Transactions on Engineering and Computing Sciences - Vol. 12, No. 1
Publication Date: February 25, 2024
DOI:10.14738/tecs.121.16525.
Pödör, Z., & Hekfusz, M. (2024). Comparing Feature Selection Methods on Metagenomic Data using Random Forest Classifier. Transactions on Engineering and Computing Sciences, 12(1). 175-187.

SERVICES FOR SCIENCE AND EDUCATION

Comparing Feature Selection Methods on Metagenomic Data using Random Forest Classifier

Zoltán Pödör
Eötvös Loránd University, Faculty of Informatics, Budapest, H-1117, Hungary

Máté Hekfusz
Eötvös Loránd University, Faculty of Informatics, Budapest, H-1117, Hungary

ABSTRACT

Feature selection (FS) as a data preprocessing strategy is an efficient way to prepare input data for various fields, such as metagenomics, where datasets tend to be very high-dimensional. The objectives of feature selection include creating lower dimensional and cleaner input data, along with building simpler and more coherent machine learning models. One of the promising applications of machine learning is in precision medicine, where disease risk is predicted using patient genetic data, which needs to be preprocessed with feature selection. In this article we provide a general overview of different feature selection methods and their applicability for disease risk prediction. From these, we selected and compared six different FS methods on two freely available metagenomic datasets using the same machine learning algorithm (Random Forest) for comparability. Based on the results of the individual FS methods, ensemble feature sets were created in multiple ways to improve the accuracy of Random Forest predictions.

Keywords: Feature selection, Metagenomics, Random Forest, Ensemble feature selection.

INTRODUCTION

Nowadays, machine learning algorithms and artificial intelligence are unavoidable parts of big data analysis. There are many scientific and practical problems where the number of independent variables (known as the features) is often so high that they are affected by what is known as the curse of dimensionality. As the number of dimensions or features increases, the amount of data needed to generalize the machine learning model accurately increases exponentially, and it is more difficult to achieve high accuracy values [1]. Also, with a huge number of features, learning models often tend to overfit, which may cause performance degradation on unseen, new data. Data of high dimensionality can also significantly increase the memory storage requirements and computational costs for data analytics [2]. This problem occurs in many areas, but it is a constant challenge in genomics, and especially when dealing with metagenomic data, which sequences entire microbial communities, resulting in huge, diverse, and noisy feature sets. [3]

The advancement of genetic sequencing and machine learning algorithms over the last 10-15 years has increased interest in precision medicine and genome data-based disease detection [4] [5]. The advent of high-throughput, next-generation sequencing (NGS) has brought a huge influx of metagenomic data, [6] and today there is an abundance of metagenomic samples

available in public databases. Metagenomics is a field devoted to understanding the workings of microbes by sequencing and analysing their genomes. To extract knowledge and patterns and to make decisions based on this sequenced metagenomic data, artificial intelligence and machine learning methods have been instrumental.

A typical (digitised) microbiome sample is made up of millions of raw sequences reads, which are converted with bioinformatics tools into a two-dimensional table (the rows are the samples, the columns are the microbial taxa) containing the relative abundances of each microbial taxa (the features in this case) present in the sample [7]. This table tends to be sparse with high noise, which makes for suboptimal training data for machine learning models [8]. Metagenomic data is extremely high-dimensional: the number of microbial features in each sample is orders of magnitude greater than the number of samples available for analysis [7] [8]. This number depends on the dataset, but it is usually on the order of tens of thousands [9].

Thus, to be able to apply machine learning approaches effectively, metagenomic data needs to be preprocessed: redundant and noisy features must be removed, and dimensionality must be drastically reduced before a dataset could be used to train an ML model. This preprocessing is known as feature selection (FS), and it has become an indispensable part of most bioinformatics pipelines [10]. Well-prepared input datasets are necessary conditions for effective and reliable data analysis.

This paper has two main purposes: on the one hand to compare several FS methods on the same databases with the same machine learning algorithm. On the other hand, to create, examine and compare ensemble feature sets based on the results of individual FS methods. Our research questions – connected to metagenomic data – were: (1) what type of FS methods give the best feature subset for machine learning, (2) do the different FS methods define similar feature subsets, (3) are the ensemble feature sets better than the result of individual selections' results? The remainder of this paper is organized as follows: Section 2 provides a literature review on the most important FS methods and their applications in metagenomics, as well as on our applied machine learning algorithm. Section 3 discusses the performance of the FS methods we selected, both individually and in ensemble, on the same input datasets. Section 4 answers our research questions and summarizes the main findings and implications of this study.

