively balanced match-ups of the three new configurations, against the consistently beaten original configuration is clear to see.

Table 4.6 Pairwise wins, draws and losses in terms of dataset accuracies between the ensemble configuration on the row against the configuration on the column.

	CAWPE_R	CAWPE_M	CAWPE_M_DW	CAWPE
CAWPE_R	-	17/4/18	23/0/16	32/0/7
CAWPE_M	18/4/17	-	23/0/16	31/0/8
CAWPE_M_DW	16/0/23	16/0/23	-	34/0/5
CAWPE	7/0/32	8/0/31	5/0/34	-

4.4.4.2 Analysis

CV is such a commonly used method of evaluating a model on a given dataset because of its robustness and completeness relative to, for example, singular held-out validation sets [69]. A single fold of a CV procedure in isolation is of course simply the latter, and equivalent to a single subsample within a bagging context [17]; it is the repeated folding of the data that leads to each instance being predicted as a validation case once that makes the process complete.

All weighted ensembles rely to some extent on the reliability of the error estimates of their members, but CAWPE especially does given that it accentuates the differences in those estimates. We wish to analyse the extent to which the quality of error estimates suffers, and its effects on the ensemble's own performance and variance.

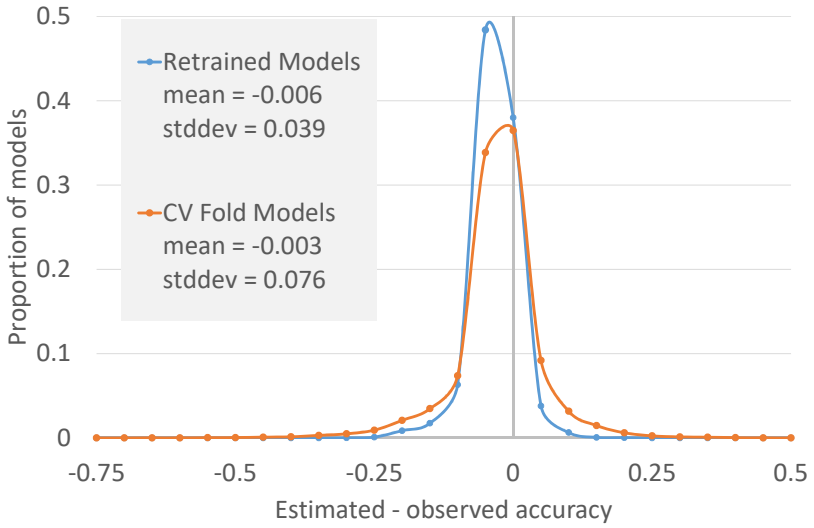
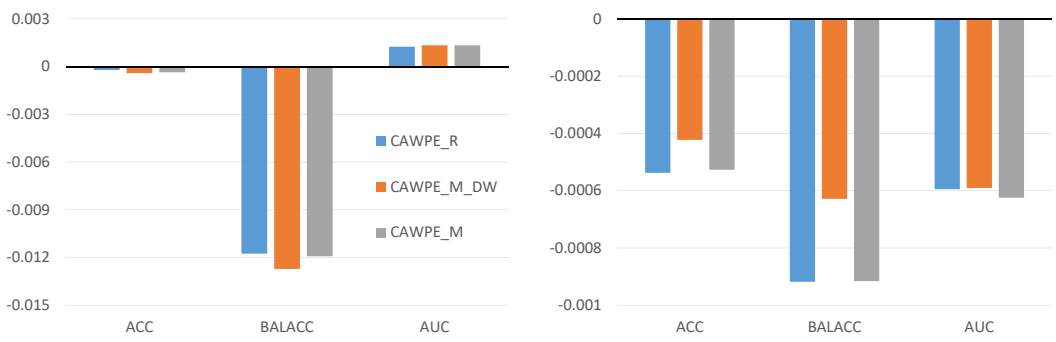


Fig. 4.17 Normalised counts of differences in estimated (on train data) and observed (on test data) accuracy for the retrained (blue) and individual CV fold (orange) models across all datasets and resamples. Positive x values indicate a larger estimated than observed accuracy, i.e. a classifier overestimating its performance.

Figure 4.17 measures the counts of differences in estimated (on train data) and observed (on test data) accuracies and confirms expectations that completing the CV process and retraining models on the full dataset results in more accurate estimates of accuracy on average than the individual models on CV folds. Overall standard deviation almost doubles, but the number and degree of the outliers is perhaps the most important thing. The retrained models never have performance under-estimated by more than 0.3, and less than 2% of the models under estimate by more than 0.1.

Meanwhile, the individuals fold estimates have some extreme outliers in terms of underestimating in particular, with a small tail on Figure 4.17 stretching all the way to -0.75. 7.6% of all fold models underestimate accuracy by more than 10%. Many of the extreme outliers were localised to two datasets, spread out across different learning algorithms. The breast-tissue dataset is a relatively balanced six class problem, while post-operative is a heavily imbalanced three class problem. These factors along with them being the datasets with the least instances likely lead to difficult folds to classify for certain models and seeds, which are of course averaged over when considering the remaining CV folds.

In context, however, the difference really is not too stark. The errors in estimates may double in variance, and these are being accentuated by CAWPE's combination scheme, but there are also fifty more models to average over. Figure 4.18 summarises the differences in variance across test performances between the configurations that maintain the fold models and the original CAWPE along two dimensions - variance in performance on arbitrary datasets, and variance in performance over formulations of the same dataset through resampling. Variance across resamples is reduced, while variance over datasets is less clear. It seems as though cases such as breast-tissue and post-operative affect this particular comparison as with the above, and this shows with variance in balanced accuracy still being clearly reduced.



(a) Standard deviations of performances overs dataset

(b) Average standard deviations in performance over dataset resamples

Fig. 4.18 Standard deviations in performance metrics for the three proposed CAWPE configurations over (a) datasets averaged over resamples and (b) individual dataset resamples, expressed as differences to the original CAWPE. NLL is omitted due to the improper scaling factor brought about by it not being a measure in the range 0 to 1.

When there are only five members, erroneously discounting a classifier to the extent that its outputs are effectively worthless is a large blow to the overall strength of the ensemble. In the case of ensembles with 50 or 55 members though, erroneously discounting one or two classifiers is not so harmful. Practitioners of homogeneous ensembles will of course be familiar with this, and it is the underpinning of the design of such an ensemble - averaging over high variance inputs to produce a low variance output [18].

4.5 Conclusions

In this Chapter, we have developed and evaluated a heterogeneous weighted ensemble scheme, CAWPE. Our initial hypothesis was simple: forming heterogeneous ensembles of approximately equivalent classifiers produces on average a significantly better classifier (in terms of error, ordering and probability estimates) than a wide range of potential base classifiers, and that when we use a weighted probabilis-

tic combination mechanism, ensembles of simple classifier can be at least as good as homogeneous ensembles, heterogeneous ensembles or tuned classifiers. The CAWPE method we propose is significantly better than many equivalent methods and, if the number of classifiers being ensembled is relatively small, represents a sensible starting point. CAWPE is quick, simple and easy to understand. The CAWPE of five simple untuned classifiers is not significantly worse than heavily tuned support vector machines, multilayer perceptron, random forest and XGBoost. CAWPE is significantly better than similar heterogeneous schemes based on predictions rather than probabilities. Clearly, CAWPE is not always the best approach, but given the short time it takes to build the simple classifiers we have used to test it, it seems a sensible starting point.

CAWPE has limitations or areas where it is untested. Firstly, as the train set size increases, the value in ensembling, as opposed to just picking the best, reduces. However, picking best rather than ensembling requires a similar amount of work, and ensembling is unlikely to make things worse. Secondly, with a larger pool of classifiers, it may be better to select a subset rather than use all classifiers using some ES type algorithm. We have not tested this, because unless we choose the overproduce and select methodology of including multiple copies of the same learning algorithm, there are not that many learning algorithms that would be considered equivalent. Our approach is to use fewer very different base classifiers, then combine their output in a way that retains the maximum information. Thirdly, it may well be possible that advanced classifiers such as boosting, deep learning and support vector machines can be designed to beat CAWPE, but if this is the case it is not trivial, as we have shown. Finally, the data we have used has only continuous attributes. We made this decision based on the fact that we wanted to extend previous research and because we come to this problem from TSC, where all data is real valued. It may be that the variation in classifier performance on nominal data is such that the ensembling does not benefit. However, given that

CAWPE is classifier neutral, it seems unlikely that the pattern of results would be much different.

The next stage in this thesis is to evaluate the use of CAWPE, among other algorithms, on our alcohol authentication problem. CAWPE's strong probabilistic predictions, ease of use, and ability to be decomposed and analysed for the relative performances of its base classifiers should all work to its advantage.

Chapter 5

The Prediction of Methanol Content in Genuine Spiked Spirits

Contributing Publications

- Large, J., Bagnall, A., Malinowski, S., and Tavenard, R.(2019a). On time series classification with dictionary-based classifiers. Intelligent Data Analysis, 23(5):1073–1089.
- Middlehurst, M., Large, J., and Bagnall, A. (2020). The canonical interval forest (CIF) classifier for time series classification. arXiv preprint arXiv:2008.09172.
- Middlehurst, M., Large, J., Cawley, G., and Bagnall, A. (2021a). The temporal dictionary ensemble (TDE) classifier for time series classification. arXiv preprint arXiv:2105.03841.
- Middlehurst, M., Large, J., Flynn, M., Lines, J., Bostrom, A., and Bagnall, A. (2021b). Hive-cote 2.0: a new meta ensemble for time series classification. arXiv preprint arXiv:2104.07551.

5.1 Introduction

In Chapter 3, we reported initial investigations on synthesised alcohol solution datasets and brand differentiation. In the former, ethanol concentration was predictable, but methanol concentration was much more difficult. Consuming as little as 10 ml of methanol can cause permanent blindness, while 30 ml can be fatal [132]. Methanol, either by itself or as part of more complex substances such as pectin which are broken down during digestion, occurs naturally in very small amounts in fruit and vegetables. More concentrated methanol can be produced in alcoholic drinks during the fermentation process, particularly when low-quality processes, equipment, and raw inputs are used. Larger scale and more regulated manufacturers will make use of processes to remove any amounts of methanol produced during fermentation. However, producers in less regulated areas or those making it at home (legally or not) are at risk. The detection of methanol to minimal dangerous concentrations with a system such as that put forward by this thesis, where arbitrary spirits in arbitrary containers could be tested, would constitute a large step forward for public health and safety where such analysis can take place prior to consumption. While scenarios such as the identification of particular products are important, they have a more narrow scope of applicability, requiring data collection for each product of interest both now and in the future as different products come to market. It has therefore been identified as a topic that would benefit from more direct work and investigation.

In this chapter, we describe the collection of a dataset of 41 progressively methanol-spiked genuine spirits in their original bottles, spiked from 0.25% to 5% v/v. The lower end (1.75ml in 700ml product) is worthy of seizure and investigation into production processes (especially in commercially sold spirits), but with minimal risk to long term health, while the upper end (35ml in 700ml product) is almost certainly fatal when consumed in large quantities. We show that the stan-

dard chemometric pipeline of normalisation, smoothing, and partial least squares regression cannot form a reasonable fit for predictions. We evaluate the use of more modern classification techniques once more on this data.

Time series classification methods have seen continual and rapid development since the bakeoff discussed in Chapter 2, Section 2.4.2, and the experiments of Chapter 3. The meta-ensemble of time series representations, HIVE-COTE, has always been either solely or among the state of the art for classification performance since its conception, as it and other methods in the literature have improved. Improvements to the overall ensemble have come from advances in individual representations and in their method of combination. We describe the updates to HIVE-COTE since Chapter 3 to form HIVE-COTE 2.0, currently the stand-alone state of the art for TSC as measured on the UCR archive. For individual representation updates, some contributions are in collaboration with other authors, while CAWPE, developed in the previous chapter, is now the ensembling scheme of choice.

We apply the updated TSC methods and CAWPE itself with simple classifiers alongside standard classification techniques and initially find that ten methanol concentrations can be distinguished in arbitrary bottles and spirits with an accuracy of 0.723. The use of stricter collection techniques in particular allows for much clearer signals to be analysed than those in Chapter 3. We investigate this further, looking at the sources of improvement and confounding factors. We ultimately find that using the combined prediction of multiple repeat spectra of a particular sample can improve accuracy to 0.921.

We first outline improvements in TSC algorithms, both contributions solely from this thesis and in collaboration with other authors, in Section 5.2. We introduce and summarise the methanol concentration dataset collected in Section 5.3 and present the results of classification experiments in Section 5.4. We analyse different

aspects of the problem and model performance on it in Section 5.5, and finally conclude in Section 5.6.

5.2 Time series classification improvements

Updates to the state of the art in TSC have been continuous, in part through work of the authors of this thesis. Considerable ground is covered in this section. We catalogue the classifiers used through HIVE-COTE's lifespan for reference in Table 5.1, and illustrate the updates for visual reference in Figure 5.1, which also indicates visually the components and ensembles touch and improved through work towards this thesis.

We summarise in brief here the updates made to the meta-ensemble HIVE-COTE and its constituents, which were made in collaboration with other researchers at the University of East Anglia. The reasons for brevity are due to the main focus of this thesis being on alcohol authentication and the collaborative nature of these TSC-related works. We leave the full experimental details to the respective published works and the future theses of other authors. The HIVE-COTE heterogeneous ensemble of different time series representations, alongside the individual representations themselves, forms the bulk of the time series approaches applied to our methanol concentration dataset. We otherwise employ the state of the art deep learning representative, InceptionTime.

Chapter 2, Section 2.4.2.6 described the original incarnation (alpha), comprised of the TSF, RISE, STC, BOSS, and EE classifiers, with predictions being combined by a majority vote weighted by an estimate of train accuracy. Then, formally defined in early 2020, HIVE-COTE 1.0 dropped EE for efficiency at little to no cost in accuracy and updated BOSS to cBOSS, reducing the total size of and injecting randomness and subsequently diversity into BOSS's hyperparameter search space.

Table 5.1 List of classifiers and acronyms used in summarising the progression of HIVE-COTE and its constituents.

Algorithm Name	Acronym	Source	Representation(s)
Elastic Ensemble	EE	[85]	Whole-series distance
Bag of Symbolic Fourier Approximation Symbols	BOSS	[122]	Dictionary
Spatial BOSS	SBOSS	[77]	Dictionary
Contractable BOSS	cBOSS	[100]	Dictionary
Word Extraction for Time Series Classification	WEASEL	[123]	Dictionary
Temporal Dictionary Ensemble	TDE	[98]	Dictionary
Time Series Forest	TSF	[36]	Interval
Canonical Interval Forest	CIF	[97]	Interval
Diverse Representation CIF	DrCIF	[99]	Interval/Spectral
Shapelet Transform	ST-HESCA	[60]	Shapelets
Shapelet Transform Classifier	STC	[16]	Shapelets
Random Interval Spectral Ensemble	RISE	[87]	Spectral
Random Convolutional Kernel Transform	ROCKET	[34]	Convolutional
Arsenal	-	[99]	Convolutional
Hierarchical Vote Collective of Transformation-based Ensembles (HIVE-COTE)	HC-alpha	[87]	TSF, RISE, ST-HESCA, BOSS, EE Weighted Majority Confidence
	HC1	[4]	TSF, RISE, STC, cBOSS CAWPE
	HC2	[99]	DrCIF, STC, TDE, Arsenal CAWPE

HIVE-COTE 1.0 also adopted CAWPE as its combination scheme. Both CAWPE and its application to HIVE-COTE were described in Chapter 4. We now describe the updates from HIVE-COTE 1.0 to 2.0.

