MACHINE LEARNING ANALYSIS ON FISH OIL DATA

Jesse Wood, Bing-Xue, Mengjie Zhang, Bach Hoai Nguyen, Daniel Killeen

Victoria University
Engineering and Computer Science
Kelburn, Wellington, New Zealand

ABSTRACT

Gas chromatography (GC) can be used to identify chemical compounds present within tissue samples for quality assurance in food science. Existing analytical chemistry techniques for processing GC data are manual and time-consuming. Here, we explore classification algorithms for fish oil data that automate and significantly reduce the time required to process GC data. We find the Linear SVC model can predict the fish species with near-perfect accuracy. The fish oil data is high-dimensional and low sample size. We compare state-of-the-art feature selection methods to reduce the dimensionality of the data. High accuracy is possible with very few features for the MRMR and ReliefF feature selection methods. Visualisation is used to explore the interpretability of the models such that their efficacy can be verified for use in a factory setting. The exploration reveals there are many feature subsets all capable of producing high-accuracy predictions. No clear superset of important features emerges, which indicates there are many important features to choose from. Submitted in partial fulfilment of the Directed Individual Study.

***Index Terms*—** Feature Selection, Gas Chromatography, Support Vector Machines, Visualisation

1. INTRODUCTION

Identifying fish by their species and part from tissue samples is a common task in food science. In a factory setting, this is required for quality assurance, optimizing production lines and maximising profits. Gas chromatography (GC) [1] is a method that can identify chemicial structures in these fish oils [2]. This produces high-dimensional low sample size data from the fish oils. Chemists compare a given sample to a reference sample to determine what chemicals are present. The existing analytical techniques to perform these tasks are time-consuming and laborious. Previous work has shown it possible to use machine learning to automate very similar tasks [3, 4].

Thanks to New Zealand Plant & Food Research for datasets, funding and expertise.

In this paper, we explore machine learning techniques to automate the process of identifying fish species/parts from GC data. Firstly, classification algorithms are evaluated for their ability to determine the fish species and part. Secondly, feature selection is used to reduce the dimensionality of the data, whilst maintaining high-accuracy predictions. Finally, visualisation is used to explore the interpretability of successful models.

Specifically, our work is divided into main sections:

1. Classification
2. Feature Selection
3. Visualisation

2. BACKGROUND

This paper is a multi-disciplinary effort. Domain expertise in chemistry and machine learning is required to extract the full value from the fish oil data. Before we explore the data, we provide context on the knowledge - from both chemistry and machine learning - needed to understand this paper.

Specifically, the background covers:

1. Gas Chromatography
2. Classification
3. Feature Selection
4. Visualisation

2.1. Gas Chromatography

Gas chromatography (GC) is a technique for the analysis of chemical compounds [1, 5, 2]. The process separates compounds based on their boiling point and molecular weight. A compound is injected as a liquid, it is then vaporized into a gas by applying heat. A boiling point is a temperature at which it changes phase, from liquid to gas. This process is referred to as a phase transition. The speed at which a compound is

vaporized depends on its boiling point. The vaporized gases then travel through a long coil tube. That tube has a detector at the end, this detects the rate and intensity at which compounds reach the tube's end.

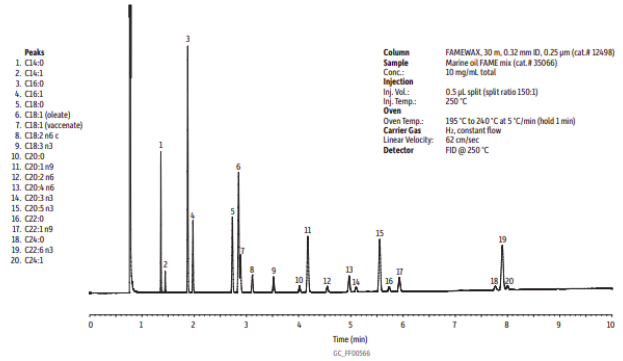


Fig. 1: Gas chromatograph: the artifact of the GC method [2]. The detection is used to visualize intensity (y) and time (x) on a chromotrogaph.

Chemists use the chromatograms of known compounds as a reference when classifying new ones. Take a known example, say *methyl eladiate*. We compare this reference sample to an unknown sample. Analysis can infer the unknown compound since they share the same peaks as the known one. GC is not a definitive technique [5], so it is often used in conjunction with other techniques. Mass spectrometry is one such technique [2].

The existing task of classifying chemical compounds based on a chromatograph is laborious [1, 2]. The spikes on the graph represent peaks. Each peak represents a resolved chemical compound. Chemists integrate the area under each peak, and compare this to a reference sample, to classify the compound. GC must be performed slowly to ensure that the peaks are not too broad. This ensures each peak resolves and represents a single compound. Once we know what compounds are present in a sample it becomes possible to identify what the sample is. For this fish oil data, we classify a sample into two categories:

1. Species
2. Part

Using machine learning techniques it may be possible to speed the process up. Machine learning algorithms identify patterns in the data. These patterns can be used to classify the sample efficiently. An interpretable and accurate model has the potential to be deployed in a factory setting. It would eliminate the need for manual work. Additionally, an algorithm that can classify unresolved peaks would have an impact on the chemistry field. This increases the speed at which GC is performed, increasing the volumetric efficiency of the production line [6].

2.2. Classification

The classification task is to identify the class label for an instance from the dataset. We perform two classification tasks and measure their performance:

1. Species: Identify the species of fish (i.e. Bluecod).
2. Part: Identify the part of the fish (i.e. Head).

A supervised learning method creates a model from a labelled dataset - the train set. We measure the ability of the model to generalize on unseen data - the test set. We give the model a tissue sample from a fish. Based on what it has learnt, it predicts the species and part for that fish. The existing analytical chemistry techniques for performing this task are laborious and time-consuming. We desire a model that is interpretable and has high predictive accuracy. We can then use the model to automate this task. Thus, it has real-world applications for quality assurance in a factory setting.

2.2.1. Support Vector Machines

Cortes and Vapnik proposed the Support Vector Machine (SVM) [7]. This model creates a hyperplane that can draw distinct class boundaries between classes. We call these class boundaries the support vectors. We are performing multi-class classification, so it used a one-vs-all approach [8]. This creates a divide between one class and the rest, then repeats for the other classes.

2.2.2. Model

The sklearn library provides several SVM models for classification. The default model uses the RBF kernel. Other models use different kernels and parameters [8]. Some models remove trainable parameters. Instead, the user can set the number of support vectors [9].

2.2.3. Kernel

The model requires a kernel function. This determines the shape of the support vectors in the hyperplane. Different kernels capture data of varying complexities. The original hyperplane algorithm used a linear kernel [10]. Later, non-linear kernels we introduced. These employ the kernel trick [11]. Figure 2 shows support vectors for each kernel on a 2D plane. This provides an intuition for each kernel.