METHODS

Data preprocessing is the foundation of successful and reliable data analysis. Metagenomic databases pose several challenges, the most important one is their high dimensionality: even if the number of samples is not too large, the number of features is. This means that an appropriate preprocessing step is very critical for a high-quality analysis [11].

In this chapter we introduce the main types of the feature selection methodologies, their applications in metagenomics, and our one chosen machine learning algorithm, Random Forest (RF), which builds a model from the selected features. Our aim was to compare the result of the selected FS algorithms with the same machine learning algorithm (RF) in all cases for consistency.

Feature Selection

One of the main goals of the preprocessing step is to reduce the dimensionality and the complexity of a dataset, which is accomplished by feature selection. There are five main types of feature selection methods:

Basic Methods:

1. Filter methods: Features are selected on the basis of their scores in various statistical tests for their correlation with the dependent variable. Features are ranked by an evaluation criterion, and those above a certain threshold (usually chosen by the user) are selected as the feature subset, independently of the learning model. Because these evaluations are fast and independent of machine learning classifiers, filter methods are considered the simplest and least computationally intensive of the FS methods [12]. Filter methods can be univariate (testing each feature independently, for example Fischer-, t-, Mann-Whitney-test, Pearson, correlation) or multivariate (testing subsets of features simultaneously, for example Fast Correlation-based filter, Minimal-redundancy-maximal-relevance, Relief-based algorithms).
2. Wrapper methods: Unlike filter techniques, wrapper methods are invariably tied to a given ML classifier, as they use it to evaluate different combinations of features and select the subset that performs the best [12]. Their main issue is cost [10]: given the high-dimensional nature of metagenomic data, evaluating every possible subset is computationally infeasible, requiring search strategies to narrow down the options. Perhaps the best-known search strategies are forward selection and backward elimination.
3. Embedded methods: They integrate feature selection and ML model training into one step. During training, the ML algorithm automatically determines the importance of each feature. These methods can be considered a middle ground between filters and wrappers: like filters, they are reasonably fast, and like wrappers, they consider the characteristics of the classifier to achieve higher performance [12]. Random Forest and LASSO methods are typical examples of this type of FS.

Advanced Methods:

They Are Built from The Three Basic Types Mentioned Above

1. Hybrid methods: They implement different types of FS algorithms within one, multi-step sequential process, taking advantage of their different characteristics. The most intuitive way to construct a hybrid method is to start with a fast filter technique and then give its (lower dimensional) output to a wrapper or embedded method, reducing their higher computational cost while retaining their higher accuracy – the best of both worlds [12].
2. Ensemble methods: Ensemble methods also utilize multiple FS algorithms, but unlike hybrid techniques, they do not implement them step-by-step, but rather, in parallel. In an ensemble process, multiple FS algorithms are run on the dataset separately, each of which returns a subset of features. Then, these subsets are aggregated in some manner to find the final feature set. This aggregation can be a simple intersection or union of the individual subsets or some kind of weighting of each feature based on its position in each individual subset [12].

Application of FS Algorithms in Bioinformatics

We studied the bioinformatics literature to find feature selection methods effectively used for disease risk prediction from metagenomic and other genomic data. While the focus of this paper is metagenomics, feature selection of genomic data in general faces many of the same challenges: high dimension, sparsity, noise. Thus, we include papers and algorithms that deal with other forms of genomic data (such as micro-array and genotype data) as well to present a more comprehensive picture of the field.

Filter methods have proven enduringly popular in bioinformatics, with several different algorithms being used: Hacilar et al. [13] used the minimum redundancy – maximum relevance (mRMR) multivariate FS method on an Inflammatory Bowel Disease (IBD) dataset to find the subset of features most associated with the disease. Urbanowitz et al. [14] analysed the popular Relief family of multivariate FS methods on simulated genotype datasets, finding that they accurately detected two-way feature interactions.

Despite their higher cost, wrapper methods have also been used in genomics. He et al. [15] devised a novel wrapper FS algorithm based on the mRMR filter to predict genetic traits. Kavakiotis et al. [16] also created a new wrapper, Frequent Item Feature Selection (FIFS), which outperformed other FS methods in the informative marker selection task. Shen et al. [17] used the Boruta wrapper, paired with the Random Forest classifier, to find microbes from the gut microbiome that were important to predicting schizophrenia.

Studies that do not specify a feature selection step but use a classifier like Random Forest could be considered to be using an embedded method. More explicitly, Kumar & Rath [18] used Support Vector Machines (SVM) as an embedded way of feature selection, along with statistical filter methods, on leukaemia datasets. Sasikala et al. [19] proposed a new Genetic Algorithm (GA), integrating it with four different ML classifiers to produce a highly accurate model for breast cancer diagnosis.