5.2.1 TSF $\rightarrow$ CIF $\rightarrow$ DrCIF

The upgrade from TSF to CIF in 2020 was led by collaborators, with equal (~50%) input on algorithmic and experimental design, and smaller amounts (~30%) of experimental execution and results analysis from myself. CIF expands on TSF, incorporating a larger feature set known as catch22 [92]. catch22 is a set of 22 time series features from the 7658 contained in the highly comparative time-series analysis (hctsa) toolbox [49], selected through a multi-stage process specifically for classification performance on the UCR archive. The features cover a wide range

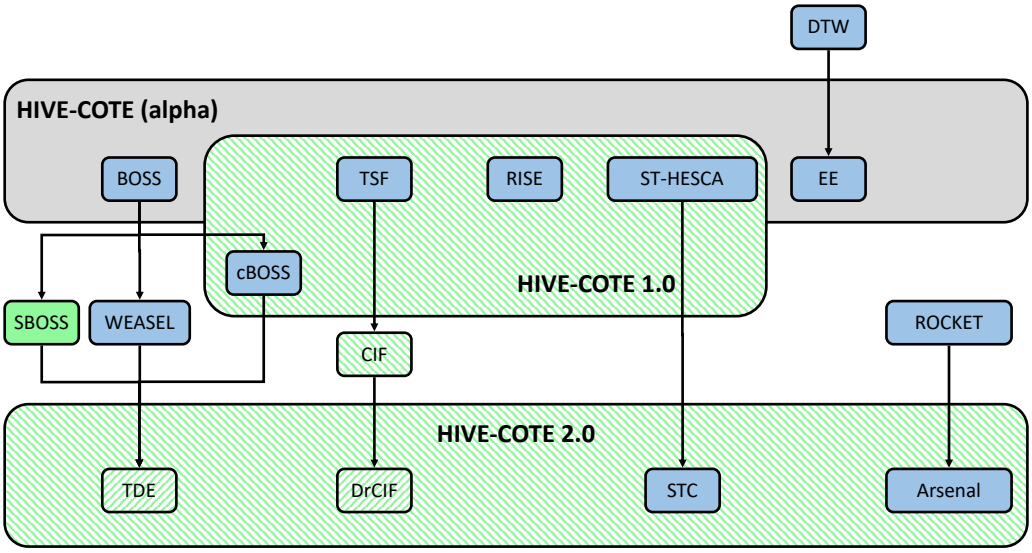


Fig. 5.1 An overview of the updates to individual representations and versions of HIVE-COTE, from alpha to 2.0. Algorithms in solid green (SBOSS) are contributions of this thesis alone. Algorithms in patterned green (HIVE-COTE 1.0/2.0, CIF, DrCIF, TDE) have been developed in collaboration with other authors.

of concepts such as basic statistics of time series values, linear correlations, and entropy. For classification, the obvious way to use catch22 is as a transform prior to building a classifier. However, this method with a decision tree or random forest does not ultimately create a classifier any stronger than DTW [97]. CIF incorporates the catch22 features in the TSF structure, and generates further diversity by sampling a (default 8) of the 25 features (22 plus the mean, slope, and standard deviation from TSF) to use in each tree. k phase dependent intervals with randomly selected positions and lengths are extracted, and summarised by the a features. The tree is then built on the concatenated set of interval summaries, length ka. Middlehurst et al. [97] showed that CIF achieves significantly improved accuracy over TSF on the UCR archive, and when it replaces TSF in HIVE-COTE with no other changes, HIVE-COTE is significantly improved.

CIF was then further enhanced to DrCIF in 2021. This expansion incorporates ideas from the Supervised Time Series Forest (STSF) [24], and aims to incorprate

back into HIVE-COTE 2.0 the spectral information that previously was captured by RISE. The same tree structure is used as in CIF, and intervals randomly selected for use and summary in each tree. However, intervals given to the tree are taken from one of three representations: the interval from the base series; from the first order differences; and intervals from the periodograms of the whole series. The feature pool is expanded further to include four more basic summary statistics: the median; inter-quartile range; min; and max. This takes the candidate pool of features to 29, of which a are still randomly selected for use in each tree. DrCIF significantly improves further over CIF on the UCR archive, and forms the current state of the art interval approach [99].

5.2.2 BOSS $\rightarrow$ SBOSS $\rightarrow$ TDE

Chapter 2, Section 2.4.2.4, laid out how a variety of direct successors to BOSS were spawned - SBOSS, cBOSS, and WEASEL. The current state of the art dictionary representation is TDE, which draws from aspects of all three. SBOSS constitutes my own contribution, while the incorporation into TDE was led by collaborators, with design decisions about the method of incorporation being contributed by me (~20%).

The underlying essence of dictionary classifiers is that they summarise the frequency of patterns any where in the series. SBOSS, originally proposed in 2018, aims to reintroduce temporal location information through the use of spatial pyramids [80], from the field of computer vision. Starting from the initial histogram (bag of words) across the whole series, histograms on subsections are formed by repeatedly dividing the series L times. These histograms are weighted by $\frac{1}{2^{L-l}}$, i.e. weighted inversely proportional to the level l at which they are found. All histograms are then combined and normalised to form a single elongated histogram feature. Because of the weighting, similarity between features found at smaller

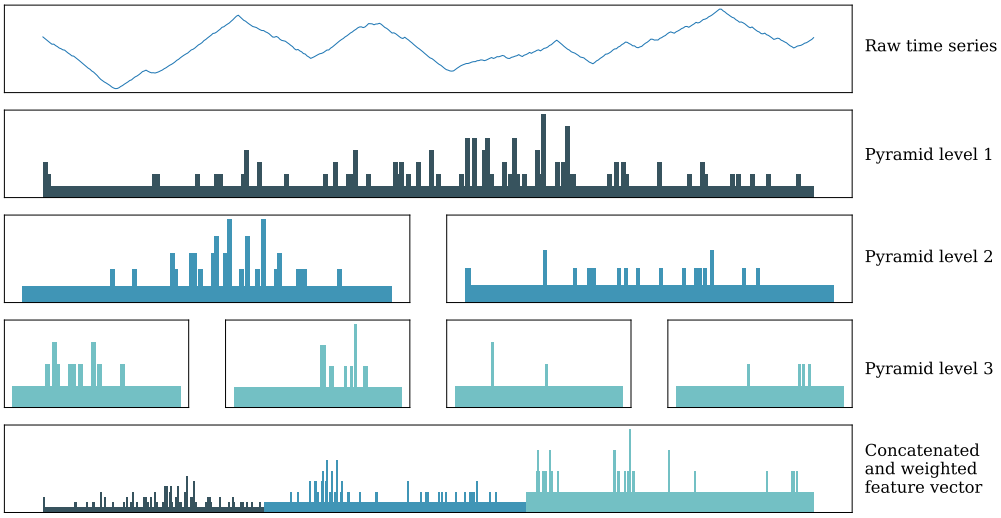


Fig. 5.2 An example transformation of an OSULeaf instance to demonstrate the additional steps to form SBOSS from BOSS. Note that each histogram is represented in a sparse manner; the set of words along the x-axis of each histogram at higher pyramid levels may not be equal.

divisions on the series have a more significant effect than those found on a more global scale, as their temporal location becomes increasingly dissimilar. It is also worth noting that a pyramid with one level is equivalent to the basic bag of words, as no division has occurred. Figure 5.2 illustrates the augmentation. The histogram intersection distance function is also used in place of BOSS's bespoke distance, defined for a histogram of length k as

$$HI(\mathbf{a},\mathbf{b}) = \sum_{i=1}^{k} min(a_i, b_i)$$

SBOSS constitutes a small in absolute accuracy terms, but consistent and significant improvement over BOSS [77].

TDE constitutes a unification of the various improvements and expansions made to BOSS in the literature up to 2020. It takes the spatial pyramid method and use of histogram intersection from SBOSS, the contracting and randomisation of the search space and final ensemble from cBOSS, and the use of bigrams from WEASEL. The increased hyper-parameter search space did make direct porting of

the methods of cBOSS less robust in forming strong ensembles, as a larger space over more dimensions is searched with the same number of evaluations. Because of this, TDE uses a parameter search guided by a Gaussian process. The first 50 (and subsequent) parameter sets and their performances are used to train a Gaussian process regressor [140] which predicts the best performing parameter set out of those still available. Similar to CIF replacing TSF, TDE was found to significantly improve HIVE-COTE when it replaced BOSS and cBOSS [98].

5.2.3 HIVE-COTE 2.0

HIVE-COTE 2.0 contains the strongest individual representations as of 2021: DrCIF; TDE; STC; and Arsenal, with CAWPE as the ensembling mechanism. The structure is summarised by Figure 5.3. For the sake of space on figures and in tables, we refer to HIVE-COTE 2.0 as HC2 henceforth. STC is consistent with the version used in HIVE-COTE 1.0. DrCIF and TDE take the forms described above. HC2 constitutes a further collaborative effort, with contribution from me in algorithm design and experimental design and execution (~20%).

Arsenal is an adaptation of ROCKET for use in HC2, described in Middlehurst et al. [99]. Recall from Chapter 2, Section 2.4.2.8, that ROCKET generates huge feature spaces from the use of randomised convolutional kernels, which it then trains a linear classifier on to make predictions. HIVE-COTE combines weighted probability distributions to form predictions, and so is reliant on the probabilistic output of its base classifiers. The linear classifier (typically ridge regression) used by ROCKET, however, produces essentially one-hot distributions. The CAWPE scheme benefits from its base classifiers producing probability distributions that are already good, beyond generally making them better. The authors of ROCKET found that more complex classifiers such as random forests, which may produce better distributions, did not perform as well on the large and sparsely-important

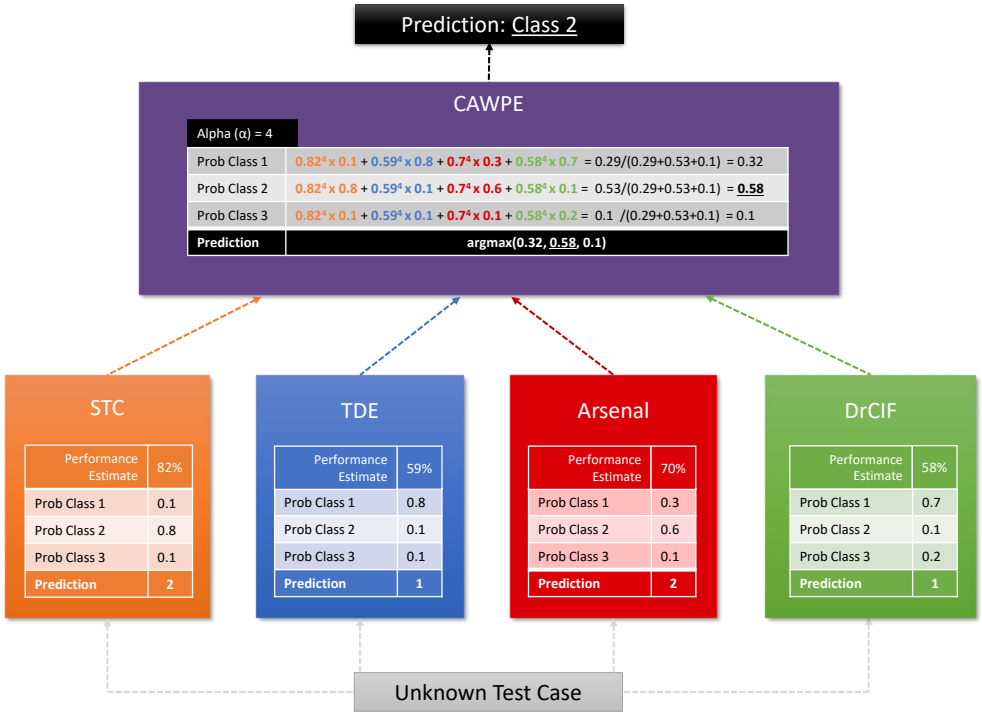


Fig. 5.3 An overview of the ensemble structure of HIVE-COTE 2.0 for a three class problem. Each module is trained independently and produces an estimate of the probability of membership of each class for unseen data. CAWPE combines these probabilities, weighted by an estimate of the quality of the module found on the train data.

feature space as simpler linear classifiers, besides taking much longer to train. For HC2, this is overcome by generating multiple smaller ROCKET transforms and ensembling over them. It was found that while there is very little difference in accuracy between Arsenal and ROCKET, HC2 with Arsenal is significantly better than HC2 with ROCKET, due entirely to the improvements in probabilistic output [99].

Figure 5.4 summarises HC2's accuracy in relation to the previous state of the art, and shows that it significantly improves over all of them. Further analysis and breakdown can be found in the supporting paper [99].

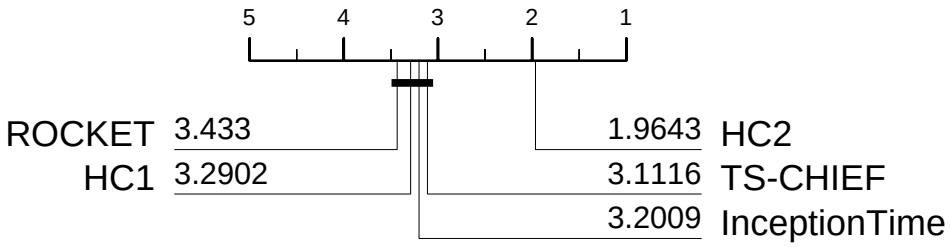


Fig. 5.4 Critical difference diagram for HC2 against the previous state of the art on 112 UCR TSC problems. It demonstrates that there is no difference between HIVE-COTE 1.0 (HC1), InceptionTime, ROCKET and TS-CHIEF, but HC2 is significantly higher ranked than all of them.

5.3 Methanol Concentration Data

Having updated the time series literature to the current state of the art, we now look to apply it to our methanol concentration problem. We first describe the data collected.

Table 5.2 summarises the samples used. To protect producers and their assets, product names are anonymised. Most samples were provided by industry partners via the Scotch Whisky Research Institute, who were given minimal instruction as to the types of samples desired. Samples 33 to 40, however, were acquired separately. Alcohol strengths marked with ^ were 'spiked' with water or ethanol *prior* to any methanol spiking, to provide a wider range of base alcohol strengths against which to try and detect methanol. This was done where we had multiple samples of the same product. Bottle sizes marked with * were samples that had the product moved from their original bottle to a different, unique, bottle. This occurred in cases where the original bottles themselves could not be utilised, either due to being fitted with anti-tampering devices (preventing post-production spiking, as is their intention), or where labelling covered the entire bottle by design, blocking any signal from transmitting through.

We can see that a wide range of spirit types, background alcohol concentrations, bottle sizes, and (although anonymised) bottle types are covered by this dataset. We believe the coverage achieved lends evidence towards the wider general utility of the system for determining methanol concentration under realistic scenarios.

The data collection procedure followed is largely similar to that used throughout Chapter 3. A single BLUE-Wave spectrometer allowing measurements between 339nm and 1174.5nm, with a sampling rate of 0.5nm, was used throughout. Based on domain knowledge of the NIR band of alcohol, we loosely crop to the range 600nm to 950nm in the first instance. Spectra are formed over a total of two seconds; ten readings of 200ms each are averaged.

500ml of each sample was measured out to use as the base 0% methanol. After a 'round' of reading at a methanol concentration, the solution was further spiked with methanol by weight to achieve the desired vol/vol percentage. We spiked to ten targets, 0, 0.25, 0.5, 0.75, 1, 1.5, 2, 3, 4 and 5% methanol. Practically, all spiking degrees and readings of an individual bottle occurred in a single day, and bottles remained sealed outside of weighing and spiking. Alcohol evaporation through exposure to air is not a considerable factor at these timescales. For each sample at each concentration, the sample was once more placed and replaced five times with minimal effort put into recreating the exact signal path, outside of ensuring it avoids labelling and embossing. With 41 bottles, ten spiking quantities, and five repeat placements, we have collected 2050 spectra in total of genuine, spiked, spirits.

Reference spectra were taken prior to each placement, and dark readings after each spiking stage. We present all spectra in this chapter, both in figures and as presented to classifiers, in transmission format following discussion with industry partners. For wavelength m, this is calculated as:

$$transmittance(m) = \frac{sample(m) - dark(m)}{reference(m) - dark(m)} * 100 \tag{5.1}$$

Table 5.2 Summary of the samples used in the genuine spirit methanol concentration dataset. Samples with the same bottle ID are different instantiations of the same product in the same bottle type (including labelling, etc.). Alcohol strengths listed are prior to any methanol being introduced.

Sample ID	Product ID	Spirit Type	Alcohol Strength	Bottle Size (ml)
0	0	Blend	40	750
1	0	Blend	40	750
2	1	Blend	40	700 ->700 *
3	2	Blend	40	450 ->250 *
4	3	Blend	40 ->52.0 ^	500
5	3	Blend	40	500
6	4	Blend	40	700
7	5	Malt	43	750
8	5	Malt	43	750
9	6	Blend	45	700 ->500 *
10	7	Blend	40	700
11	8	Blend	43	750
12	9	Blend	40	1000
13	10	Blend	40	1140
14	11	Malt	45.8	700
15	11	Malt	45.8	700
16	12	Malt	43	700
17	12	Malt	43	700
18	13	Malt	43	700
19	14	Malt	43	700
20	15	Gin	37.5	700
21	16	Gin	37.5	700
22	17	Gin	37.5	700
23	18	Gin	37.5	1000
24	19	Gin	43	1000
25	20	Gin	47.3	750
26	21	Gin	43.1	700
27	22	Gin	40	750
28	23	Vodka	37.5	1000
29	23	Vodka	37.5	750
30	24	Vodka	35	750
31	25	Vodka	40	750
32	26	Vodka	35	375
33	27	Blend	40 ->25.7 ^	700
34	27	Blend	40 ->57.1 ^	700
35	27	Blend	40 ->29.1 ^	700
36	27	Blend	40 ->45.1 ^	700
37	28	Blend	40 ->57.1 ^	700
38	28	Blend	40 ->25.7 ^	700
39	28	Blend	40 ->29.1 ^	700
40	28	Blend	40 ->45.1 ^	700

All spectra are normalised for global intensity once in transmittance format.