2.2.4. Hyperplane Coefficients

The l1 regularization term leads to sparse models. So, they include fewer features - making them easier to interpret. Eq 1 defines the total hyperplane as

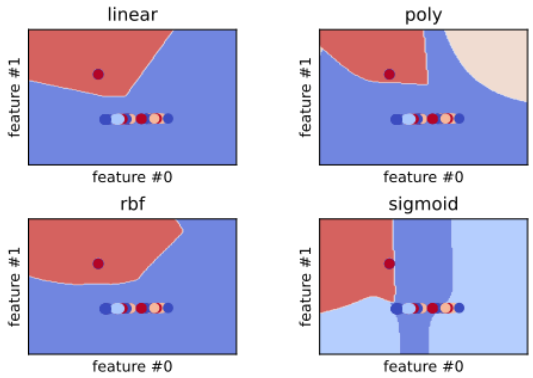


Fig. 2: SVM kernels shapes are shown. Specifically, linear, polynomial, radial basis function (rbf) and sigmoidal kernal are shown.

$$\beta_{\mathrm{t}} = \mathrm{minmax}(\sum_{\mathrm{c}\in\mathrm{C}} |\beta_{\mathrm{c}}|) \tag{1}$$

where there is the number of classes ($c \in C$) sets of hyperplane coefficients. β_{t} coefficient as the sum of hyperplane coefficients magnitude for each class β_{c}. We normalize the coefficients with a min-max feature scaling. The total hyperplane for both datasets is given in Figure 3.

2.3. Feature Selection

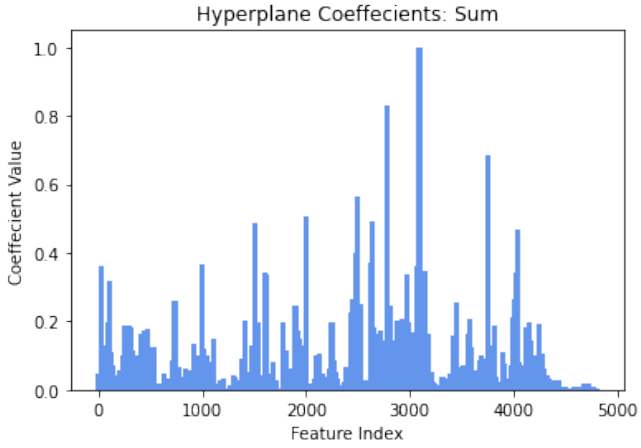
Feature selection reduces the complexity of the problem space. This helps counteract the curse of dimensionality [12]. Reducing the complexity improves computational efficiency, increases interpretability, and can improve performance. More interpretable models are easier for humans to understand. This means we can verify their efficiency using domain expertise in biochemistry. This is an important factor for real-world applications in a factory setting.

2.4. Visualisation

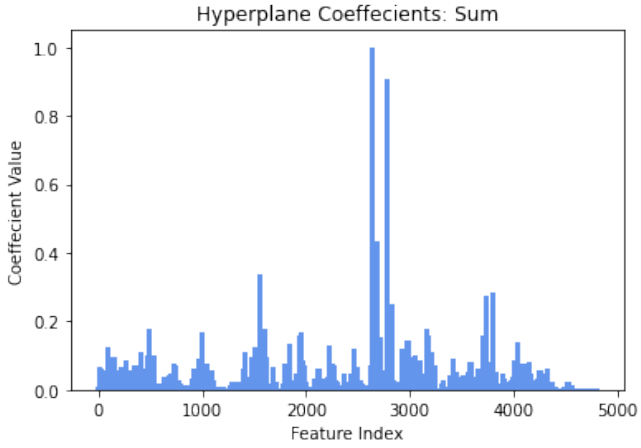
Two heuristics are optimized for when selecting a suitable model:

1. Interpretability
2. Accuracy

Interpretability is important for verification in a safety-critical environment. We intend to employ the chosen model in a factory setting. Accuracy is preferable, but not at the expense of interpretability. The efficacy of the model must be explainable through domain knowledge. Or else it is difficult to ensure reliability. The focus on interpretability ensures the model can be used in the real world.



(a) Fish Species Hyperplane Coefficients



(b) Fish Part Hyperplane Coefficients

Fig. 3: Hyperplane coefficients β_t. The normalized sum of the magnitude of the coefficients for each class is given in Eq 1. (a) Coefficients for the fish species dataset. (b) Coefficients for the fish part dataset.

Model interpretability is explored through visualisation. We aim to uncover learnt patterns that can be verified with domain knowledge. The desired algorithm should strike a balance between predictive performance and semantically meaningful features.

What constitutes semantic meaning varies from one domain to another. It is easy to build intuition for semantic meaning in computer vision and natural language processes, they correspond to recognisable images and structured text. In the domain of food science, our meaning is derived from performance on the classification task(s) and similarity to underlying chemical compounds.

3. RELATED WORKS

Here, we provide a literature review of state-of-the-art papers related to the topics discussed in the background. These papers are foundational and motivate the methods presented in this paper.

Specifically, we examine:

1. Gas Chromatography
2. Classification
3. Feature Selection
4. Visualisation

3.1. Gas Chromatography

Tomasi et al. investigated correlation optimisation warping (COW) and dynamic time warping (DT) for preprocessing chromatography data [13]. Unconstrained dynamic time warping was found to be too flexible. The algorithm overcompensated when trying to fix the alignment in the data.

Zhang et al. proposed a 2-D COW algorithm for aligning gas chromatography and mass spectrometry data [14]. The algorithm warps local regions of the data to maximise the correlation with known reference samples. This work uses data fusion with labelled reference samples, to improve the quality of new samples.

3.2. Classification

Bi et al. proposed a CNN model that incorporated GC-MS data fusion for food science [3]. The high-dimensional data was naturally suited towards the CNN. Their work classified the flavour quality of peanut oil with 93% accuracy. Similar to this project, the existing technique for analysis was intractable large scale. The fusion of existing datasets improved the efficacy of their model.

Matyshuin et al. proposed a stacking model for analysis of gas-chromatograph data [4]. It stacked the results of 1DConv, 2DConv, Deep Residual MLP and XGBoost. Their model predicted the retention index for samples. A retention index is a standardized value that only depends on the chemical structure of a compound. Once identified the retention index can be used for further identification. GC-MS data has underlying patterns that correspond to chemical compounds.

3.3. Feature Selection

Demelza et al. [15] proposed a feature and latent variable selection method for regression models in food science. The vibrational spectroscopy dataset shared similarities in its high dimensionality and food science domain. The purposes GA-PLSR generalized better and produced fewer complex models. The study showed that Genetic Algorithms are powerful tools for feature selection in food science.

Brown et al. [16] proposed a PSO with novel initialising and updating mechanisms. The initialization strategy utilized both forward and backwards selection. The updating mechanism overcame the limitations of the traditional method by considering the number of features. The proposed algorithm had better performance in terms of computing, fewer features selected and classification accuracy.

Nguyen et al. [17] proposed a wrapper based PSO technique for feature selection in classification. The algorithm uses a wrapper based fitness function of the classification error rate. The local search only considers the global best using a filter based method. It draws from the strengths of filter and wrapper based feature selection. This proposed method outperformed three state-of-the-art and two traditional feature selection methods.

Tran et al. [18] propose a Variable-Length PSO. Traditional PSO methods for feature selection are limited in the fixed length of their representation. This leads to both high memory usage and computational cost. The proposed algorithm allows particles to have shorter and different variable lengths. Their length changing mechanism allows PSO to escape local optima. Results across several high dimensional datasets showed improved performance in terms of computational time, fewer features selected and classification accuracy.

3.4. Visualisation

Mikolov et al. found the word embeddings used in NLP were semantically meaningful [19]. They showed arithmetic could be applied to these word vectors that were interpretable. For example "King" - "Man" + "Woman" = "Queen". The feature space was semantically meaningful, which serves as a powerful representation, that we intuitively reason with. Similar thought has been applied to computer vision [20, 21]. Semantically meaningful feature spaces allow for intuition about the behaviour of complex models, be it through visualisation or arithmetic.