Hybrid methods have recently become quite popular in the field, with some studies calling it the ‘best practice’ for feature selection [10] [12]. Jafari et al. [20] combined two univariate filters (Pearson correlation and information gain) and a multivariate filter (ReliefF) with a Genetic Algorithm wrapper to infer gene networks. Wang & Cai [21] analysed five different types of cancer with a two-step FS framework followed by an SVM classifier, manually confirming that the hybrid process selected near-optimal feature subsets.

Studies show that ensemble methods outperform single-algorithm methods in a variety of genomic tasks. Verma et al. [22] used a variety of filter and embedded methods – wrappers are rarely present in ensembles because of their high cost – to show that different FS algorithms selected different features from genetic data, and thus an ensemble method is needed for the best performance. Farid et al. [23] proposed an ensemble feature selection and clustering method specifically for high-dimensional genomic data, showing that it worked better on a Brugada syndrome dataset than non-ensemble alternatives. Sarkar et al. [24] combined no less than eight different FS methods into an ensemble process and devised an innovative

aggregation step that delivered high classification accuracy on breast cancer microRNA biomarkers.

The above-mentioned applications show the importance and inevitability of FS in genomic data analysis, including in metagenomic data, as a preprocessing step.

Random Forest

Random Forest is a machine learning classifier which combines the output of multiple decision trees to reach a single result. It conducts feature selection during training, making it an embedded FS algorithm as well. Its ease of use and flexibility have fuelled its adoption, as it handles both classification and regression problems [12]. It is highly data adaptive, which makes it well-suited for problems with "large p, small n" problems (where p is the number of features and n is the number of samples) and can account for correlation as well as interactions among the features [25].

RF is a widely used machine learning algorithm for classification tasks in metagenomic analysis, and it outperforms LASSO and SVM on metagenomics datasets for colorectal cancer, which is a disease we also analyse [26]. Degenhardt et al. [27] established that RF has been successfully applied on metagenomics datasets. In [28] the authors have found that RF showed better generalizability and robustness than KNN and SVM methods, making it suitable for use with high-dimensional data. Shen et al. [29] show examined RF, stochastic gradient boosting, and SVM methods on metagenomics data. They selected Random Forest due to its slightly better performance from a holistic perspective and capability of ranking the predictor variables.
These properties of RF and its popularity were the reasons we chose this learning model to examine the results of our selected feature selection methods.

RESULTS

During our literature review, we have identified several feature selection methods effectively used to identify diseases from genomic data, which we believe are well-suited to be used with metagenomic data given the similarity of the challenges. Based on this we have selected six for our experiments: Chi-squared (Chi2), Mutual Information (MI), Minimum Redundancy-Maximum Relevance (mRMR), Fast Correlation-Based Filter (FCBF), MultiSURF, and Random Forest. The first five are all filter methods, which are frequently used in ensemble feature selection because of their speed and simplicity, allowing several algorithms to be executed in tandem. Random Forest is an embedded method, meaning it conducts feature selection and model training in the same step. We use it not just for feature selection, but also as the machine learning classifier with which we test all the other methods, providing us an appropriate baseline. No wrappers were included as their computational cost tends to be too excessive for very high-dimensional datasets, and thus they rarely feature in ensemble algorithms.

We used two publicly available metagenomic datasets. The first dataset deals with Parkinson's disease (PD), with 366 gut microbiome samples from patients in the United States, as described in the study of Hill-Burns et al. [31]: 211 of these samples have the disease, while the other 155 are healthy controls. The second dataset, published by Zeller et al. [32], contains 182 samples related to colorectal cancer (CRC), with a good balance of 90 cancerous and 92 healthy samples.

The samples in both datasets have already been processed from raw reads into two feature abundance tables, which can directly be fed into feature selection algorithms. The PD table, processed during previous research, contains 10,631 features, each corresponding to a microbial taxon. The CRC table, processed and made available by Qu et al. [33], contains 18,170 features.

The feature selection algorithms we chose were drawn from various open-source libraries and implemented in Python. MI, Chi2, and RF were taken from the popular machine learning toolkit scikit-learn [34]. Accompanying their review of the Relief feature selection family, Urbanowicz et al. published a suite of the algorithms, which included MultiSURF. Finally, mRMR and FCBF algorithms were taken from separate open-source repositories (Available at https://github.com/smazzanti/mrmr under the MIT license).

Individual Performance on Feature Subsets

First, we examined how well each of our feature selection methods performed individually on feature subsets of various (but always small, as dimensionality reduction is one of the main goals of feature selection) sizes. Each method produces a ranking of every feature based on its own measure of relevance and importance. After a ranking was produced, we took the highest ranked features and reduced the dataset to just these features. We then divided the samples of this reduced dataset into train and test splits (80% and 20% of the data, respectively) and trained the Random Forest classifier with 10-fold cross-validation, recording mean accuracy and standard deviation for each experiment.

