Advancing Feature Selection Techniques for Supervised Machine Learning within Functional Genomic Experiments by Means of Overlapping Analysis

Anusa Suwanwong

A thesis submitted for the degree of
Doctor of Philosophy

School of Mathematics, Statistics and Actuarial Science
University of Essex

Date of submission for examination: August 2025

The work submitted in this thesis is the result of my own investigation, except where otherwise stated. It has not already been accepted for any degree, and is also not being concurrently submitted for any other degree.

Dedicated to

my beloved parents,

Sunthorn & Kamnueng

Acknowledgements

I would like to express my sincere gratitude to my supervisors, Dr. Osama Mahmoud and Dr. Andrew Harrison (Harry), for their invaluable support and guidance throughout my PhD studies and research. I am appreciative of their patience, motivation, enthusiasm, and immense knowledge. I am deeply thankful for the consistent encouragement they provided and for their unwavering belief in my abilities. I could not imagine having better supervisors for my PhD study. My heartfelt thanks go to both Osama and Harry for their trust and confidence in me.

There are many other people who supported me throughout my PhD. I owe my deepest gratitude to Dr. Hüseyin Yıldırım for his help and suggestions whenever needed. Thanks, Hüseyin for your efforts to explain things clearly and simply. Thanks to my Supervisory Panel Chair Dr. Yanchun Bao who gave me valuable feedback and suggestions. Thanks to all the staff at School of Mathematics, Statistics, and Actuarial Science (SMSAS), University of Essex for their kindness and support.

I am grateful to the Royal Thai Government Scholarship, Princess of Naradhiwas University, and the Royal Thai Embassy in London for offering me great opportunity, supporting and sponsoring my PhD research. I equally thank Dr. Noodchanath Kongchouy, and Dr. Attachai Ueranantasun whose support will be remembered always.

I wish to thank my entire family for their love, care, understand, and support. I particularly thank my parents, Sunthorn Suwanwong and Kamnueng Suwanwong, who have been a great source of motivation. Thanks also goes to friends for their love and emotional support.

Last but not least, I would like to express my heartfelt gratitude to Fahim for his invaluable partnership, friendship, steadfast support, and willingness to engage in discussions about the topics addressed within the thesis.

Abstract

Microarray technology enables the simultaneous measurement of tens of thousands of genes (features) with a small number of tissue samples (observations). This common characteristic of high dimensionality has a great impact on the classification tasks, since most genes are noisy, redundant or non-relevant.

A statistical learning approach aims at understanding and modeling complex datasets. Given a set of training data, its primary goal is to create a model that captures the relationship between a set of input features and the corresponding response in a predictive manner. Therefore, applying classification methods to microarray data is a crucial task which helps reduce dimensionality as well as categorise biological samples into distinct classes, such as different stages of a disease.

The prediction accuracy and interpretability of a model can be improved when the learning process is conducted using only the selected informative features. Two novel statistical methods are proposed; 3-class Proportional Overlapping Scores (3cPOS) and multiple Proportional Overlapping Scores (mPOS). Both methods exploit overlapping analysis to measure the level of overlap between different expression intervals, resulting in 3cPOS and mPOS scores. These scores help identify the informative genes (features) of three and multiple classes. Smaller 3cPOS and mPOS scores indicate a higher discriminative capability of gene i.

The 3cPOS and mPOS methods are validated on several publicly available gene expression datasets using widely used classifiers to examine the impact of feature selection on model performance through classification accuracy. Selection stability is also used to address the captured biological knowledge in the obtained results. The experimental results reveal that the 3cPOS performs better than comparative feature selection methods. Additionally, the experimental results demonstrate that the mPOS either outperforms or demonstrates comparable performance. Both methods consistently deliver reliable performance, even with limited sample

sizes, underscoring their versatility and effectiveness in gene selection.

Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

1.1 Introduction

Functional genomics experiments, such as gene expression microarrays, play important roles in exploring the mechanisms of biological systems, identifying phenotypes and their associations with biological processes. The difficulty in using these data lies in creating new tools in statistical learning.

Statistical learning refers to a set of approaches for constructing a predictive model based on a given dataset. It encompasses many methods including, Random Forest [22], k Nearest Neighbour [100], Support Vector Machines [79], and eXtreme Gradient Boosting [32]. The primary objective is to construct an efficient prediction rule that can be employed for the prediction of new or unknown data by using a given set of training data.

The selected informative features could improve a developed classifier's interpretability and predictive accuracy. One of the main ambitions of gene expression analysis is to identify genes that are expressed differentially between different classes. Several studies have investigated the identification of these discriminative genes to use in classification.

Microarray data consists of tens of thousands of genes with only a small number of samples; dimensionality. This results in low model performance, model overfitting, or difficult-to-interpret results. To address these issues, the role of feature selection in machine learning is considered.

Alternatively, several search schemes are proposed for feature selection, including Least Absolute Shrinkage Selector Operator (LASSO) [93], and Minimum Redundancy and Maximum Relevance (mRMR) [148]. The identification of discriminative genes can be based on different criteria including, p-values of statistical tests, e.g. Wilcoxon rank sum test (Wilcoxon) [46] or Kruskal Wallis Test (Kruskal) [126]. Some methods utilised overlapping scores to assess a gene's importance, including Painter's feature selection [7], MaskedPainter [6], and Proportional Overlapping Score [123].

The overlap between gene expression for various classes is considered. The approach described in this thesis finds genes that are expressed differently in the target classes by considering the information given by samples' classes and gene expression data. This study explores the potential to improve classification performance and predictive accuracy by identifying genes that are discriminative and relevant to the specific classification task.

The thesis proposes a procedure that considers the core expression intervals and overlap between gene expression of different classes, to identify gene's discriminative characteristic on a training set whilst avoiding the effects of expression outliers. Based on this procedure, a novel feature selection technique for three classes, named as 3-class Proportional Overlapping Scores (3cPOS), is proposed for selecting discriminative features for a considered classification task. This method results in a measure, called 3cPOS score, of a feature's relevance to the classification problem.

Random Forest (RF), k Nearest neighbour (kNN), Support Vector Machines (SVM), and Extreme Gradient Boost (XGBoost) are employed to evaluate the effectiveness of the proposed approach in enhancing the learning process. To assess the performance of 3cPOS method, seven publicly available gene expression datasets are used with comparison with three well-established gene selection techniques: Kruskal; LASSO; mRMR. The experimental results of classification accuracies computed using the considered classifiers reveal that 3cPOS achieves a superior performance.

The minimum subset of genes plays an important role in identifying the smallest set of genes that can effectively classify the largest number of samples in the training phase. This process of final gene selection involves integrating both the minimum subset of genes and gene

ranking. Two distinct ideas are proposed to facilitate gene ranking: Idea 1 and Idea 2. To evaluate the effectiveness of these procedures, the performance of Idea 1, Idea 2, and 3cPOS is compared across RF, k-NN, SVM, and XGBoost classifiers using fourteen publicly available gene expression datasets. The experimental results indicate that inclusion of the minimum subset of genes and gene ranking (through Idea 1 and Idea 2) does not yield significant improvements when compared to the performance achieved by using the 3cPOS method alone.

The 3cPOS method is further extended to handle multi-class problems, called as multiple Proportional Overlapping Scores (mPOS). This extension also utilises proportional overlapping analysis, taking into account the class intervals and overlaps between class intervals are considered to derive mPOS measure. To evaluate the effectiveness of mPOS method, nine publicly available gene expression datasets involving binary classification tasks are employed with comparison with three well-established gene selection techniques: Wilcoxon; LASSO; mRMR. Additionally, fifteen publicly available gene expression datasets with multi classification tasks are used with comparison with three well-established gene selection techniques: Kruskal; LASSO; mRMR. The experimental results demonstrate that mPOS either outperforms or demonstrates comparable performance. In additions, mPOS is not limited by the number of genes it can handle. It maintains reliable performance, even with small sample sizes, highlighting its versatility and effectiveness in gene selection.

Simulation studies play important role in getting insight into ability of statistical methods under various conditions. Simulation datasets are generated based on balanced class distributions with degrees of overlaps, resulting in a total of 12 scenarios for each classification task. 3cPOS and mPOS demonstrate robust and effective performance under conditions characterised by a balanced class distribution, with a focus on assessing the impact of noise in input features, increased variance differences among classes, and varying degrees of class overlap.

1.2 Motivation

Functional genomics advancements, including gene expression microarrays, have played a crucial role in offering insights into disease systems, biological mechanisms, and phenotypic associations. However, technologies such as gene expression microarrays provide the datasets,

measuring tens of thousands of genes in a small number of samples. By analysing these datasets using statistical techniques, this may result in overestimation, poor interpretation, and inaccurate prediction. Feature selection techniques are considered to overcome these problems.

Feature selection methods are employed to mitigate these issues by addressing the unique complexities of gene expression data. For example, filter methods, such as the Wilcoxon or Kruskal methods, may not be suitable for biological data due to assumptions (e.g., normality distribution or independence). This may result in inaccurate gene ranking and sensitivity to noise from measurement errors and variations. Wrapper methods are used to evaluate subsets using a classifier and require high computation because of the search space. Embed methods, such as LASSO, provide sparse models, but this method ignores considering feature correlation or redundancy. Furthermore, mRMR is a mutual information-based method that aims at reducing redundancy while prioritising relevant features. However, this method ignores classifier-specific performance.

Overlapping-based approaches highlight significant insights into gene discrimination power by assessing core expression overlaps. For instance, reducing overlap to interval lengths in multi-class problems is considered in Painter's method, while ignoring sample counts and proportions. By doing this, outliers may occur. Gene masks are exploited to benefit classification coverage in the MaskedPainter method, but it relies on full expression lengths, making it sensitive to extreme values and a lack of weighted proportion. In addition, interquartile ranges and proportions analysis are applied in the POS method. These components provide core expression intervals and gene discriminative scores, making POS an outstanding technique due to significant improvement in terms of classification accuracy and stability. However, POS is restricted to binary classes. These gaps point out the significant need for overlapping-based methods, including mitigating outliers, incorporating proportional analysis, as well as extending to multi-class problems. This helps identify discriminative genes in high-dimensional datasets.

The primary motivation of this thesis is to identify differentially expressed genes that can overcome these limitations by distinguishing between classes, such as medical stages, cancer stages or phenotypes. Proportional overlapping analysis is exploited to develop more effective feature selection approaches that boost model performance, prediction accuracy, as well as

precise interpretation. Additionally, by converting complicated gene data into valuable biological information, it provides a deeper understanding of gene discriminative traits. These implications may help enhance treatment methods and diagnostic tools in areas such as cancer categorisation.

1.3 Research Objectives

The main objectives of this thesis are to develop feature selection techniques for gene expression analysis, with a focus on multi-class classification problems. The objectives are described as follows:

- A 3-class Proportional Overlapping Scores (3cPOS) method is proposed for three-class problems. This method exploits core expression intervals and overlaps between classes to derive gene discriminative powers or capabilities.
- The 3cPOS method is extended to multi-class problems, called multiple Proportional Overlapping Scores (mPOS) method. This technique allows for scalable applications to expression datasets with a changing number of classes.
- The proposed methods are assessed in terms of classification accuracy, stability, and computational complexity against Wilcoxon, Kruskal, LASSO, and mRMR techniques using RF, k-NN, SVM, and XGBoost classifiers across several gene expression datasets.
- The integration of a minimum subset of genes and gene ranking, Idea 1 and Idea 2, offers final gene selection, and assesses their performance on classification accuracy.
- Simulation studies are generated under different conditions, where balanced classes with adjusted overlaps, noise, and variance are considered. These studies provide the robustness of the 3cPOS and mPOS methods with respect to controlled environments.
- An R package is developed to facilitate users in implementing these methods in real-world research.

These objectives are used to cover gaps in current feature selection procedures to provide more precise and interpretable models for functional genomics.

1.4 Research Questions

This thesis aims to address the following research questions:

- How can the Proportional Overlapping Score (POS) method be extended for application to multi-class problems?
- How do the 3cPOS and mPOS methods improve classification performance in comparison to Wilcoxon, Kruskal, LASSO, and mRMR, especially in terms of classification accuracy, stability, and computational complexity, across multi-class problems?
- What is the main effect of including a minimum subset of genes and gene ranking strategies on the overall feature selection efficacy?
- How robust are the 3cPOS and mPOS methods under simulated conditions with balanced class distributions regarding factors such as noisy input features, variance differences among classes, and adjusting degrees of overlap, in comparison to other feature selection methods?

These research questions play important roles in examining both the theoretical and practical improvements, which provide a framework for evaluating methodological development.

1.5 Contributions

Several contributions are made in this thesis, particularly in focusing on feature selection in gene expression data;

- **Novel methods**: This thesis introduces the 3-class Proportional Overlapping Scores (3cPOS) for three-class problems and extends the concept to the multiple Proportional Overlapping Score (mPOS) for multi-class problems. Unlike other feature selection techniques, Painter's method [7] relies on interval length overlaps without considering sample counts and proportions. However, MaskedPainter [6] is developed to utilise full ranges sensitive to outliers. Furthermore, the Proportional Overlapping Score (POS) [123] exploits

Interquartile Range (IQR) to determine core expression intervals to enhance robustness. It also incorporates proportional class contribution to derive gene discriminative scores. In comparison, 3cPOS and mPOS are proposed to directly address three-class problems and multi-class problems, respectively.

- **Empirical Validation**: The 3cPOS and mPOS methods are evaluated across seven and twenty-four publicly available gene expression datasets, respectively, in terms of classification accuracy, stability, and computational complexity. These findings reveal that these methods demonstrate either superior or comparable performance to Wilcoxon [187], Kruskal [105], LASSO [175], and mRMR [148] techniques, using RF, kNN, SVM, and XGBoost classifiers.

- **Integration Strategies**: The minimum subset of genes and gene ranking ideas are integrated to provide important insights into their limited additive value over standalone 3cPOS across several gene expression datasets. This approach not only enhances the balance in imbalanced datasets but also goes beyond the relative dominant class approach of POS method [123].

- **Simulation Studies**: For each classification task, twelve scenarios are generated with a balanced class distribution where noise, variance, and overlap variation are adjusted to investigate the methods' effectiveness as well as robustness and usefulness.

- **Practical Tools**: An R package is developed to provide users with easy implementation.

For the fields of functional genomics and statistical learning, these contributions provide researchers with significant tools in bioinformatics and machine learning.

1.6 Thesis Organization

Chapter 2 provides an overview of key concepts in biological learning, including Deoxyribonucleic Acid (DNA) and Ribonucleic Acid (RNA), gene expression, Affymetrix GeneChips, CEL files, as well as the classification of cancers. For DNA and RNA, it emphasises their structural

differences and roles in gene regulation. The chapter discusses the role of gene expression in cellular function and disease progression. Affymetrix GeneChips, also known as a high-throughput tool for measuring gene expression, are discussed. The structure and function of CEL files, which store raw microarray data, are explained. Pre-processing and normalisation of these files are essential for accurate analysis. The chapter also discusses how gene expression data supports cancer classification.

Chapter 3 provides an overview of key concepts in statistical learning, emphasising three main topics: classifiers, model evaluation, and feature selection methods. Classification algorithms are discussed, including Random Forest, k-Nearest Neighbours (k-NN), Logistic Regression, Support Vector Machines (SVM), and eXtreme Gradient Boosting (XGBoost). Model evaluation is also included to get an understanding of the performance of classification models. Classification model accuracy is employed to provide a comprehensive framework for assessing model performance. Feature selection is considered to address the dimensionality of datasets as well as improve classification accuracy. Five feature selection techniques are discussed, including the Wilcoxon Rank Sum Test, the Kruskal-Wallis Test, Least Absolute Shrinkage and Selection Operator (LASSO), Minimum Redundancy Maximum Relevance (mRMR), and Proportional Overlapping Score (POS).

Chapter 4 provides details of data preprocessing as well as a description of the datasets in this thesis. Data preprocessing is crucial for turning intensity values into expression measures. However, a description of the datasets includes names of data and diseases, genes, samples, class distribution, and sources, offering a summary of gene expression datasets.

Chapter 5 proposes a novel feature selection method, the 3-class Proportional Overlapping Score (3cPOS). Proportional overlapping analysis is exploited to assess a gene's discriminative power for three-class problems. for each gene and class, the core expression interval is determined to mitigate the effects of expression outliers. The overlap between expression intervals is assessed, offering a gene's discriminative characteristics. Both the core intervals and the overlaps between intervals are considered to derive the 3cPOS measure, providing a gene's discriminative capability. A smaller 3cPOS score offers higher discriminative capability to distinguish their correct target classes. This approach improves feature selection for three-class gene expression analysis.

Chapter 6 outlines a procedure for identifying the minimum subset of genes that yield the best classification accuracy for a given set of training data. This chapter includes the concept of a gene mask to measure the discriminative power of each gene, and the minimum subset of genes is discussed to mitigate the effects of expression outliers as well as removing redundant information. Furthermore, the relative dominant class is defined to address class imbalance by identifying the dominant class based on its relative roles. The final gene selection is determined based on the minimum subset of genes and gene ranking. This chapter discusses two ideas which are utilised to determine gene ranking. The first approach is to apply 3cPOS scores and the relative dominant class to generate gene ranking by using a round robin fashion. Another idea is to sort the remaining genes in ascending order according to 3cPOS score to determine gene ranking. The performance of feature selection is evaluated through a comparison of Idea 1, Idea 2, and 3cPOS.

Chapter 7 proposes an extension of 3cPOS method, which is called the multiple Proportional Overlapping Score (mPOS). mPOS performs a crucial role in identifying the discriminative ability of genes for multi-class problems by considering proportional overlapping analysis. For each gene and class, the class interval and overlapping between intervals are considered to alleviate the impacts on the expression outliers and to indicate gene discriminative characteristics, respectively. The mPOS scores are generated based on the class interval and overlapping between intervals, which offers discriminative power for gene i. A smaller mPOS score represents higher discriminative capability to distinguish their correct target classes. This method improves feature selection for multi-class gene expression analysis.

Chapter 8 presents simulation studies to assess the performance of the 3cPOS and mPOS methods across various experimental setups. To provide for a thorough assessment of the approaches' robustness and usefulness in a variety of settings, balanced class distributions with varied degrees of overlap are simulated. For each dataset, both non-informative and informative features are generated based on a standard normal distribution and a multivariate normal distribution, respectively. Two simulation models are utilised to generate informative features, resulting in 4 experiments. For each experimental setup, three distinct scenarios are designed to simulate varying degrees of overlaps among the classes. This offers different levels

of difficulty for feature selection methods, providing a comprehensive evaluation of the abilities of 3cPOS and mPOS methods to deal with complex classification tasks.

Chapter 9 summaries the conclusions of the thesis and suggests future directions in which this research might be extended.

1.7 Research Articles

Peer-reviewed Papers:

1. Feature Selection in Ternary Classification of Microarray Gene Expressions via Proportional Overlapping Analysis (Under Review).

2. Proportional Overlapping Analysis for Feature Selection in Multi-Class Classification of Microarray Gene Expressions (Manuscript under preparation for submission).

R Packages:

1. mPOS: An R Package for a 3-class Proportional Overlapping Score (Manuscript under preparation for submission).

CHAPTER 2

Background for Biological Learning

This chapter provides an overview of the fundamental principles of biology, with a particular focus on Deoxyribonucleic Acid (DNA), Ribonucleic Acid (RNA), and base pairing, as detailed in Section 2.1. Gene expression, Affymetrix GeneChip technology, CEL files, and associated software tools are discussed in Sections 2.2 - 2.4. Additionally, Section 2.5 provides an overview of cancers and their classification which is essential for understanding the disease's complexity, heterogeneity, and underlying biology.

2.1 Deoxyribose Nucleic Acid and Ribose Nucleic Acid

DNA contains the instructions needed to build proteins, and specific molecules to develop and function in the body. For example, human DNA determines the colours of eyes, hairs, or skin tones.

DNA contains units of biological building blocks which are called nucleotides. Figure 2.1 demonstrates the example of nucleotide structure which consists of the following three joined components: phosphate group, pentose sugar, and nitrogenous base.

Purines and Pyrimidines are nitrogenous bases that form the two different kinds of nucleotide bases in DNA and RNA. Purines have double hydrogen-carbon rings with four nitrogen atoms and consist of Adenine (A) and Guanine (G), while pyrimidines have a single hydrogen-carbon ring with two nitrogen atoms and comprise Cytosine (C), Thymine (T), and Uracil (U) as the

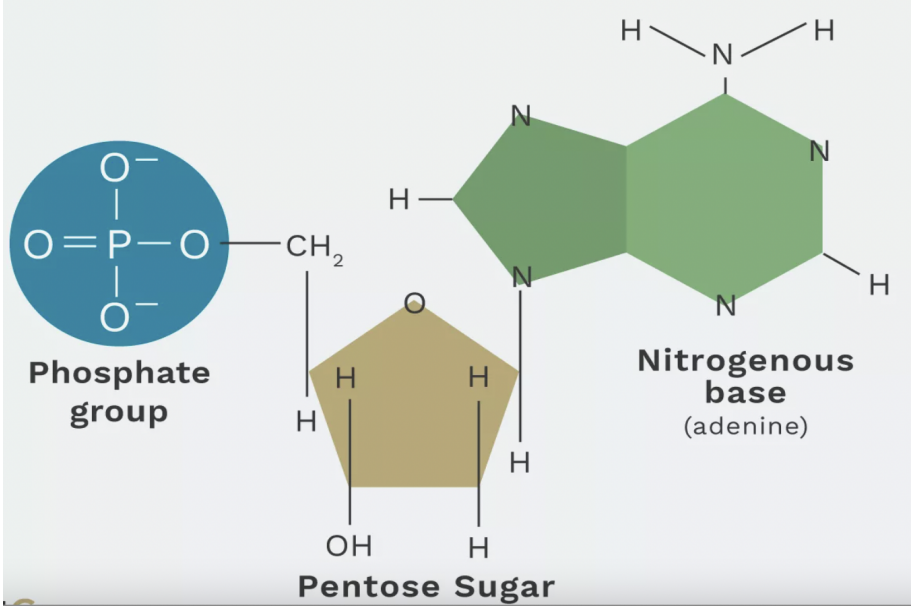

Figure 2.1: The example of a nucleotide structure [174].

nucleobase. Among these five different bases, Adenine, Guanine, Cytosine and Thymine are found in the synthesis of DNA, while Adenine, Guanine, Cytosine and Uracil are involved in the synthesis of RNA [165]. Figure 2.2 shows the structure of the five nitrogenous bases and their division according to two groups.

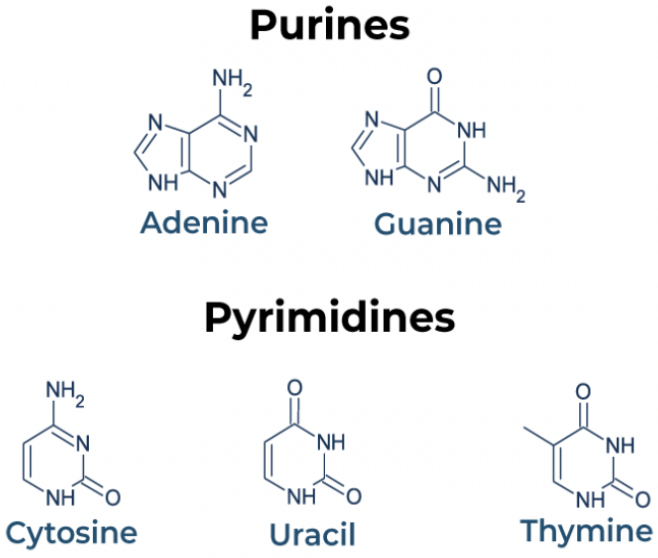

Figure 2.2: Five different nucleobases involved in the synthesis of DNA and RNA molecules which fall within two categories; purines and pyrimidines [65].

Nucleotides are linked together by the phosphate group of one nucleotide joining the sugar unit in a second nucleotide. This unit is joined to the next nucleotide, and the process is iterated

to create a long nucleic acid strand. The sugar-phosphate backbone provides a direction to the strand. Figure 2.3 shows that it has two ends which are different from each other. One strand runs from 5' end (5 prime) to 3' end (3 prime), while the other strand runs from 3' end (3 prime) to 5' end (5 prime). The direction of DNA sequences is usually written from 5' to 3' which means that the nucleotide at the 5' end comes first and the nucleotide at the 3' end comes last [139].

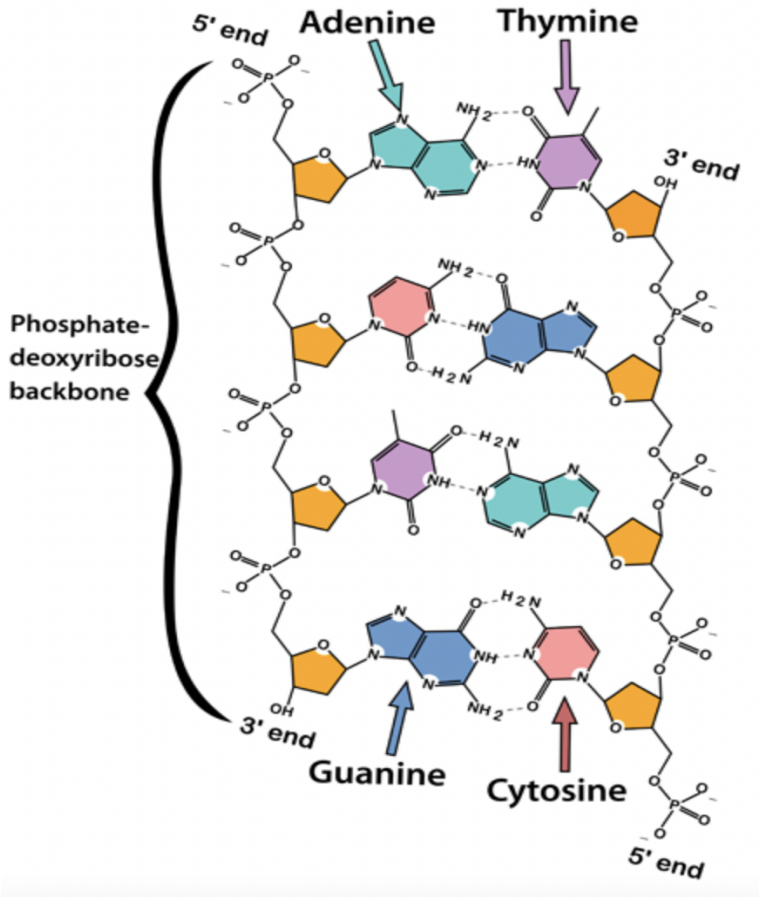

Figure 2.3: Construction of DNA strand [62].

Within a molecule of DNA, each unit of bases on a single strand links with another unit of bases on complementary strand base pairing [111]. For nitrogenous bases of DNA, adenine always pairs with thymine and guanine always binds with cytosine. Each pair of bases is linked together by hydrogen bonds. There are two hydrogen bonds between adenine and thymine and three hydrogen bonds between guanine and cytosine which are represented as $A = T$ or $T = A$ and $G \equiv C$ or $C \equiv G$, respectively [27]. The two complementary DNA strands are linked to each other by opposite directions and are therefore sometimes also referred to as *antiparallel* in their

nature. A sequence on one strand 5' CACGACTT 3' will pair to 5' AAGTCGTG 3' on the other strand as Figure 2.4. Moreover, the sugar-phosphate backbones are located on the outside of the ladder to link the chemical building blocks of DNA together. The two strands twist like a spiral staircase to form the shape of a double-helix structure as Figure 2.4. For nitrogenous bases of RNA, RNA contains adenine, guanine, and cytosine, similar to DNA. However, thymine is replaced with uracil in RNA. Uracil pairs with adenine, while guanine pairs with cytosine.

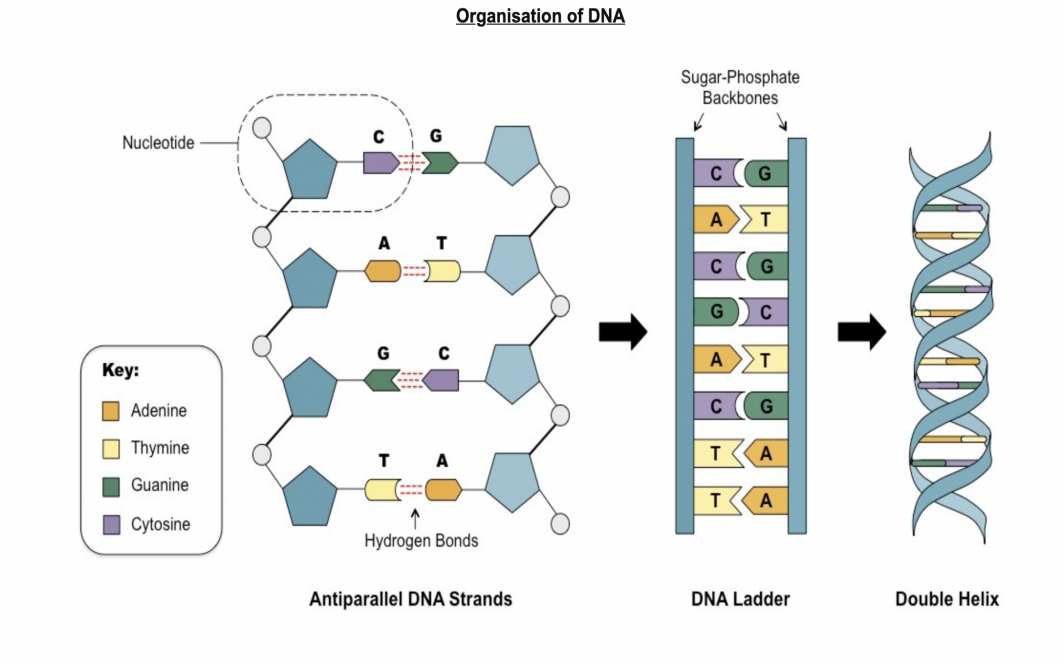

Figure 2.4: Construction of double helix DNA [149].

RNA is a molecule that is present in the majority of living cells, organisms and viruses. The nucleotide structure comprises ribose sugar attached to nitrogenous base and phosphate groups which has structural similarities to DNA. Unlike DNA, the nitrogenous bases of RNA include Adenine (A), Guanine (G), Uracil (U), and Cytosine (C) and RNA mostly exists in the single-stranded form, seen in Figure 2.5.

Three main types of RNA are involved in protein synthesis: messenger RNA (mRNA), transfer RNA (tRNA), and ribosomal RNA (rRNA) [42]. mRNA is used in transcription which is the process of RNA formation from DNA. mRNA is transcribed from DNA and contains the genetic blueprint to provide proteins. The tRNAs are RNA molecules that translate mRNA into proteins. The primary role of tRNA is to transport amino acids to the ribosome, aided by the enzyme aminoacyl-tRNA synthetase. The specific amino acid attached to a tRNA is

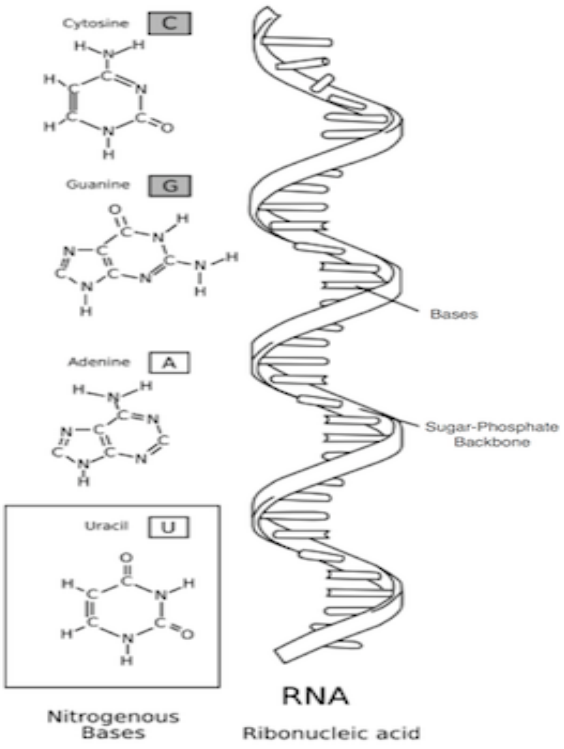

Figure 2.5: Construction of RNA strand [70].

determined by the mRNA codon, a three-nucleotide sequence that encodes for that amino acid. This protein-building process is called translation. rRNA forms ribosomes, which play a crucial role in protein synthesis. A ribosome consists of a large subunit and a small subunit. Ribosomes consist of three sites: the exit (E), peptidyl (P), and acceptor (A) site, which facilitate the binding of aminoacyl-tRNAs and the joining of amino acids to form polypeptides.

Gene expression refers to the process by which a gene is activated to produce a functional protein, involving two key stages: transcription and translation. The following section will provide an in-depth discussion on the mechanisms and regulation of gene expression.

2.2 Gene Expression

In recent years, the analysis of the expression of thousands of genes has become one of the main developments in the field of biology which helps researchers to examine and address issues within tissues, organs, or cells. Genes perform an important role in genetic information that carries characteristics in an organism and body. Genes comprise the instructions for creating proteins. As illustrated in Figure 2.6, the process of activating a gene to a protein is called gene

expression which consists of two main stages: transcription and translation. Transcription is the first stage in gene expression, where information in a strand of DNA is copied into a new molecule of messenger RNA (mRNA) by RNA polymerase. RNA polymerase unbinds the DNA double helix to allow one of the DNA strands to perform as a template for the synthesis of mRNA. In addition, as RNA polymerase moves forward along the DNA template, it also appends complementary RNA nucleotides to build mRNA strands. The synthesis of the RNA molecule appears in the 5' to 3' direction. Once a termination signal is reached in the DNA sequence, transcription is terminated. Translation is the second stage that uses the genetic information encoded in mRNA to synthesize protein. This process provides the mRNA sequence in triplets and assembles the corresponding amino acids to form a specific protein.

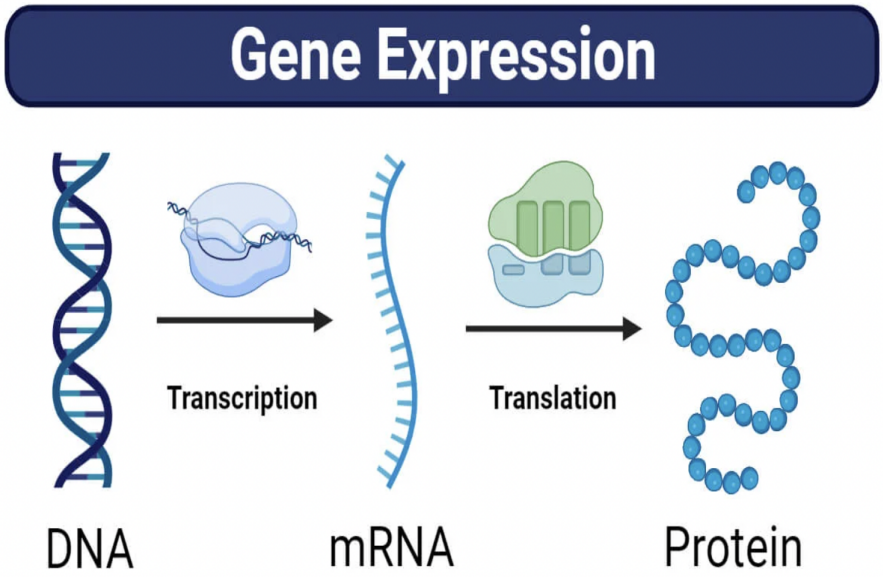

Figure 2.6: Gene expression [159].

The following section focuses on the acquisition of gene expression data through the use of microarray technology.

2.3 Affymetric Genechip

A powerful tool used to measure gene expression are called microarray. Microarray technology is widely used in molecular research. The genome is the entire set of DNA instructions within a cell, containing all the necessary information for an organism's development and functioning.

Approximately 30,000 genes from the human genome are stored and printed into a glass or silicon chip on microarray. The chip comprises a two-dimensional array that arranges biological samples for analysis of genetic information and each spot represents a single gene or a single-stranded sequence of DNA which is called a probe. Therefore, the configuration of probes holds importance, including the specific placement of each probe within the array [138].

Hybridization is the process in which two complementary single-stranded DNA and/or RNA molecules from different sources combine together through base pairing to form a double-stranded molecule. For the process of hybridization, the double-stranded DNA fragments are heated to various temperatures, which breaks the hydrogen bonds between the two strands of the DNA double helix, resulting in two separate single strands of DNA. The next step involves combining and cooling the strands to allow the formation of hydrogen bonds, resulting in the synthesis of double-stranded DNA. The degree of hybridization is categorized into 3 groups; complete hybridization, partial hybridization, and no hybridization. It is important to note that DNA is used directly for hybridization; however, when working with RNA, the mRNA must first be reverse transcribed into complementary DNA (cDNA) before hybridization occurs [11]. Figure 2.7 shows DNA-RNA hybridization where G base is paired with C, A of the RNA pairs with T of the DNA, and U or the RNA links with A of the DNA.

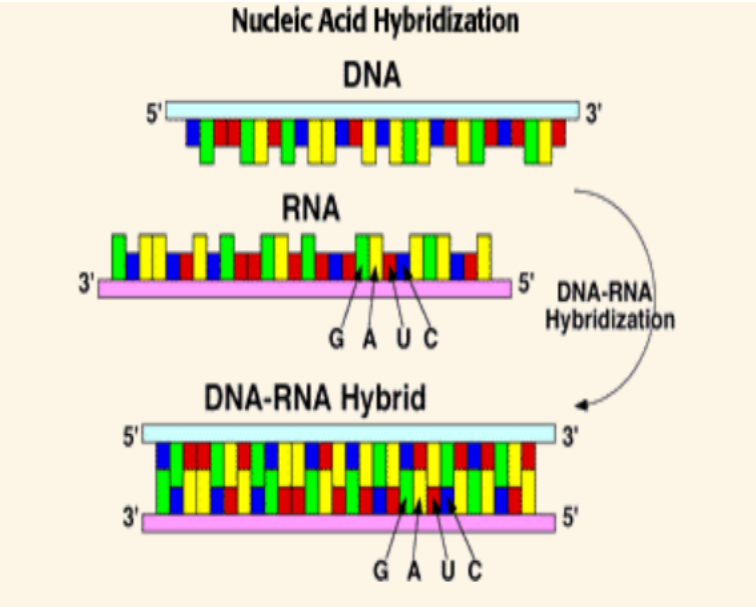

Figure 2.7: DNA-RNA hybridization [151].

Hybridization microarrays identify the RNA sequences present in a sample, thereby informing

researchers about the genes being expressed by a particular organism. When a gene is highly expressed, a greater number of RNA molecules will bind to the probes, resulting in a strong fluorescent signal when illuminated by a laser. Conversely, if a gene is expressed at lower levels, fewer RNA molecules will adhere to the probes, leading to a diminished fluorescent intensity at those probe locations.

Affymetrix has manufactured Affymetrix DNA Chips by integrating oligo synthesis and photolithographic computer chip technology [66]. Figure 2.8 shows an Affymetrix DNA Chip in which the surface of the grid is approximately 1.28 centimetres by 1.28 centimetres and each small grid is 11 micrometres by 11 micrometres holding a unique probe which is a distinct DNA sequence. Affymetrix DNA Chip contains many 25-mer oligonucleotide bases, called a probes[117]. To measure expressed RNA, the RNA samples are extracted from blood, tumours, or any other body tissues and these RNA strands will search for a corresponding match with a DNA probe on the array. When the sequence base of the RNA sample and the DNA probe are matched, they will hybridize to each other. For measuring gene expression, scientists next illuminate the array with laser light, inducing fluorescence. According to Figure 2.9, if a gene is highly expressed, numerous RNA molecules will attach to the probe, and the probe address will fluoresce brightly upon laser impact. In contrast, if a gene is expressed at a lower level, less of RNA will bind to the probe, resulting in a significantly dimmer response when exposed to the laser [44].

Figure 2.10 demonstrates pairs that each pair consist of a perfect match (PM) and mismatch (MM) probe. The PM probe matches perfectly to the target sequence while the MM probe is identical to the PM excluding the middle of the sequence (13th position base). Multiple probes are used for a given transcript/gene.

There will be a perfect match and the sample will stick to the probe seen in Figure 2.11 (a), while Figure 2.11 (b) shows that the RNA samples are rejected by the probes because C does not pair with another C. Therefore, there is no match.

The following section will explore methods for accessing gene expression data from CEL files and other data sources using various software tools.

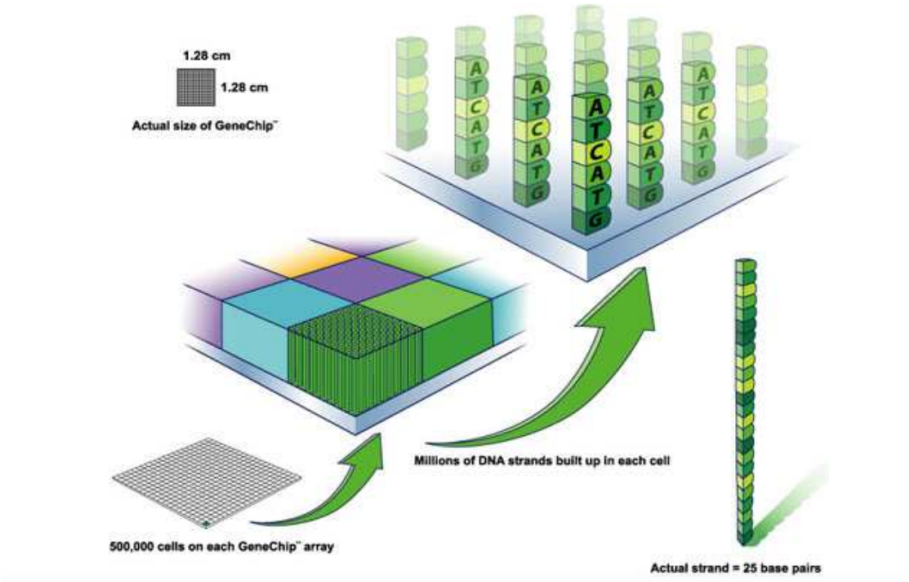

Figure 2.8: Structure of an Affymetrix DNA Chip [52].

2.4 CEL files

From the process of manufacturing to the experimental use, DNA chip data is provided by different electronic files. A CEL file is one of the file formats widely used to access gene expression data created from Affymetrix GeneChip. The level of gene expression measured by fluorescence therefore is produced in raw images, and Affymetrix image analysis software is used to convert raw images to intensity values. In our study, we downloaded datasets from the Gene Expression Omnibus (GEO) which is a public repository that provides and freely distributes high-throughput gene expression and other functional genomics data sets. In addition, dataset records include additional resources, including cluster tools and differential expression queries [40]. The data of thousands of experiments of Affymetrix GeneChip are available at GEO which are related to several designs of GeneChip for different organisms. The GEO homepage can be accessed at http://www.ncbi.nlm.nih.gov/geo/.

The expression level of a particular probe is expressed by the intensity values of each cell. Separate CEL files store the intensity values of each array. In addition, all the CEL files have unique file names which start with 'GSM' [64] . For instance, GSE23938 is the dataset of breast cancer which consists of 18586 genes from 41 samples based on three class problems. A CEL file consists of other information about the chip such as the dimension of the chip which represents

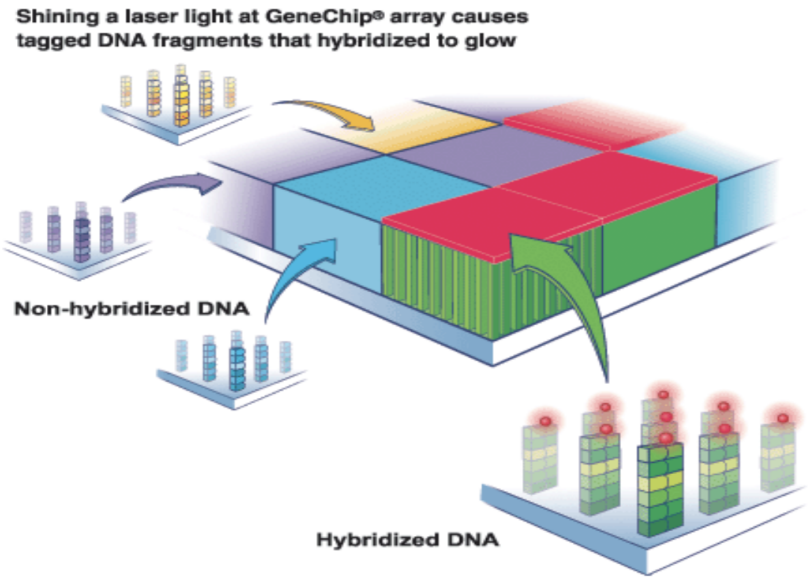

Figure 2.9: DNA detection [54].

the total number of rows and columns.

CEL files are essential for gene expression analysis, particularly in microarray studies. They store probe intensity data produced by Affymetrix GeneChips, which researchers can use to analyze and extract biological insights. Several built-in methods are proposed to turn probe intensity data into expression measures. In our study, we exploit a Robust Multiarray Average (RMA) [91] that includes three relevant steps; background correction, normalization, summary expression value computation. Background correction is employed to deal with background noise that occur in every scanner image. The RMA algorithm aims to ignore the MM probes and addresses background correction directly as a property of hybridisation. The procedure is based on the assumption that the observed probe signal comprises of a normally distributed background component and an exponentially distributed signal component. In hybridization, the amount of RNA in a sample can vary slightly between different chips. Even when the same sample is applied to multiple chips, there will be differences in the overall distribution of probe intensity values from one chip to another. To address these issues, normalization is considered. Normalization aims to identify and adjust for systematic differences between chips, allowing data from different chips to be directly compared. For RMA algorithm, sample normalization is achieved through quantile normalization [195] of the probe-level data. As we know that each gene is represented on the GeneChip by one or more probe sets and each probe set includes

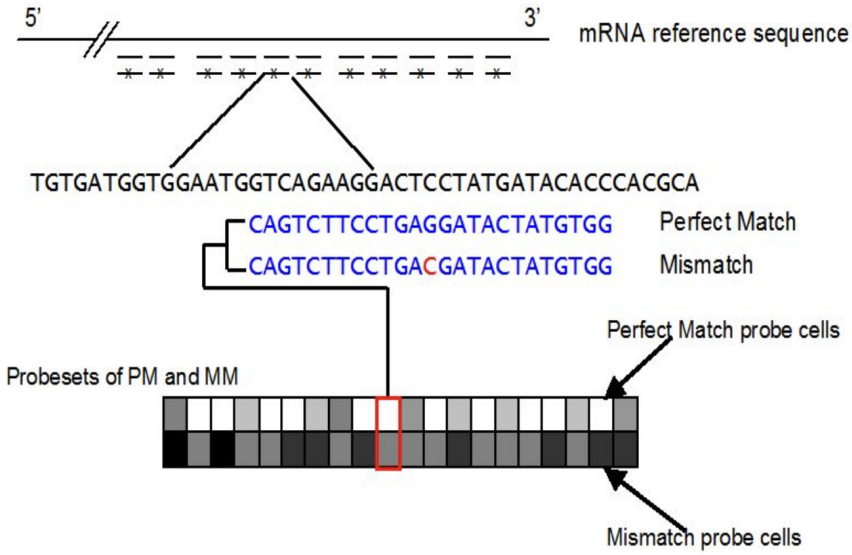

Figure 2.10: Perfect match and mismatch for multiple probes per gene [67].

11–20 probe pairs. To convert multiple probes into expression values, median polish method [69] is used as a part of the RMA algorithm which performs separately for each probe set and find the average of fitted values to use as probe-set-specific expression measures .

Analysing GeneChip data requires high computational techniques to handle such large amount of data. The softwares, R and Bioconductor, are assigned to deal with GeneChip data analysis and are widely used in a large number of research studies. R is a free programming language and is used for statistical computing and graphical facilities to analyse and visualize data. In addition, it is available on Windows, Linux, and MacOS [181]. R programming is effective and contains a variety of mathematical and statistical functions. Users can generate new functions or packages and add additional functionality to the R and Bioconductor systems. In our studies, the affy package [90] is used for analysing the GeneChip data as data processing and more information on affy is available at: https://www.bioconductor.org/packages/devel/bioc/vignettes/affy/inst/doc/affy.pdf. Moreover, statistical functions and packages are implemented in part of feature selection methods and classification that will be described in next chapter.

Proteins are crucial in regulating essential functions, including cell growth. Genetic mutations can alter the structure, function, and amount of protein. These effects lead to changes in a cell's behavior that may transform it from a normal state to a cancerous one. The following section

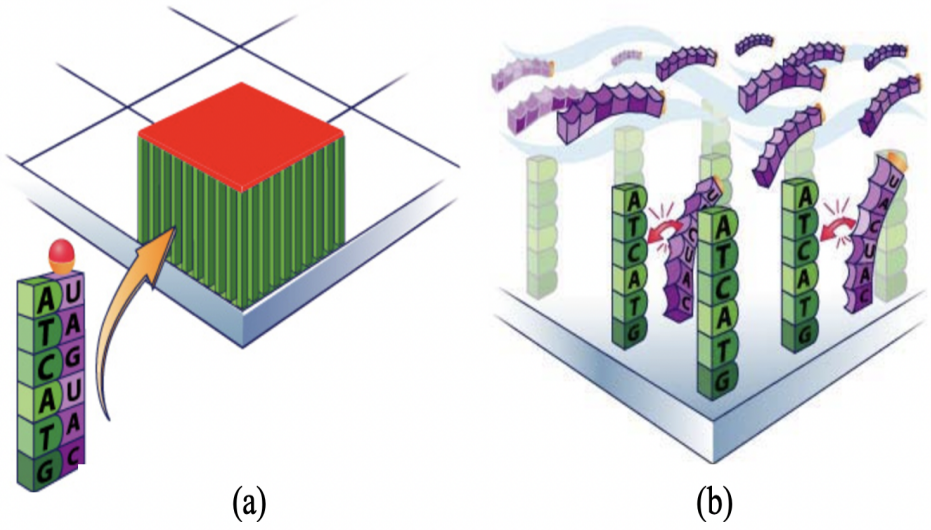

Figure 2.11: (a) the RNA sample and DNA probe are matched and (B) the RNA sample and DNA probe are not struck [97].

will provide discussion on diseases with a particular emphasis on several types of cancers.

2.5 Cancers and their classification

For many decades, a substantial number of individuals globally have faced severe diseases that contribute to rising mortality rates. Cancer is one of the most common causes of death and an important obstacle to enhancing life expectancy in countries worldwide [21]. Cancers are caused by environmental factors or lifestyle choices [182]. For instance, tobacco consumption is a significant contributor to deaths associated with lung cancer. Alcohol, diet, and obesity are also important contributors to the development of various cancers, including those of the mouth, larynx, throat, gastrointestinal tract, kidney, gallbladder, and breast. Genetic factors also play a significant role in the development of cancer [60]. For example, families with a strong genetic tendency toward breast cancer are likely to develop the disease more frequently at earlier ages.

To understand the types of cancer, symptoms and staging of cancer is crucial for providing efficient diagnoses and the best treatment for patients. This may help mitigate the number of severe cases and deaths. In this thesis, we will focus on different cancer as follows;

2.5.1 Breast Cancer

Breast cancer is cancer that can be found as a growth of cells in the breast tissue. It is the commonest cause of cancer death among women across the world. According to [101], incidence rates are notably higher in developed nations. The incidence of breast cancer rises significantly with age during the reproductive years, subsequently increasing at a more gradual pace after the age of 50.

There are different types of breast cancer and breast conditions. **Invasive breast cancer**, as known as invasive ductal carcinoma, is the most common type of breast cancer. Invasive breast cancer indicates that cancer cells have grown through the lining of the ducts into the surrounding breast tissue. **Invasive lobular breast cancer** is the second most common type of breast cancer and it is also known as invasive lobular carcinoma. Invasive lobular breast cancer refers to cancer that begins in the cells lining the lobules and then spreads to the nearby breast tissue. **Ductal carcinoma in situ** (DCIS) is an early breast cancer. This means that some cells in the lining of the ducts of the breast tissue have started to turn into cancer cells. These cells are all contained inside the ducts. They have not started to spread into the surrounding breast tissue. This means that certain cells in the lining of the breast ducts have begun to transform into cancer cells. However, these cells remain confined within the ducts and have not yet spread to the surrounding breast tissue. The difference between invasive breast cancer and DCIS lies in the behavior of the cancer cells. In invasive breast cancer, cells have broken out of the duct and spread to surrounding breast tissue, while in DCIS, some cells have started to turn into cancer, but remain confined within the ducts, as illustrated in Figure 2.12.

Doctors utilise the grade and stage of a cancer to help them to provide treatments for patients. Grading refers to how abnormal the cells look under a microscope which DCIS grade can be categorized into 3 grades; low grade (more slowly growing), intermediate grade, and high grade (more quickly growing). Staging is used to indicate how big the cancer is and how far it has spread. The most common one is the TNM system that can be employed to stage of breast cancer. TNM system stands for Tumour, Node, and Metastasis which can be spitted into 5 main stages [14].

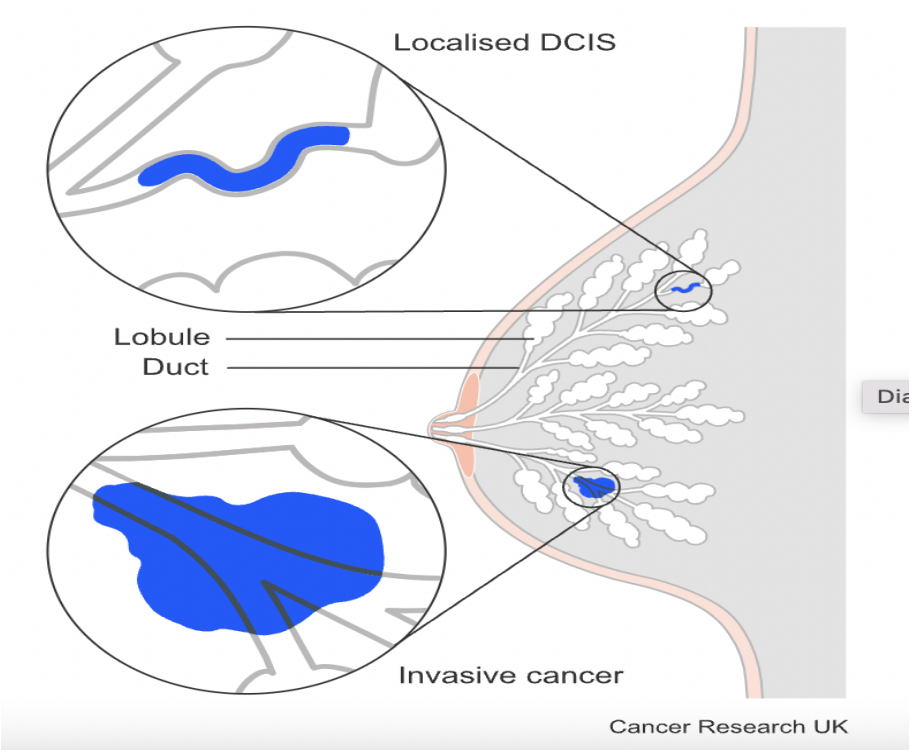

Figure 2.12: Three Types of Breast Cancer [178].

- Stage 0. This stage is considered as DCIS.
- Stage 1. The cancer is small and located in the breast tissue or lymph nodes which is nearby the breast.
- Stage 2. The cancer is either in the breast or in the nearby lymph nodes or both. It can be assessed as an early stage breast cancer.
- Stage 3. The cancer has spread from the breast to the lymph nodes close to the breast, the skin of the breast or to the chest wall.
- Stage 4. The cancer has spread to other parts of the body such as the liver,bones, lungs and brain. It is secondary breast cancer.

2.5.2 Colorectal Cancer

Colorectal cancer is cancer developed anywhere in the large bowel, which includes the tissues of the colon and rectum. The study of [144] states that colorectal cancer has been ranked as the fourth most common cancer in male and the third most common cancer in female across the

world. The rising prevalence of obesity, coupled with a decline in physical activity, contributes to the increasing global burden of colorectal cancer [26]. Knowing the stage of colorectal cancer is very important to determines the plan of treatment. Based on the TNM system, colorectal cancer is generally grouped into 5 stages, seen in Figure 2.13 .

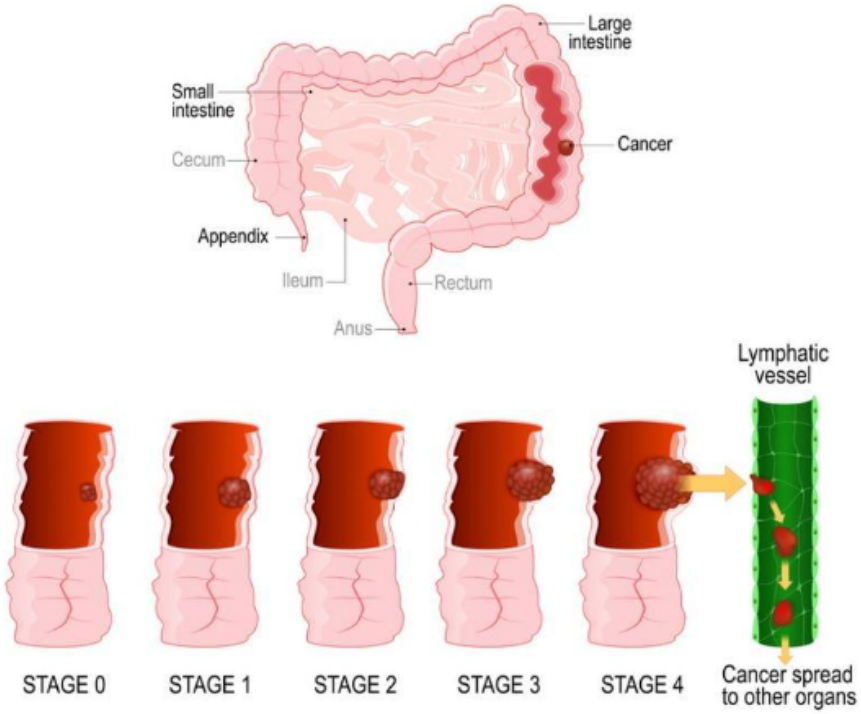

Figure 2.13: Stages of Colorectal Cancer [50].

