Key Fragmentomics Features for Cancer Detection

An Analytical Approach to Identifying Essential Characteristics for Cancer Detection and Classification Using DNA Fragments from Blood Samples

David-Ștefan Pața

Supervisor(s): Bram Pronk, Daan Hazelaar, Stavros Makrodimitris, Prof.dr.ir. Marcel Reinders

EEMCS, Delft University of Technology, The Netherlands

A Thesis Submitted to EEMCS Faculty Delft University of Technology,
In Partial Fulfilment of the Requirements
For the Bachelor of Computer Science and Engineering
June 23, 2024

Name of the student: David-Ștefan Pața
Final project course: CSE3000 Research Project
Thesis committee: Prof.dr.ir. Marcel Reinders, Bram Pronk, Daan Hazelaar, Stavros Makrodimitris, Dr.ir. Johan Pouwelse

Abstract

Cancer represents a huge challenge in the medical world, necessitating early detection methods to improve treatment outcomes. The field of fragmentomics emerged as a promising option towards developing efficient non-invasive cancer diagnosis tools. By analysing the differences between the cfDNA fragments from blood samples of healthy patients and patients with cancer, this study aims to determine the most important fragmentomics features for cancer detection. The methods present in this work involve extracting features from the cfDNA fragments available in the experimental dataset, applying a pipeline of feature selection techniques that removes the redundant features, training and evaluating a logistic regression and random forest classifiers to differentiate between healthy and diseased samples, and finally extracting the feature weights from the trained models to understand which features contributed the most to the classification task. Filter-based variance thresholding and Correlation-based Feature Selection (CFS) were employed to refine the dataset. Independent t-test and the Mann-Whitney U test are used to calculate the relationship between the cancer and healthy samples. The Pearson correlation coefficient calculates the correlation between each pair of features. The classification performance of the two proposed models is assessed using the train/test split and the nested cross-validation techniques. The evaluation reveals that logistic regression constantly outperforms the random forest and that removing the redundant features increases the performance of both classifiers. Certain genomic bins, mostly on chromosomes 1, 7 and 8, contain significant features for the classification task. These findings suggest that understanding the importance of the fragmentomics features can lead to improved diagnostic tools such as cancer detection based on blood tests.

1 Introduction

Cancer is a widespread and complicated illness which entails considerable challenges for the diagnosed patients, but also for the doctors and researchers who try to develop efficient care and treatment options. Early diagnosis of cancer is crucial for determining a successful treatment and there is a lot of ongoing research into developing diagnostic tests that are capable of achieving this.

Recently, studying circulating DNA fragments present in the bloodstream of individuals with cancer emerged as a promising path in cancer investigation. When a cell dies, nucleic acids, including DNA, are released into the bloodstream. This also applies to cancer cells. These resulting cell-free DNA (cfDNA) fragments from tumour cells contain genetic variations that differ from those in healthy tissues. By analysing the differences between the cfDNA fragments of healthy and diseased samples, valuable information is obtained that can lead to improved cancer diagnostics [1]. The field of study that analyses the cfDNA fragments in blood is called fragmentomics. Fragmentomics involves looking at markers such as the size of the cfDNA fragment, which can be used to differentiate between cancer and healthy samples [2].

The use of cfDNA for cancer diagnosis has increased its popularity in healthcare because it reduces the need for invasive tests like tumour biopsies [1]. The big advantage of this approach is that the circulating free DNA fragments can be collected from blood samples. Blood samples are routinely collected from patients as part of regular medical check-ups. This eliminates the burden of complicated and unpleasant procedures such as biopsies, making the entire experience smoother and easier.

Diagram 1 summarizes the process of getting from the real DNA to DNA data objects. The DNA data objects are stored as .bam files, accompanied by their indices (.bam.bai files). These files represent the aligned paired reads of the DNA fragments ending up in the blood samples taken from patients. They also contain the DNA sequence of the reads and their position on the reference.

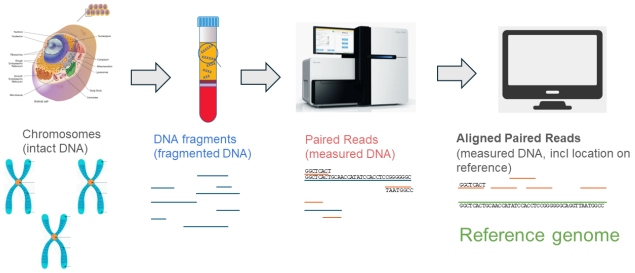

Figure 1: Diagram showing the process of obtaining DNA data objects from the actual DNA.

Literature has described multiple methods to extract features from cfDNA fragments for computational tasks such as classifying healthy or diseased samples [3]–[5]. However, understanding how these features determine the outcome of the classification remains incomplete. Identifying which features are the most important for the classification task represents a large knowledge gap worth studying since it can provide insights into underlying biology and valuable insights into the field.

The main focus of this Research Project is to answer the question:

Which fragmentomics features are most important for cancer detection?

The following sub-questions have been identified based on the given one:

1. Which methods are suited for determining the importance of fragmentomics features for cancer detection and what approaches can be used to select those features?

2. How do different classification algorithms perform in detecting cancer based on the selected fragmentomics features and what are their results?
3. What approach can be used to identify how much each selected feature contributed to the classification task?

To answer these questions, the objective is to apply feature importance and selection techniques to obtain the most important fragmentomics features from the available data. This subset of features will be used to classify the samples as healthy or diseased. With these features, early cancer detection and diagnosis can be significantly improved. Understanding the level of importance of the fragmentomics features can lead to improved diagnostic tools such as cancer detection based on blood tests.

2 Related Work

Literature has shown that fragmentomics features offer great insights into cancer detection, origin and treatment response [3], [6]. Scientific papers related to the research topic were read mainly to gather knowledge about the biological underpinnings needed to understand the data available and how exactly feature selection and importance play a role in obtaining the most important features. This helped in choosing an approach for gathering the features from the samples and provided insights into the possible classifiers that could be used for the classification task.

Cristiano et al. proposed the DELFI (DNA Evaluation of Fragments for Early Interception) approach for analyzing cfDNA fragmentation patterns to detect cancer [3]. Genome-wide fragmentation features were incorporated into a machine learning model (gradient tree boosting) to classify patients as being healthy or having cancer [3]. Their results suggested that fragment coverage was their major contributor to their classifier [3].