Overall, we conducted 240 experiments at this stage, testing all six of our algorithms on both datasets, with feature subsets ranging from size 10 to 200. The results can be found in Table 1 and Table 2. We also ran the full, unreduced datasets through Random Forest to get a baseline accuracy to compare against. We chose RF for this task partly because it also implicitly employs feature selection, and thus we believe it provides a higher baseline accuracy on metagenomic data than classifiers without FS. For the PD dataset, this resulted in 62.57% classification accuracy with a standard deviation of 4.02. For the CRC dataset, which appears to be easier to classify, the baseline was 73.04% accuracy with 8.03 standard deviation.

Table 1: Classification accuracy (Acc in %) and standard deviation (Std) values for the individual FS algorithms on various subsets of the Parkinson's dataset. Highest accuracy is highlighted in bold for each algorithm.

	RF		CHI^2		MUTUAL		MRMR		FCBF		MULTISURF	
FEATURES	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**
10	62.5	11.5	57.3	5.3	60.65	12.72	62.3	7.88	63.4	10.4	54.6	8.4
20	68.0	6.63	63.9	6.8	59.03	8.22	59.5	8.25	64.9	8.93	58.7	6.6
30	68.2	9.74	64.2	6.5	65.81	5.66	69.9	9.52	66.4	8.65	62.3	8.0
40	70.4	7.64	63.4	9.3	60.34	10.07	72.4	9.08	65.0	9.08	60.6	9.2
50	**71.5**	**7.88**	65.0	5.9	68.27	5.36	74.0	11.0	66.1	9.15	62.0	8.4
60	71.2	8.36	65.8	6.0	65.87	9.77	74.3	9.99	63.1	8	64.2	7.4
70	69.6	10.0	65.5	5.7	66.71	9.47	72.7	8.96	63.4	7.17	63.3	6.0
80	70.7	6.99	65.6	6.9	57.92	7	73.2	9.24	63.9	7.24	63.1	5.4

90	66.3	9.8	64.7	5.8	61.46	8.2	74.6	8.59	63.3	8.05	64.2	7.0
100	68.2	8.78	66.1	5.9	63.37	8.49	72.9	9.68	63.9	8.85	65.2	7.8
110	67.1	7.27	66.6	8.6	64	9.96	74.6	10.1	64.4	7.24	**65.2**	**7.9**
120	69.8	8.14	68.3	6.8	61.46	8.05	73.0	9.93	63.6	6.13	64.4	6.4
130	67.7	8.56	67.2	6.8	65.53	10.12	**75.6**	**9.15**	63.0	6.26	64.7	7.8
140	68.5	7.44	67.4	5.9	66.36	10.38	74.6	8.04	63.3	6.7	64.7	7.3
150	67.1	7.79	66.9	6.9	65.26	8.61	73.2	7.8	**67.7**	**7.3**	64.7	7.0
160	69.6	5.64	66.6	7.0	69.64	10.35	72.3	9.34	61.1	6.45	63.3	8.1
170	68.7	9.16	**68.5**	**7.4**	66.66	9.53	74.8	9.22	64.4	9.67	63.9	6.4
180	66.9	7.86	66.6	9.3	66.07	7.08	72.6	7.81	63.1	4.37	61.1	7.7
190	67.4	9.47	66.9	5.7	**70.68**	**10.77**	71.0	9.16	63.1	5.3	63.0	5.7
200	69.6	6.97	66.4	7.4	65.55	9.76	70.4	9.33	63.9	6.18	63.6	7.4

Table 2: Classification accuracy (Acc in %) and standard deviation (Std) values for the individual FS algorithms on various subsets of the CRC dataset. Highest accuracy is highlighted in bold for each algorithm.