- Stage 0. Abnormal cells are found in the innermost layer of the colon or rectum, known as the mucosa, which is the thin, moist tissue lining the colorectal wall
- Stage 1. The cancer has grown through the mucosa and has spread into the muscular layer of the colon or rectum but it has not spread to any lymph nodes or nearby tissue..
- Stage 2. The cancer has spread into the outer layers of the colon or rectum but has not spread to any lymph nodes.
- Stage 3. The cancer has spread into nearby lymph nodes, but has not reached other areas of the body.
- Stage 4. The cancer has been carried through the lymph and blood systems to distant parts of the body especially the lungs and liver.

2.5.3 Gastric tumors

Gastric tumors also known as stomach cancer; starts in the cells lining the stomach. Gastric cancer is now the fifth most commonly diagnosed cancer and the third leading cause of cancer-related deaths worldwide. Nonetheless, the global incidence of gastric cancer has been significantly decreasing, largely due to lifestyle changes that address dietary and environmental risk factors [135]. Comprehending the staging of gastric tumors is essential for identifying the most effective treatment. The TNM classification system is widely utilized to categorize the stages of gastric tumors, which are organized into four distinct stages, seen in Figure 2.14.

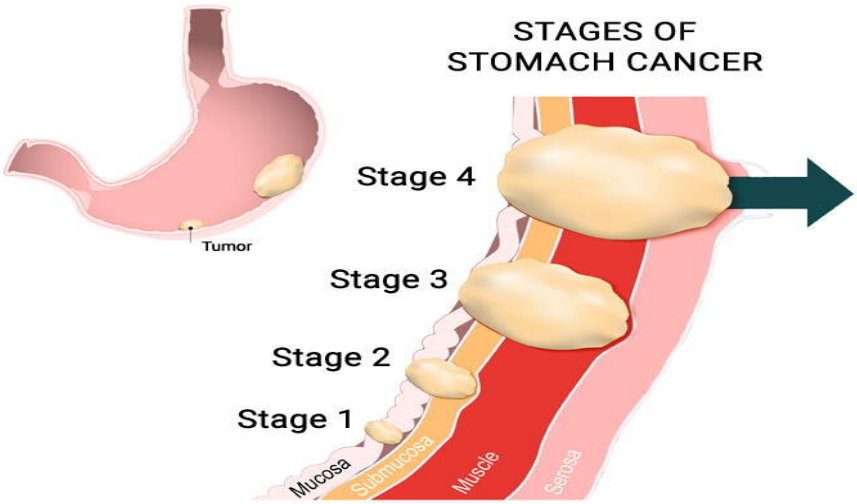

Figure 2.14: Stages of Stomach Cancer [128].

- Stage I. The cancer has grown through the mucosa and may have spread to the submucosa, lymph nodes, or the muscle layer.
- Stage II. The cancer has spread to the submucosa and nearby lymph nodes, muscle layer and lymph nodes, or the serosa.
- Stage III. The cancer has spread to nearby organs, such as the spleen, colon, liver, diaphragm, pancreas, abdomen wall, adrenal gland, kidney, or small intestine, or to the back of the abdomen.
- Stage IV. The cancer has spread to other parts of the body, such as the lungs, liver, distant lymph nodes, and the tissue that lines the abdomen wall.

2.5.4 Leukemia

Leukemia is a type of cancer that affects the body's blood-forming tissues, such as the bone marrow and the lymphatic system. The four primary subtypes of leukemia are Acute lymphoblastic leukemia (ALL), Acute myelogenous leukemia (AML), Chronic lymphocytic leukemia (CLL) and Chronic myelogenous leukemia (CML). The ALL and AML subtypes are commonly found in both children and adults, whereas CLL and CML typically affect older individuals [36]. Knowing the stages of four primary subtypes is crucial to provide the best treatment.

The staging of ALL is primarily determined by the white blood cell (WBC) count at the time of diagnosis. In children with ALL, staging is categorized into two risk groups: low risk and high risk. For adults, ALL is classified into three stages: untreated, in remission, and recurrent.

Instead of staging of AML, doctors generally use the French-American-British (FAB) system to assign this disease into subtypes. The FAB system has been developed by [116] which classifies AML into subtypes from M0 to M7 as follows;

- M0 undifferentiated acute myeloblastic leukemia
- M1 acute myeloblastic leukemia with minimal maturation
- M2 acute myeloblastic leukemia with maturation
- M3 acute promyelocytic leukemia
- M4 acute myelomonocytic leukemia
- M4 eos acute myelomonocytic leukemia with eosinophilia
- M5 acute monocytic leukemia
- M6 acute erythroid leukemia
- M7 acute megakaryoblastic leukemia

Even through FAB subtypes have not provided staging of AML, the FAB subtype does affect your survival odds. Three main survival rate are defined as follows;

- High survival rate. Patients with subtypes M1, M2, M3, or M4eos has high survival rate
- Average survival rate. Patients with subtypes M3, M4, and M5 have average AML survival rates.
- Low survival rate. Patients with subtypes M0, M6, and M7 experience a poorer prognosis, as these subtypes have a lower survival rate compared to the average for all AML subtypes.

The Rai staging system is frequently employed in evaluating chronic lymphocytic leukemia (CLL) [153]. This system is based on three factors: the number of cancerous white blood cells (WBCs) in your body, the number of red blood cells (RBCs) and platelets in your body, and whether or not your lymph nodes, spleen, or liver are enlarged. The five RAI stages for CLL are defined as follows;

- stage 0. There are too many abnormal WBCs (generally more than 10,000 in a sample), called lymphocytes, in your body. Stage 0 is assigned as low risk.
- Stage I. This stage is identical to stage 0 where there is a lymphocyte count of more than 10,000 per sample. However,the lymph nodes will also be swollen. This stage is considered intermediate risk.
- Stage II. The liver or spleen has also become enlarged, along with the swollen lymph nodes. The level of lymphocytes is high, however, other blood counts are normal. This stage is regarded as having an intermediate risk.
- Stage III. Other blood cells start to be affected. Patients with this stage are anemic and don't have adequate RBCs. A sustained elevation in lymphocyte count, coupled with the swelling of lymph nodes, spleen, and liver, is commonly observed in advanced stages of disease. Stage III is classified as high risk due to the increased likelihood of systemic involvement and potential for severe complications.
- Stage IV. In addition to all of the symptoms from the previous stages, platelets and RBCs are affected, and the blood will not be able to clot properly. Stage IV is considered high risk.

The staging of CML is determined based on the percentage of cancerous WBCs in your body [177]. This can be divided into the following three substages;

- Chronic phase. There are less than 10 percent of the cells in your bone marrow and blood are blast cells in the chronic phase. Most patients with this stage have fatigue and other mild symptoms.
- Accelerated phase. Between 10 and 19 percent of the cells occurs in the bone marrow and blood are blast cells. The accelerated phase arises when the cancer fails to respond to treatment in the preceding phase.
- Blastic phase CML. This stage is an aggressive stage of CML because more than 20 percent of the cells in the your blood and bone marrow cells will be blast cells.

2.5.5 Lung Cancer

Lung cancer is one of the most common and serious type of cancer. It starts in the windpipe (trachea), the main airway (bronchus) or the lung tissue. Smoking is the most common cause of lung cancer. The authors of [1] reveals that approximately 90% of lung cancer cases are attributable to smoking. Therefore. knowing the stages of lung cancer can help patients to obtain the best plan for treatments which can minimize the severity of the symptoms and the number of deaths. The TNM system is commonly used to classify lung cancers [154]. Figure 2.15 shows the four stages for lung cancer which are denoted as follows;

- Stage 1. This stage can be divided into 2 substages; stage 1A and stage 1B. For the stage 1A, it includes three possible stages; stage 1A1, stage 1A2 and stage 1A3. According to stage 1A1, the cancer is 1cm or less at its widest part and it has not grown into the membranes that surround the lungs (pleura). It has not grown into the main branches of the airways and has not spread to nearby lymph nodes and distant parts of the body. For stage 1A2, the size of cancer is between 1 cm and 2 cm and it has not grown into the membranes that surround the lungs (pleura) and the main branches of the airways. It has not spread to nearby lymph nodes and distant parts of the body. Stage 1A3 is identical to previous stage but the size of cancer is between 2 cm and 3 cm. For Stage 1B, the size of cancer is

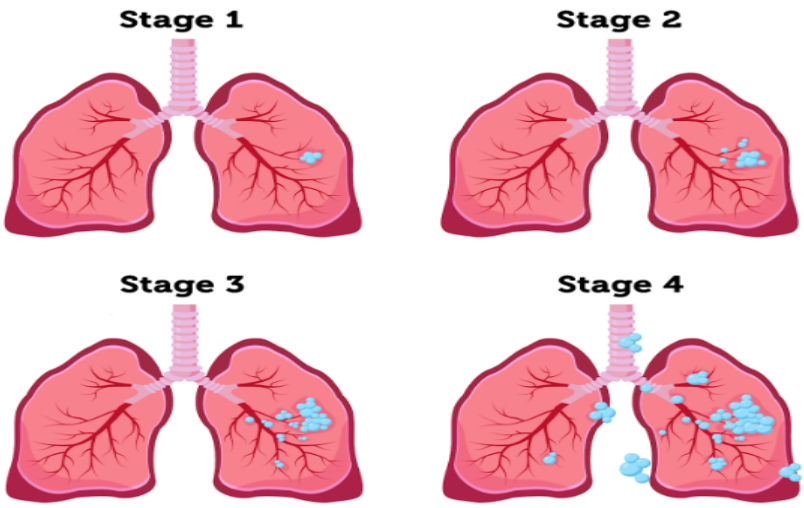

Figure 2.15: Stages of Lung Cancer [122].

between 3 cm and 4 cm it has grown into the main airway of the lung (main bronchus) and the membrane covering the lung (visceral pleura). Additionally, It has resulted in the lung partially or fully collapsing by obstructing the airway or causing inflammation of the lung tissue (pneumonitis). This stage is called early lung cancer.

- Stage 2. This can be spitted into stages 2A and stage 2B. For stage 2A, the size of cancer is between 4 cm and 5 cm and it has grown into the main airway of the lung (main bronchus) and the membrane covering the lung (visceral pleura). It has caused the lung to partly or completely collapse by blocking the airway or causing inflammation of the lung tissue (pneumonitis). However, it has not spread to the lymph nodes and different parts of the body. For stage 2B, the size of cancer is between 3 cm and 5 cm. It has grown into the chest wall, the inner lining of the chest wall (parietal pleura), the nerve close to the lung (the phrenic nerve), the layers of the sac that covers the heart (parietal pericardium). Two or more tumours in the same lobe of the lung have been found. However, it has not spread to the lymph nodes and different parts of the body.

- Stage 3. The size of cancer can be between 5 cm and 7 cm or larger than 7 cm and it has usually spread to lymph nodes. It may also be growing into such as other parts of the lung, the airway, and surrounding areas outside the lung. The cancer may also have spread to tissues and structures further from the lung, however, it has not spread to other parts of the

body. Stage 2 and 3 lung cancer is called locally advanced lung cancer.

- Stage 4. The size of cancer can be any size. It may have spread to lymph nodes and the lung on the other side. Cancer cells in fluid in the pleura or around the heart have been found. Additionally, the cancer has spread to another part of the body namely the liver, bones or brain. Stage 4 of lung cancer is called metastatic or secondary lung cancer.

2.5.6 Ovarian Cancer

Ovarian cancer is a cancerous tumor that starts in the tissues of an ovary. The ovaries are two female reproductive glands responsible for producing eggs and female hormones. Ovarian cancer is the most common cause of cancer death in women. According to [24], it has found that the risk of a woman developing ovarian cancer up the age of 95 is estimated to be 1.1%. Assessing the stages of ovarian cancer can help determine effective prognosis and treatment, which benefits patients with severe symptoms and reduces mortality. The authors of [142] have developed the FIGO system, International Federation of Gynecology and Obstetrics, to be ulitsed in determining stages of ovarian cancer. Figure 2.16 demonstrates 4 stages of ovarian cancer which can be indicated as follows;

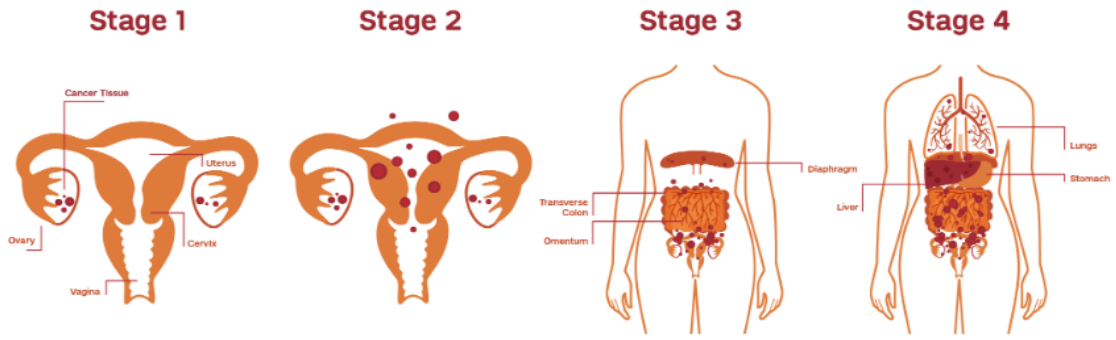

Figure 2.16: Stages of Ovarian Cancer [5].

- Stage 1. This stage occurs only in the ovaries which can divided into 3 groups; 1A, 1B, and 1C. Stage 1A and Stage 1B can find cancer inside one ovary and both ovaries, respectively. Stage 1C is split in to 3 subgroups: stage 1C1, stage 1C2, and stage 1C3. For stage 1C1, the cancer can be found in one or both ovaries and the ovary ruptures (bursts) during

surgery. For stage 1C2, the cancer is present in one or both ovaries, and either the ovary ruptures (bursts) prior to surgery or there is some cancer located on the surface of an ovary. For stage 1C3, the cancer is located in one or both ovaries and cancer cells in fluid taken from inside your abdomen during surgery have been found.

- Stage 2. The cancer has grown outside the ovary or ovaries which is growing within the area circled by your hip bones (the pelvis). Cancer cells in the abdomen may be found. This stages is divided into 2 subgroups: stage 2A and stage 2B. In stage 2A, the cancer has invaded the fallopian tubes or the uterus, whereas in stage 2B, it has spread to adjacent pelvic tissues, specifically the bladder or bowel (rectum or sigmoid colon).

- Stage 3. The cancer has spread outside the pelvis to the lining of the abdominal cavity (peritoneum) and it can also spread to the lymph nodes in the back of your abdomen. This can be divided into 3 subgroups; stage 3A, stage 3B and stage 3C. For stage 3A, the cancer cells has spread to the lymph nodes in the back of your abdomen in stage 3A1. For stage 3A2, cancer cells from tissue samples have been found from the lining of your abdomen (peritoneum) and it might also be in your lymph nodes. In stage 3B, the cancer is characterized by a size of 2 cm or smaller and may be located on the lining of the abdomen (peritoneum). Additionally, there may be evidence of cancer in the lymph nodes. For stage 3C, the size of cancer is larger than 2 cm on the lining of your abdomen (peritoneum) and it might found cancer in your lymph nodes.

- Stage 4. The cancer has spread to other body organs some distance away from the ovaries such as the liver or lungs which is divided into 2 subgroups: stage 4a and stage 4b. The cancer has led to an accumulation of fluid in the lining of the lungs, known as the pleura, in the stage 4a. For the stage 4b, the cancer has spread to the inside of the liver or spleen, lymph nodes outside the abdomen, and other organs such as the lungs.

2.5.7 Prostate Cancer

Prostate cancer is one of the most common types of cancer. The prostate is a small gland in males, shaped like a walnut, that generates seminal fluid, which nourishes and transports sperm.

[168] states that prostate cancer is the fifth leading cause of cancer-related deaths among men and the most frequently diagnosed malignancy in over 50% of countries (112 out of 185). Staging is employed to describe the extent of the disease which is vital to guide the treatment plan and predict the patient's prognosis. [19] has discussed the staging of prostate cancer and the TNM system is a common way to stage prostate cancer. The TNM system can be defined as follows;

- Tumor (T). Tumor describes the size or area of the prostate cancer that can be divided into 4 main T stages; T1, T2, T3, and T4, seen in Figure 2.17. For stage T1, the cancer is too small to be seen on a scan, or felt during an examination of the prostate. It's divided into T1a, T1b and T1c. The cancer is in less than 5% and in 5% or more of the removed tissue for stage T1a and stage T1b, respectively. For the stage T1c, the cancers can found by biopsy. For the stage T2, the cancer is completely inside the prostate gland. For the stage T3, the cancer has broken through the capsule (covering) of the prostate gland which id divided into T3a and T3b. The cancer has broken through the capsule (covering) of the prostate gland in the stage T3a while spreading into the tubes that carry semen (seminal vesicles) in the stage T3b. For the stage T4, the cancer has spread into other body organs nearby namely the back passage, bladder, or the pelvic wall.

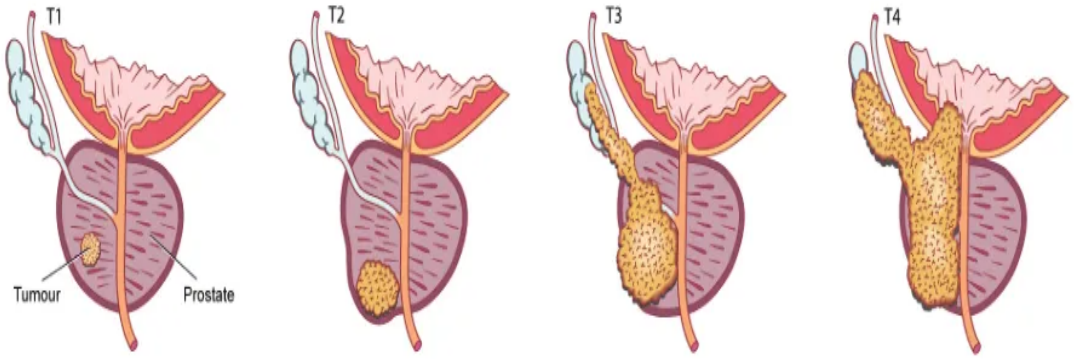

Figure 2.17: Stages of Prostate Cancer [179].

- Node (N). It describes whether the cancer has spread to the lymph nodes where is grouped into N0 and N1. The cancer cells are not found nearby lymph nodes in the stage N0, while the cancer cells in lymph nodes locate near the prostate in the stage N1.

- Metastasis (M). It describes whether the cancer has spread to a different part of the body which are split into 2 M stages; M0 and M1. The cancer hasn't spread to other parts of

your body in the stage of M0 and the cancer has spread to other parts of the body outside the pelvis. For M1, it is split into 3 subgroups; M1a, M1b and M1c. There are cancer cells in lymph nodes outside the pelvis and the bone for stage M1a and stage M1b, respectively. For the stage of M1c, the cancer cells are found in other parts of the body such as the lungs.

2.6 Direction of Chapter two to three

The use of gene expression data in cancer research facilitates the diagnosis, analysis of progression and aggressiveness, prognosis, and prediction of therapeutic outcomes. It also enables the identification of patients who may benefit from specific treatments, thereby enhancing the understanding of the disease and its underlying biology. To fully leverage these benefits, the integration of biological insights with statistical learning approaches is crucial, as it allows for more accurate and personalized assessments of cancer dynamics and treatment responses. The following chapter will explore how statistical learning methods can be applied to gene expression data to identify and differentiate cancer stages, enabling more accurate and efficient detection and prognosis.

CHAPTER 3

Background for Statistical Learning

This chapter mainly discusses the fundamentals of statistics in particular classifiers, model evaluation, and feature selection methods. Feature selection techniques play important roles in high dimensional data by eliminating non-informative features or noise. Besides feature selection methods, the classifier is one of the types of machine-learning algorithms that is employed to classify data into one or more target classes. For example, Random Forest (RF), k-Nearest Neighbour (k-NN), Support Vector Machine (SVM), and eXtreme Gradient Boosting (XGBoost) are classification models that assign samples to target classes across a specific set of features. These classifiers are clarified in Section 3.1, which includes definitions, relevant theories, or their applications. Assessment of predictive ability is a critical component of evaluating machine learning models which is included in Section 3.2. In addition to classifiers and model evaluation, selecting relevant features is a crucial step in the process of building effective predictive models. Several statistical methods are employed to identify the most informative features that contribute to improving model performance. For instance, Wilcoxon Rank Sum Test (WRS), Kruskal Wallis Test (KW), Least Absolute Shrinkage Selector Operator (Lasso), Minimum Redundancy and Maximum Relevance (mRMR) and Proportional Overlapping Scores (POS) are feature selection approaches that can be handled with binary or multiple class problems. The details of these feature selection techniques are discussed in Section 3.3.

3.1 Classification Models

Machine learning has become an important tool for improving performance and making accurate predictions. It is typically divided into four different learning methods: supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. The supervised learning technique has achieved significant success in classification and regression by learning from training data sets as input data. When the output consists of qualitative variables assigned as labels to the input, it results in a classification task. When the output takes quantitative variables, it leads to a regression analysis [77]. In supervised learning approaches for classification, five primary methods are discussed: Decision Trees (DT), Random Forest (RF), K-Nearest Neighbors (KNN), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost).

3.1.1 Decision Trees

Decision Trees (DT) are a set of rules which are used to classify data into categories by a two-step process: learning step and prediction step. In the learning step, the model is developed based on given training data while the model is used to predict the response for given data in the prediction step. There are three different elements; root node; decision node; leaf node. 'Root node' represents the entire population, sample, or objective which is divided into two homogeneous sets. 'Decision nodes' show a decision to be made or split into further sub-nodes. 'Leaf nodes", also called terminal nodes, are nodes which are not split into more nodes. In other words, leaf nodes represent classes which are assigned by majority vote. To build a decision tree, there are two main parameters which are used to consider splits: Entropy and Information Gain.

Entropy is used for calculating the homogeneity of a sample. When the entropy is zero, the probability can have the value of 0 or 1. Moreover, a split with an entropy of zero is a leaf node, while a split with an entropy more than zero needs further splitting [170]. For a single attribute, the entropy is defined as:

$$E(S) = -\sum_{i=1}^{c} p_i log_2 p_i \tag{3.1}$$

where S is current state, and p_i is probability of an event i of state S or percentage of class i in a

node of state S . For multiple attributes, the entropy is given by:

$$E(T,X) = \sum_{c \varepsilon X} P(c)E(c) \tag{3.2}$$

where T and X are current state and selected attributes, respectively. $P(c)$ is the probability or proportion of the occurrence of class c within the attribute space X, and $E(c)$ is the entropy computed for class c.

Information Gain (IG) is a statistical property that is employed to determine the order of attributes in the nodes of a decision tree. In other words, it measures how well a given attribute separates the training examples according to their target classification. Constructing a decision tree is all about finding an attribute that turns the highest information gain and the smallest entropy [152]. The formula used for obtaining the information gain is written as:

$$IG(T,X) = E(T) - E(T,X) \tag{3.3}$$

3.1.2 Random Forest

Random Forest (RF) is one of machine learning techniques that is used for solving regression and classification problems. A random forest consists of multiple decision trees in different samples and takes their majority vote for classification and average in case of regression. In Random Forest, bagging, also called Bootstrap Aggregation, is used for choosing a random sample from the data set. Therefore, each tree is build from the samples (Bootstrap Samples) provided by the original data with replacement. This step is called 'bootstrap'. Each bootstrap is then trained independently to generate results. In the final output, a majority voting or an average are employed to combine the results of all models for classification and regression, respectively as Figure 3.1. These steps involving combining all the results and generating output based on majority voting or average are known as *aggregation* [110].

To build a random forest, there are two main parameters which are *ntree* and *mtry*. *ntree* is the number of trees to use for growing a tree, while *mtry* is the number of variables randomly sampled as candidates at each split. In addition, *mtry* for classification is $sqrt(p)$ and *mtry* for

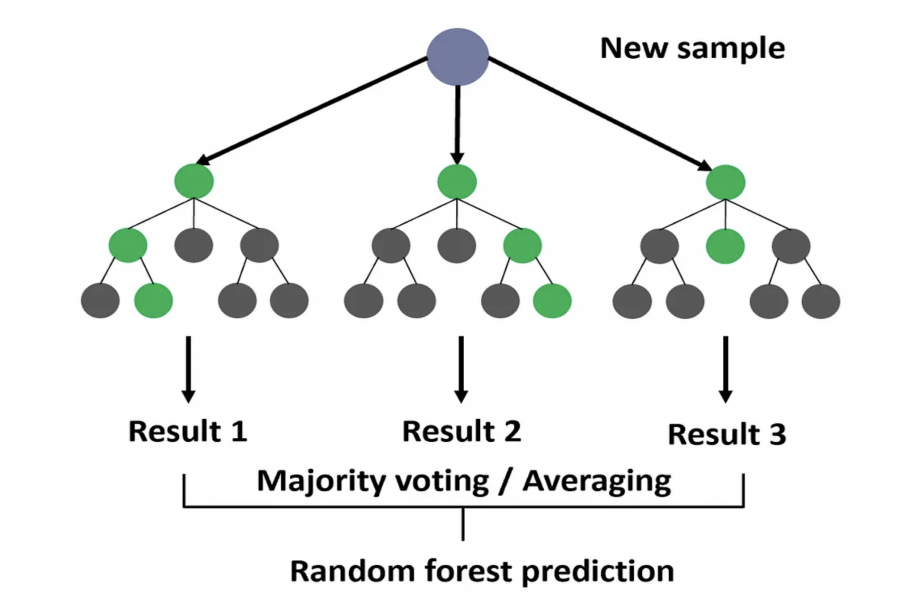

Figure 3.1: The random forest classifier [189].

regression is $p/3$ where p is number of features [22].

The key differences between a decision tree and a random forest are that the random forest addresses the issue of overfitting by relying on majority voting or averaging for its output. Additionally, the random forest provides more stable results, as the final predictions are based on the average of multiple trees.

3.1.3 K-nearest neighbor

The k-nearest neighbor algorithm, known as the K-NN algorithm, is a simple method which is commonly used for classification and regression in supervised machine learning algorithms. The K-NN is one of non-parametric methods that does not rely on any assumptions, moreover, this method sometimes is called a lazy learning because it does not implement the training data points to make any generalization or it learns a discriminative function from the training data but *memorizes* the training dataset instead. In this thesis, we are focusing on the K-NN classification in which this technique is built by identifying the nearest neighbors to a query example and using those neighbors to determine the class of the query by using majority votes. The K-NN is simple and easy to implement but there are few drawbacks of using this method as the K-NN does not work effectively with large datasets and high dimensions. For high dimensions, using the K-NN becomes difficult for the algorithm to calculate the distance in each dimension. These drawbacks

cause a high computational cost at the classification time [100].

Next, we are going to discuss the main stages of using the K-NN, and steps of implementing K-NN using the R programming package. As we know the K-NN classification classifies instances based on their similarity to instances in the training data. There are three main stages to consider for the K-NN algorithm which are rescaling features, determining the optimal k value, and choosing the distance metric. For the rescaling features, this stage is to transform all features into a standard range before applying the K-NN algorithm.The main benefit of rescaling values is that it avoids the problem of wrong interpretation because there are no features that have much larger values than others, and so the distance measurement will not be dominated by the larger values. There are two traditional methods of rescaling features that are the min-max normalization and $z-score$ standardization. The Min-Max normalization is a normalization method that performs linear transformations of the original data to values between 0 (the minimum) and 1 (the maximum). The formula of this method is written as

$$X_{new} = \frac{X - min(X)}{max(X) - min(X)} \tag{3.4}$$

where X_{new} is the new value from the normalized results, X is old value, $max(X)$ is the maximum value, and $min(X)$ is the minimum value [80]. Another rescaling method is $z-score$ standardization. The $z-score$ standardization is a method of normalization based on the mean value and standard deviation of the data. The $z-score$ values are ranged between infinite negative and positive numbers unlike the Min-Max normalization. This method becomes useful when the actual minimum and maximum values are not known. The formula of calculating the $z-score$ is

$$Z = X_{new} = \frac{X - \mu}{\sigma} \tag{3.5}$$

where Z is the new value from the normalized results, X is the old value, μ is population mean, and σ is standard deviation value [89]. The second stage is to determine the optimal K value. This is one of the most important stages since we must define how many neighbors to use for the K-NN in order to provide the best performance for the future data. Choosing a large K leads to a high bias and a low variance, while a small K provides a low bias and high variance. Therefore

the best K value is somewhere between these two extreme conditions. Figure 3.2 shows the examples of determining different k values. For the optimal K, it ensures high performance in identifying the correct neighbors. However, both small and large values of K can lead to poor classification: a small K may be overly sensitive to noise points, while a large K may result in the neighborhood containing too many points from other classes. [145] suggests a small K is not always good for a small data set as well as a large K is not always suitable for a big data set. In the machine learning classification of the k-NN, the square root of the size of the training set is commonly used as the optimal K value [57, 75, 194, 51].

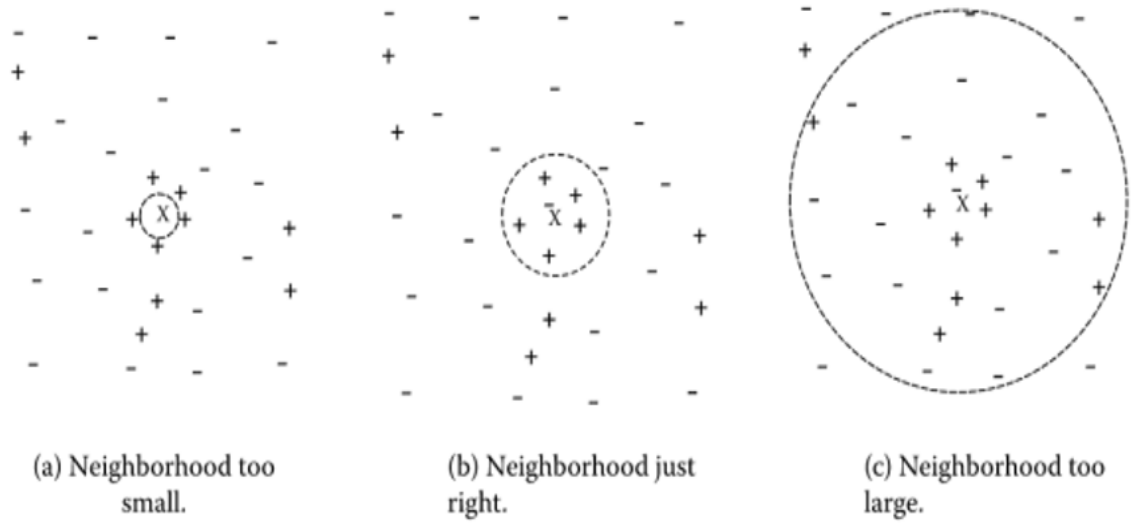

Figure 3.2: k-nearest neighbor classification with small, medium and large k [167].

Lastly, distance metrics are a method to find distance between a new data point and existing training dataset. There are four main measurements that are commonly used in K-NN; namely Euclidean distance, Manhattan distance, Minkowski distance, and Chebyshev distance. Euclidean distance is the most commonly used distance metric in K-NN which represents distance between two points. In other words, the two points in the Euclidean space are defined as the length of the line segment between two points. The Euclidean distance is referred to as the Pythagorean distance because using the coordinate points can be found as Pythagoras theorem. The formula for calculating the distance between the two variables is defined as

$$d_x, y = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2} \tag{3.6}$$

where d_x, y is the Euclidean distance, (x_1, y_1) is the coordinate of the first point, and (x_2, y_2) is the coordinate of the second point. Moreover, the range of values can be from 0 to infinity [129].

Minkowski distance or p-norm distance is the generalized form of Euclidean distance. it is written as

$$d_(x,y) = \left[\sum_{i=1}^{n} |x_i - y_i|^p\right]^{(1/p)} \tag{3.7}$$

where $(x_1, x_2, ..., x_n)$ and $(y_1, y_2, ..., y_n)$ are two vectors in n-dimensional Euclidean space, and p is the positive real number. When p is 1,it represents the Manhattan distance. When p is 2, it indicates the Euclidean distance, while p is infinity represents the Chebyshev distance [120].

Manhattan distance is used to estimate the distance between two points or vectors. In other words, it is calculated as the sum of absolute differences between points across all the dimensions. The formula for calculating the distance between the two variables is represented as

$$d_(x,y) = \sum_{i=1}^{n} |x_i - y_i| \tag{3.8}$$

where $d_{x,y}$ is the Manhattan distance, n = number of dimensions, and x_i and y_i are the data points [162].

Chebyshev distance is also called maximum value distance. It calculates the absolute magnitude of the differences between two points. Moreover, ordinal and quantitative variables can be used for this method. the formula is expressed as

$$d_\infty(x,y) = \max_{i=1}^{n} |x_i - y_i| \tag{3.9}$$

where $d_{x,y}$ is the Chebyshev distance, n = number of dimensions, and x_i and y_i are the data points [157].

This section describes how to implement the K-NN algorithm in the R programming environment. Firstly, class is imported as a library and data is loaded. The second step is to normalize the variables to the same scale in order to avoid overfitting or getting poor model performance. Next, the data is spit into two sets which are the training and testing dataset. The training set is used to build the K-NN mode by using the kNN() function [156], while the testing set is employed to

evaluate the model providing accuracy scores at the end from calculating confusion matrix by using table() function. This function will compare the observations between the testing datasets and training datasets, so the performance of the K-NN model can be evaluated. Moreover, to build the model, K is initialized to be a chosen number of neighbors and then the Euclidean is used to measure the distance between the query example and the current example from the data. Next, it is essential to sort the calculated distances in ascending order based on their values. After sorting, the first K entries from the sorted list are selected, and the corresponding labels of these K entries are retrieved. The final step is to return the mode of the K labels for classification. In other words, the majority or distance weighted voting is used to obtain the final outcome for classification.

3.1.4 Logistic Regression

Logistic Regression is one of the supervised learning methods in machine learning which is used for classification problems to predict the probability of target value. This method is employed when the dependent variable is categorical and the independent variable is continuous or categorical. Therefore, there are three different types of logistic regression; namely Binary logistic regression, Multinomial Logistic Regression, and Ordinal Logistic Regression. The binary logistic regression is used when the dependent variable has only two types of values: Disease /Non-disease, Yes/No and 0/1. If the dependent variable has more than two types of values and these values are not in an order, the multinomial logistic regression is then involved. To conduct logistic regression, we need to consider four different assumptions which have to be met. The first assumption is that the observations are independent of each other. In other words, the observations should not be repeated measurements of the same individual or be related to each other. The second assumption is linearity which means that there is a linear relationship between each explanatory variable and the logit of the response variable. The third assumption is the absence of severe multicollinearity among the explanatory variables. The last assumption is no extreme outliers or influential observations in the dataset. If there are too many outliers, the model's overall accuracy could be compromised. Therefore, outliers should be eliminated in data processing [169].

Binary logistic regression is a type of regression analysis which it measures the relationship between the categorical target class and independent variables. This method is suitable for situations in which the outcome variable is binary class (0,1), while the predictor variables can be continuous or categorical. In the case of binary logistic regression, the target variable is binary (0,1) as Figure 3.3.

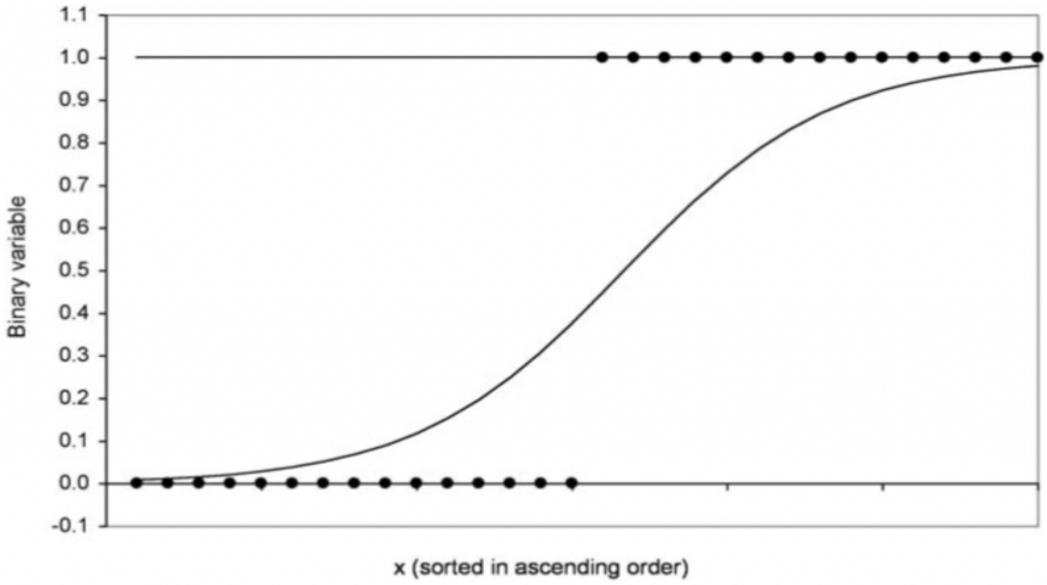

Figure 3.3: Example distribution with logistic function [121].

The logistic model's fitting or prediction of the value is based on the link function, $log(p/(1-p))$. In order to establish a linear relationship between the predicted value, p, and the linear predictor, the following equation is used:

$$ln\frac{p}{1-p} = x_i b = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + ... + \beta_p x_{ip} \tag{3.10}$$

where p is the probability of response value y that is equal to 1. Notice that $p/(1-p)$, is the formula for odds. The odds are the ratio of probability of its success divided by the probability of its failure [82]. The log of the odds has been called the logit function. In order to determine p on the basis of the linear predictor, xb, we solve the logit function for p as

$$p = \frac{exp(xb)}{1+exp(xb)} \tag{3.11}$$

To estimate the parameters or coefficients, we need to use Maximum Likelihood Estimation. The

Maximum Likelihood Estimation (MLE) is a method of estimating the parameters of a logistic regression model. In other words, it can be said that this method finds the best fitting line in a logistic regression model. A main reason why the MLE is used to calculate parameters in logistic regression is that the y-axis is transformed from the probabilities to log(odds). Therefore, the least-squares can not be used to find the best fitting line because the residuals are also equal to positive and negative infinity. The maximum likelihood method is the most popular method to estimate the parameter which relates to a probability function of a discrete stochastic variable X. It is based on the observations $x_1;x_2;...;x_n$ which are independently sampled from the distribution. Note that for a continuous stochastic variable X, the probability density function is indicated as

$$P(X < r|\theta) = \int_{-\infty}^{\infty} rp(x|\theta)dx \tag{3.12}$$

The maximum likelihood estimate is the value which maximize the likelihood function that is defined by

$$L(\theta) = \Pi_{i=1} nP(X = x_i|\theta) = P(X = x_1|\theta)P(X = x_2|\theta)...P(X = x_n|\theta) \tag{3.13}$$

when X is a discrete stochastic variable and

$$L(\theta) = \Pi_{i=1} nP(x_i|\theta) = P(x_1|\theta)P(x_2|\theta)...P(x_n|\theta) \tag{3.14}$$

when X is a continuous stochastic variable. That is, the maximum likelihood estimation chooses the model parameter which is the most likely to generate the observed data [131].

After building the model with the Maximum likelihood, the next important step is to examine the significance of the coefficients. There are several approaches to test these coefficients, however, we exploited two common tests; wald test and likelihood ratio test. For the hypothesis test in logistic regression, the null hypothesis is the coefficient of the independent variable is equal to zero, while the alternative hypothesis is the coefficient is nonzero - that is $H_0 : \beta 1 = 0$ versus $H_1 : \beta 1 \neq 0$.

Lets start with the Wald statistic. The Wald statistic or Wald χ^2 statistics is obtained by dividing the maximum likelihood estimate (MLE) by the estimate of its standard error or it is

calculated as follows.

$$Wald = (\frac{\hat{\beta}}{SE(\hat{\beta})})^2 \tag{3.15}$$

where $\hat{\beta}$ is the estimated coefficient and $SE(\hat{\beta})$ is the standard error of the coefficient. Each Wald statistic is compared with a χ^2 distribution with 1 degree of freedom. Wald statistics are easy to calculate but it is not good when the samples are small. For data that produce large estimates of the coefficient, the standard error is often inflated, leading to a lower Wald statistic, and therefore the explanatory variable may be incorrectly assumed to be unimportant in the model. Therefore, likelihood ratio tests (LRT) are generally considered to be superior [15].

LRTs are used to test the hypothesis that an independent variable is zero by comparing the likelihood of obtaining the data when the parameter is zero (L0) with the likelihood (L1) of obtaining the data evaluated at the MLE of the parameter. The LRT test statistic is

$$-2 * ln(likelihoodratio) = -2 * ln(L_0/L_1) = -2 * (lnL_0 - lnL_1) \tag{3.16}$$

It is compared with a χ^2 distribution with 1 degree of freedom. Moreover, the degrees of freedom for the chi-square distribution is the difference between the number of parameters in the full model and the number of parameters in the reduced model [59].

After building the model and examining the coefficients, the next step is to assess how good the model fits the data. In this section, three approaches are proposed to evaluate the model, that are Hosmer-Lemeshow Test, Pseudo R^2, and classification model accuracy.

The Hosmer-Lemeshow Test (HL test) is a commonly used goodness of fit test for evaluating logistic models, especially binary outcomes. This approach aims to examine that the specified model is correct as the null hypothesis, while the specified model is incomplete as the alternative hypothesis. This test is implemented by sorting the n instances in the data set according to the value of the estimated success probability and splitting the sorted data set into m groups. Then the test statistic is written as

$$T_m = \sum_{j=1}^{m} \frac{(e_i - o_j)^2}{ne_j(1 - e_j)} \tag{3.17}$$

where e_j is the sum of the estimated success probabilities of the jth group while o_j is the sum of the observed success items of the jth group, and the term $e_j = ne_j$ is the mean of the estimated

success probabilities of the jth group [84].

Next, Pseudo R^2 is used to evaulate *goodness of fit* in logistic regression models instead of R^2 in linear regression analysis. The reason why the Pseudo R^2 is developed is because the model estimates from a logistic regression are maximum likelihood estimates arrived at through an iterative process. Therefore, the Pseudo R^2 are proposed to measure the goodness of fit for models with binary or multinomial outcome.There are several pseudo R^2 which have been developed to evaluate models with categorical outcome that are McFadden, McKelvey and Zavoina, Maddala, Agresti, Nagelkerke, Cox and Snell, Ash and Shwartz, Zheng and Agresti. For two common examples, consider the following;

Cox & Snell Pseudo R^2

$$R^2 = 1 - [\frac{-2LL_{null}}{-2LL_{full}}]^{2/n} \tag{3.18}$$

where -2LL(Null) and 2LL(Full) are the likelihood functions for the intercept-only model and full model, respectively. Because the Cox & Snell R^2 value cannot reach 1.0, Nagelkerke modified it by rescaling this statistic. The correction increases the Cox and Snell version to make 1.0 a possible value for R^2 [9].

Nagelkerke Pseudo R^2

$$R^2 = \frac{1 - [\frac{-2LL_null}{-2LL_full}^{2/n}]}{1 - (-2LL_{null})^{2/n}} \tag{3.19}$$

where the re scaling has been conducted by dividing Pseudo R^2 of Cox & Snell by its maximum possible value. Therefore, the range of OLS R^2 is produced from this resulting statistic [164].

3.1.5 Support Vector Machine

One of the most common classifiers is the Support Vector Machine (SVM). An SVM model is a machine learning algorithm that employs a supervised learning model to deal with regression and classification. The main task of the SVM algorithm is to assign a hyperplane that can separate the data points of different class problems. There are three important components in drawing the SVM model namely Hyperplane, Support Vectors, and Margin. As illustrated in Figure 3.4, hyperplanes are flat lines that separate data into groups of identical elements, while support vectors are the points that are nearest to the hyperplane that aid in determining the data points.

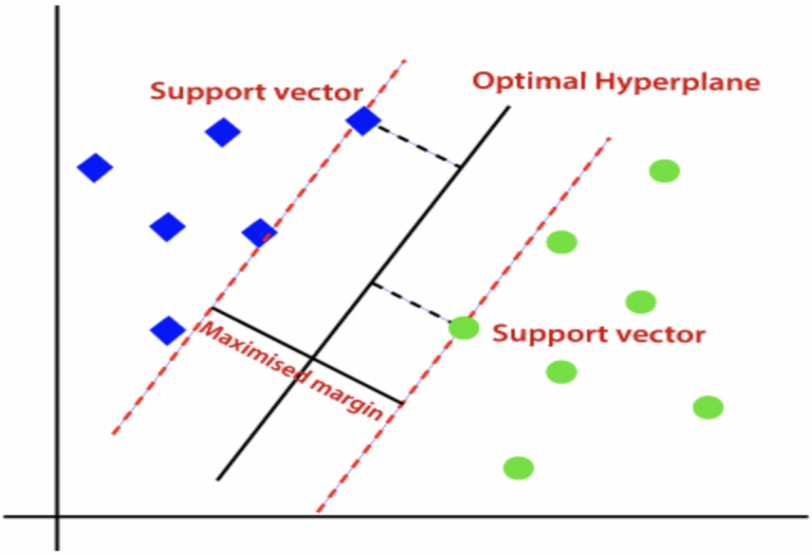

Figure 3.4: The representation of SVM [4].

Besides hyperplanes and support vectors, the margin is the area between the hyperplane and support vectors. A larger margin represents a better hyperplane, and this is the main key of an SVM model.

In real-world applications, most of the data sets are faced with non-linear relationships between variables. A SVM model is capable of training on such data based on the condition of the slack variable, and it also allows some samples to be misclassified. Therefore, another main key of SVM is to convert lower dimensional space to a higher dimensional space which helps us to find a decision boundary to classify the data points. This process is known as the **kernel tricks**. Figure 3.5 (a) demonstrates the data of two classes indicating red and blue points and it can be observed that this data is not linearly separable in the 2-dimensional space. To deal with this situation, the data is applied with kernel trick to transform data from 2-dimensional space to 3-dimensional space and these data points can be separated by a hyperplane as illustrated in Figure 3.5 (b).Therefore, kernel tricks are efficient techniques that can solve non-linear classification problems. Four popular kernels are widely applied as kernel functions: linear kernel, polynomial kernel, sigmoid kernel, and Gaussian RBF kernel [193].

The **linear kernel** is the easiest of all the kernels since it is a special case of polynomial kernel with a degree of 1 and a coefficient of 0. Additionally, the x and y use the inner product

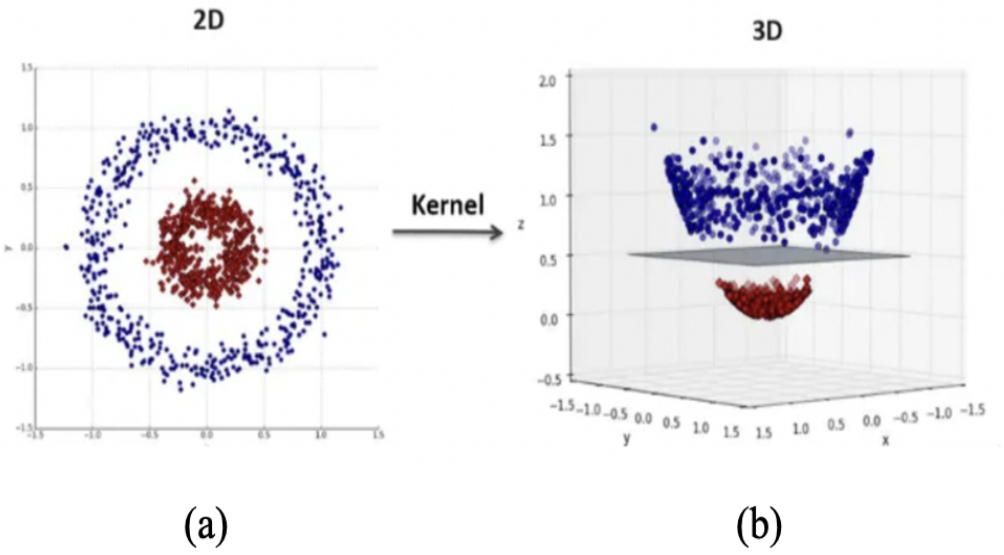

Figure 3.5: Transforming the data from 2-dimensional space to 3-dimensional space [188].

and the linear kernel can be defined as:

$$k(x,y) = x^T y \tag{3.20}$$

where x and y are the input vectors.

The **polynomial kernel** is a simple non-linear transformation of data and calculates the degree-d polynomial kernel between two vectors. The polynomial kernels consider the similarity between vectors under the same dimension and across dimensions. The polynomial kernel is expressed as:

$$k(x,y) = (\gamma x^T y + c_0)^d \tag{3.21}$$

where x and y are the input vectors. The d is the kernel degree, while If $c_0 = 0$ the kernel is said to be homogeneous.

The **sigmoid kernel** uses two vectors to compute the sigmoid kernel. The sigmoid kernel is called a hyperbolic tangent, or Multilayer Perceptron. The sigmoid kernel is denoted as:

$$k(x,y) = tanh(\gamma x^T y + c_0) \tag{3.22}$$

where x and y are the input vectors. γ and c_0 are slope and intercept, respectively.

The **RBF kernel** takes two vectors to compute the radial basis function (RBF) kernel. The RBF kernel is defined as:

$$k(x,y) = exp(-\gamma \parallel x-y \parallel^2) \tag{3.23}$$

where x and y are the input vectors. If $\gamma = \sigma^{-2}$, the kernel is known as the Gaussian kernel of variance σ^2.

3.1.6 eXtreme Gradient Boosting

XGBoost, eXtreme Gradient Boosting, is a supervised machine learning algorithm that creates a series of models and combines them to create an overall model that is more accurate than any individual model in the sequence. This technique can be used in regression and classification, moreover, it is designed for speed, ease of use, and performance on large datasets.

In order to minimise overfitting, XGBoost finds the optimal answer by taking the loss function's Taylor expansion up to the second order and adding a regularisation term. This balances the objective function's decline with the model's complexity [184]. The XGBoost model can be defined as;

$$\hat{y_i} = \sum_{d=1}^{D} f_d(x_i) \tag{3.24}$$

Let $f_d \varepsilon C$ denote the function corresponding to the $d-th$ decision tree, where C represents the set of all possible Classification and Regression Trees (CART). Given an input x_i, the function $f_d(x_i)$ outputs the prediction from the $d-th$ tree. $\hat{y_i}$ expresses the predicted value. Two terms are included for the objective function of XGBoost includes two parts that are training error and regularization. It is defined as following;

$$X_{obj} = \sum_{i=1}^{n} R(y_i, \hat{y_i}) + \sum_{d=1}^{D} \Omega(f_d) \tag{3.25}$$

where $\sum_{i=1}^{n} R(y_i, \hat{y_i})$ is employed to measure the difference between the predicted value and the real value of the loss function. $\sum_{d=1}^{D} \Omega(f_d)$ defines the regularization term in which

$$\Omega(f_d) = \gamma N + \frac{1}{2}\lambda \parallel \omega \parallel^2 \tag{3.26}$$

N is the number of leaf nodes, Ω represents the scores of leaf node, γ expresses the leaf penalty coefficient, and λ ensures that the scores of the leaf node is not too large.

In order to continually repair the previous test results by fitting the residuals of the last prediction, the XGBoost method employs the gradient boosting strategy, adds one new tree at a time rather than obtaining all the trees at once. For the t-th decision tree, the objective function can be updated as

$$L^{(D)} = \sum_{i=1}^{n} R(y_i, \hat{y}_i^{D-1}) + \Omega(f_d) \tag{3.27}$$

During the training phase, the leaf node with the largest gain loss is selected by computing the node loss. New trees are added by continuously splitting features. Each time a new tree is added, the residual of the previous prediction is fitted by learning a new function $f_d(X, \theta_D)$. The features of the prediction samples will have a corresponding leaf node in each tree when D trees are created after training, and each leaf node corresponding to a score. Eventually, the sample's recognition prediction value is calculated by adding the matching scores for each tree [32].

3.2 Model Performance

Classification model accuracy is used to evaluate the performance of a model. The model prediction is estimated by accuracy and prediction errors using a testing data set. The performance of the predictive model can be assessed by comparing the predicted outcome values against the known outcome values; confusion matrix. The confusion matrix is a two by two table formed by counting the number of the four outcomes of classification as follows.

- True positive (TP): correct positive prediction
- False positive (FP): incorrect positive prediction
- True negative (TN): correct negative prediction
- False negative (FN): incorrect negative prediction

Table 1 shows the confusion matrix of binary classification or two by two confusion matrix. From this table, we can perceive accuracy (ACC) by calculating the number of all correct

		Predicted	
		Positive	Negative
Observed	Positive	TP	FN
	Negative	FP	TN

Table 3.1: The two by two table of confusion matrix.

predictions against the total number of the dataset. The formula is written as

$$ACC = \frac{TP+TN}{TP+TN+FN+FP} = \frac{TP+TN}{P+N} \tag{3.28}$$

where TP is true positive, TN is true negative, FP is false positive, and FN is false negative. The best accuracy is 1.0, while the worst is 0.0 [155].

3.3 Features Selection Methods

In recent years, the data and information have been stored easily and it is efficient to build the model. There are hundreds or thousands of features or variables in the data to be employed to consider model performance. To implement with a large number of features or all available features would lead to inefficiency performance due to over-fitting, high maintenance workload, model interpretation difficulty, noise data, missing data and irrelevant information. Therefore, feature selection is proposed to deal with these problems. Feature selection is one of the important methods which is used to improve the prediction accuracy by reducing irrelevant and redundant features in order to provide better model interpretation. There are a number of benefits of feature selection methods. Firstly, it reduces the computational cost by dimension reduction. For example, a smaller set of features accelerates in the model training and validation process. Secondly, it improves the classification accuracy by reducing irrelevant features, avoiding overfitting, and fitting more training samples into models by using the number of relevant features. Lastly, it produces more interpretable features that can assist in identifying and monitoring the target classes in terms of diseases or function types, in other words, it only includes the important feature set which leads to easier interpretation of what features and information are based on. There are three categories in feature selection technique; filter

methods, wrapper methods, and embedded methods.

Filter Methods are used to assess the relevance of features by considering the properties of the data/information. A feature relevance score is calculated and then the lower-scoring features are eliminated. Afterwards, the selected features with the high score are implemented as relevant features to classification algorithms. In other words, filter methods select features by ranking them with correlation coefficients. The advantages of filter methods include their ability to handle high-dimensional datasets with ease, their simplicity and speed in computational processes, and their independence from the classification algorithm. However, filter methods have some common drawbacks. They overlook the interaction with the classifier, and most of the proposed techniques are univariate. Each feature is considered separately, which may lead to worse classification performance. A number of multivariate filter techniques were introduced to overcome the problem of ignoring feature dependencies by aiming at the incorporation of feature dependencies to some degree [163].

Wrapper Methods use a predictive model which is run on training and testing sets to evaluate gene subsets. Each gene subset is employed with a training dataset to train the model, and the testing dataset then validates the model. Model prediction errors are calculated from the testing dataset and given a score for the gene subset. The gene subset with the highest performance is selected as the final set to run the particular model [31]. In other words, it can be said that wrapper methods assess the subset of features based on their usefulness to a given classifier. There are three common techniques which are used under wrapper methods:Forward selection, Backward elimination, and Recursive feature elimination. Forward selection is an iterative method in which it is begun without a feature in the model. In each iteration, the best feature that improves the performance is added until there is no additional feature which improves the performance of the model. Backward elimination starts with all the features and removes the least significant feature at each iteration, which improves the performance of the model. It is repeated until no improvement is observed on the removal of features. Recursive feature elimination is a greedy optimization algorithm that aims to find the best performance of a feature subset. In each iteration, it repeatedly builds models and leaves out the best or the worst features. It constructs the next model with the left features until all the features are exhausted.

Then it ranks the features according to the order of elimination. The advantages of using wrapper methods include improved performance by considering feature dependencies. However, the disadvantages are that they are computationally expensive, as a new model must be fitted for each gene subset, and they are prone to overfitting [119].

Embedded Methods combine the qualities' of filter and wrapper methods. The search for an optimal subset of features is built into the classifier construction and can be seen as a search in the combined space of feature subsets and hypotheses. Like wrapper approaches, embedded approaches are thus specific to a given learning algorithm. The advantages of embedded methods include the interaction with the classification model and it is less computationally expensive than the wrapper method by simultaneously integrating models with feature selection. Moreover, it is less prone to overfitting [108].