Tegmark et al. developed they AI Feynman [22]. This algorithm can derive physics equations from data using symbolic regression. Symbolic regression is a difficult task, but by simplifying properties exhibited by physics equations (i.e symmetry, composability, separability), the problem can be reduced. Their work uses blackbox neural networks, to derive interpretable models that can easily be verified by humans.

4. RESULTS & DISCUSSIONS

4.1. Dataset

We perform analysis on a dataset of 154 gas chromatography samples. Each sample has 4800 features that correspond to timestamps. Figure 1 shows an example. This is low sample-size high-dimensional data. Each sample has two labels. Species - the species of fish the sample belongs to. Part - the part of the body the sample is from. We create two labelled datasets: species and part - these both share the same features but have different class labels.

4.2. Classification

We measure the predictive ability of classifiers on both the fish species and part dataset. We are looking for the model with the highest accuracy. As a result, we start broadly by exploring a variety of models from the different families of AI, then we narrow and refine the search.

Specifically, we examine:

1. Ensemble
2. SVM Model
3. SVM Kernel

For each of the following experiments, the same experiment setup is used. We use stratified cross-validation ($k = 10$) to measure the classification accuracy. Each method has its performance recorded on the same cross-folds. Then we average over 30 independent runs. This experimental setup evaluates performance on both the fish species and part datasets.

4.3. Ensemble

This is a broad search for an effective classification model. We explore a classifier from each family of AI. The model with the highest classification accuracy on both datasets is then selected, and explored by later sections in further detail.

We examine 5 classification models:

1. K-Nearest Neighbors [23]
2. Random Forest [24]
3. Naive Bayes [25]
4. Decision Tree [26]
5. Support Vector Machine [7]

Table 1 shows for random forest, decision tree and support vector machine have perfect training accuracy. The decision tree and random forest overfit the training data. Only the SVM achieves similar performance on the test data. The SVM classifier outperforms the other classifiers. It does so for the test set for both the species and part datasets.

Dataset	Method	AvgTrain $\pm$ Std	AveTest $\pm$ Std
Species	KNN	83.57 $\pm$ 1.80	74.88 $\pm$ 12.54
	RF	1.00 $\pm$ 0.00	85.65 $\pm$ 10.76
	DT	1.00 $\pm$ 0.00	76.98 $\pm$ 13.12
	NB	79.54 $\pm$ 1.60	75.27 $\pm$ 4.35
	SVM	**1.00 $\pm$ 0.00**	**98.33 $\pm$ 5.00**
Part	KNN	68.95 $\pm$ 3.49	43.61 $\pm$ 13.48
	RF	1.00 $\pm$ 0.00	72.60 $\pm$ 16.15
	DT	1.00 $\pm$ 0.00	60.14 $\pm$ 14.57
	NB	65.54 $\pm$ 2.69	48.61 $\pm$ 12.19
	SVM	**1.00 $\pm$ 0.00**	**87.14 $\pm$ 8.52**

Table 1: Accuracy for different classification techniques. Accuracy is given as the stratified k-fold cross validation over 30 independent runs. We compare K-nearest neighbours (KNN), random forest (RF), decision tree (DT), naive bayes (NB) and support vector machines (SVM).

4.3.1. SVM Model

The classification results showed that SVM was the most effective classifier. Now, we explore the variations in models for the SVM classifier. We use the same cross-validation setup as before.

We examine 3 SVM models [8]:

1. Suport Vector Classification [7]
2. Nu-Support Vector Classification [9]
3. Linear Support Vector Classification

Dataset	Method	AvgTrain $\pm$ Std	AveTest $\pm$ Std
Species	svc	88.96 $\pm$ 1.40	80.00 $\pm$ 12.33
	nusvc	88.30 $\pm$ 1.17	81.73 $\pm$ 12.75
	lsvc	**1.00 $\pm$ 0.00**	**98.33 $\pm$ 5.00**
Part	svc	73.25 $\pm$ 3.54	49.03 $\pm$ 12.14
	nusvc	90.31 $\pm$ 1.97	62.36 $\pm$ 15.18
	lsvc	**1.00 $\pm$ 0.00**	**87.16 $\pm$ 8.56**

Table 2: Accuracy for different SVM models. Accuracy is given as the stratified k-fold cross validation over 30 independent runs. We compare Support-Vector Classification (SVC), Nu-Support Vector Classification (Nu-SVC) and Linear Support-Vector Classification (LSVC).

Table 2 shows for fish species, SVC and Nu-SVC models have similar performance on both train and test. The Nu-SVC outperforms the SVC for both train and test for the part dataset. Yet, the linear SVC outperforms both models. It achieves perfect training accuracy for both datasets. For the test, near-perfect (98.33%) on species, and reasonable performance (87.16%) on the part.

4.3.2. SVM Kernel

Now we know that SVM is the most effective classifier, and the LSVC is the most effective model. To provide an exhaustive search, we explore all possible kernels. We use the same cross-validation setup as before.

We examine 4 SVM kernels [8]:

1. Polynomial
2. Radial Basis Function (rbf)
3. Sigmoid
4. Linear [10]

Dataset	Method	AvgTrain ± Std	AveTest ± Std
Species	poly	76.83 ± 1.18	71.37 ± 15.86
	rbf	88.96 ± 1.40	80.00 ± 12.33
	sigmoid	33.19 ± 2.36	30.18 ± 6.50
	linear	**1.00 ± 0.00**	**97.50 ± 5.34**
Part	poly	70.63 ± 2.27	53.89 ± 6.94
	rbf	73.25 ± 3.54	49.03 ± 12.14
	sigmoid	37.47 ± 1.78	33.47 ± 8.59
	linear	**1.00 ± 0.00**	**87.36 ± 10.77**

Table 3: Accuracy for different SVM kernals. Accuracy is given as the stratified k-fold cross validation over 30 independent runs. We compare polynomial (poly), radial basis function (rbf), sigmoidal (sigmoid) and linear.

Table 3 shows the sigmoid kernel performs very poorly on training and test for both datasets. The polynomial and RBF kernel achieve comparable performance for both datasets. The linear kernel outperforms all other kernels for both datasets. It has near-perfect (97.50%) test accuracy on fish species. And reasonable performance (87.36%) on the fish part.

4.3.3. Discussion

We evaluated an ensemble of classification techniques. Naive Bayes performed poorly. This is likely due to the assumption of conditional independence between features. KNN also performed poorly. This is likely due to the high dimensionality of the data. Points drawn from high dimensional spaces tend to never be close together. SVM provided the best results. This model can identify fish species from gas chromatography data with near-perfect accuracy. This prompted further investigation into this technique.

Within support vector machines, the Linear SVC model showed the best performance. Naturally, within SVM kernels, the linear kernel also showed the best performance. There is high predictive performance on the linear kernel. This suggests an underlying pattern that is linearly separable in a hyperplane. Non-linear kernels - polynomial, RBF or sigmoidal - produced diminishing returns. These kernels try to fit complex patterns that are not present in the data.

Performance for all models was better for the fish species than the part. This suggests tissue samples for different species may have distinct chemical compositions. Yet, different fish parts may have fewer underlying structural differences. For GC data the intra-class variation between species provides a larger signal than part variation. For example, we expect there to be more difference between a tarakihi and a bluecod, than there is a similarity between two livers from each species.