	RF		CHI^2		MUTUAL		MRMR		FCBF		MULTISURF	
FEATURES	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**	**Acc**	**Std**
10	68.04	9.37	60.96	9.37	46.64	9.9	72.95	7.44	63.65	10.8	66.49	6.11
20	73.01	10.08	64.91	6.98	60.41	7.43	81.31	8.94	59.24	8.98	70.73	9.87
30	72.98	9.55	66.43	6.96	56.46	13.06	85.67	7.55	62.02	9.94	72.46	6.31
40	74.65	9.16	64.8	7.21	61.55	11.78	84.5	8.24	63.63	14.42	71.29	9.1
50	74.68	7.24	70.79	9.17	64.82	9.03	87.31	6.65	65.26	11.58	79.53	8.3
60	79.65	8.65	72.46	8.03	62.63	10.14	84.53	7	63.13	11.16	77.34	8.16
70	76.96	7.59	77.43	10.14	67.02	10.44	86.2	6.75	63.13	13.94	75.2	7.21
80	77.46	7.21	74.68	5.82	66.05	9.84	85.61	5.76	71.35	10.66	75.18	8.73
90	76.37	9.61	71.35	9.07	70.41	12.92	85.58	8	69.12	8.17	79.59	7.75
100	79.65	9	70.76	10.16	75.26	7.56	85.06	7.51	**71.9**	**8.55**	**80.15**	**6.22**
110	79.65	9	74.09	12.04	69.15	8.87	88.95	6.14	70.76	11.79	79.53	8.67
120	75.88	7.62	75.18	7.3	68.63	9.74	**90.01**	**6**	66.84	11.88	76.29	7.48
130	80.79	7.48	75.79	6.29	66.99	9.07	87.84	7.38	69.12	11.52	76.81	9.63
140	77.98	9.68	75.23	8.42	58.22	9.68	86.81	5.53	70.85	8.66	79.59	9.86
150	78.54	7.29	75.29	7.43	66.52	10.09	86.73	6.74	65.82	14.91	75.76	6.73
160	79.12	6.92	74.15	4.42	74.65	6.29	86.75	8.28	70.23	10.49	75.82	7.68
170	79.68	6.55	76.4	8.47	68.71	11.1	87.31	7.01	66.43	10.89	76.32	8.14
180	76.32	9.98	**78.54**	**8.82**	**77.37**	**6.97**	86.2	5.2	67.46	10.3	77.34	9.56
190	80.2	8.33	76.35	7.08	77.37	14.63	90	8.9	69.15	13.04	79.06	7.31
200	**81.87**	**7.44**	77.49	7.89	70.15	13.6	85.64	5.18	69.74	10.87	74.18	6.01

These individual results in Table 1 and Table 2 confirm that feature selection in general is an important preprocessing step, generally providing higher classification performance with drastically reduced feature counts, which also speed up model training. This performance increase, however, is dependent on several factors. Perhaps the most important is the dataset

itself: we can see that using same methods and the same parameters nonetheless resulted in significantly (10-15%) lower accuracy on the PD dataset than the CRC one, indicating that the former's samples are more difficult to classify. Another important factor is the number of features selected: unlike wrappers which determine the optimum number themselves, the filter and embedded methods we use require the user to make that choice. The tables show that each FS algorithm has a 'sweet spot', an ideal size for the feature set which provides the best performance, and that this sweet spot is different for each algorithm – meaning each method must be tested individually to find it.

The choice of the algorithm itself can make a substantial difference as well. Across the two tested datasets, we have identified mRMR as the best performer by a considerable margin, indicating that it is a powerful tool for selecting features from metagenomic datasets. We initially believed that our multivariate filters (FCBF and MultiSURF) would perform better than our univariate ones (Chi^2 and Mutual Info), given the complex nature of the data. And while MultiSURF delivered the second-highest accuracy on the CRC dataset, overall, we could not establish such a pattern, with FCBF actually performing the worst from all methods on that same dataset.

We have also observed high standard deviation values across algorithms and datasets, meaning that the individual classification accuracies during cross-validation varied widely. This is indicative of the instability of metagenomic datasets: there are subsets of our data for which the classifier performs much better (and subsets for which it performs much worse) than the average. For this study, we judged the performance of our FS algorithms by their accuracy, but their stability (measured by standard deviations) could be an equally valid metric.

Ensemble Feature Set Creation

After finding the optimal subsets for each feature selection algorithm, we took those subsets (50 features from RF, 170 features from Chi2, etc. for Parkinson's, for example) and aggregated them in two different ways. First, we took the union of the optimal subsets. Even though the CRC dataset had 7,500 more features than the PD dataset and the subset sizes for each dataset were different, the dimensions of the unions ended up being remarkably similar: 686 features for PD, 678 for CRC.

Next, we checked each feature in the union set to see how many subsets they were included in (in other words, how many FS algorithms ranked them high enough) and split them based on this number.

Table 3: Breakdown of how many subsets each union feature appears in, for both datasets.

Subsets	6	5	4	3	2	1
PD features	0	0	7	12	69	598
CRC features	0	4	7	36	93	538

Table 3 lists the breakdown of feature counts after this split and shows that while most features were only selected by one algorithm, there was enough of an overlap to construct consensus

sets, especially in the CRC dataset. We thus created one ensemble set with features selected by at least two algorithms, and another with features selected by at least three.

We tested all three of our aggregated sets with the same RF classifier that was used for the individual subsets.

