Machine Learning for Fish Oil Anaylsis

No Author Given

No Institute Given

Abstract. In fish processing worldwide, 60% of fish is not served. This fish wastage can be repurposed into other products such as fish oils and fish feed. It is difficult and requires domain expertise to effectively repurpose fish biomass. Gas chromatography (GC) can be used to identify tissue samples in fish processing. The existing analytical chemistry techniques for processing GC data are manual and time-consuming. To reduce biomass waste we must incentivise biomass re-use, we reduce this barrier of entry through automation. Here, we explore classification algorithms for fish oil data that automate and significantly reduce the time required to process GC data. Visualisation is used to explore the interpretability of the models such that their efficacy can be verified for use in a factory setting. The fish oil data is high-dimensional and low sample size data.

Keywords: Feature Selection · Gas Chromatography · Support Vector Machines · Food Science

1 Introduction

Fish biomass can be repurposed for fish oils - such as omega-3 vitamin supplements - or fish meal, both of which can for used in animal feeds. Further processing can produce refined fish oil that is suitable for fish oil capsules that we buy in shops and pharmacies. Some fish species or parts may be richer in omega-3 fatty acids than others, these can be used to make high-value products. To identify valuable fish oils we need to classify the species and part of tissue samples.

Gas chromatography [7] is a chemistry technique used to analyze fish oils. It can determine the structure of chemical compounds present in a given sample [17]. This is important for quality assurance in a factory setting, especially in food science. To repurpose fish biomass efficiently, first, we need to know what is in it. We can identify the contents of fish oil using classification techniques that compare it to known samples. Given a fish oil sample, we can identify the fish species (i.e. Bluecod, Tarakihi), and part (Head, Fins). Existing involve chemists comparing a given sample to reference samples to determine which class it likely belongs to.

1.1 Goals and Objectives

In this paper, we explore machine learning techniques to automate the process of identifying fish species and part on GC data. We achieve this objective by identifying effective classification algorithms, visualizing their models, and removing redundant features. Firstly, classification algorithms are evaluated for their ability to determine the fish species and part. Visualisation is used to explore the interpretability of successful models. It is important to verify their efficacy with domain knowledge before these algorithms can be deployed in a real-world setting. Secondly, feature selection is used to eliminate redundant features, whilst maintaining high-accuracy predictions.

2 Background

2.1 Gas Chromatorgraphy

Gas chromatography (GC) is a technique for the analysis of chemical compounds [7,17,1]. The process separates compounds based on their boiling point and molecular weight. A compound is injected as a liquid, then heat is applied to vaporize it into a 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 travel through a long coiled tube. That tube has a detector at the end, this detects the rate and intensity at which compounds reach the tube's end.

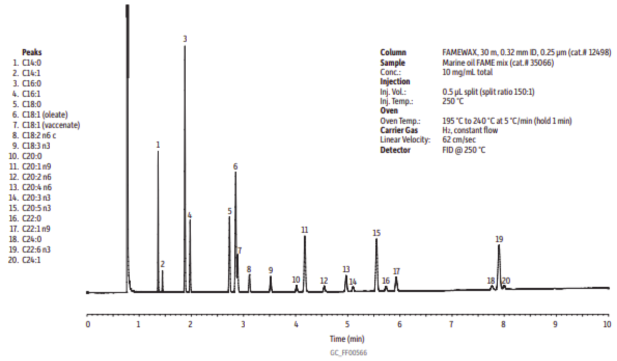


Fig. 1: Gas chromatograph: the artifact of the GC method [17]. The detection is used to visualize intensity (y) and time (x) on a chromotrogaph.

Chemists use the chromatograms of known compounds as a reference when classifying new ones. 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 [1], so it is often used in conjunction with other techniques like Mass spectrometry [17].

The existing task of classifying chemical compounds based on a chromatograph is laborious [7,17]. 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 - species and part.