4.4. Feature Selection

For each method, we measure classification accuracy with an SVM model [8]. It has linear kernel, l1 regularization [27] and 10,000 maximum iterations. We examine 4 feature selection methods [28]:

1. Chi^2 [29]
2. Minimum Redundancy Maximum Relevance [30]
3. ReliefF [27]
4. Particle Swarm Optimization [31, 32]

We first provide a detailed accuracy comparison for a set feature number ($k = 500$). Then we explore the accuracy for the general case (any k).

4.4.1. Classification Accuracy $k = 500$

We measure the classification accuracy at $k = 500$ for each method. To allow comparison with PSO, we take the top k features suggested by the algorithm and compare this to the others.

Dataset	Method	AvgTrain ± Std	AveTest ± Std
Species	Chi^2	95.17 ± 3.52	81.85 ± 9.65
	MRMR	99.79 ± 0.41	95.09 ± 6.90
	ReliefF	**99.71 ± 0.44**	**95.12 ± 6.26**
	PSO	99.71 ± 4.30	93.30 ± 8.16
Part	Chi^2	96.32 ± 0.88	64.86 ± 19.01
	MRMR	97.44 ± 0.97	78.79 ± 13.21
	ReliefF	97.82 ± 1.04	80.28 ± 5.58
	PSO	**97.62 ± 0.91**	**82.36 ± 10.72**

Table 4: Accuracy for different feature selection methods. Accuracy is given as the stratified k-fold cross validation over 30 independent runs. We compare chi^2 (chi), maximum relevance - minimum redundancy (MRMR), reliefF, particle swarm optimisation (PSO).

Table 4 shows for the training set, MRMR, ReliefF and PSO have comparable accuracy for both datasets. The Chi^2 method does not, instead if performs very poorly. For the test set, ReliefF performs best for species, PSO performs best for the part.

4.4.2. *Classification Accuracy (all k)*

We measure classification accuracy as a function of feature number. We compared this for several FS methods. Due to limitations, PSO optimizes feature number k automatically. So, to compare its performance, we plot the results of 30 independent runs.

Specifically, we compare the following methods:

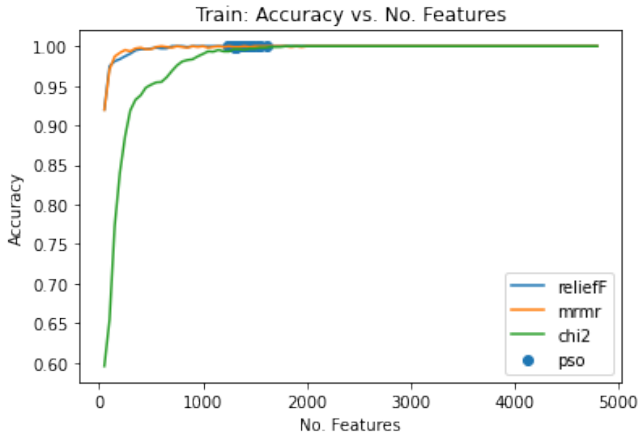
- reliefF [10]
- MRMR [30]
- PSO [31, 32]
- chi^2 [29]

Figure 4a shows accuracy for fish species. We show accuracy on the training set for each feature selection method. At $k = 1050$ all feature selection methods achieve 100% accuracy on the training set. The SVM fits the training data for each method using a fraction of the full feature set. Figure 4b shows accuracy for fish species. We show test set accuracy for each feature selection method. The accuracy reaches a plateau (96% accuracy) at around $k = 1050$ features for all methods. The test performance is less than the train performance, yet the test accuracy is still very high. This suggests the model can generalize well on unseen data for the fish species.

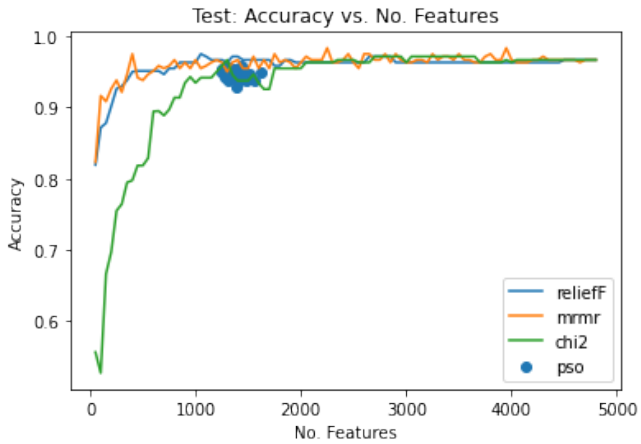
Figure 5a shows accuracy for part dataset. We show train accuracy for each feature selection method. All feature selection methods struggle to fit the training set for the fish part. Even with the full feature set, a perfect train accuracy is never reached. Figure 5b shows accuracy for part dataset. We show the test accuracy for each feature selection method. The classification accuracy fluctuates for all feature selection methods. At around $k = 1050$ features, it begins to decrease. The training accuracy improves, as the test does not from this point onwards. The SVM is overfitting to noise (redundant features) in the training set.

4.4.3. *Disucssion*

Feature selection methods helped reduce dimensionality. We evaluated performance with an SVM classifier. In which, ReliefF and PSO were best for fish species and part, respectively. ReliefF can identify conditional dependencies between features when providing feature rankings. ReliefF algorithms are robust and noise-tolerant, which explains their superior



(a) Training set



(b) Test set

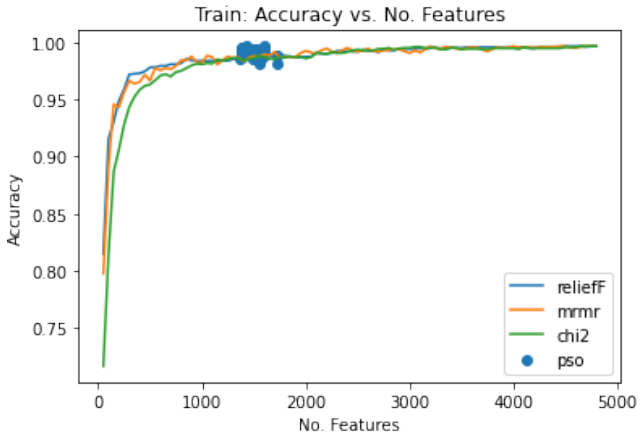
Fig. 4: Fish species dataset: Classification accuracy for feature selection methods for a given k. We measure the balanced accuracy of k-fold cross-validation. We compare reliefF, maximum relevance - minimum redundancy (MRMR), chi^2, and particle swarm optimisation (PSO). Fish part dataset. (a) Training set. (b) Test set.

performance. PSO provides a combination of global and local searches. A search through a near-infinite combinatorial space of possible feature subsets. This stochastic method is computationally expensive but can offer effective solutions.

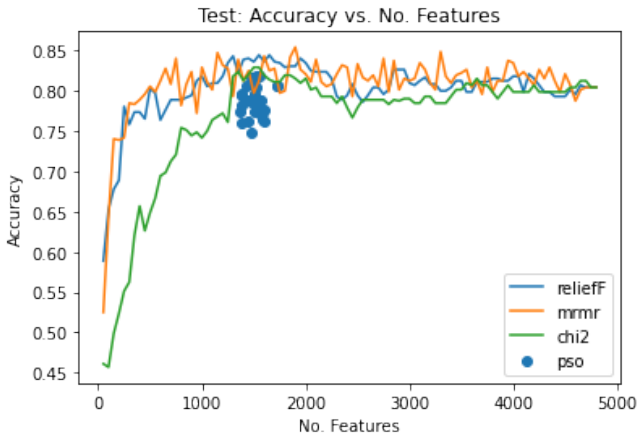
For both general and specific cases, and across all methods, the fish species have lower variance than the fish part in classification accuracy. The classification results (§ 4.2) support this, they also show higher test accuracy for fish species, than fish part. They suggest different fish parts may have fewer underlying structural differences.