Moldovan et al. introduced the FrEIA (fragment end integrated analysis) score, a metric derived from investigating fragment ends of cfDNA [5]. This score assesses the presence of cancer-related cfDNA fragments in the blood [5]. cfDNA fragmentomics features, such as fragment size, fragment end diversity, and trinucleotide patterns are used to enhance the detection and classification of cancer [5]. Four supervised machine learning models (k-neighbours, logistic regression, random forest, support vector classifier) were trained using the mentioned features for the classification task [5].

In their work, *Renaud et al.* developed an unsupervised method named non-negative matrix factorization (NMF) for analyzing fragment length distributions in cfDNA [4]. This method helps determine the contributions of different cfDNA sources to a sample and identify fragment length signatures without prior knowledge of genomic alterations or sample information [4]. They demonstrate that accurate detection of various early-stage cancers is achieved by using multiple NMF components [4]. For the classification task, fragment length signatures were used with a linear Support Vector Machine to see if cancer and healthy samples could be separated based on their signature weights for a given number of signatures [4].

An et al. proposed a cfDNA ending preference-based metric for cancer diagnosis, whose performance was validated on multiple cancer datasets [7]. *Adalsteinsson et al.* apply ichorCNA (software that quantifies tumour content in cfDNA) to blood samples to demonstrate high concordance of cfDNA and metastatic tumour whole-exome sequencing [8]. In recent work, *Eledkawy et al.* use the eXtrem Gradient Boosting feature importance method to select the top ten most important features [9]. These features are then used to train a Light Gradient Boost Machine for the classification task [9].

Except for [3] which mentions fragment coverage as the major contributor to their classification task, and [4] which briefly indicates that using NMF can provide the contributions of cfDNA features, these works do not cover in detail the actual contributions of the features used for the classification task. This further emphasizes the knowledge gap presented in the introduction. This knowledge gap is what this work is trying to tackle.

3 Methodology

The method used throughout this research can be divided into several key components: using the available data for feature extraction, applying feature selection techniques, training and evaluating several classifiers for predicting cancer versus healthy samples and extracting the feature weights (coefficients) from the trained classifiers. Figure 2 visualizes the pipeline of the method used to conduct this research.

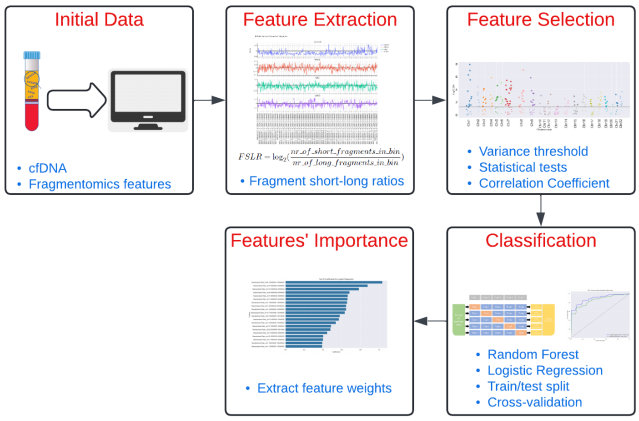

Figure 2: Pipeline describing the method used in this research.

Experimental dataset & setup

The sample datasets used throughout the research project were available through The Delft AI Cluster (DAIC) [1] as .bam files and their indices (.bam.bai files). The procedure used to obtain these files is shown in Figure 1. There are four sample types within the available data: .bam files representing blood samples taken from healthy patients (**control**) and patients with breast (**BRCA**), colorectal (**CRC**) and lung cancer (**LUAD**).

[1] https://daic.tudelft.nl/

The setup used for conducting this research involved code written in Python 3.11 [2] (which runs through Jupyter Notebooks [3]) and Linux command line commands [4] for using the DAIC cluster.

Feature extraction from the available samples

Gathering features from all DNA data objects was the first step required for conducting the experiments. The approach proposed by *Cristiano et al.* was reproduced for the feature extraction process. Every DNA data object was tiled into 5 mega base pair (5 million nucleotides) non-overlapping bins [3]. These bins were taken across all chromosomes within a sample. Chromosomes X and Y were excluded, ending with 22 chromosomes per sample. A short fragment is defined as having a length between 100-150 base pairs (bp), while a long fragment ranges from 151-220 bp [3]. Equation 1 was used to derive the fragment short-long ratios (FSLR) of the fragment lengths for every window within each chromosome [3]. This is done for all samples in the datasets. All ratios are standardized across the chromosomes within a sample using the z-score.

$$FSLR = \log_2\left(\frac{nr\_of\_short\_fragments\_in\_bin}{nr\_of\_long\_fragments\_in\_bin}\right) \quad (1)$$

Feature selection approaches

Among the extracted features, there might be irrelevant features that do not contribute to the classification task. Feature selection techniques are incorporated to exclude such features. An overview of the feature selection approaches used during this research can be visualized in the Diagram 3.
Inspired by [10] and [11], these feature selection approaches work as follows: A filter-based variance thresholding is applied on the initial set of features. This filters the quasi constant features which show the same value in almost all the observations in the dataset. Because of this, such features are not very useful for making predictions [10]. The threshold value is fixed to 0.01 [10], meaning that features with a lower variance than this threshold are removed.

Correlation-based Feature Selection (CFS) is then applied to the remaining features. CFS is a method used to select features strongly correlated with the target variable but with a low correlation between each pair of features. CFS aims to choose a subset of features that provide the most information about the target variable while reducing redundancy among features [12]. In the case of the provided data, the target variables are the labels assigned to each sample according to their type. Control samples are labelled with '0', while cancer samples with '1'.

Two statistical tests are used to compute the relationship between the features and the target variables: the Independent t-test and the Mann-Whitney U test (also known as the rank sum test). The independent t-test uses the t-statistic, representing a ratio of a difference in means to the standard error of that difference. It is used to determine if there is a