Figure 5.5 provides example spectra to illustrate the dataset. Four bottles (chosen randomly, and limited to four for space) are shown on the rows. On the left, random example spectra of each methanol-spiked concentration are plotted. The right plots averaged spectra for no methanol spiking (in blue) and the maximal spiking, 5% (in black). Standard deviations of the spectra of each concentration are also plotted as areas. There are two main points that inspection of these plots tell us.

First, variation by bottle still appears to dominate the overall variance present in the full dataset, consistent with what was found in Chapter 3. Each bottle has a distinct trace that seems easily recognisable and separable from the others. The fourth row is of product 12. The greater effect of within-series variance suggests a lower overall amount of transmitted signal (pre-normalisation), which is caused by it having darker green glass. The use of the leave-one-bottle-out sampling scheme, introduced in Chapter 2, Section 2.5.1.3 and used previously in Chapter 3, shall isolate this factor and again allow us to test for the presence of methanol in a target product regardless of the containing bottle properties.

Second, by eye it is difficult to separate the spectra in a meaningful and consistent way, even when viewing the figures in an interactive manner as opposed to the static view presented here. Considering on the right the most stark and contrasting case we experiment with, 0% and 5% methanol, perhaps the largest separation to be seen are around wavelengths 600-750 for the second bottle (row). Yet, the average spectra look identical in this region for the first bottle.

The consistent use of reference spectra before each prediction, and the use of the transmission mode of spectra presentation, does clearly result in very stable spectra. This does, however, add an additional step to the analysis of new suspect samples,

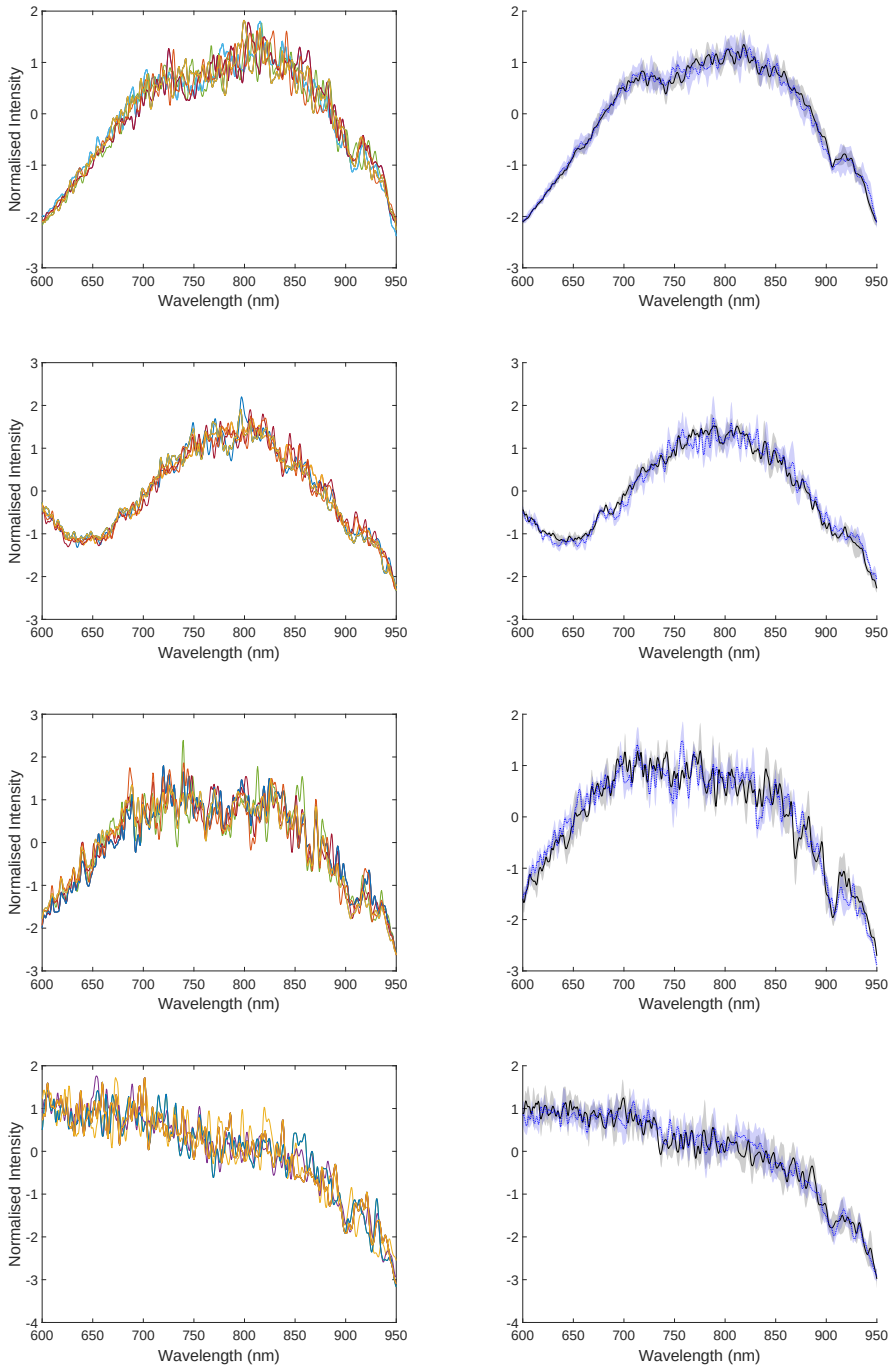


Fig. 5.5 Example spectra from the genuine spirit methanol concentration dataset. On the left, random example spectra of different methanol concentrations are plotted. On the right, average (lines) and standard deviations (areas) of 0% (black) and 5% (blue) methanol are drawn for maximal chance of contrast compared to intermediary concentrations. Each row pertains to a different bottle to display contrast between bottles also.

and increases the total prediction time during field use. We determine the impact of this stability and whether models can reduce the need for it in Section 5.5.4.

As with Chapter 3, a simple statistical summary of the data used for all datasets in this Chapter can be found in Table 6.1 in the Appendix.

5.4 Results

We conduct our main experiments to determine the following:

- With tighter controls on the data production process, are standard chemometric techniques suitable for predicting methanol concentration non-invasively?
- On finding the answer to the previous to be in the negative, can ensemble or TSC techniques make up the difference?

To better account for learning in spite of bottle differences, we use the leave-one-category-out resampling scheme once more throughout all experiments. Because some samples are duplicates of the same product, we adopt a leave-one-product-out (LOPO) sampling such that results are an average over 29 folds with each unique product ID from Table 5.2 being taken in turn to form the test set, with the rest taken for training.

Once the overall results are presented here, we breakdown and analyse different aspects of problem in Section 5.5.

5.4.1 Standard Chemometric Pipelines

We first assess whether the typical chemometric pipeline, initially laid out in Chapter 2, Section 2.3.3, of Savitzky-Golay smoothing followed by partial least squares regression is able to better handle the cleaner data for methanol prediction.

For consistency in data presentation across modelling methods, we achieve intensity correction through normalising spectra, as opposed to e.g. taking the first derivative via the Savitzky-Golay filter.

Given the relative speed of this approach compared to the more complex methods tested next, we perform a large hyperparameter space search over the number of PLS components and SG filter parameters, the window size and polynomial order, to give it the fairest chances. We searched through 20 values of the number of PLS components in 1 through m (followed by a search of values 1 to 10 on seeing that this range is clearly superior for this problem); followed by a grid search of Savitzky-Golay filter parameters of 10 filter window sizes from 5 to 95; and through polynomial orders in the filter of 2, 3 and 5.

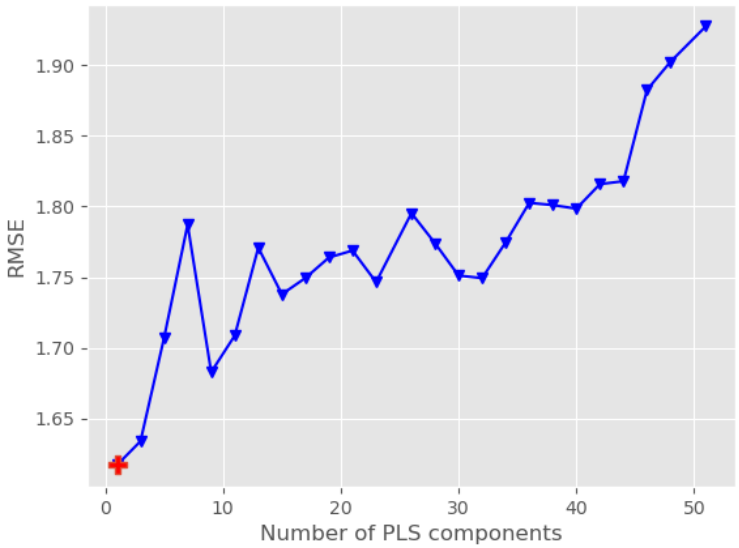


Fig. 5.6 Results of a search over the number of PLS components from one to m for the methanol concentration problem.

To better illustrate the degrees of fit found in the context of the surrounding literature, we phrase the problem back into regression onto methanol concentrations. We first give a brief outline of the parameter search. Figure 5.6 shows that for this problem, one component is best. Then, in short, we find that across all folds of our problem minimal smoothing is actually needed, and that as long as the magnitude

of smoothing is overall small, the performance is fairly robust to these parameters. The modal optimal parameters were a window size of 11, and polynomial order 5, which was only slightly better than no smoothing at all.

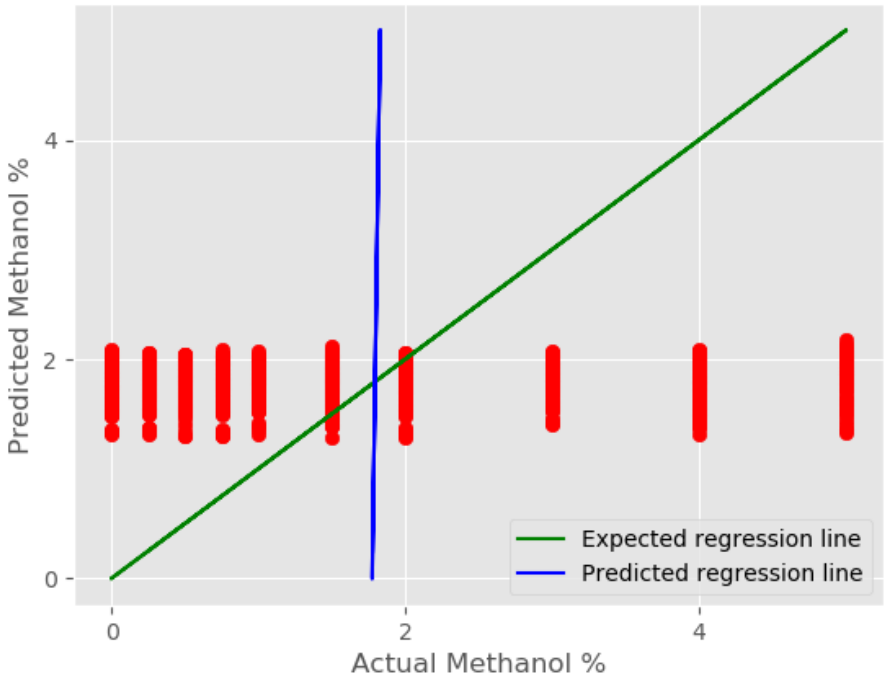


Fig. 5.7 The quality of fit for PLS over all methanol concentration predictions with optimal parameters for each fold. The blue line is observed fit to the predictions (red dots, n=2050), while the green line shows the perfect fit. Essentially no fit is found.

With optimal parameters for each fold of the LOPO-sampled problem, Figure 5.7 shows the average fit over all predictions. We can see that PLS with this comprehensive search simply cannot fit to the data, with an R^2 of 0.007. Phrased back in terms of a ten-class classification problem, the average accuracy when classifying the discretised methanol concentrations is 0.131. We can conclude that the classical chemometric pipeline is insufficient to predict methanol concentrations from the data that we have collected.

5.4.2 Modern Approaches

We now investigate the use of more modern machine learning approaches once more. One of our hypotheses throughout this thesis has been the requirement for non-linear models to account for the distinctly non-linear structural changes to the underlying spirit spectra. In particular, we evaluate TSC approaches to leverage structural changes that can be captured through information contained in the ordering of wavelength values. Another hypothesis was that ensemble approaches, particularly heterogeneous ones, would be able to correct for different problems in the data and average over them to improve robustness and probabilistic output.

Consistent with Chapter 3, we evaluate and compare a quadratic-kernel support vector machine (SVMQ), one nearest neighbour with Euclidean distance (ED), random forest (RandF), and eXtreme gradient boosting (XGBoost). We update HESCA to now be CAWPE-S, and the previous TSC representatives to HC2 and its constituents (DrCIF, TDE, STC, Arsenal). ResNet is updated to InceptionTime, the current state of the art deep learning approach for TSC. CAWPE-S takes the form described in Chapter 4, Section 4.3.5, where the logistic regression classifier was removed for the generally larger UCR data. As there, we have data here with many attributes and many classes. Logistic regression (as implemented in Weka) proved to not be computationally viable for the experiments we ran. The removal of one of five base classifiers makes CAWPE-S weaker in terms of predictive performance. For convenience and space in Figures, we refer to CAWPE-S as described as CAWPE henceforth in this chapter.

Table 5.3 summarises the averaged performances of classifiers on the LOPO-sampled methanol concentration data. CAWPE and ED achieve the best accuracies, 0.723 with a standard deviation of 0.137 and 0.722 with a standard deviation of 0.142 respectively, while CAWPE stands alone as the best for instance ordering and

Table 5.3 Average predictive performances over all 29 folds of the leave-one-bottle-out-sampled methanol concentration dataset. Classifiers grouped by being considered as standard, ensemble, or TSC-bespoke classifiers.

	ACC ↑	AUC ↑	NLL ↓
ED	0.722	0.851*	3.061*
PLS	0.131	0.516	5.340
SVMQ	0.552	0.841	3.453
RandF	0.619	0.881	2.662
CAWPE	**0.723**	**0.925**	**1.581**
XGBoost	0.560	0.869	2.051
InceptionTime	0.449	0.796	3.090
DrCIF	0.594	0.861	2.827
TDE	0.606	0.868	1.973
Arsenal	0.527	0.831	2.288
STC	0.485	0.814	2.640
HC2	0.638	0.897	1.962

probabilistic output, as measured by AUC and NLL. We include PLS results, as originally found in the previous section, for direct comparison.

ED is a one nearest neighbour with Euclidean distance, and so in reality is not capable of producing probabilities estimates. The implementation used in the Weka toolkit employs a Laplace-like correction on initialisation of its probability distributions, preventing larger skews in NLL values (Chapter 2, Section 2.5.2). The AUC and NLL values for ED should not be over-interpreted, but are included for the benefit later comparisons (Section 5.5.3) where distributions for ED make sense.

TSC algorithms do not perform as well as tabular classifiers and ensembles, with STC and Arsenal being the worst of the individual representations. DrCIF and TDE perform relatively better, and the ensemble over all four, HC2, performs better still. HC2 in fact produces the second best probability estimates despite lower accuracy. Overall however, the computational and complexity overhead of the TSC algorithms is not rewarded by improved performance on this data.

In Chapter 3, the presence of methanol was very difficult to predict in a two-class setup, distinguishing 0% methanol as the first class from solutions with 1%, 2% and 5% methanol as the second. In the experiments here, we are predicting methanol concentration directly as a ten-class problem, with finer gradations in concentration. Instead of an accuracy of 0.864 over a minimum expected accuracy of 0.75 (Chapter 3), we now observe accuracies of 0.723 over a minimum expected 0.1. While CAWPE has been advanced in Chapter 4 and performs best here, the fact that ED performs as well as it does shows that a large degree of the difference can be accounted for by the data itself, with possible improvements coming from differences in data preparation, presentation, and quantity. In Chapter 3, three repeat placements were recorded per bottle and per contents, instead of five now. We test the effect of this in Section 5.5.3. The transmission representation, or in particular the collection and use of reference spectra prior to every reading, also has an effect, tested in Section 5.5.4.