For the general case and both datasets, a lot of interesting behaviour happens at $k = 1050$. The fish species reach a plateau, but the fish part accuracy begins to decrease. An accuracy comparable or better than the full dataset is possible with 21.8% of its features.

For the fish species dataset, we see high accuracy with



(a) Training set



(b) Test set

Fig. 5: Fish part dataset: classification accuracy for feature selection methods for a given k. We measure the balanced accuracy of k-fold cross-validation. We compare reliefF, maximum relevance - minimum redundancy (MRMR), chi^2, and particle swarm optimisation (PSO). (a) Training set. (b) Test set.

very few features. ReliefF and MRMR can achieve above 90% classification accuracy with $k = 50$. Chi^2 is not able to mirror this performance. This shows ReliefF and MRMR are very effective feature selection methods for this task.

The PSO may not have a hyperparameter for feature number k. Instead, it automates the selection of this parameter. Yet, it achieves comparable results to other state-of-the-art methods. This automation may prove useful for automating the classification task for online learning. In a factory, we may want to train a model as new data arrives. PSO requires less human intervention, yet still, provides competitive performance.

MRMR and ReliefF both have high accuracy with very few features on the fish species dataset. This suggests that few features are required to construct a reasonable representation of a fish tissue sample. This is a good indication that the fish species dataset contains less noise. This also warrants further investigation into which features are considered important for low k values. This motivates the following section on visualisation.

4.5. Visualisation

In this section, we explore the interpretability of the feature selection methods. Firstly, We the features and their rankings for low k values. Then investigate if there is a superset of important features - these would be universally recognized by all methods - that is, the overlap or common features. Thirdly, the classification accuracy of the overlap is compared to its parts to see if the superset even exists. Finally, we compare the most effective methods (reliefF and MRMR) to the most effective classifier (SVM).

Specifically, we examine:

1. Feature Rankings ($k = 50$)
2. Overlap: MRMR $\cap$ ReliefF
3. Accuracy: Overlap, MRMR, ReliefF
4. Overlap: FS Method $\cap$ Hyperplane Coefficients

4.5.1. Experimental Setup

When evaluating classification accuracy we use a stratified cross-validation ($k = 10$). We give the average balanced accuracy over those k folds. When experiments are examining feature selections/rankings, we fit the entire dataset. We repeat each experiment for both fish species and part datasets.

The number of features k is the independent variable for many of these methods. We exclude PSO [31, 32] from the analysis. Since k is not a hyper-parameter for this algorithm. We analyse feature importance for three feature selection methods.

Specifically, we compare the following methods:

- reliefF [10]
- MRMR [30]
- chi^2 [29]

4.5.2. Feature Rankings (k=50)

We compare the feature rankings for the top 50 ($k = 50$) features for each FS method. We normalize scores ($x \in [0, 1]$) to allow comparison between methods. We see if each FS method selects similar features for a low number of features ($k = 50$). If true, this suggests that there are important features. All techniques would recognize these. Previous work

[33, 34] has shown worse performance for chi^2. The visualisation also helps gain an intuition for why this could be the case.

These results relate to the fish species dataset. Figure 6a shows relierF favors higher feature indexes ($i \in [4000, 4800]$). The feature indexes tend to bunch towards this range. Figure 6b shows MRMR samples from the entire set of feature indexes ($i \in [0, 4800]$). This method samples feature indexes with a more uniform distribution than reliefF. Figure 6c shows chi^2 selects chains of features adjacent to each other. These chains hint the method is not able to make a good selection for low values of k. We would expect poor performance for the chi^2 method for low values of k - the feature selection (§ 4.4) results confirm that claim.

These results relate to the fish part dataset. Figure 7a shows reliefF favours feature indexes bunching towards the higher range. Figure 7b shows MRMR gives a more uniform distribution of features. Figure 7c shows chi^2 tends to cluster of features around index $i = 3000$. The features cluster more for the fish part than the species.

4.5.3. Overlap: MRMR ∩ ReliefF

We give overlap - the intersection of features selected by MRMR and ReliefF for a given k - as MRMR ∩ ReliefF. The overlap measures the similarities between the feature subsets chosen by each method. If the overlap value is high, this suggests there are a distinct set of important features. Here, we would expect both methods to recognize these early, or, there may be low overlap. A low overlap would suggest that there are many important features, each method can find unique important features on its own. We exclude chi^2 from this analysis because there was little overlap for lower k values with other methods.

Figure 8 shows this overlap. Intuitively, the limit of the overlap, when k approaches all features k_{max}, is the number of features k_{max}. Due to this limit, we are more interested in the left-hand side of the graph, the low k values. For low k values the overlap is very small. There are a few features important enough to be selected early by both methods, this result is consistent for both datasets.

4.5.4. Accuracy: Overlap, MRMR, ReliefF

We compare the classification accuracy of the MRMR, ReliefF and their overlap (MRMR ∩ ReliefF) for a given k. Chi^2 had poor performance in the classification task [34]. Also, this method has little to no overlap. So, we no longer consider chi^2 in our analysis. Now, we determine if the first k features selected by each FS method are the most important. Say classification accuracy for the overlap exceeds that of its parts (MRMR, ReliefF). This would mean the most important features are the first k features selected by each method. Or, the overlap may share the same classification accuracy as its parts. This would suggest many features are all important. Feature subsets with the same accuracy are as important as each other.

These results relate to fish species. Figure 9a suggests all subsets of features - MRMR, reliefF and overlap - can fit the data for any k. Figure 9b shows different feature subsets achieve similar test accuracy - so, there are many important features. This suggests that each feature subset are equally important. The feature selection results (§ 4.4) support this, they showed high accuracy with few features selected ($k = 50$). This result supports that conclusion.

These results relate to the fish part. Figure 10a suggests all subsets of features - MRMR, reliefF and overlap - can fit the data for any k. Figure 10b fluctuates wildly. This is consistent with previous findings [33, 34]. This supports that classifying fish parts is more difficult than fish species. Due to the noise in the results, it is difficult to distinguish a clear winner for the part dataset.

4.5.5. Overlap: FS Method ∩ Hyperplane Coefficients

We compare the hyperplane coefficients β_t to the FS methods method. The SVM classifier has high predictive capabilities on both datasets [33, 34]. Thus, we expect the FS method most like SVM is of interest.

Figure 11 shows reliefF with the most consistent overlap with SVM coefficients, this is true for both datasets. The chi^2 method shows wild fluctuations, it is similar for high k values for the fish species, but then similar for low k for the fish part. Apart from a small region of low k values, MRMR shares little overlap for the fish species. For the fish part, MRMR shares little overlap with SVM coefficients, when compared to others. In terms of features selected, reliefF is the most similar to the Linear SVM.

4.5.6. Discussion

The visualisation has shown us why chi^2 is a poor fs method. It tends to bunch features together. MRMR and reliefF are far better. This is true for the distribution of feature indexes, overlap and classification accuracy. MRMR has the most uniform distribution of feature indexes. Yet, reliefF selects features most like SVM coefficients.

When we select a few features, there was little overlap between MRMR and reliefF. There were no clear k best features consistent for both techniques. The overlap of features selected did not perform better than its parts (MRMR and reliefF). This shows there are lots of important features.