3.3.1 Wilcoxon Rank Sum Test

The Wilcoxon Rank Sum Test (WRS) is a non-parametric statistical method used to compare two independent populations when the observations are either ordinal or continuous measurements. Due to non-parametric statistics, the Wilcoxon Rank Sum Test does not assume assumptions such as normality and equal variance. For the hypothesis test, the null hypothesis (H_0) is that there is no difference between samples of group 1 and group 2, while the alternative hypothesis (H_1) is that there is a difference between samples of group 1 and group 2. For test statistics, let n_1 and n_2 are from one population and a second population, respectively. There are N observations in all, where $N = n_1 + n_2$. Rank all N observations. The sum of the ranks for smaller populations is the Wilcoxon rank sum statistics. When the two populations have the same continuous distribution, then W has mean

$$\mu_W = \frac{n_1(N+1)}{2} \tag{3.29}$$

and standard deviation

$$\sigma_W = \sqrt{\frac{n_1 n_2(N+1)}{12}} \tag{3.30}$$

The Wilcoxon Rank Sum Test rejects the null hypothesis that the two populations have similar distributions when the rank sum W is far from its mean [133]. In other words, the Wilcoxon

Rank Sum Test involves three main steps: First, combine all observations from the two samples and rank them in ascending order of value. If there are tied values, each tied observation is given the average rank. Then, sum the ranks corresponding to the observations from the smaller group, which results in the Wilcoxon statistic. Finally, the $p-value$ associated with the Wilcoxon statistic is obtained from the Wilcoxon rank sum distribution table or by using statistical software such as R, MATLAB, or SAS.

For gene expression data, the significance level threshold will be applied after testing two populations using the Wilcoxon Rank Sum Test. The significance level threshold is used to find how many information genes appear in the data which are associated with a $p-value$. The idea is to take a significance level threshold α_{max}, and then only the genes whose $p-values$ that are less than the threshold are selected. There are four main steps of the informative gene selection as follows.

1. Define a significance level threshold α_{max}(e.g. 0.01), to indicate the quality requirement of the informative gene selection;

2. Compute the Wilcoxon-statistic for every gene;

3. Use the statistics to compute $p-values$;

4. Select the genes whose $p-values$ are smaller than the significance level threshold α_{max}, which means the distributions between phenotypes are not identical [46].

After getting informative genes, the selected genes are ranked in order of their $p-values$ and select top k genes from them as the top-ranking genes [161].

3.3.2 Kruskal Wallis Test

The Kruskal–Wallis test is one of the statistical methods that is utilized to compare more than two groups for a continuous or discrete variable in which each group has an independent measure condition. In addition, this test is non-parametric, so, there are no requirements such as normality and difference of variance in data [126]. To investigate differences among groups, two hypotheses are employed for consideration: the Null and Alternative hypotheses. The null hypothesis (H_0)

assumes that the observations are from identical populations, while the alternative hypothesis (H_1) considers the observations are from different populations. Due to the similarity between the Kruskal-Wallis test and Wilcoxon's Rank Sum test, comparing the sum of ranks is applied to the data. The statistical test is expressed as

$$K = \frac{N}{N(N+1)} \sum_{i=1}^{g} \frac{R_i^2}{n_i} - 3(N+1) \tag{3.31}$$

where K is the Kruskal-Wallis statistical Test and N is the total number of samples in all groups. g is the number of groups, while R_i^2 and n_i are the rank total for each group and n_i is the number of samples in each group, respectively [83]. There are 5 main steps for verifying the hypothesis:

1. Arranging the data from entire samples in a single series in ascending order and assigning rank to them in ascending order. If there are tie/repeated values, it requires taking the average their rank position.

2. Summing up the different ranks for each of the different groups.

3. Calculating statistical test by using equation 3.31

4. Assessing the significance of K depends on the number of samples and the number of groups. The Kruskal-Wallis statistical test is approximately a chi-square distribution, with $g-1$ degrees of freedom where n_i should be greater than 5. If the K value is less than the critical chi-square value, then the null hypothesis is not rejected. If the K value is greater than the critical chi-square value, then the null hypothesis can be rejected. Therefore, the sample comes from a different population.

5. Taking the consequence from Step 4 to provide a conclusion.

A significant remark is that there is a function in R programming providing the K statistic and the p-value; kruskal.test (). If a small p-value is less than 0.05, it leads to reject the null hypothesis. It can be observed that at least one of our groups likely originates from a different distribution than the others. In contrast, If the p-value is greater than 0.05, the null hypothesis is not rejected. It means that there is no difference among groups [85].

For gene expression, each gene generates the p-value through the Kruskal-Wallis test and these p-values are sorted in ascending order in which are ranked from the smallest p-vales to the highest p-values. The p-value is utilised to investigate the differences among phenotypes. If a gene has a small amount of p-value, it means that there are differences among multiple class problems, or it represents the high discriminative power to distinguish correct target classes. Therefore, a smaller p-value represents a higher informative gene. Thanks to ranking the p-values, top n genes are selected as a set of informative genes to build models based on classifiers. Each model with a particular set of informative genes provides classification accuracies.

3.3.3 Least Absolute Shrinkage Selector Operator

In the real world data, we can not deny the fact that the models fit well on data by using only the least sqaure estimator or maximum likelihood estimator. Most of the time, when we try to fit the models, we will face the problem of overfiting and high variance; when the model fits quite well on the training data but it performs worse in testing data. Therefore, we need to consider alternative fitting methods to deal with these situations and yield better prediction accuracy and model interpretability. There are three important approaches to reduce overfitting, that are subset selection, shrinkage, and dimension reduction. The subset selection involves a particular subset of p predictions to fit a model using least squares, while the shrinkage involves fitting a model with all p predictors, but the coefficients are shrunken toward zero. Nevertheless, dimension reduction involves projecting the p predictors into a M dimension subspace and these M projections are employed as predictors to fit a linear regression model at the end. In this section, we are going to discuss shrinkage methods. Before diving into shrinkage methods, let's revise the residual sum of squares.

Residual sum of squares or RSS is a statistical indicator that is used to measure the amount of variance. In other words, it is used to decide how well a statistical model fits on the data and it estimates the variance in the error term by calculating the distance between actual points and predicted point as Figure 3.6. For the linear regression model, the coefficients are estimated by the least squares method and it aims to minimize the residual sum of squares. The RSS can be

written as

$$RSS(\beta) = \sum_{i=1}^{n}(Y_i - \beta_0 - \beta_1 X_{i1} - ...\beta_p X_{ip})^2 \tag{3.32}$$

where the $B_0, B_1, ..., B_p$ are estimated coefficients, Y_i is the response target, and n is the number of observations. When the lower value of RSS represents the regression function is well-fit to the data, while the greater of the RSS means that the model fits the data poorly [132].

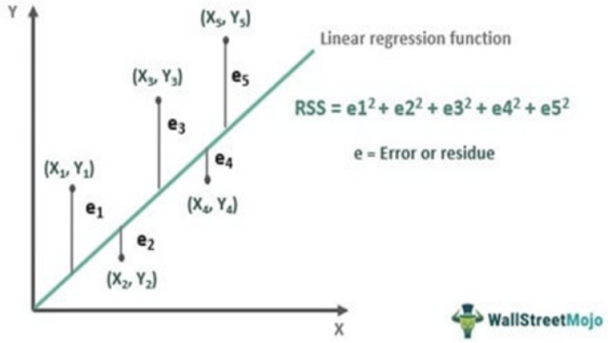

Figure 3.6: The calculation of RSS based on the graph [146].

Shrinkage Methods (also known as Regularizations) are an important technique which is used to reduce the error and avoid overfitting by fitting an appropriate function on the training set. Regularization adds penalties to models, making the models become more complex and then sorts potential models from least overfit to greatest. When the models have the lowest overfitting score that performs the best choice for predictive power. There are two important penalty terms that are used widely for regularization; $L1$ regularization and $L2$ regularization. $L1$ regularization adds an $L1$ penalty to the model that is the sum absolute value of the magnitude of coefficients,$\lambda \sum_{i=1}^{p} |\beta_j|$. $L1$ can perform sparse models, the models with fewer coefficients. In this process, some coefficients can be zero and these coefficients are eliminated from the model. This method is called LASSO. Whereas $L2$ regularization adds an $L2$ penalty to the model that is the sum square of the magnitude of coefficients,$\lambda \sum_{i=1}^{p} \beta_j^2$. $L2$ will not eliminate all coefficients but it will shrink all coefficients close to zero. This method is called RIDGE [23]. Since our work aims to select relevant features, therefore we are going to focus on LASSO.

Least Absolute Shrinkage Selector Operator (LASSO) is one of the most popular methods that is used for removing redundant or irrelevant features when there is collinearity or multicollinearity in the input values. The LASSO method can be used for the linear models and the generalized

linear models by performing two main tasks; shrinkage and feature selection. The LASSO can be written as

$$RSS + \lambda \sum_{i=1}^{p} |\beta_j| \tag{3.33}$$

where RSS is Residual sum of squares, $\lambda \sum_{i=1}^{p} |\beta_j|$ is the $L1$ regularization or a shrinkage penalty, and $\lambda \geq 0$ is a tuning parameter. When $\lambda = 0$, there is no effect in the penalty term, and the LASSO will produce only RSS. However, if $\lambda\beta$ is close to infinity, the impact of the penalty term grows, and the LASSO coefficient's estimation will become zero [93]. Therefore, choosing a good value for λ is essential and the best way to choose the optimal λ is to implement the training data with cross validation. The LASSO shrinks the coefficient estimates to be exactly zero when the λ is sufficiently large, then the features with coefficient equal to zero are excluded from the model. Therefore, the LASSO performs feature selection [58].

The main advantages of using the LASSO method are efficient prediction accuracy and increasing the model interpretability. For very good prediction accuracy, the LASSO method will shrink and remove the coefficients that can reduce variance without increasing the bias. For the model interpretability, the LASSO method will eliminate irrelevant features that are not associated with the response variables which reduces the problem of overfitting [137].

To implement LASSO with R program, there are two packages that are Glmnet and Lars. **Glmnet** (Lasso and Elastic-Net Regularized Generalized Linear Models) is a R package that is used to fit linear regression, logistic and multinomial regression models, Poisson regression, Cox model, multiple-response Gaussian, and the grouped multinomial regression. The cyclical coordinate descent is used in this algorithm in a path-wise fashion [61]. While **Lars** (Least Angle Regression, Lasso and Forward Stagewise) [76] is the newest model-selection method which is based on the traditional forward selection. It means that it selects the one having the largest absolute correlation with the response y. In the Lars package there is also an implementation of the LASSO method. In this thesis,we are going to use the Glmnet package with a cyclical coordinate descent on a grid of possible values for λ. The implementation of LASSO for feature selection involves two main steps.

1. The data is splitted into two sets, training data and testing data. The training data is used to find the best λ via cross validation by using the cv.lasso function.

2. The best λ is implemented in the LASSO using glmnet function to find k relevant genes.

3.3.4 Minimum Redundancy and Maximum Relevance

Minimum Redundancy and Maximum Relevance (mRMR) is one of the statistical approaches to deal with situations when there are a large number of features in data. This method will reduce redundancy and irrelevant features.

Initially, mutual information (MI) is introduced due to its key role in mRMR feature selection methods. MI is a valuable mathematical tool used to assess the relationship between features. In simpler terms, mutual information quantifies the degree of similarity between two variables, denoted as $I(X;Y)$. For continuous variables, mutual information can be expressed as:

$$I(X,Y) = \iint p(x,y) log \frac{p(x,y)}{p(x)p(y)} dxdy \tag{3.34}$$

When the MI among categorical variables can be indicated as

$$I(X,Y) = \sum\sum p(x,y) log \frac{p(x,y)}{p(x)p(y)} dxdy \tag{3.35}$$

where $p(x,y)$ is the joint probability density, while $p(x)$ and $p(y)$ are the marginal density function [196]. A large value of MI identifies a closer relationship between the two random variables which have larger correlation, while the value of MI is zero, it means that the two random variables are uncorrelated and independent of each other. Therefore, MI is employed to measure the similarity among features and the correlation between features and target classes in the mRMR.

The mRMR is proposed by [148] and it uses the MI as a measure standard for finding relevance features between features and target class and reducing redandancy among features. In other words, the mRMR framework is used for measuring the quality of a feature subset. The main goal is to find the feature subset with the highest correlation with the class while having the smallest redundancy among features. The mRMR is combined with two constraints that are minimum redundancy and maximum relevance. **Minimum Redundancy** is used to measure the quantity of the MI between features. If the value of MI is large, it means that there is a large

amount of information duplication between two features.

Whereas a lower value of redundancy measure means a better feature selection criterion. Therefore, a redundancy measure is used to find the feature which has the minimal value of MI among all features. The minimal redundancy condition is defined as

$$minR(S), R = \frac{1}{|S|^2} \sum I(x_i, x_j) \tag{3.36}$$

where $|S|$ is the number of features in features subset S, while $I(x_i, x_j)$ is MI between feature i and j. **Maximum Relevance** is to search features based on the MI. If the value of MI is small, it means that there is a little correlation between the feature and target class. On the other hand, the value of MI is large, it indicates that the feature has a greater amount of information to classify the target class. Therefore, it is crucial to select the maximum value of MI between the features and target classes. The maximal relevance criterion can be expressed as

$$maxD(S, c), D = \frac{1}{|S|} \sum I(x_i, c) \tag{3.37}$$

where $|s|$ is the number of features in features subset S and c is the target class, while $I(x_i, c)$ is the MI between feature i and the target class c. Moreover, $maxD(S, c), D = I(x_1, i = 1, 2, 3, ..., m; c)$ is used to find a feature set S with m features x_i, which jointly have the largest dependency on the target class c. These two constraints are combined in the simplest form to optimize D and R as

$$max\Phi(D, R), \Phi = D - R \tag{3.38}$$

or

$$f^{mRMR}(X_i) = I(x_i, x_j) - \frac{1}{|S|} \sum I(x_i, c) \tag{3.39}$$

The higher the value of f^{mRMR} , the higher the evaluation of the feature subset. There are three main steps for the incremental procedures of mRMR feature selection as follows:

1. In the original features set Ω, the optimal feature x_i can be selected by $I(x_i; c)$ and then put into the optimal features subset S;

2. In the features subset $\Omega_S = \Omega - S$, the next optimal feature x_j is selected which satisfies

equation 3.38;

3. Repeat Step 2 to identify the optimal subset of features, S, that satisfies the final size requirement [56];

The best benefit of the mRMR is to reduce mutual redundancy within the feature set; these features represent the class characteristics. The mRMR features improve prediction accuracy and provide better generalization properties. Moreover, the mRMR approach with fewer feature sets can effectively cover the same class characteristic space as more features in the baseline approach [49].

To implement mRMR with R program, there is a package that is mRMRe. This package consists of a set of functions to calculate mutual information matrices for continuous, categorical and survival variables. Moreover, it also performs feature selection with the mRMR and a new ensemble mRMR technique. The function that performs the mRMR feature selection is mrmr.classic and it returns the values such as paths, scores and mim. The paths are index vectors in which elements relate to the feature selected and the scores represent the score vector corresponding to the mRMR feature selection, while the mim is a mutual information matrix that is used for the mRMR feature selection [45].

3.3.5 Proportional Overlapping Score

Proportional Overlapping Scores (POS) [123] is a key feature selection method that identifies informative features through overlapping analysis. This technique computes a relevance score for each gene by evaluating the overlap between gene expression measures across different classes, taking into account three factors to calculate the POS score; (1) length of overlapping region; (2) number of overlapped samples; (3) the proportion of classes' contribution to the overlapped samples. The score of genes is ranked in ascending order and gene masks with their overlapping scores is considered to allow the detection of a minimum subset of genes. Combining the minimum gene subset with the top ranked genes provides the final gene set. Moreover, this method is appropriate for binary classes and it is defined based on the interquartile range to avoid outlier. First of all, the definition of core interval is proposed in order to determine mask genes and POS measure.

For the core interval, a gene i with binary classes can be indicated by two expression intervals;

$$I_{i,c} = [a_{i,c}, b_{i,c}], i = 1, ..., P, c = 1, 2 \tag{3.40}$$

which

$$a_{i,c} = Q_1^{(i,c)} - 1.5IQR^{(i,c)} \tag{3.41}$$

and

$$b_{i,c} = Q_3^{(i,c)} + 1.5IQR^{(i,c)} \tag{3.42}$$

where $Q_1^{(i,c)}$,$Q_3^{(i,c)}$, and $IQR^{(i,c)}$ are the first, third empirical quartiles, and the interquartile range of gene i expression values for class c respectively. the value of 1.5 is multiplied in the equation for detecting the outlier. Each gene mask is defined by observed expression values and assigned core intervals, moreover, it is used to report the samples that gene i can assign to their correct target classes, the non overlapping samples set $V_i^{'}$. A gene mask element is indicated as

$$m_{ij} = \begin{cases} 1 & j \varepsilon V_i^{'}, i = 1, .., P \\ 0 & otherwise, j = 1, .., N \end{cases} \tag{3.43}$$

where $V_i^{'}$ is a non overlapping sample set or the set that doesn't fall within the overlapping region. When a sample is a member of the non overlapping set, it is defined as i. Otherwise, it is set to zero. The constructed core expression intervals $I_{i,1}$ and $I_{i,2}$ for a certain gene I are shown in Figure 3.7, together with the gene mask for that gene. Circles are used to indicate the observations that do not overlap. According to the observations organized by increasing expression values, the gene mask is sorted accordingly.

Next, the proportional overlapping score (POS) or overlapping measure is proposed to estimate the overlapping degree between different expression intervals. The POS_i is calculated as follows for each gene i

$$POS_i = 4 \frac{\left\langle I_i^{(v)} \right\rangle}{\left\langle I_i \right\rangle} \frac{v_i}{l_i} \left(\prod_{c=1}^{2} \theta_c \right) \tag{3.44}$$

where v_i is number of overlapping samples, l_i denotes total number of samples, and θ_c denotes

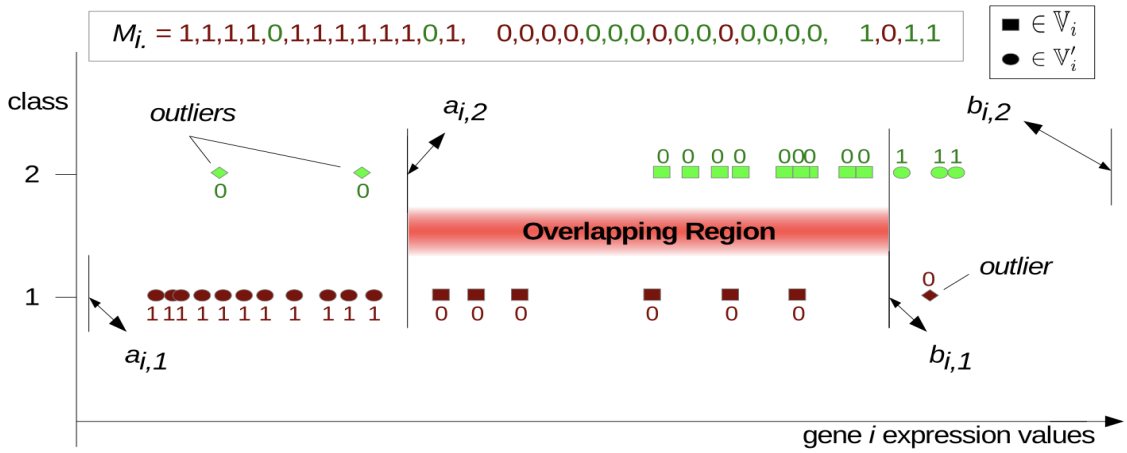

Figure 3.7: Core intervals with gene mask. An example for core expression intervals of a gene with 18 samples belongs to class 1, while a gene with 14 samples relates to class 2. The squares and circles denote the highlighted portions of the overlapping samples set and the non-overlapping samples set, respectively [123].

the proportion of class c samples among overlapping samples. Hence, θ_c can be defined as:

$$\theta_c = \frac{|V_{i,c}|}{V_i} \tag{3.45}$$

where $V_{i,c}$ defines a set of overlapping samples belonging to class c. According to equation 3.44 and 3.45, the value of *POS* measure for gene i shown in Figure 3.7 is

$$4 * \frac{15}{29} * (\frac{6}{15} * \frac{9}{15}) * \frac{\left\langle I_i^{(v)} \right\rangle}{\left\langle I_i \right\rangle} = \frac{72}{145} \frac{\left\langle I_i^{(v)} \right\rangle}{\left\langle I_i \right\rangle} \tag{3.46}$$

There are two scenarios that lead to an increase in the POS value. One occurs when the number of overlapping samples increases, and the other happens when θ_1 and θ_2 are close to each other. Overall, a lower *POS* measure represents genes with higher discriminative power. After defining gene masks and POS measures, the relative dominant class (RDC) to each gene to perceive which class they belong to. The RDC for gene i is defined as follows:

$$RDC_i = argmax_c \left(\frac{(\Sigma_{j \varepsilon U_c} I(m_{ij} = 1)}{|U_c|} \right) \tag{3.47}$$

where U_c is the set of class c samples. In this definition, the samples that belong to the non-overlapping samples set (V_i) into their target classes are only considered for each class. The class

with the highest proportion is the relative dominant class of the gene. After having gene masks, POS measures, and RDC assignments, the gene masks and POS measures are employed to select the minimum subset of genes that correctly classify the maximum number of samples [123].

The process of selecting the minimum subset of genes is proposed. Selecting the minimum subset of genes involves POS scores and gene masks. In order to mitigate the impact of expression outliers, this subset is intended to be the minimum one that properly classifies the greatest number of samples in a given training set. Such a process enables the removal of unnecessary data, such as genes with comparable expression characteristics. In order to avoid the impact of outliers, this subset is intended to be the minimum one that properly classifies the greatest number of samples in a given training set. Moreover, a greedy approach is performed to obtain a minimum gene subset. This stage determines the smallest gene collection to offer the best classification for a specific training set. Also, genes are arranged in descending order by the quantity of 1 bits in the minimum set.

The gene ranking is assigned by considering both POS scores and RDC. Due to binary classes, all genes have not been selected within each relative dominant class c where c can be 1 or 2. For whose RDC are equivalent to c, these are sorted by a rising order of POS values. The two disjoint of ranked genes are produced one for each class and the topmost gene is selected from each group to consist the gene ranking list. To have the final gene selection, the minimum subset of genes and the gene ranking process are considered. The smallest subset of genes, regardless of their POS values, are included in the final set because they enable the classifier under consideration to correctly categorise the greatest number of training examples.

3.4 Summary

This section introduces statistical learning which includes details of features selection techniques, classifiers, and model evaluation.

Models based on supervised machine learning are generated using sets of irrelevant features in order to create predictions based on classification. The supervised learning methods in classification are discussed in this chapter, particularly Random Forest (RF), k Nearest Neighbour (kNN), Logistic Regression (LR), Support Vector Machine (SVM), and eXtreme Gradient

Boosting (XGBoost).

RF constructs several decision trees using various random selections of the data and class. Every tree offers a classification prediction for the data. It compiles the votes from multiple decision trees to provide final decision.

kNN maintains the entire training dataset as a reference during the training phase. To make the prediction, the distance between the input data point and all the training examples is calculated. The K nearest neighbours are then determined to the input data point across their distances. A voting technique is used to determine the predicted label for the input data point by considering the class label of a new data point among its nearest "k" neighbours. The predicted label is derived from the majority class label among the K neighbours.

LR classifies binary outcomes based on multiple categorical or continuous independent variables. This technique estimates probabilities to makes these predictions providing ranges between 0 and 1. 0 is assigned to the event not occurring while 1 is defined to the event occurring. Although it is a rather quick method, it often performs poorly when the decision boundary is nonlinear.

SVM takes these data points and outputs to search the best line to classify your data points which is called the hyperplane. This line separates the data among classes. This implies that each data point on one side of the line will represent a category, and each data point on the other side will be assigned to a different category.

XGBoost combines the predictions of multiple individual models to produce a strong prediction. The individual models in XGBoost are decision trees, which are trained using gradient boosting. This means that at each iteration, the algorithm fits a decision tree to the residuals of the previous iteration. A weighted average is used to make predictions by combining the predictions of all the trees and each tree's weights are determined by applying the same objective function during the training phase. The sum of the predictions of all the trees provides the final prediction.

Examining performances is crucial after conducting classifiers using the collection of informative features. In order to comprehend the model's performance as well as its strengths and weaknesses, this chapter also describes model evaluation. One of the most straightforward

ways for evaluating a classifier's performance using test data sets is the confusion matrix. The confusion matrix shows how accurate a classification model is by dividing the total number of input samples by the number of correct predictions. The optimal model is indicated when the accuracy of classification is near to 1.

Feature selection techniques play important roles in especially gene expression data. The main concept of feature selection methods is to eliminate irrelevant features by using several statistical approaches. For instance, Wilcoxon Rank Sum Test (WRS), Kruskal Wallis Test (KW), Least Absolute Shrinkage Selector Operator (LASSO), Minimum Redundancy and Maximum Relevance (mRMR), and Proportional Overlapping Score (POS) are covered in this chapter.

WRS is used to compare two independent samples from non-parametric alternatives. As in all rank tests, the data in both groups is compared by arranging and listing in order of increasing value. The next step is to assign a rank value to each number in the two categories. A rank is assigned to each number which the smallest number in either group begin with a rank of 1.0. If duplicate numbers are called ties. This procedure repeats until every number is ranked. Finally, the "sum of ranks" is obtained by adding the values from each group's rankings column.

KW is utilised to compare the means of at least three or more independent groups in non-parametric statistical techniques. This test evaluates data ranks to arrange all the sample data from low to high and the ranks for all groups are averaged. If the p-values of the KW test are statistically significant, it means that the average group ranks are not all equal. As a result, the analysis shows if there are any values that rank differently among the groups.

LASSO is a regularization technique that performs both variable selection and regularization. This method shrinks the coefficients of non-informative features to zero, effectively selecting only the most informative features in the model. LASSO is useful in situations where there are many variables that may be contributing to a particular outcome which can help to simplify the model and improve its accuracy.

mRMR ranks features based on their importance in predicting the target variable, where importance has a redundancy and relevance component. Redundancy measures the correlation between features and a lower redundancy represents a better feature selection criterion. In contrast, relevance measures correlation between the feature and target class. When larger

relevance indicates the feature has a greater amount of information to classify the target class. Therefore, the mRMR tends to select features with a high correlation with the class and a low correlation between themselves.

POS utilizes overlapping analysis to propose core intervals, gene masks, and POS scores. The ability to accurately classify target classes is demonstrated by the smaller POS scores. In addition, POS scores and gene masks provide a minimum subset of genes, while POS scores and relative dominant classes are employed to generate top-ranked genes. The final subset of informative genes is considered based on the minimum subset of genes and the top-ranked genes. However, this approach can deal with only binary class problems.

In the following chapter, we propose a new feature selection technique that extends the POS method. A key aspect of this approach is determining class intervals and the overlap between classes, which are essential for calculating overlapping scores. Additionally, we provide illustrative examples to help users gain a better understanding of our algorithm. Experiments are conducted to evaluate the performance of our proposed method and compare its predictive effectiveness with other commonly used feature selection techniques using machine learning models.

CHAPTER 4

Datasets

This chapter includes a description of the datasets that are used in this thesis. It details data preprocessing and a summary of the used gene expression datasets.

4.1 Data Preprocessing of Microarray Data

Affymetrix CEL files are used for gene expression analysis. Separate CEL files store the intensity values of 25 base probes on each array, and the intensities of multiple probes are used to derive the expression of individual genes. In our study, we exploited the robust multiarray average (RMA) to turn intensity values into expression measures [91]. The RMA algorithm has three steps: background correction, normalization, and summary expression value computation. Affy, an R/Bioconductor package designed for Affymetrix oligonucleotide arrays, was used to implement the RMA procedure [90]. The crucial note is that the few gene expression datasets are provided in .tab and .csv file formats, for which data preprocessing is not guaranteed. These datasets include MLL, Leukaemia, Carcinoma, Lung(1), Srbct, Brain Tumour, and Lung(2).

4.2 Gene Expression Datasets

Several gene expression datasets were employed in this thesis. These datasets are organised into three groups, each aligned with a specific research objective. The first group, a summary of the

first seven gene expression datasets is described in Section 4.2.1, where these datasets are used to evaluate the 3cPOS method in Chapter 5. In Section 4.2.2, additional gene expression datasets are included to assess the minimum subset of genes in Chapter 6. The remaining twenty-four gene expression datasets are summarised in Section 4.2.3 and are used to evaluate the mPOS method in Chapter 7.

4.2.1 First Group of Datasets — Evaluation of the 3cPOS Method

For a benchmarking experiment, seven different gene expression datasets are taken as three-class problems. Table 4.1 describes the summary of used gene expression datasets, which provides names of data and diseases, genes, samples, class distribution, and sources. Each of the datasets are high-dimensional, with large numbers of genes, small samples and datasets are imbalanced (classification data with skewed class proportions). The data sets implemented in our experiment can be accessed through public resources.

The GSE23938 dataset [198] pertains to breast cancer and includes data from three distinct tumor models: hybrid strain (129B6/FVB), common inbred mouse strain (Balb/c), and inbred mouse strain (FVB). Each strain has a unique genetic background that influences immune responses, cancer susceptibility, and the progression of breast cancer. The GSE22093 dataset [92] is a breast cancer dataset that includes absence, complete pathological response (pCR), and residual disease (RD). The GSE102287 dataset [130] comprises information from three stages of lung cancer: stage 1, stage 2, and stage 3. The GSE17951 dataset, provided by [96], includes different sample types of prostate cancer such as biopsy, control, and tumor. The GSE102079 dataset [38] includes three types of carcinoma tissue: non-tumorous tissue, tumorous tissue, and normal liver. Data on leukaemia samples from bone marrow, lymph nodes, and peripheral blood are available in the GSE21029 dataset [81]. The MLL dataset [8] encompasses three types of leukemia: acute lymphocytic leukemia (ALL), acute myeloid leukemia (AML), and mixed-lineage leukemia (MLL).

Table 4.1: Summary of characteristics across gene expression datasets

Datasets	Diseases	Genes	Samples	Class distribution	Sources
GSE23938	Breast Cancer	18586	41	5/7/29	[198]
GSE22093	Breast Cancer	22283	103	6/28/69	[92]
GSE102287	Lung Cancer	54675	66	36/19/11	[130]
GSE17951	Prostate Cancer	54675	154	32/13/109	[96]
GSE102079	Carcinoma	54613	257	91/152/14	[38]
GSE21029	Leukemia	54675	62	19/17/26	[81]
MLL	Leukemia	12533	72	24/28/20	[106]

4.2.2 Second Group of Datasets — Evaluation of the Minimum Gene Subset

For the benchmarking experiments, we made use of the seven gene expression datasets described in the previous section (Section 4.2.1) to evaluate the minimum subset of genes. In addition, the characteristics of an additional seven gene expression datasets are included in Table 4.2 to provide a more comprehensive evaluation. Several publicly accessible datasets, each with a different number of classes according to cancer type and stage, are included in this analysis.

These datasets are classified as three-class classification problems: GSE13911 [43], GSE2990 [166], and GSE26712 [18, 180]. In contrast, the GSE40595(1) [190] and GSE27854(1) [103] were originally classified as four-class classification problems. To evaluate the minimum subset of genes, GSE40595(1) combines patients with ovarian cancer in stages III and IV into a single class. Similarly, GSE27854(1) have classified colorectal cancer patients into three distinct groups: one class encompasses patients in stages I and II, while two additional classes represent patients in stages III and IV. GSE162228(1) [34] was originally classified as five-class classification problems. Patients in stages I and II are included in a single class and contrasted against patients in stages III and IV to assess the purposes of this study. The GSE30219 [158] dataset represents a six-class problem, categorising patients with lung cancer into three distinct classes. Specifically, patients with stages T0, T1, and T4 are combined into a single class, while those in stages T2 and T3 are grouped separately, and patients with the main cancer (primary), TX, are treated as a distinct category. All datasets used in this study are publicly available.

Table 4.2: Summary of characteristics across gene expression datasets

Datasets	Diseases	Genes	Samples	Class distribution
GSE23938	Breast Cancer	18586	41	5/7/29
GSE22093	Breast Cancer	22283	103	6/28/69
GSE102287	Lung Cancer	54675	66	36/19/11
GSE17951	Prostate Cancer	54675	154	32/13/109
GSE102079	Carcinoma	54613	257	91/152/14
GSE21029	Leukemia	54675	62	19/17/26
MLL	Leukemia	12533	72	24/28/20
GSE13911	Gastric Tumors	54675	69	30/19/20
GSE2990	Breast Cancer	22283	189	33/31/125
GSE26712	Ovarian Cancer	22283	185	24/129/32
GSE40595(1)	Ovarian Cancer	54675	77	8/31/38
GSE27854(1)	Colorectal Cancer	54675	115	23/17/75
GSE162228(1)	Breast Cancer	54675	133	9/82/42
GSE30219	Lung Cancer	54675	307	14/241/52

4.2.3 Third Group of Datasets — Evaluation of the mPOS Method

Twenty-four different gene expression datasets with two, three, four, and fives classes are utilised to conduct our benchmarking experiment to facilitate the evaluation of the mPOS method. Each dataset, including the names of the data, diseases, number of genes, number of samples, and class distribution, is shown in Table 4.3. The majority of the datasets demonstrate high dimensionality, with large numbers of genes measured with relatively small sample sizes, and class distributions are mostly imbalanced, except for the Carcinoma dataset. All datasets are accessible through publicly available resources.

For binary class problems, nine datasets are included. The GSE6861 [17] is a breast cancer dataset, including two groups: no pathological complete response (npCR) and pathological complete response (pCR). The GSE10780 dataset [30] is a breast cancer dataset that includes Invasive ductal carcinoma (IDC) and unremarkable breast ducts (Normal). The GSE19615 dataset [113] pertains to breast cancer and consists of tumor recurrence, while the GSE22513 dataset [13] includes tumor recurrence and no tumor recurrence. For colorectal cancer, two gene expression datasets are exploited: GSE24514 [2] and GSE4045 [107]. The GSE24514 dataset is classified into normal and colorectal cancer samples, while the GSE4045 dataset focuses on conventional serrated colorectal carcinomas (CRCs) and serrated colorectal carcinomas (CRCs)

with serrated morphology. The Leukaemia dataset [71] includes acute lymphoblast leukemia sample (ALL) and acute myeloid leukemia sample (AML). Moreover, the Carcinoma dataset [141] pertains to Carcinoma including normal and abnormal tissue samples, and the Lung(1) dataset [10] focuses on Atypical carcinoid (AC) and malignant pleural mesothelioma.

For three class problems, six gene expression datasets are employed to facilitate the evaluation of mPOS method. The descriptions of GSE21029, GSE22093, GSE23938, GSE102079 and MLL are described in Section 4.2.1. In addition, the GSE21510 dataset [176] is colorectal cancer including three tissue types: normal homogenization, homogenized colorectal cancer, and liver colorectal metastases.

For four class problems, six gene expression datasets are considered. The GSE15852 dataset [140] includes grades of breast cancer; grade 1, grade 2, grade 3, and normal. The GSE27854(2) [103] includes four stages of colorectal cancer: stage 1, stage 2, stage 3, and stage 4. The GSE27651 [104] is an ovarian cancer dataset, including high-grade ovarian serous carcinoma, low-grade ovarian serous carcinoma, low-malignant tumors of the ovary, and normal ovarian surface epithelials cells. The GSE38666 [115, 87] is an ovarian cancer dataset, classified into four different tissues: ovarian cancer epithelium, ovarian cancer stroma, ovarian normal stroma, and ovarian surface epithelium. While the GSE40595(2) [190] is an ovarian cancer dataset that consists of four different tissues: microdissected normal ovarian stroma, microdissected ovarian cancer stroma, microdissected ovarian surface epthelium, and microdissected ovarian tumor epthelial component. The Srbct dataset [102] represents small-blue-round-cell tumour including Ewing's sarcoma (EWS), burkitt's lymphoma (BL), neuroblastoma (NB), and rhabdomyosarcoma (RMS).

For five-class problems, three gene expression datasets are included in our experiment: GSE162228(2) [34], Brain Tumour [150], and Lung(2) [16]. The GSE162228(2) dataset consists of five stages of breast cancer; stage 0, stage 1, stage 2, stage 3, and stage 4. The Brain Tumour dataset comprises five diagnostic classes: medulloblastoma, malignant glioma, rhabdoid tumor, normal cerebellum, and primitive neuroectodermal tumor (PNET). The Lung(2) dataset includes lung diagnostic classes: adenocarcinoma (AD), normal lung (NL), small cell lung cancer (SMCL), squamous cell carcinoma (SQ), and pulmonary carcinoid (COID).

Table 4.3: Summary of characteristics across gene expression datasets

Datasets	Diseases	Samples	Genes	Class distributions
GSE6861	Breast Cancer	161	61359	95/66
GSE10780	Breast Cancer	185	54675	42/143
GSE19615	Breast Cancer	115	54675	100/15
GSE22513	Breast Cancer	28	54675	20/8
GSE24514	Colorectal Cancer	49	22283	34/15
GSE4045	Colorectal Cancer	37	22215	29/8
Leukaemia	Blood Cancer	72	7130	49/23
Carcinoma	Carcinoma	36	7457	18/18
Lung(1)	Lung Cancer	181	12533	150/31
GSE21029	Leukemia	62	54675	19/17/26
GSE22093	Breast Cancer	103	22283	6/28/69
GSE23938	Breast Cancer	41	18586	5/7/29
GSE102079	Carcinoma	257	54613	91/152/14
GSE21510	Colorectal Cancer	148	54675	19/104/25
MLL	Leukemia	72	12533	24/28/20
GSE15852	Breast Cancer	86	22283	8/23/12/43
GSE27854(2)	Colorectal Cancer	115	54675	16/41/35/23
GSE27651	Ovarian Cancer	49	54675	22/13/8/6
GSE38666	Ovarian Cancer	45	54675	18/7/8/12
GSE40595(2)	Ovarian Cancer	77	54675	8/31/6/32
Srbct	Small-blue-round-cell Tumour	83	2308	11/29/18/25
GSE162228(2)	Breast Cancer	133	54675	9/29/53/36/6
Brain Tumour	Brain Tumour	40	7129	10/8/4/6/10
Lung(2)	Lung Cancer	203	12600	139/20/17/6/21

4.3 Summary

This chapter mainly discusses data preprocessing and the description of the used gene expression datasets. RMA is employed for data preprocessing to turn intensity values into expression measures via three steps: background correction, normalization, as well as summary expression value computation. Several gene expression datasets are exploited in this thesis; we provide an overview and description of the used datasets. The first group of data aims to assess the evaluation of the 3cPOS method, while the second and third groups are used to support the evaluation of the minimum subset of genes and the mPOS method, respectively. The datasets varied in dimensionality, sample size, disease type, and class distribution across all categories, according to binary, three-class, four-class, and five-class classification problems involving different types of cancers. High dimensionality and class imbalance are characteristics of the majority of datasets, which reflects the typical characteristics of microarray data. An overview of the datasets is presented in each section, including the number of genes, sample sizes, class distributions, and disease classifications. The following chapters make use of these datasets to provide a comprehensive evaluation of methods across the study.

CHAPTER 5

3-class Proportional Overlapping Score Method

5.1 Introduction

Health and biomedical research have become essential area of science for finding ways to prevent illness and death in both people and animals. Biotechnology techniques, along with experimental and statistical methods, are being developed to improve treatments and cures [41]. Gene expression data is widely used to gain genetic information to carry out a wide range of biological functions and insights. The main challenge of using gene expression data is high dimensionality with low sample size [55]. Selecting a subset of informative genes is crucial, as it helps improve predictive performance and eliminates irrelevant features [47].

In this chapter, we propose a novel feature selection method based on overlapping analysis. This method helps reduce dimensionality and improves model performance and interpretability by selecting informative features. Feature selection methods play an important role in handling high-dimensional and noisy data. For example, the Wilcoxon Rank Sum test is implemented to select relevant features by ranking the informative features based on their p-value. The smallest p-values represent the higher discriminative power to classify the correct target class [46]. Whilst minimum redundancy and maximum relevance is utilised to eliminate the redundancy and irrelevant features by finding the minimum values among all features and selecting the maximum values between feature and target classes [56]. [93] has proposed the least absolute shrinkage

operator selector. This method produces the informative feature by shrinking the coefficient of non-informative features to zero while retaining features that are relevant in the model. [6] has proposed Maskedpainter (MP) to select the informative features based on the overlapping analysis. The core expression intervals, gene masks, dominant classes, and overlapping scores are assigned for each gene, moreover, the minimum subset of gene and gene ranking are provided to select the informative features in the final subset of features. Another important feature selection method is Proportional Overlapping Scores (POS) proposed by [123]. [123] developed a novel generalized version of the overlapping score (OS) measure, proposed in [6]. POS employs the interquartile range to detect outliers. POS scores and gene masks are utilized to find the maximum samples that can correctly classify the correct target classes in the minimum subset of genes, while POS scores and relative dominant classes provide the discriminative power in the top ranked genes. Moreover, the smallest POS scores represent the higher discriminative power to distinguish the correct target classes. The results from [123] demonstrated that POS performed better than Wil-RS, mRMR, LASSO, and MP via classification error rate but the restriction of using POS is that this method can only be implemented for binary class problems. To deal with these limitations, it is necessary to extend the version of POS to handle multiple class problems. In our study, an extended version of POS is proposed for implementation in three-class classification problems.

Microarray data provides the form of gene expression matrix, $X = [x_{ij}]$ such x_{ij} represents the observed expression value of gene i for sample j where $i = 1, ..., p$ and $j = 1, ..., n$ as Figure 5.1. Each sample is also identified by the target class label; y_i and it denoted as the phenotype of the tissue sample. Moreover, each element y_i has a single value c which is 1, 2, or 3.

Analyzing the overlap between the distribution of gene expressions among different classes can provide valuable information on the classification capabilities of genes. The main idea is that a gene i is likely to play a key role in unambiguously classifying tissue samples from a class c to their correct class when its distribution of expressions for this class is not overlapping with its distributions of expressions for other classes. In other words, when gene i expressions lie within the interval of a single class, gene i is capable of classifying the correct samples. For instance, Figure 5.2 (a) ,(b), (c), and (d) demonstrate possible scenarios that could be related

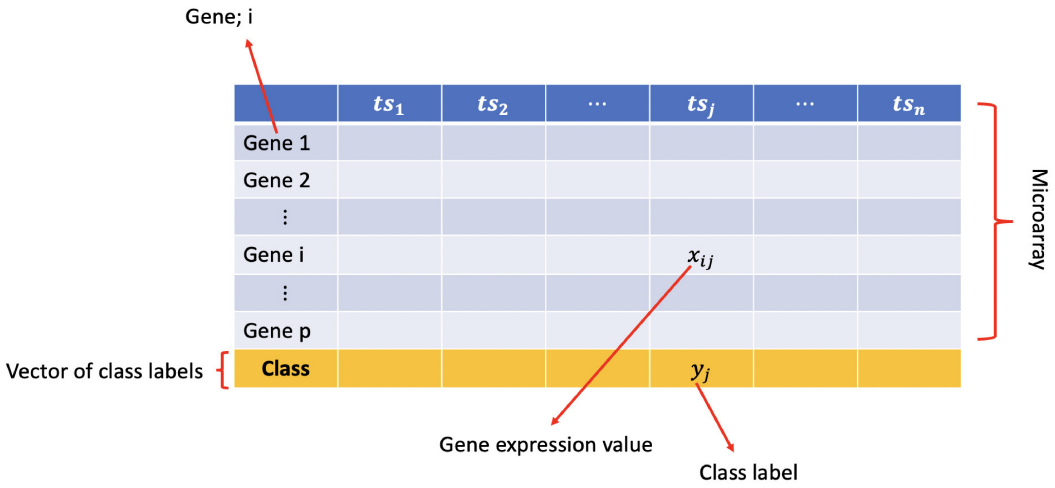

Figure 5.1: The structure of microarray data

to overlapping intervals based on three class problems. In this example, there are 60 samples belonging to three different classes and each class has 20 samples. Figure 5.2 (a) demonstrates gene i_1 expression is therefore relevant for the considered classification target outcome and it can serve as an informative feature in distinguishing the target classes of interest in this problem. This theoretical paradigm offers an extreme level of separation which is unlikely to occur in many complicated real-world classification problems. However, this highlights the key benefit of analysing the overlapping degrees of gene expressions in multi-class problems, *e.g.* for providing informative gene ranking. Unlike gene i_1, several expression values of genes i_2 and i_3 fall into overlapping regions of different classes, making them less capable in distinguishing between these classes, seen in Figures 5.2 (b) and 5.2 (c). Additionally, gene i_4 expressions highly overlap among regions of the three classes, making it even less capable to differentiate between them, see Figure 5.2 (d).

This chapter proposed the novel feature selection technique, called 3-Class Proportional Overlapping Scores (3cPOS) method for three class problems. Our proposed method, 3cPOS, addresses the limitation of the POS, which can only be applied to binary classification problems. As also highlighted in [123], this is then used to derive a score, referred to as 3cPOS, for each gene considering three factors: (1) length of the overlapping regions; (2) number of overlapped samples; and (3) ratio of the contribution of classes to the overlapped samples.

Further definitions of the 3cPOS method are described below.

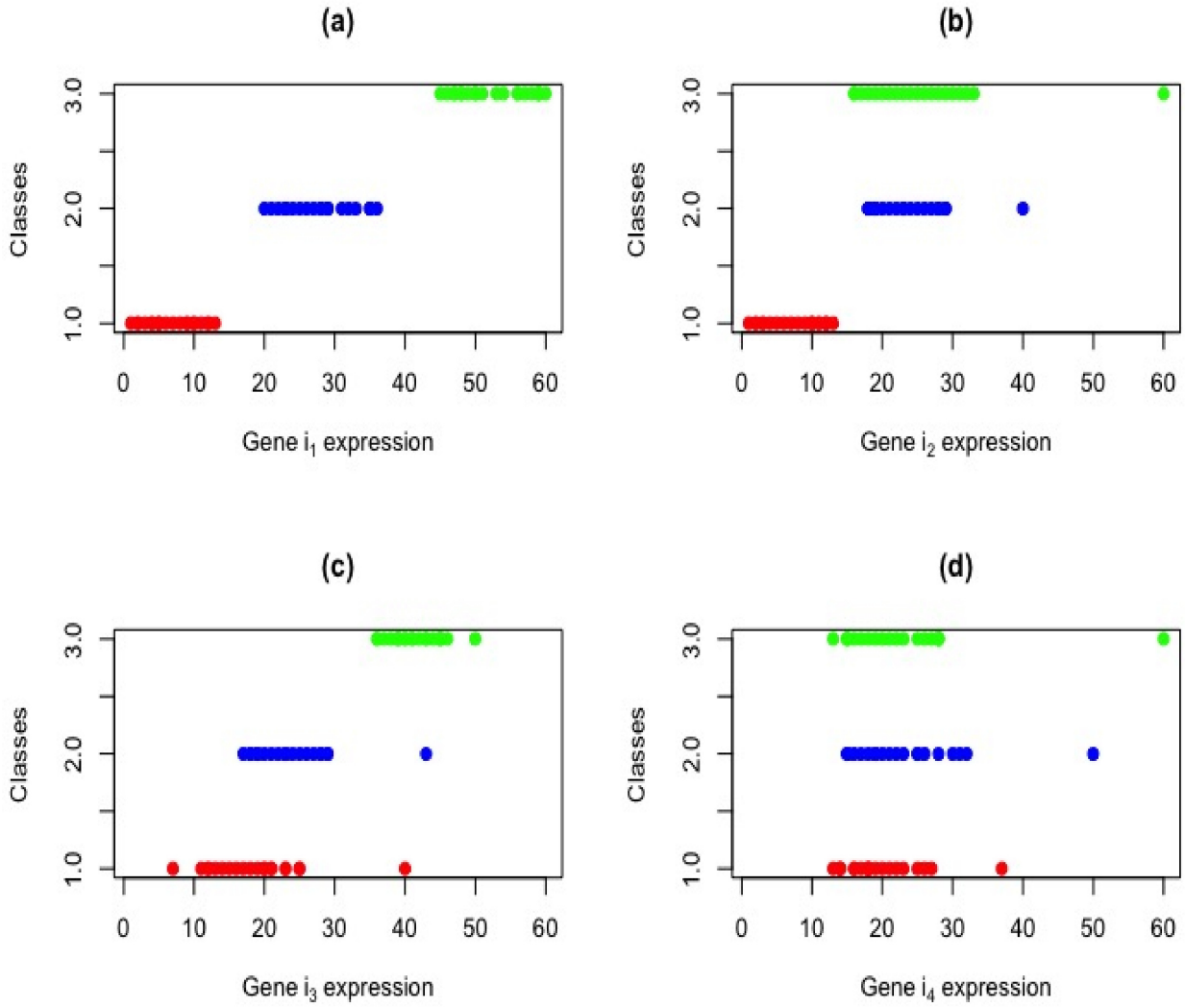

Figure 5.2: An example for four different genes with different overlapping patterns. Expression values of four different genes (i_1, i_2, i_3, and i_4) each of which with 60 observations belonging to 3 classes, 20 observations for each class: (a) expression values of gene i_1, (b) expression values of gene i_2, (c) expression values of gene i_3, and (d) expression values of gene i_4

5.2 Core Intervals

Due to noisy values and dimensionality in microarray data, it might lead to the influence of outlier. To deal with these problems, the gene expression values firstly are required to transform or rescale into a standard range. This helps mitigate the impact of potential gene's heterogeneity on classification [3]. The z score standardisation [127] is considered to transform the values across gene expression data. The z-score standardisation is proposed to normalize the values based on the mean and standard deviation, resulting each gene being center at 0 with standard deviation of

1 [186]. To determine the core expression intervals, several techniques can be exploited. The interquartile range (IQR) is used to construct the core interval to detect outliers [123], while some studies used the Median Absolute Deviation (MAD) to generate the expression interval to deal with the effect of outliers [74]. For our study, 1.96 standard deviations of gene expression around the mean of each class are considered to define the core interval of the class. This helps mitigate the effects of outliers. Our choice of using 1.96 standard deviations corresponds to a widely accepted standard in research for balancing precision and interpretability. Our procedure can be described as follows:

Definition 5.2.1. Core expression interval is expressed by $I_{i,c}$. For each class c of gene i, it can be defined for two expression intervals; the minimum and maximum boundary. The core expression interval can be expressed in the form:

$$I_{i,c} = \bar{z}_{i,c} \pm 1.96 s_{i,c} \tag{5.1}$$

where $\bar{z}_{i,c}$ is the mean of standardised gene i expressions belonging to class c and $s_{i,c}$ represents the standard deviation of standardised gene i expressions that belonging to class c.

Definition 5.2.2. Number of non-outlier samples, n_i. For each gene i, it is the set of samples whose standardised expression values lie within their own class core interval. It is defined by;

$$n_i = \{j : z_{ij}\ \varepsilon\ I_{i,c}, j = 1,2,3,...n\}, \tag{5.2}$$

Definition 5.2.3. Length of total core interval, l_i. It is length of the region between the global minimum and global maximum boundaries of the core expression intervals across the three classes.

$$l_i = [a_i, b_i], \tag{5.3}$$

where $a_i = min(a_{i,c})$ such that $a_{i,c} = \bar{z}_{i,c} - 1.96 s_{i,c}$ and $b_i = max(b_{i,c})$ such that $b_{i,c} = \bar{z}_{i,c} + 1.96 s_{i,c}$

5.3 Overlapping between Intervals

Based on [6] and [123], an crucial component of a gene's characteristic can be given to a classifier by assessing the overlap between the expression intervals of the gene for various classes. For three-class problems, the overlap between intervals are defined as follows:

Definition 5.3.1. Interval of two-way overlapping region, $l^{(2)}_{i(c_1c_2)}$, which represents the intersection region between the core expressions of any two classes. it can be expressed as;

$$l^{(2)}_{i(c_1c_2)} = I_{i,c_1} \cap I_{i,c_2} \tag{5.4}$$

where c_1 and c_2 represent class labels, with $c_1 < c_2$, *i.e.* $l^{(2)}_{i(c_1c_2)}$, $l^{(2)}_{i(c_1c_3)}$, and $l^{(2)}_{i(c_2c_3)}$. $\bar{l}^{(2)}_i$ represents the average length of two-way overlapping regions across the three possible pairs. it can be defined as;

$$\bar{l}^{(2)}_i = \frac{l^{(2)}_{i(c_1c_2)} + l^{(2)}_{i(c_1c_3)} + l^{(2)}_{i(c_2c_3)}}{3} \tag{5.5}$$

Definition 5.3.2. Interval of three-way overlapping region, $l^{(3)}_i$. It demonstrates the intersection region among the core expression intervals of the three classes. It is given as;

$$l^{(3)}_i = I_{i,1} \cap I_{i,2} \cap I_{i,3} \tag{5.6}$$

Definition 5.3.3. Number of two-way overlapping sample, $n^{(2)}_i$, which consists of the samples that fall within intervals of a two-way overlapping region, $l^{(2)}_{i(c_1c_2)}$. The number of two-way overlapping samples is given as;

$$n^{(2)}_i = \{j : j \; \varepsilon \; n_i \; \wedge \; z_{ij} \; \varepsilon \; l^{(2)}_{i(c_1c_2)}\} \tag{5.7}$$

Definition 5.3.4. Number of three-way overlapping sample, $n^{(3)}_i$. It contains the samples that fall inside the interval of the three-way overlapping region; $l^{(3)}_i$. The number of three-way

overlapping samples is expressed as;

$$n_i^{(3)} = \{j : j \, \varepsilon \, n_i \wedge z_{ij} \, \varepsilon \, l_i^{(3)}\} \tag{5.8}$$

5.4 The 3cPOS Measures

Our novel method has extended the POS measure proposed by [123]. Our method generates a novel generalised version, called the 3cPOS score, and it takes the two and three-ways overlapping of a certain gene i into consideration to derive an overlapping score for three class problems. Our measure is an aggregation of two terms; a score for the two-way overlaps; a score for the three-way overlap. Three possible pairs of two-way overlapping could be considered: class $1-2$, class $1-3$, and class $2-3$. The two-way overlapping score can then be defined as δ_i:

$$\delta_i = \frac{\bar{l}_i^{(2)}}{l_i} \frac{n_i^{(2)}}{n_i} \sum_{c_1,c_2=1,c_1<c_2}^{3} \frac{\theta_{i,c_1}\theta_{i,c_2}}{3} \tag{5.9}$$

where $\bar{l}_i^2$ is the average length of the two-way overlapping regions across the three possible pairs, $n_i^{(2)}$ is expressed as the total number of the two-way overlapping samples across the three possible pairs of two-way overlaps, and $\theta_{i,c_1}\theta_{i,c_2}$ represents the proportions of two-way overlapping samples belonging to class c_1 and c_2, respectively. Their multiplication can be expressed as follows:

$$\theta_{i,c_1}\theta_{i,c_2} = \frac{n_{i,c_1}}{n_{i,c_1c_2}} \frac{n_{i,c_2}}{n_{i,c_1c_2}} \tag{5.10}$$

where n_{i,c_1} and n_{i,c_2} are expressed as the number of two-way overlapping samples assigned to class c_1 and the number of two-way overlapping samples assigned to class c_2, respectively. n_{i,c_1c_2} is denoted as the total number of two-way overlapping samples belonging to class c_1 and c_2.

The three-way overlapping score, denoted by as γ_i, is defined as follows:

$$\gamma_i = \frac{l_i^{(3)}}{l_i} \frac{n_i^{(3)}}{n_i} \prod_{c=1}^{3} \beta_{i,c} \tag{5.11}$$

where $l_i^{(3)}$ is the length of the three-way overlapping region, $n_i^{(3)}$ is expressed as the number of three-way overlap samples, while $\beta_{i,c}$ represents the proportion of class c samples among three-way overlapping samples. It is given by:

$$\beta_{i,c} = \frac{n_{i,c}}{n_i^{(3)}} \tag{5.12}$$

where $n_{i,c}$ represents the number of three-way overlapping samples belonging to class c.

Therefore, the 3cPOS score can be derived using a weighted aggregation of the two-way and three-way overlapping scores to estimate degrees of overlap between classes as follows:

$$3cPOS_i = 2\delta_i + 3\gamma_i \tag{5.13}$$

For an individual gene i, a smaller 3cPOS score indicates higher discriminative capability.

The pseudo-code of the 3-class Proportional Overlapping Scores (3cPOS) algorithm is presented in Algorithm 1. Let $\mathbb{G}$ denote the set of all genes, where $\mid \mathbb{G} \mid = p$. Initially, the observed expression values, X, across the entire dataset are standardised (lines 3-5). The standardized expressions, along with their true target class labels, are then utilised to derive the core interval for class 1, 2, and 3, (*i.e.* $I_{i,1}$, $I_{i,2}$, $I_{i,3}$) (lines 7-9). For each gene i, the number of non-outliers (lines 10) and the total core expression interval (line 11) are computed. To analyse the overlap between intervals, the average length of the two-way overlapping regions is calculated (lines 13) as well as the interval of the three-way overlapping regions (line 14). Similarly, the number of two-way overlapping samples (line 15) and the number of three-way overlapping samples (line 16) are computed. Consequently, the 3cPOS score for each gene i is calculated (lines 18) before creating a sequence of genes, $\mathbb{G}^*$, (lines 20) that ranked in an ascending order based on the 3cPOS. Finally, the top r genes in $\mathbb{G}^*$ is selected for the corresponding classification task (line 21).