An interpretable and accurate model has the potential to be deployed in a factory setting. This eliminates 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 [16].

2.2 Visualisation

Two heuristics are optimized when selecting a suitable model: interpretability and 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.

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 domains of gas chromatography and fish processing, our meaning is derived from performance on the classification task(s) and similarity to underlying chemical compounds. We expect models that generate knowledge that can be verified with domain expertise. For example, important features will correspond to timestamps of important chemicals in the GC data.

2.3 Support Vector Machines

Cortes and Vapnik proposed the Support Vector Machine (SVM) [5]. 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 [20]. This creates a divide between one class and the rest, then repeats for the other classes.

RBF is the default kernel, other models have different kernels and introduce hyperparameters. The NuSVC has hyperparameter *Nu*, this determines the number of support vectors [19]. This is a trainable parameter in the other SVM models.

The kernel function shapes the support vectors in the hyperplane. Each kernel can capture patterns of varying complexity. The original SVM had a linear kernel

[2], Later, non-linear kernels that employ the kernel trick were introduced [3]. Figure 2 shows support vectors for each kernel on a 2D plane, this provides an intuition for each kernel.

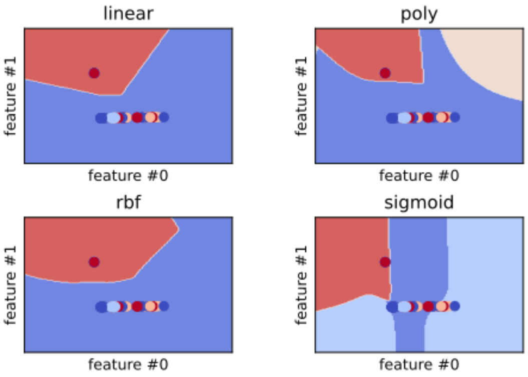


Fig. 2: SVM kernels shapes are shown. Specifically, linear, polynomial, radial basis function (rbf) and sigmoidal kernal are shown.

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

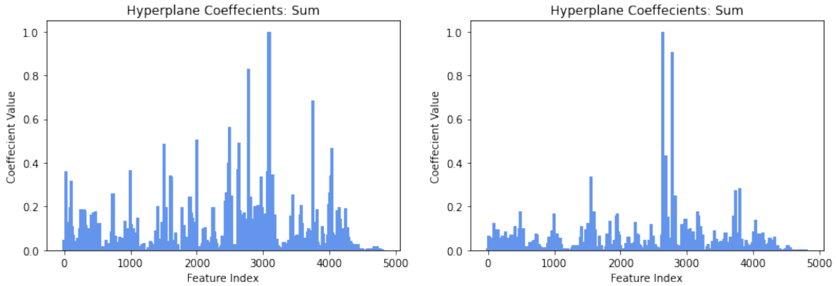
$$\beta_{\mathrm{t}} = \mathrm{minmax}(\sum_{\mathrm{c} \in \mathrm{C}} |\beta_{\mathrm{c}}|) \tag{1}$$

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. We visualize the hyperplane to approximate the important features. The outliers correspond to feature timestamps that are important for drawing class boundaries. These are chemical compounds that separate the fish part and species, respectively.

2.4 Feature Selection

Feature selection reduces the complexity of the problem space. This helps counteract the curse of dimensionality [13]. 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.



(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.

3 Data processing

- Why the raw data is not applicable to existing classification algorithms?
- Extracting datasets that are ready for classification algorithms:
  - Sum up the intensity.
  - Aligning missing packets.
- Overview of extracted data.

4 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, and then we narrow and refine the search.

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.

4.1 Classification Algorithms

We examine 5 classification models:

1. K-Nearest Neighbors [8]
2. Random Forest [10]
3. Naive Bayes [9]
4. Decision Tree [15]

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

Table 1: Accuracy for different classification techniques. Accuracy is given as the stratified k-fold cross validation over 30 independent runs.