[2]https://www.python.org/

[3]https://jupyter.org/

[4]https://ubuntu.com/

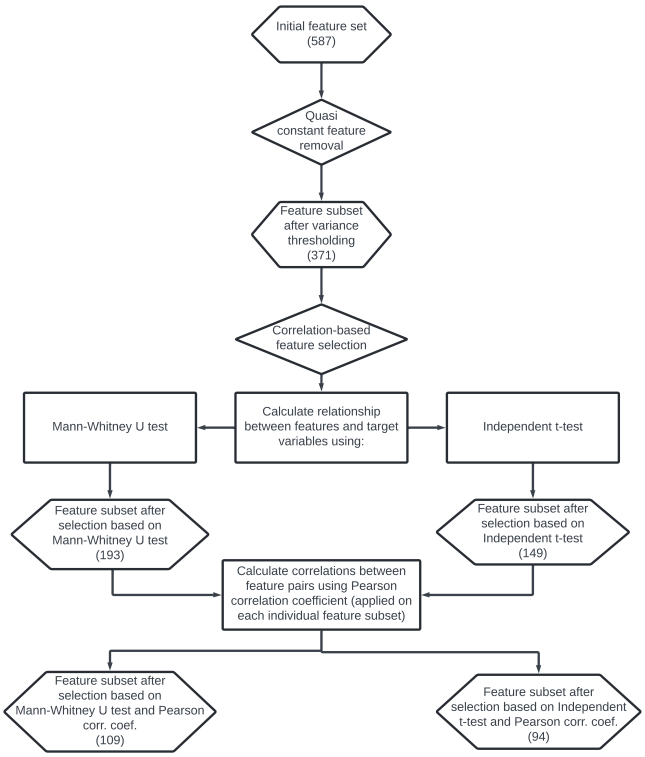

Figure 3: Flowchart describing the feature selection approaches used. The number at each hexagon's end represents the amount of features selected in each subset. The shown numbers are obtained after the selection techniques are applied to the training set derived after splitting the initial feature set into train/test sets.

significant difference between the means of the healthy and diseased groups. The Mann-Whitney U test assesses whether two groups of observations come from the same distribution. In both cases, features with a p-value lower than 0.05 were considered because these will discriminate well between the groups. These features have a statistically significant difference. Two selected feature subsets are obtained for each statistical test.

The newly obtained feature subsets are filtered once more, based on the correlation between each feature pair. For this, only the Pearson correlation coefficient was considered. If two features are strongly correlated, selecting both implies redundancy, thus it is viable to discard one of the features. However, features with low correlations might provide unique information and be important for the classification task since they offer different information. A correlation matrix is calculated for the present feature subset. One feature is discarded for each pair of features with an absolute correlation value higher than 0.85. This filtering is applied to each feature subset obtained after the statistical tests, resulting in two final sets of selected features.

Classification & Evaluation

Following the feature selection procedure, the following experiment involves the classification task. A logistic regression classifier and a random forest tree-based model were

used to classify all the available data as healthy or diseased. They were tested on the data using an 80/20 train/test split. The feature selection approaches detailed in the previous subsection were applied only to the training dataset. By doing this, the feature selection process is isolated from the test set, avoiding exposure to information from the test set and preventing data leakage. To assess the model's performance accurately, the test data must undergo the same filtering criteria applied during the feature selection for the training data. This guarantees that the model's predictions are based on the same features it learned from, enabling a fair evaluation.

All classifiers are trained with all the initial gathered features, but also with every subset of selected features obtained after each selection step described previously (the hexagons from Figure 3). The reason behind this is to compare the results of the feature selection steps in the classification task in between them, but also against the entire initial feature set. The models' performances are evaluated using the test set to ensure that classification models perform well on unseen data.

Separately, another random forest tree-based classifier was used for the classification task, but now the mentioned feature selection approaches were omitted. An attribute of the random forest model is that it builds a tree of features used as decision rules, which can be interpreted to be ranked on importance. This time, the model-dependent feature importance method provided by the random forest was used to assess the features' importance and how the selection of the most important ones affects the classification. Diagram 4 presents how the features were selected based on the features' importance obtained after training the Random Forest model with the initial feature set.

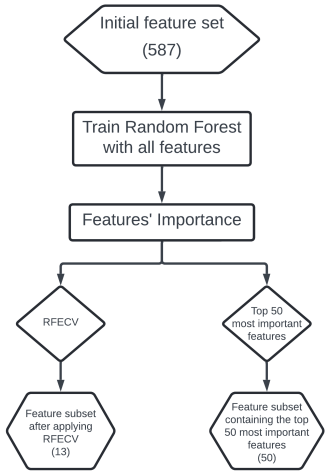

Figure 4: Flowchart describing the feature selection approaches used based on the features' importance derived from the trained Random Forest classifier model with all the initial features.

This random forest model was trained using the initial training set. After the importance of each feature was derived, this classifier was trained again with the top 50 most important features and with the features obtained from the Recursive Feature Elimination with Cross-Validation (RFECV) approach. RFECV is a future selection algorithm that iteratively removes the least important features of the model while using cross-validation to ensure the optimal number of features are selected for the best model performance. This random forest is retrained to observe if the model using the selected most important features outperforms the model using the entire feature set.

To ensure that the findings of the classification task do not depend on the specific split provided by the train/test split, nested cross-validation was also used to evaluate the performance of the classifiers. Nested cross-validation is treating model hyperparameter optimization as part of the model itself. This is evaluated within an inner 3-fold cross-validation procedure for assessing models for comparison and selection. A grid search is applied for each training dataset to identify the optimal set of model hyperparameters. Each hyperparameter configuration is evaluated using a separate 10-fold cross-validation on the specific train dataset (not the original full dataset), further splitting it into 10-folds [13]. This procedure was used to evaluate the performance of the logistic regression and random forest classifiers, using the same feature sets used during the train/test split. Figure 12 (from the appendix) visually explains the nested cross-validation procedure.

The parameters used for hyperparameter tuning are listed below.

- Logistic regression:
  - C: 0.01, 0.1, 1, 10, 100;
  - $penalty$: l1, l2;
  - $solver$: liblinear
- Random forest:
  - $n\_estimators$: 500, 1000, 10000;
  - max_depth: None, 10, 20, 30

Extracting the feature coefficients

The last step involves extracting the features' weights from the trained classifiers. These coefficients indicate the final importance of each feature in predicting the class labels. Furthermore, the chromosomes in which these features are located are also determined, resulting in a better understanding of the chromosomes containing the most important features. Therefore, these feature weights lead to the answer to the main research topic, in the sense that the features with the highest weights of the best-performing classifier represent the most important fragmentomics features for the task of cancer detection.

4 Results

The approach of generating features inspired by [3] led to 587 features for each sample type. Similar to [3], Figure 5 examines the FSLRs in windows throughout the genome. The bins with no short or long fragments lead to a ratio of 0 based on the used implementation, resulting in a high value after the ratios are standardized. These bins are excluded from the plot because they are considered outliers. The control samples had concordant fragmentation profiles because the majority had a negative or slightly positive ratio. The diseased ones however

had highly variable profiles with mostly decreased correlation to the median healthy profile. This plot indeed also reflects the results obtained in [3].