Averaged accuracy does not give a full picture of predictive performance. Figure 5.8 shows the summed and normalised over rows confusion matrix for test predictions over all 29 folds of the LOPO-sampled dataset. Because we have essentially an ordinal classification problem, we would expect to see most errors closer to the main diagonal. If a sample has a true concentration of 0.5% but a classifier predicts incorrectly, we would prefer it to predict 0.25% or 0.75% over 5%. On inspection, there is some evidence of this effect happening around true concentrations closer to zero, but overall the errors seem randomly spread. Slightly more errors lie below the diagonal than above (57% of errors below, 43% above), suggesting that CAWPE trends slightly more towards predicting higher concentrations than the true value.

We can inspect the severity of errors by calculating the RMSE of predicted against true methanol concentrations. Because the errors are essentially random in scale, CAWPE's average RMSE is in fact 1.243% methanol. To be clear, this

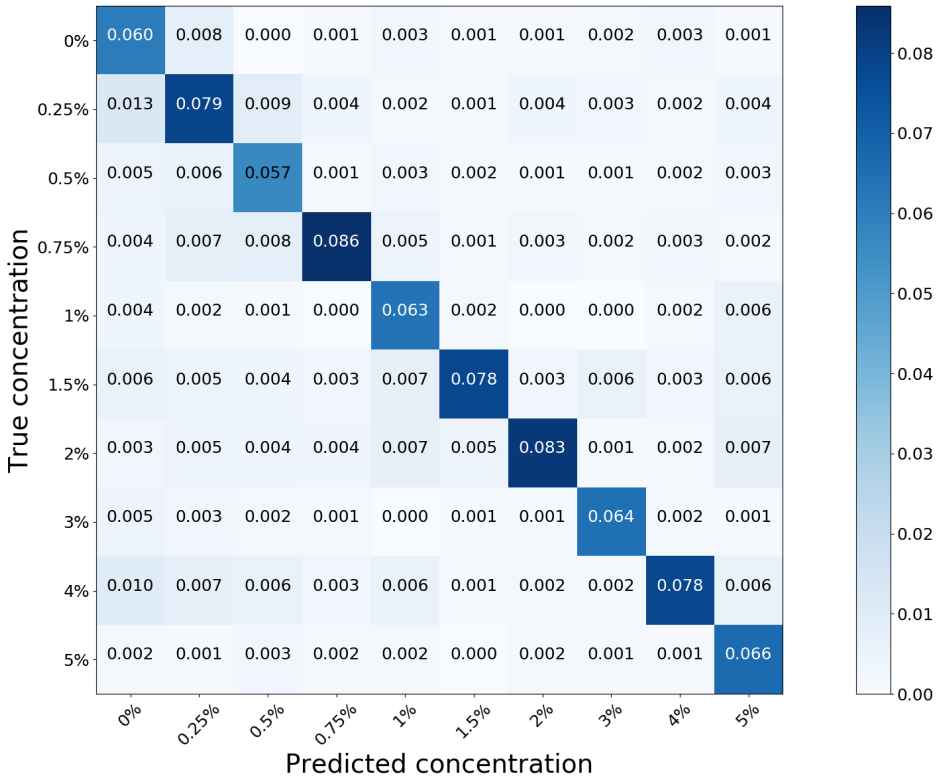


Fig. 5.8 The confusion matrix of CAWPE, summed and normalised over all folds of the LOPO-sampled methanol concentration problem.

would mean that if a sample is predicted to have 2% methanol, the sample could, on average, have between 0.757% and 3.243% methanol, not between plus/minus 1.243% *of* 2%. Phrased in this manner, the ability to correctly determine methanol concentrations using such a setup is shown to be less clear cut than it may initially seem.

5.5 Analysis

We have presented headline results on the methanol concentration problem, where we are predicting one of ten possible concentrations. We now dissect different aspects of the problem and test the limits of detection.

5.5.1 Confounding factors

With the data and results that we have, we can investigate sample properties and their correlations to predictive performance. We identify and plot three properties of interest from the predictions across all folds, summarised by Table 5.2: product ID (29 unique values, aligned with the LOPO sampling folds), spirit type (four categories: Blend, Malt, Gin, Vodka), base alcohol strength (29.1 to 57.1). Bottle size could be an interesting discriminator. However, the real intent behind comparing that would be better suited by measuring over path length, which we do not have access to.

Note that because we sum over all folds, not all predictions are made with entirely identical data. However, the vast majority (27 of 29 products) of the train data is the same fold to fold, and so we take the predictions to be comparable and summarisable in this way.

We first look errors across product IDs, i.e. the bottle shape and notional contents, in Figures 5.9 and 5.10. In other words, these are the test errors of each fold, the average of which is reported in the previous Section. In Figure 5.9, we plot the accuracies of classifiers for each product. This gives us an overview of the relative difficulty of different products regardless of the model, and whether this is consistent. Then in Figure 5.10, we focus on CAWPE's fold accuracies in particular.

We can see that there are definitely some products that are more consistently difficult than others. With the exception of PLS (which is close to random guessing anyway, and is always the lower end outlier in Figure 5.9), the easiest product for all classifiers is product 4. This is a clear glass, cylindrical bottle with clear (albeit narrow) light paths available directly through the centre of the bottle. This makes sense as to why it is easier to obtain a clear signal. However, why it is consistently

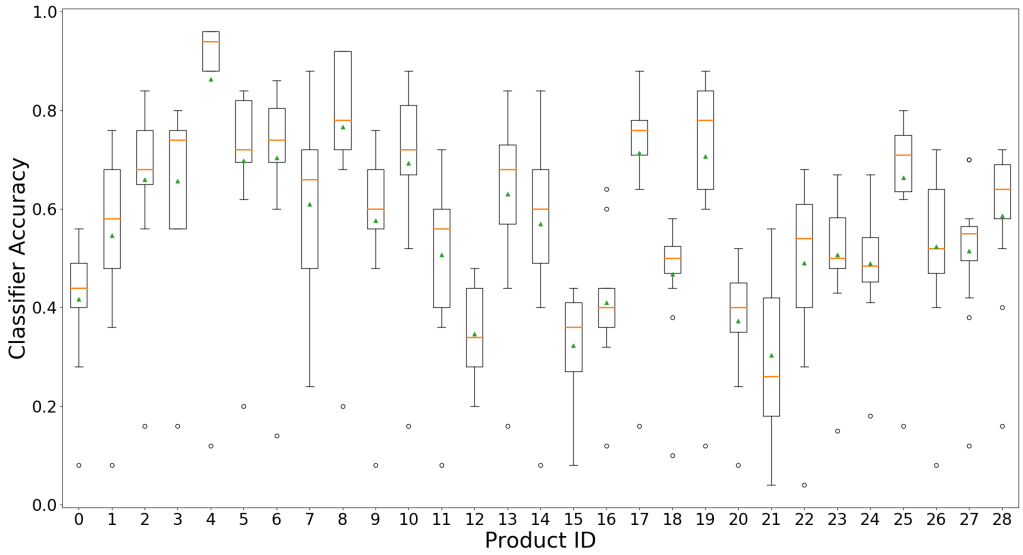


Fig. 5.9 Boxplots of all classifiers' accuracy over different product ID's for the ten-class methanol concentration problem. Orange lines indicate the median, and green triangles indicate the mean.

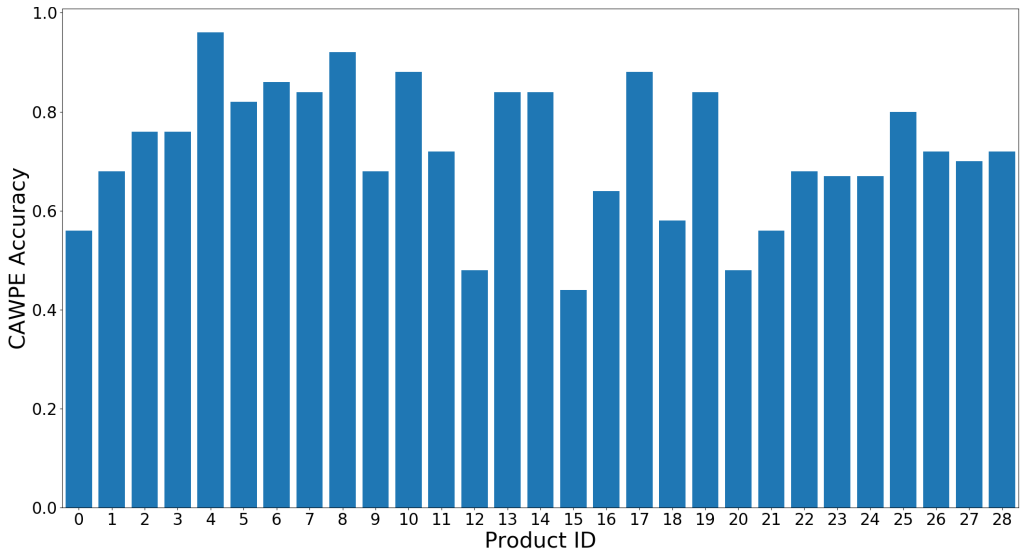


Fig. 5.10 CAWPE's accuracy over different product ID's for the ten-class methanol concentration problem.

easier than other bottles of the same description is not obvious. Products 12, 15, 16, 20 and 21 are the most difficult on average, with 21 being the most so. The sample of product 21 was originally contained in a bottle with an anti-tampering device, and therefore was transferred to an alternative real whisky bottle. The replacement had clear glass and was cylindrical, but had patterned labelling (thin

lines with space in between) covering the majority of the bottle. Finding truly clear paths around these patterns is evidently difficult. Products 12 and 20 have dark green glass lowering the overall signal received. Products 15 and 16 have a non-cylindrical shape (three angled edges, and a curved front to connect). All of these properties making classification more difficult aligns with expectations.

Next, we break down errors between spirit types. The conclusions of this particular piece of analysis should not be overstated. First there is imbalance in the counts of different spirit types. Second, a large amount of variance in difficulty between spirit types could already be captured and explained by the typical designs of bottles in each spirit industry, rather than being explainable by the relative difficulty of detecting methanol in each spirit solution type specifically. Regardless, different typical bottle types between industries is a factor that appears in real markets, and so if one type is more difficult than another due to typical bottle structures, that is still worth noting.

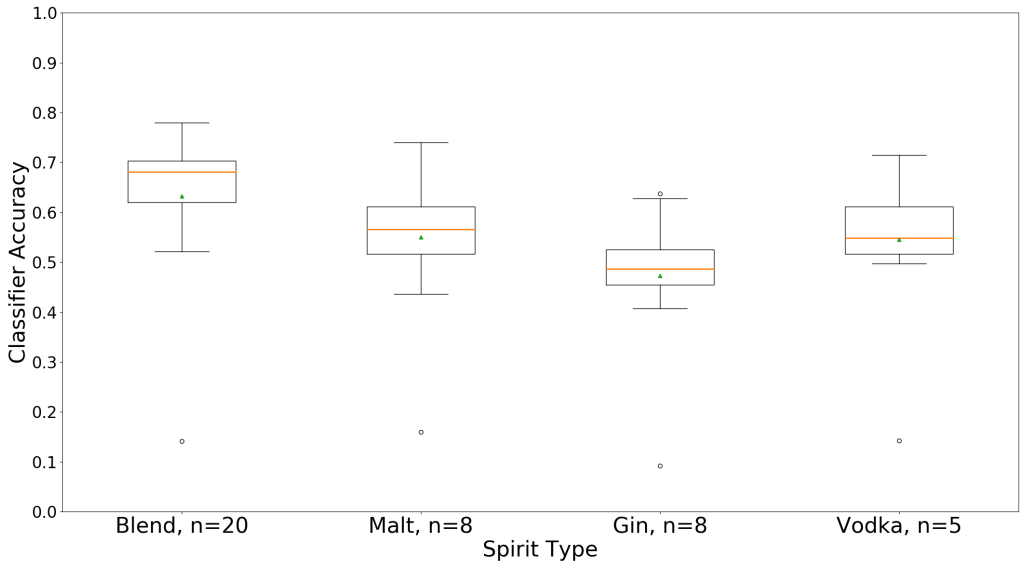


Fig. 5.11 Box plots of all classifiers' average accuracy over spirits grouped and averaged into their spirit type classifications for the ten-class methanol concentration problem. *n* refers to the number of unique samples available for this type (total 41, Table 5.2). Orange lines indicate the median, and green triangles indicate the mean.

Figure 5.11 plots accuracies for classifiers on folds averaged into their spirit category (see Table 5.2). We can see that blended whiskies are generally easier to classify in our data, while gins are the hardest. Notionally, there are no consistent differences in blended whisky bottle design and malt whisky bottle design, at least in our samples, and so the difference in methanol concentration predictability is interesting. The blended whisky category contains the most data (20 samples of the 41 total), and so better performance here makes some sense. However, the fact that this extra data appears to allow classifiers to specialise towards methanol concentration classification in blended whiskies at all suggests that there is a differentiable factor to specialise towards that is separate to malts. Most of the gin samples are of the shape described for product ID 15 and 16 previously; non-cylindrical. This translates to reduced accuracy relative to the other categories.

Lastly, we turn to differences in the ability to classify methanol concentration based on the underlying alcohol (ethanol) strength. Prior to any spiking, we altered some duplicate samples (detailed in Table 5.2 once more) by adding pure ethanol or water to change this base strength, which gives us a wider range of values to work with in this analysis. Alcohols (particularly ethanol and methanol) share overtone bands in the NIR region. Figure 5.12 plots accuracies of CAWPE over the base alcohol strength of the samples. Due to our experimental and results writing mechanisms, we unfortunately present strengths as average strengths per product ID, such that Figure 5.12 has 29 data points. Even with this limitation, however, we can conclude that there is minimal if any correlation between total alcohol strength and the difficulty to classify methanol contents.

A more detailed view of CAWPE's results, giving signed errors in methanol concentration per prediction (as opposed to total accuracy of methanol concentration predicted), is shown in Figure 5.13. Here, we can see that beyond there being no correlation between total errors and base strength, there is equally no correlation between the *direction* of errors and base strength.

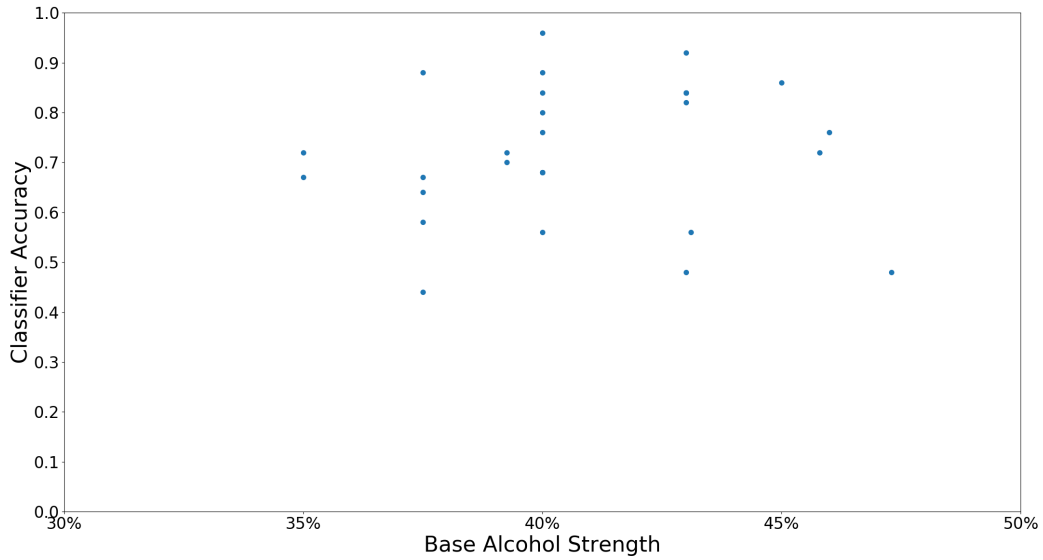


Fig. 5.12 CAWPE's total accuracy over the average base alcohol concentrations of different spirit products.