These FS methods were not able to converge on a small superset of important features. That is because there are lots of important features for both datasets. Each feature represents an intensity at a particular timestamp in the gas chromatograph. These results show we can use many chemical combinations to classify fish oil data. It may be easier to

match some reference chemicals than others. There is a range of possibilities for combinations of reference chemicals. This research shows we may reduce the cost of this analysis in future.

5. CONCLUSION

The analysis of fish oil data in this paper has focussed on interpretability. Not only have we found effective classification and feature selection techniques, but we have also tried to explain their performance with visualisation and analytical results. We can draw many conclusions from the analytical results and visualisations, but here we recall the most important:

1. Fish species are easier to predict than fish part - there is more intra-class variation within fish species than there is a similarity between the same part from different fish.
2. The Linear SVM classifier performs better for both classification tasks - the fish oil data is linearly separable on a hyperplane.
3. Near-perfect accuracy can be achieved with very few features for fish species - if this predictive ability exceeds human error this model has real-world applications.
4. Comparison with PSO may be difficult (due to automatic k selection), but this automation may be useful in a factory setting.
5. There are many important features and their subsets that can be used to classify fish oil - thus, there is a range of possibilities for combinations of reference chemicals.

Feature selection is not guaranteed to improve classification accuracy. Yet, it does reduce the complexity, increase interpretability and improve computational efficiency. As with the SVM using an l1 regularization, feature selection is also a trade-off between interpretability and performance. Arthur C. Clarke said, "[a]ny sufficiently advanced technology is indistinguishable from magic." [35]. A perfect blackbox model is equivalent to magic, and, it is difficult to have faith in magic, especially if it were to be involved in our food making. Faith isn't needed when our models when are interpretable.

6. APPENDIX

An interactive notebook for the analysis of the data is available online on Google Collab:

1. Classification [33]
2. Feature Selection [34]
3. Visualisation [36]

7. REFERENCES

[1] K Eder, "Gas chromatographic analysis of fatty acid methyl esters," *Journal of Chromatography B: Biomedical Sciences and Applications*, vol. 671, no. 1-2, pp. 113–131, 1995.

[2] Restek, "High-resolution gc analyses of fatty acid methyl esters (fames)," `https://www.restek.com/en/technical-literature-library/articles/high-resolution-GC-analyses-of-fatty-`

[3] Kexin Bi, Dong Zhang, Tong Qiu, and Yizhen Huang, "Gc-ms fingerprints profiling using machine learning models for food flavor prediction," *Processes*, vol. 8, no. 1, pp. 23, 2020.

[4] Dmitriy D Matyushin and Aleksey K Buryak, "Gas chromatographic retention index prediction using multimodal machine learning," *Ieee Access*, vol. 8, pp. 223140–223155, 2020.

[5] Khan Academy, "Gas chromatography," 2013.

[6] Elon Musk, "2020 annual meeting of stockholders and battery day: Tesla," Sep 2020.

[7] Corinna Cortes and Vladimir Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[8] Sklearn, "1.13. feature selection," `https://sklearn.org/modules/feature_selection.html`.

[9] Bernhard Schölkopf, Alex J Smola, Robert C Williamson, and Peter L Bartlett, "New support vector algorithms," *Neural computation*, vol. 12, no. 5, pp. 1207–1245, 2000.

[10] Mark A Aizerman, "Theoretical foundations of the potential function method in pattern recognition learning," *Automation and remote control*, vol. 25, pp. 821–837, 1964.

[11] Bernhard E Boser, Isabelle M Guyon, and Vladimir N Vapnik, "A training algorithm for optimal margin classifiers," in *Proceedings of the fifth annual workshop on Computational learning theory*, 1992, pp. 144–152.

[12] Mario Köppen, "The curse of dimensionality," in *5th Online World Conference on Soft Computing in Industrial Applications (WSC5)*, 2000, vol. 1, pp. 4–8.

[13] Giorgio Tomasi, Frans Van Den Berg, and Claus Andersson, "Correlation optimized warping and dynamic time warping as preprocessing methods for chromatographic data," *Journal of Chemometrics: A Journal of the Chemometrics Society*, vol. 18, no. 5, pp. 231–241, 2004.

[14] Dabao Zhang, Xiaodong Huang, Fred E Regnier, and Min Zhang, "Two-dimensional correlation optimized warping algorithm for aligning gc× gc- ms data," *Analytical Chemistry*, vol. 80, no. 8, pp. 2664–2671, 2008.

[15] Bing Demelza, Robinson, Mengjie Zhang, and et al, "Genetic algorithm for feature and latent variable selection for nutrient assessment in horticultural products," pp. 1–8, 2020.

[16] Bing Xue, Mengjie Zhang, and Will N Browne, "Particle swarm optimisation for feature selection in classification: Novel initialisation and updating mechanisms," *Applied soft computing*, vol. 18, pp. 261–276, 2014.

[17] Hoai Bach Nguyen, Bing Xue, Ivy Liu, and Mengjie Zhang, "Filter based backward elimination in wrapper based pso for feature selection in classification," in *2014 IEEE congress on evolutionary computation (CEC)*. IEEE, 2014, pp. 3111–3118.

[18] Binh Tran, Bing Xue, and Mengjie Zhang, "Variable-length particle swarm optimization for feature selection on high-dimensional classification," *IEEE Transactions on Evolutionary Computation*, vol. 23, no. 3, pp. 473–487, 2018.

[19] Tomáš Mikolov, Wen-tau Yih, and Geoffrey Zweig, "Linguistic regularities in continuous space word representations," in *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, 2013, pp. 746–751.

[20] Chris Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev, "The building blocks of interpretability," *Distill*, vol. 3, no. 3, pp. e10, 2018.

[21] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, "Analyzing and improving the image quality of stylegan," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8110–8119.

[22] Silviu-Marian Udrescu and Max Tegmark, "Ai feynman: A physics-inspired method for symbolic regression," *Science Advances*, vol. 6, no. 16, pp. eaay2631, 2020.

[23] Evelyn Fix and Joseph Lawson Hodges, "Discriminatory analysis. nonparametric discrimination: Consistency properties," *International Statistical Review/Revue Internationale de Statistique*, vol. 57, no. 3, pp. 238–247, 1989.

[24] Tin Kam Ho, "Random decision forests," in *Proceedings of 3rd international conference on document analysis and recognition*. IEEE, 1995, vol. 1, pp. 278–282.

[25] David J Hand and Keming Yu, "Idiot's bayes—not so stupid after all?," *International statistical review*, vol. 69, no. 3, pp. 385–398, 2001.

[26] Wei-Yin Loh, "Classification and regression trees," *Wiley interdisciplinary reviews: data mining and knowledge discovery*, vol. 1, no. 1, pp. 14–23, 2011.

[27] Marko Robnik-Šikonja and Igor Kononenko, "Theoretical and empirical analysis of relieff and rrelieff," *Machine learning*, vol. 53, no. 1, pp. 23–69, 2003.

[28] Charlie Chappers, "charliec443/scikit-feature," `https://github.com/charliec443/scikit-feature`.

[29] Huan Liu and Rudy Setiono, "Chi2: Feature selection and discretization of numeric attributes," in *Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence*. IEEE, 1995, pp. 388–391.

[30] Chris Ding and Hanchuan Peng, "Minimum redundancy feature selection from microarray gene expression data," *Journal of bioinformatics and computational biology*, vol. 3, no. 02, pp. 185–205, 2005.

[31] James Kennedy and Russell C Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*. IEEE, 1995, vol. 4, pp. 1942–1948.