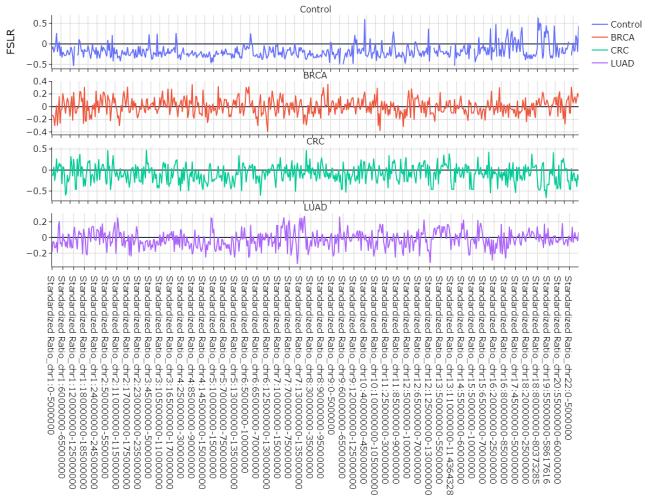

Figure 5: Fragmentation profiles for healthy individuals and patients with cancer obtained per cancer type using 5-Mb windows. The median healthy profile is indicated for the control samples. For patients with cancer, the Pearson correlation between individual profiles and the healthy median is shown.

The first step in the feature selection procedure, the quasi constant feature removal, leads to a subset of 371 selected features out of the 587 initial ones. After the first step of the CFS where the relationship between the features and the labels are calculated, the independent t-test filtering results in a selected subset of 149 features, whereas the rank sum test results in a selected subset of 193 features. Figure 6 visualizes the selection process based on the independent t-test. In this example, each dot represents a genomic bin. The plot's peaks show genomic bins with a statistically significant difference between cancer and healthy samples. The higher the peak, the stronger the statistical significance. All the points above the dashed horizontal line (representing the previously established 0.05 threshold) are selected by the selection pipeline during this step because they are considered statistically significant.

The last step of CFS involved calculating the correlation between each pair of features using the Pearson correlation coefficient. This ultimate selection step led to two final feature subsets: one of size 94 (t-test & Pearson) and one of size 109 (rank sum test & Pearson), as shown in Figure 3. After this final selection step, the remaining features are mostly not correlated (Appendix Figure 13). The strongly correlated ones are filtered out(Appendix Figure 14).

When considering the feature selection methods that involved the random forest trained separately, the top 50 most important features selected 50 features, whereas RFECV selected 13 (4).

The accuracy scores obtained using the train/test split are shown in table 1.

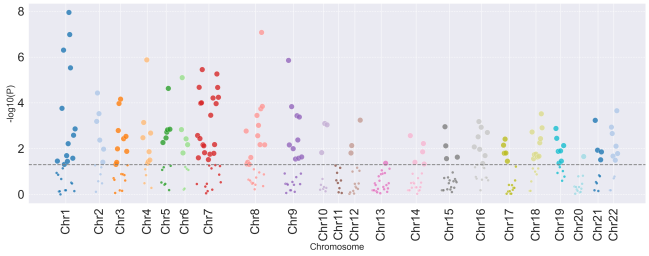

Figure 6: Manhattan plot that displays the genomic coordinates on the x-axis and the -log10 of the p-value obtained from calculating the independent t-test statistic between cancer and control samples on the y-axis. The ratios above the dashed line are selected during the selection process.

Table 1: Accuracy scores for the classifiers using train/test split for evaluating their performance. The columns represent the classifiers used, while the rows indicate each hexagon from Figure 3. These hexagons serve as a feature subset obtained along the feature selection process.

	Logistic Regression	Random Forest
All	78%	72%
Variance Filtering	80%	74%
t-test	76%	76%
Rank sum test	76%	68%
t-test & Pearson	82%	68%
Rank sum test & Pearson	82%	68%
Top 50 most important features	-	66%
RFECV	-	68%

Table 2 indicates the performances of the classifiers when they were evaluated using the nested cross-validation technique.

Table 2: Accuracy scores for the classifiers using nested cross-validation for evaluating their performance. The average of the accuracy scores of every fold of the cross-validation is illustrated. The columns represent the classifiers used, while the rows indicate each hexagon from Figure 3. These hexagons serve as a feature subset obtained along the feature selection process.

	Logistic Regression	Random Forest
All	81%	74%
Variance Filtering	80%	74%
t-test	83%	76%
Rank sum test	81%	74%
t-test & Pearson	81%	79%
Rank sum test & Pearson	79%	77%

Receiver Operating Characteristic (ROC) curves illustrate the performance of a classifier model at different threshold values. The curve plots the True Positive Rate (TPR) against the False Positive Rate (FPR). The Area under the ROC curve (AUC) provides an aggregate measure of performance across all possible classification thresholds. Figure 7 visualizes the performance of the classifiers when the train/test split is used. Similarly, Figure 8 shows the performance of the classifiers when using nested cross-validation.

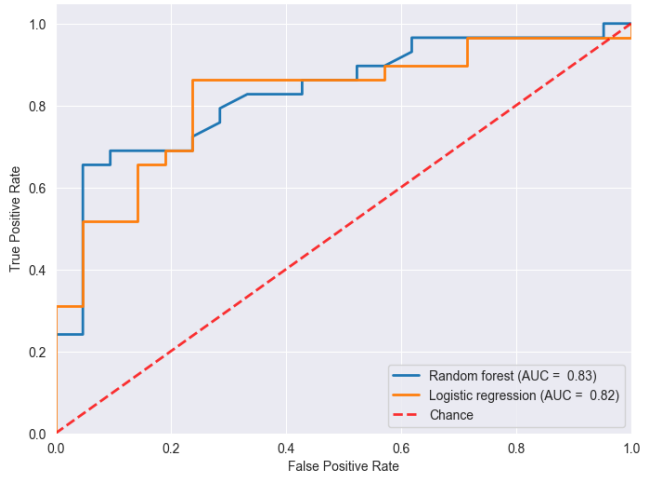

Figure 7: ROC curve illustrating the performance of the two classifiers when train/test split is used for evaluation.