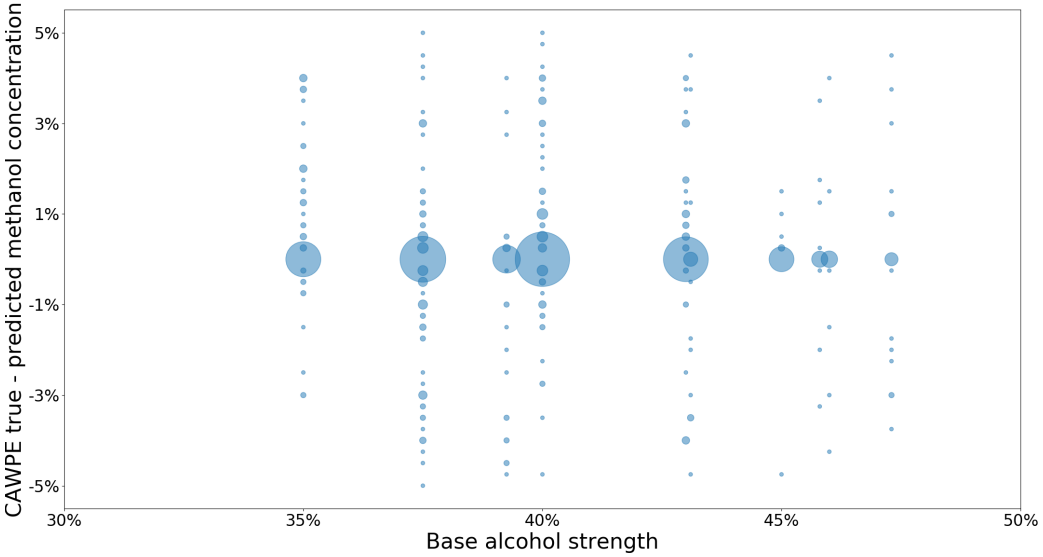


Fig. 5.13 CAWPE's individual prediction (n=2050) errors over the average base alcohol concentrations of different spirit products. The size of bubbles indicates the relative number of predictions at that base alcohol strength and degree of error.

5.5.2 Direct tests for limits of detection

The results above are on a classification problem with ten classes; each methanol concentration. Errors for CAWPE were relatively randomly spaced across the

concentrations. We can get a better idea of the relative difficulty of discriminating successive amounts of methanol concentration by seeing how accuracy changes over two-class formulations of the problem, 0% methanol against x%. This also gives a further indication of what such a proposed system as it stands could reasonably discriminate, should a particular percentage of methanol concentration be identified as being critical and ground for seizure.

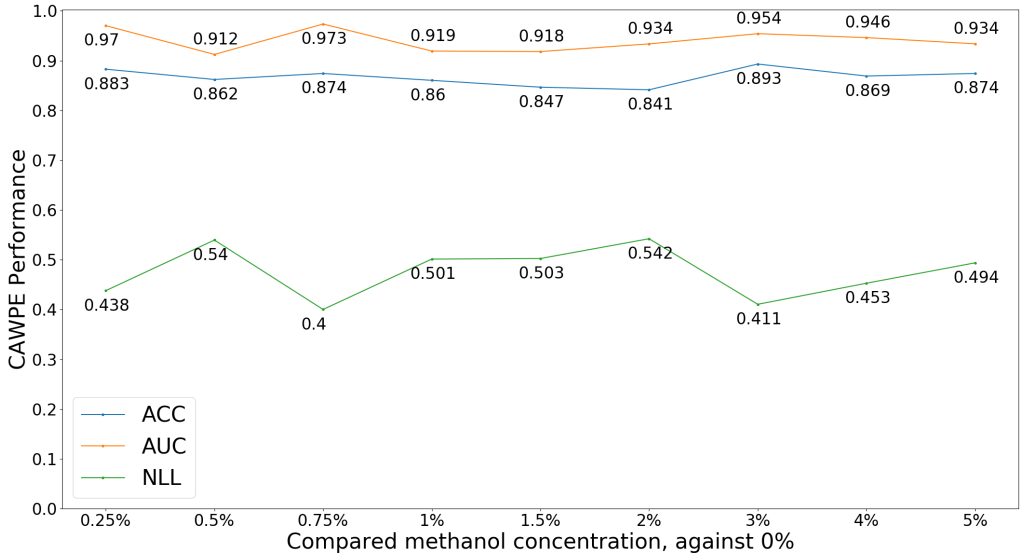


Fig. 5.14 Performance of CAWPE in accuracy, area under the receiver operator curve, and negative log likelihood, in reduced two-class dataset formulations to detect methanol at increasing concentrations.

Figure 5.14 plots the performance of CAWPE on sub-problems of 0% vs the remaining methanol concentrations collected. We would expect the accuracy to increase as the difference in methanol increases, as there should be a progressively larger difference in the underlying spectra.

The first, most obvious conclusion is that no increase in performance is observed as increasingly larger differences in methanol concentration are modelled. Strangely, there is more evidence for the opposite. In a paired t-test, $\alpha = 0.05$, between the fold accuracies of CAWPE on the 0% vs 0.25% (average accuracy 0.883) and 0% vs 5% (average accuracy 0.874) methanol concentration sub-problems, no significant

difference is found. The equivalent tests for AUC and NLL yield the same. We take the slight decrease in *observed* performance to be down to chance, and interpret these results as performance being stable over increasing methanol concentrations. Regardless, we would *expect* performance to have increased, even if the effect was not significant. Taken at face value, what we can say based on these results is that the high-level or summed changes to a sample's spectra as methanol is added may not necessarily be linear in nature. Before drawing such conclusions, more data and experimentation would be needed, however.

The second point to draw from Figure 5.14 is that better predictive performances are achieved on these more focused sub-problems than the ten class variant with all concentrations in one. This has possible implications for dataset structure and modelling methods. We suggested in Chapter 1 that classification of legitimacy or safety is the preferred final output compared to regression onto exact values. These results give some confirmation that generalised detection of the presence of methanol is easier than direct regression.

5.5.3 The utility of repeat placements

We can achieve high predictive accuracy to the precision available to us by our dataset. One avenue to further progress is to see how little data we need to maintain this. Reducing the requirements for data, even if extra data beyond the minimum requirement is always useful, can reduce costs of data collection in future scenarios where new bottle designs, new equipment, etc. are encountered. Further, retraining times (and their own associated compute costs) can be reduced, allowing for more reactive decision making where needed.

One avenue to reduce data requirements is the number of repeat readings taken per bottle and contents, which was five throughout our data collection. These were taken to capture variation within placements, largely a human user factor, and

the scale of its effects on predictions. Are all of these repeats needed to capture variability between placements and the resulting light paths, or can we maintain performance with less?

We repeat the same experimental setup, but progressively removing repeat placements (incidentally, in the order of collection due to the naming schemes of the spectra files) from the train set such that we can compare algorithm performance when trained on all 5 (results above), 4, 3, 2, and only 1 placement. Figure 5.15 plots the degradation in performance for CAWPE as the number of repeats available in the train set decreases.

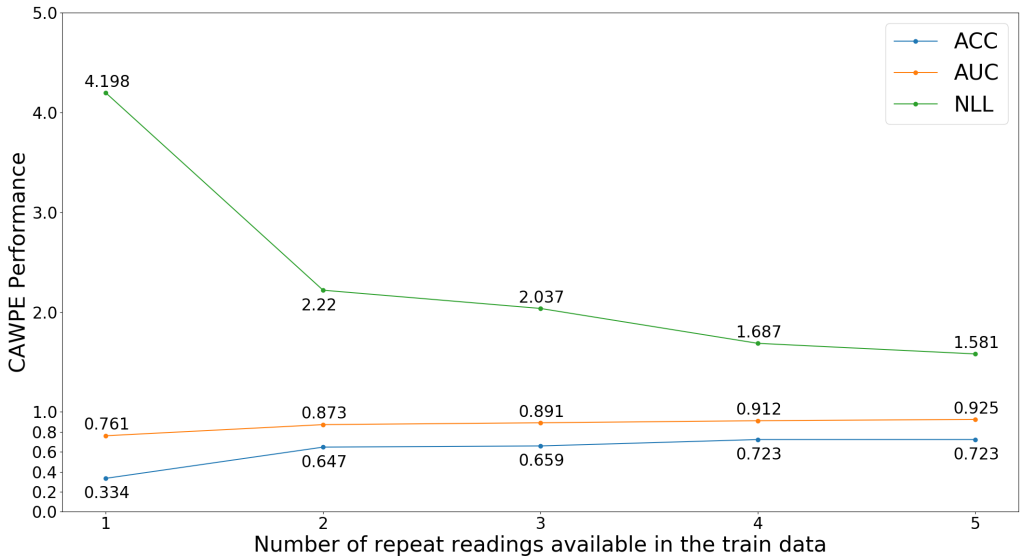


Fig. 5.15 Performance of CAWPE in accuracy, area under the receiver operator curve, and negative log likelihood, as repeat readings are successively removed from the train set.

We observe that classification performance does indeed drop off with less repeats available. This confirms that the additional spectra provide useful information about variance in bottle positioning, or just more examples to learn the true signal from against the random noise. Moving from five to four repeats comes with minimally reduced performance, but more removals come with gradually more cost until only having one reading per sample in particular yields very poor accuracies.

Table 5.4 Classifier performance comparison of individual- and multiple-spectra predictions, formed through averaging over the predictions of repeat placements. Performances under single-spectra predictions are copied over from Table 5.3 for convenience.

	Single-spectra predictions			Multi-spectra predictions		
	ACC ↑	AUC ↑	NLL ↓	ACC ↑	AUC ↑	NLL ↓
CAWPE	0.723	0.925	1.581	0.921	0.989	0.835
ED	0.722	0.851	3.061	0.831	0.957	0.979

In all results presented so far, models have classified individual spectra in isolation. We have now shown the benefits of repeat placements for classifier training, but they could also be useful for improving predictions as well. Given repeat placements of a sample with particular contents, we can effectively turn these into a single multivariate instance to classify. We naively implement this in a post-hoc manner from saved prediction information by averaging the five probability distributions from repeat readings to form a single distribution for a sample that has been read multiple times.

Table 5.4 summarises the differences in performance that occur for CAWPE and ED when we adopt this approach. Performance notably jumps, in particular for CAWPE. We include ED in this comparison to further the point that although the two classifiers had essentially the same accuracy previously, the high-quality probabilistic output of CAWPE meant that it can achieve a larger increase in accuracy here, when averaging over multiple predictions. Products 15 and 21 are still difficult to classify when leveraging multiple spectra, with individual fold accuracies of 0.5 and 0.6 respectively, however, 19 of 29 folds now results in an accuracy of 1. Using this mechanism, we can now say that using the system proposed by this thesis, we can classify methanol concentration in arbitrary spirits and bottles from ten possible values to an accuracy of 0.921, standard deviation 0.137.

5.5.4 The utility of reference spectra

One of the differences in collection practices between the data for this chapter and the alcohol concentration data of Chapter 3 is that reference spectra have now been collected prior to every sample reading, and the spectra subsequently presented in the transmission format.

While the total time to analyse a sample is aimed to be in the seconds (two seconds of actual read time by the hardware, data handling and prediction within a second, plus human handling of the sample) and the recording of reference spectra would only add an extra few, reducing the total work needed and room for human error in a high-throughput scenario is still useful if it can come with no degradation in performance.

We repeat the main experiment, predicting from ten possible methanol concentrations, but present the model with spectra in their raw form with just a dark reading subtracted (referred to as just raw form henceforth). We maintain the use of dark readings as it is assumed that they shall be collected infrequently, and they account for ambient lighting sufficiently well to warrant their collection still.

In short, large decreases in performance are observed. For ED in particular, average accuracy drops from 0.722 to 0.227. ED clearly benefits greatly from the corrections that the use of reference spectra provides, which results in the stability between readings observed in Figure 5.5. The largest decrease in accuracy over a single fold was 0.88, on product 4, which took it from being the easiest product to classify down to random guessing. The smallest decrease was 0.23, and several products could still have their methanol concentrations classified with much higher accuracy than random guessing. Clearly, the use of reference spectra and transmission format benefits products to different degrees. They are always of benefit, however. We can conclude that authentication systems such as those

proposed by this thesis should always collected reference spectra prior to every reading.

5.6 Conclusions

In this chapter we have shown the utility of our proposed non-invasive spirit authentication system for detecting dangerous levels of methanol. In contrast to the inability to determine methanol concentrations in the experiments of Chapter 3, here we have been able to achieve an accuracy of 0.921 on a ten-class methanol concentration prediction dataset.

We have also outlined significant improvements made to the state of the art for general purpose TSC, ultimately culminating in HIVE-COTE 2.0. We hypothesised in Chapter 1 that TSC approaches could leverage order information that is typically ignored by tabular approaches to spectra. However, when HC2 and its constituents are applied to our methanol concentration problem, they perform worse than tabular techniques, despite considerable extra computational resources being required. We can conclude that order information is unimportant to the classification of spectra, likely in large part due to the perfectly aligned nature of the series.

Another hypothesis was that heterogeneous ensembles could correct for defects of different kinds in the data and deliver improved predictive performance overall, but especially so for probabilistic output. CAWPE, introduced in the previous chapter, recorded the highest accuracy by a small margin (a simple one nearest neighbour with Euclidean distance scored only 0.2% less accuracy), but by far better probabilistic performance compared to the rest. Similarly, although overall worse, HC2 still improved over its constituents, and provided the second best probabilistic output despite raw accuracy being poorer.

Regardless of these algorithmic hypotheses and results though, it must be said that the improved ability to predict methanol concentration, relative to Chapter 3, comes largely from improvements in data collection and presentation procedures. These procedures are namely the collection of more repeats per bottle and contents, the collection of reference spectra for each reading, and the presentation of spectra in transmission format. We believe that such practices applied the brand detection problem, and indeed all problems in this space, should lead to similarly stronger results than those exhibited in Chapter 3.

We analysed the results, paying particular attention to CAWPE and ED, being the best classifiers evaluated. We searched for correlations in accuracy to various sample properties, and found strong differences between products and bottles (consistent to prior expectation), tenuous differences between spirit types, and no correlation between the accuracy of methanol concentration predictions and the base alcohol concentration of the sample. Accuracy dropped off heavily with less than four repeats to learn from with our data, suggesting at least that many should be collected for a future spirits spectra database to be robust. Finally, the utility of reference spectra and the transmission format was proven, and should be used in future fielded systems.

Algorithmically, further work on this data would be to fine-tune the ensemble used. We elected to use the pre-defined simple base classifier set used in Chapter 4 for consistency. Automatic and human-guided base classifier selections could improve performance, and would form the basis of our future modelling investigations in this domain.

Chapter 6

Conclusions and Future Work

This thesis has outlined work towards a system that can non-invasively detect fraudulent and/or dangerous spirits. We have described and experimented with a near infrared spectroscopy setup that allows for readings to be taken and analysed non-invasively in seconds, and produced three datasets covering different aspects of spirit authentication. We developed novel general-purpose ensembling and time series classification algorithms, advancing the state of the art, and evaluated them alongside standard chemometric approaches on our new data. Finally we demonstrated that such a system can predict methanol concentration out of ten possible values in arbitrary spirits and bottles with an accuracy of 0.921.

We hypothesised in Chapter 1 that the environmental and sample presentation challenges posed by the system in question, as opposed to standardised lab conditions with direct access to the sample, would result in structural changes and different forms of noise affecting the sample spectra. These changes would not be linear in nature, and therefore would confound the standard chemometric analysis pipeline. This aspect has certainly been shown, with differences in bottle shape being shown repeatedly to be the dominating source of variation in Chapters 3

and 5. In Chapter 5 in particular, is was shown that partial least squares was unable to fit to the methanol concentration data at all.

We further hypothesised that these factors could be overcome by the use of modern classification algorithms. That ensemble algorithms could correct for multiple different sources of error and detect many different underlying features, while time series classification algorithms could leverage an additional feature type; order information. Results indicate that while heterogeneous ensembles of tabular classifiers, represented by CAWPE, did provide improved accuracy and was the best approach evaluated overall, the use of TSC approaches conferred no particular advantage. We conclude as a result of this thesis that tabular (aka vector) modelling approaches are superior to those that specifically leverage order information.

As a result of this thesis, we can suggest that such a system is worthy of further evaluation in a lab scenario. Immediate field use in a law enforcement situation would not be possible; greater efforts are needed to raise the overall reliability, accuracy, and explainability of the analysis pipeline to allow for actionable law enforcement. The system could be used, however, as an indicative method of detecting correct alcohol concentrations in its current state.