5.5 Illustrated Examples

This section aims to provide a detailed discussion of how to compute 3cPOS scores using Gene 1 and Gene 4 Expression. As indicated in Section 5.1, there are 20 samples in each class of

Algorithm 1 3cPOS Method For Gene Selection

Input: The observed expression values of all genes (X), target class labels (Y) and number of genes to be selected (r).

Output: Sequence of the selected genes $(\mathbb{T})$.

for all $i \in \mathbb{G}$ **do**
  Data Standardisation
  for $j = 1$ to N **do**
    Transform x_{ij} into their z-score standardisation using $z_{ij} = (x_{ij} - \bar{x}_i)/s_i$
  end for
  Getting Class Intervals
  for $c = 1$ to 3 **do**
    Calculate $I_{i,c}$ as defined in equation (5.1), representing the core expression interval for each class c of gene i.
  end for
  Compute the number of non-outlier sample, n_i, as defined in equation (5.2).
  Compute the total core interval, l_i, as defined in equation (5.3).
  Getting Overlapping between Intervals
  Compute the average length of two-way overlapping region, $\bar{l}_i^{(2)}$, as defined in equation (5.5).
  Compute interval of three-way overlapping region, $l_i^{(3)}$, as defined in equation (5.6).
  Compute the number of two-way overlapping sample, $n_i^{(2)}$, as defined in equation (5.7).
  Compute the number of three-way overlapping sample, $n_i^{(3)}$, as defined in equation (5.8).
  Getting 3cPOS scores
  Calculate $3cPOS_i$ as defined in equation (5.13) using $\bar{l}_i^{(2)}$, $l_i^{(3)}$, $n_i^{(2)}$, and $n_i^{(3)}$.
  Getting Sequence of Genes
  Create $\mathbb{G}^*$ which is an ordered list of features (genes) in $\mathbb{G}$, sorted by ascending order of $3cPOS$ values.
  Define $\mathbb{T}$ as first r genes in $\mathbb{G}^*$.
end for
return $\mathbb{T}$

genes. As a result, each gene has 60 samples, which is determined by three class problems. Figure 5.3 (a) displays the Gene 1 Expression scatter plot based on three classes. Equation 5.1 is utilised to represent the intervals of the core expression, and each class is assigned a straight line. Three samples are therefore outliers. The number of non-outliers sample is 57 according to Definition 5.2.2. Furthermore, the core expression intervals' global minimum and global maximum boundaries are 3.95 and 42.45, respectively, shown as black dashed lines. These lines also show the length of the entire core interval. The scatter plot of Gene 1 Expressions between classes 1 and 2 is shown in Figure 5.3 (b). The overlapping region between class 1 and class 2 core expression intervals is represented by the black dashed lines. Applying Definition 5.3.1 into account, the length of the two-way overlapping region is 21.95. Furthermore, by using Definition 5.3.3, the total number of two-way overlapping samples can be found by examining 19 samples from classes 1 and 17 samples from class 2 that fall within this region. Figure 5.3 (c) demonstrates a scatter plot of Gene 1 Expressions between classes 1 and 3. Black dashed lines are represented two-ways overlapping region between two classes. As a results, length of two-ways overlapping region is 21.95, and the total number of two-way overlapping samples is 38. Figure 5.3 (d) illustrates a scatter plot of Gene 1 Expressions between classes 2 and 3. Two-ways overlapping region between two classes are represented by the black dashed lines. Therefore, length of two-ways overlapping region and the total number of two-way overlapping samples are 31.73 and 38, respectively. According to Equations 5.9 and 5.11, we obtain 0.64 and 0.06, respectively and these values are combined to generate the single value of 3cPOS. Consequently, the 3cPOS score for Gene 1 Expression is determined using Equation 5.13, which yields a value of 0.70.

The scatter plot of Gene 4 Expressions across three classes is shown in Figure 5.4 (a). According to Definition , the intervals of the core expression are assigned to each class by a straight line. It observes that each sample falls within its core intervals, so there are no outliers in each class. Moreover, there are no overlapping regions based on three class problems. As a result, The number of non-outliers sample, length of total core interval, length of three-ways overlapping region, and number of three-ways overlapping samples are 60, 62.46, 0, and 0, respectively. In Figure 5.4 (b), (c), and (d), the Gene 4 Expressions scatter plot between two

classes is shown. These show that there are no areas in which classes 1-2, 1-3, and 2-3 overlap. Consequently, both the length of the two-way overlapping region and the number of two-way overlapping samples become zero over three pairs. Hence, by using equations 5.9, 5.11, and 5.13, it is evident that Gene 4 Expression's 3cPOS score is 0.

In following section, the performance of 3cPOS method is evaluated alongside other feature selection techniques using multiple classifiers across several gene expression datasets.

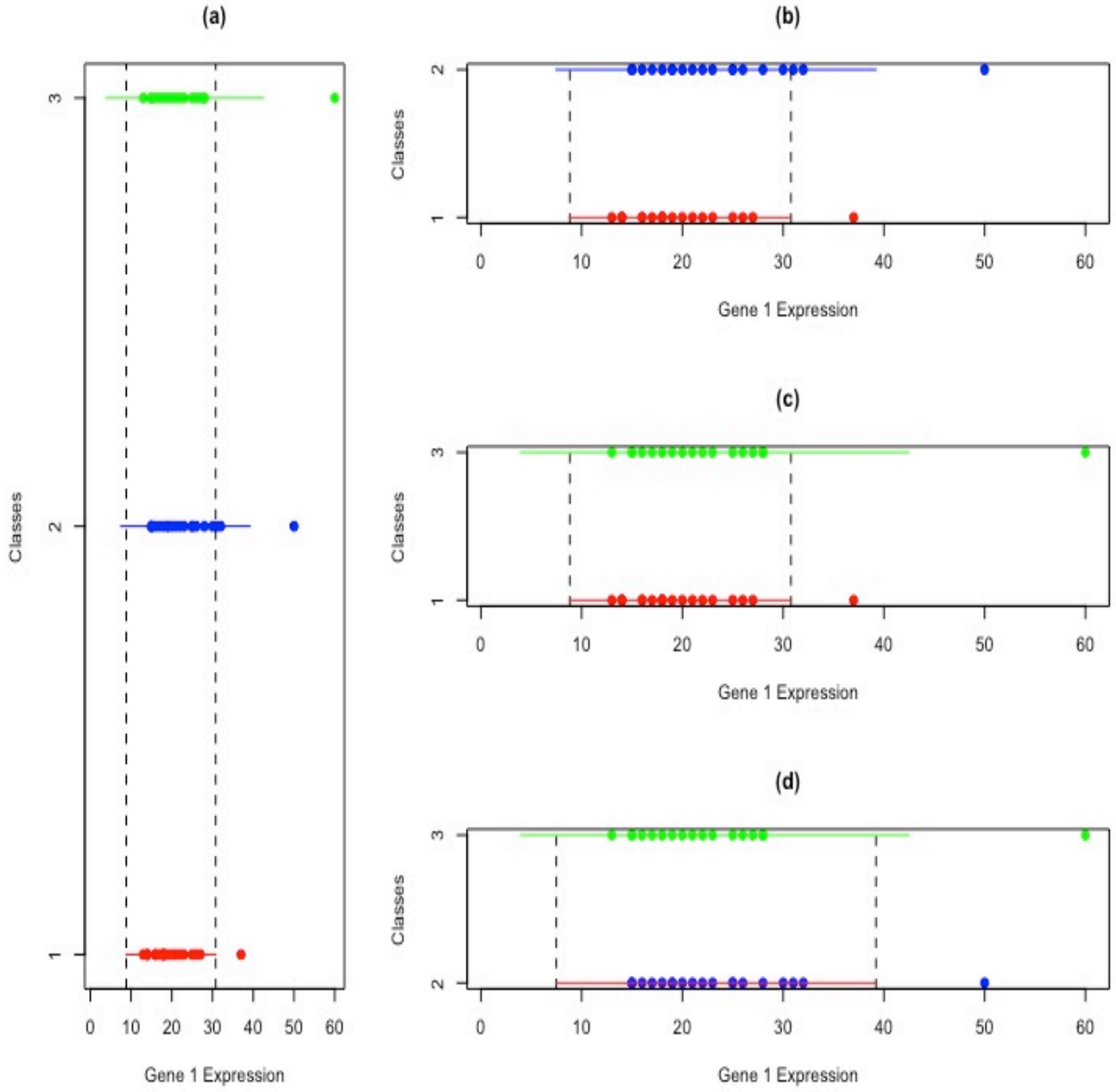

Figure 5.3: Scatter plot of Gene 1 Expression values across three classes. (a) Expression levels for all three classes. (b) Pairwise comparison between class 1 and 2. (c) Pairwise comparison between class 1 and 3. (d) Pairwise comparison between class 2 and 3.

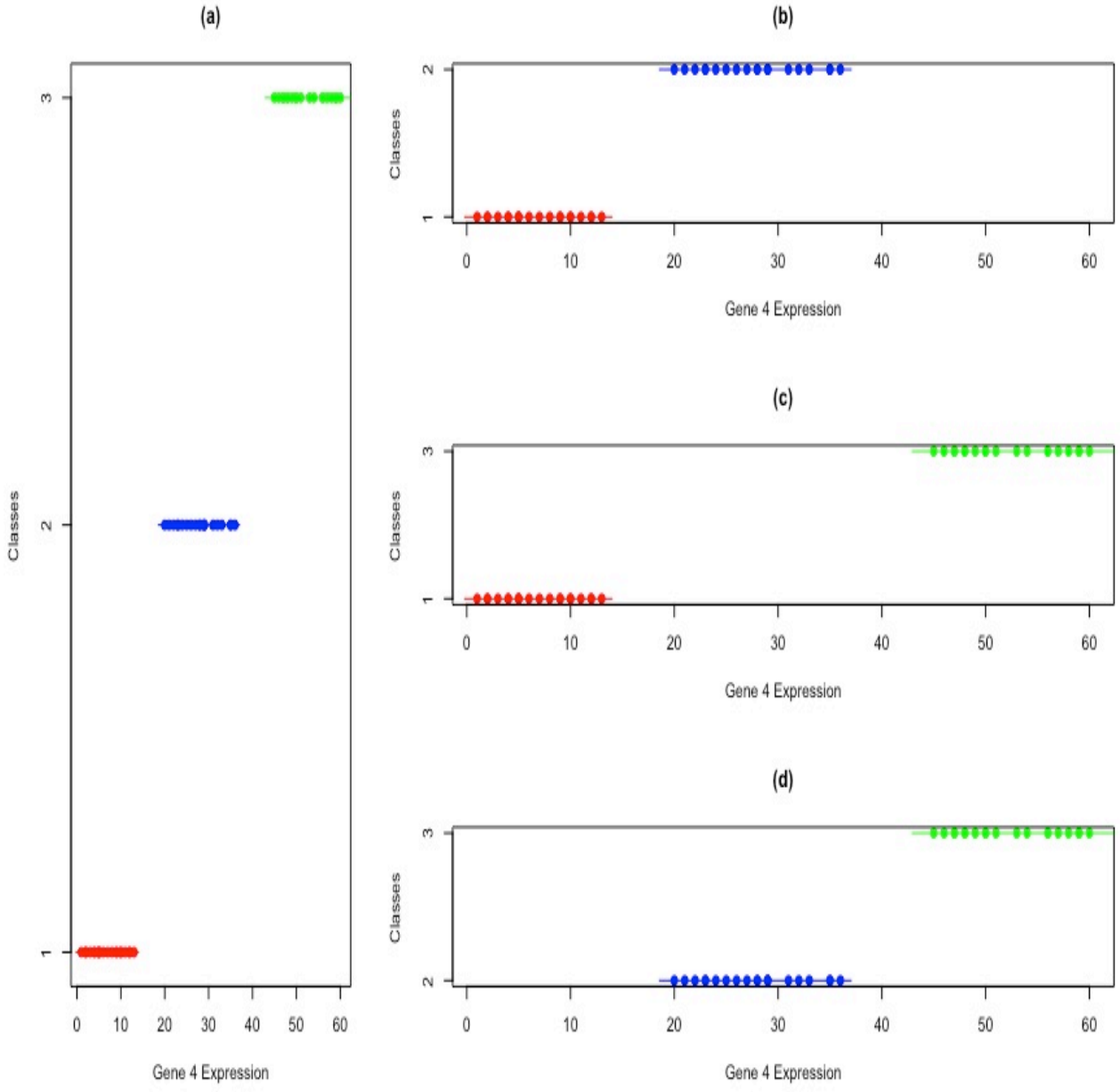

Figure 5.4: Scatter plots of Gene 4 Expression values across three classes. (a) Expression levels for all three classes. (b) Pairwise comparison between class 1 and 2. (c) Pairwise comparison between class 1 and 3. (d) Pairwise comparison between class 2 and 3.

5.6 Experimental Setup

To validate the performance of feature selection techniques, one can assess the accuracy of a classifier applied subsequent to the feature selection process. Consequently, classification accuracy has been employed to summarize the performance of a classification model based only on a subset of selected genes. The performance of the identification of discriminative genes can be verified through assessment. For instance, the authors of [29] utilised classification accuracy in a comparative evaluation of different feature selection methods, and the authors of [143] assessed the influence of feature selection techniques through classification accuracy.

In this study, seven gene expression datasets are used to conduct our experiment. Our proposed method, 3cPOS, is validated by comparing it with three well-known gene selection methods. Moreover, the model performance is assessed through classification accuracy from four classifiers: Random Forest (RF), k nearest Neighbor (kNN), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost).

Twenty repetition of 5-fold cross-validation analysis was performed for each combination of datasets, gene selection techniques, and the number of selected genes (20 informative genes) and the classifiers. For each feature selection technique, the top 20 most informative genes were selected to compare the quality of gene selection using RF, kNN, SVM, and XGBoost classifiers. These classifiers are well-established and frequently employed in machine learning research [197]. Comparison of these classifiers provides valuable insight into the impact of feature selection on model performance and helps identify the most informative features for analyses. The R package "randomForest" [114] is used to implement Random Forest, and its default parameters are 500, the square root of the number of predictores, and 1 for ntree, mtry, and nodesize, respectively. The R package "class" [156] is employed to implement kNN classifiers with a default parameter: k, the closest odd number of neighbors. The R package "e1071" [48] performs Support Vector Machine along with different types of kernels. For simplicity, linear kernel is applied for SVM. The R package "xgboost" [33] is used to conduct Extreme Gradient Boosting.

For each fold, a subset of genes of size r, $r = 1, 2, ..., 20$, is selected using the Kruskal Wallis

Test (Kruskal), Minimum Redundancy and Maximum Relevance (mRMR), the Least Absolute Shrinkage Operator Selector (LASSO), and our proposed method, 3cPOS. The R package "stats" [173], "mRMRe" [45], and "glmnet" [61] are utilized to perform Kruskal, mRMR, and LASSO, respectively.

Some limitations have been involved while implementing "mRMRe" and "glmnet" for some datasets due to the large number of genes and small size of some classes. For the R package on "mRMRe", the mRMR technique cannot be analysed for datasets that includes more than 46340 features. Hence, the datasets of GSE21029, GSE17951, GSE102079, and GSE102287 are excluded from the analysis of the mRMR approach. Similarly, the R package glmnet encounters constraints when a small subset of samples from a class is selected to form training folds. Consequently, the datasets GSE23938, GSE22093, GSE17951, and GSE102287 are not implemented for the LASSO method.

The evaluation is carried out according to the following procedure:

1. Dividing each data set into training and testing data by random splitting. 5-fold cross-validation is applied by conducting 80% for training data and another 20% for testing data. This step is repeated 20 times resulting in 100 runs.

2. Implementing the Kruskal, LASSO, mRMR, and our proposed method, 3cPOS, on the training data to select the ranked top 20 informative genes out of all genes.

3. Fitting Random Forest, K-Nearest Neighbours, Support Vector Machine, and Extreme Gradient Boost on training data using the top r selected informative genes for each $r = 1, 2, ..., 20$ from the ranked gene set of four different feature selection methods.

4. Predicting the class probabilities for the testing data using the fitted classification models, trained on the 20 different sets of genes with sizes $r = 1, 2, ..., 20$.

5. Computing the average classification accuracy based on the predictions and the true class labels of the testing data across the total of 100 runs.

5.7 Results

5.7.1 3cPOS Method Quality Performance

Based on the experimental setup, we evaluated the performance of various feature selection algorithms in terms of classification accuracy. Classification accuracy is the most common performance metric for verifying the effectiveness of feature selection. To compare the performance of feature selection on the given datasets and learning models, we adopt the following criterion: a feature selection method is considered superior if it yields higher classification accuracy compared to other feature selection approaches. A similar evaluation scheme has been used in several studies, including [39, 94, 99, 183]

The average classification accuracy yielded on the GSE23938 datasets using RF, kNN, SVM, and XGBoost classifiers is shown in Figure 5.5. It reveals that 3cPOS performs better than other techniques through RF, k-NN, and SVM. Furthermore, 3cPOS provides 88%, 87%, and 89% classification accuracy through RF, k-NN, and SVM classifiers, respectively. In contrast, Kruskal performs better than other feature selection techniques across XGBoost with 64% classification accuracy.

Figure 5.6 shows the average classification accuracy yields on GSE22093 datasets using RF, kNN, SVM, and XGBoost classifiers. Based on RF, 3cPOS outperforms all other feature selection techniques, starting from the set of 10 informative genes up to the set of 20 informative genes. Kruskal performs better than all other compared techniques based on the k-NN classifier, while mRMR and 3cPOS have the comparable performance at a single informative gene across SVM classifiers. Based on the XGBoost classifiers, 3cPOS is the best technique at the small and large set of informative genes.

Figure 5.7 indicates average classification accuracy obtained with RF, kNN, SVM, and XGBoost classifiers on the on GSE102287 datasets. 3cPOS performs better than Kruskal at a single informative gene through a RF classifier, while 3cPOS underperforms all other techniques based on k-NN classifiers. In contrast, it can be observed that 3cPOS becomes the optimal technique based on SVM and XGBoost classifiers, and it provides 54% and 51% classification accuracy, respectively.

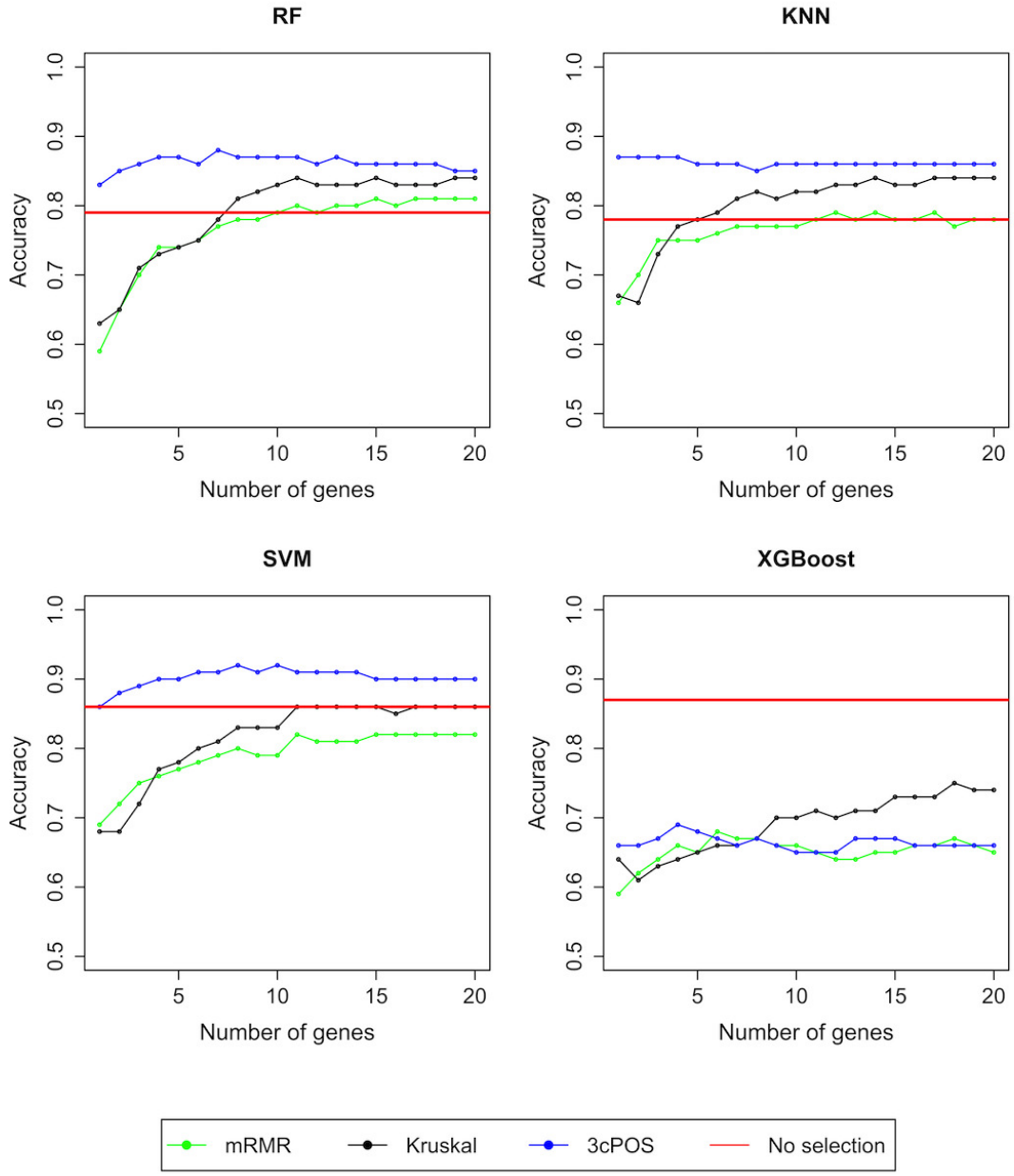

Figure 5.5: Average classification accuracy for GSE23938 based on 20 repetitions 5-fold cross validation using mRMR, Kruskal, 3cPOS, and the full set of features.

The average classification accuracy yielded on the GSE17951 datasets using RF, kNN, SVM, and XGBoost classifiers is shown in Figure 5.8. Our proposed approach, 3cPOS, outperforms all other techniques across RF, kNN, SVM, and XGBoost classifiers, and the highest classification accuracy can be found in the set of 20 informative genes.

Average classification accuracy yields on GSE102079 datasets using RF, kNN, SVM, and XGBoost classifiers are demonstrated in Figure 5.9. 3cPOS performs better than all other selected

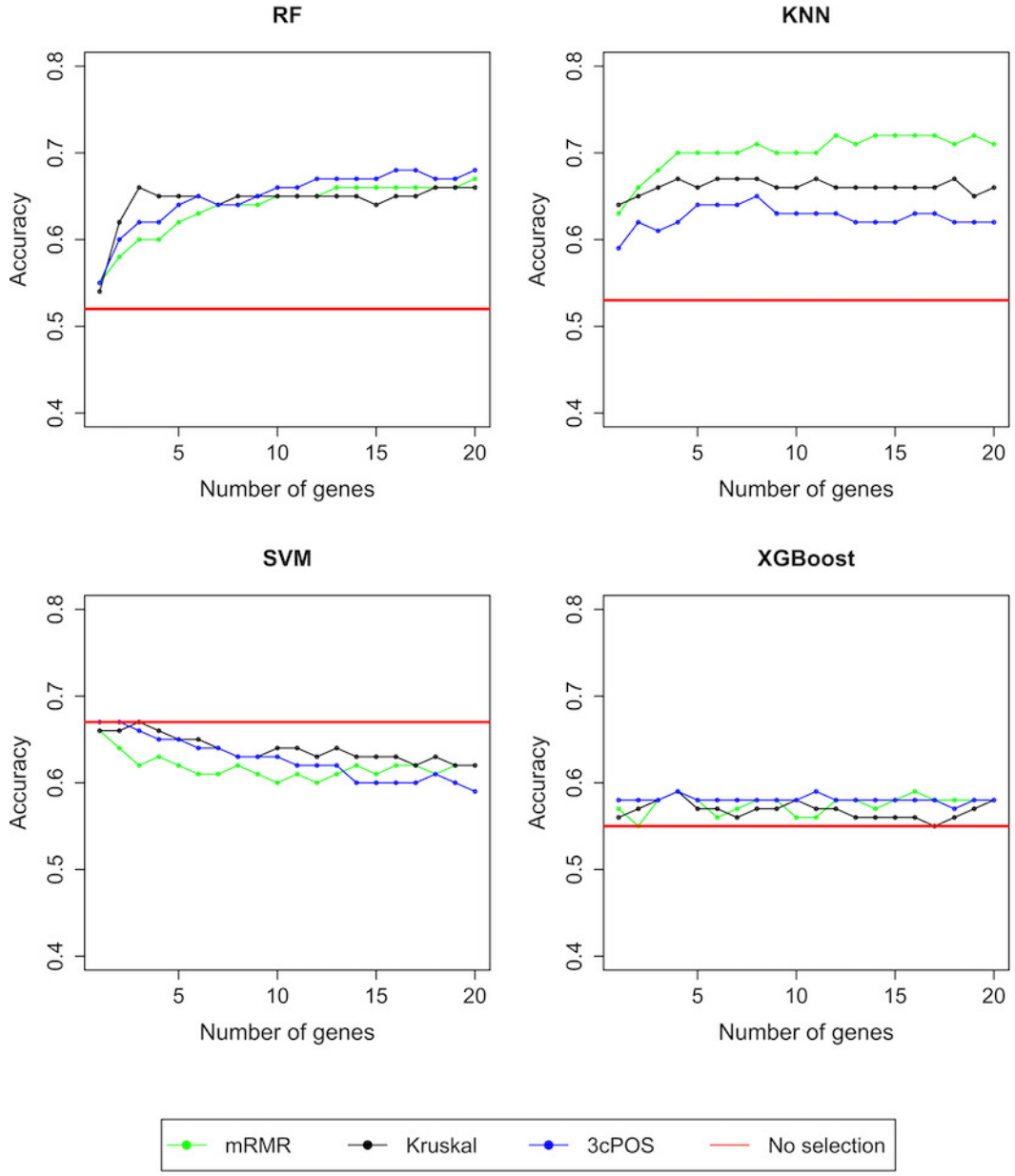

Figure 5.6: Average classification accuracy for GSE22093 based on 20 repetitions 5-fold cross validation using mRMR, Kruskal, 3cPOS, and the full set of features.

approaches and provides 92% classification accuracy based on RF, kNN, SVM, and XGBoost classifiers.

Figure 5.10 shows the average classification accuracy obtained with RF, kNN, SVM, and XGBoost classifiers on the GSE21029 datasets. 3cPOS outperforms all other compared approaches and provides classification accuracy between 89% and 93% across RF, kNN, SVM, and XGBoost classifiers.

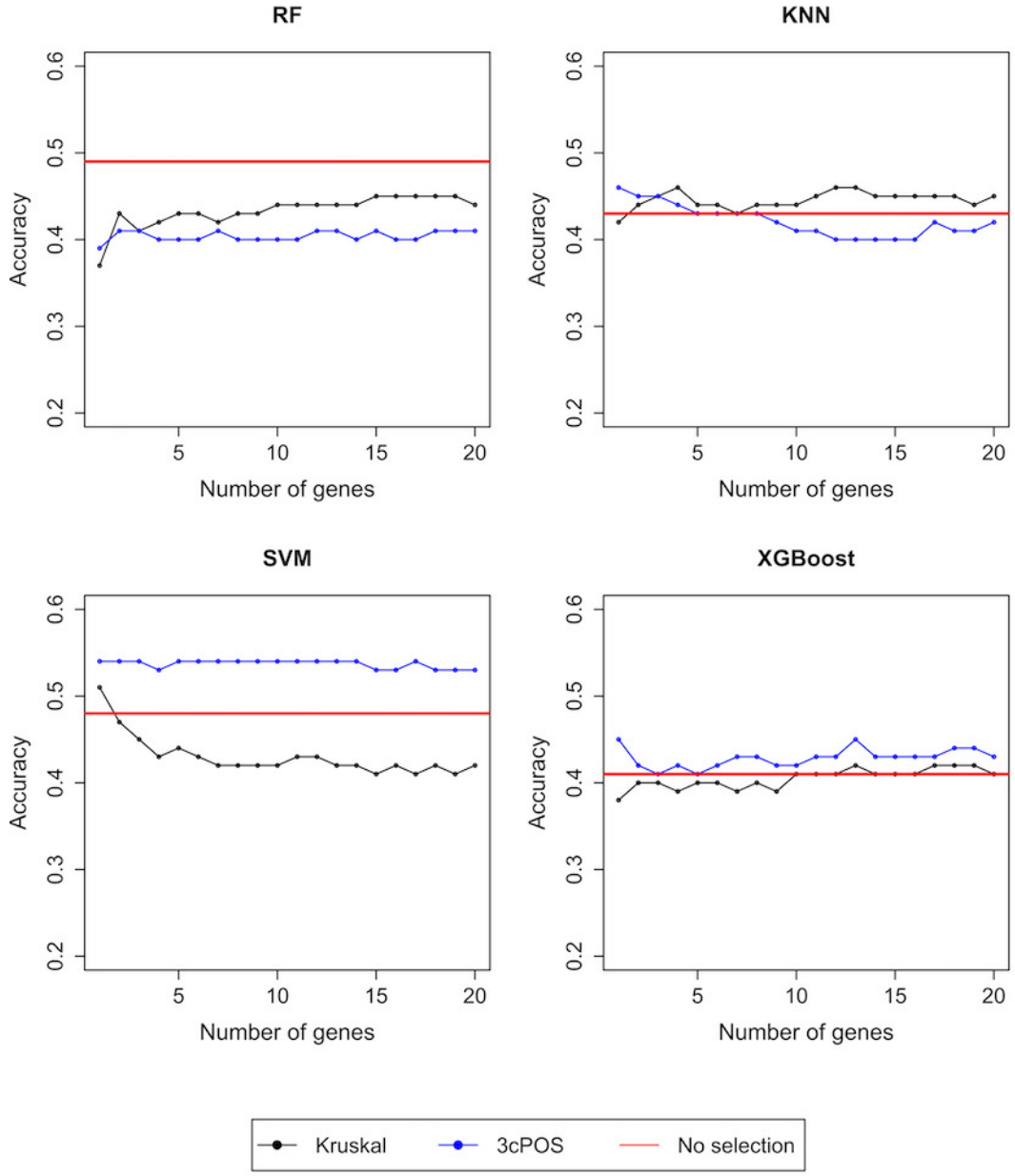

Figure 5.7: Average classification accuracy for GSE102287 based on 20 repetitions 5-fold cross validation using Kruskal, 3cPOS, and the full set of features.

Figure 5.11 demonstrates the average classification accuracy obtained with RF, kNN, SVM, and XGBoost classifiers on the MLL datasets. 3cPOS has outstanding performance based on RF classifiers, while LASSO performs better than all other selected methods across kNN and SVM classifiers, except for the single set of informative genes. Additionally, 3cPOS performs better than all other selected techniques through XGBoost classifiers, excluding the set of 3 to 5 informative genes.

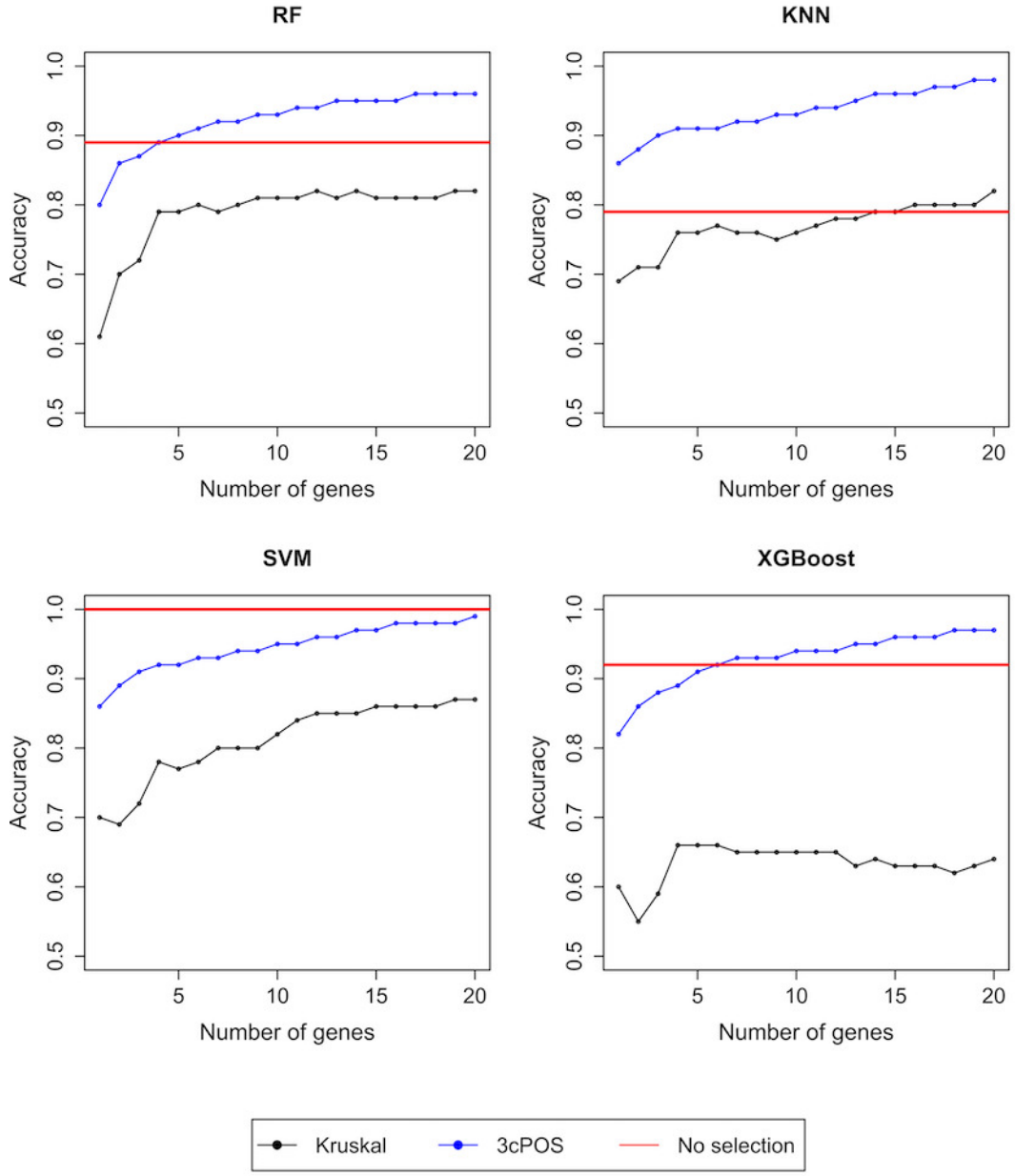

Figure 5.8: Average classification accuracy for GSE17951 based on 20 repetitions 5-fold cross validation using Kruskal, 3cPOS, and the full set of features.

The performance of the compared techniques varies with different gene set sizes, datasets, and classifiers. According to Random Forest (RF) and Extreme Gradient Boosting (XGBoost) classifiers across separate datasets, the 3cPOS algorithm provides superior performance for small and moderate gene set sizes. In contrast, 3cPOS maintains its optimal performance at a single selected gene or large gene set sizes using k-Nearest Neighbors (k-NN) and Support Vector Machine (SVM) classifiers. As a result, the 3cPOS feature selection approach is more

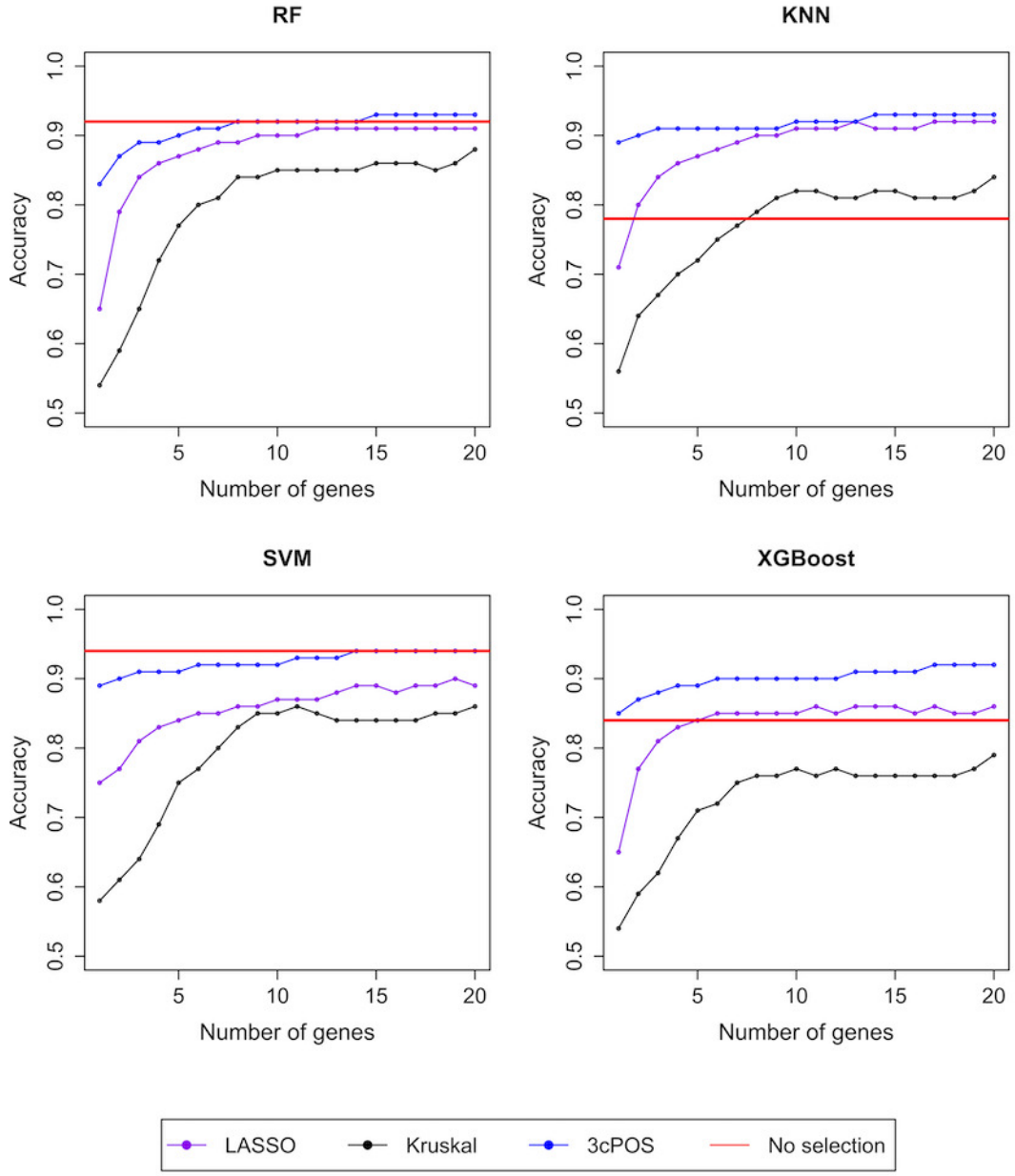

Figure 5.9: Average classification accuracy for GSE102079 based on 20 repetitions 5-fold cross validation using LASSO, mRMR, Kruskal, 3cPOS, and the full set of features.

adaptable to different data patterns and classifier types than the other techniques. In contrast, the performance of the alternative techniques is more sensitive to variations in data characteristics and the choice of classifier.

We also compared the maximum classification accuracies achieved by each method to highlight a comprehensive comparison of the performance of the methods relative to our proposed approach. Each method attains its highest accuracy at a different gene set size. Tables 7.1 and

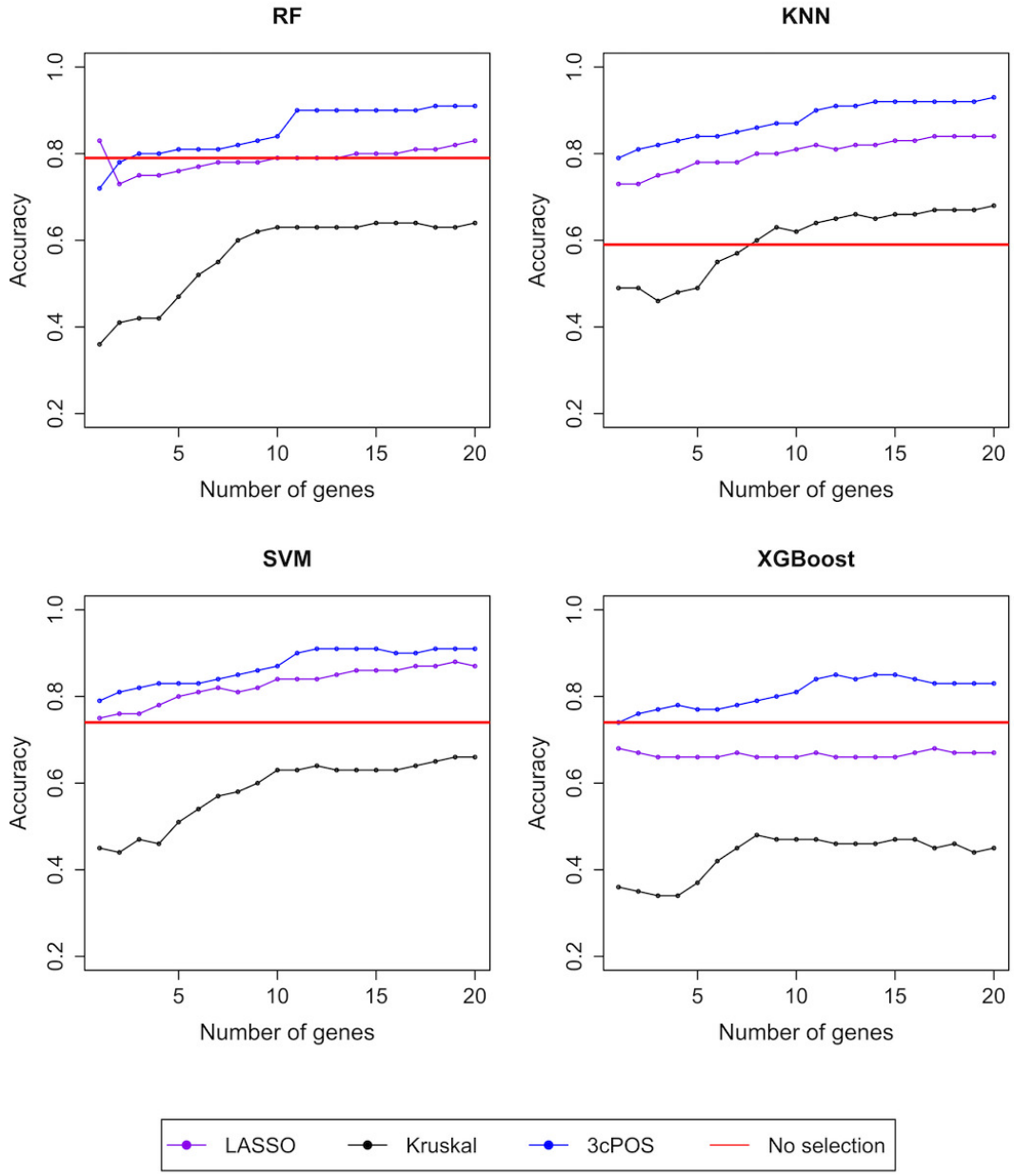

Figure 5.10: Average classification accuracy for GSE21029 based on 20 repetitions 5-fold cross validation using LASSO, Kruskal, 3cPOS, and the full set of features.

7.2 summaries these results for RF, kNN, SVM, and XGBoost classifiers respectively. Each row displays the gene set size (along with its corresponding maximum classification accuracy, shown in brackets) obtained by all methods for a specific dataset, as reported in the first column. Additionally, the classification accuracies for the corresponding classifier using the full set of features, without feature selection, are presented in the sixth and last columns of Tables 7.1 and 7.2. A similar comparison scheme is performed in [136].

Table 5.1: The maximum classification accuracies yielded by Random Forest and k-Nearest Neighbor classifiers with feature selection methods along-with the classification accuracy without selection

	RF					k-NN				
Datasets	LASSO	mRMR	Kruskal	3cPOS	Full set	LASSO	mRMR	Kruskal	3cPOS	Full set
GSE23938		15(0.81)	11(0.84)	4(**0.87**)	0.79		12(0.79)	14(0.84)	1(**0.87**)	0.78
GSE22093		20(0.67)	3(0.66)	16(**0.68**)	0.52		12(**0.72**)	4(0.67)	5(0.64)	0.53
GSE102287			15(0.45)	2(0.41)	**0.49**			4(**0.46**)	1(**0.46**)	0.43
GSE17951			12(0.82)	17(**0.96**)	0.89			20(0.82)	19(**0.98**)	0.79
GSE102079	12(0.91)		15(0.86)	15(**0.93**)	0.92	17(0.92)		20(0.84)	14(**0.93**)	0.78
GSE21029	20(0.83)		15(0.64)	18(**0.91**)	0.79	17(0.84)		20(0.68)	20(**0.93**)	0.59
MLL	16(0.90)	20(0.91)	20(0.63)	20(0.91)	**0.94**	19(**0.91**)	19(0.87)	19(0.68)	17(0.87)	0.80

The numbers outside brackets represent the size of the gene set that corresponding to the maximum classification accuracy. The boldface numbers in brackets indicate the the highest classification accuracy among the compared methods for the corresponding datasets, while blank spaces indicate where no analysis or implementation was performed.

Table 5.2: The maximum classification accuracies yielded by Support Vector Machine classifier with feature selection methods along-with the classification accuracy without selection

	SVM					XGBoost				
Datasets	LASSO	mRMR	Kruskal	3cPOS	Full set	LASSO	mRMR	Kruskal	3cPOS	Full set
GSE23938		15(0.82)	11(0.86)	6(**0.91**)	0.86		6(0.68)	18(0.75)	4(**0.69**)	0.87
GSE22093		1(0.66)	3(**0.67**)	1(**0.67**)	**0.67**		4(**0.59**)	4(**0.59**)	4(**0.59**)	0.55
GSE102287			1(0.51)	1(**0.54**)	0.48			13(0.42)	1(**0.45**)	0.41
GSE17951			19(0.87)	20(0.99)	**1.00**			4(0.66)	18(**0.97**)	0.92
GSE102079	14(0.89)		11(0.86)	14(**0.94**)	**0.94**	11(0.86)		20(0.79)	17(**0.92**)	0.84
GSE21029	19(0.88)		19(0.66)	12(**0.91**)	0.74	1(0.68)		8(0.48)	14(**0.85**)	0.74
MLL	18(0.94)	17(0.86)	11(0.63)	19(0.90)	**0.96**	17(0.86)	20(**0.89**)	11(0.49)	18(**0.89**)	0.87

The numbers outside brackets represent the size of the gene set that corresponding to the maximum classification accuracy. The boldface numbers in brackets indicate the the highest classification accuracy among the compared methods for the corresponding datasets, while blank spaces indicate where no analysis or implementation was performed.

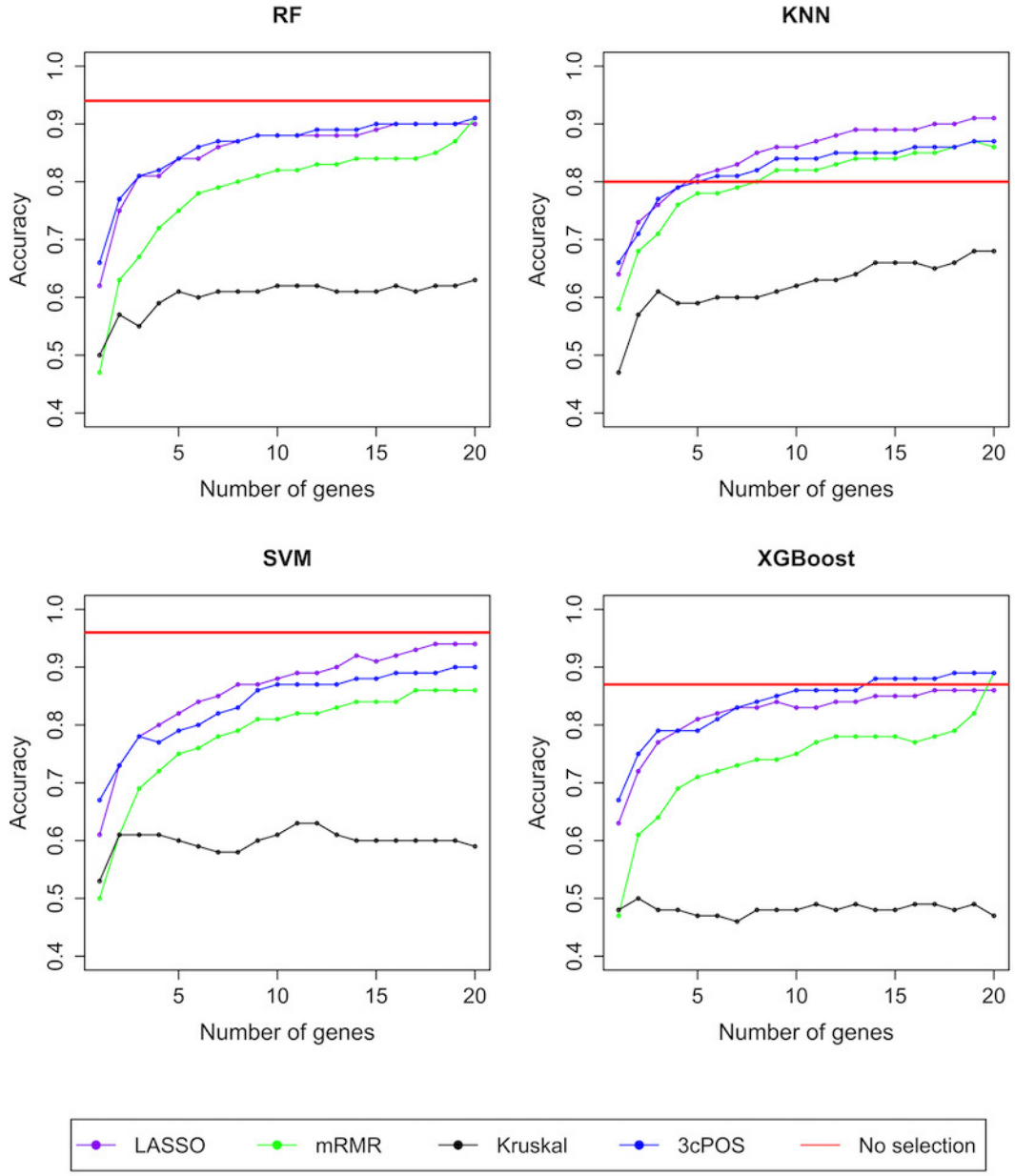

Figure 5.11: Average classification accuracy for MLL based on 20 repetitions 5-fold cross validation using LASSO, mRMR, Kruskal, 3cPOS, and the full set of features.

5.7.2 Stability Evaluation for 3cPOS Method

It is anticipated that an effective feature selection method will produce consistent outcomes across multiple sub-samples of the considered dataset. In biomarker selection, the key to finding a stable feature subset is to focus on frequently selected biological markers. It is also important to consider randomly chosen features. A well-known method like stability index was proposed

by [109] which is used to assess the stability of comparative methods at different feature set sizes. $1/\lambda$ can be used to calculate the stability values, where λ represents the total number of sub-samples used (in our example, $\lambda = 100$). Since the stability score ranges from 0 to 1, a larger value that is closer to 1 indicates full stable selection.

Figures 5.12 (a), (c), and (f) show that the 3cPOS method achieves superior stability at different set sizes across GSE23938, GSE102287, and GSE21029 datasets. These datasets are characterized by small sample sizes and class imbalance, e.g., 5/7/29 in GSE23938, 36/19/11 in GSE102287, and 19/17/26 in GSE21029. From these scenarios, it can be observed that feature selection is likely to be sensitive to perturbations in the training phase because of the under-representation of small class samples from certain sub-samples. Methods such as LASSO, mRMR, and Kruskal produce unstable feature ranking. However, 3cPOS can deal with sensitivity to sampling variation by evaluating the consistency of features across class distributions in the training phase. This results in improved stability, particularly in imbalanced and small-sample settings [78, 125].

In contrast, Figures 5.12 (b), (d) and (e) demonstrate that the Kruskal method achieves higher stability across GSE22093, GSE17951, and GSE102079 datasets. Moreover, Figure 5.13 shows that the Kruskal method performs superior stability on the MLL dataset. It can be observed that these datasets are characterized by larger sample sizes and more dominant class-specific expression patterns, e.g., 6/28/69 in GSE22093, 32/13/109 in GSE17951, 91/152/14 in GSE102079, and 24/28/20 in MLL. In such conditions of larger sample sizes and more extreme class imbalance, Kruskal shows strong and consistent marginal differences between classes. This consistency helps produce the same top-ranked genes across different sub-samples, resulting in high stability [160].

Figure 5.12 (a) and (b) and Figure 5.13 reveal that mRMR achieve lower stability on GSE23938, GSE22093, and MLL datasets. The trade-off in redundancy-based selection is reflected in these scenarios. Although mRMR aims to select complementary genes, when several features have similar relevance, it may choose different correlated genes across several sub-samples. Consequently, sets of selected genes are varied, which reduces stability, especially when smaller feature subsets are taken into account [160]. Similarly, Figure 5.12 (e) and (f)

display that LASSO performs lowest stability on GSE102079 and GSE21029 datasets. Despite its advantages in regularization, the LASSO is nonetheless susceptible to sample variability when highly correlated genes vie for inclusion in sparse models [98].

These findings suggest that the stability of feature selection is not only influenced by the selection criterion, but also the robustness of the selection technique, dimensionality, class imbalance, as well as dataset complexity. 3cPOS achieves higher stability in gene expression datasets characterized by small sample sizes as well as class imbalance, where feature selection is sensitive to sampling fluctuation. However, Kruskal is effective when datasets are characterised by clearer class separation and dominant biomarkers, typically associated with larger sample sizes despite the presence of class imbalance. Therefore, stable feature selection is crucial for finding reliable biomarkers, enhancing reproducibility, and boosting the confidence in selected genes. This reflects true biological signals rather than sampling variability artefacts [53, 98].

The relevance of the selected features to the considered response of the target class labels is not ensured by a stable selection. It's also important to emphasise the prediction accuracy of a classifier using the selected features. For the relationship between accuracy and stability, the GSE21029 dataset has been examined in Figures 5.14.

The stability scores were integrated with the corresponding classification accuracy yielded by four different classifiers: RF, kNN, SVM, and XGBoost classifiers. Different set sizes of selected features correspond to different dots for the same feature selection technique. As a result of increasing stability scores from the bottom to the top on the vertical axis and growing classification accuracy from left to right on the horizontal axis, the optimal method is the one whose dots are indicated in the upper-right corner of the plot. For all classifiers, our proposed method generate a good trade-off between stability and accuracy for GSE21029 dataset, see Figure 5.14.

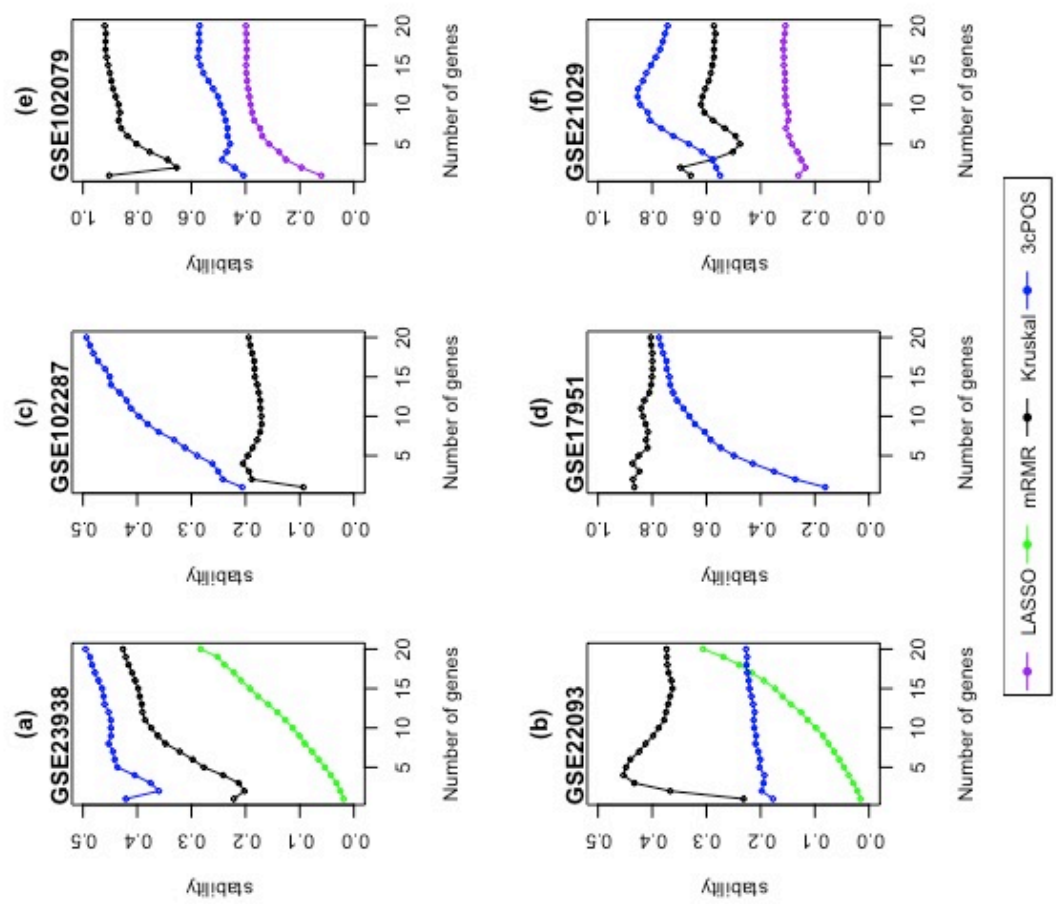

Figure 5.12: Stability scores for 6 datasets at different set sizes that selected by LASSO, mRMR, Kruskal, and 3cPOS: (a) GSE23938 dataset, (b) GSE22093 dataset, (c) GSE102287 dataset, (d) GSE17951 dataset, (e) GSE102079 dataset, and (f) GSE21029 dataset.