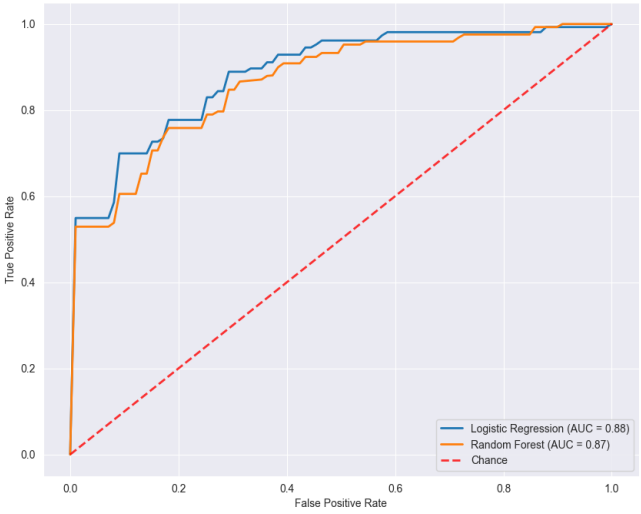

Figure 8: ROC curve illustrating the performance of the two classifiers when nested cross-validation is used for evaluation.

In both plots, the ROC curves show the performance of the classifiers trained with the feature subsets that provided the best accuracy scores. In Figure 7, the logistic regression was trained with the subset of features obtained after applying the t-test and correlation filtering, while the random forest was trained with the feature subset obtained after the t-test filtering. In the case of the second curve, logistic regression was trained using the subset obtained after the t-test filtering. The random forest was trained using the subset obtained after selecting the features based on the t-test and correlation.

The classifiers trained with the feature subsets that provided the highest accuracy scores are chosen for extracting the features' coefficients used during the training. Figure 9 displays the weights of the best 20 features for the logistic regression classifier, while Figure 10 displays the 20 most important features for the classification task that involved the random forest. These results are shown for the classifiers trained using the train set obtained after the train/test split with their corresponding feature subset mentioned in the plots above.

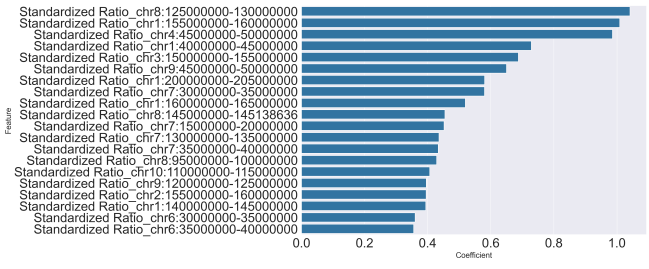

Figure 9: The coefficient values of the 20 features that contributed the most to the classification task using logistic regression. For this setting, the classifier was trained with the feature subset obtained after applying t-test and correlation-based filtering.

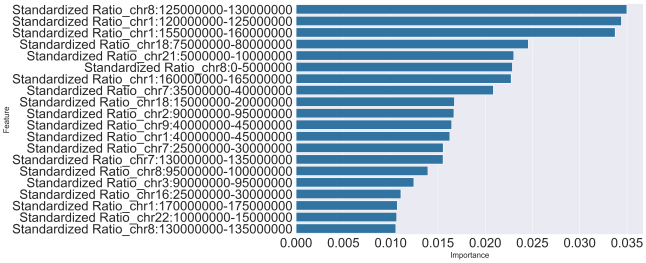

Figure 10: The importance of the 20 features that contributed the most to the classification task using random forest. For this setting, the classifier was trained with the feature subset obtained after applying t-test filtering.

5 Discussion

The logistic regression classifier outperforms the random forest classifier in all scenarios. When the data is split into train/test sets, the logistic regression classifier performs the best when trained with the feature subset obtained after selecting the features given by any of the statistical tests used combined with the Pearson correlation coefficient filtering. The resulting accuracy score is 82%. This result resembles the one obtained by [5] using the logistic regression classifier. This outperforms the classifier using all the initial features, but also the subsets of features that contained more features. Thus, while discarding features, the classification task is improved, meaning that irrelevant features are removed. Evaluating the performance of the logistic regression with the nested cross-validation approach provides the best score when the classifier uses the features selected by the t-test filtering. This yields an accuracy score of 83%. The fold that provided the best score used *C:10, penalty: 'l2'* and *solver: 'liblinear'* as parameters from the trained grid search, which resulted in a 92% accuracy score. The worst-performing fold achieved a 76% accuracy using *C:1, penalty: 'l1'* and *solver: 'liblinear'* from the grid search.

The best performance obtained by the random forest with the train/test split is when the subset of features obtained by applying the independent t-test filtering is used for training the classifier. The resulting accuracy score is 76%. This outperforms the classifier using all the initial features but does not provide better results than after applying the feature-feature correlation filtering, which lowers the accuracy score to 68%. This means that too much relevant information is lost after this filtering step in the random forest scenario. Evaluating the performance of the random forest with the nested cross-validation approach provides the best score when the classifier uses the features selected by the t-test and correlation-based filtering. This yields an accuracy score of 79%. The fold that provided the best score used *max_depth: None* and *n_estimators: 500* as parameters from the trained grid search, which resulted in a 92% accuracy score. The worst-performing fold achieved a 68% accuracy using *max_depth: None* and *n_estimators: 500* from the grid search.

Using the selected features based on the importance metrics generated by the random forest when trained with all initial features does not provide better results than the classifier that uses all samples. It is however important to mention that when using the subset obtained after applying RFECV, the performance drops from 72% to 68%. Still, the number of features used during the classifier's training drops from 587 to 13. This represents a huge trade-off, and the importance of these features is already established since they were chosen as the most important ones by the RFECV approach based on the random forest classifier training. Similarly, using the 50 most important features to train the random forest decreases the model's accuracy from 72% to 66%.

The first ROC curve (7) indicates that both models have an AUC above 0.8, which denotes good performance. With a slightly higher AUC (0.83) compared to the logistic regression model (0.82), the random forest has a marginally greater ability to distinguish between positive and negative classes. In the second ROC curve (8), both models have an AUC close to 0.9, showcasing a good performance for both classifiers. The blue curve (logistic regression) generally lies slightly above the orange curve (random forest). This depicts an overall better performance in terms of TPR at various thresholds of FPR.

The nested cross-validation provides higher accuracy scores for most of the feature subsets obtained along the selection pipeline that are used to fit both classifiers. This procedure considerably increases the performance of the random forest classifier. For example, when train/test split is used to evaluate the model's performance, the random forest scores a 68% accuracy when trained with the feature subset derived after the t-test and correlation-based filtering. In the same scenario, nested cross-validation provides a 79% accuracy score. The big advantage of nested cross-validation compared to the classic k-fold cross-validation approach is that the hyperparameter optimization is performed in a separate cross-validation, within the cross-validation used to evaluate the model's performance. Using the same cross-validation for both tuning the parameters and assessing the model's performance can lead to an optimistically biased evaluation of the model's performance. Nested cross-validation overcomes this bias.