6.1 Discussion of Contributions

During this thesis, we have produced three datasets for use in the literature, on the prediction of alcohol contents in alcohol-water solutions, the prediction of particular product in a given bottle, and the prediction of methanol concentration in genuine spirits. Time series classification literature continues to advance, as knowledge about different representations, representation manufacture (through deep learning or ROCKET-like classifiers), and their combinations improve. For TSC and indeed general purpose classification literature to meaningfully continue to progress, the

generation and donation of new datasets for study is an important task. Benchmark datasets and archives need expending and updating to prevent or at least lessen iterative archive-wide optimisation and overfitting. The datasets collected throughout this thesis shall contribute to this process. The dataset EthanolConcentration, derived from experiments of Chapter 3, already appears in the UCR archive of time series classification datasets and has been used as part of general-purpose, multi-domain evaluations across the archive, both in our own works evaluating TSC as documented in Chapter 5, and in the wider community.

We proposed a new ensembling scheme, CAWPE, in Chapter 4. This is composed of cross validation of the base classifiers on the train data to generate an estimate of performance, and accentuating this weighting by raising to a power. Predictions of new cases are formed by combining the tilted class probability distributions of the base classifiers weighted by their accentuated performance estimate. We showed that the addition of the accentuation outperforms alternative ensemble schemes and heavily tuned classifiers across large sets of diverse arbitrary datasets from two different dataset archives. On the UCR time series archive, we showed that incorporating the CAWPE combination scheme into HIVE-COTE makes it significantly better, and advances the state of the art for that field. This is due to the importance of different representations and their correct application to different dataset properties in the TSC space. We performed a series of analyses to better understand CAWPE: an in depth comparison to simple model selection instead of combination, an ablative study of CAWPE's stages, a sensitivity analysis of its parameter, α, and an investigation into expansions of the scheme to include homogeneity through the continued use of the cross validation fold models. We concluded that the best start in a new data domain is to heterogeneously ensemble different classifiers and/or representations instead of focusing attention directly onto one, and that CAWPE is a reasonable place to start until reason is found to attempt more complex stacking or selection schemes.

We described the advancement of the state of the art in time series classification in Chapter 5. Novel to this thesis, this included the improvement of BOSS to SBOSS through the reincorporation of temporal information into the otherwise global dictionary representation. Then, in collaboration with others, we outlined the further incorporation of SBOSS into TDE, the new best dictionary representation, and the improvement of the interval-based TSF classifier into CIF and DrCIF, the new best interval representation. Finally, these were included in the new overall state of the art for time series classification, HIVE-COTE 2.0, comprised of DrCIF, TDE, STC, and Arsenal, with their predictions being ensembled through the use of CAWPE. We showed that the new formulation of HIVE-COTE significantly improves over the previous version and the competing state of the art classifiers. The aspects of HIVE-COTE 2.0 that are entirely not attributed to work undertaken within this thesis are the 'Shapelet Transform' and 'Arsenal' base classifiers of the ensemble.

6.2 Limitations

Throughout this thesis, we have considered and evaluated a wide range of methodologies. However, through discussions with industry partners, the interests of the authors and the research group, and issues of timing and work disruption through COVID19, particular aspects have received arguably disproportionate attention.

First, the end goal has been phrased as a classification problem: authentic versus not. This is what a production-ready system would output in the first instance when faced with an arbitrary sample. We have, however, mainly considered sub-problems building towards this end goal which are more innately ordinal regression. The concentrations of different substances within a sample constitute an important part of authenticity, clearly. The main regression approach we have taken and compared to is a pipeline involving Partial Least Squares Regression. This has been suggested

to us and has been evidenced to be the de facto method for determining substance concentrations from spectra both in the literature and in common chemometrics software. This is particularly the case where the data collection process is highly standardised and separated from confounding factors - which is distinctly not the case for the experiments presented here.

Given this, a larger focus on (more advanced, beyond the industry standard) regression modelling techniques could have been investigated. We instead focused on the classification domain, perhaps prematurely due to the end goal desired, and also due to our inherent expertise and interests.

Similarly, a focus has been placed on Time Series Classification methods, particularly given that the results of Chapter 3 had already suggested a lack of utility within the given spectroscopy domain. Algorithms and results within the general-purpose TSC literature space have been advancing rapidly, in part due to the work done in this thesis. One of our original hypotheses was that TSC methods could correct for high-level distortions brought about by the non-invasive nature of the data collection. While this was not fully realised in the results of Chapter 3, development within the TSC space were deemed worthy of a revisit. This is particularly so with the interval-based classifier advancements of DrCIF and the introduction of the convolutional-based ROCKET classifier, both by themselves and within the context of the improved meta-ensemble HIVE-COTE 2.0.

Lastly, it can be said that Chapters 3 and 5 are very similar in experimental design, and that Chapter 3 should be presented with the methods of Chapter 5, or simply combined. The main reasons for their separation are due to the updated and improved methods of data collection between the datasets of the Chapters, ultimately fueled by the time difference between the dataset collections. The experiments of Chapter 3 represent data collection and presentation procedures and

a set of algorithms that represent our chronologically early investigations into this domain. Chapter 5 represents an improvement in all these aspects.

6.3 Future work

Again, we have worked from the initial principle that classification is the preferred final outcome, based on the final desire for a traffic light system for the aid of final human decision making. When predicting concentrations of substances, we therefore disctretised the class values to reduce what is innately a regression problem, to a classification problem. Another part of the reasoning for this though, is the relative advancement of time series classification techniques in particular as compared to time series regression (TSR) techniques. TSR has seen minimal attention in the literature, while classification has dominated. Instead of reducing the problem to classification through data manipulation, a strand of investigation could be the conversion of existing classification algorithms to the regression task, or the development of new regression algorithms.

For some methods, conversion to the regression task is notionally trivial. Deep learning classifiers architecturally need only replace the final layer. Fully transform-based representational classifiers need only add a regression algorithm to learn on the transformed data, rather than a classifier. However, in some cases such as the shapelet transform, internal processes would need to be updated too. The measure of quality for shapelets would need to be altered for the continuous regression task instead of a binary classification setting.

Besides the use of deep learners for direct prediction, some other deep learning paradigms could be interestingly applied to the alcohol authentication problem. Generative Adversarial Networks [55] could be employed for data augmentation. Learning bottle or environmental properties with Variational Autoencoders [68] to

better learn around them could be fruitful too. There is much potential work in this space.

As covered in Chapter 2, Section 2.3.4, an important consideration when training models for field use in analytical systems is calibration transfer between spectroscopic hardware. Analogous to the use of reference and dark spectra to account for ambient light and the intentional light source's intensity, techniques to account for differences in hardware brought about by production process imperfections are critical if two or more instrument's data are to be used together. Largely for practical reasons of cost, this factor is not present throughout this thesis. Important work before fielding a system would be to incorporate from the literature and ensure the adequacy of calibration transfer techniques.

In terms of the future development of a practical system that can be used in the field, the large-scale future work would be the collection and labelling of a spirits database. In this thesis, we have collected and organised data to form distinct sub-problems to work on. However, for an arbitrary suspect bottle, the way in which the contents are incorrect to the expectation of the label could be for any of these reasons or others not discussed. Models for the different sub-problems that have been discussed in this thesis - correct total alcohol concentration, the presence of methanol, correct brand detection - along with the detection of other means of fraud or adulteration of interest, need to be unified into an overall system. Preferably, one that can a) detect that the contents are incorrect in general with high accuracy and b) give information about the manner in which it is incorrect. The latter would give greater confidence to the initial agent taking a reading to seize the sample, but also make later confirmatory analysis faster and potentially cheaper by immediately targeting analytical efforts.

Our initial route to such a system would be to make models for each sub-problem of major interest, and heterogeneously ensemble them. For a given suspect

sample, models that have learned to predict total alcohol concentration and the presence of methanol may give the green light, but other models for classifying the brand that is on the label may predict that this is a different whisky. In such a scenario, it may then be that a cheaper, but itself legal, whisky has decanted into the bottle of a more expensive one. These models for individual sub-problems can also be ensembled alongside one-class classifiers for particular brands of interest for which a large volume of data is available. Chances are, not all types of abnormality can be modelled (either because they are unknown about, rare to find in fakes, or expensive to reproduce), and so having more generic models that learn the acceptable distributions of a known product can give a greater indication towards a), even if they cannot provide b).

This thesis has proven that a near infrared spectroscopy system with modern machine learning techniques is able to extract useful discriminatory information about a sample non-invasively. The bulk of the future work would be to bring everything together into a practical system, backed by a sizeable database of catalogued and analysed spirits.

Appendix

Table 6.1 Summaries of the alcohol authentication datasets used throughout Chapters 3 and 5. For the PCA-transformed datasets, 95% of the dataset variance is maintained from a transform that is computed *per fold*. As such, the exact number of attributes remaining varies from fold to fold. Generally speaking this is a single digit number, however. LOBO is the leave-one-bottle-out resampling scheme, and LOPO is similarly leaving out one *product*, where multiple bottles contain the same categorical contents. RSR is random stratified resampling, where 70% of the data is taken for training, 30% is reserved for testing.

Dataset	Section	Specific experiment	# Unique Bottles	# Classes	# Attributes	# Instances	Sampling Strategy
Chapter 3 Ethanol Concentration	3.3.2.1	Leave-one-bottle-out Cross Validation	44	3	449	1188	LOBO (44 folds)
	3.3.2.2	Classifying the bottle		3	449	1188	RSR (30 folds)
	3.3.2.3	PCA Transforms		3	95% var*	1188	LOBO (44 folds)
Chapter 3 Methanol Concentration	3.3.2.1	Leave-one-bottle-out Cross Validation	44	4	449	1584	LOBO (44 folds)
	3.3.2.2	Classifying the bottle		4	449	1584	RSR (30 folds)
	3.3.2.3	PCA Transforms		4	95% var*	1584	LOBO (44 folds)
Chapter 3 Brand Authentication	3.4.2.1	Stratified Random Resample	8	2	1101	640	RSR (30 folds)
	3.4.2.2	PCA Transforms		2	95% var*	640	RSR (30 folds)
	3.4.2.3	Testing on different user's data		2	1101	640	RSR (30 folds)
Chapter 5 Methanol Concentration	5.4.1	Standard Chemometric Pipelines	41	10	701	2050	LOPO (29 products)
	5.4.2	Modern Approaches		10	701	2050	LOPO (29 products)

Table 6.2 A full list of the UCI datasets used in Chapter 4.

Dataset	Atts	Classes	Cases	Dataset	Atts	Classes	Cases
abalone	8	3	4177	monks-1	6	2	556
acute-inflammation	6	2	120	monks-2	6	2	601
acute-nephritis	6	2	120	monks-3	6	2	554
adult	14	2	48842	mushroom	21	2	8124
annealing	31	5	898	musk-1	166	2	476
arrhythmia	262	13	452	musk-2	166	2	6598
audiology-std	59	18	196	nursery	8	5	12960
balance-scale	4	3	625	oocytes_m_nucleus_4d	41	2	1022
balloons	4	2	16	oocytes_m_states_2f	25	3	1022
bank	16	2	4521	oocytes_t_nucleus_2f	25	2	912
blood	4	2	748	oocytes_t_states_5b	32	3	912
breast-cancer	9	2	286	optical	62	10	5620
breast-cancer-w	9	2	699	ozone	72	2	2536
breast-cancer-w-diag	30	2	569	page-blocks	10	5	5473
breast-cancer-w-prog	33	2	198	parkinsons	22	2	195
breast-tissue	9	6	106	pendigits	16	10	10992
car	6	4	1728	pima	8	2	768
cardio-10clases	21	10	2126	pit-bri-MATERIAL	7	3	106
cardio-3clases	21	3	2126	pit-bri-REL-L	7	3	103
chess-krvk	6	18	28056	pit-bri-SPAN	7	3	92
chess-krvkp	36	2	3196	pit-bri-T-OR-D	7	2	102
congressional-voting	16	2	435	pit-bridges-TYPE	7	6	105
conn-bench-sonar...	60	2	208	planning	12	2	182
conn-bench-vowel...	11	11	990	plant-margin	64	100	1600
connect-4	42	2	67557	plant-shape	64	100	1600
contrac	9	3	1473	plant-texture	64	100	1599
credit-approval	15	2	690	post-operative	8	3	90
cylinder-bands	35	2	512	primary-tumor	17	15	330
dermatology	34	6	366	ringnorm	20	2	7400
echocardiogram	10	2	131	seeds	7	3	210
ecoli	7	8	336	semeion	256	10	1593
energy-y1	8	3	768	soybean	35	18	683
energy-y2	8	3	768	spambase	57	2	4601
fertility	9	2	100	spect	22	2	265
flags	28	8	194	spectf	44	2	267
glass	9	6	214	statlog-aus-credit	14	2	690
haberman-survival	3	2	306	statlog-ger-credit	24	2	1000
hayes-roth	3	3	160	statlog-heart	13	2	270
heart-cleveland	13	5	303	statlog-image	18	7	2310
heart-hungarian	12	2	294	statlog-landsat	36	6	6435
heart-switzerland	12	5	123	statlog-shuttle	9	7	58000
heart-va	12	5	200	statlog-vehicle	18	4	846
hepatitis	19	2	155	steel-plates	27	7	1941
hill-valley	100	2	1212	synthetic-control	60	6	600
horse-colic	25	2	368	teaching	5	3	151
ilpd-indian-liver	9	2	583	thyroid	21	3	7200
image-segmentation	18	7	2310	tic-tac-toe	9	2	958
ionosphere	33	2	351	titanic	3	2	2201
iris	4	3	150	trains	29	2	10
led-display	7	10	1000	twonorm	20	2	7400
lenses	4	3	24	vert-col-2clases	6	2	310
letter	16	26	20000	vert-col-3clases	6	3	310
libras	90	15	360	wall-following	24	4	5456
low-res-spect	100	9	531	waveform	21	3	5000
lung-cancer	56	3	32	waveform-noise	40	3	5000
lymphography	18	4	148	wine	13	3	178
magic	10	2	19020	wine-quality-red	11	6	1599
mammographic	5	2	961	wine-quality-white	11	7	4898
miniboone	50	2	130064	yeast	8	10	1484
molec-biol-promoter	57	2	106	zoo	16	7	101
molec-biol-splice	60	3	3190				

Table 6.3 The 85 UCR time series classification problems used in the experiments for Chapter 4. Experiments were conducted on 30 stratified resamples of each dataset and all classifiers were aligned on the same folds. Each UCR dataset has an initial default train and test partition that was used for the first experiment, and each subsequent experiment was conducted using resamples of the data that preserve the class distributions and size of the original training and test partitions.

Dataset	Atts	Classes	Train	Test	Dataset	Atts	Classes	Train	Test
Adiac	176	37	390	391	MedicalImages	99	10	381	760
ArrowHead	251	3	36	175	MidPhalOutAgeGroup	80	3	400	154
Beef	470	5	30	30	MidPhalOutCorrect	80	2	600	291
BeetleFly	512	2	20	20	MiddlePhalanxTW	80	6	399	154
BirdChicken	512	2	20	20	MoteStrain	84	2	20	1252
Car	577	4	60	60	NonInvasiveThorax1	750	42	1800	1965
CBF	128	3	30	900	NonInvasiveThorax2	750	42	1800	1965
ChlorineConcentration	166	3	467	3840	OliveOil	570	4	30	30
CinCECGtorso	1639	4	40	1380	OSULeaf	427	6	200	242
Coffee	286	2	28	28	PhalOutCorrect	80	2	1800	858
Computers	720	2	250	250	Phoneme	1024	39	214	1896
CricketX	300	12	390	390	Plane	144	7	105	105
CricketY	300	12	390	390	ProxPhalOutAgeGroup	80	3	400	205
CricketZ	300	12	390	390	ProxPhalOutCorrect	80	2	600	291
DiatomSizeReduction	345	4	16	306	ProximalPhalanxTW	80	6	400	205
DisPhalOutAgeGroup	80	3	400	139	RefrigerationDevices	720	3	375	375
DisPhalOutCor	80	2	600	276	ScreenType	720	3	375	375
DislPhalTW	80	6	400	139	ShapeletSim	500	2	20	180
Earthquakes	512	2	322	139	ShapesAll	512	60	600	600
ECG200	96	2	100	100	SmallKitchApps	720	3	375	375
ECG5000	140	5	500	4500	SonyAIBORSurface1	70	2	20	601
ECGFiveDays	136	2	23	861	SonyAIBORSurface2	65	2	27	953
ElectricDevices	96	7	8926	7711	StarlightCurves	1024	3	1000	8236
FaceAll	131	14	560	1690	Strawberry	235	2	613	370
FaceFour	350	4	24	88	SwedishLeaf	128	15	500	625
FacesUCR	131	14	200	2050	Symbols	398	6	25	995
FiftyWords	270	50	450	455	SyntheticControl	60	6	300	300
Fish	463	7	175	175	ToeSegmentation1	277	2	40	228
FordA	500	2	3601	1320	ToeSegmentation2	343	2	36	130
FordB	500	2	3636	810	Trace	275	4	100	100
GunPoint	150	2	50	150	TwoLeadECG	82	2	23	1139
Ham	431	2	109	105	TwoPatterns	128	4	1000	4000
HandOutlines	2709	2	1000	370	UWaveAll	945	8	896	3582
Haptics	1092	5	155	308	UWaveX	315	8	896	3582
Herring	512	2	64	64	UWaveY	315	8	896	3582
InlineSkate	1882	7	100	550	UWaveZ	315	8	896	3582
InsectWingbeatSound	256	11	220	1980	Wafer	152	2	1000	6164
ItalyPowerDemand	24	2	67	1029	Wine	234	2	57	54
LargeKitchApps	720	3	375	375	WordSynonyms	270	25	267	638
Lightning2	637	2	60	61	Worms	900	5	181	77
Lightning7	319	7	70	73	WormsTwoClass	900	2	181	77
Mallat	1024	8	55	2345	Yoga	426	2	300	3000
Meat	448	3	60	60					

Table 6.4 Raw average scores for error, balanced error, AUC and NLL of the classifiers referenced throughout Section 4.3 of Chapter 4. Scores are averaged over all datasets and resamples of the UCI and UCR archives respectively, except for the tuned classifiers on the UCI archive which had the adult, chess-kvrk, miniboone, and magic datasets removed due to computational restraints.