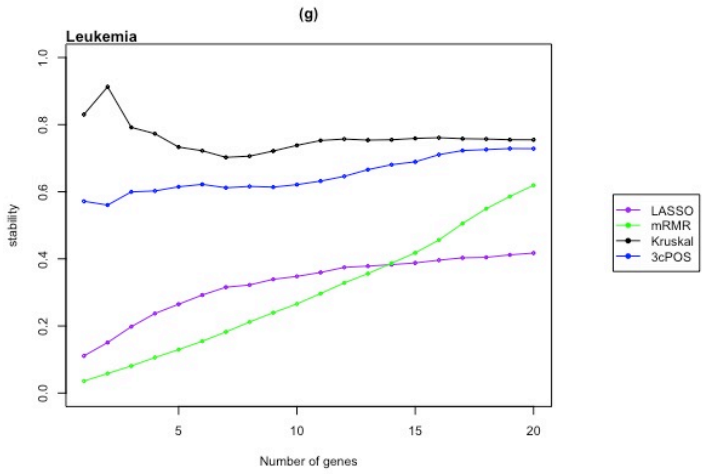

Figure 5.13: Stability scores for MLL datasets (g) at different set sizes that selected by LASSO, mRMR, Kruskal, and 3cPOS.

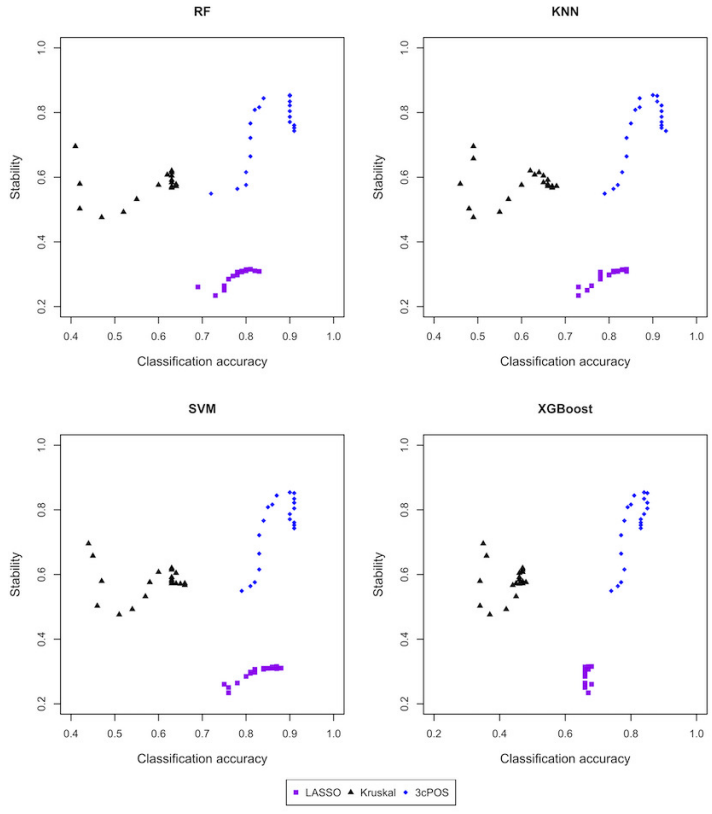

Figure 5.14: Stability - accuracy plot for GSE21029 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE21029 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

5.7.3 Computational Complexity for 3cPOS Method

Feature selection is employed to target the most informative features across the entire dataset to help improve the performance of machine learning models. The evaluation of model performance can be yielded through classification accuracy and stability, as shown in previous sections; however, it may not be sufficient. To cover this gap, computational complexity is considered for the efficiency, scalability, and practicality of feature selection techniques, particularly in real-world applications where datasets are high-dimensional. [191, 25] have examined time complexity to evaluate the efficiency of feature selection algorithms.

An algorithm's time complexity provides a formal evaluation of the computer resources needed. This highlights the duration of execution as a function of the input size by focusing on the growth of an algorithm's running time relative to increasing input size, typically expressed as n. To consider this behavior, Big O notation [37] is used to describe the upper bound of an algorithm's time complexity using mathematical notation. This approach offers systematic evaluation and comparison of algorithmic efficiency, resulting in a fundamental tool in algorithm design and selection, guiding the development of scalable as well as efficient computational solutions.

For the 3cPOS method's time complexity, let m and N be the number of genes (features) and the number of samples per gene in the dataset, respectively. The 3cPOS feature selection method operates on c classes where c is fixed at 3. The time complexity of each step of the 3cPOS algorithm, as outlined in Algorithm 1 and Sections 5.2-5.4 of this thesis, is evaluated using Big O notation. The overall time complexity of the 3cPOS algorithm is then summarised as follows.

1. **Z-score Standardisation**: The expression values of each gene are standardized into standardised values by calculating the mean and standard deviation (line 4 in Algorithm 1). This involves iterating through all N samples, resulting in $O(N)$. All m genes are implemented, the total time cost of this step is $O(m \cdot N)$.

2. **Core Interval Calculation**: The core expression interval is computed for each class c for gene i by calculating the class-specific mean and standard deviation of that gene's N standardized values (line 8 in Algorithm 1). By doing this, $O(N)$ is contributed for

each gene. However, the number of classes is fixed at 3, the loop iterates across m genes. Therefore, the total time cost of this step results in $O(m \cdot N)$ overall.

3. **Non-Outlier Counting**: The number of non-outlier samples is computed by checking each of the N samples and determining if its value $z_{i,j}$ lies within its own class core interval $I_{i,c}$ (line 10 in Algorithm 1). This is crucial for obtaining the count for a given gene. In the worst-case scenario, this contributes $O(N)$ per gene. Therefore, the total time cost of this step results in $O(m \cdot N)$ across all m genes.

4. **Overlapping Region Determination**: To estimate the class's expression intersection, the overlap between class core intervals is assessed. Initially, the lengths of all two-way overlapping regions, for each pair of classes, and the length of three-way overlapping region are computed (lines 13–14 in Algorithm 1). Only three possible pairs of two-way overlaps and a three-way overlap are considered because c is fixed at 3. By calculating these intersections, constant time $O(1)$ is contributed per gene. In contrast, the number of two-way overlapping samples as well as the number of three-way overlapping samples are computed by counting samples that fall within these overlapping regions (lines 15–16 in Algorithm 1). By doing this, N samples are validated to count those falling in overlaps, resulting in $O(N)$ per gene. By computing both overlap lengths and overlapping samples, the total time cost of this step results in $O(m \cdot N)$ across all m genes.

5. **3cPOS Score Computation**: By including both the two-way overlap and three-way overlap scores, the 3cPOS score is computed (line 18 in Algorithm 1). This contributes $O(1)$ for each gene. For all m genes, the total time cost of this step is $O(m)$.

6. **Final Selection**: For all m genes, 3cPOS scores are sorted in ascending order to rank the top r genes (line 21 in Algorithm 1). This contributes a total time cost of this step as $O(m)$.

Overall time complexity, the dominant cost of the 3cPOS algorithm is scaled on the order of $O(m \cdot N)$ by considering both the standardisation and overlap analysis. Because the number of classes c is fixed at 3 (a constant), class-dependent loops and overlap calculations do not alter the asymptotic order (they only multiply the workload by a constant factor). The final selection

step contributes $O(m)$. For the worst-case scenario, the time complexity of the 3cPOS method can be denoted as

$$O(m(N+1)) \tag{5.14}$$

To provide a comprehensive evaluation of computational efficiency, Table 5.3 shows a comparison of the time complexity of the 3cPOS alongside LASSO, mRMR, and the Kruskal methods using Big O notation. Table 5.3 reveals that the relative scalability of 3cPOS with respect to both sample size and feature dimensionality.

Table 5.3: Comparison of theoretical time complexity for different feature selection methods

Methods	**Theoretical Time Complexity**
LASSO	$O(N \cdot m \cdot I)$, where I denotes the number of iterations
mRMR	$O(m^2 \cdot N)$
Kruskal	$O(m \cdot N \log N)$
3cPOS	$O(m(N+1))$

5.8 Summary

This chapter discusses the concept of selecting informative genes by verifying the overlap of their expression within three different phenotypes while taking into account the percentage of overlapping samples. Core intervals are assigned to prevents the impact of outliers. In addition, the 3-class Proportional Overlapping Score (3cPOS) is proposed to estimate the overlapping degree between classes for each gene, with an increasing 3cPOS score highlighting larger overlapping intervals. Consequently, smaller 3cPOS scores are used to identify informative gene sets.

Using examples from Gene 1 and Gene 4 Expression, we demonstrate how to calculate 3cPOS scores. It shows that Gene 1 Expression is less capable to differentiate between classes, compared to Gene 4 Expression with two primary reasons; the number of overlapping samples, and the proportion of the contribution of classes to the overlapped samples.

Our novel approach is also applied to seven publicly available gene expression datasets with different expression patterns. The informative gene sets of different sizes, up to 20 genes, are selected using widely used feature selection techniques: the Kruskall-Wallis Test (Kruskal), Minimum Redundancy and Maximum Relevance (mRMR), Least Absolute Shrinkage Operator Selector (LASSO), and our proposed approach, 3cPOS. Then, four different classifiers Random Forest; k Nearest Neighbor; Support Vector Machine; Extreme Gradient Boost; are utilised to construct classification models along with the sets of informative genes. The average classification accuracy given by the considered classifiers is employed for assessing the performance of 3cPOS.

For the Random Forest classifier, 3cPOS outperforms all other compared feature selection techniques on the datasets ‘GSE23938’, ‘GSE17951’, ’GSE21029’, and ‘MLL’, at the different sets of all informative genes that have been evaluated. 3cPOS also performs better than the compared feature selection methods on ’GSE22093’at the small set (i.e., < 6), and large sets of informative genes (i.e., > 10). 3cPOS also outperforms all other compared feature selection methods on ‘GSE102287’ at the small sets of informative genes (i.e., < 2). On average, 3cPOS boosts the performance of the compared methods by up to 96% of the classification accuracies

conducted using their selections at different sets of informative gene sizes.

For the k Nearest Neighbour classifier, 3cPOS outperformed all other methods on 'GSE23938', 'GSE17951', and 'GSE102079' at all different sets of informative genes that have been investigated. 3cPOS performs better than compared selection methods on 'GSE102287' at the informative gene (i.e., $= 1$), while the performance of 3cPOS is optimal on 'GSE102287' excepting a single set of an informative gene (i.e., > 2). On average through all experiments, the 3cPOS technique improves the performance of the compared methods by up to 98% of the classification accuracy achieved using their selections at various sets of sizes.

For the Support Vector Machine classifier, 3cPOS outperformed all other methods on 'GSE23938', 'GSE102287', 'GSE17951', 'GSE102079', and 'GSE21029' at at all different sets of informative genes. In addition, 3cPOS performed better compared selection methods on the 'GSE23938' dataset with the set of only four informative genes. On average across datasets, 3cPOS improves the performance of the compared methods by up to 98% of the classification accuracies created using their selections at different sets of informative genes. In constant, LASSO provides the maximum classification accuracy on 'MLL' with 91% classification accuracy.

For the Extreme Gradient Boost classifier, 3cPOS performed more than all other methods on 'GSE102287', 'GSE17951', 'GSE102079', and 'GSE21029' at all the different sets of informative genes. The 3cPOS produces the highest classification accuracy, which is 98%. Additionally, the 3cPOS approach outperforms all compared techniques on 'GSE22093' and 'MLL' at the small sets of informative genes (i.e., < 4) with 63% and 76% on average classification accuracy, respectively. Nevertheless, Kruskal provides the maximum classification accuracy on 'GSE23938' with 78% classification accuracy.

The stability of the selections produced by the compared feature selection techniques using cross-validation has been evaluated. The stability scores obtained from the different set sizes of selected features indicate that the 3cPOS method is likely to provide enhanced stability when applied to gene expression datasets with few samples and class imbalance. Furthermore, the 3cPOS shows a good trade-off between stability and classification accuracy.

The computational complexity of the 3cPOS method compared with other feature selection techniques is evaluated. 3cPOS method not only shows superior performance in terms of

classification accuracy and stability, but also meets a relatively low computational cost.

Overall, our novel approach, 3cPOS, performs better than the Kruskal, mRMR, and LASSO techniques. Based on RF classifiers, 3cPOS outperforms other comparative selection approaches in 6 of the 7 datasets. On 4 out of 7 datasets across k-NN classifiers, 3cPOS surpasses all other methods; on 5 out of 7 datasets within SVM classifiers it outperforms other selection procedures. Furthermore, 3cPOS beats all other feature selection approaches on 6 out of 7 datasets through XGBoost classifiers. This implies that the 3cPOS method is more resilient and adaptable when dealing with various classifier combinations and different data patterns.

CHAPTER 6

Minimum Subset of Genes

6.1 Introduction

Gene scores have been demonstrated to enhance predictive power in Chapter 5; however, they may not be sufficient to further improve predictive accuracy due to the presence of redundant information and imbalanced class sizes. [6, 7, 123] have proposed approaches that integrates a minimum subset of genes with gene ranking to determine the final gene selection. These combinations have significantly improved classification performance.

This chapter focuses on the procedure of final gene selection, in which a minimum subset of genes and gene ranking are incorporated with gene scores. Firstly, the details of gene masking are introduced to validate the discriminating power of genes, as discussed in Section 6.2.1. To mitigate the influence of expression outliers and eliminate redundant information, a minimum subset of genes is then considered, as elaborated in Section 6.2.2. Additionally, the Relative Dominant Class (RDC) is employed to address misleading assignments that may arise from imbalanced class sizes, as detailed in Section 6.2.3.

The final gene selection process, which is the culmination of this analysis, incorporates both the minimum subset of genes and gene ranking. We introduce two distinct approaches to derive the final gene set. These approaches are evaluated on the remaining genes, specifically those not included in the minimum subset. The first approach utilises both the 3cPOS score and RDC to

establish gene rankings, whereas the second approach relies solely on the 3cPOs score for gene ranking. Further information on these approaches can be found in Section 6.2.4.

6.2 The method

The 3cPOS technique provides efficient 3cPOS scores, expressed in Chapter 5, which sort genes based on their discriminative power, genes with a higher rank representing higher 3cPOS scores. This determines the capability of genes to identify the considered target classes.

6.2.1 Gene Masks

Gene masks are assigned to each gene based on their standardised expression values and core expression intervals that is presented in Section 5.2. Gene i's mask provides information on samples that gene i can detect to their correct target classes in the set of non-overlapping samples. Therefore, a gene mask represents a gene's discriminating power. For a certain gene i, sample j of its mask is set to 0 or 0.5 if the corresponding standardised expression values x_{ij} fall within the interval of three-way overlapping region or interval of two-way overlapping region, respectively. Otherwise, it is set to 1.

The gene masks matrix, $G_i = [g_{ij}]$, is defined in which the mask of gene i is indicated by G_i (the ith row of G) such that gene mask element g_{ij} is expressed as;

$$g_{ij} = \begin{cases} 0 & \text{if} \quad j \, \varepsilon \, l_i^{(3)} \\ 0.5 & \text{if} \quad j \, \varepsilon \, l_{i(c_1c_2)}^{(2)}, l_{i(c_1c_3)}^{(2)}, l_{i(c_2c_3)}^{(2)} \\ 1 & \text{otherwise} \end{cases} \tag{6.1}$$

6.2.2 Identifying the Minimum Subset of Genes

The information provided by the 3cPOS scores, outlined in Section 5.4, and gene mask, described in Section 6.2.1 are exploited to determine the minimum subset of genes. The goal of this subset is to determine the minimum set of gene that classify correctly the maximum set of samples in

the training phrase. This procedure helps mitigate the effects of expression outliers as well as removing redundant information (e.g., genes with similar expression profiles)[123].

To find the best coverage in the training phase, the generated 3cPOS measure along with the gene mask are exploited to select the minimum subset of genes. Let G be the set of the entire genes (so the total number of genes is p, which means $\mid G \mid = p$). Also, let $M..(G)$ represent the combined mask of genes. This combined mask is made by doing a logical OR (also called disjunction) between all masks corresponding to genes that belong to the set. It can be denoted as:

$$M..(G) = M_{1.} \vee ... \vee M_{p.} \tag{6.2}$$

The ambition is to search for the minimum subset, denoted by G^*. [6] has exploited the greedy search approach to search for the minimum subset of genes. The pseudo code for our proposed method is demonstrated in Algorithm 2. The matrix of gene masks, M; the combined mask of genes, $M..(G)$; and 3cPOS scores are taken as inputs. The minimum set of genes, G^*, is generated as output.

Algorithm 2 Greedy Search Approach - Minimum set of genes

Input: M, $M..(G)$ and 3cPOS for all genes.
Output: G^*.

1: $k = 0$ {Initialization}
2: $G^* = \phi$
3: $M..(G^*) = 0_N$
4: **while** $M..(G^*) \neq M..(G)$ **do**
5: $k = k + 1$
6: $S_k = \arg\max_{i \in G}(\sum_{j=1}^{n} I(g_{ij} = 1))$ {Assigning gene set whose masks have the maximum bits of 1}
7: $g_k = \arg\min_{i \in S_k}(3cPOS_i)$ {Choosing the candidate with the lowest 3cPOS score among the assigned set}
8: $G^* = G^* + g_k$ {Updating the target set by including the selected candidate}
9: **for all** $i \in G$ **do**
10: $M_{i.}^{(k+1)} = M_{i.}^{(k)} \wedge !M_{..}(G^*)$ {Updating gene masks such that the uncovered samples are only considered}
11: **end for**
12: **end while**
13: **return** G^*

At the initial step $(k = 0)$, we determine $G^* = \phi$ and $M..(G^*) = 0_N$ (lines 3); where $M..(G^*)$ is the combined gene mask of the set G^*, while 0_N is a vector of zeros with the length N. Then,

at each iteration, k, the following steps are performed:

1. The gene that has the highest number of bits equals 1 is included in the set s_k (line 6). As long as the loop condition, $M..(G^*) \neq M..(G)$ is still satisfied, this set cannot be empty. Under this condition, our selected genes do not cover the maximum number of observations that should be covered by the target gene set. Note that the definition of gene masks enables $M..(G)$ to determine in advance which observations must be covered by the minimum subset of genes. Consequently, there will always be at least one gene mask with at least one bit set to 1 in order for this condition to be satisfied

2. if there are multiple genes that exhibit the same maximum number of bits set to 1, the selection process prioritises the gene with the smallest value of the 3cPOS scores (line 7). It is given by g_k.

3. The selected genes, g_k, are added to the set G^* (line 8).

4. All the gene masks are updated by performing a logical AND with the inverted combined mask of the set G* (this happens on line 10). The inverted mask, $!M_{..}(G^*)$, is derived by applying logical negation (logical complement) of the mask, $M_{..}(G^*)$. As a result, the bits of ones corresponding to the classification of still uncovered observations are only considered. Note that $M_{i.}^{(k)}$ represents the updated mask of gene i at the k^{th} iteration, while $M_{i.}^{(1)}$ is the original mask of gene i, with its elements computed according to Equation 6.1.

5. The procedure is repeated and ends when all updated gene masks have no 1 bits in the process updated mask, i.e. the maximum number of samples is covered by the selected genes; $M..(G^*) = M..(G)$.

This process determines the minimal gene set needed to provide a given training set the best classification coverage. Moreover, genes are arranged in descending order in the minimum set G^* based on the number of 1 bits.

6.2.3 Relative Dominant Class

The concept of the Relative Dominant Class (RDC) is crucial for associating each gene with the class it is most likely to distinguish. According to [123], the RDC can mitigate misleading assignments resulting from imbalanced class sizes by identifying the dominant class of a gene based on its relative role. We utilised this approach to assign each gene to its relative dominant class by considering samples from a set of non-overlapping samples. Specifically, gene masks, as defined in Section 6.2.1, are employed to designate each gene to its RDC. These samples represent those that can be accurately classified into their respective target classes. This can be formally expressed as follows:

$$RDC_i = \arg\max_c \left(\frac{\sum_{j \in N_c} I(g_{ij} = 1)}{\mid N_c \mid}\right) \tag{6.3}$$

where N_c is the set of class c, while $\sum_c \mid N_c \mid$ represents the total number of samples that belong to class c. The class with the highest proportion indicates the relative dominant class of gene i. However, in cases where there is no single highest proportion among three-class problems, ties are resolved by randomly assigning the gene to one of the three classes. Genes are assigned to their RDC to associate each gene with the class, it is more able to distinguish. As a result, the number of selected genes could be balanced per class at the final selection process when the RDC is taken into account.

6.2.4 Final Gene Selection

We have reached the final step in which yields the final gene selection. Relevant works are presented in [6] and [123]. We propose two distinct ideas as outlined below:

Idea 1

In this idea, we follow the concepts that have been proposed by [123]. The process of gene rank is determined by considering both 3cPOS scores and RDC. For each Relative Dominant Class c (where $c = 1, 2, 3$), all genes that were not selected in the minimum set, G^*, identified by Algorithm 2, and for which the RDC equals c, are sorted in ascending order based on their

3cPOS values. Thus, given three separate groups of ranked genes (one for each class), the top gene from each group is chosen in a round-robin fashion to create the final gene ranking list.

The minimum subset of genes, G^*, is expanded by including the top v ranked genes from the gene ranking list. Here, v represents the number of genes needed to extend the minimum subset to the total number of desired genes, r, which is specified by the user as an input to the 3cPOS method. The final gene selection comprises the minimum subset of genes as well as gene ranking using Idea 1. The building blocks of the 3cPOS method with selecting final gene selection based on idea 1 are shown in Figure 6.1.

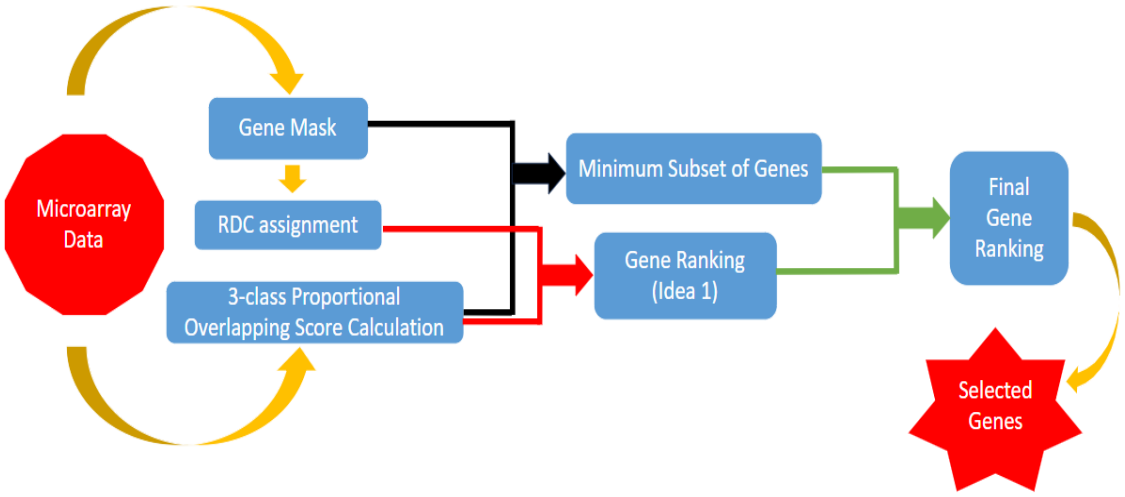

Figure 6.1: Building blocks of the 3cPOS method with selecting final gene selection based on Idea 1

Idea 2

The gene ranking process is based solely on 3cPOS scores. All genes that were not included in the minimum set, G^*, as determined by Algorithm 2, are sorted in ascending order according to their 3cPOS values. The minimum subset of genes, G^*, is expanded by adding the top v ranked genes from the gene ranking list. Here, v is the number needed to reach the user-specified total r for the 3cPOS method. The final set includes minimum subset of gene (G^*) and the additional genes (gene ranking from Idea 2). The building blocks of the 3cPOS method with selecting final gene selection based on Idea 2 are shown in Figure 6.2.

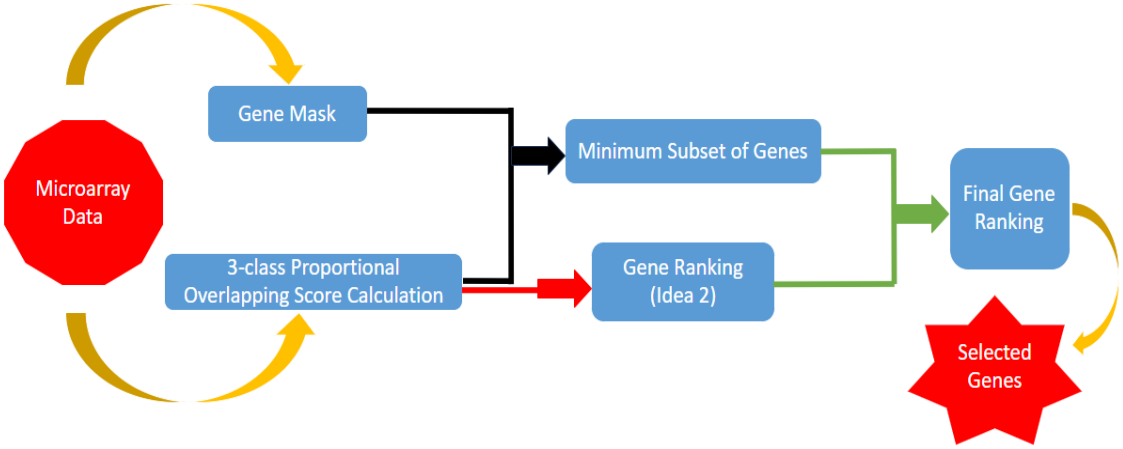

Figure 6.2: Building blocks of the 3cPOS method with selecting final gene selection based on Idea 2

6.2.5 Illustrative Examples

To demonstrate the steps of obtaining final gene subset, presented in Section 6.2.4, we provide two simulated examples.

The first example, seen in Figure 6.3, is mainly focusing on Idea 1 to provide the final gene selection. Each gene is associated with its constructed gene mask, its 3-class proportional overlapping score (3cPOS), and its relative dominant class (RDC), presented in Figure 6.3 (a). For instance, genes with updated masks are considered to focus only on uncovered samples by the selected gene, g_9, (i.e., the second and third samples). The best updated masks (i.e.,g_1, g_4, g_6, and g_{10} which all have the same number of 1 bits) are then considered. g_1 is selected as the second member of the minimum gene subset due to its lower 3cPOS score. The best updated masks (i.e.,g_4 and g_6) are considered. g_4 is selected due to its lowest 3cPOS score. In this example, the minimum number of genes consists of three genes; g_9, g_1, and g_4 . Figure 6.3 (b) reports the chosen minimum subset. The remaining genes are categorised by relative dominant class (RDC) and sorted according to 3cPOS in an ascending order within each category of RDC. The procedure of gene ranking is accomplished by selecting the topmost gene from each category of RDC in a round-robin fashion (e.g., g_5 from the class 1 category, followed by g_{10} from class 2, then g_7 from class 3, etc.) as shown in Figure 6.3 (b). If I suppose that r = 5, then the two top ranked genes (i.e., g_5 and g_{10}) are added to the selected minimum subset of

genes (three genes). The final ranking and the final selection are shown in Figure 6.3 (c).

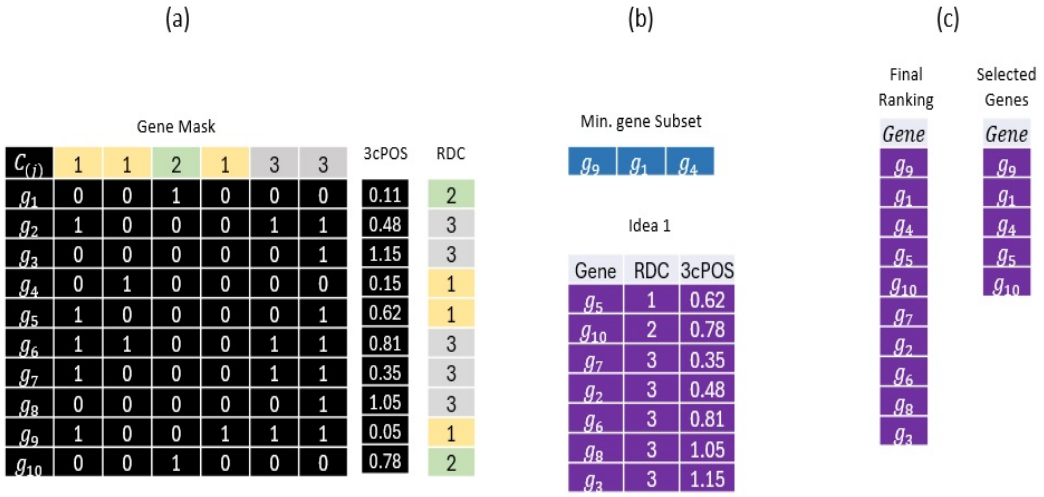

Figure 6.3: An example of the Idea 1: (a) genes with their masks, 3-class proportional overlapping scores, and relative dominant classes; (b) minimum gene subset obtained by Algorithm 2, and gene list ranked by 3cPOS and RDC; (c) final ranking, and selected genes at the end of the process.

The following example illustrates the process of obtaining the final gene subset as outlined in Idea 2, as presented in Figure 6.4. Each gene is associated with its corresponding gene mask and its three-class proportional overlapping score (3cPOS), as depicted in Figure 6.4 (a). To determine the minimum subset of genes, we employ Algorithm 2. This results in the inclusion of only three genes—g_9, g_1, and g_4—in the minimum subset. Figure 6.4 (b) displays this selected minimum subset and the remaining genes are then sorted in ascending order based on their 3cPOS values. Assuming a parameter $r = 5$, the two highest-ranked genes (g_7 and g_2) are subsequently added to the initially selected minimum subset. The final ranking and selection process are illustrated in Figure 6.4 (c).

6.3 Results

To validate the performance of idea 1 and idea 2, one can assess the accuracy of a classifier applied subsequent to the feature selection process. Consequently, the classification is based exclusively on the selected gene expressions. This method can examine the effectiveness of the techniques in identifying discriminative genes. [29] have implemented seven different feature

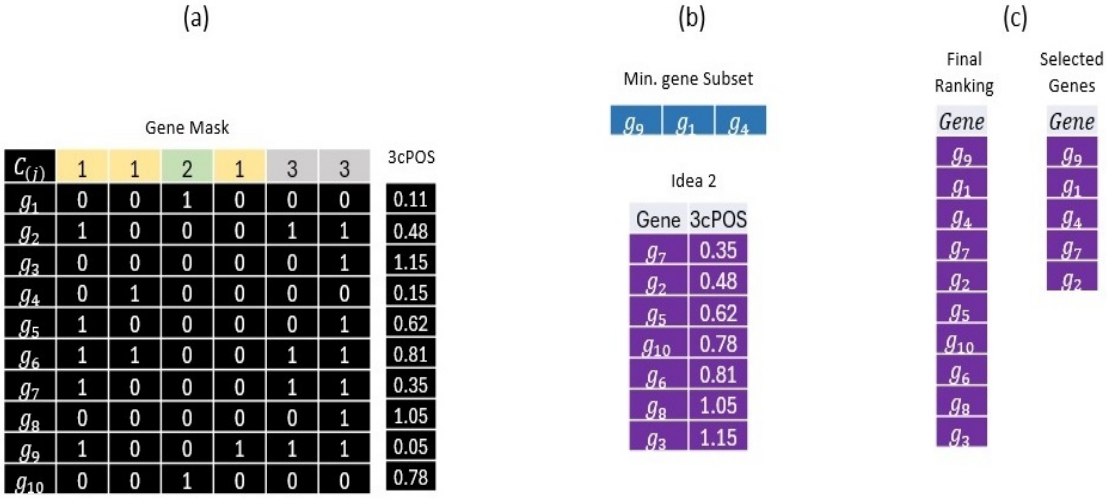

Figure 6.4: An example of the Idea 2: (a) genes with their masks and 3-class proportional overlapping scores; (b) minimum gene subset obtained by Algorithm 2, and gene list ranked by 3cPOS; (c) final ranking, and selected genes at the end of the process.

selection methods and shown that the gene selection technique plays an important role in a classifier's accuracy. This strategy has been employed in numerous studies, including [143] and [192].

Twenty repetitions of 5-fold cross-validation analysis were performed for each combination of the datasets, gene selection techniques, selected gene sizes (20 different gene set sizes) and classifiers. The R package randomForest [114] is used to implement the Random Forest with its default values for ntree, mtry, and nodesize: 500; the square root of the number of predictors; and 1. The R package class [156] is utilized to implement the K-Nearest Neighbors classifiers using a default parameter: k, the closest odd number of neighbors. The R package e1071 [48] performs a Support Vector Machine Classifier with different types of kernels. For simplicity, we used the linear kernel for SVM. The R package xgboost [33] is used to conduct Extreme Gradient Boosting Classification.

In this chapter, fourteen gene expression datasets are utilized to assess the quality performance of Idea 1 and Idea 2 alongside 3cPOS method. For each fold, subsets of genes of size r (where$r = 1, 2, \ldots, 20$) are selected using Idea 1, Idea 2, and 3cPOS method to construct classification models with four algorithms: Random Forest (RF), k-Nearest Neighbors (kNN), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost). Specifically, the averaged classification accuracies are computed across 100 runs.

The evaluation is carried out according to the following procedure:

1. Each dataset is randomly divided into training and testing subsets using 5-fold cross-validation, with 80% of the data put aside for training and 20% for testing. This step is iterated 20 times, which results in a total of 100 runs.

2. Idea 1, Idea 2, as well as the 3cPOS method are applied to the training data. This step aims to rank and select the top 20 informative genes from the entire set of informative genes.

3. The top r selected informative genes for each $r = 1, 2, ..., 20$ obtained from the ranked gene set across Idea 1, Idea 2, and 3cPOS approach are employed to fit the classification algorithms such as Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on the training data.

4. The fitted classification models predict class probabilities on the testing data by training on the 20 different gene sets corresponding to sizes, $r = 1, 2, ..., 20$..

5. Lastly, the average classification accuracy is computed by comparing the testing data's true class labels to the predictions across 100 runs.

Table 6.1 shows the average classification accuracy of the 3cPOS method, Idea 1, as well as Idea 2 using the RF, kNN, SVM, and XGBoost classifiers on the GSE23938 dataset. Idea 1 and Idea 2 perform comparable performance to the 3cPOS method using the RF and kNN classifiers. In addition, Idea 1 and Idea 2 outperform the 3cPOS method across different set sizes of informative genes using the XGBoost classifier. In contrast, the 3cPOS method demonstrates superior performance using the SVM classifier.

The average classification accuracy of the 3cPOS method, Idea 1, and Idea 2 using the RF, kNN, SVM, and XGBoost classifiers on the GSE220938 dataset is presented in Table 6.2. The results reveal that Idea 1 and Idea 2 show comparable performance to the 3cPOS method across the RF, SVM, and XGBoost classifiers. Moreover, by using the kNN classifier, Idea 1 and Idea 2 outperform the 3cPOS method.

Table 6.1: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE23938**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.83	0.68	0.68	0.87	0.71	0.71	0.86	0.75	0.75	0.66	0.68	0.68
2	0.85	0.78	0.78	0.87	0.80	0.80	0.88	0.81	0.81	0.66	0.71	0.71
3	0.86	0.81	0.81	0.87	0.83	0.83	0.89	0.83	0.83	0.67	0.73	0.72
4	0.87	0.84	0.85	0.87	0.84	0.85	0.90	0.83	0.85	0.69	0.74	0.74
5	0.87	0.85	0.86	0.86	0.85	0.85	0.90	0.85	0.86	0.68	0.73	0.74
6	0.86	0.86	0.86	0.86	0.85	0.85	0.91	0.85	0.87	0.67	0.74	0.75
7	0.88	0.86	0.86	0.86	0.85	0.86	0.91	0.85	0.87	0.66	0.75	0.74
8	0.87	0.86	0.86	0.85	0.87	0.86	0.92	0.86	0.88	0.67	0.75	0.75
9	0.87	0.87	0.87	0.86	0.87	0.86	0.91	0.86	0.89	0.66	0.76	0.75
10	0.87	0.87	0.86	0.86	0.86	0.86	0.92	0.86	0.88	0.65	0.75	0.74
11	0.87	0.87	0.85	0.86	0.85	0.86	0.91	0.86	0.87	0.65	0.75	0.74
12	0.86	0.87	0.86	0.86	0.85	0.87	0.91	0.86	0.88	0.65	0.75	0.74
13	0.87	0.87	0.86	0.86	0.86	0.86	0.91	0.87	0.88	0.67	0.75	0.74
14	0.86	0.87	0.86	0.86	0.87	0.87	0.91	0.87	0.88	0.67	0.76	0.74
15	0.86	0.87	0.86	0.86	0.86	0.87	0.90	0.87	0.88	0.67	0.76	0.74
16	0.86	0.87	0.86	0.86	0.86	0.87	0.90	0.87	0.88	0.66	0.76	0.74
17	0.86	0.87	0.86	0.86	0.87	0.87	0.90	0.87	0.88	0.66	0.76	0.74
18	0.86	0.86	0.85	0.86	0.87	0.86	0.90	0.87	0.88	0.66	0.76	0.74
19	0.85	0.87	0.86	0.86	0.86	0.87	0.90	0.87	0.88	0.66	0.76	0.74
20	0.85	0.86	0.86	0.86	0.86	0.87	0.90	0.87	0.88	0.66	0.76	0.74

Table 6.2: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE22093**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.55	0.56	0.56	0.59	0.60	0.60	0.67	0.66	0.66	0.58	0.56	0.56
2	0.60	0.62	0.63	0.62	0.63	0.63	0.67	0.65	0.65	0.58	0.58	0.58
3	0.62	0.63	0.63	0.61	0.63	0.63	0.66	0.63	0.63	0.58	0.58	0.58
4	0.62	0.64	0.64	0.62	0.64	0.64	0.65	0.64	0.64	0.59	0.58	0.58
5	0.64	0.64	0.64	0.64	0.64	0.64	0.65	0.65	0.65	0.58	0.57	0.57
6	0.65	0.64	0.63	0.64	0.65	0.65	0.64	0.63	0.63	0.58	0.58	0.58
7	0.64	0.64	0.64	0.64	0.66	0.66	0.64	0.63	0.63	0.58	0.57	0.57
8	0.64	0.64	0.64	0.65	0.66	0.66	0.63	0.63	0.63	0.58	0.57	0.57
9	0.65	0.64	0.64	0.63	0.65	0.65	0.63	0.63	0.63	0.58	0.57	0.57
10	0.66	0.65	0.65	0.63	0.66	0.66	0.63	0.61	0.62	0.58	0.57	0.57
11	0.66	0.66	0.65	0.63	0.67	0.67	0.62	0.62	0.62	0.59	0.58	0.57
12	0.67	0.66	0.65	0.63	0.67	0.66	0.62	0.62	0.62	0.58	0.58	0.57
13	0.67	0.66	0.66	0.62	0.67	0.66	0.62	0.62	0.62	0.58	0.58	0.57
14	0.67	0.66	0.65	0.62	0.68	0.67	0.60	0.63	0.62	0.58	0.58	0.57
15	0.67	0.66	0.65	0.62	0.67	0.66	0.60	0.64	0.61	0.58	0.58	0.58
16	0.68	0.66	0.65	0.63	0.68	0.67	0.60	0.64	0.61	0.58	0.58	0.57
17	0.68	0.66	0.65	0.63	0.68	0.67	0.60	0.63	0.61	0.58	0.58	0.57
18	0.67	0.67	0.65	0.62	0.67	0.66	0.61	0.63	0.61	0.57	0.59	0.58
19	0.67	0.66	0.65	0.62	0.68	0.66	0.60	0.63	0.62	0.58	0.58	0.58
20	0.68	0.66	0.66	0.62	0.68	0.67	0.59	0.62	0.62	0.58	0.58	0.57

The average classification accuracy of Idea 1, Idea 2, and the 3cPOS technique using the RF, kNN, SVM, and XGBoost classifiers on the GSE21029 dataset is shown in Table 6.3. By using RF and SVM classifiers, Idea 1 and Idea 2 outperform the 3cPOS method at small and moderate sets of informative genes. In addition, Idea 1 and Idea 2 remain comparable with the 3cPOS method using the kNN classifier. In contrast, the 3cPOS method outperforms both Idea 1 and Idea 2 using the XGBoost classifier.

The average classification accuracy of Idea 1, Idea 2, and the 3cPOS technique using the RF, kNN, SVM, and XGBoost classifiers on the GSE102287 dataset is shown in Table 6.4. The results indicate that Idea 1 and Idea 2 perform better than the 3cPOS method using the RF classifier, while achieving performance comparable to that of the 3cPOS method using the kNN classifier. Additionally, the 3cPOS method outperforms both Idea 1 and Idea 2 across different set sizes of informative genes across the SVM and XGBoost classifiers.

The average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, utilising the RF, kNN, SVM, and XGBoost classifiers on the GSE102279 dataset is displayed in Table 6.5. Idea 1 and Idea 2 demonstrate comparable performance to that of the 3cPOS method across the RF, kNN, and SVM classifiers. In contrast, the 3cPOS method outperforms both Idea 1 and Idea 2 across different set sizes of informative genes using the XGBoost classifier.

Table 6.6 illustrates the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2 using the RF, kNN, SVM, and XGBoost classifiers on the GSE17951 dataset. For the RF classifier, Idea 1 and Idea 2 exhibit comparable performance to that of the 3cPOS method. Using the kNN and SVM classifiers, Idea 1 and Idea 2 outperform the 3cPOS method for small and moderate set sizes of informative genes but show comparable performance to the 3cPOS method for larger set sizes of informative genes. In addition, Idea 1 and Idea 2 remain comparable to the 3cPOS method using the XGBoost classifier.

Table 6.3: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE21029**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.72	0.69	0.69	0.79	0.69	0.77	0.79	0.77	0.77	0.74	0.70	0.69
2	0.78	0.81	0.81	0.81	0.81	0.82	0.81	0.81	0.81	0.76	0.76	0.75
3	0.80	0.83	0.83	0.82	0.83	0.83	0.82	0.84	0.84	0.77	0.77	0.75
4	0.80	0.85	0.84	0.83	0.85	0.83	0.83	0.85	0.84	0.78	0.75	0.73
5	0.81	0.86	0.85	0.84	0.86	0.84	0.83	0.87	0.85	0.77	0.77	0.74
6	0.81	0.87	0.86	0.84	0.87	0.85	0.83	0.87	0.86	0.77	0.79	0.76
7	0.81	0.88	0.87	0.85	0.88	0.87	0.84	0.88	0.87	0.78	0.78	0.77
8	0.82	0.88	0.86	0.86	0.88	0.87	0.85	0.88	0.87	0.79	0.78	0.77
9	0.83	0.89	0.86	0.87	0.89	0.88	0.86	0.89	0.88	0.80	0.79	0.77
10	0.84	0.89	0.87	0.87	0.89	0.88	0.87	0.89	0.88	0.81	0.80	0.76
11	0.90	0.89	0.88	0.90	0.89	0.88	0.90	0.89	0.89	0.84	0.80	0.77
12	0.90	0.90	0.88	0.91	0.90	0.88	0.91	0.90	0.89	0.85	0.80	0.77
13	0.90	0.90	0.88	0.91	0.90	0.88	0.91	0.89	0.89	0.84	0.80	0.77
14	0.90	0.90	0.89	0.92	0.90	0.89	0.91	0.90	0.90	0.85	0.80	0.77
15	0.90	0.91	0.90	0.92	0.91	0.90	0.91	0.90	0.90	0.85	0.81	0.78
16	0.90	0.91	0.90	0.92	0.91	0.90	0.90	0.90	0.91	0.84	0.81	0.78
17	0.90	0.91	0.90	0.92	0.91	0.91	0.90	0.89	0.90	0.83	0.81	0.78
18	0.91	0.91	0.90	0.92	0.91	0.91	0.91	0.89	0.90	0.83	0.81	0.78
19	0.91	0.91	0.90	0.92	0.91	0.91	0.91	0.90	0.91	0.83	0.81	0.78
20	0.91	0.91	0.90	0.93	0.91	0.91	0.91	0.90	0.91	0.83	0.80	0.77

Table 6.4: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE102287**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.39	0.38	0.38	0.46	0.40	0.40	0.54	0.52	0.52	0.45	0.39	0.38
2	0.41	0.40	0.40	0.45	0.39	0.39	0.54	0.48	0.48	0.42	0.40	0.39
3	0.41	0.41	0.42	0.45	0.39	0.39	0.54	0.47	0.47	0.41	0.39	0.38
4	0.40	0.41	0.42	0.44	0.40	0.40	0.53	0.47	0.46	0.42	0.40	0.40
5	0.40	0.42	0.43	0.43	0.41	0.41	0.54	0.45	0.45	0.41	0.39	0.39
6	0.40	0.43	0.43	0.43	0.42	0.42	0.54	0.45	0.45	0.42	0.39	0.38
7	0.41	0.42	0.42	0.43	0.42	0.42	0.54	0.45	0.44	0.43	0.38	0.38
8	0.40	0.44	0.43	0.43	0.42	0.42	0.54	0.45	0.44	0.43	0.38	0.38
9	0.40	0.43	0.44	0.42	0.43	0.42	0.54	0.45	0.44	0.42	0.39	0.38
10	0.40	0.43	0.43	0.41	0.42	0.42	0.54	0.44	0.43	0.42	0.39	0.38
11	0.40	0.43	0.43	0.41	0.42	0.42	0.54	0.45	0.43	0.43	0.39	0.38
12	0.41	0.44	0.43	0.40	0.42	0.42	0.54	0.44	0.44	0.43	0.38	0.38
13	0.41	0.42	0.44	0.40	0.42	0.42	0.54	0.44	0.44	0.45	0.37	0.37
14	0.40	0.43	0.43	0.40	0.42	0.42	0.54	0.44	0.44	0.43	0.37	0.37
15	0.41	0.43	0.43	0.40	0.42	0.41	0.53	0.44	0.44	0.43	0.37	0.37
16	0.40	0.43	0.43	0.40	0.42	0.41	0.53	0.44	0.43	0.43	0.37	0.37
17	0.40	0.43	0.43	0.42	0.42	0.41	0.54	0.44	0.43	0.43	0.37	0.37
18	0.41	0.43	0.43	0.41	0.42	0.41	0.53	0.44	0.43	0.44	0.37	0.38
19	0.41	0.43	0.43	0.41	0.43	0.41	0.53	0.45	0.43	0.44	0.38	0.38
20	0.41	0.43	0.43	0.42	0.43	0.41	0.53	0.44	0.43	0.43	0.38	0.38

Table 6.5: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE102279**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.83	0.78	0.78	0.89	0.85	0.85	0.89	0.84	0.84	0.85	0.78	0.78
2	0.87	0.87	0.87	0.90	0.89	0.89	0.90	0.89	0.89	0.87	0.83	0.83
3	0.89	0.89	0.89	0.91	0.90	0.90	0.91	0.91	0.90	0.88	0.86	0.86
4	0.89	0.90	0.89	0.91	0.91	0.91	0.91	0.91	0.91	0.89	0.86	0.86
5	0.90	0.90	0.89	0.91	0.91	0.91	0.91	0.91	0.91	0.89	0.86	0.86
6	0.91	0.90	0.90	0.91	0.91	0.91	0.92	0.92	0.92	0.90	0.86	0.86
7	0.91	0.90	0.90	0.91	0.91	0.91	0.92	0.92	0.92	0.90	0.86	0.86
8	0.92	0.90	0.90	0.91	0.92	0.92	0.92	0.92	0.92	0.90	0.86	0.86
9	0.92	0.91	0.90	0.91	0.92	0.92	0.92	0.92	0.92	0.90	0.86	0.86
10	0.92	0.91	0.91	0.92	0.92	0.92	0.92	0.92	0.92	0.90	0.86	0.86
11	0.92	0.91	0.91	0.92	0.92	0.92	0.93	0.92	0.92	0.90	0.86	0.87
12	0.92	0.90	0.91	0.92	0.92	0.92	0.93	0.92	0.92	0.90	0.86	0.87
13	0.92	0.90	0.91	0.92	0.92	0.92	0.93	0.91	0.92	0.91	0.86	0.87
14	0.92	0.91	0.91	0.93	0.92	0.92	0.94	0.92	0.92	0.91	0.87	0.86
15	0.93	0.91	0.91	0.93	0.92	0.92	0.94	0.92	0.92	0.91	0.87	0.86
16	0.93	0.91	0.91	0.93	0.91	0.92	0.94	0.92	0.92	0.91	0.87	0.87
17	0.93	0.91	0.91	0.93	0.91	0.92	0.94	0.92	0.93	0.92	0.87	0.87
18	0.93	0.91	0.92	0.93	0.91	0.92	0.94	0.91	0.93	0.92	0.87	0.87
19	0.93	0.91	0.91	0.93	0.92	0.92	0.94	0.92	0.93	0.92	0.87	0.87
20	0.93	0.91	0.91	0.93	0.91	0.92	0.94	0.92	0.93	0.92	0.87	0.87

Table 6.6: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE17951**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.80	0.84	0.84	0.86	0.85	0.85	0.86	0.86	0.86	0.82	0.84	0.84
2	0.86	0.90	0.90	0.88	0.92	0.92	0.89	0.91	0.91	0.86	0.87	0.87
3	0.87	0.92	0.92	0.90	0.94	0.94	0.91	0.94	0.94	0.88	0.89	0.89
4	0.89	0.93	0.93	0.91	0.95	0.95	0.92	0.95	0.95	0.89	0.89	0.89
5	0.90	0.93	0.93	0.91	0.95	0.95	0.92	0.96	0.96	0.91	0.90	0.91
6	0.91	0.94	0.94	0.91	0.96	0.96	0.93	0.97	0.97	0.92	0.91	0.92
7	0.92	0.94	0.94	0.92	0.97	0.96	0.93	0.97	0.97	0.93	0.91	0.94
8	0.92	0.95	0.95	0.92	0.97	0.96	0.94	0.98	0.97	0.93	0.92	0.94
9	0.93	0.95	0.95	0.93	0.97	0.96	0.94	0.98	0.98	0.93	0.92	0.95
10	0.93	0.95	0.95	0.93	0.97	0.96	0.95	0.98	0.98	0.94	0.92	0.95
11	0.94	0.96	0.95	0.94	0.98	0.96	0.95	0.99	0.97	0.94	0.94	0.95
12	0.94	0.96	0.95	0.94	0.98	0.96	0.96	0.99	0.98	0.94	0.94	0.94
13	0.95	0.96	0.96	0.95	0.98	0.96	0.96	0.99	0.98	0.95	0.94	0.95
14	0.95	0.97	0.96	0.96	0.98	0.96	0.97	0.99	0.98	0.95	0.95	0.95
15	0.95	0.97	0.96	0.96	0.98	0.97	0.97	0.99	0.98	0.96	0.95	0.95
16	0.95	0.97	0.96	0.96	0.98	0.97	0.98	0.99	0.98	0.96	0.95	0.95
17	0.96	0.97	0.96	0.97	0.98	0.97	0.98	0.99	0.98	0.96	0.95	0.95
18	0.96	0.97	0.96	0.97	0.98	0.97	0.98	0.99	0.98	0.97	0.95	0.95
19	0.96	0.97	0.96	0.98	0.98	0.98	0.98	0.99	0.99	0.97	0.95	0.95
20	0.96	0.97	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.97	0.95	0.95

Table 6.7 presents the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, using the RF, kNN, SVM, and XGBoost classifiers on the MLL dataset. Idea 1 and Idea 2 outperform the 3cPOS method at moderate set sizes of informative genes when using the RF classifier. Idea 1 and Idea 2 achieve superior performance at different set sizes of informative genes across the kNN and SVM classifiers, except for a single gene. However, the 3cPOS method achieves the highest classification accuracy for smaller set sizes of informative genes, but shows performance comparable to both Idea 1 and Idea 2 using the XGBoost classifier.

Table 6.8 presents the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, using the RF, kNN, SVM, and XGBoost classifiers on the GSE40595(1) dataset. For the RF classifier, Idea 1 and Idea 2 outperform the 3cPOS method for smaller set sizes of informative genes but show comparable performance at moderate and large set sizes of informative genes. By using the kNN and SVM classifiers, Idea 1 and the 3cPOS method outperform Idea 2. In contrast, Idea 1 and Idea 2 show comparable performance to the 3cPOS method when using the XGBoost classifier.

The average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, using the RF, kNN, SVM, and XGBoost classifiers on the GSE27854(1) dataset is presented in Table 6.9. Idea 1 and Idea 2 demonstrate comparable performance to the 3cPOS method across different set sizes of informative genes when using the RF classifier. In contrast, Idea 1 and Idea 2 outperform the 3cPOS method for smaller and larger set sizes of informative genes when evaluated with the kNN classifier. For the SVM classifier, Idea 1 and Idea 2 outperform the 3cPOS method for smaller set sizes of informative genes, but the 3cPOS method shows superior performance at large set sizes of informative genes. In addition, Idea 1 and Idea 2 remain comparable in performance to the 3cPOS method across different set sizes of informative genes using the XGBoost classifier.