Table 4: Feature counts and classification performance of three different ensemble sets on both datasets.

Dataset	Method	Features count	Accuracy	Std. dev.
PD	Union	686	68.81	8.59
	At least two	88	68.79	7.95
	At least three	19	66.65	9.82
CRC	Union	678	81.78	9.73
	At least two	140	79.12	7.18
	At least three	47	81.84	5.01

As Table 4 shows, all three ensemble sets had comparable performances on both datasets. It should be noted, however, that the consensus sets achieved their accuracy with a fraction of the features of the union set. For any given machine learning task, it is advised to have fewer features than samples to avoid the curse of dimensionality, which we discussed earlier – and the consensus sets achieve this. But it is especially important in metagenomics, where each feature is a microbial taxon which can be extracted and identified. To demonstrate this, we extracted three features from each dataset, each of which was selected by at least four of our six FS algorithms, and then identified the taxa they represented through the GreenGenes reference database (available at https://greengenes.secondgenome.com/). These microbial features are presented in Table 5 below.

Table 5: Operational Taxonomic Unit (OTU) code and taxonomical identification of some features selected by at least four FS algorithms.

Dataset	OTU	Taxonomy
PD	370287	c__Clostridia; o__Clostridiales; f__Ruminococcaceae; g__Faecalibacterium; s__prausnitzii
	336012	c__Bacteroidia; o__Bacteroidales; f__Bacteroidaceae; g__Bacteroides
	1078587	c__Clostridia; o__Clostridiales; f__Lachnospiraceae; g__Blautia
CRC	364048	c__Clostridia; o__Clostridiales; f__Lachnospiraceae; g__Blautia
	514523	c__Clostridia; o__Clostridiales; f__Ruminococcaceae; g__Faecalibacterium; s__prausnitzii
	301910	c__Clostridia; o__Clostridiales; f__Lachnospiraceae; g__Coprococcus

Identifying the exact connection (whether these features correspond to disease or health) is beyond the scope of this study, but our example shows that having few features makes it easier to pinpoint specific bacteria species and families that appear to be highly correlated with certain diseases [8] [24] which could improve their detection and treatment. Classification accuracy on the ensemble sets was similar or better than on the individual FS algorithm subsets, with mRMR being the one notable exception which performed considerably better than the

ensemble. This suggests that rather than augment the strength of mRMR (which is the idea behind ensemble feature selection), the findings of the other algorithms dragged it down, although it could equally be said that they elevated the worse-performing methods. Throughout our experiments, all algorithms had equal say in constructing the ensemble, which could help explain this phenomenon – in future research, a weighted voting scheme could be instituted which prioritises the features selected by higher-performing algorithms such as mRMR. This could also help in increasing stability, as for now, the standard deviations of the ensemble sets remained high, meaning that the ensemble frameworks did not make feature selection more stable.

CONCLUSION

In this study, we have explored feature selection in the field of bioinformatics, where it has become an increasingly crucial preprocessing step in a machine learning pipeline because of the massive amounts of metagenomic data available today, most of which is high-dimensional, sparse, and noisy. By conducting a thorough literature review, we have described the three basic categories of feature selection methods, naming specific algorithms used with genomic data. We have also looked at how combining them in a hybrid or ensemble framework has the potential to improve classification accuracy and stability. From the gamut of FS algorithms used in relevant studies, we have selected six and tested their performance extensively on two publicly available metagenomic datasets, one for Parkinson's disease, the other for colorectal cancer. One of our selected FS methods was Random Forest, which we used both as an ensemble FS algorithm and a machine learning classifier. Afterwards, we have created a new ensemble framework with these six algorithms, constructing ensemble sets from their highest-performing subsets through different aggregation methods, before comparing their accuracy with each other and with our individual results using the Random Forest classifier. We have also identified a few microbial features found through our ensemble framework, which appear to be highly correlated with our chosen diseases – showing that feature selection can also aid in precision medicine.

The main conclusion we can draw from our findings is that choices matter, and care must be taken at every step of the feature selection process. We have shown that the dataset itself affects the quality of the classification, with all FS methods performing much better on the CRC dataset than the PD one. The choice of FS algorithm also has a profound difference on classification performance: we have identified the minimal redundancy – maximum relevance (mRMR) algorithm as being particularly effective on our data, while another promising method, the Fast Correlation-Based Filter (FCBF) did not result in a meaningful improvement over the baseline. We have also found that despite the general potential of (and large interest in) ensemble methods, they do not always give better results than a single effective FS algorithm, at least in terms of our metric, which was classification accuracy. One cannot simply throw together different algorithms and hope for improvement; the participating methods and the way their subsets are aggregated must both be carefully chosen.