Analyzing Figure 6 can offer insights into which chromosomes contain more important bins towards the classification task. Since the plot's peaks show genomic bins with a statistically significant difference between cancer and healthy samples, looking at the chromosomes containing high peaks can lead to the bins that contributed most to the classification task. Although most chromosomes contain peaks, interpretations can still be derived. Some genomic bins within chromosome 1 form the highest peak throughout the entire genome. Similarly, chromosomes 7 and 8 contain groups of high bins. Looking further into these chromosomes, Figures 9 and 10 reinforce these observations. Within the top 20 features that contributed the most to both classifiers, chromosome 1 has the most bins (five). This is followed by four bins from chromosome 7 and three bins from chromosome 8 in the logistic regression case. For the random forest model, the second-most bins come from chromosome 8 (four), followed by the number of bins from chromosome 7 (three). Within these bins, there are seven that coincide and are found in both groups of features that contributed the most for the two classifiers. Figure 11 shows these features.

Chr1_bin[40000000-45000000]
Chr1_bin[155000000-160000000]
Chr1_bin[160000000-165000000]
Chr7_bin[35000000-40000000]
Chr7_bin[130000000-135000000]
Chr8_bin[95000000-100000000]
Chr8_bin[125000000-130000000]

Figure 11: Bins whose values are within the 20 most important features in both classifiers.

The two numbers within the brackets represent the start and end position of the bin. The ratio values within these bins can be considered valuable fragmentomics features for cancer detection. Looking at the chromosomes that do not offer many significant differences between the healthy and diseased samples, chromosomes 12, 13 and 20 stand out, as they don't show any peaks in the Manhattan plot (6). This is backed up by the fact that the lists of 20 features that contributed the most towards the classification tasks do not contain any bin from these chromosomes (9, 10).

The numbers in Flowcharts 3 and 4 showing the size of the feature subsets after each filtering step are obtained when each selection procedure is applied to the training set obtained from the train/test split on the initial dataset. After each filtering step, the total of selected features differs for every nested cross-validation procedure fold. This is due to the variance that differs for each feature when a fold of distinctive features is established. Thus, this leads to different numbers across all filtering steps, since the variance thresholding is applied at the beginning of the feature selection pipeline.

Limitations

The conducted research presents some limitations. The features used during the experiments are extracted based on a single extraction approach. Thus, the analysis of the obtained results is limited to the extraction process used.

The features are standardized per sample when extracted from the samples (according to the [3] approach) based on the implementation used. Another approach could be to standardize the features per each bin. This would imply the removal of the variance filtering from the feature selection pipeline since all the features would have variance 1. This could lead to different results and a distinctive interpretation would be needed.

Given this work, a conclusion about the features' importance in the context of cancer detection can only be derived based on manually analysing the plot that shows the statistical difference between cancer and healthy samples (6) and the plots that show how much each feature contributes to the classification tasks (9 and 10). An automated pipeline that evaluates the features' importance would be a valuable addition to the field. Nevertheless, the results achieved in this work provide enough evidence towards a conclusion about the importance of features in the context of cancer detection. However, these results are obtained from a purely computer science perspective. Thus, one must be cautious about their interpretation. It is highly recommended that other researchers with a strong background in bio-informatics validate these results to confirm their validity.

Future work

The work presented in this paper can be further improved. The first improvement would represent having a bigger dataset of samples. At the moment, a total of 248 samples were used throughout the conducted experiments. A higher number of samples would ensure more realistic results for the classification task. A further improvement in this regard would be balancing the number of samples from each type. During this research process, 103 control samples, 47 BRCA samples, 23 CRC samples and 75 LUAD samples were used. The discrepancy between the amount of samples could lead to biased results towards the healthy samples, which is not ideal. Having a similar number of samples for each type would lead to more fair results and the risk of bias would disappear.

Extracting multiple feature types could potentially lead to improved results. Different approaches from literature for gathering features could be considered for this task. Using these approaches will generate new features. These new features can be added to the existing feature set as a feature engineering step. Expanding the feature set with other features of distinctive types can boost the classification performance.

Finally, using a different classifier than the ones already proposed (like a Neural Network) could further improve the obtained results. Combining multiple classifiers and using them together to classify the data can also be considered as an option.

6 Responsible Research

The work presented in this paper aims to respect responsible research and ethical practices.

The study involves real patient data. The available data samples are anonymous, and no personal information about the patients is available. The patients have provided informed consent for their samples to be used for scientific purposes since their blood samples were collected willingly. The results of this work will be used responsibly to improve the diagnostic techniques and not for any discrimination.

The research presented throughout this paper is reproducible since it is based on real data, which can be made available based on request. Furthermore, multiple other .bam files are available publicly, if enlarging the dataset is needed. All the implementations used to obtain the results and run the experiments explained in this paper are available at [14]. Following the steps mentioned in the methodology Section (3) can lead to full reproducibility of the experimental setup.

7 Conclusion

This research paper aimed to provide insights into analyzing the key fragmentomics features used in classifying healthy and cancer samples. Feature extraction from DNA data object representations of cfDNA fragments found in patients' blood samples was the first step needed in this process. Feature selection techniques were used to prove that the classification task can be improved while reducing the number of features. Finally, extracting the weights of the features used in the training process of the classifiers helped in deriving an answer to the initial research question. That is, the features with the highest coefficient values represent the most important fragmentomics features used in cancer detection using blood.

References

[1] H. Wilson, *Fragmentomics – the future of cfdna testing?* Accessed: 2024-06-08, 2023. [Online]. Available: https://www.phgfoundation.org/blog/fragmentomics-the-future-of-cfdna-testing/.

[2] P. Jiang, K. Sun, W. Peng, *et al.*, "Plasma dna end-motif profiling as a fragmentomic marker in cancer, pregnancy, and transplantation," *Cancer Discovery*, vol. 10, no. 5, pp. 664–673, 2020.

[3] S. Cristiano, A. Leal, J. Phallen, *et al.*, "Genome-wide cell-free dna fragmentation in patients with cancer," *Nature*, vol. 570, no. 7761, pp. 385–389, Jun. 2019, ISSN: 1476-4687. DOI: 10.1038/s41586-019-1272-6. [Online]. Available: https://doi.org/10.1038/s41586-019-1272-6.

[4] G. Renaud, M. Nørgaard, J. Lindberg, *et al.*, "Unsupervised detection of fragment length signatures of circulating tumor dna using non-negative matrix factorization," *eLife*, vol. 11, D. Weigel and A. Thierry, Eds., e71569, Jul. 2022, ISSN: 2050-084X. DOI: 10.7554/eLife.71569. [Online]. Available: https://doi.org/10.7554/eLife.71569.