Table 6.7: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**MLL**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.66	0.61	0.61	0.66	0.61	0.64	0.67	0.64	0.64	0.67	0.60	0.61
2	0.77	0.77	0.76	0.71	0.77	0.78	0.73	0.78	0.78	0.75	0.72	0.73
3	0.81	0.81	0.82	0.77	0.81	0.80	0.78	0.82	0.82	0.79	0.75	0.76
4	0.82	0.84	0.84	0.79	0.82	0.82	0.77	0.84	0.84	0.79	0.77	0.78
5	0.84	0.86	0.86	0.80	0.85	0.83	0.79	0.86	0.86	0.79	0.80	0.82
6	0.86	0.90	0.90	0.81	0.87	0.85	0.80	0.88	0.88	0.81	0.86	0.88
7	0.87	0.91	0.90	0.81	0.88	0.85	0.82	0.88	0.88	0.83	0.87	0.89
8	0.87	0.92	0.90	0.82	0.89	0.85	0.83	0.89	0.88	0.84	0.88	0.89
9	0.88	0.91	0.90	0.84	0.89	0.86	0.86	0.89	0.89	0.85	0.87	0.89
10	0.88	0.92	0.90	0.84	0.90	0.86	0.87	0.90	0.89	0.86	0.87	0.89
11	0.88	0.92	0.90	0.84	0.90	0.87	0.87	0.89	0.89	0.86	0.87	0.89
12	0.89	0.92	0.91	0.85	0.90	0.87	0.87	0.89	0.89	0.86	0.87	0.89
13	0.89	0.92	0.90	0.85	0.91	0.87	0.87	0.90	0.89	0.86	0.87	0.89
14	0.89	0.92	0.91	0.85	0.91	0.87	0.88	0.90	0.89	0.88	0.88	0.88
15	0.90	0.92	0.90	0.85	0.91	0.87	0.88	0.90	0.89	0.88	0.87	0.88
16	0.90	0.92	0.90	0.86	0.91	0.87	0.89	0.90	0.90	0.88	0.87	0.88
17	0.90	0.92	0.91	0.86	0.91	0.88	0.89	0.91	0.90	0.88	0.88	0.89
18	0.90	0.92	0.91	0.86	0.90	0.87	0.89	0.91	0.90	0.89	0.88	0.89
19	0.90	0.92	0.91	0.87	0.90	0.87	0.90	0.91	0.90	0.89	0.88	0.89
20	0.91	0.92	0.91	0.87	0.91	0.87	0.90	0.90	0.90	0.89	0.88	0.89

Table 6.8: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE40595(1)**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.80	0.83	0.83	0.80	0.83	0.80	0.80	0.83	0.81	0.80	0.83	0.83
2	0.85	0.87	0.87	0.86	0.87	0.84	0.87	0.88	0.85	0.84	0.85	0.85
3	0.86	0.85	0.86	0.89	0.89	0.86	0.89	0.89	0.87	0.84	0.85	0.85
4	0.87	0.87	0.87	0.90	0.91	0.87	0.91	0.91	0.88	0.85	0.85	0.85
5	0.88	0.88	0.88	0.91	0.92	0.88	0.92	0.93	0.89	0.86	0.86	0.86
6	0.89	0.89	0.89	0.91	0.93	0.88	0.93	0.94	0.90	0.87	0.87	0.86
7	0.90	0.90	0.90	0.92	0.93	0.89	0.94	0.95	0.90	0.87	0.87	0.87
8	0.90	0.91	0.90	0.93	0.93	0.89	0.94	0.94	0.91	0.87	0.87	0.87
9	0.90	0.92	0.91	0.93	0.93	0.89	0.95	0.95	0.91	0.88	0.87	0.88
10	0.92	0.92	0.92	0.94	0.94	0.91	0.95	0.96	0.91	0.88	0.88	0.88
11	0.92	0.93	0.92	0.94	0.94	0.91	0.95	0.96	0.91	0.88	0.87	0.88
12	0.92	0.93	0.92	0.94	0.95	0.90	0.95	0.96	0.91	0.88	0.88	0.88
13	0.93	0.93	0.93	0.95	0.95	0.91	0.95	0.96	0.91	0.88	0.88	0.88
14	0.93	0.94	0.93	0.94	0.95	0.91	0.95	0.96	0.91	0.88	0.88	0.88
15	0.93	0.94	0.93	0.95	0.95	0.91	0.96	0.97	0.91	0.88	0.88	0.88
16	0.94	0.94	0.94	0.95	0.95	0.91	0.96	0.96	0.92	0.88	0.88	0.88
17	0.94	0.95	0.94	0.95	0.95	0.91	0.96	0.97	0.92	0.88	0.88	0.88
18	0.94	0.95	0.94	0.95	0.95	0.91	0.96	0.97	0.92	0.89	0.88	0.88
19	0.93	0.96	0.94	0.95	0.95	0.91	0.96	0.97	0.92	0.89	0.88	0.89
20	0.94	0.95	0.94	0.95	0.95	0.91	0.97	0.97	0.92	0.89	0.88	0.89

Table 6.9: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE27854(1)**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.49	0.50	0.50	0.50	0.57	0.58	0.58	0.65	0.65	0.49	0.50	0.50
2	0.54	0.52	0.52	0.52	0.55	0.55	0.55	0.64	0.65	0.43	0.43	0.43
3	0.55	0.52	0.53	0.53	0.55	0.56	0.56	0.64	0.64	0.41	0.43	0.43
4	0.55	0.54	0.54	0.54	0.54	0.54	0.54	0.62	0.63	0.40	0.43	0.43
5	0.56	0.55	0.55	0.55	0.55	0.55	0.55	0.61	0.62	0.41	0.42	0.42
6	0.56	0.56	0.56	0.56	0.55	0.55	0.55	0.61	0.61	0.42	0.42	0.42
7	0.57	0.56	0.57	0.57	0.56	0.57	0.57	0.60	0.60	0.42	0.43	0.43
8	0.57	0.56	0.56	0.56	0.57	0.57	0.57	0.59	0.58	0.41	0.44	0.44
9	0.57	0.56	0.56	0.56	0.58	0.58	0.58	0.58	0.57	0.41	0.44	0.44
10	0.58	0.57	0.57	0.57	0.58	0.58	0.58	0.56	0.56	0.41	0.46	0.46
11	0.58	0.57	0.57	0.57	0.59	0.59	0.59	0.56	0.56	0.42	0.45	0.45
12	0.58	0.57	0.57	0.57	0.59	0.60	0.60	0.55	0.55	0.41	0.44	0.45
13	0.57	0.57	0.57	0.57	0.60	0.61	0.61	0.55	0.55	0.42	0.44	0.45
14	0.58	0.57	0.57	0.57	0.61	0.61	0.61	0.56	0.54	0.42	0.45	0.45
15	0.58	0.57	0.57	0.57	0.61	0.62	0.62	0.55	0.55	0.42	0.45	0.46
16	0.57	0.58	0.57	0.57	0.61	0.62	0.62	0.56	0.55	0.42	0.44	0.45
17	0.58	0.58	0.58	0.58	0.61	0.62	0.62	0.55	0.55	0.42	0.44	0.45
18	0.58	0.59	0.58	0.58	0.61	0.62	0.62	0.55	0.55	0.42	0.44	0.45
19	0.58	0.58	0.58	0.58	0.61	0.62	0.62	0.55	0.56	0.42	0.45	0.43
20	0.58	0.59	0.58	0.58	0.61	0.63	0.63	0.55	0.55	0.44	0.45	0.44

Table 6.10 demonstrates the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, using the RF, kNN, SVM, and XGBoost classifiers on the GSE162228(1) dataset. Idea 1 and Idea 2 achieve superior performance when using a single informative gene using the RF classifier. In addition, Idea 1 and Idea 2 achieve the highest classification accuracy with a single informative gene and a set of two informative genes when using XGBoost classifier. In contrast, the 3cPOS method performs better than Idea 1 and Idea 2 across different sets of informative genes across the kNN and SVM classifiers.

Table 6.11 presents the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, obtained using the RF, kNN, SVM, and XGBoost classifiers on the GSE26712 dataset. The 3cPOS method outperforms Idea 1 and Idea 2 for small set sizes of informative genes but demonstrates comparable performance at moderate and large sets when using the RF classifier. For the kNN classifier, the 3cPOS method achieves the highest classification accuracy at both small and moderate sets of informative genes. When evaluated with the SVM classifier, all three methods—3cPOS, Idea 1, and Idea 2 show identical performance at the small set of informative genes, with a classification accuracy of 70%. Furthermore, Idea 1 and Idea 2 exhibit comparable performance to the 3cPOS method when using the XGBoost classifier.

Table 6.12 presents the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, obtained using the RF, kNN, SVM, and XGBoost classifiers on the GSE30219 dataset. Idea 1 and Idea 2 outperform the 3cPOS method in various set sizes of informative genes when evaluated with the RF, kNN, and XGBoost classifiers. However, the 3cPOS method achieves the highest accuracy for large sets of informative genes when using the SVM classifier.

Table 6.13 presents the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, yielded using the RF, kNN, SVM, and XGBoost classifiers on the GSE13911 dataset. Idea 1 and Idea 2 achieve the highest performance at the different set sizes of informative genes across RF, kNN, SVM, and XGBoost classifiers

Table 6.14 presents the average classification accuracy of the 3cPOS method, Idea 1, and Idea 2, using the RF, kNN, SVM, and XGBoost classifiers on the GSE2990 dataset. Idea 1 and Idea 2 outperform the 3cPOS method in different set sizes of informative genes across the RF and SVM classifiers. Idea 1 and Idea 2 also show comparable performance to that of the 3cPOS

method using the kNN classifier. In addition, Idea 1 and Idea 2 outperform the 3cPOS method at larger set sizes of informative genes when using the XGBoost classifier.

Table 6.10: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE162228(1)**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.43	0.48	0.48	0.58	0.54	0.54	0.63	0.61	0.61	0.42	0.48	0.48
2	0.52	0.52	0.51	0.58	0.55	0.55	0.63	0.60	0.60	0.50	0.52	0.52
3	0.55	0.54	0.54	0.58	0.53	0.53	0.63	0.59	0.59	0.53	0.53	0.53
4	0.56	0.54	0.54	0.57	0.53	0.53	0.63	0.59	0.59	0.54	0.54	0.54
5	0.57	0.56	0.56	0.59	0.53	0.53	0.64	0.59	0.59	0.55	0.54	0.54
6	0.58	0.55	0.56	0.59	0.54	0.54	0.64	0.58	0.58	0.55	0.55	0.55
7	0.59	0.56	0.56	0.60	0.55	0.55	0.64	0.58	0.58	0.56	0.54	0.54
8	0.59	0.56	0.55	0.61	0.56	0.56	0.64	0.57	0.57	0.56	0.55	0.55
9	0.59	0.56	0.56	0.61	0.56	0.56	0.63	0.57	0.57	0.56	0.55	0.55
10	0.59	0.56	0.56	0.61	0.56	0.56	0.63	0.56	0.56	0.56	0.55	0.55
11	0.59	0.56	0.56	0.61	0.56	0.56	0.63	0.55	0.55	0.56	0.55	0.55
12	0.59	0.56	0.57	0.61	0.56	0.56	0.63	0.54	0.54	0.56	0.56	0.56
13	0.60	0.56	0.56	0.61	0.57	0.57	0.62	0.53	0.53	0.55	0.55	0.55
14	0.59	0.57	0.57	0.61	0.56	0.57	0.63	0.54	0.55	0.55	0.54	0.55
15	0.59	0.56	0.56	0.60	0.56	0.57	0.62	0.54	0.55	0.55	0.55	0.54
16	0.59	0.57	0.56	0.60	0.57	0.56	0.62	0.55	0.54	0.55	0.55	0.55
17	0.59	0.57	0.57	0.60	0.57	0.56	0.62	0.55	0.54	0.55	0.56	0.55
18	0.59	0.57	0.57	0.60	0.57	0.57	0.62	0.55	0.55	0.55	0.55	0.55
19	0.59	0.58	0.58	0.60	0.57	0.57	0.62	0.54	0.55	0.55	0.55	0.55
20	0.59	0.59	0.58	0.60	0.57	0.58	0.61	0.55	0.55	0.54	0.55	0.55

Table 6.11: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE26712**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.51	0.54	0.54	0.7	0.67	0.67	0.70	0.70	0.70	0.51	0.54	0.54
2	0.63	0.59	0.59	0.69	0.65	0.65	0.70	0.70	0.70	0.59	0.59	0.59
3	0.64	0.62	0.62	0.69	0.65	0.65	0.70	0.70	0.70	0.60	0.61	0.61
4	0.65	0.64	0.64	0.69	0.66	0.66	0.70	0.70	0.70	0.62	0.62	0.62
5	0.65	0.65	0.65	0.69	0.65	0.65	0.70	0.70	0.70	0.62	0.63	0.63
6	0.65	0.65	0.65	0.69	0.64	0.64	0.70	0.69	0.69	0.62	0.64	0.64
7	0.66	0.66	0.66	0.69	0.64	0.64	0.70	0.69	0.69	0.62	0.64	0.64
8	0.66	0.66	0.66	0.69	0.65	0.65	0.70	0.68	0.68	0.62	0.63	0.63
9	0.66	0.66	0.66	0.69	0.66	0.66	0.70	0.68	0.68	0.63	0.63	0.63
10	0.66	0.66	0.66	0.69	0.66	0.66	0.70	0.67	0.67	0.64	0.63	0.63
11	0.66	0.66	0.66	0.69	0.67	0.67	0.70	0.67	0.67	0.64	0.63	0.63
12	0.66	0.66	0.66	0.69	0.67	0.67	0.70	0.67	0.67	0.64	0.64	0.64
13	0.66	0.66	0.66	0.69	0.67	0.67	0.70	0.66	0.66	0.64	0.63	0.63
14	0.66	0.66	0.66	0.69	0.67	0.67	0.70	0.65	0.65	0.64	0.64	0.64
15	0.67	0.66	0.66	0.69	0.67	0.67	0.69	0.64	0.64	0.63	0.63	0.63
16	0.66	0.66	0.66	0.69	0.68	0.68	0.69	0.64	0.64	0.63	0.63	0.63
17	0.67	0.66	0.66	0.69	0.68	0.68	0.69	0.63	0.63	0.63	0.64	0.64
18	0.67	0.66	0.66	0.69	0.68	0.68	0.69	0.63	0.63	0.63	0.64	0.64
19	0.67	0.66	0.66	0.69	0.67	0.68	0.69	0.63	0.63	0.62	0.64	0.65
20	0.67	0.66	0.67	0.69	0.67	0.68	0.69	0.63	0.63	0.63	0.64	0.64

Table 6.12: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE30219**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.69	0.70	0.70	0.80	0.79	0.79	0.80	0.79	0.79	0.69	0.70	0.70
2	0.76	0.75	0.74	0.80	0.78	0.78	0.81	0.80	0.80	0.73	0.72	0.72
3	0.77	0.76	0.76	0.80	0.78	0.78	0.81	0.80	0.80	0.74	0.74	0.74
4	0.78	0.77	0.77	0.81	0.79	0.79	0.81	0.81	0.81	0.75	0.75	0.75
5	0.78	0.78	0.78	0.81	0.79	0.79	0.81	0.81	0.81	0.75	0.76	0.76
6	0.78	0.79	0.78	0.81	0.79	0.79	0.81	0.81	0.81	0.75	0.77	0.77
7	0.78	0.78	0.78	0.81	0.79	0.79	0.81	0.81	0.81	0.75	0.77	0.77
8	0.78	0.78	0.78	0.81	0.78	0.78	0.81	0.80	0.80	0.75	0.76	0.76
9	0.78	0.78	0.79	0.81	0.79	0.79	0.81	0.80	0.80	0.75	0.77	0.77
10	0.79	0.78	0.78	0.81	0.79	0.79	0.81	0.80	0.80	0.76	0.76	0.76
11	0.79	0.78	0.79	0.80	0.79	0.79	0.81	0.80	0.80	0.76	0.76	0.76
12	0.79	0.79	0.79	0.81	0.79	0.79	0.81	0.79	0.79	0.76	0.76	0.76
13	0.79	0.79	0.79	0.81	0.79	0.79	0.82	0.79	0.79	0.76	0.77	0.77
14	0.79	0.79	0.79	0.81	0.79	0.79	0.82	0.79	0.79	0.76	0.76	0.76
15	0.79	0.79	0.79	0.80	0.79	0.79	0.82	0.79	0.79	0.76	0.77	0.77
16	0.80	0.79	0.8	0.81	0.79	0.80	0.82	0.79	0.79	0.76	0.77	0.76
17	0.80	0.79	0.79	0.80	0.79	0.80	0.82	0.79	0.79	0.76	0.77	0.77
18	0.80	0.80	0.79	0.81	0.79	0.80	0.82	0.79	0.79	0.76	0.77	0.77
19	0.80	0.80	0.79	0.81	0.79	0.79	0.81	0.79	0.79	0.76	0.77	0.76
20	0.80	0.80	0.80	0.81	0.79	0.79	0.81	0.79	0.79	0.76	0.77	0.77

Table 6.13: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE13911**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.39	0.62	0.62	0.39	0.65	0.65	0.43	0.64	0.64	0.38	0.63	0.63
2	0.47	0.70	0.70	0.44	0.73	0.73	0.46	0.73	0.73	0.47	0.68	0.68
3	0.50	0.73	0.73	0.47	0.75	0.75	0.47	0.75	0.75	0.45	0.69	0.69
4	0.53	0.76	0.76	0.51	0.77	0.77	0.48	0.77	0.77	0.47	0.70	0.70
5	0.55	0.76	0.76	0.51	0.79	0.79	0.49	0.78	0.78	0.48	0.70	0.70
6	0.57	0.76	0.77	0.51	0.79	0.79	0.50	0.78	0.78	0.47	0.70	0.70
7	0.60	0.76	0.75	0.54	0.79	0.79	0.53	0.78	0.78	0.48	0.70	0.70
8	0.59	0.76	0.76	0.56	0.79	0.79	0.54	0.78	0.78	0.50	0.69	0.69
9	0.62	0.75	0.76	0.57	0.79	0.79	0.57	0.78	0.78	0.52	0.70	0.69
10	0.65	0.76	0.76	0.60	0.78	0.79	0.59	0.78	0.78	0.53	0.69	0.69
11	0.66	0.75	0.75	0.63	0.78	0.79	0.6	0.78	0.78	0.55	0.70	0.69
12	0.69	0.76	0.75	0.67	0.79	0.79	0.62	0.78	0.79	0.58	0.69	0.69
13	0.71	0.76	0.75	0.69	0.79	0.79	0.66	0.78	0.78	0.62	0.69	0.69
14	0.72	0.76	0.75	0.69	0.79	0.79	0.67	0.78	0.78	0.62	0.69	0.69
15	0.73	0.76	0.75	0.70	0.79	0.79	0.68	0.78	0.78	0.62	0.69	0.69
16	0.74	0.77	0.75	0.70	0.79	0.79	0.69	0.79	0.78	0.61	0.69	0.68
17	0.74	0.76	0.75	0.71	0.79	0.79	0.71	0.78	0.78	0.62	0.69	0.69
18	0.74	0.76	0.76	0.70	0.79	0.80	0.71	0.78	0.78	0.63	0.69	0.69
19	0.75	0.77	0.76	0.71	0.80	0.79	0.72	0.79	0.78	0.64	0.69	0.69
20	0.75	0.77	0.76	0.71	0.80	0.80	0.74	0.79	0.78	0.65	0.69	0.69

Table 6.14: Average classification accuracy of 3cPOS method, Idea 1 and Idea 2 yielded by Random Forest, k Nearest Neighbors, Support Vector Machine and Extreme Gradient Boosting classifiers on '**GSE2990**' dataset over all the 20 repetitions of 5-fold cross validation

Gene	RF			KNN			SVM			XGBoost		
	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2	3cPOS	Idea 1	Idea 2
1	0.81	0.78	0.78	0.81	0.79	0.79	0.79	0.78	0.78	0.81	0.78	0.78
2	0.83	0.83	0.83	0.82	0.81	0.81	0.81	0.82	0.82	0.84	0.81	0.81
3	0.84	0.85	0.85	0.82	0.82	0.82	0.81	0.85	0.85	0.84	0.83	0.83
4	0.83	0.86	0.86	0.82	0.82	0.82	0.81	0.85	0.85	0.84	0.84	0.84
5	0.83	0.86	0.86	0.82	0.82	0.82	0.81	0.86	0.86	0.84	0.84	0.84
6	0.83	0.86	0.86	0.82	0.81	0.81	0.82	0.86	0.86	0.84	0.84	0.84
7	0.83	0.86	0.86	0.82	0.81	0.81	0.82	0.87	0.87	0.84	0.84	0.84
8	0.83	0.86	0.86	0.82	0.81	0.81	0.82	0.87	0.87	0.84	0.84	0.84
9	0.83	0.86	0.87	0.82	0.81	0.81	0.83	0.87	0.87	0.84	0.84	0.84
10	0.82	0.86	0.87	0.81	0.81	0.81	0.83	0.87	0.87	0.84	0.84	0.84
11	0.82	0.87	0.87	0.81	0.81	0.81	0.83	0.88	0.88	0.84	0.84	0.84
12	0.82	0.87	0.88	0.81	0.81	0.81	0.83	0.88	0.89	0.84	0.84	0.86
13	0.82	0.87	0.89	0.82	0.81	0.81	0.83	0.88	0.91	0.84	0.85	0.87
14	0.82	0.88	0.89	0.82	0.81	0.82	0.83	0.89	0.91	0.84	0.86	0.88
15	0.82	0.89	0.89	0.82	0.81	0.82	0.83	0.90	0.91	0.84	0.87	0.88
16	0.82	0.89	0.89	0.82	0.81	0.83	0.84	0.90	0.91	0.84	0.87	0.87
17	0.82	0.89	0.89	0.82	0.81	0.84	0.84	0.90	0.91	0.84	0.87	0.87
18	0.82	0.89	0.88	0.82	0.82	0.84	0.84	0.90	0.91	0.84	0.87	0.87
19	0.83	0.89	0.89	0.82	0.81	0.84	0.84	0.90	0.91	0.84	0.87	0.87
20	0.82	0.89	0.88	0.82	0.82	0.85	0.84	0.90	0.91	0.84	0.87	0.87

6.4 Summary

The procedure of integrating both a minimum subset of genes and gene ranking to obtain the final gene selection is focused in this chapter. Firstly, gene masks are assigned to each gene to assess their discriminative power. Next, the three-class proportional overlapping score (3cPOS), as presented in Chapter 5, along with the gene masks are exploited to determine the minimum subset of genes. The minimum subset of genes offers the smallest set of genes that accurately classifies the largest number of samples during the training phase. The Relative Dominant Class (RDC) is employed to associate each gene with the class it is most likely to differentiate, with the class showing the highest proportion designated as the Relative Dominant Class for gene i.

Two distinct approaches, Idea 1 and Idea 2, were proposed to obtain the final gene selection. For Idea 1, gene ranking takes into account genes that are not included in the minimum subset. A set of the remaining genes are categorised by their RDC and then sorted in ascending order based on their 3cPOS scores within each RDC category. Selecting the highest-ranked gene from each RDC category in a round-robin fashion offers gene ranking. Both the minimum subset of genes and the gene rankings from Idea 1 are incorporated into the final gene selection.

Unlike Idea 1, Idea 2 aims at exploiting 3cPOS scores alone to establish the gene ranks. The remaining genes are sorted in ascending order based on their 3cPOS scores to provide gene ranks. By incorporating both the minimum subset and the rankings obtained from Idea 2, the final ranking is generated to provide the final gene selection.

Two illustrative examples are presented that shows the usage of Idea 1 and Idea 2 to obtain the final gene set. Three genes are selected for the minimum subset of genes in both examples: g_9, g_1, and g_4. Since we select $r = 5$ as the final gene selection, the genes g_5 and g_{10} are included for Idea 1 and g_7 and another g_2 are included for Idea 2 as the gene ranking. Therefore, the final gene selection for Idea 1 is g_9, g_1, g_4, g_5 and g_{10} and the final gene selection for Idea 2 is g_9, g_1, g_4, g_7 and g_2. .

To evaluate the performance of the minimum subset of genes, 14 gene expression datasets are used to validate the performance of the final gene selection with 20 iterations of 5-fold cross-validation, resulting in 100 runs. For each run, genes are selected up to 20, $r = 1, 2, 3, ..., 20$,

based on 3cPOS, Idea 1 and Idea 2. These sets of informative genes are employed to build analyses with the Random Forest (RF), k-Nearest Neighbors (KNN), Support Vector Machine (SVM), and XGBoost classifiers, providing the average classification accuracy across 100 runs.

Based on the RF classifier, Idea 1 and Idea 2 outperform the 3cPOS method across seven datasets: GSE21029, GSE102287, GSE17951, Leukemia, GSE40595(1), GSE13911, and GSE2990. Moreover, Idea 1 and Idea 2 demonstrate comparable performance to 3cPOS across three datasets: GSE22093, GSE26712, and GSE30219. However, the 3cPOS method outperforms all other techniques across three datasets: GSE102079, GSE27854(1), and GSE162228(1). The 3cPOS method also performs better than Idea 1 and Idea 2 on the small set of informative features for the GSE23938 dataset.

For the KNN classifier, Idea 1 and Idea 2 outperform the 3cPOS method across six datasets: GSE22093, GSE17951, Leukemia, GSE40595(1), GSE27854(1), and GSE13911. In addition, Idea 1 and Idea 2 perform better than the 3cPOS method on moderate and large set sizes of informative features across three datasets: GSE23938, GSE102287, and GSE2990. Idea 1 and Idea 2 also outperform 3cPOS on small and moderate set sizes of informative features on the GSE21029 dataset. In contrast, the 3cPOS method outperforms all other techniques across three datasets: GSE162228(1), GSE26712, and GSE30219, and shows superior performance on small and large set sizes of informative features in the GSE102079 dataset.

When evaluated using the SVM classifier, Idea 1 and Idea 2 outperform the 3cPOS method across five datasets: GSE17951, Leukemia, GSE40595(1), GSE13911, and GSE2990. In addition, both Idea 1 and Idea 2 show superior performance over the 3cPOS method on small and moderate sets of informative features across two datasets: GSE21029 and GSE27854(1). Furthermore, Idea 1 and Idea 2 achieve better results than the 3cPOS method on large sets of informative features in the GSE22093 dataset. Conversely, the 3cPOS method outperforms all other techniques across four datasets: GSE23938, GSE102287, GSE162228(1), and GSE30219. Additionally, 3cPOS demonstrates superior performance on both small and large sets of informative features for the GSE102079 dataset, and outperforms all other techniques on moderate and large sets of informative features in the GSE26712 dataset.

For the XGBoost classifier, both Idea 1 and Idea 2 demonstrate superior performance com-

pared to the 3cPOS method across three datasets: GSE23938, GSE27854(1), and GSE13911. Moreover, Idea 1 and Idea 2 show comparable performance to 3cPOS across eight additional datasets: GSE22093, GSE17951, Leukemia, GSE40595(1), GSE162228(1), GSE26712, GSE30219, and GSE2990.

Overall, both Idea 1 and Idea 2 achieve comparable performance to the 3cPOS method across various data characteristics and multiple classifiers, including RF, KNN, SVM, and XGBoost classifiers. When considering the minimum subset of genes and gene ranking in conjunction with 3cPOS for final gene selection, it is observed that their inclusion have not yield significant improvements compared to utilising the 3cPOS method alone. Consequently, we have decided not to incorporate the minimum subset of genes and gene ranking into the mPOS method.

CHAPTER 7

Multiple Proportional Overlapping Scores

7.1 Introduction

In gene expression data, machine learning techniques are employed find important genes for cancer classification, reveal relationships among genes, and classify cancer. As a result, it was shown that all of the training samples are classified correctly, corresponding to high classification accuracy in test samples [88]. A classification model built using a small set of the most informative genes helps improving classification accuracy, faster training times as well as easier interpretability [124]. Several studies have utlised similar approaches to extract the informative genes from gene expression data [123, 147, 35, 63].

Feature selection aims to target important features, while eliminating redundant ones among the entire feature space. This leads to a set of informative features that can help improve learning performance in several aspects, including data mining, pattern recognition, and machine learning, as well as image recognition [118, 73, 171, 160]. Several statistical techniques have been used for feature selection, which can be broadly split into three main categories: wrapper methods; filter methods; embedded methods.

Analysing the overlap between the gene expression of different classes could be a key aspect to identify the discriminative capability of a gene [123]. The entire set of gene expression along with its target class information are utilised to select a set of informative genes that

can substantially enhance predictive classification model and its interpretation. An efficient Proportional Overlapping Score, called POS [123], have been proposed using three main factors to provide the overlapping score for each gene; the length of overlapping regions, the number of overlapping samples, and the ratio of each class' contribution to the overlapped samples. However, the key limitation is that POS only designed for binary classification problems. To address the restriction, we exploited the POS method to propose a novel feature selection technique, called 3cPOS, that can work effectively for three classification tasks. This method has balanced between simplicity of binary class problems and complexity of multi-class problems which provides ideal testing ground for new methodologies, addressing nuanced challenges without excessive complexity, see Chapter 5.

In this chapter, we aim to introduce extended versions of the POS and 3cPOS methods for multi-class problems, the multiple Proportional Overlap Score, referred to as mPOS.

7.2 Methods

Microarray gene expression data are commonly presented in the form of a matrix, (*i.e.*, $X = [x_{ij}]$, such that $X \in \mathbb{R}^{p \times n}$ and x_{ij} indicates the observed expression of gene i and tissue sample j, with $i = 1, \ldots, p$ and $j = 1, \ldots, n$. In multi-class problems, each tissue sample is assigned to one of k classes, *e.g.*, disease phenotypes, denoted by y_j which takes a distinct value k, with $k = 1, 2, 3, ..., K$. Figure 7.1 illustrates the structure of a gene expression matrix alongside its corresponding target class labels. The genes (features) typically arranged in rows while the tissue samples (observations) from which these genes are expressed are organized in columns. The class labels associated with each tissue sample are displayed in the last row of the matrix.

Analysing the overlap between the distributions of gene expressions across different classes can provide valuable insights into the classification effectiveness of the genes. The core concept is that a gene i is more likely to be crucial in accurately classifying tissue samples from class k to their correct class if its expression distribution for that class does not overlap with the distributions of other classes. In other words, gene i has the potential to correctly classify samples when their i^{th} expression values fall within a region (interval) of a single class, without overlapping with the i^{th} expression values of other classes.

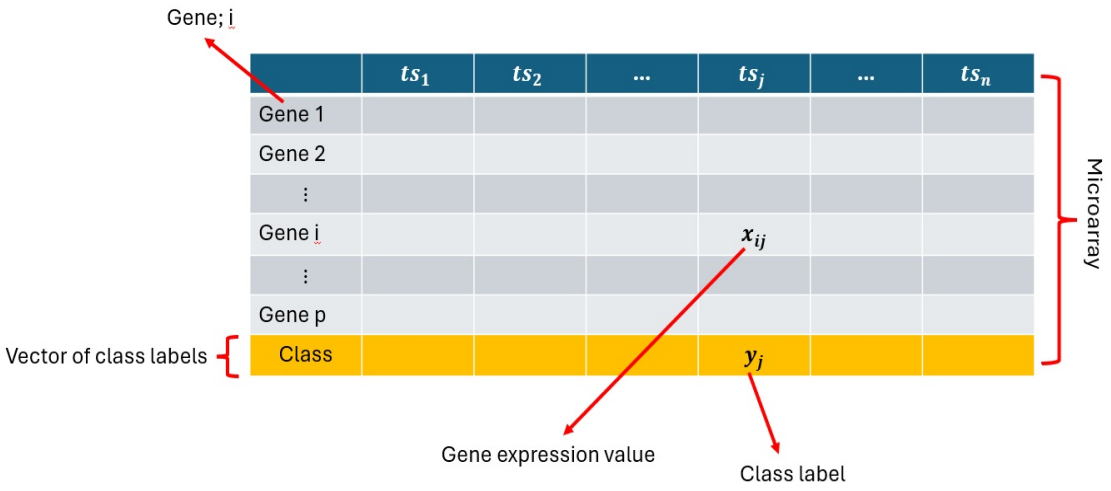

Figure 7.1: Layout of gene expression data along with their corresponding class labels

In Figure 7.2 (a), the expressions of gene i_1 from 80 samples lie within four non-overlapping ranges, with each range corresponding to one of the four classes. Consequently, gene i_1 is relevant to the classification task and serves as an informative feature for differentiating between the target classes in this problem. This framework demonstrates a perfect example of complete separation of expressions being categorized, with a clear gene indicating the preferred choice. This scenario is unlikely to occur in many complex and real-world classification problems. However, this idea suggests the main advantage of analysing the overlapping degrees of gene expressions in multi-class problems such as for providing informative gene ranking. Unlike gene i_1, many expression values belonging to genes i_2 and i_3 fall within overlapping regions of different classes, resulting in reducing their capability to differentiate between these classes, as illustrated in Figures 7.2 (b) and 7.2 (c). Moreover, Figure 7.2 (d) shows that gene i_4 expressions highly overlap among regions of the four classes, making it even less capable of differentiating between them. Therefore, gene i_4 expressions is identified as a non-informative gene.

A similar concept is outlined to consider to higher class problems, see in Figure 7.3. The expressions of gene i_1 from 100 samples fall into non-overlapping ranges are shown in Figure 7.3 (a) where each range corresponding to one of the five classes. As a result, gene i_1 is relevant to the classification task and serves as an informative feature to identify the correctly target classes in this scenario. Unlike gene i_1, many expression values corresponding to genes i_2 fall within overlapping regions of two different classes, resulting in reducing their capability to distinguish

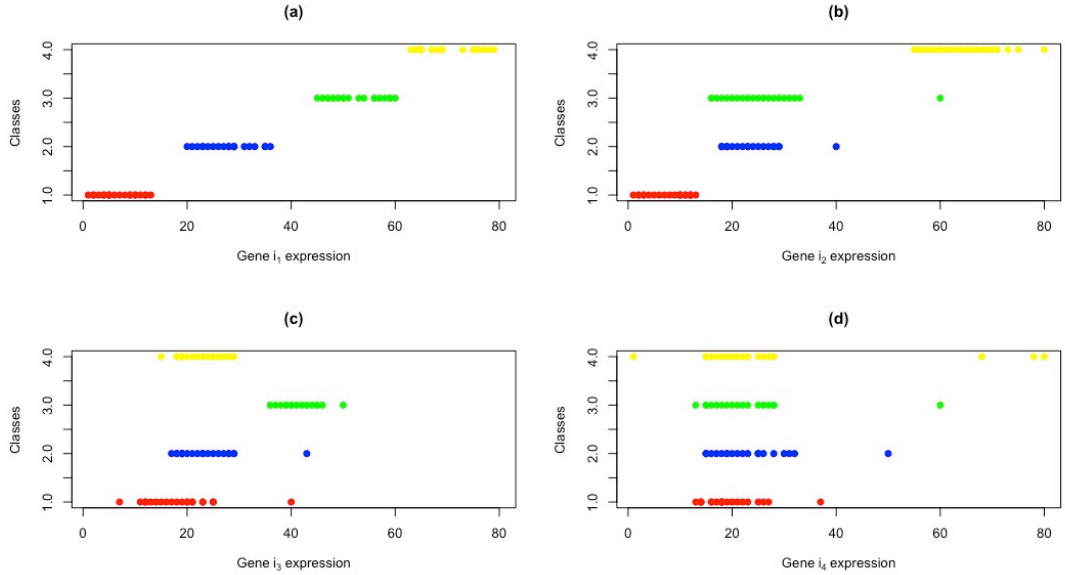

Figure 7.2: An example of four distinctive genes with different overlapping patterns. Expression values of four different genes (i_1, i_2, i_3, and i_4) each of which with 80 observations belonging to 4 classes, 20 observations for each class: (a) expression values of gene i_1, (b) expression values of gene i_2, (c) expression values of gene i_3, and (d) expression values of gene i_4.

between these classes, as illustrated in Figures 7.3 (b). Figures 7.3 (c) and 7.3 (d) demonstrates that the expression value of i_3 and i_4 fall into overlapping regions of three different classes, making them less capable to distinguish among these classes. The genes i_5 lie within overlapping regions of four different classes, making them even less capable to differentiate among these classes, see in Figure 7.3 (e). Furthermore, the expression values of gene i_6 highly overlap across the regions of all five classes. This results in a further reduction in ability to distinguish between them and it is identified as a non-informative gene, as shown in Figure 7.3 (f).

By utilising this approach, we propose mPOS method to improve the performance of the classification models for multi-class problems by selecting the most discriminative features via proportional overlap analysis. Our method initially exploits a standardisation approach to centre and scale expressions within standard ranges. The central limit theorem is applied to define a condensed region for each class, resulting in mitigating potential effects of outlier expressions on our analyses. To measure the capability of a gene to unambiguously detect tissue samples, the overlaps between the condensed expression intervals of that gene for different classes are considered. As highlighted in [123] and chapter 5, this is then employed to calculate an overlapping score, known as mPOS, for each gene by considering three factors: length of the

overlapping regions; number of overlapped samples; and ratio of the contribution of classes to the overlapped samples.

Further definitions used for the mPOS method are given below.

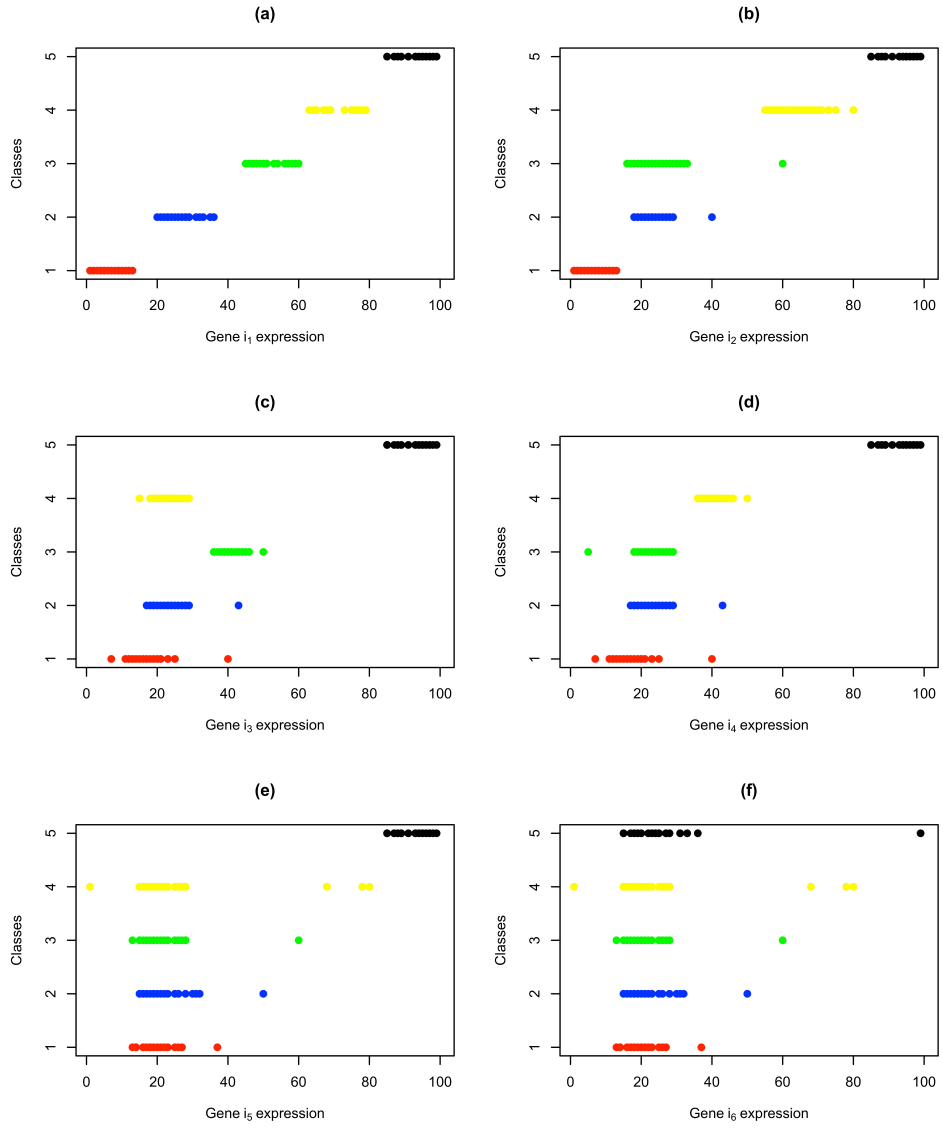

Figure 7.3: An example of five distinctive genes with different overlapping patterns. Expression values of five different genes (i_1, i_2, i_3, i_4, i_5, i_6) each of which with 100 observations belonging to 5 classes, 20 observations for each class: (a) expression values of gene i_1, (b) expression values of gene i_2, (c) expression values of gene i_3, (d) expression values of gene i_4, (e) expression values of gene i_5, and (f) expression values of gene i_6.

7.2.1 Class Intervals

Gene expression data contains noise values and outliers, which might lead to inaccurate results, poor predictions and wrong interpretations. This could be substantially more problematic when gene expressions are measured in the small number of samples as the variability of gene expression in a population may not be well accurately represented, resulting in poor decisions in gene selection [95]. To mitigate these challenges, the z score standardisation [127] is exploited. The entire set of expression values, irrespective of target classes, is standardized such that each gene has a mean of 0 and a standard deviation of 1 [186].

Definition 7.2.1. Core expression interval, denoted as $I_{i,k}$, can be defined for each gene i of class k as follows;

$$I_{i,k} = \bar{z}_{i,k} \pm \rho s_{i,k} \tag{7.1}$$

where $\bar{z}_{i,k}$ denotes the mean of standardised gene i expressions within class k and ρ represents the z-score corresponding to a specific percentage of the data in a standard distribution. The $s_{i,k}$ refers to the standard deviation of the expressions of the standarised gene i that belong to the class k.

Definition 7.2.2. Number of inlier observations, denoted as n_i, refers to the set of observations for each gene i whose standardised expression values fall within the corresponding core interval of their class. It can be defined by;

$$n_i = \{j : z_{ij} \varepsilon I_{i,k}, j = 1,2,3,..,n\} \tag{7.2}$$

Definition 7.2.3. The size of the total core interval, denoted as l_i, refers to the distance between the global minimum and maximum boundaries of the core expression intervals across the k classes.

$$l_i = [a_i, b_i] \tag{7.3}$$

where $a_i = min(a_{i,k})$ such that $\bar{z}_{i,k} - \rho_{i,k} s_{i,k}$ and $b_i = max(a_{i,k})$ such that $\bar{z}_{i,k} + \rho_{i,k} s_{i,k}$.

7.2.2 Overlapping between Intervals

Analysing the overlap between expression intervals of a gene for different classes is a crucial concept to consider for a gene's discriminative characteristic. Some related work can be found in [7, 12, 6, 123]. For multi-class problems, the overlap between intervals are defined as follows:

Definition 7.2.4. The interval of the ω-way overlapping intervals, denoted as $l_{\omega i(k_1k_2k_3\ldots k_K)}$, represents the intersection intervals among the core expressions of ω, where ω ranges from 2 to K (order of overlap). This can be expressed as

$$l_{\omega i(k_1k_2k_3\ldots k_K)} = \bigcap I_{i,k} \tag{7.4}$$

when $K = 3$ and $\omega = 2$, it provides as $l_{2i(k_1k_2)}$, $l_{2i(k_1k_3)}$, and $l_{2i(k_2k_3)}$. When we define $K = 3$ and $\omega = 3$, $l_{3i(k_1k_2k_3)}$ is considered. $\bar{l}_{\omega i}$ represents the average length of ω-way overlapping intervals across the binomial coefficient of K from ω. It can be defined as

$$\bar{l}_{\omega i} = \frac{\sum l_{\omega i(k_1k_2k_3\ldots k_K)}}{\binom{K}{\omega}} \tag{7.5}$$

Definition 7.2.5. The number of ω-way overlapping observation, expressed as $n_{\omega i}$, contains any observations that fall inside the interval of ω-way overlapping interval; $l_{\omega i(k_1k_2k_3\ldots k_K)}$. The number of ω-way overlapping observations is expressed as

$$n_{\omega i} = \{j : j \wedge z_{ij} \varepsilon l_{\omega i(k_1k_2k_3\ldots k_K)}\} \tag{7.6}$$

Definition 7.2.6. The total number of ω-way overlapping interval, expressed as $R_{\omega i}$, refers to the count of distinct intervals for gene i where the standardised expression values lie within the intervals of ω-way overlaps, $l_{\omega i(k_1k_2k_3\ldots k_K)}$. The total number of ω-way overlapping interval is expressed as;

$$R_{\omega_i} = \begin{cases} \left|\{(k_1k_2k_3...k_K)\} \mid \exists j \in n_i,\ z_{ij} \in l_{\omega_i}(k_1k_2k_3...k_K)\right|, & \text{if the set is not empty} \\ 0, & \text{otherwise} \end{cases} \tag{7.7}$$

7.2.3 mPOS Measure

Our novel method has extended versions of the POS [123] and 3cPOS in Chapter 5. A novel generalised version, called the mPOS score, is proposed in this chapter. Unlike two classification [123] as well as three classification problems [Chapter 5], we aim at focusing on the multi-class problems of a certain gene i into consideration to derive an overlapping score for multi-class problems. The overlapping score can then be defined as follows:

$$mPOS_i = \frac{1}{n_i l_i} \sum_{\omega=2}^{K} \left[\omega \bar{l}_{\omega i} n_{\omega i} \frac{R_{\omega i}}{\binom{K}{\omega}} \prod_{k; n_{\omega i,k}>0} \beta_{\omega i,k} \right] \tag{7.8}$$

where n_i is expressed as number of inlier observations and l_i is the size of the total core interval. The $\bar{l}_{\omega i}$ refers the average size of ω-way overlapping intervals, while the $n_{\omega i}$ represents number of ω-way overlapping observations. $R_{\omega i}$ is the total number of ω-way overlapping intervals and $\beta_{\omega i,k}$ represents the proportion of total contribution of class k observations within the ω-way overlapping observations. It is given by:

$$\beta_{\omega i,k} = \frac{n_{\omega i,k}}{N_{\omega i}} \tag{7.9}$$

where $n_{\omega i,k}$ represents the number of inlier observations from class k that whose standardised expressions fall inside the intervals of ω-way overlap for gene i and $N_{\omega i}$ refers the total number of inlier observations that whose standardised expressions lie within the interval of ω-way overlaps for gene i. For individual gene i expression, a smaller mPOS score offers higher discriminative capability in classifying between classes.

The pseudo-code of the multiple Proportional Overlapping Scores (mPOS) algorithm is shown in Algorithm 3. Let $\mathbb{G}$ denote the set of all genes, where $\mid \mathbb{G} \mid = p$. Initially, the observed expression values, X, across the entire dataset are standardised (lines 2-4). The standardised

expressions, along with their true target class labels, are then utilised to derive the core interval for class 1, 2, 3, ..., and K (*i.e.* $I_{i,1}$, $I_{i,2}$, $I_{i,3}$,...,$I_{i,K}$) (lines 5-7). For each gene i, the number of inliers observation (lines 8) and the size of total core expression interval (line 9) are calculated. To analyse the overlap between intervals, the average length of the ω-way overlapping intervals is computed (lines 10). The number of ω-way overlapping observations (line 11) and the total number of ω-way overlapping intervals (line 12) are calculated. Consequently, the mPOS score for each gene i is computed (lines 13) before creating a sequence of genes, $\mathbb{G}^*$, (lines 14) that are ranked in an ascending order based on the mPOS score. Finally, the top r genes in $\mathbb{G}^*$ is selected for the corresponding classification task (line 15).

Algorithm 3 mPOS Method For Gene Selection

Input: The observed expression values of all genes (X), target class labels (Y) and number of genes to be selected (r).

Output: Sequence of the selected genes $(\mathbb{T})$.

1: **for all** $i \in \mathbb{G}$ **do**
2: **for** $j = 1$ to n **do**
3: Transform x_{ij} into their z-score standardisation using $z_{ij} = (x_{ij} - \bar{x}_i)/s_i$
4: **end for**
5: **for** $k = 1$ to K **do**
6: Calculate $I_{i,k}$ as defined in equation (7.1), representing the core expression interval for each class k of gene i.
7: **end for**
8: Compute the number of inlier observation, n_i, as defined in equation (7.2).
9: Compute the size of total core interval, l_i, as defined in equation (7.3).
10: Compute the average length of ω-way overlapping region, $\bar{l}_{\omega i}$, as defined in equation (7.5).
11: Compute the number of ω-way overlapping observation, $n_{\omega i}$, as defined in equation (7.6).
12: Compute the total number of ω-way overlapping interval, $R_{\omega i}$, as defined in equation (7.7).
13: Calculate $mPOS_i$ as defined in equation (7.8).
14: Create $\mathbb{G}^*$ which is an ordered list of features (genes) in $\mathbb{G}$, sorted by ascending order of *mPOS* values.
15: Define $\mathbb{T}$ as first r genes in $\mathbb{G}^*$.
16: **end for**
17: **return** $\mathbb{T}$

7.3 Illustrated Examples

In this section, we provide simulated examples to demonstrate the utility of the mPOS method. Each example is accompanied by simulated data points corresponding to their respective target class labels. For the first example, a total of 18 samples are included, with 6, 3, 4, and 5 samples corresponding to class 1, 2, 3, and 4, respectively. Figure 7.4 shows the distribution of Gene 1 expressions, which involves only 2-way overlapping regions. According to definition 7.2.1, we take the z-score for 95% confidence, resulting in ρ as 1.96. This yields the following intervals: $I_{1,1} = (-0.02, 5.62)$, $I_{1,2} = (6.15, 10.4)$, $I_{1,3} = (3.14, 5.26)$, and $I_{1,4} = (1.07, 2.85)$.

By the following definition in 7.2.2, the number of inlier observations (n_1) is 18. The size of the total core interval (l_1) is 10.42 by considering definition 7.2.3. Since this example consists of a 4-class problems, the mPOS can be computed as follows:

For 2-way overlaps; the average length of 2-way overlapping intervals can be computed as follows;

$$\begin{aligned}\bar{l}_{2i} &= \left[\frac{l_{2i(k_1k_2)} + l_{2i(k_1k_3)} + l_{2i(k_1k_4)} + l_{2i(k_2k_3)} + l_{2i(k_2k_4)} + l_{2i(k_3k_4)}}{\binom{4}{2}}\right] \\ &= \frac{0 + 2.12 + 1.78 + 0 + 0 + 0}{6} \\ \bar{l}_{2i} &= 0.65\end{aligned}$$

Next, the number of 2-way overlapping observations is computed, resulting in n_{2i} as 13. The total number of 2-way overlapping intervals is 2 and the proportion of the total contribution of class k observations among the 2-way overlapping observations is $(\frac{4}{13})(\frac{4}{13})(\frac{5}{13})$.

There are no 3-way or 4-way overlaps observed, making an average length of the 3-way and 4-way overlapping intervals become zero. Consequently, the number of 3-way and 4-way overlapping observations, as well as the proportion of class k observations among these overlapping observations, are all zero. According to equation 7.9, the mPOS for Gene 1 Expression is calculated to be 0.0011.

In the next example, a total of 22 samples are included, with 3, 3, 4, 5, and 7 samples corresponding to class 1, 2, 3, 4, and 5 respectively, see in Figure 7.5. This example displays

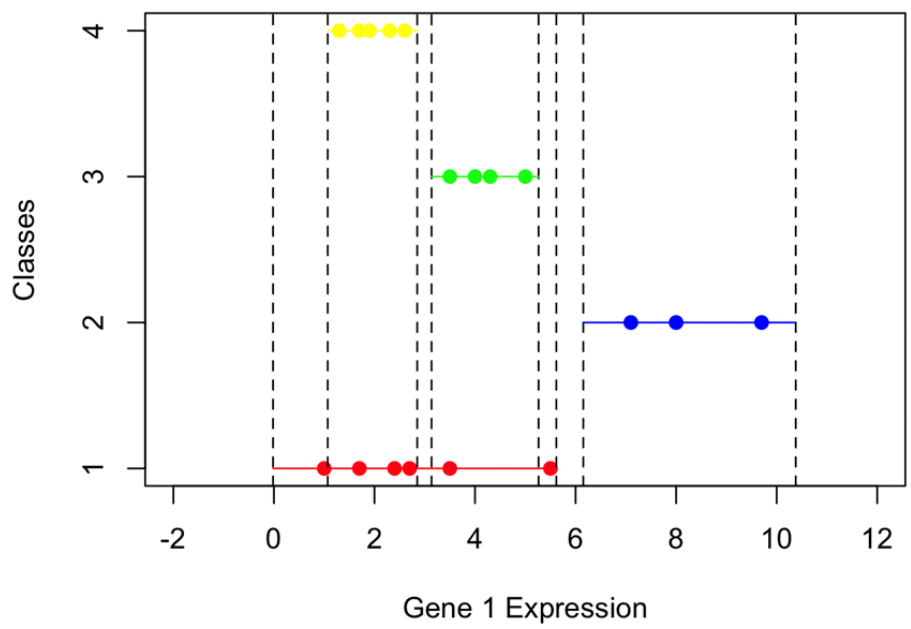

Figure 7.4: Distribution of Gene 1 expressions.

the distribution of Gene 2 expressions, where it involves of 2-way, 3-way, and 4-way overlaps. According to definition 7.2.1, we take the z-score for 95% confidence, resulting in ρ as 1.96. The following intervals are obtained: $I_{2,1} = (7.33, 9.20)$, $I_{2,2} = (6.24, 10.6)$, $I_{2,3} = (3.88, 8.77)$, $I_{2,4} = (1.38, 4.78)$, and $I_{2,5} = (1.69, 11.2)$.

By following definition 7.2.2, the number of inlier observations (n_2) is 22. The size of the total core interval (l_2) is 9.82, as calculated according to the definition in 7.2.3. Since this example consists of 5 class problems, mPOS can be computed as follows;

For 2-way overlaps; the average length of 2-way overlapping intervals can be computed as follows;

$$\bar{l}_{2i} = \left[\frac{\begin{array}{c} l_{2i(k_1k_2)} + l_{2i(k_1k_3)} + l_{2i(k_1k_4)} + l_{2i(k_1k_5)} + l_{2i(k_2k_3)} \\ + l_{2i(k_2k_4)} + l_{2i(k_2k_5)} + l_{2i(k_3k_4)} + l_{2i(k_3k_5)} + l_{2i(k_4k_5)} \end{array}}{\binom{5}{2}} \right]$$

$$= \frac{1.87 + 1.44 + 0 + 1.87 + 2.53 + 0 + 4.36 + 0.9 + 4.89 + 3.09}{10}$$

$$\bar{l}_{2i} = 2.005$$

Next, the number of 2-way overlapping observations is computed, resulting in n_{2i} as 7. The total

number of 2-way overlapping intervals is 2 and the proportion of the total contribution of class k observations among the 2-way overlapping observations is $(\frac{1}{7})(\frac{3}{7})(\frac{3}{7})$.

For 3-way overlaps; the average length of 3-way overlapping intervals can be computed as follows;

$$\begin{aligned}\bar{l}_{3i} &= \left[\frac{\begin{array}{c} l_{3i(k_1k_2k_3)}+l_{3i(k_1k_2k_4)}+l_{3i(k_1k_2k_5)}+l_{3i(k_1k_3k_4)}+l_{3i(k_1k_3k_5)} \\ +l_{3i(k_1k_4k_5)}+l_{3i(k_2k_3k_4)}+l_{3i(k_2k_3k_5)}+l_{3i(k_2k_4k_5)}+l_{3i(k_3k_4k_5)}\end{array}}{\binom{5}{3}}\right] \\ &= \frac{1.44+0+1.87+0+1.44+0+0+2.53+0+0.9}{10} \\ \bar{l}_{3i} &= 0.82\end{aligned}$$

Next, the number of 3-way overlapping observations is computed, resulting in n_{3i} as 7. The total number of 3-way overlapping intervals is 2 and the proportion of the total contribution of class k observations among the 3-way overlapping observations is $(\frac{1}{7})(\frac{4}{7})(\frac{1}{7})(\frac{1}{7})$.

For 4-way overlaps; the average length of 4-way overlapping intervals can be computed as follows;

$$\begin{aligned}\bar{l}_{4i} &= \left[\frac{l_{3i(k_1k_2k_3k_4)}+l_{3i(k_1k_2k_3k_5)}+l_{3i(k_1k_2k_4k_5)}+l_{3i(k_1k_3k_4k_5)}+l_{3i(k_2k_3k_4k_5)}}{\binom{5}{4}}\right] \\ &= \frac{0+1.44+0+0+0}{5} \\ \bar{l}_{4i} &= 0.29\end{aligned}$$

Next, the number of 4-way overlapping observations is computed, resulting in n_{4i} as 7. The total number of 4-way overlapping intervals is 1 and the proportion of the total contribution of class k observations among the 4-way overlapping observations is $(\frac{3}{7})(\frac{1}{7})(\frac{3}{7})$.

Due to no 5-way overlaps, the average length of 5-way overlapping intervals, the number of 5-way overlapping observations, and the proportion of class k observations among the 5-way overlapping observations become zero. By using 7.9, mPOS for Gene 2 Expression is 0.0009.

In the following example, a total of 26 samples are included, with 3, 3, 4, 5, 7, and 4 samples corresponding to class 1, 2, 3, 4, 5, and 6 respectively. Figure 7.6 demonstrates the distribution of

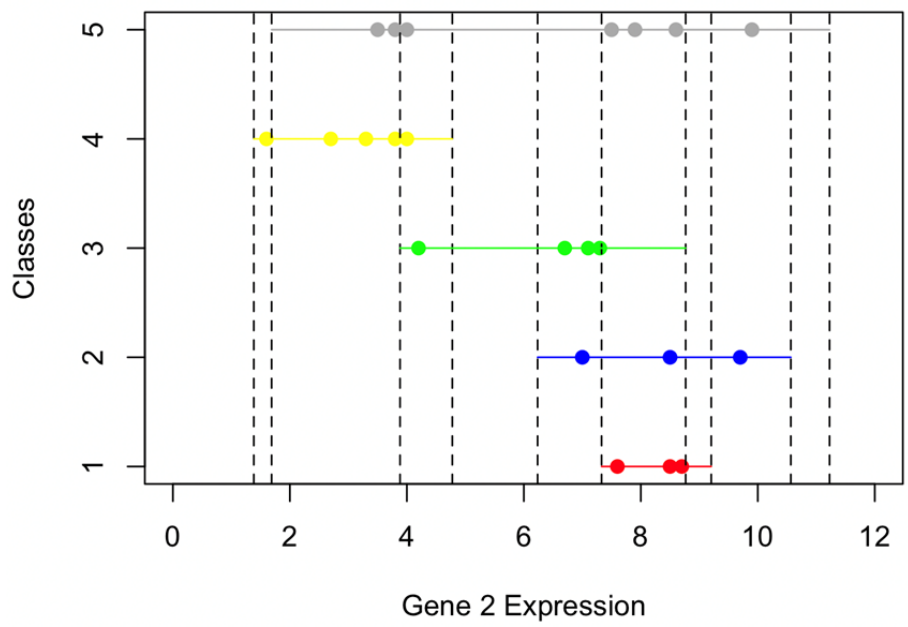

Figure 7.5: Distribution of Gene 2 expressions.

Gene 3 expressions, which involves 2-way, 3-way, and 4-way overlaps. According to definition 7.2.1, we take z-score for 95% confidence, resulting in ρ as 1.96. The following intervals are then obtained: $I_{3,1} = (7.33, 9.20)$, $I_{3,2} = (6.24, 10.6)$, $I_{3,3} = (3.88, 8.77)$, $I_{3,4} = (1.38, 4.78)$, $I_{3,5} = (1.69, 11.2)$, and $I_{3,6} = (9.40, 11.6)$.

By following the definition in 7.2.2, the number of inlier observations (n_3) is 26. The size of the total core interval (l_3) is 10.22, as calculated from the definition in 7.2.3. Since this example consists of 6 class problems, mPOS can be computed as follows

For 2-way overlaps; the average length of 2-way overlapping intervals can be computed as follows;

$$\bar{l}_{2i} = \left[\frac{\begin{array}{l} l_{2i(k_1k_2)} + l_{2i(k_1k_3)} + l_{2i(k_1k_4)} + l_{2i(k_1k_5)} + l_{2i(k_1k_6)} + l_{2i(k_2k_3)} + l_{2i(k_2k_4)} \\ \quad + l_{2i(k_2k_5)} + l_{2i(k_2k_6)} + l_{2i(k_3k_4)} + l_{2i(k_3k_5)} + l_{2i(k_3k_6)} + l_{2i(k_4k_5)} + l_{2i(k_4k_6)} + l_{2i(k_5k_6)} \end{array}}{\binom{6}{2}} \right]$$

$$= \frac{1.87 + 1.44 + 0 + 1.87 + 0 + 2.53 + 0 + 4.36 + 1.2 + 0.9 + 4.89 + 0 + 3.09 + 0 + 1.8}{15}$$

$$\bar{l}_{2i} = 1.60$$

Next, the number of 2-way overlapping observations is computed, resulting in n_{2i} as 7. The total

number of 2-way overlapping intervals is 2 and the proportion of the total contribution of class k observations among the 2-way overlapping observations is $(\frac{3}{7})(\frac{2}{7})(\frac{2}{7})$.