A voting scheme of the FS algorithms has the potential to create a high-performing ensemble set, and we will continue exploring in this direction in future research. We also aim to gather more datasets and more feature selection methods, so that we can help advance understanding

of feature selection for metagenomic data. Finally, to improve the generalization possibilities of the results, we also plan to include other classification methods in our experiments, like SVM, logistic regression, and gradient boosting.

ACKNOWLEDGEMENT

The research was supported by the project No. 2019-1.3.1-KK-2019-00011 financed by the National Research, Development and Innovation Fund of Hungary under the Establishment of Competence Centers, Development of Research Infrastructure Programme funding scheme.

References

[1]. Berisha, V., Krantsevich, C., Hahn, P.R. et al. Digital medicine and the curse of dimensionality. npj Digit. Med. 2021; 4:153, https://doi.org/10.1038/s41746-021-00521-5.

[2]. Jundong Li, Kewei Cheng, Suhang Wang, Fred Morstatter, Robert P. Trevino, Jiliang Tang, and Huan Liu. Feature Selection: A Data Perspective. ACM Comput. Surv. 2018;50:6, https://doi.org/10.1145/3136625.

[3]. Wooley J C, Godzik A, Friedberg I. "A Primer on Metagenomics," PLoS Computational Biology 2010; 6(2); https://doi.org/10.1371/journal.pcbi.1000667

[4]. Spiegel, A. M., and Hawkins, M. 'Personalized Medicine' to Identify Genetic Risks for Type 2 Diabetes and Focus Prevention: Can it Fulfill its Promise? Health Aff. (Millwood), 2012;31:43–49. doi:10.1377/hlthaff.2011.1054.

[5]. Manolio, T. A. Bringing Genome-wide Association Findings into Clinical Use. Nat. Rev. Genet. 2013;14:549–558. doi:10.1038/nrg3523.

[6]. Oulas A, Pavloudi C, Polymenakou p, Pavlopoulos G A, Papanikolaou N, Kotoulas G, et al., "Metagenomics: Tools and Insights for Analyzing Next-Generation Sequencing Data Derived from Biodiversity Studies," Bioinformatics and Biology Insights 2015;9:75-88, doi: 10.4137/BBI.S12462

[7]. Oudah M, Henschel A. "Taxonomy-aware feature engineering for microbiome classification," BMC Bioinformatics 2018;19:227

[8]. Zhu Q, Li B, He T, Li G, Jiang X. Robust biomarker discovery for microbiome-wide association studies. Methods 2020;173:44–51. DOI: 10.1016/j.ymeth.2019.06.012

[9]. Lai K, Twine N, O'Brien A, Guo Y, Bauer D. Artificial Intelligence and Machine Learning in Bioinformatics. Encyclopedia of Bioinformatics and Computational Biology, Elsevier, 2019, p. 272–286.

[10]. Tadist K, Najah S, Nikolov N S, Mrabti F, Zahi A. Feature selection methods and genomic big data: a systematic review. Journal of Big Data 2019;6:79. https://doi.org/10.1186/s40537-019-0241-0

[11]. Berrar D, Bradbury I, Dubitzky W. Avoiding model selection bias in small-sample genomic datasets. Bioinformatics. 2006;22:10:1245.12.

[12]. Pudjihartono N, Fadason T, Kempa-Liehr A W, O'Sullivan J M. A Review of Feature Selection Methods for Machine Learning-Based Disease Risk Prediction. Frontiers in Bioinformatics 2022;2. doi: 10.3389/fbinf.2022.927312

[13]. Hacilar H, Nalbantoglu O U, Bakir-Gungor B. Machine Learning Analysis of Inflammatory Bowel Disease-Associated Metagenomics Dataset. 3rd International Conference on Computer Science and Engineering (UBMK), 2018.

[14]. Urbanowicz R J, Olson R S, Schmitt P, Meeker M, Moore J H. Benchmarking relief-based feature selection methods for bioinformatics data mining. Journal of Biomedical Informatics 2018; vol. 85:168–188. https://doi.org/10.1016/j.jbi.2018.07.015

[15]. He D, Rish I, Haws D. Parida L. MINT: Mutual Information Based Transductive Feature Selection for Genetic Trait Prediction. IEEE/ACM Transactions on Computational Biology and Bioinformatics 2016; vol. 13:578–583. https://doi.org/10.48550/arXiv.1310.1659

[16]. Kavakiotis I, Samaras P, Triantafyllidis A. Vlahavas I. FIFS: A data mining method for informative marker selection in high dimensional population genomic data. Computers in Biology and Medicine 2017;vol. 90:146–154. https://doi.org/10.1016/j.compbiomed.2017.09.020

[17]. Shen Y, Xu J, Li Z, Huang Y, Yuan Y, Wang J, et al. Analysis of gut microbiota diversity and auxiliary diagnosis as a biomarker in patients with schizophrenia: A cross-sectional study. Schizophrenia Research 2018;vol. 197:470–477. DOI: 10.1016/j.schres.2018.01.002

[18]. Kumar M, Rath S K. Classification of microarray using MapReduce based proximal support vector machine classifier. Knowledge-Based Systems 2015; vol. 89:584–602.