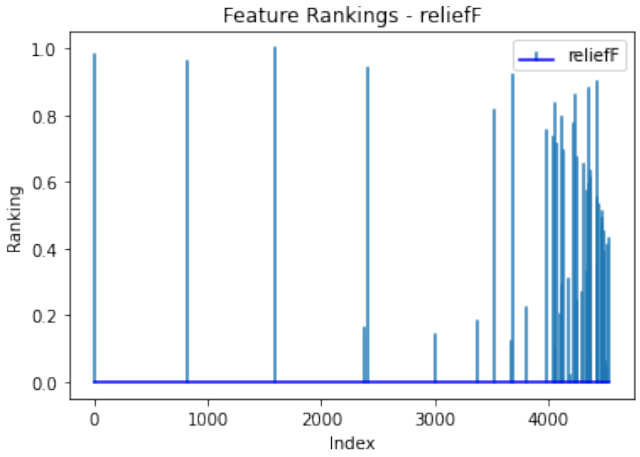
[32] James Kennedy and Russell C Eberhart, "A discrete binary version of the particle swarm algorithm," in *1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation*. IEEE, 1997, vol. 5, pp. 4104–4108.

[33] Jesse Wood, "Fish 1 - google colab notebook," `https://colab.research.google.com/drive/1h303x3Z7kVyJwggfvhq8WN7YFvIH5zt_?usp=sharing`.

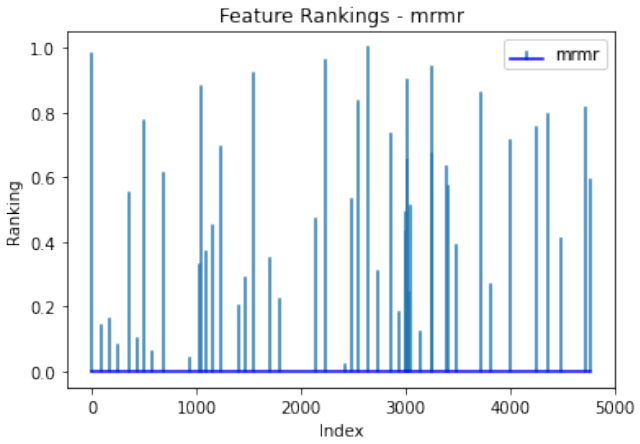
[34] Jesse Wood, "Feature selection methods on fish oil data," *Victoria University of Wellington*, 2022, DIS Assignment 1.

[35] Arthur C Clarke, *Profiles of the Future*, Hachette UK, 1962.

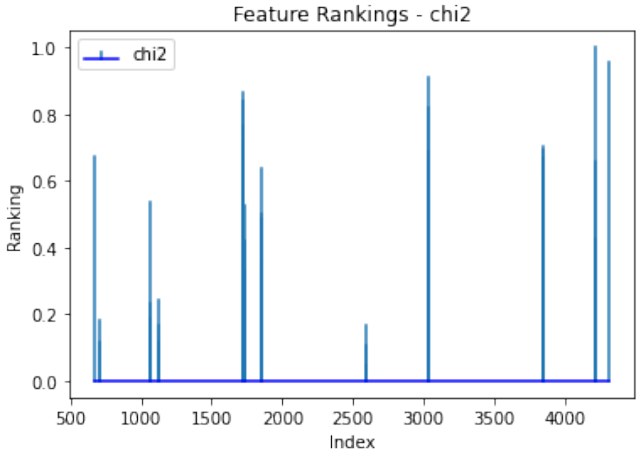
[36] Jesse Wood, "Fish 3 - google colab notebook," `https://colab.research.google.com/drive/16mpV3eKdjTeFM0Hp4o5-N2LAy1h4gpqo?usp=sharing`.



(a) reliefF

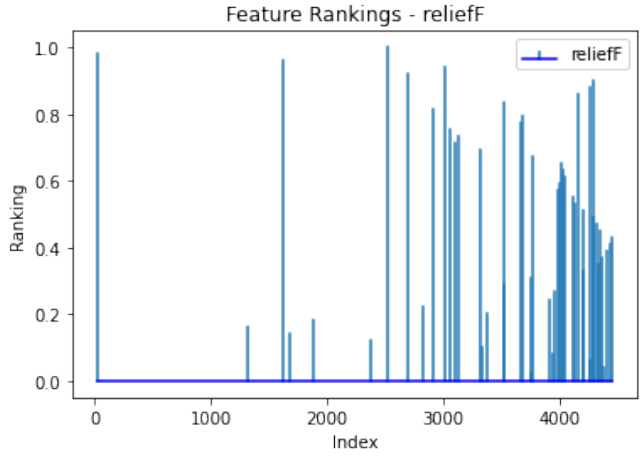


(b) Maximum Relevance — Minimum Redundancy (MRMR)

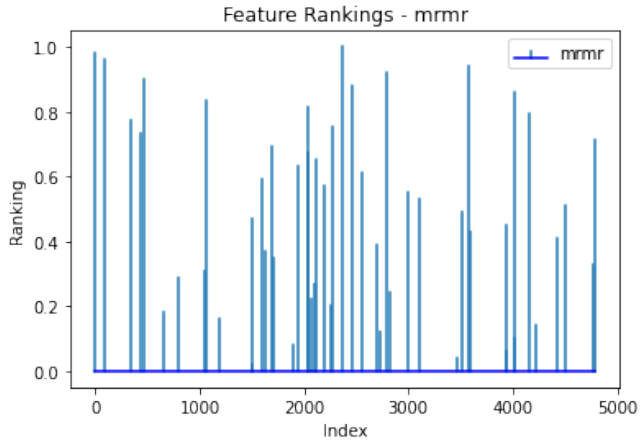


(c) chi^2

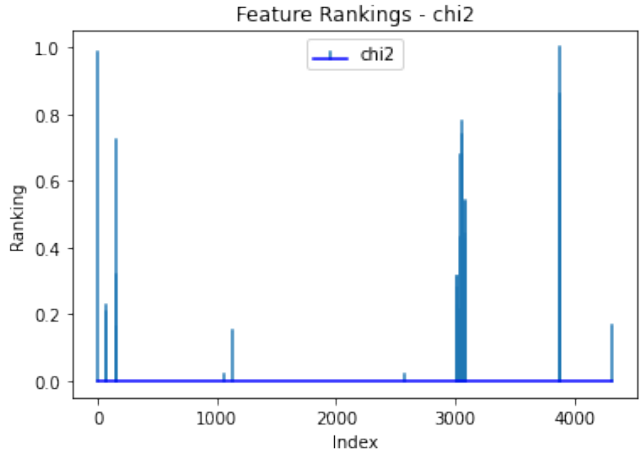
Fig. 6: Fish species: normalized feature rankings ($k = 50$). This gives the feature rankings for each feature index for feature selection methods. The results are normalized to facilitate easy comparison. (a) ReliefF. (b) MRMR. (c) Chi^2.