For 3-way overlaps; the average length of 3-way overlapping intervals can be computed as follows;

$$\bar{l}_{3i} = \left[\frac{\begin{array}{c} l_{3i(k_1k_2k_3)} + l_{3i(k_1k_2k_4)} + l_{3i(k_1k_2k_5)} + l_{3i(k_1k_2k_6)} + l_{3i(k_1k_3k_4)} + l_{3i(k_1k_3k_5)} + l_{3i(k_1k_3k_6)} \\ + l_{3i(k_1k_4k_5)} + l_{3i(k_1k_4k_6)} + l_{3i(k_1k_5k_6)} + l_{3i(k_2k_3k_4)} + l_{3i(k_2k_3k_5)} + l_{3i(k_2k_3k_6)} + l_{3i(k_2k_4k_5)} \\ + l_{3i(k_2k_4k_6)} + l_{3i(k_2k_5k_6)} + l_{3i(k_3k_4k_5)} + l_{3i(k_3k_4k_6)} + l_{3i(k_3k_5k_6)} + l_{3i(k_4k_5k_6)} \end{array}}{\binom{6}{3}} \right]$$

$$= \left[\frac{\begin{array}{c} 1.44+0+1.87+0+0+1.44+0+0+0+0 \\ +0+2.53+0+0+0+1.20+0.90+0+0+0 \end{array}}{20} \right]$$

$$\bar{l}_{3i} = 0.47$$

Next, the number of 3-way overlapping observations is computed, resulting in n_{3i} as 11. The total number of 3-way overlapping intervals is 3 and the proportion of the total contribution of class k observations among the 3-way overlapping observations is $(\frac{2}{11})(\frac{4}{11})(\frac{1}{11})(\frac{2}{11})(\frac{2}{11})$.

For 4-way overlaps; the average length of 4-way overlapping intervals can be computed as follows;

$$\bar{l}_{4i} = \left[\frac{\begin{array}{c} l_{4i(k_1k_2k_3k_4)} + l_{4i(k_1k_2k_3k_5)} + l_{4i(k_1k_2k_3k_6)} + l_{4i(k_1k_2k_4k_5)} + l_{4i(k_1k_2k_4k_6)} \\ l_{4i(k_1k_2k_5k_6)} + l_{4i(k_1k_3k_4k_5)} + l_{4i(k_1k_3k_4k_6)} + l_{4i(k_1k_3k_5k_6)} + l_{4i(k_1k_4k_5k_6)} \\ + l_{4i(k_2k_3k_4k_5)} + l_{4i(k_2k_3k_4k_6)} + l_{4i(k_2k_3k_5k_6)} + l_{4i(4i(k_2k_4k_5k_6)} + l_{4i(k_3k_4k_5k_6)} \end{array}}{\binom{6}{4}} \right]$$

$$= \left[\frac{\begin{array}{c} 0+1.44+0+0+0+0+0 \\ 0+0+0+0+0+0+0 \end{array}}{15} \right]$$

$$\bar{l}_{4i} = 0.10$$

Next, the number of 4-way overlapping observations is computed, resulting in n_{4i} as 7.

The total number of 4-way overlapping intervals is 1 and the proportion of class k observations among the 4-way overlapping observations is $(\frac{3}{7})(\frac{1}{7})(\frac{3}{7})$.

Since there are no 5-way and 6-way overlaps observed, the average length of the 5-way and 6-way overlapping intervals, the number of 5-way and 6-way overlapping observations, and the proportion of class k observations among the 5-way and 6-way overlapping observations all become zero. According to 7.9, mPOS for Gene 3 Expression is 0.0004.

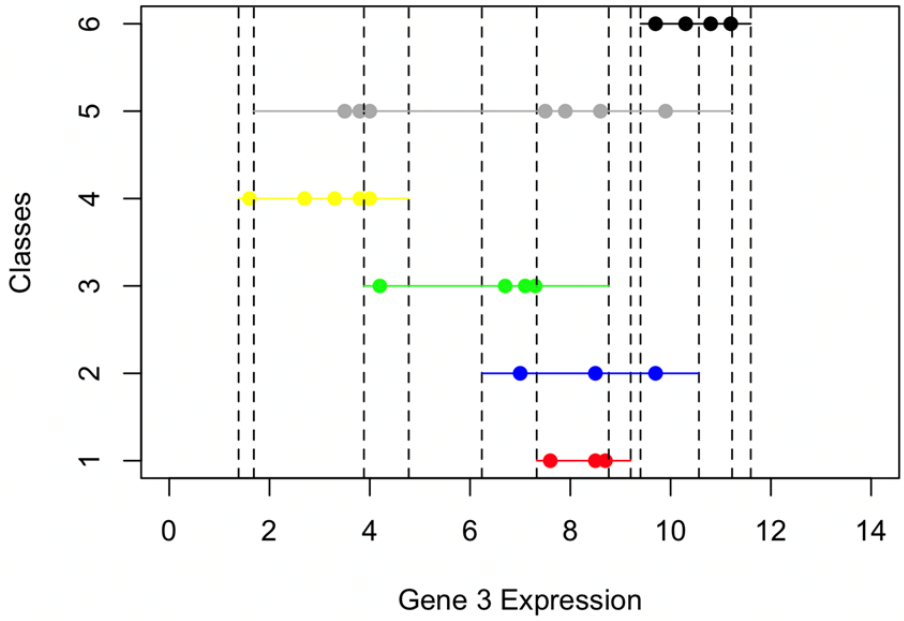

Figure 7.6: Distribution of Gene 3 expressions.

7.4 Experimental Setup

For the performance evaluation of feature selection techniques, one can measure the accuracy of a classifier that is applied after the feature selection process. As a result, the classification is solely based on the selected gene expressions. This approach can assess the effectiveness of the feature selection techniques in identifying discriminative genes. Eight gene selection methods are applied and demonstrate that gene selection methods play an effective role in validating a classifier's accuracy [112]. This strategy has been employed in numerous studies, including [143, 192, 29, 183].

In this chapter, we conducted an experiment using twenty-four gene expression datasets

to evaluate the mPOS method. The evaluation involved a comparison with five well-known gene selection methods depending on different class scenarios; Wilcoxon, Kruskal, LASSO, mRMR, and our proposed method, mPOS. Specifically, the Wilcoxon, LASSO, mRMR, and mPOS techniques were applied to datasets that contain binary class problems. Nevertheless, the Kruskal method is implemented for datasets corresponding to multiple-class problems instead of the Wilcoxon method, while LASSO, mRMR, and mPOS techniques were maintained. The performance of these methods was validated by obtaining the classification accuracy from four classifiers: Random Forest (RF), k-Nearest Neighbors (kNN), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost).

To avoid the under-estimation of prediction error, twenty repetition of 5-fold cross-validation analysis were performed for each combination of the datasets, gene selection techniques, selected gene sizes (20 different gene set sizes) and classifiers. The top 20 most informative genes were selected from each feature selection technique to compare the quality of gene selection using RF, kNN, SVM, and XGBoost classifiers which are well-established and frequently utilised in machine learning research [197]. The purpose of comparing these classifiers is to help understand the impact of feature selection on model performance as well as detecting the most informative features for analyses.

For software evaluation, we used an R programming language [28], statistical computing and visualization, for the entire analysis. The R package randomForest [114] is used to implement Random Forest with its default values for ntree, mtry, and nodesize: 500; the square root of the number of predictors; and 1. The R package class [156] is employed to perform the K-Nearest Neighbors classifier with a default parameter k set to the closest odd number of neighbors. The R package e1071 [48] is utilised to implement Support Vector Machine classifier with different types of kernels. For simplicity, linear kernel is applied for SVM. The R package xgboost [33] performs Extreme Gradient Boosting classification.

For each fold of feature selection, the Least Absolute Shrinkage Operator Selector (LASSO), Minimum Redundancy and Maximum Relevance (mRMR), Wilcoxon test (Wilcoxon), and our proposed method (mPOS) are applied to analyse binary classification problems. These methods identify a subset of informative genes, with the subset size denoted as r, $r = 1, 2, ..., 20$. In

place of the Wilcoxon, the Kruskal Wallis test (Kruskal) was considered as an alternative to complement multiple classification problems, alongside the other feature selection techniques. The R package stats [172] is used to implement Wilcoxon and Kruskal. The R package mRMRe [45] is employed for mRMR, while the R package glmnet [61] is used to apply LASSO.

The large number of genes, coupled with the small size of certain classes in the datasets, poses challenges in implementing the mRMRe and glmnet packages. The key limitation of the R package mRMRe is that this package cannot be implemented with datasets including beyond 46340 features. This results in the exclusion of GSE6861, GSE10780, GSE19615, GSE22513, GSE21029, GSE102079, GSE21510, GSE27854(2), GSE27651, GSE38666, GSE40595(2), and GSE162228(2). Similarly, the R package glmnet imposes a restriction on the analysis of some datasets when training folds are formed by a small subset of samples from a class. As a result, the GSE22513, GSE4045, GSE22093, GSE23938, GSE15852, GSE27651, GSE38666, GSE40595(2), GSE162228(2), Brain Tumour, and Lung(2) datasets are excluded for the LASSO method.

The evaluation is carried out according to the following procedure:

1. Each data set is divided into training and testing data using random splitting. 5-fold cross-validation is applied by conducting 80% for training data and another 20% for testing data. This step is repeated 20 times, resulting in 100 runs.

2. For feature selection process, LASSO, mRMR, Wilcoxon, and our proposed method, mPOS, are implemented to the training data to select the top 20 ranked informative genes from the entire set of genes for binary clssification problems. Whilst LASSO, mRMR, Kruskal, and mPOS are used to analyse the training data to select the top 20 ranked informative genes out of all genes for multi classification problems.

3. Random Forest, K-Nearest Neighbours, Support Vector Machine, and Extreme Gradient Boost are employed to construct classification models using the top r selected informative genes for each $r = 1, 2, ..., 20$ from the ranked gene set derived from feature selection methods. These models are trained on the corresponding training data to assess their performance based on varying subsets of the most informative genes

4. The class probabilities for the testing data are predicted using the fitted classification models, which were trained on the 20 different gene subsets, with sizes $r = 1, 2, ..., 20$. This allows for an assessment of their generalised performance on unseen data.

5. The average classification accuracy is calculated based on the predictions and the true class labels of the testing data across the total of 100 runs.

7.5 Results and Discussion

7.5.1 Performance Analysis for Classification Accuracy

Based on the previously described experimental setup, we evaluated the performance of feature selection algorithms concerning classification accuracy. Classification accuracy is the most common performance metric for verifying the effectiveness of feature selection. For given datasets and learning algorithms, we use the following criteria to compare the performance of feature selection algorithms: If a feature selection method achieves a higher classification accuracy than the other feature selection approaches, then its performance is considered superior [183].

The average classification accuracies on the GSE6861 dataset using RF, kNN, SVM, and XGBoost classifiers are shown in Figure 7.7. It demonstrates that Wilcoxon performs better than other feature selection methods at the different set sizes of informative genes using the RF classifier. Moreover, the Wilcoxon provides the best performance at the small and moderate set sizes of informative genes using the KNN and XGBoost classifiers. Specifically, the Wilcoxon outperforms all other feature selection techniques at the small set sizes of informative genes using the SVM classifier.

Figure 7.8 demonstrates the average classification accuracies on the GSE10780 dataset using RF, KNN, SVM, and XGBoost classifiers. It reveals that mPOS performs better than other feature selection techniques at a single informative gene across four different classifiers. Moreover, mPOS shows performance comparable to that of LASSO from a set size of two informative genes onward across all classifiers.

Figure 7.9 demonstrates the average classification accuracies on the GSE19615 dataset using

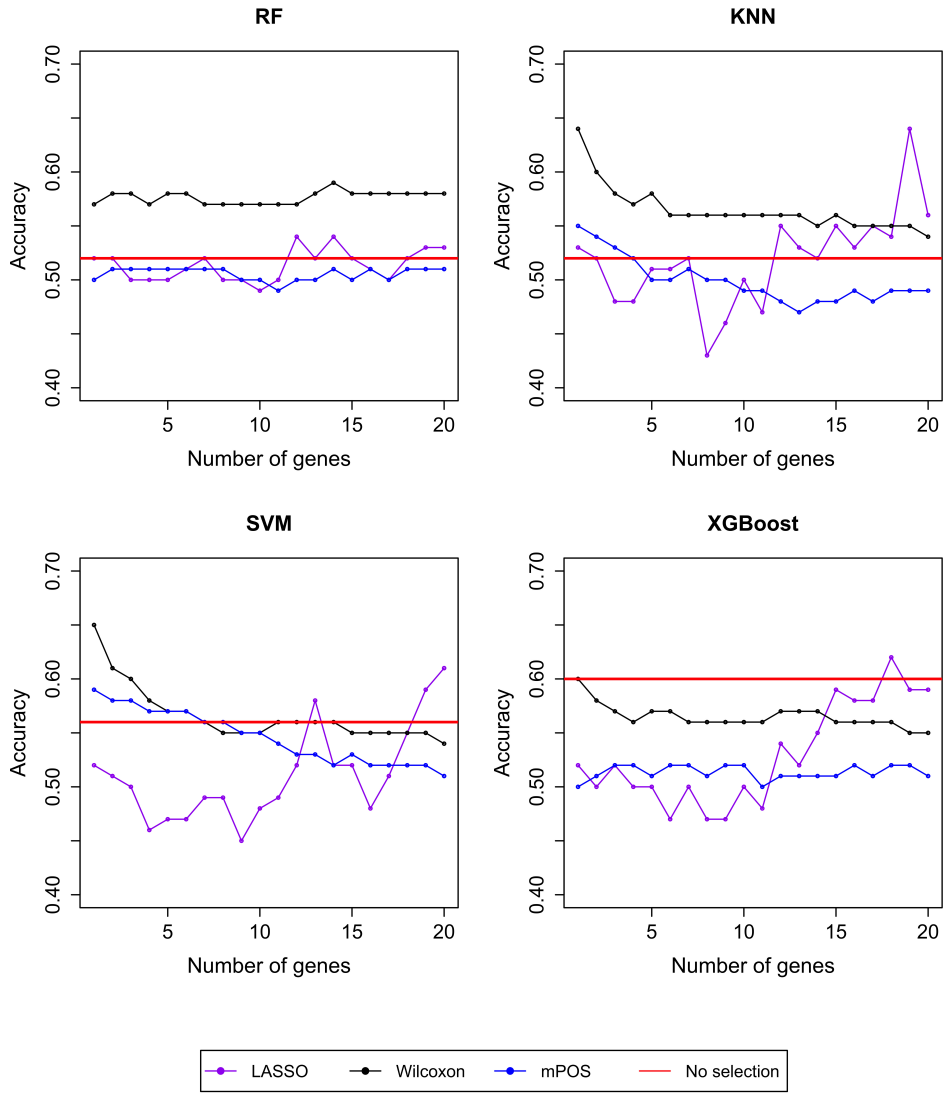

Figure 7.7: Averages of classification accuracy for GSE6861 dataset. Average classification accuracy for GSE6861 data based on 20 repetitions 5-fold CV using LASSO, Wilcoxon, mPOS, and the full set of features.

RF, KNN, SVM, and XGBoost classifiers. It shows that mPOS outperforms all other feature selection methods at the small and moderate set sizes of informative genes using RF, KNN, SVM, and XGBoost classifiers. However, LASSO provides the best performance at the large set size of informative genes across all classifiers.

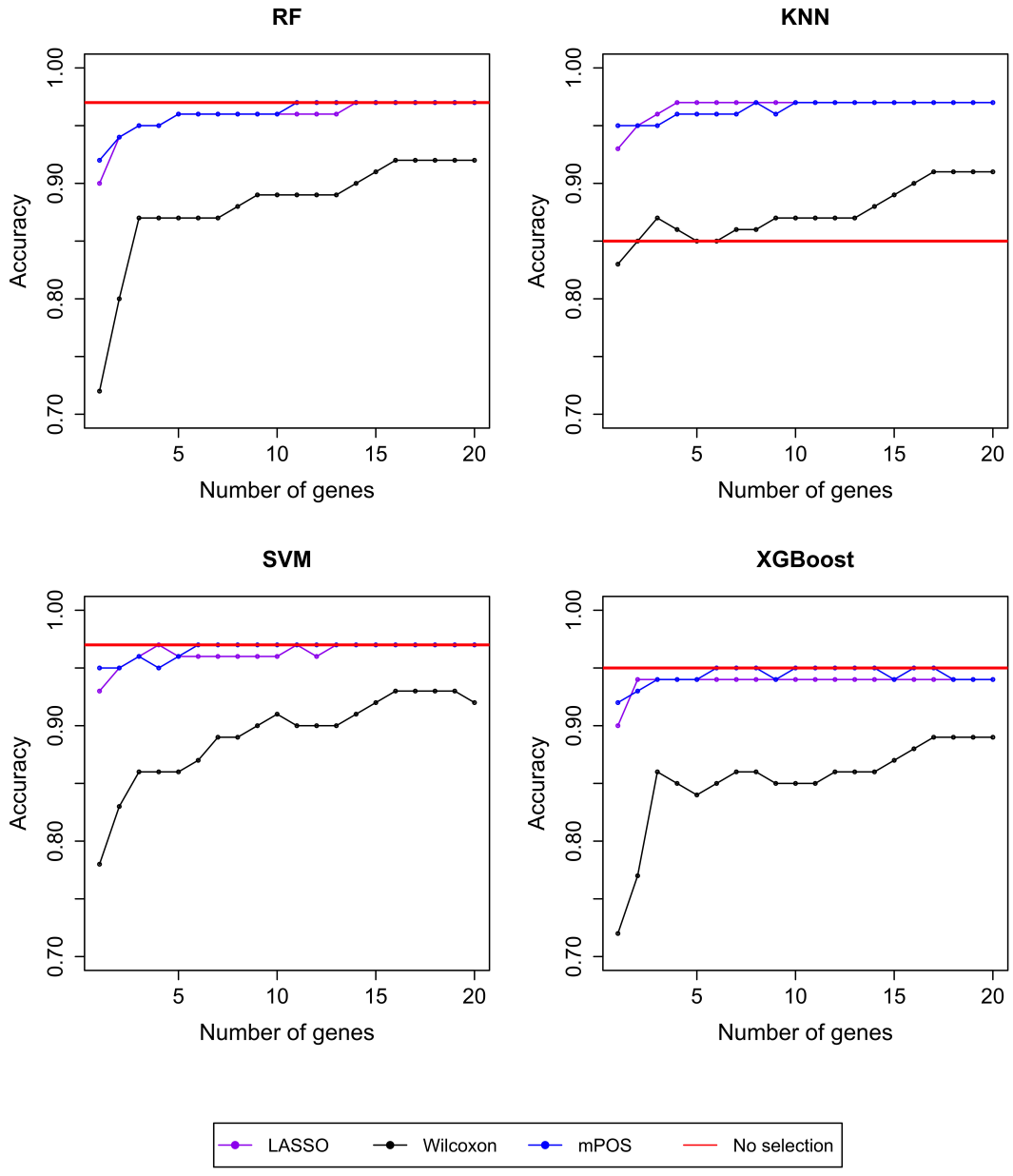

Figure 7.8: Average of classification accuracy for the GSE10780 dataset. Average classification accuracy for GSE10780 data based on 20 repetitions of 5-fold CV using LASSO, Wilcoxon, mPOS, and the full set of feature.

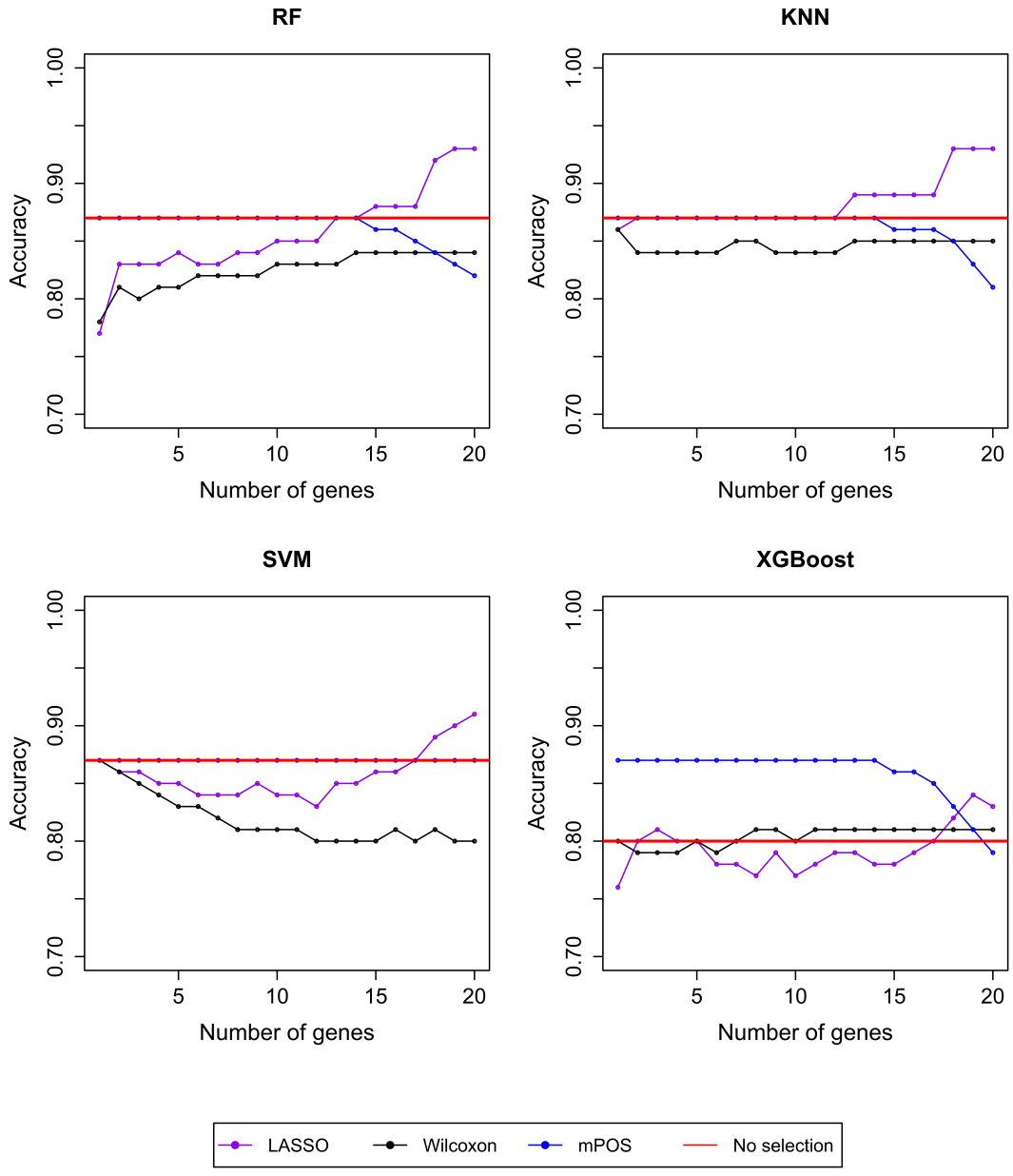

Figure 7.9: Average of classification accuracy for the GSE19615 dataset. Average classification accuracy for GSE19615 data based on 20 repetitions of 5-fold CV using LASSO, Wilcoxon, mPOS, and the full set of feature.

The average classification accuracies on the GSE22513 dataset using RF, KNN, SVM, and XGBoost classifiers is presented in Figure 7.10. It shows that mPOS outperforms Wilcoxon method at the different set sizes of informative genes across four classifiers with classification accuracy of up to 93%.

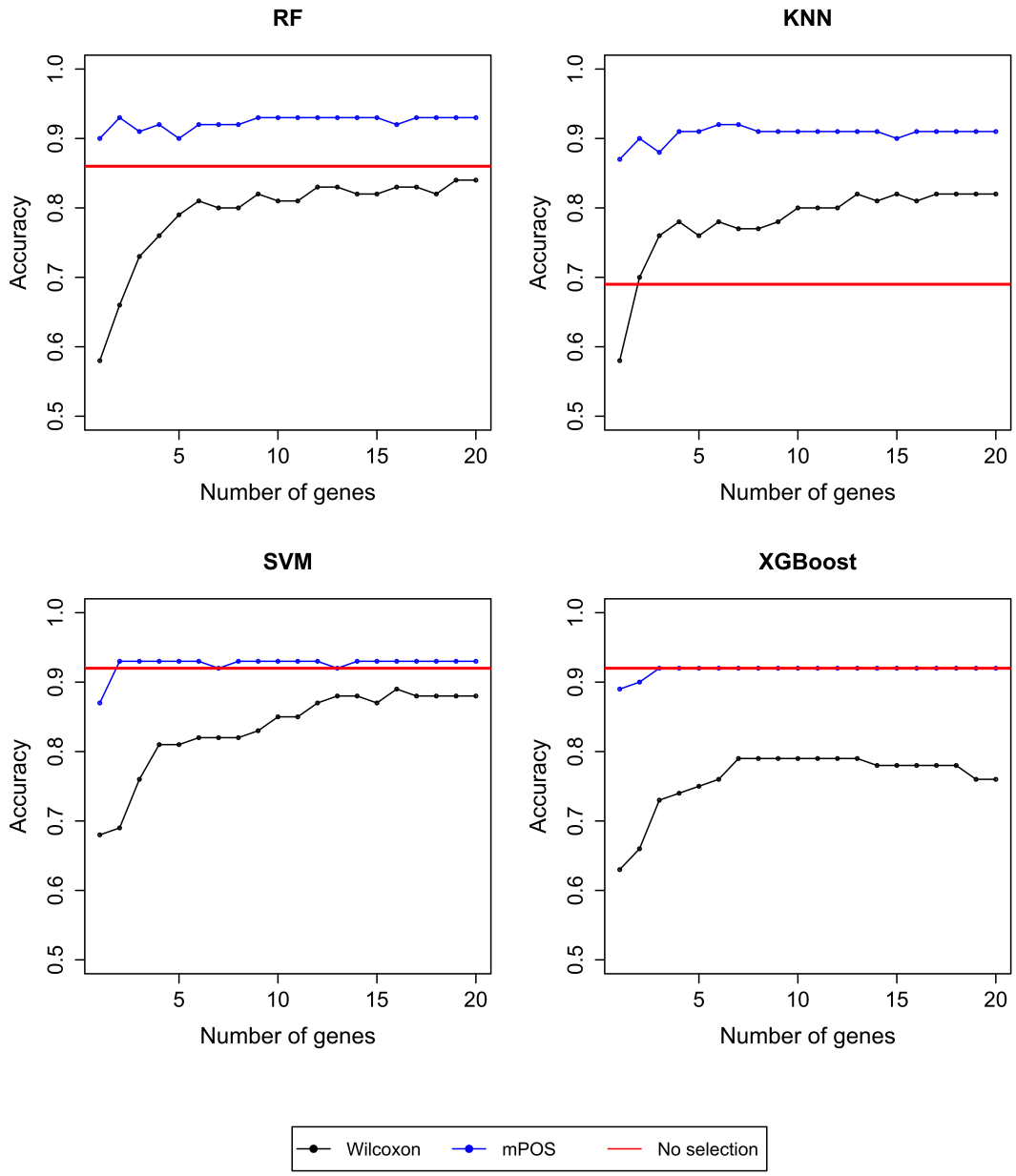

Figure 7.10: Average of classification accuracy for the GSE22513 dataset. Average classification accuracy for GSE22513 data based on 20 repetitions of 5-fold CV using Wilcoxon, mPOS, and the full set of feature.

Figure 7.11 shows the average classification accuracies on the GSE24514 dataset using RF, KNN, SVM, and XGBoost classifiers. It reveals that mPOS performs better than other

techniques at a single informative gene across four classifiers. The mRMR outperforms all other feature selection methods at a set of 2 informative genes onward using the KNN classifier, while the mRMR's performance is the best at the small and moderate set sizes of informative genes using RF and XGBoost classifiers. Furthermore, the LASSO performs better than other feature selection techniques at the moderate and large set sizes of informative genes using the SVM classifier.

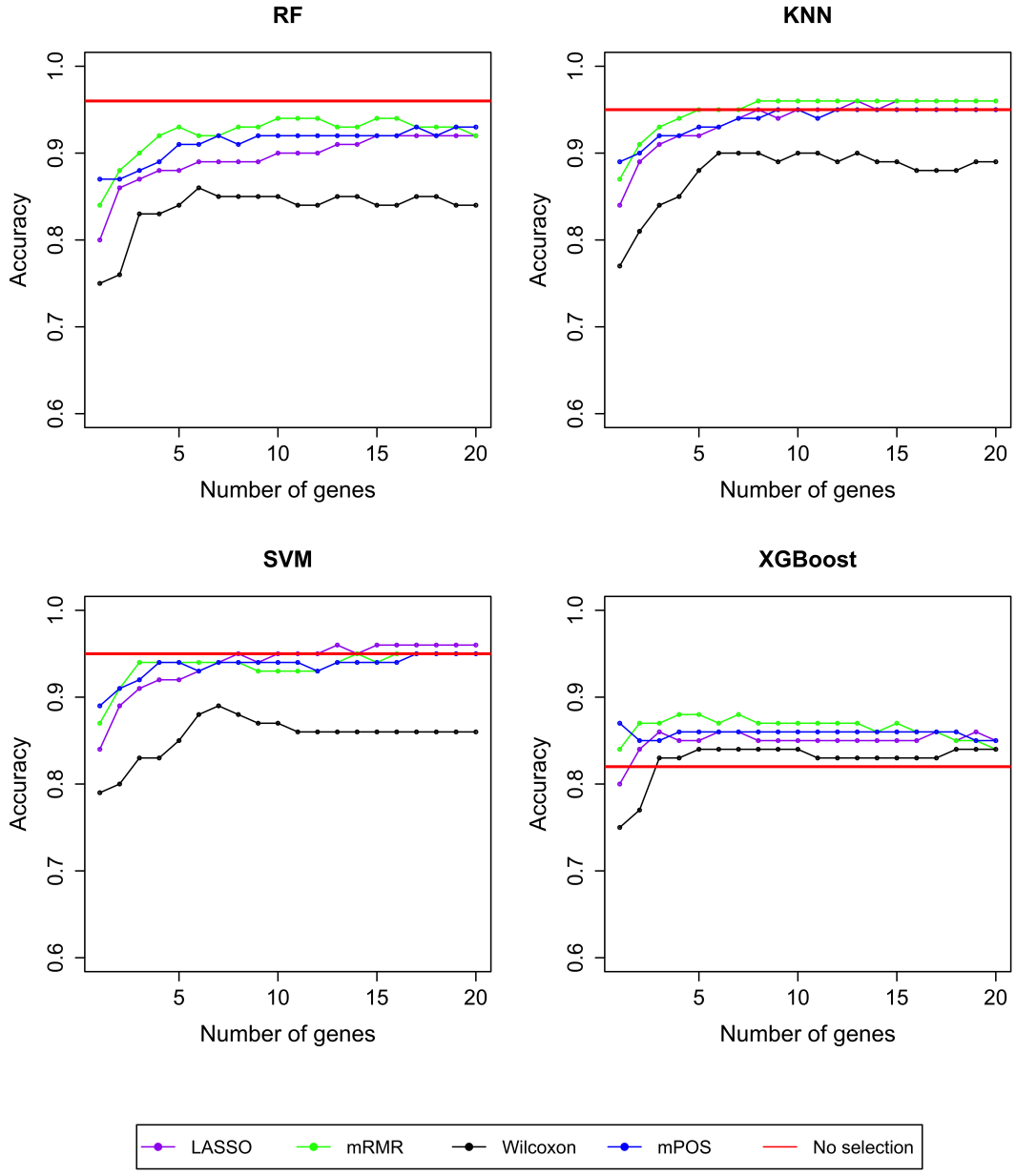

Figure 7.11: Average of classification accuracy for GSE24514 dataset. Average classification accuracy for GSE24514 data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Wilcoxon, mPOS, and the full set of feature.

Figure 7.12 shows the average classification accuracies on the GSE4045 dataset using RF, KNN, SVM, and XGBoost classifiers. It demonstrates that mPOS performs better than other techniques at a single informative gene using KNN and SVM classifiers. The mRMR outperforms all other feature selection methods at the small and moderate set sizes of informative genes across four classifies. Wilcoxon is the best method for the large set sizes of informative genes across four classifiers.

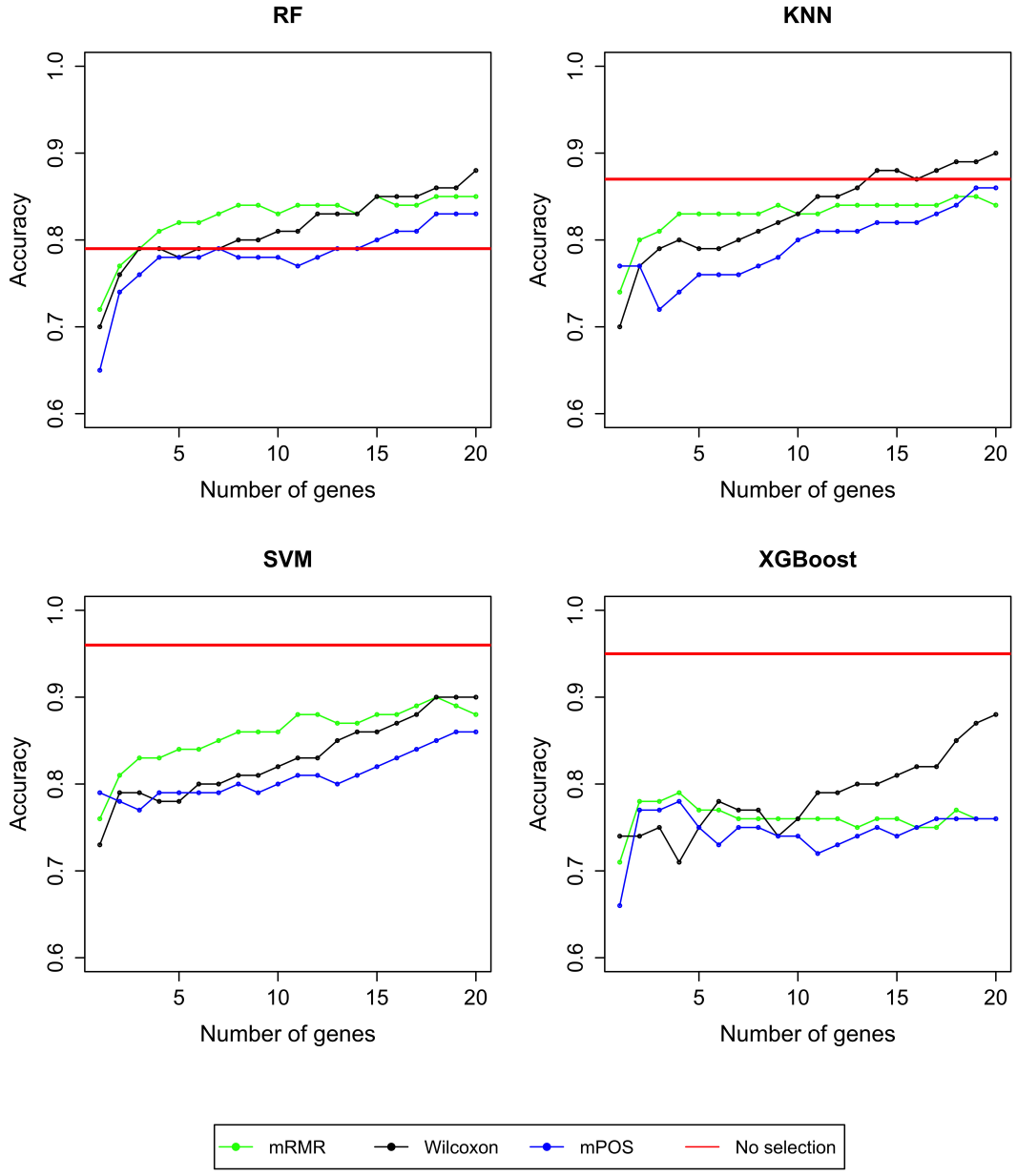

Figure 7.12: Average of classification accuracy for GSE4045 dataset. Average classification accuracy for GSE4045 data based on 20 repetitions of 5-fold CV using mRMR, Wilcoxon, mPOS, and the full set of feature.

Figure 7.13 shows the average classification accuracies on the Leukaemia dataset using RF, KNN, SVM, and XGBoost classifiers. It observes that LASSO outperforms all other techniques at the different set sizes of informative genes using RF, KNN, and XGBoost classifiers. However, mPOS performs better than other feature selection techniques at the moderate and large set sizes of informative genes when evaluating with the SVM classifier.

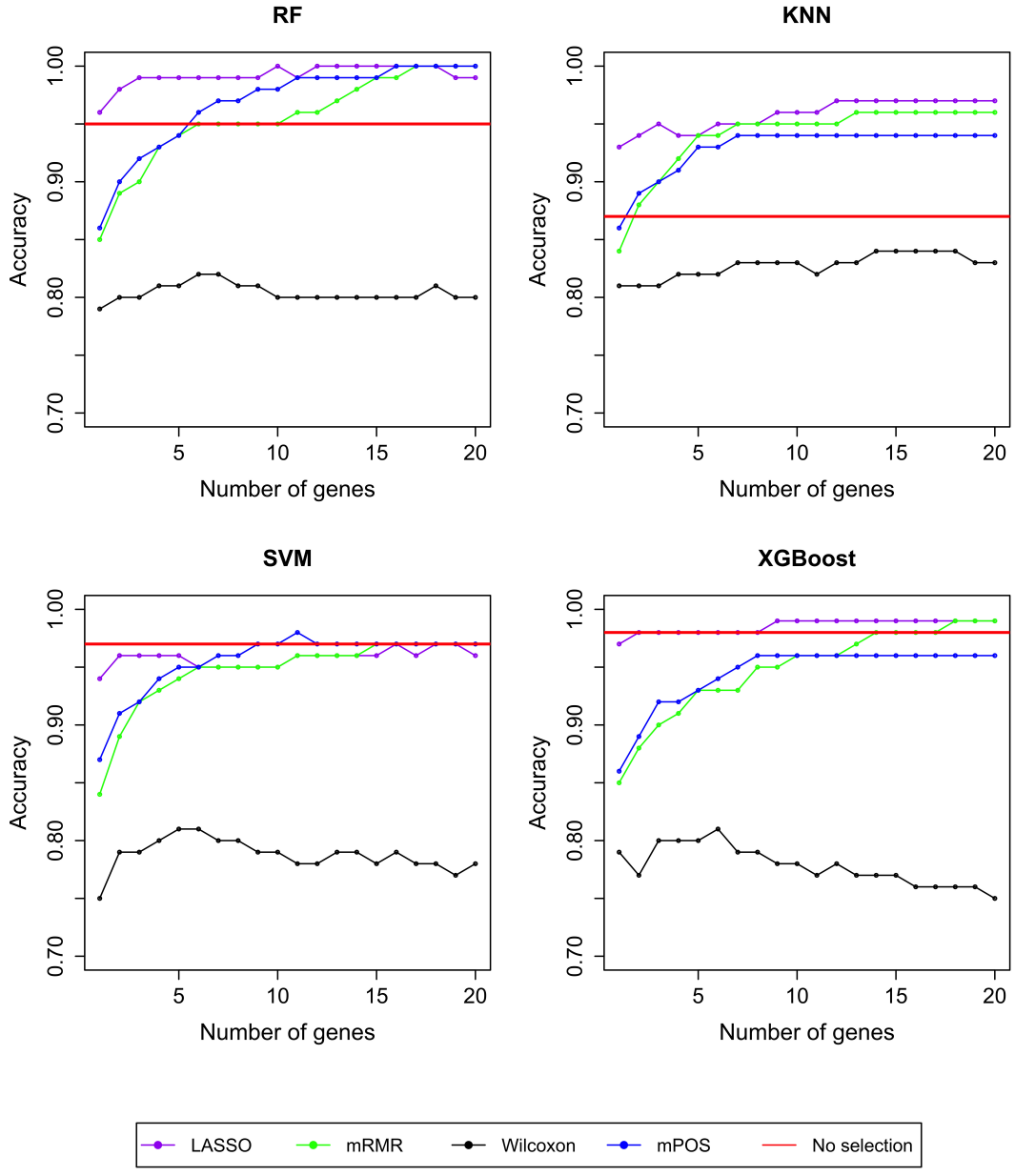

Figure 7.13: Average of classification accuracy for Leukaemia dataset. Average classification accuracy for Leukaemia data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Wilcoxon, mPOS, and the full set of feature.

The average classification accuracies on the Carcinoma dataset using RF, KNN, SVM, and XGBoost classifiers is shown in Figure 7.14. It reveals that mPOS outperforms all other techniques at the moderate and large set sizes of informative genes using RF classifier, while mPOS demonstrates performance comparable to that of LASSO and the mRMR method at larger set sizes of informative genes when evaluated using the KNN classifier. Additionally, LASSO achieves superior performance at small set size of informative genes using SVM classifier. In contrast, LASSO performs better than other feature selection techniques across different set sizes of informative genes when assessed using the XGBoost classifier.

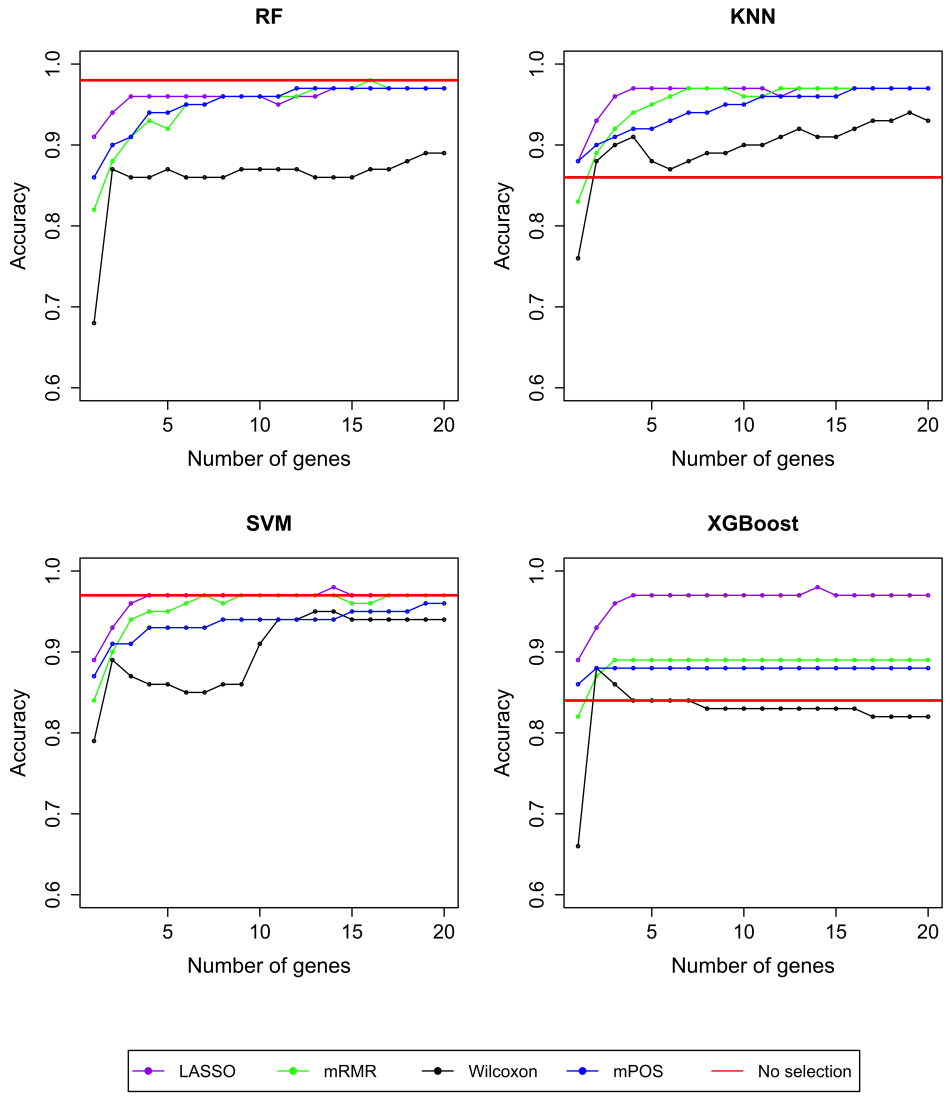

Figure 7.14: Average of classification accuracy for Carcinoma dataset. Average classification accuracy for Carcinoma data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Wilcoxon, mPOS, and the full set of feature.

Figure 7.15 shows the average classification accuracies on the Lung(1) dataset using RF, KNN, SVM, and XGBoost classifiers. It indicates that mPOS achieves comparable performance to mRMR method at moderate and large set sizes of informative genes, except for a set of 20 informative genes, using the RF classifier. mPOS also shows comparable performance to LASSO at moderate and large set sizes of informative genes using the SVM classifier. Furthermore, mPOS demonstrates performance comparable to that of the LASSO and mRMR methods at large set sizes of informative genes when using the KNN classifier, while it is comparable to the mRMR method when using the XGBoost classifier.

Figure 7.16 shows the average classification accuracies on the GSE21029 dataset using RF, KNN, SVM, and XGBoost classifiers. It demonstrates that mPOS provides the best performance using RF, KNN, and XGBoost classifiers, except for a single informative gene. For the SVM classifier, the mPOS outperforms other feature selection techniques at different set sizes of informative genes, except for the set consisting of only 1 or 2 informative genes.

Figure 7.17 shows the average classification accuracies on the GSE22093 dataset using RF, KNN, SVM, and XGBoost classifiers. It reveals that mPOS performs better than other feature selection techniques at the moderate and large set sizes of informative genes, except a set of size of 20 informative genes, using RF classifier. Furthermore, the mPOS provides a better performance at the moderate and large set sizes of informative genes using the XGBoost classifier, excepting the set size of 16 and 17 informative genes. In contrast, the mRMR provides the best performance using KNN classifier with a classification accuracy of 72% and Kruskal outperforms all other feature selection techniques using the SVM classifier.

Figure 7.18 shows the average classification accuracies on the GSE23938 dataset using RF, KNN, SVM, and XGBoost classifiers. It indicates that mPOS is the best feature selection technique at the different set sizes of informative genes using RF, KNN, SVM, and XGBoost classifiers. It also provides a classification accuracy of up to approximately 90%.

Figure 7.19 shows the average classification accuracies on the GSE102079 dataset using RF, KNN, SVM, and XGBoost classifiers. It demonstrates that mPOS is the best at the small set sizes of informative genes using RF, KNN, and XGBoost classifiers. Furthermore, mPOS achieves the best performance at the different set sizes of informative genes using the SVM classifier.

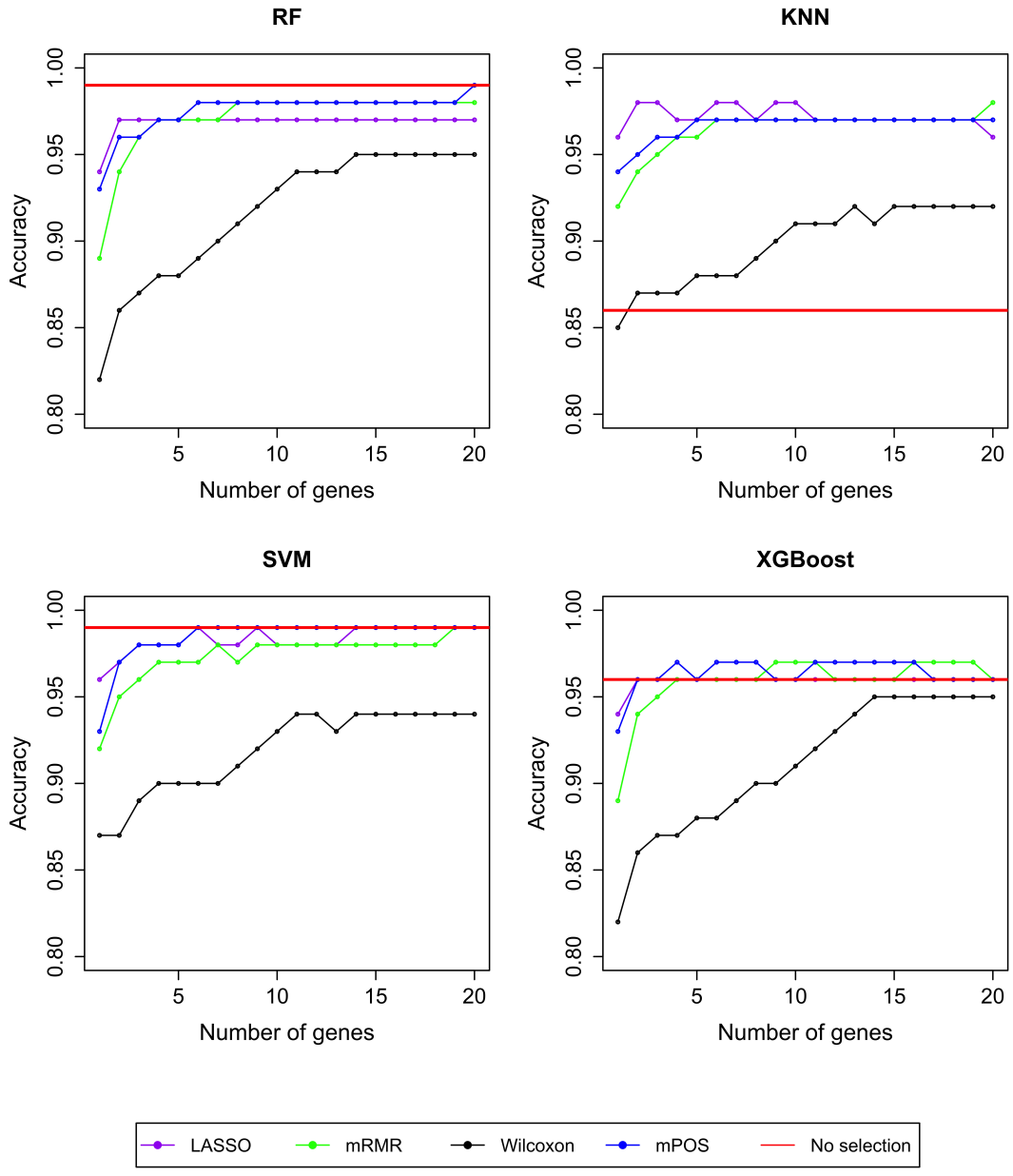

Figure 7.15: Average of classification accuracy for Lung(1) dataset. Average classification accuracy for Lung(1) data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Wilcoxon, mPOS, and the full set of feature.

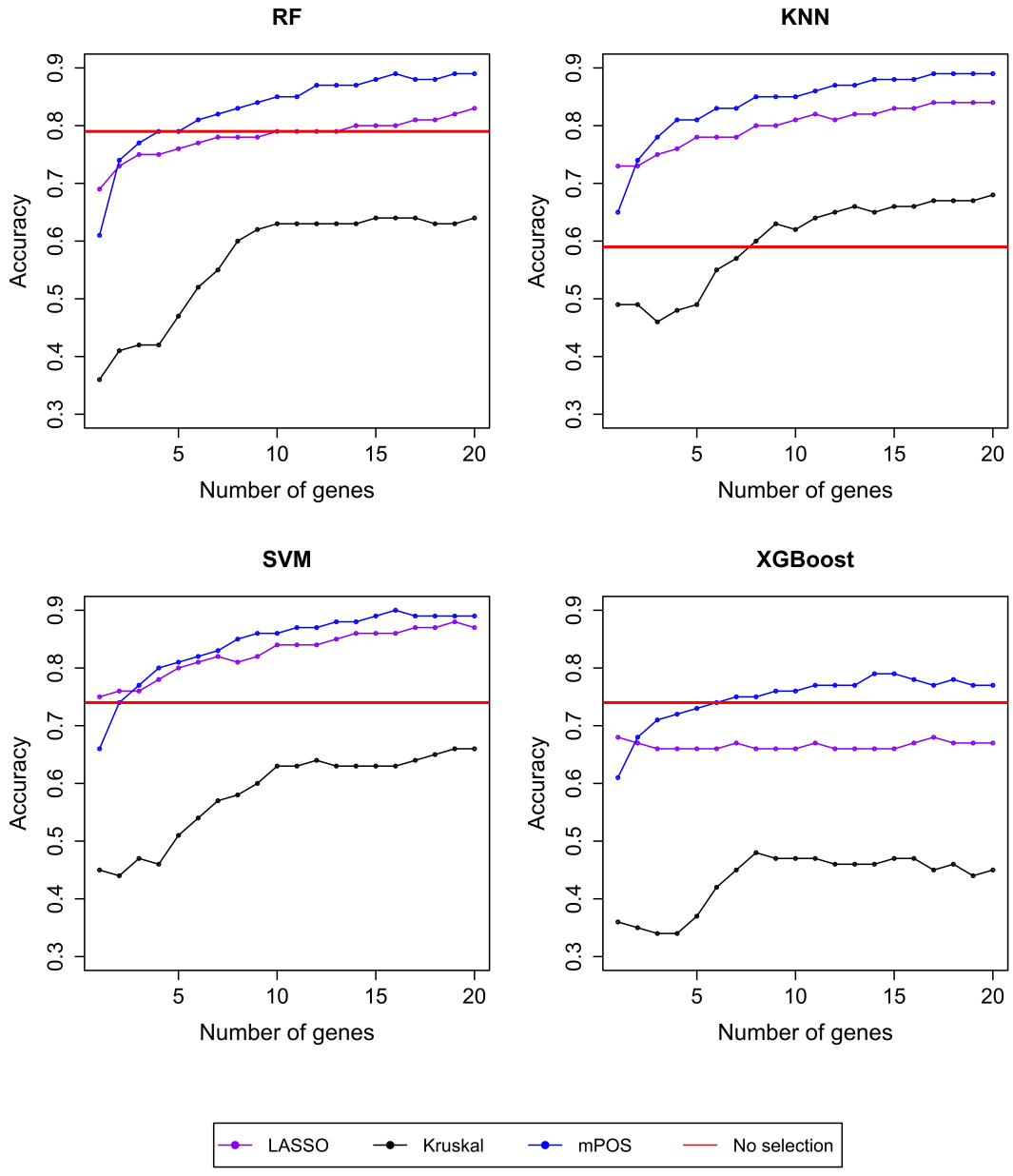

Figure 7.16: Average of classification accuracy for GSE21029 dataset. Average classification accuracy for GSE21029 data based on 20 repetitions of 5-fold CV using LASSO, Kruskal, mPOS, and the full set of feature.

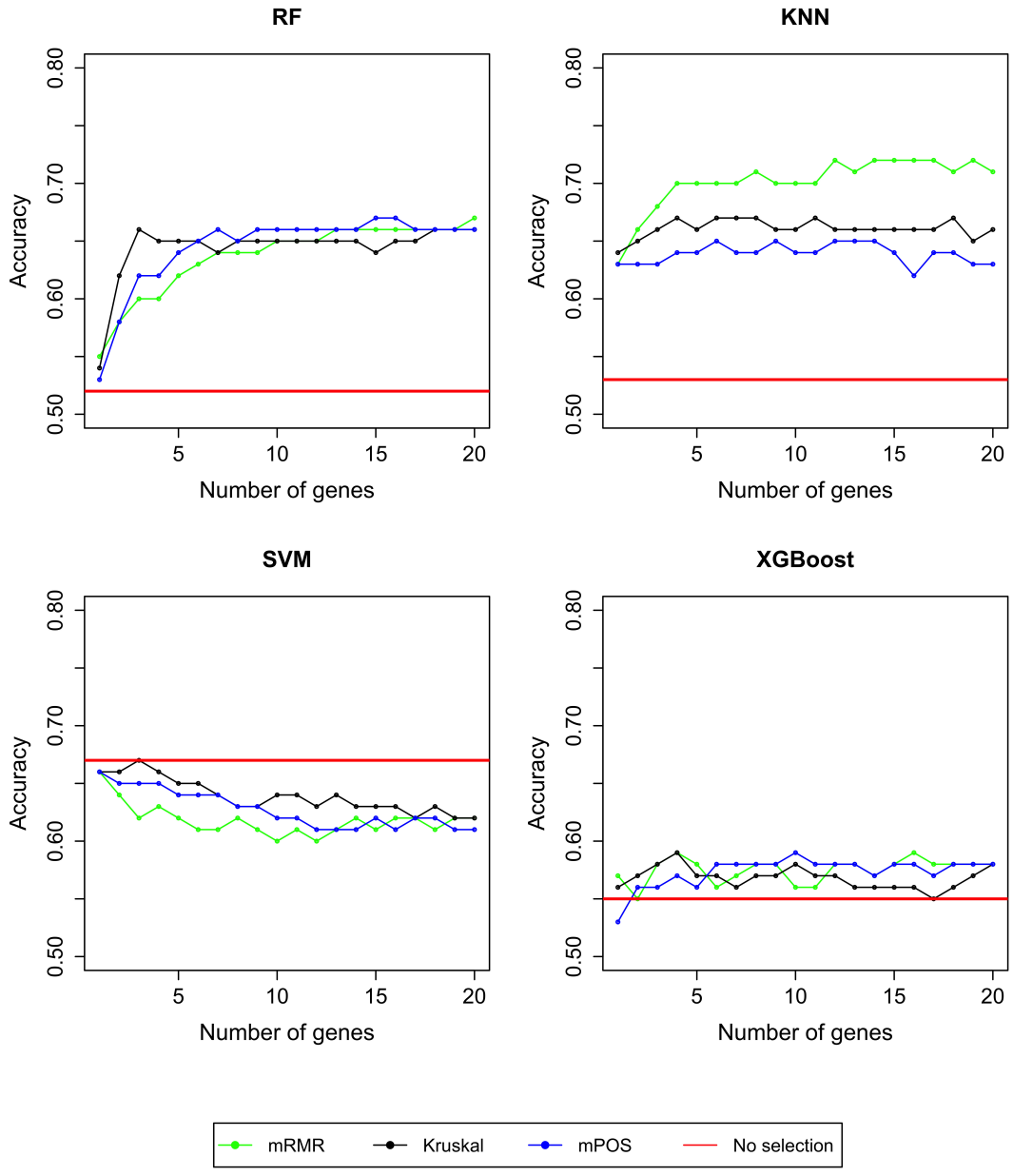

Figure 7.17: Average of classification accuracy for GSE22093 dataset. Average classification accuracy for GSE22093 data based on 20 repetitions of 5-fold CV using mRMR, Kruskal, mPOS, and the full set of features.