[19]. Sasikala S, Balamurugan S A, Geetha S. A Novel Feature Selection Technique for Improved Survivability Diagnosis of Breast Cancer. Procedia Computer Science 2015;vol. 50:16–23. https://doi.org/10.1016/j.procs.2015.04.005

[20]. Jafari M, Ghavami B, Sattari V. A hybrid framework for reverse engineering of robust Gene Regulatory Networks. Artificial Intelligence in Medicine 2017;vol. 79:15–27. https://doi.org/10.1016/j.artmed.2017.05.004

[21]. Wang S, Cai Y. Identification of the functional alteration signatures across different cancer types with support vector machine and feature analysis. Biochimica et Biophysica Acta (BBA) - Molecular Basis of Disease 2018;vol. 1864:2218–2227. DOI: 10.1016/j.bbadis.2017.12.026

[22]. Verma S S, Lucas A, Zhang X, Veturi Y, Dudek S, Li B, et al. Collective feature selection to identify crucial epistatic variants. BioData Mining 2018;vol. 11:5 DOI: 10.1186/s13040-018-0168-6

[23]. Farid D M, Nowe A, Manderick B. A feature grouping method for ensemble clustering of high-dimensional genomic big data. Future Technologies Conference (FTC), 2016. doi: 10.1109/FTC.2016.7821620

[24]. Sarkar J P, Saha I, Sarkar A, Maulik U. Machine learning integrated ensemble of feature selection methods followed by survival analysis for predicting breast cancer subtype specific miRNA biomarkers. Computers in Biology and Medicine 2021;vol. 131:104-244. DOI: 10.1016/j.compbiomed.2021.104244

[25]. Chen X, Ishwaran H. Random forests for genomic data analysis, Genomics 2012;vol. 99:6:323-329, ISSN 0888-7543, https://doi.org/10.1016/j.ygeno.2012.04.003.

[26]. Gao Y, Zhu Z, Sun F. Increasing prediction performance of colorectal cancer disease status using random forests classification based on metagenomic shotgun sequencing data. Synthetic and Systems Biotechnology 2022; vol 7:574–585. https://doi.org/10.1016/j.synbio.2022.01.005

[27]. Degenhardt F, Seifert S, Szymczak S. Evaluation of variable selection methods for random forests and omics data sets. Brief Bioinform. 2019; vol. 20(2):492-503. doi: 10.1093/bib/bbx124.

[28]. Ai D, Pan H, Han R, Li X, Liu G, Xia LC. Using Decision Tree Aggregation with Random Forest Model to Identify Gut Microbes Associated with Colorectal Cancer. Genes (Basel) 2019; vol 10(2):112. doi: 10.3390/genes10020112.

[29]. Shen J, McFarland A G, Blaustein R A et al. An improved workflow for accurate and robust healthcare environmental surveillance using metagenomics. Microbiome 2022; vol 10. https://doi.org/10.1186/s40168-022-01412-x

[30]. Montesinos L, Montesinos López O A, Crossa A. Random Forest for Genomic Prediction. In: Multivariate Statistical Machine Learning Methods for Genomic Prediction. Springer, Cham. 2022; https://doi.org/10.1007/978-3-030-89010-0_15

[31]. Hill-Burns E M, Debelius J W, Morton J M, Wissemann W T, Lewis M R, Wallen Z D, et al. Parkinson's disease and Parkinson's disease medications have distinct signatures of the gut microbiome. Movement Disorders 2017;vol. 32:739–749. DOI: 10.1002/mds.26942

[32]. Zeller G, Tap J, Voigt A Y, Sunagawa S, Kultima J R, Costea P I, et al. Potential of fecal microbiota for early-stage detection of colorectal cancer. Molecular Systems Biology 2014;vol. 10:766. doi: 10.15252/msb.20145645

[33]. Qu K, Gao F, Guo F, Zou Q. Taxonomy dimension reduction for colorectal cancer prediction. Computational Biology and Chemistry 2019;vol. 83:107-160. DOI: 10.1016/j.compbiolchem.2019.107160

[34]. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, et al. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011;vol. 12:2825-2830.