121 UCI datasets	Classifier	Sections	Error	Balanced Error	AUC	NLL
CAWPE	CAWPE-A	4.1,4.2,4.4	0.174	0.243	0.893	0.651
	CAWPE-S	4.1,4.2,4.3,4.4	0.184	0.258	0.884	0.706
Simple components	C4.5	4.1	0.23	0.301	0.736	1.161
	Logistic	4.1	0.238	0.309	0.841	8.134
	MLP1	4.1	0.213	0.287	0.86	1.297
	NN	4.1	0.216	0.303	0.798	1.116
	SVML	4.1	0.229	0.306	0.849	1.073
Advanced components	MLP2	4.1	0.204	0.276	0.858	1.26
	RandF	4.1,4.3	0.185	0.259	0.886	0.713
	RotF	4.1	0.187	0.265	0.868	0.704
	XGBoost	4.1,4.3	0.193	0.261	0.876	0.843
	SVMQ	4.1	0.216	0.281	0.863	1.454
Heterogeneous Ensembles, simple components	ES-S	4.2	0.19	0.266	0.813	0.884
	MV-S	4.2	0.195	0.273	0.808	0.877
	NBC-S	4.2	0.193	0.26	0.82	0.999
	PB-S	4.2	0.229	0.306	0.847	0.95
	RC-S	4.2	0.195	0.288	0.811	0.912
	SMLR-S	4.2	0.195	0.272	0.737	1.144
	SMLRE-S	4.2	0.214	0.288	0.734	1.251
	SMM5-S	4.2	0.195	0.271	0.744	1.046
	WMV-S	4.2	0.192	0.27	0.814	0.872
Heterogeneous ensembles, advanced components	ES-A	4.2	0.176	0.246	0.817	0.847
	MV-A	4.2	0.176	0.249	0.815	0.833
	NBC-A	4.2	0.183	0.249	0.821	1.031
	PB-A	4.2	0.193	0.261	0.876	0.843
	RC-A	4.2	0.177	0.262	0.813	0.87
	SMLR-A	4.2	0.19	0.263	0.752	1.141
	SMLRE-A	4.2	0.203	0.275	0.747	1.232
	SMM5-A	4.2	0.188	0.261	0.757	1.019
	WMV-A	4.2	0.175	0.248	0.817	0.837
Homogeneous ensembles (RandF and XGBoost repeated)	AdaBoost	4.3	0.353	0.469	0.775	3.258
	Bagging	4.3	0.206	0.303	0.868	0.775
	LogitBoost	4.3	0.241	0.302	0.836	8.246
	RandF	4.1,4.3	0.185	0.259	0.886	0.713
	XGBoost	4.1,4.3	0.193	0.261	0.876	0.843
Tuned Classifiers (on 117 UCI datasets)	TunedMLP	4.4	0.227	0.318	0.857	1.009
	TunedRandF	4.4	0.188	0.271	0.879	0.719
	TunedSVM	4.4	0.188	0.255	0.857	0.955
	TunedXGBoost	4.4	0.194	0.267	0.869	0.86
	CAWPE-T	4.4	0.175	0.244	0.891	0.653

85 UCR datasets	Classifier	Sections	Error	Balanced Error	AUC	NLL
CAWPE	CAWPE-S	4.5	0.241	0.267	0.88	1.071
	CAWPE-A	4.5	0.226	0.254	0.903	0.906
DTW	DTW	4.5	0.224	0.246	-	-
Simple Components	C4.5	4.5	0.36	0.384	0.685	2.168
	Logistic	4.5	-	-	-	-
	MLP1	4.5	0.275	0.301	0.842	3.323
	NN	4.5	0.27	0.301	0.78	1.654
	SVML	4.5	0.312	0.337	0.823	4.733
Advanced components	MLP2	4.5	0.276	0.304	0.858	1.538
	RandF	4.5	0.245	0.28	0.893	1.036
	RotF	4.5	0.251	0.279	0.881	1.019
	SVMQ	4.5	0.276	0.295	0.856	5.755
	XGBoost	4.5	0.267	0.297	0.881	1.156

References

[1] Bagnall, A., Bostrom, A., Cawley, G., Flynn, M., Large, J., and Lines, J. (2018a). Is rotation forest the best classifier for problems with continuous features? *ArXiv e-prints*, arXiv:1809.06705.

[2] Bagnall, A. and Cawley, G. (2017). On the use of default parameter settings in the empirical evaluation of classification algorithms. *ArXiv e-prints*.

[3] Bagnall, A., Dau, H., Lines, J., Flynn, M., Large, J., Bostrom, A., Southam, P., and Keogh, E. (2018b). The UEA multivariate time series classification archive, 2018. *ArXiv e-prints*, arXiv:1811.00075.

[4] Bagnall, A., Flynn, M., Large, J., Lines, J., and Middlehurst, M. (2020). On the usage and performance of HIVE-COTE v1.0. In *proceedings of the 5th Workshop on Advances Analytics and Learning on Temporal Data*, volume 12588 of *Lecture Notes in Artificial Intelligence*.

[5] Bagnall, A. and Janacek, G. (2014). A run length transformation for discriminating between auto regressive time series. *Journal of Classification*, 31:154–178.

[6] Bagnall, A., Lines, J., Bostrom, A., Large, J., and Keogh, E. (2017). The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances. *Data Mining and Knowledge Discovery*, 31(3):606–660.

[7] Bagnall, A., Lines, J., Hills, J., and Bostrom, A. (2015). Time-series classification with COTE: The collective of transformation-based ensembles. *IEEE Transactions on Knowledge and Data Engineering*, 27:2522–2535.

[8] Baydogan, M. and Runger, G. (2016). Time series representation and similarity based on local autopatterns. *Data Mining and Knowledge Discovery*, 30(2):476–509.

[9] Baydogan, M., Runger, G., and Tuv, E. (2013). A bag-of-features framework to classify time series. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(11):2796–2802.

[10] Benavoli, A., Corani, G., and Mangili, F. (2016). Should we really use post-hoc tests based on mean-ranks? *Journal of Machine Learning Research*, 17:1–10.

[11] Bevin, C. J., Dambergs, R. G., Fergusson, A. J., and Cozzolino, D. (2008). Varietal discrimination of Australian wines by means of mid-infrared spectroscopy and multivariate analysis. *Analytica Chimica Acta*, 621(1):19–23.

[12] Bewick, Sharon and Parsons, Richard and Forsythe, Therese and Robinson, Shonna and Dupon, Jean (2020). The Electromagnetic Spectrum. [Online; accessed 2021-06-23].

[13] Bloomfield, M., Andrews, D., Loeffen, P., Tombling, C., York, T., and Matousek, P. (2013). Non-invasive identification of incoming raw pharmaceutical materials using Spatially Offset Raman Spectroscopy. *Journal of Pharmaceutical and Biomedical Analysis*, 76:65–69.

[14] Bloomfield, M., Loeffen, P. W., and Matousek, P. (2010). Detection of concealed substances in sealed opaque plastic and coloured glass containers using SORS. *Proceedings of SPIE*, 7838(June 2016):783808–1 – 783808–15.

[15] Bostrom, A. and Bagnall, A. (2015). Binary shapelet transform for multiclass time series classification. *proceedings of 17th International Conference on Big Data Analytics and Knowledge Discovery*.

[16] Bostrom, A. and Bagnall, A. (2017). Binary shapelet transform for multiclass time series classification. *Transactions on Large-Scale Data and Knowledge Centered Systems*, 32:24–46.

[17] Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

[18] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1):5–32.

[19] Briandet, R., Kemsley, E. K., and Wilson, R. H. (1996). Discrimination of arabica and robusta in instant coffee by fourier transform infrared spectroscopy and chemometrics. *Journal of agricultural and food chemistry*, 44(1):170–174.

[20] Brown, G. and Kuncheva, L. I. (2010). "good" and "bad" diversity in majority vote ensembles. In *International workshop on multiple classifier systems*, pages 124–133. Springer.

[21] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. (2020). Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

[22] Buckley, K. and Matousek, P. (2011). Non-invasive analysis of turbid samples using deep Raman spectroscopy. *The Analyst*, 136(15):3039–50.

[23] Burnett, A. D., Edwards, H. G. M., Hargreaves, M. D., Munshi, T., and Page, K. (2011). A forensic case study: The detection of contraband drugs in carrier solutions by Raman spectroscopy. *Drug Testing and Analysis*, 3(9):539–543.

[24] Cabello, N., Naghizade, E., Qi, J., and Kulik, L. (2020). Fast and accurate time series classification through supervised interval search. In *proceedings of the IEEE International Conference on Data Mining*.

[25] Caruana, R. and Niculescu-Mizil, A. (2004). Ensemble selection from libraries of models. In *Proceedings of the 21st International Conference on Machine learning*.

[26] Chen, T. (2016). XGBoost: A Scalable Tree Boosting System. In *Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[27] Chung, H., Ku, M.-S., and Lee, J.-S. (1999). Comparison of near-infrared and mid-infrared spectroscopy for the determination of distillation property of kerosene. *Vibrational Spectroscopy*, 20(2):155–163.

[28] Corduas, M. and Piccolo, D. (2008). Time series clustering and classification by the autoregressive metric. *Computational Statistics and Data Analysis*, 52(4):1860–1872.

[29] Cozzolino, D. (2015). Sample presentation, sources of error and future perspectives on the application of vibrational spectroscopy in the wine industry. *Journal of the Science of Food and Agriculture*, 95(5):861–868.

[30] Cozzolino, D., Holdstock, M., Dambergs, R. G., Cynkar, W. U., and Smith, P. A. (2009). Mid infrared spectroscopy and multivariate analysis: A tool to discriminate between organic and non-organic wines grown in Australia. *Food Chemistry*, 116(3):761–765.

[31] Cozzolino, D., Kwiatkowski, M. J., Dambergs, R. G., Cynkar, W. U., Janik, L. J., Skouroumounis, G., and Gishen, M. (2008). Analysis of elements in wine using near infrared spectroscopy and partial least squares regression. *Talanta*, 74(4):711–716.

[32] Danezis, G. P., Tsagkaris, A. S., Brusic, V., and Georgiou, C. A. (2016). Food authentication: state of the art and prospects. *Current Opinion in Food Science*, 10:22–31.

[33] Dau, H., Bagnall, A., Kamgar, K., Yeh, M., Zhu, Y., Gharghabi, S., Ratanamahatana, C., Chotirat, A., and Keogh, E. (2019). The UCR time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6):1293–1305.

[34] Dempster, A., Petitjean, F., and Webb, G. (2020). ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels. *Data Mining and Knowledge Discovery*, 34:1454–1495.

[35] Demšar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30.

[36] Deng, H., Runger, G., Tuv, E., and Vladimir, M. (2013). A time series forest for classification and feature extraction. *Information Sciences*, 239:142–153.

[37] Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

[38] Dietterich, T. (2000). An experimental comparison of three methods for constructing ensembles of decision trees: bagging, boosting, and randomization. *Machine learning*, 40(2):139–157.

[39] Džeroski, S. and Ženko, B. (2004). Is combining classifiers with stacking better than selecting the best one? *Machine Learning*, 54(3):255–273.

[40] Eliasson, C., Macleod, N. A., and Matousek, P. (2008). Non-invasive detection of cocaine dissolved in beverages using displaced Raman spectroscopy. *Analytica Chimica Acta*, 607(1):50–53.

[41] Ellis, D., Eccles, R., Xu, Y., Griffen, J., Muhamadali, H., Matousek, P., Goodall, I., and Goodacre, R. (2017). Through-container, extremely low concentration detection of multiple chemical markers of counterfeit alcohol using a handheld sors device. *Scientific reports*, 7(1):12082.

[42] Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., and Muller, P. (2019). Deep learning for time series classification: a review. *Data Mining and Knowledge Discovery*, 33(4):917–963.

[43] Fawaz, H., Lucas, B., Forestier, G., Pelletier, C., Schmidt, D., Weber, J., Webb, G., Idoumghar, L., Muller, P., and Petitjean, F. (2020). InceptionTime: finding AlexNet for time series classification. *Data Mining and Knowledge Discovery*, 34(6):1936–1962.

[44] Fernández-Delgado, M., Cernadas, E., Barro, S., and Amorim, D. (2014). Do we need hundreds of classifiers to solve real world classification problems? *Journal of Machine Learning Research*, 15:3133–3181.

[45] Freund, Y. and Schapire, R. (1996). Experiments with a new boosting algorithm. In *Proc. International Conference on Machine Learning*, volume 96, pages 148–156.

[46] Friedman, J. (2001). Greedy function approximation: a gradient boosting machine. *Annals of Statistics*, pages 1189–1232.

[47] Friedman, J., Hastie, T., and Tibshirani, R. (1998). Additive logistic regression: a statistical view of boosting. Technical report, Stanford University.

[48] Fulcher, B. and Jones, N. (2014). Highly comparative feature-based time-series classification. *IEEE Transactions on Knowledge and Data Engineering*, 26(12):3026–3037.

[49] Fulcher, B. and Jones, N. (2017). hctsa: A computational framework for automated time-series phenotyping using massive feature extraction. *Cell Systems*, 5(5):527–531.

[50] García, S. and Herrera, F. (2008). An extension on "statistical comparisons of classifiers over multiple data sets" for all pairwise comparisons. *Journal of Machine Learning Research*, 9:2677–2694.

[51] Gashler, M., Giraud-Carrier, C., and Martinez, T. (2008). Decision Tree Ensemble: Small Heterogeneous Is Better Than Large Homogeneous. *2008 Seventh International Conference on Machine Learning and Applications*, pages 900–905.

[52] Geurts, P., Ernst, D., and Wehenkel, L. (2006). Extremely randomised trees. *Machine Learning*, 63(1):3–42.

[53] Giacinto, G. and Roli, F. (2001). An Approach to the automatic design of multiple classifier systems. *Pattern Recognition Letters*, 22(1):25–33.

[54] González-Arjona, D., López-Pérez, G., González-Gallero, V., and González, A. (2006). Supervised pattern recognition procedures for discrimination of whiskeys from gas chromatography/mass spectrometry congener analysis. *Journal of agricultural and food chemistry*, 54(6):1982–1989.

[55] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial networks. *arXiv preprint arXiv:1406.2661*.

[56] Grabocka, J. and Schmidt-Thieme, L. (2014). Invariant time-series factorization. *Data Mining and Knowledge Discovery*, 28(5):1455–1479.

[57] Hansen, L. and Salamo, P. (1990). Neural network ensembles. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(10):993–1001.

[58] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*.

[59] Heller, M., Vitali, L., Oliveira, M., Costa, A., and Micke, G. (2011). A rapid sample screening method for authenticity control of whiskey using capillary electrophoresis with online preconcentration. *Journal of agricultural and food chemistry*, 59(13):6882–6888.

[60] Hills, J., Lines, J., Baranauskas, E., Mapp, J., and Bagnall, A. (2014). Classification of time series by shapelet transformation. *Data Mining and Knowledge Discovery*, 28(4):851–881.

[61] Holland, J., Kemsley, E., and Wilson, R. (1998). Use of fourier transform infrared spectroscopy and partial least squares regression for the detection of adulteration of strawberry purees. *Journal of the Science of Food and Agriculture*, 76(2):263–269.