[5] N. Moldovan, Y. van der Pol, T. van den Ende, *et al.*, "Multi-modal cell-free dna genomic and fragmentomic patterns enhance cancer survival and recurrence analysis," *Cell Reports Medicine*, vol. 5, no. 1, 2024.

[6] P. Peneder, A. Stütz, D. Surdez, *et al.*, "Multimodal analysis of cell-free dna whole-genome sequencing for pediatric cancers with low mutational burden," *Nature Communications*, vol. 12, May 2021. DOI: 10.1038/s41467-021-23445-w.

[7] Y. An, X. Zhao, Z. Zhang, *et al.*, "Dna methylation analysis explores the molecular basis of plasma cell-free dna fragmentation," *Nature communications*, vol. 14, no. 1, p. 287, 2023.

[8] V. A. Adalsteinsson, G. Ha, S. S. Freeman, *et al.*, "Scalable whole-exome sequencing of cell-free dna reveals high concordance with metastatic tumors," *Nature communications*, vol. 8, no. 1, p. 1324, 2017.

[9] A. Eledkawy, T. Hamza, and S. El-Metwally, "Precision cancer classification using liquid biopsy and advanced machine learning techniques," *Scientific Reports*, vol. 14, no. 1, p. 5841, 2024.

[10] P. Maha, *Feature selection techniques*, Accessed: 2024-06-04, 2024. [Online]. Available: https://github.com/maha-prathamesh/Feature-Selection-Techniques.

[11] N. Azaria, *Feature importance: 7 methods and a quick tutorial*, Accessed: 2024-05-17, 2024. [Online]. Available: https://www.aporia.com/learn/feature-importance/feature-importance-7-methods-and-a-quick-tutorial/.

[12] S. Shaikh, *Correlation-based feature selection in a data science project*, Accessed: 2024-05-28, 2024. [Online]. Available: https://medium.com/@sariq16/correlation-based-feature-selection-in-a-data-science-project-3ca08d2af5c6.

[13] J. Brownlee, *Nested cross-validation for machine learning with python*, Accessed: 2024-06-18, 2024. [Online]. Available: https://machinelearningmastery.com/nested-cross-validation-for-machine-learning-with-python/.

[14] D. Peta, *Dpeta: Detection of cancer using blood*, Accessed: 2024-06-09, 2024. [Online]. Available: https://gitlab.ewi.tudelft.nl/cse3000/2023-2024-q4/Reinders_Pronk_Hazelaar/dpeta-Detection-of-cancer-using-blood.

[15] J. Repossi, *Tutorial - cross validation & nested cv*, https://www.kaggle.com/code/jacoporepossi/tutorial-cross-validation-nested-cv, Accessed: 2024-06-19, 2023.

Appendix

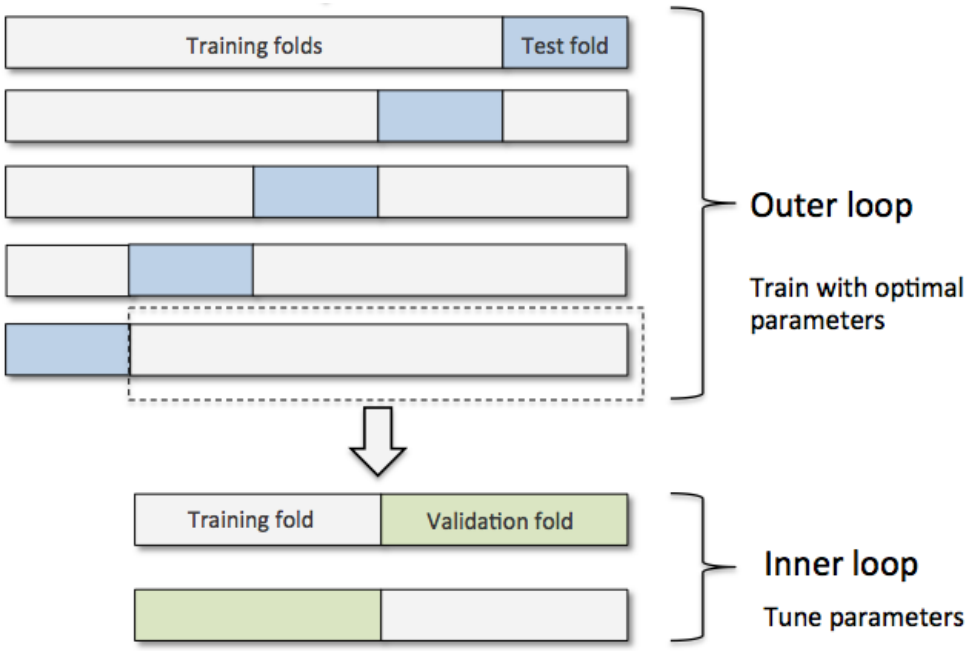

Figure 12: Explanation of the nested cross-validation procedure. [15]