5. Support Vector Machine [5]

Table 1 shows for the 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.

4.2 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 [20]:

1. Suport Vector Classification [5]
2. Nu-Support Vector Classification [19]
3. Linear Support Vector Classification

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 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

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

kernels. We use the same cross-validation setup as before.

We examine 4 SVM kernels [20]:

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

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.

4.4 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.

Classification accuracy 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 different species.

5 Feature Selection

For each method, we measure classification accuracy with an SVM model [20]. It has linear kernel, l1 regularization [18] and 10,000 maximum iterations. We examine 4 feature selection methods [4]:

1. Chi^2 [14]
2. Minimum Redundancy Maximum Relevance [6]
3. ReliefF [18]
4. Particle Swarm Optimization [11,12]

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

5.1 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.

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, and PSO performs best for the part.

5.2 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

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).

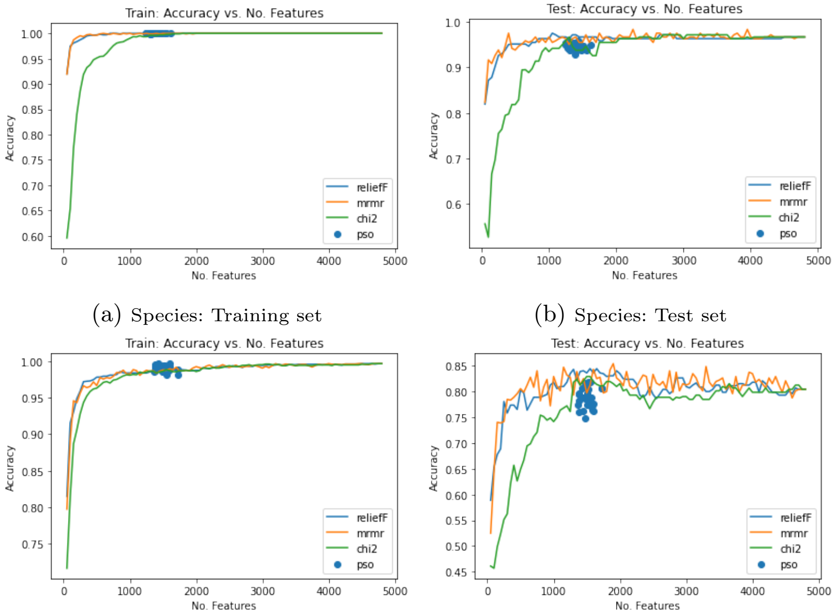
30 independent runs.

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 4c 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 4d 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.

5.3 Disucssion

Feature selection methods helped reduce dimensionality. We evaluated performance with an SVM classifier. 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 performance. PSO provides a combination of global and local searches. A search through a near-infinite combinatorial space



(a) Species: Training set

(b) Species: Test set

(c) Part: Training set

(d) Part: Test set

Fig. 4: Fish part dataset: classification accuracy for feature selection methods for a given k. 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.

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 support this, they also show higher test accuracy for fish species, than for fish parts. 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. Accuracy comparable to or better than the full dataset is possible with 21.8% of its features.

For the fish species dataset, we see high accuracy with 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 that 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.

6 Conclusions and Future Work

This paper has demonstrated an interpretable and effective method for fish oil analysis. The method can be understood, domain experts can understand the important features in the decision-making. 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 parts - 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.

LinearSVC applied to GC fish oil data can help to reduce waste in fish processing. This ensures more sustainable eco-friendly practices in fish processing. Waste reduction and repurposing is a rising tide that lifts all boats. Sustainable practices will leave plenty of fish in the sea, preserving resources for future generations.