[62] Huang, H., Yu, H., Xu, H., and Ying, Y. (2008). Near infrared spectroscopy for on/in-line monitoring of quality in foods and beverages: A review. *Journal of Food Engineering*, 87(3):303–313.

[63] Izake, E. L. (2010). Forensic and homeland security applications of modern portable Raman spectroscopy. *Forensic Science International*, 202(1-3):1–8.

[64] Izake, E. L., Sundarajoo, S., Olds, W., Cletus, B., Jaatinen, E., and Fredericks, P. M. (2013). Standoff Raman spectrometry for the non-invasive detection of explosives precursors in highly fluorescing packaging. *Talanta*, 103:20–27.

[65] Kate, R. (2016). Using dynamic time warping distances as features for improved time series classification. *Data Mining and Knowledge Discovery*, 30(2):283–312.

[66] Kiefer, J. and Cromwell, A. L. (2017). Analysis of single malt Scotch whisky using Raman spectroscopy. *Anal. Methods*, 91:790–794.

[67] Kim, M., Chung, H., Woo, Y., and Kemper, M. S. (2007). A new non-invasive, quantitative Raman technique for the determination of an active ingredient in pharmaceutical liquids by direct measurement through a plastic bottle. *Analytica Chimica Acta*, 587(2):200–207.

[68] Kingma, D. P. and Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

[69] Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proc. 14th International Joint Conference on Artificial Intelligence*, pages 1137–1143. Morgan Kaufmann Publishers Inc.

[70] Krizhevsky, A., Sutskever, I., and Hinton, G. (2012). Imagenet classification with deep convolutional neural networks. In *proceedings of Advances in Neural Information Processing Systems 25*, pages 1097–1105.

[71] Kuncheva, L. and Rodríguez, J. (2014). A weighted voting framework for classifiers ensembles. *Knowledge and Information Systems*, 38(2):259–275.

[72] Kuncheva, L. and Whitaker, C. (2003). Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning*, 51(2):181–207.

[73] Kuncheva, L. I. (2014). *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons.

[74] L. Didaci, G. F. and Roli, F. (2013). Diversity in classifier ensembles: Fertile concept or dead end? In *International Workshop on Multiple Classifier Systems*, pages 37–48. Springer.

[75] Lachenmeier, D. (2007). Rapid quality control of spirit drinks and beer using multivariate data analysis of fourier transform infrared spectra. *Food Chemistry*, 101(2):825–832.

[76] Lachenmeier, D., Haupt, S., and Schulz, K. (2008). Defining maximum levels of higher alcohols in alcoholic beverages and surrogate alcohol products. *Regulatory Toxicology and Pharmacology*, 50(3):313–321.

[77] Large, J., Bagnall, A., Malinowski, S., and Tavenard, R. (2019a). On time series classification with dictionary-based classifiers. *Intelligent Data Analysis*, 23(5).

[78] Large, J., Lines, J., and Bagnall, A. (2019b). A probabilistic classifier ensemble weighting scheme based on cross validated accuracy estimates. *Data Mining and Knowledge Discovery*, 33(6):1674—1709.

[79] Lawson-Wood, K., Robertson, I., and Seer Green, U. (2017). Comparison of near-and mid-infrared spectroscopy for herb and spice authenticity analysis.

[80] Lazebnik, S., Schmid, C., and Ponce, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 2169–2178.

[81] Lee, S., Choi, H., Cha, K., and Chung, H. (2013). Random forest as a potential multivariate method for near-infrared (NIR) spectroscopic analysis of complex mixture samples: Gasoline and naphtha. *Microchemical Journal*, 110:739–748.

[82] Lee, Y., Kim, J., Lee, S., Woo, Y. A., and Chung, H. (2012). Simple transmission Raman measurements using a single multivariate model for analysis of pharmaceutical samples contained in capsules of different colors. *Talanta*, 89:109–116.

[83] Li, Z., Wang, P.-p., and Huang, C.-c. (2013). Application of Vis / NIR Spectroscopy for Chinese Liquor Discrimination.

[84] Lin, J., Khade, R., and Li, Y. (2012). Rotation-invariant similarity in time series using bag-of-patterns representation. *Journal of Intelligent Information Systems*, 39(2):287–315.

[85] Lines, J. and Bagnall, A. (2015). Time series classification with ensembles of elastic distance measures. *Data Mining and Knowledge Discovery*, 29:565–592.

[86] Lines, J., Davis, L., Hills, J., and Bagnall, A. (2012). A shapelet transform for time series classification. In *proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[87] Lines, J., Taylor, S., and Bagnall, A. (2016). HIVE-COTE: The hierarchical vote collective of transformation-based ensembles for time series classification. In *proceedings of 16th IEEE International Conference on Data Mining*.

[88] Lines, J., Taylor, S., and Bagnall, A. (2018). Time series classification with hive-cote: The hierarchical vote collective of transformation-based ensembles. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 12(5):52.

[89] Littleford, R. E., Matousek, P., Towrie, M., Parker, A. W., Dent, G., Lacey, R. J., and Smith, W. E. (2004). Raman spectroscopy of street samples of cocaine obtained using Kerr gated fluorescence rejection. *Analyst*, 129(6):505–506.

[90] Lohumi, S., Lee, S., Lee, H., and Cho, B. (2015). A review of vibrational spectroscopic techniques for the detection of food authenticity and adulteration. *Trends in Food Science and Technology*, 46(1):85–98.

[91] López-López, M. and García-Ruiz, C. (2014). Infrared and Raman spectroscopy techniques applied to identification of explosives. *TrAC - Trends in Analytical Chemistry*, 54:36–44.

[92] Lubba, C., Sethi, S., Knaute, P., Schultz, S., Fulcher, B., and Jones, N. (2019). catch22: canonical time-series characteristics. *Data Mining and Knowledge Discovery*, 33(6):1821–1852.

[93] Macias, G., Sperling, J., Peveler, W., Burley, G., Neale, S., and Clark, A. (2019). Whisky tasting using a bimetallic nanoplasmonic tongue. *Nanoscale*.

[94] Matousek, P., Morris, M. D., Everall, N., Clark, I. P., Towrie, M., Draper, E., Goodship, A., and Parker, A. W. (2005). Numerical simulations of subsurface probing in diffusely scattering media using spatially offset Raman spectroscopy. *Applied Spectroscopy*, 59(12):1485–1492.

[95] McIntyre, A. C., Bilyk, M. L., Nordon, A., Colquhoun, G., and Littlejohn, D. (2011a). Detection of counterfeit Scotch whisky samples using mid-infrared spectrometry with an attenuated total reflectance probe incorporating polycrystalline silver halide fibres. *Analytica Chimica Acta*, 690(2):228–233.

[96] McIntyre, A. C., Bilyk, M. L., Nordon, A., Colquhoun, G., and Littlejohn, D. (2011b). Detection of counterfeit Scotch whisky samples using mid-infrared spectrometry with an attenuated total reflectance probe incorporating polycrystalline silver halide fibres. *Analytica Chimica Acta*, 690(2):228–233.

[97] Middlehurst, M., Large, J., and Bagnall, A. (2020a). The canonical interval forest (CIF) classifier for time series classification. In *proceedings of the IEEE International Conference on Big Data*, pages 188–195.

[98] Middlehurst, M., Large, J., Cawley, G., and Bagnall, A. (2020b). The temporal dictionary ensemble (TDE) classifier for time series classification. In *proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, volume 12457 of *Lecture Notes in Computer Science*, pages 660–676.

[99] Middlehurst, M., Large, J., Flynn, M., Lines, J., Bostrom, A., and Bagnall, A. (2021). Hive-cote 2.0: a new meta ensemble for time series classification. *arXiv preprint arXiv:2104.07551*.

[100] Middlehurst, M., Vickers, W., and Bagnall, A. (2019). Scalable dictionary classifiers for time series classification. In *proceedings of Intelligent Data Engineering and Automated Learning*, volume 11871 of *Lecture Notes in Computer Science*, pages 11–19.

[101] Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.

[102] Monakhova, Y., Kuballa, T., and Lachenmeier, D. (2012). Nontargeted nmr analysis to rapidly detect hazardous substances in alcoholic beverages. *Applied magnetic resonance*, 42(3):343–352.

[103] Muehlethaler, C., Leona, M., and Lombardi, J. R. (2016). Towards a validation of surface-enhanced Raman scattering (SERS) for use in forensic science: repeatability and reproducibility experiments. *Forensic Science International*, 268:1–13.

[104] Ng, S., Ong, T., Fu, P., and Ching, C. (2002). Enantiomer separation of flavour and fragrance compounds by liquid chromatography using novel urea-covalent bonded methylated β-cyclodextrins on silica. *Journal of Chromatography A*, 968(1-2):31–40.

[105] Ni, W., Nørgaard, L., and Mørup, M. (2014). Non-linear calibration models for near infrared spectroscopy. *Analytica Chimica Acta*, 813:1–14.

[106] Nordon, A., Mills, A., Burn, R., Cusick, F., and Littlejohn, D. (2005). Comparison of non-invasive NIR and Raman spectrometries for determination of alcohol content of spirits. *Analytica Chimica Acta*, 548(1-2):148–158.

[107] Numata, Y., Iida, Y., and Tanaka, H. (2011). Quantitative analysis of alcohol-water binary solutions using Raman spectroscopy. *Journal of Quantitative Spectroscopy and Radiative Transfer*, 112(6):1043–1049.

[108] Numata, Y. and Tanaka, H. (2011). Quantitative analysis of quercetin using Raman spectroscopy. *Food Chemistry*, 126(2):751–755.

[109] Olds, W. J., Jaatinen, E., Fredericks, P., Cletus, B., Panayiotou, H., and Izake, E. L. (2011). Spatially offset Raman spectroscopy (SORS) for the analysis and detection of packaged pharmaceuticals and concealed drugs. *Forensic Science International*, 212(1-3):69–77.

[110] Opitz, D. and Maclin, R. (1999). Popular ensemble methods: an empirical study. *Journal of Artificial Intelligence Research*, 11:169–198.

[111] Parmar, N., Vaswani, A., Uszkoreit, J., Kaiser, L., Shazeer, N., Ku, A., and Tran, D. (2018). Image transformer. In *International Conference on Machine Learning*, pages 4055–4064. PMLR.

[112] Partalas, I., Tsoumakas, G., and Vlahavas, I. (2012). A study on greedy algorithms for ensemble pruning. *Aristotle University of Thessaloniki, Thessaloniki, Greece*.

[113] Partridge, D. and Yates, W. (1996). Engineering multiversion neural-net systems. *Neural Computation*, 8(4):869–93.

[114] Provost, F. and Domingos, P. (2003). Tree induction for probability-based ranking. *Machine Learning*, 52(3):199–215.

[115] Ratanamahatana, C. and Keogh, E. (2005). Three myths about dynamic time warping data mining. In *proceedings of 5th SIAM International Conference on Data Mining*.

[116] Re, M. and Valentini, G. (2011). Ensemble methods: a review. *Data Mining and Machine Learning for Astronomical Applications*.

[117] Realini, M., Botteon, A., Conti, C., Colombo, C., and Matousek, P. (2016). Development of portable defocusing micro-scale spatially offset Raman spectroscopy. *The Analyst*, pages 3012–3019.

[118] Rodriguez, J., Kuncheva, L., and Alonso, C. (2006). Rotation forest: A new classifier ensemble method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(10):1619–1630.

[119] Rokach, L. (2009). Taxonomy for characterizing ensemble methods in classification tasks: A review and annotated bibliography. *Computational statistics & data analysis*, 53(12):4046–4072.

[120] Santos, V. O., Oliveira, F. C. C., Lima, D. G., Petry, A. C., Garcia, E., Suarez, P. A. Z., and Rubim, J. C. (2005). A comparative study of diesel analysis by FTIR, FTNIR and FT-Raman spectroscopy using PLS and artificial neural network analysis. *Analytica Chimica Acta*, 547(2):188–196.

[121] Schäfer, P. (2015). The BOSS is concerned with time series classification in the presence of noise. *Data Mining and Knowledge Discovery*, 29(6):1505–1530.

[122] Schäfer, P. and Högqvist, M. (2012). Sfa: a symbolic fourier approximation and index for similarity search in high dimensional datasets. In *Proceedings of the 15th International Conference on Extending Database Technology*, pages 516–527. ACM.

[123] Schäfer, P. and Leser, U. (2017). Fast and accurate time series classification with WEASEL. In *proceedings of the ACM on Conference on Information and Knowledge Management*, pages 637–646.

[124] Scott, I. M., Lin, W., Liakata, M., Wood, J. E., Vermeer, C. P., Allaway, D., Ward, J. L., Draper, J., Beale, M. H., Corol, D. I., Baker, J. M., and King, R. D. (2013). Merits of random forests emerge in evaluation of chemometric classifiers by external validation. *Analytica Chimica Acta*, 801:22–33.

[125] Senin, P. and Malinchik, S. (2013). SAX-VSM: interpretable time series classification using sax and vector space model. In *proceedings of 13th IEEE International Conference on Data Mining*.

[126] Shifaz, A., Pelletier, C., Petitjean, F., and Webb, G. (2020). TS-CHIEF: A scalable and accurate forest algorithm for time series classification. *Data Mining and Knowledge Discovery*, pages 1–34.

[127] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *proceeding of the IEEE Conference on Computer Vision and Pattern Recognition*.

[128] Tang, K., Suganthan, P., and Yao, X. (2006). An analysis of diversity measures. *Machine Learning*, 65(1):247–271.

[129] Tapp, H. S., Defernez, M., and Kemsley, E. K. (2003). Ftir spectroscopy and multivariate analysis can distinguish the geographic origin of extra virgin olive oils. *Journal of agricultural and food chemistry*, 51(21):6110–6115.

[130] Ting, K. and Witten, I. (1997). Stacking bagged and dagged models. In *Proc. 14th International Conference on Machine Learning*, pages 367–375.

[131] Ting, K. and Witten, I. (1999). Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 10:271–289.

[132] Tulashie, S. K., Appiah, A. P., Torku, G. D., Darko, A. Y., and Wiredu, A. (2017). Determination of methanol and ethanol concentrations in local and foreign alcoholic drinks and food products (banku, ga kenkey, fante kenkey and hausa koko) in ghana. *International Journal of food contamination*, 4(1):1–5.

[133] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. u., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

[134] Wainberg, M., Alipanahi, B., and Frey., B. (2016). Are random forests truly the best classifiers? *Journal of Machine Learning Research*, 17:1–5.

[135] Wainer, J. and Cawley, G. (2017). Empirical evaluation of resampling procedures for optimising svm hyperparameters. *Journal of Machine Learning Research*, 18(15):1–35.

[136] Wainer, J. and Cawley, G. (2021). Nested cross-validation when selecting classifiers is overzealous for most practical applications. *Expert Systems with Applications*, 182:115222.

[137] Wang, Z., Yan, W., and Oates, T. (2017). Time series classification from scratch with deep neural networks: a strong baseline. In *proceedings of the International Joint Conference on Neural Networks*, pages 1578–1585.

[138] Webb, G. (2000). Multiboosting: A technique for combining boosting and wagging. *Machine Learning*, 40(2):159–196.

[139] Wiesner, K., Fuchs, K., Gigler, A. M., and Pastusiak, R. (2014). Trends in near infrared spectroscopy and multivariate data analysis from an industrial perspective. *Procedia Engineering*, 87:867–870.

[140] Williams, C. K. and Rasmussen, C. E. (2006). *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA.

[141] Wolpert, H. (1992). Stacked Generalization. *Neural Networks*, 3(2):241–259.

[142] Wu, Z., Xu, E., Long, J., Zhang, Y., Wang, F., Xu, X., Jin, Z., and Jiao, A. (2015). Monitoring of fermentation process parameters of Chinese rice wine using attenuated total reflectance mid-infrared spectroscopy.

[143] Ye, L. and Keogh, E. (2011). Time series shapelets: a novel technique that allows accurate, interpretable and fast classification. *Data Mining and Knowledge Discovery*, 22(1-2):149–182.

[144] Yun, C., Bhojanapalli, S., Rawat, A. S., Reddi, S. J., and Kumar, S. (2019). Are transformers universal approximators of sequence-to-sequence functions? *arXiv preprint arXiv:1912.10077*.

[145] Zhang, X., Lin, T., Xu, J., Luo, X., and Ying, Y. (2019). Deepspectra: An end-to-end deep learning approach for quantitative spectral analysis. *Analytica Chimica Acta*, 1058:48–57.