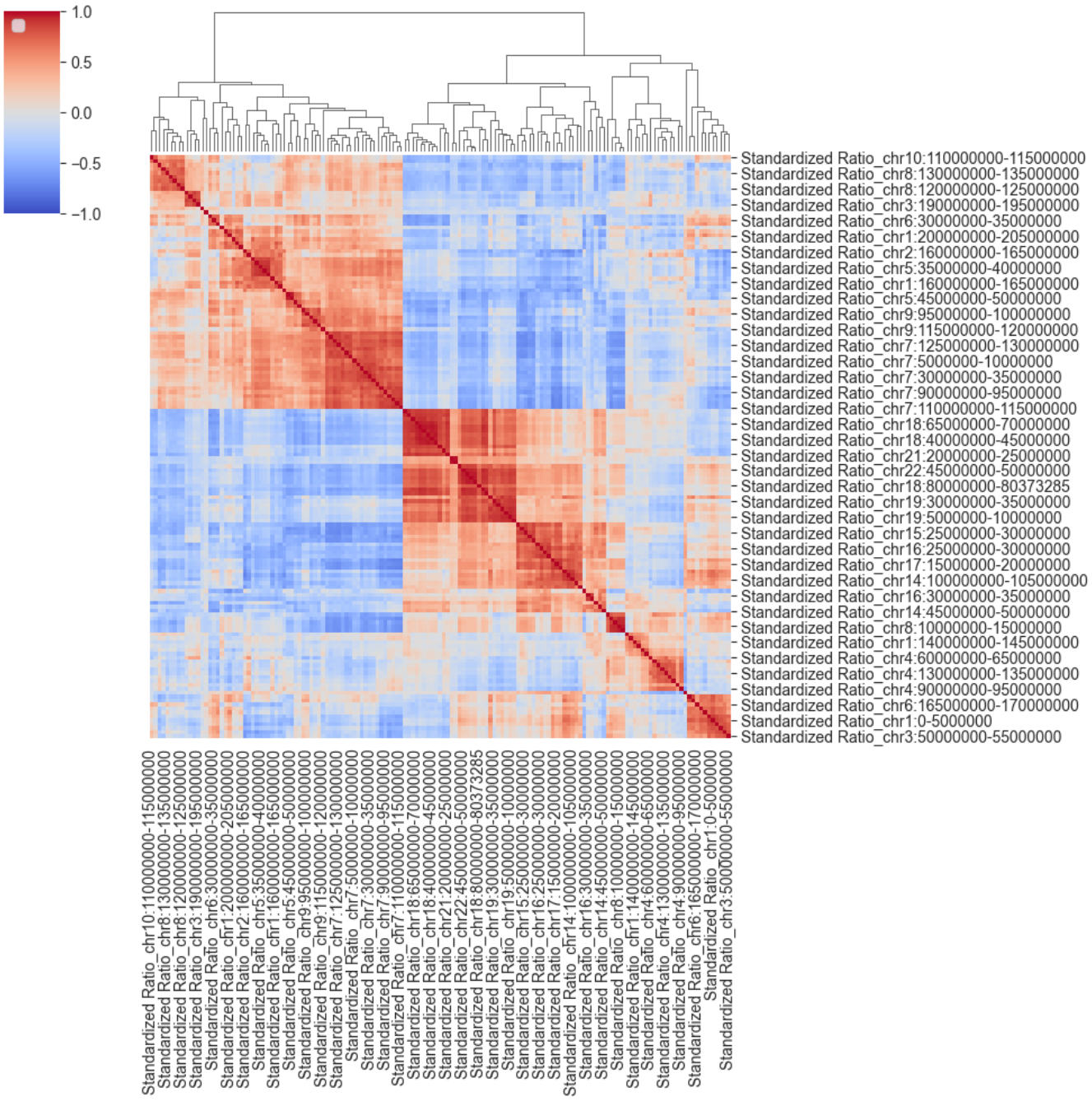

Figure 13: Heatmap showing the correlation between features before using correlation-based filtering for selecting a subset of features. The features are clustered based on their correlations.

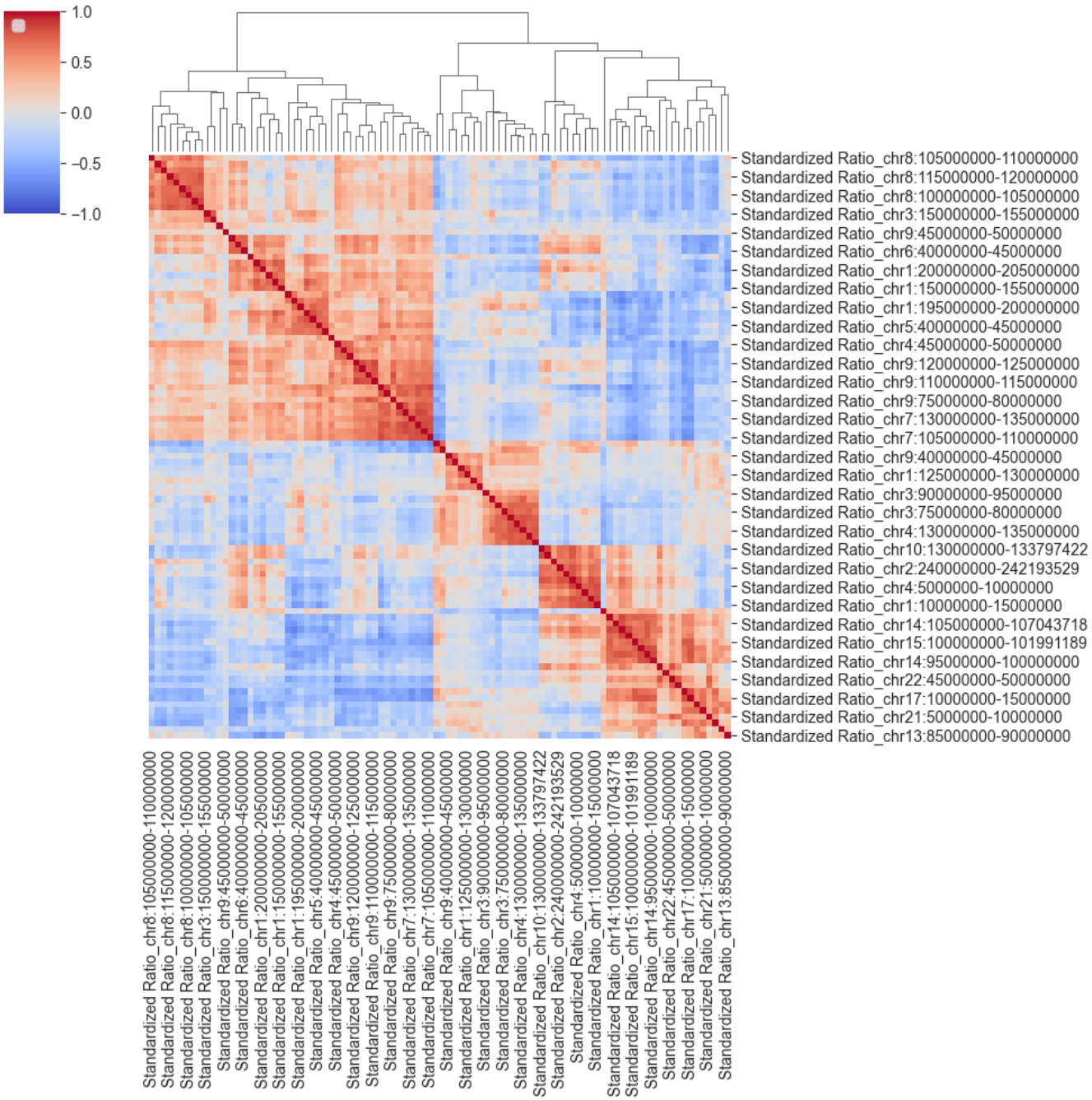

Figure 14: Heatmap showing the correlation between features after correlation-based filtering is applied to select the not strongly correlated features. The features are clustered based on their correlations.