(a) reliefF

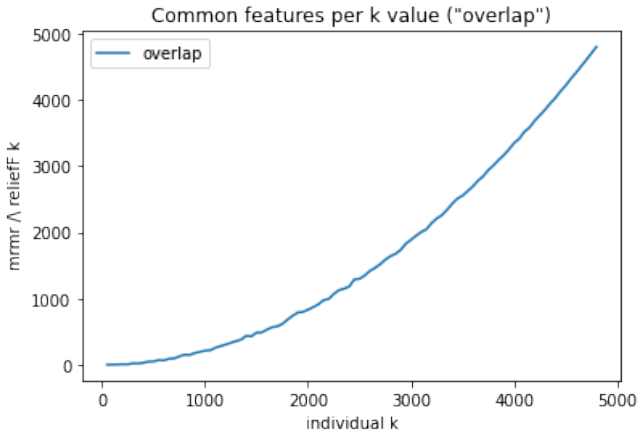


(b) Maximum Relevance — Minimum Redundancy (MRMR)

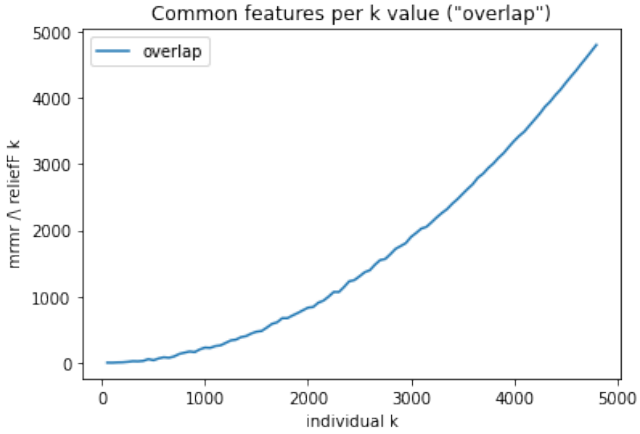


(c) chi^2

Fig. 7: Fish part: normalized feature rankings ($k = 50$). This gives the feature rankings for each feature index for feature selection methods. The results are normalized to facilitate easy comparison. (a) ReliefF. (b) MRMR. (c) Chi^2.

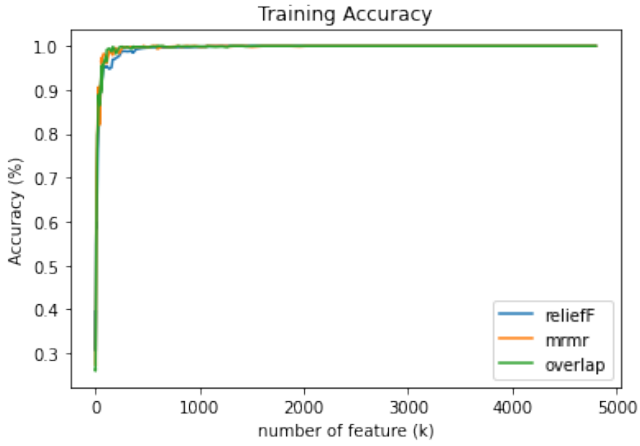


(a) Fish species dataset



(b) Fish part dataset

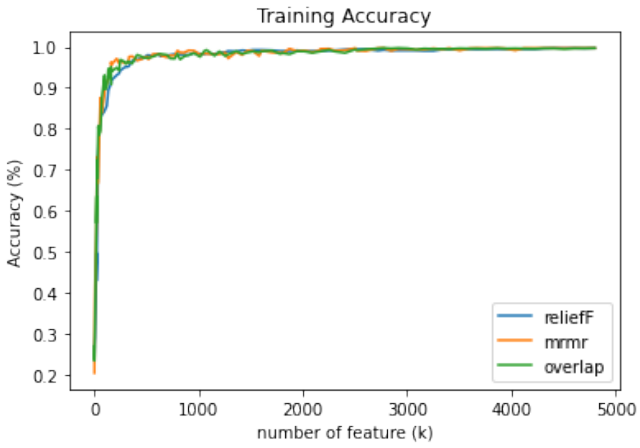
Fig. 8: Overlap - common features for MRMR and ReliefF. The overlap demonstrates the number of shared features between the two feature selection methods for a give k value. (a) Fish species. (b) Fish part.



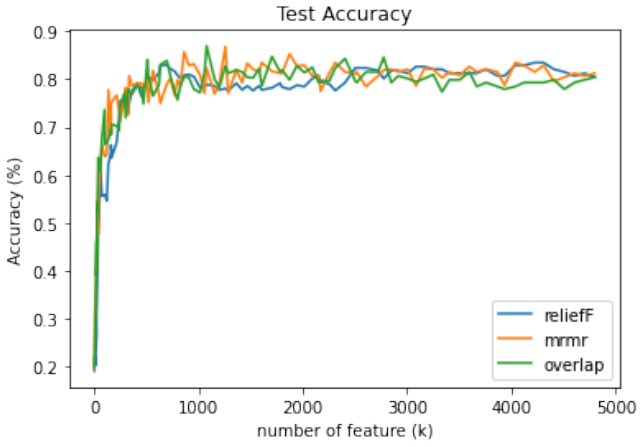
(a) Training set

(b) Test set

Fig. 9: Fish species: classification accuracy of the overlap, MRMR and ReliefF for a given k. We measure the balanced accuracy using k-fold cross-validation. The classification accuracy metric shows how important the selected features are for fish part species. (a) Training set. (b) Test set.

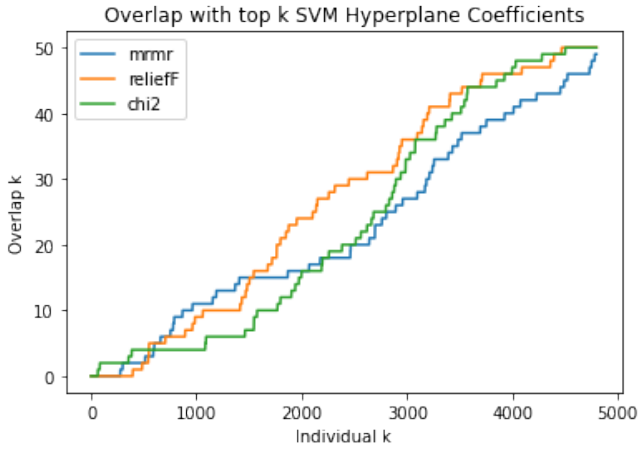


(a) Train set

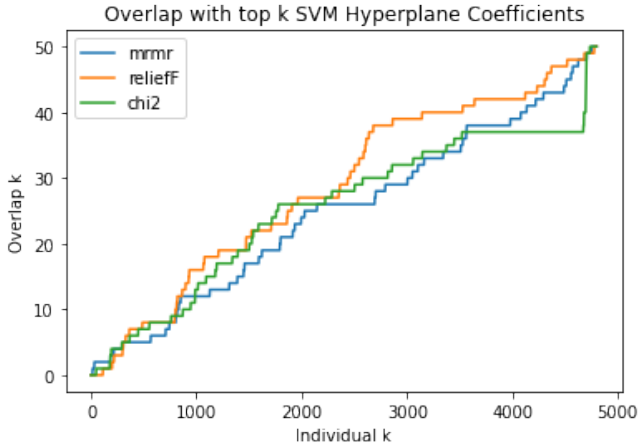


(b) Test set

Fig. 10: Fish part: classification accuracy of the overlap, MRMR and ReliefF for a given k. We measure the balanced accuracy using k-fold cross-validation. The classification accuracy metric shows how important the selected features are for the fish part dataset. (a) Training set. (b) Test set.



(a) Fish species dataset



(b) Fish part dataset

Fig. 11: Overlap - common features between SVM coefficients and feature selection methods for a given k value. We examine the maximum relevance - minimum redundancy (MRMR), relieF, and chi^2 methods. The overlap measure shows how similar each method is to the SVM classification method. (a) Fish species. (b) Fish part.