References

1. Academy, K.: Gas chromatography (2013), https://www.khanacademy.org/science/class-11-chemistry-india/xfbb6cb8fc2bd00c8:in-in-organic-chemistry-some-basic-principles-and-techniques/xfbb6cb8fc2bd00c8:in-in-methods-of-purification-of-organic-compounds/v/gas-chromatography
2. Aizerman, M.A.: Theoretical foundations of the potential function method in pattern recognition learning. Automation and remote control **25**, 821–837 (1964), https://ci.nii.ac.jp/naid/10021200712/
3. Boser, B.E., Guyon, I.M., Vapnik, V.N.: A training algorithm for optimal margin classifiers. In: Proceedings of the fifth annual workshop on Computational learning theory. pp. 144–152 (1992), https://dl.acm.org/doi/abs/10.1145/130385.130401

4. Chappers, C.: charliec443/scikit-feature. https://github.com/charliec443/scikit-feature
5. Cortes, C., Vapnik, V.: Support-vector networks. Machine learning **20**(3), 273–297 (1995), https://link.springer.com/article/10.1007/BF00994018
6. Ding, C., Peng, H.: Minimum redundancy feature selection from microarray gene expression data. Journal of bioinformatics and computational biology **3**(02), 185–205 (2005), https://www.worldscientific.com/doi/abs/10.1142/s0219720005001004
7. Eder, K.: Gas chromatographic analysis of fatty acid methyl esters. Journal of Chromatography B: Biomedical Sciences and Applications **671**(1-2), 113–131 (1995), https://www.sciencedirect.com/science/article/pii/0378434795001426
8. Fix, E., Hodges, J.L.: Discriminatory analysis. nonparametric discrimination: Consistency properties. International Statistical Review/Revue Internationale de Statistique **57**(3), 238–247 (1989), https://www.jstor.org/stable/1403797
9. Hand, D.J., Yu, K.: Idiot's bayes—not so stupid after all? International statistical review **69**(3), 385–398 (2001), https://doi.org/10.1111/j.1751-5823.2001.tb00465.x
10. Ho, T.K.: Random decision forests. In: Proceedings of 3rd international conference on document analysis and recognition. vol. 1, pp. 278–282. IEEE (1995), https://ieeexplore.ieee.org/abstract/document/598994
11. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of ICNN'95-international conference on neural networks. vol. 4, pp. 1942–1948. IEEE (1995), https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=488968
12. Kennedy, J., Eberhart, R.C.: A discrete binary version of the particle swarm algorithm. In: 1997 IEEE International conference on systems, man, and cybernetics. Computational cybernetics and simulation. vol. 5, pp. 4104–4108. IEEE (1997), https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=637339
13. Köppen, M.: The curse of dimensionality. In: 5th Online World Conference on Soft Computing in Industrial Applications (WSC5). vol. 1, pp. 4–8 (2000)
14. Liu, H., Setiono, R.: Chi2: Feature selection and discretization of numeric attributes. In: Proceedings of 7th IEEE International Conference on Tools with Artificial Intelligence. pp. 388–391. IEEE (1995), https://ieeexplore.ieee.org/abstract/document/479783
15. Loh, W.Y.: Classification and regression trees. Wiley interdisciplinary reviews: data mining and knowledge discovery **1**(1), 14–23 (2011), https://doi.org/10.1002/widm.8
16. Musk, E.: 2020 annual meeting of stockholders and battery day: Tesla (Sep 2020), https://www.tesla.com/en_nz/2020shareholdermeeting
17. 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-acid-methyl-esters-fames/
18. Robnik-Šikonja, M., Kononenko, I.: Theoretical and empirical analysis of relieff and rrelieff. Machine learning **53**(1), 23–69 (2003), https://link.springer.com/content/pdf/10.1023/A:1025667309714.pdf
19. Schölkopf, B., Smola, A.J., Williamson, R.C., Bartlett, P.L.: New support vector algorithms. Neural computation **12**(5), 1207–1245 (2000), https://doi.org/10.1162/089976600300015565
20. Sklearn: 1.13. feature selection. https://sklearn.org/modules/feature_selection.html