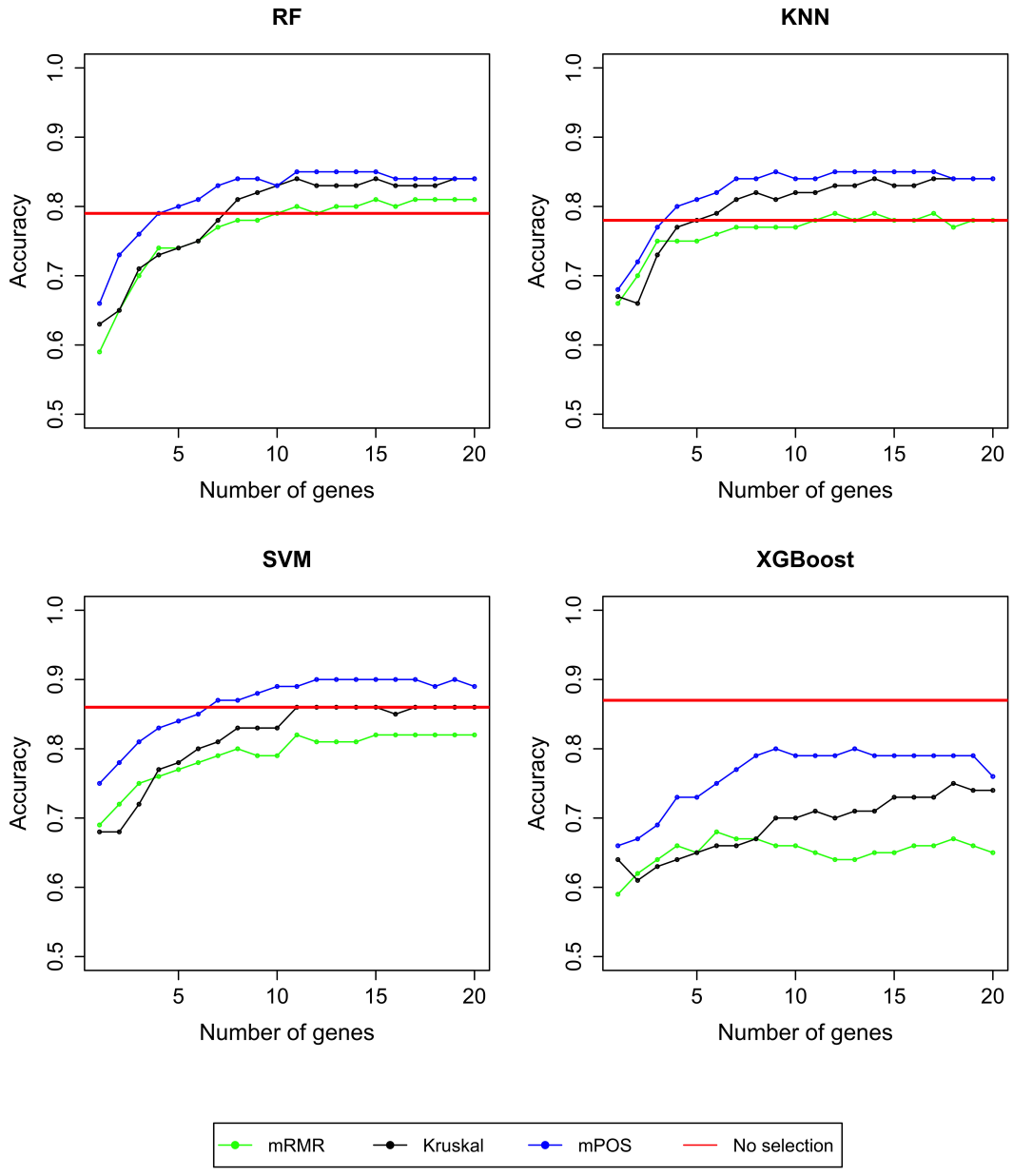

Figure 7.18: Average of classification accuracy for GSE23938 dataset. Average classification accuracy for GSE23938 data based on 20 repetitions of 5-fold CV using mRMR, Kruskal, mPOS, and the full set of Feature.

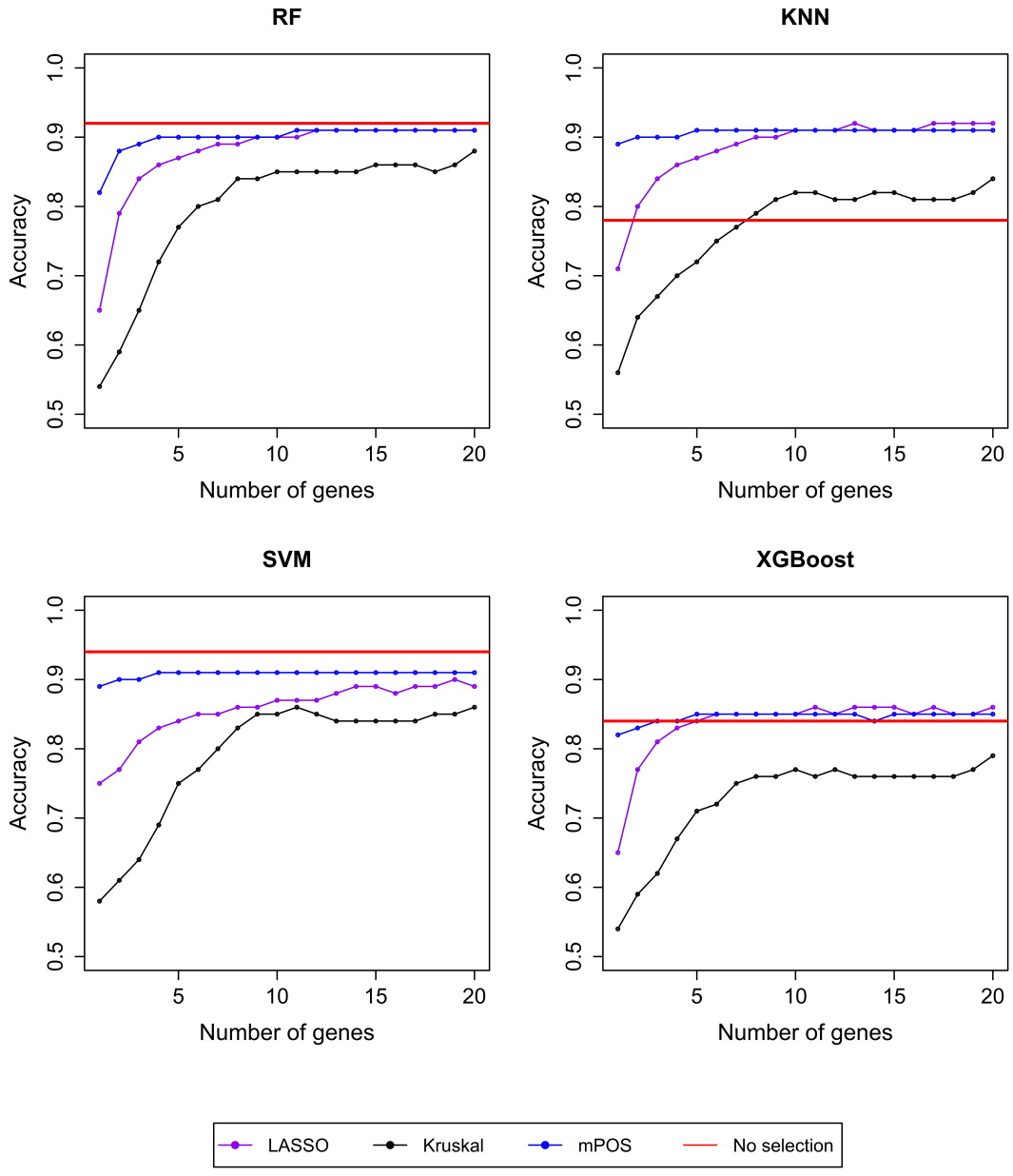

Figure 7.19: Average of classification accuracy for GSE102079 dataset. Average classification accuracy for GSE102079 data based on 20 repetitions of 5-fold CV using LASSO, Kruskal, mPOS, and the full set of features.

The average classification accuracies on the GSE21510 dataset using the RF, KNN, SVM, and XGBoost classifiers are shown in Figure 7.20. It reveals that mPOS performs better than all other feature selection techniques at the small set size of informative genes across four classifiers. mPOS performs comparable to LASSO at the moderate and large set sizes of informative genes using the RF, SVM, and XGBoost classifiers. In contrast, LASSO outperforms all other feature selection techniques at the moderate and large set sizes of informative genes using the KNN classifier.

Figure 7.21 shows the average classification accuracies in the MLL data set using the RF, KNN, SVM and XGBoost classifiers. It shows that mPOS provides comparable performance to the LASSO technique using the RF and XGBoost classifiers. mPOS also performs better than other feature selection methods at the small set size of informative genes using KNN classifier. In contrast, LASSO provides the best performance at the moderate and large set sizes of informative genes using the KNN and SVM classifiers.

Figure 7.22 shows the average classification accuracies in the GSE15852 data set using the RF, KNN, SVM and XGBoost classifiers. The results indicate that mRMR outperforms all other feature selection techniques at the small and moderate set sizes of informative genes using the RF and KNN classifiers. mRMR also performs better than other feature selection techniques at the small set size of informative genes using a SVM classifier. Furthermore, mRMR achieves the best performance at the different set sizes of informative genes using the XGBoost classifier.

The average classification accuracies on the GSE27854(2) dataset using the RF, KNN, SVM, and XGBoost classifiers are shown in Figure 7.23. It demonstrates that mPOS performs better than other feature selection techniques at the small and moderate set sizes of informative genes using the RF and KNN classifiers. mPOS also achieves the best performance at the different set sizes of informative genes using the SVM classifier, while mPOS outperforms all other feature selection techniques at the large set size of informative genes using the XGBoost classifier.

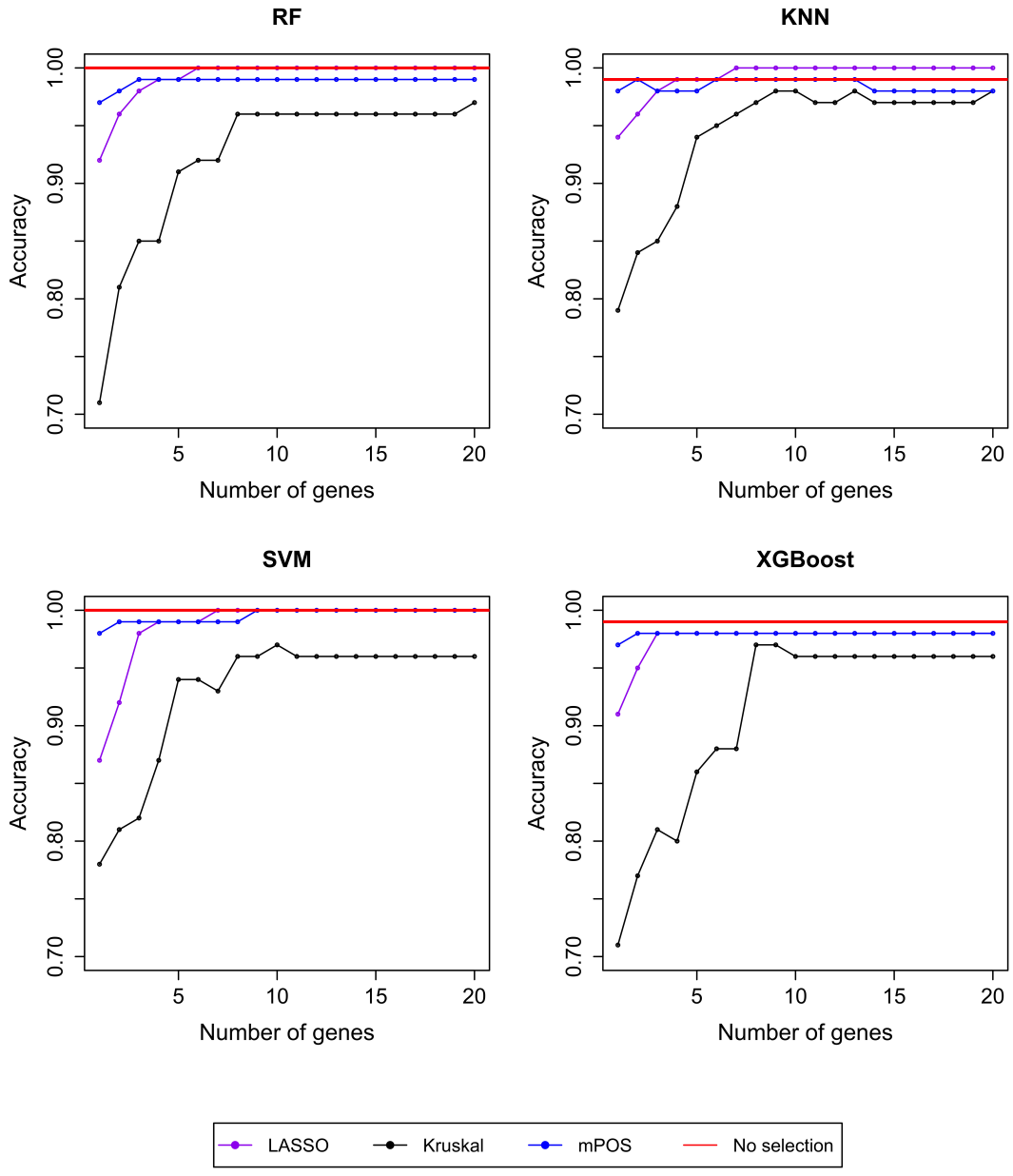

Figure 7.20: Average of classification accuracy for the GSE21510 dataset. Average classification accuracy for GSE21510 data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Kruskal, mPOS, and the full set of features.

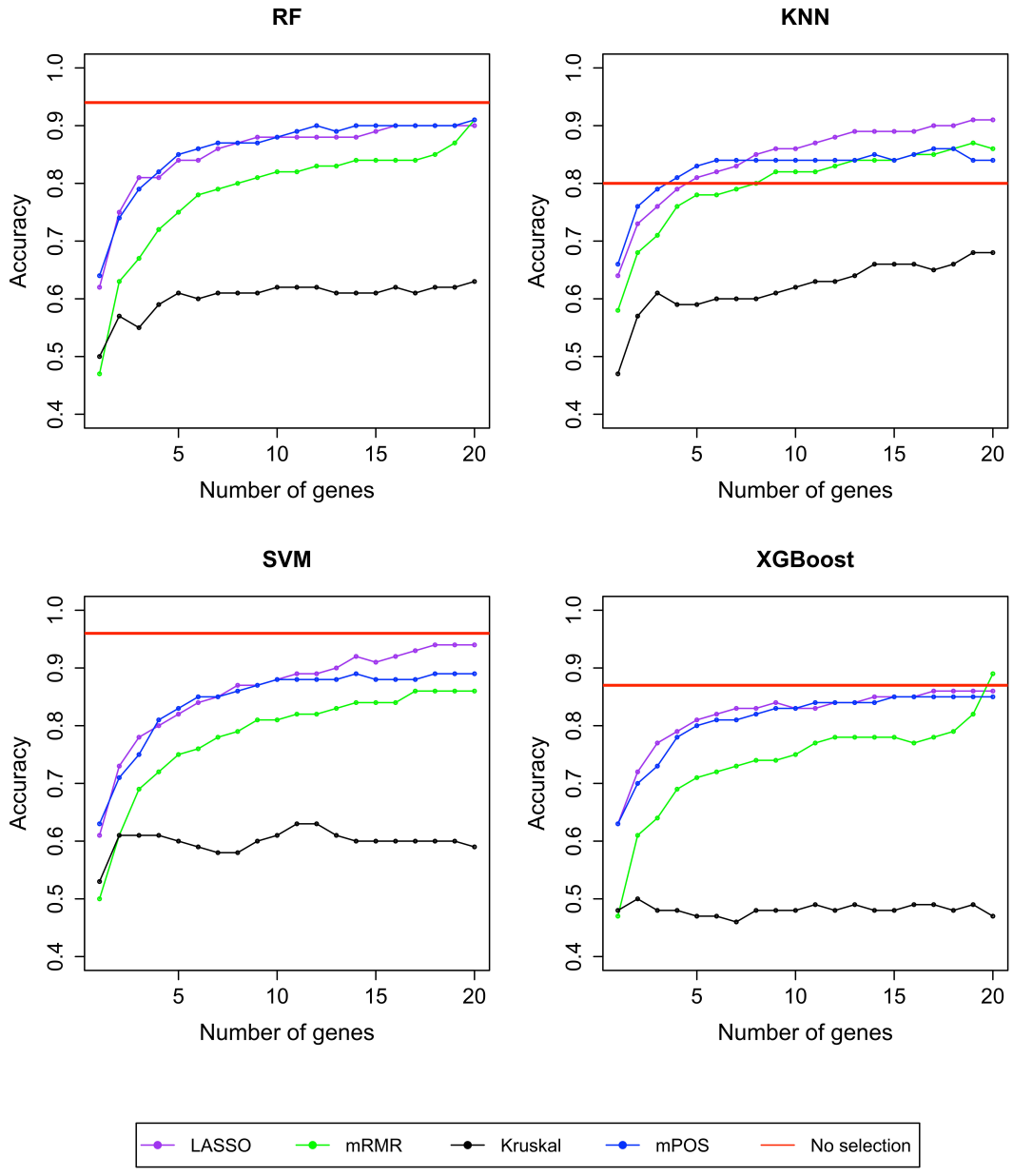

Figure 7.21: Average of classification accuracy for MLL dataset. Average classification accuracy for MLL data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Kruskal, mPOS, and the full set of features.

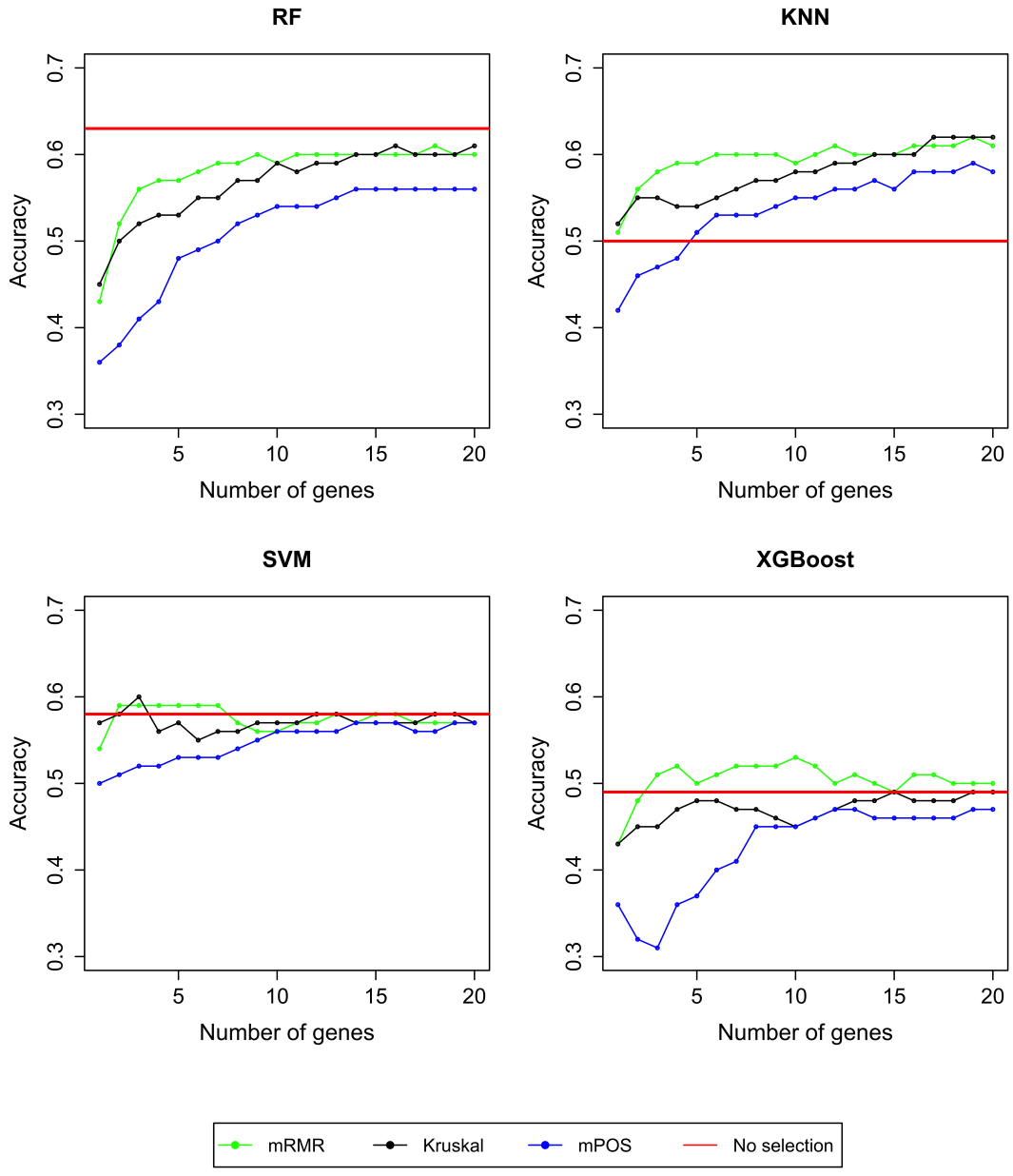

Figure 7.22: Average of classification accuracy for the GSE15852 dataset. Average classification accuracy for GSE15852 data based on 20 repetitions of 5-fold CV using mRMR, Kruskal, mPOS, and the full set of features.

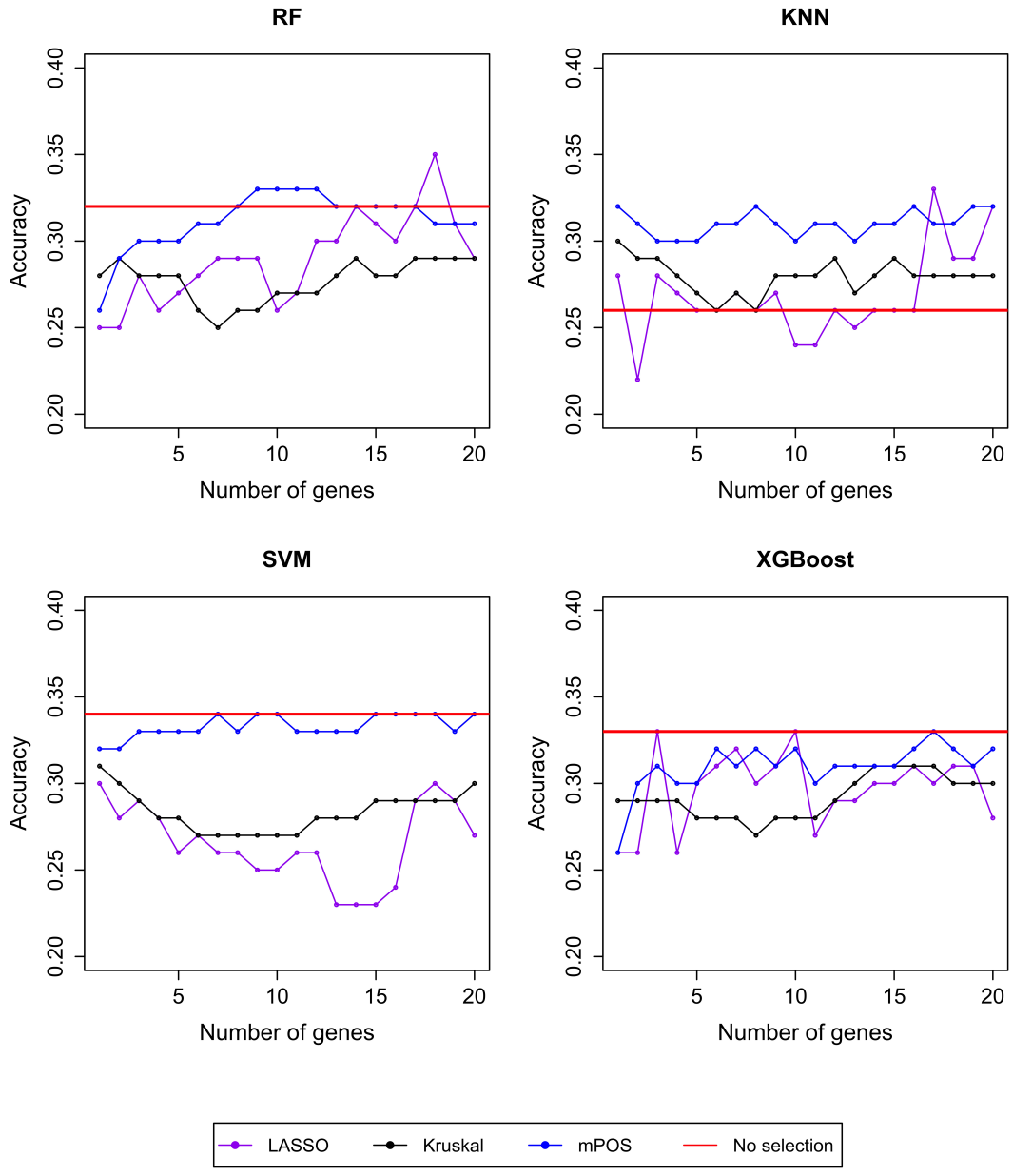

Figure 7.23: Average of classification accuracy for the GSE27854(2) dataset. Average classification accuracy for GSE27854(2) data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Kruskal, mPOS, and the full set of features.

Figure 7.24 shows the average classification accuracies in the GSE27651 data set using the RF, KNN, SVM and XGBoost classifiers. It shows that mPOS outperforms all other feature selection methods at the large set size of informative genes using RF, KNN, and SVM classifiers. However, Kruskal performs better than mPOS at different set sizes of informative genes using XGBoost classifier.

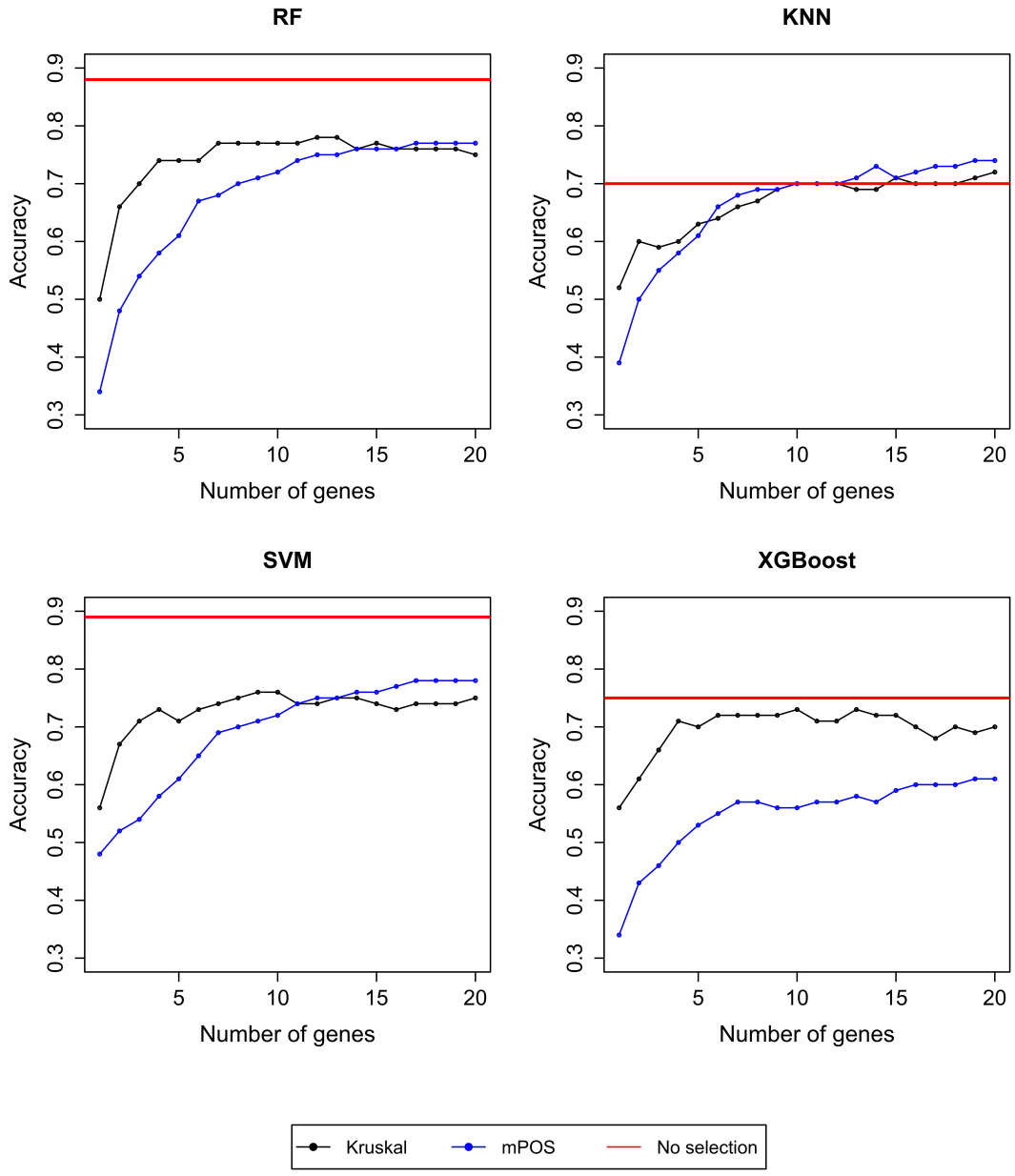

Figure 7.24: Average of classification accuracy for the GSE27651 dataset. Average classification accuracy for GSE27651 data based on 20 repetitions of 5-fold CV using Kruskal, mPOS, and the full set of features.

Figure 7.25 shows the average classification accuracies in the GSE38666 data set using the RF, KNN, SVM and XGBoost classifiers. It shows that mPOS achieves the best performance at the different set sizes of informative genes using RF, KNN, SVM, and XGBoost classifiers.

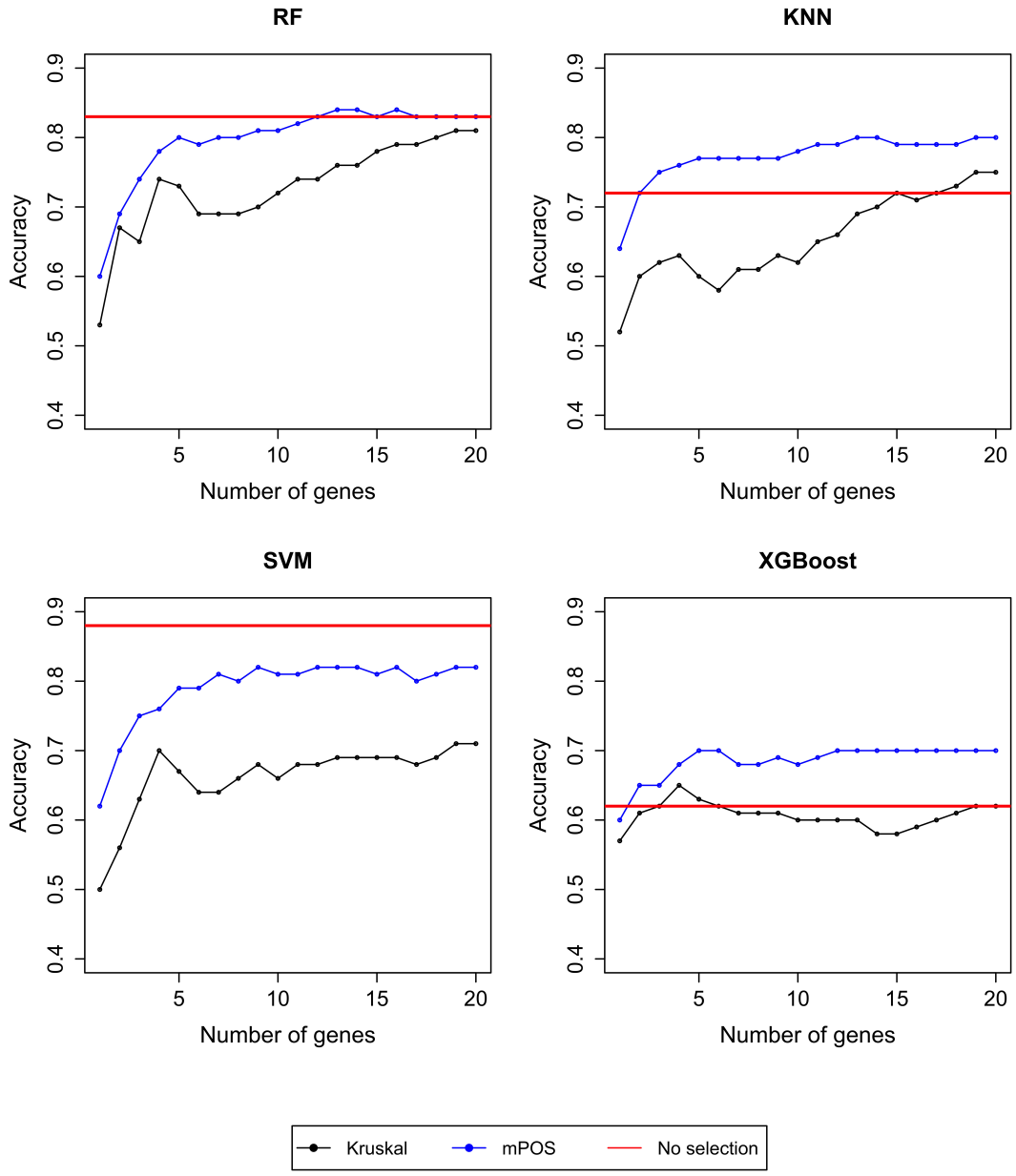

Figure 7.25: Average of classification accuracy for the GSE38666 dataset. Average classification accuracy for GSE38666 data based on 20 repetitions of 5-fold CV using Kruskal, mPOS, and the full set of features.

The average classification accuracies in the GSE40595(2) data set using the RF, KNN, SVM and XGBoost classifiers is presented in Figure 7.26. It shows that mPOS is the best feature selection technique across four classifiers, achieving the highest classification accuracy up to 96%.

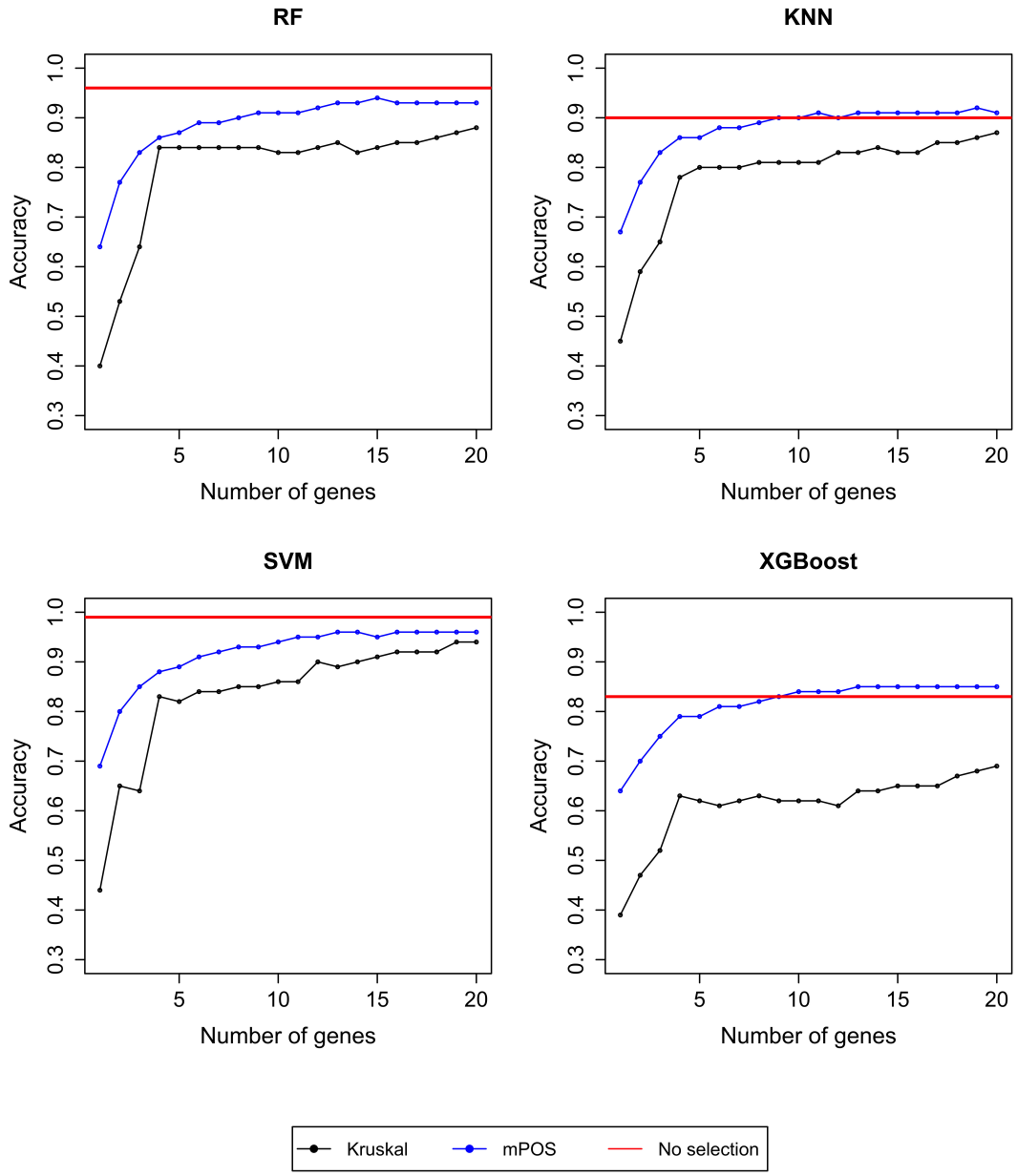

Figure 7.26: Average of classification accuracy for the GSE40595(2) dataset. Average classification accuracy for GSE40595(2) data based on 20 repetitions of 5-fold CV using Kruskal, mPOS, and the full set of Features.

Figure 7.27 demonstrates the average classification accuracies in the Srbct data set using the RF, KNN, SVM and XGBoost classifiers. It reveals that LASSO achieves the best performance at moderate and large set sizes of informative genes. However, it demonstrates performance comparable to that of the mPOS method at small set sizes of informative genes using RF, KNN, and SVM classifiers. Nevertheless, mPOS outperforms all other feature selection techniques at large set sizes of informative genes using XGBoost classifier.

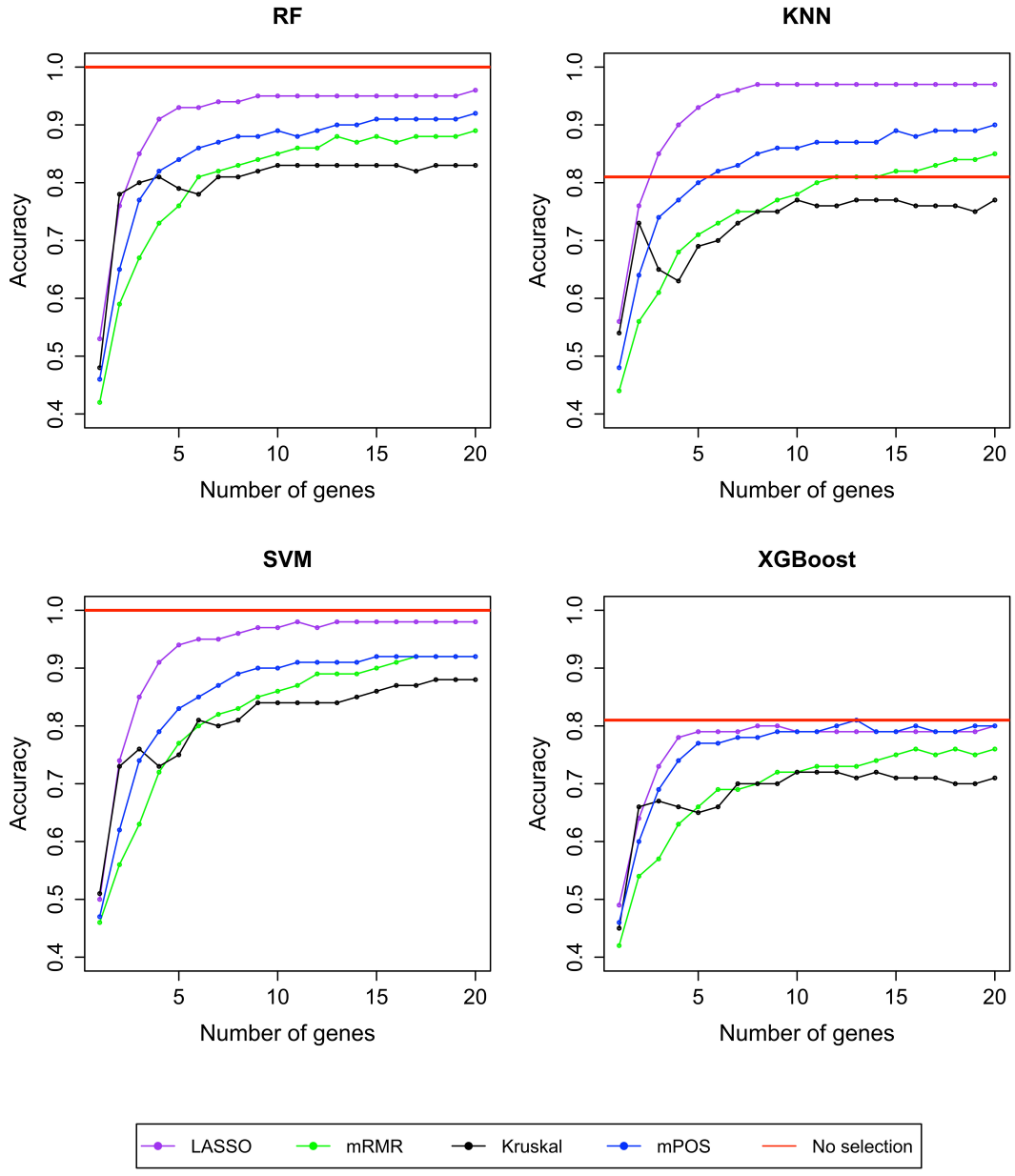

Figure 7.27: Average of classification accuracy for Srbct dataset. Average classification accuracy for Srbct data based on 20 repetitions of 5-fold CV using LASSO, mRMR, Kruskal, mPOS, and the full set of features.

Figure 7.28 demonstrates the average classification accuracies in the GSE162228(2) data set using the RF, KNN, SVM and XGBoost classifiers. The results indicate that Kruskal performs better than mPOS at the different set sizes of informative genes across RF, KNN, and XGBoost. However, mPOS provides the best performance at a single informative gene and set of 3-5 informative genes using SVM classifier.

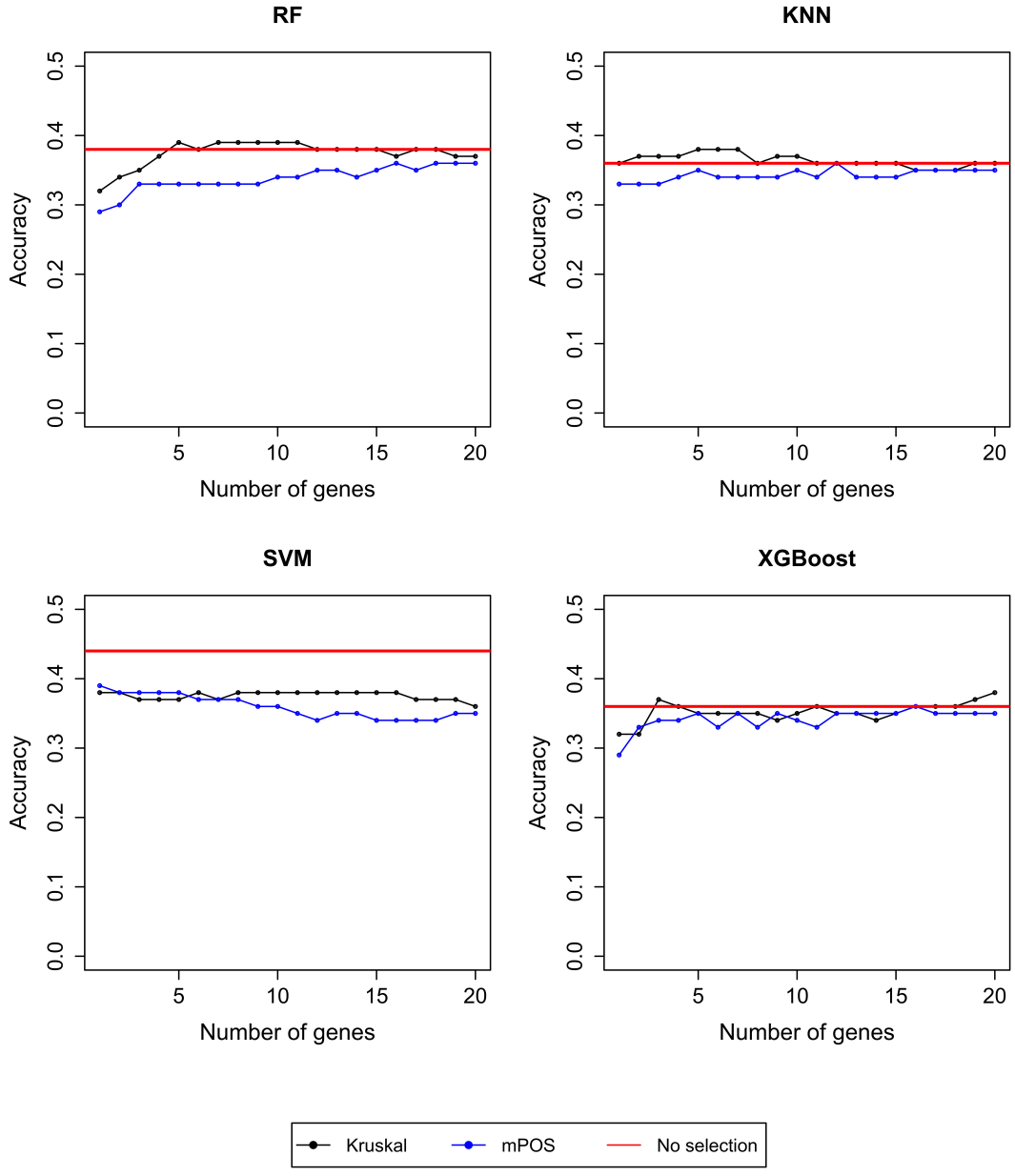

Figure 7.28: Average of classification accuracy for the GSE162228(2) dataset. Average classification accuracy for GSE162228(2) data based on 20 repetitions of 5-fold CV using Kruskal, mPOS, and the full set of features.

Figure 7.29 demonstrates the average classification accuracies in the Brain Tumour data set using the RF, KNN, SVM and XGBoost classifiers. Although Kruskal performs better than all other feature selections in the large set sizes of informative genes using RF, KNN, and SVM classifiers, mPOS outperforms all other feature selection methods at the moderate set size of informative genes using SVM classifier. In contrast, mRMR achieves the best performance at the different set sizes of informative genes using the XGBoost classifier.

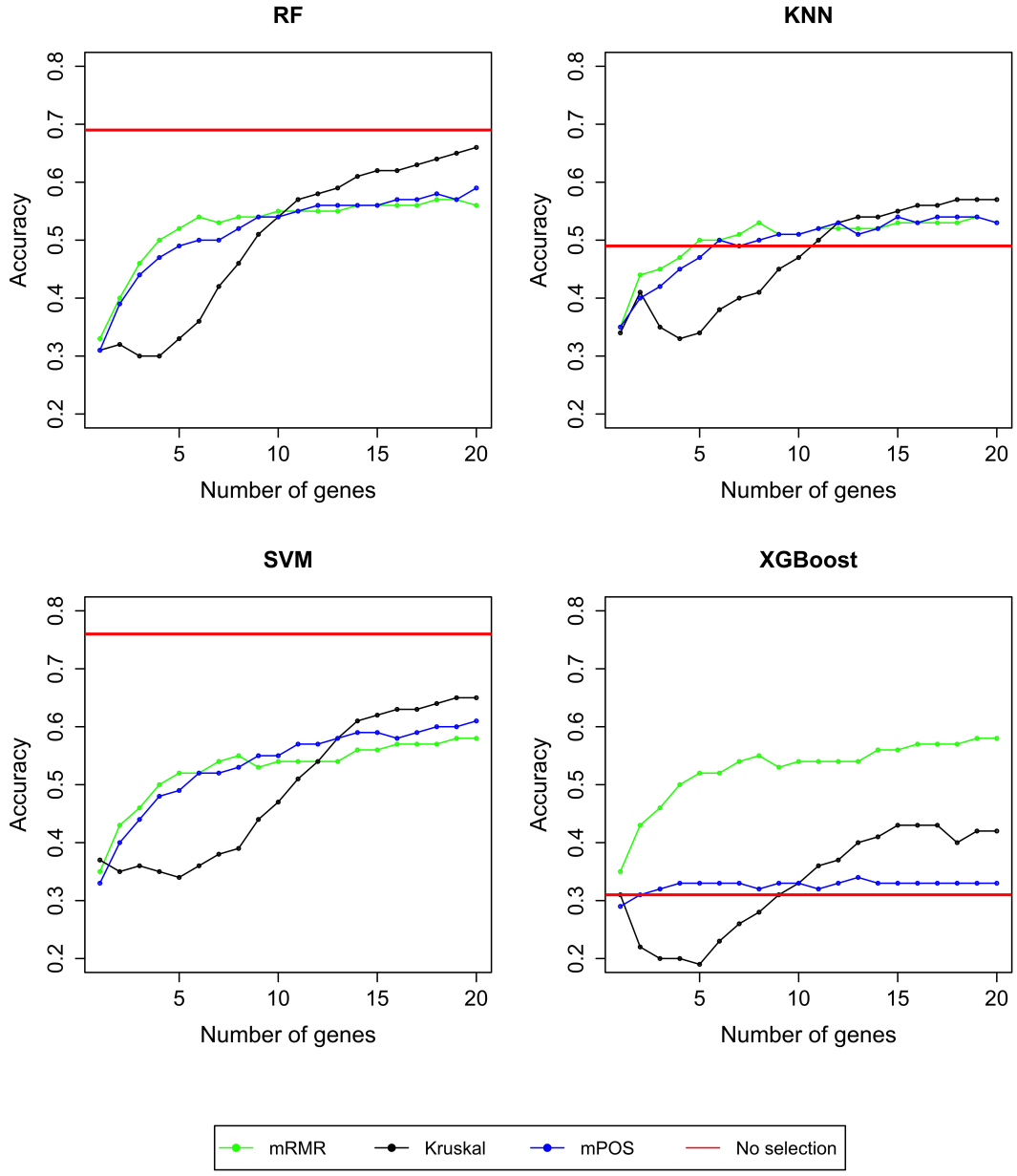

Figure 7.29: Average classification accuracy for Brain Tumour dataset. Average classification accuracy for Brain Tumour data based on 20 repetitions of 5-fold CV using mRMR, Kruskal, mPOS, and the full set of features.

Figure 7.30 demonstrates the average classification accuracies in the Lung(2) data set using the RF, KNN, SVM and XGBoost classifiers. It shows that mRMR outperforms all other feature selection techniques at the moderate and large set sizes of informative genes using the RF, KNN, and SVM classifiers. Furthermore, mRMR performs best at the different set sizes of informative genes using a XGBoost classifier.

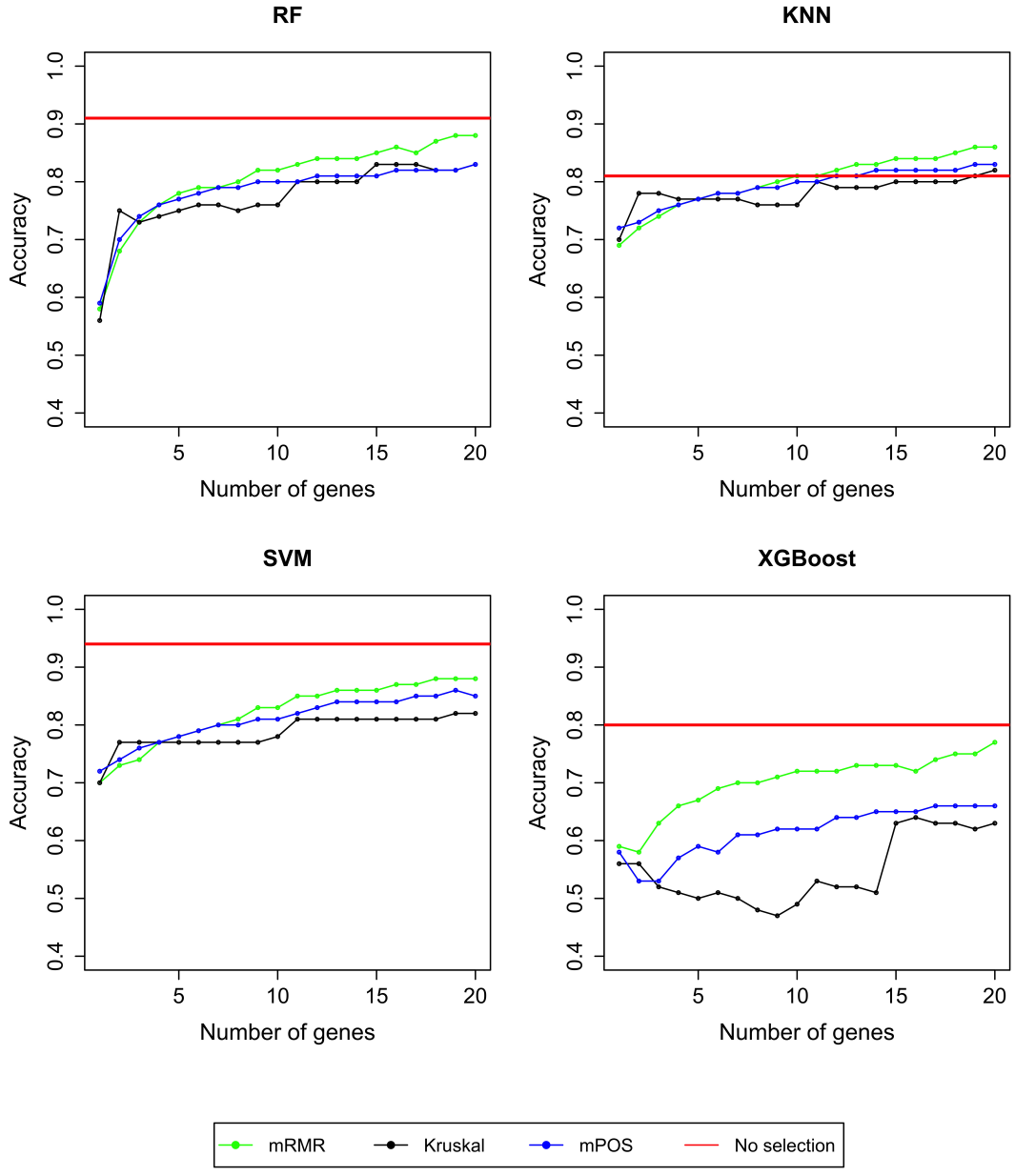

Figure 7.30: Average of classification accuracy for Lung(2) dataset. Average classification accuracy for Lung(2) data based on 20 repetitions of 5-fold CV using mRMR, Kruskal, mPOS, and the full set of features.

Considering classification accuracy alone is insufficient to provide a clear picture of model performance. We further compared the highest classification accuracies achieved by each method to highlight a comprehensive comparison of the performance of the methods relative to mPOS. This aims at providing a clearer comparison of the methods, and a deeper understanding of the strengths and restrictions of the method under consideration. We employ the following criteria to validate the performance of feature selection algorithms: when two or more algorithms achieve comparable classification accuracy, the algorithm that selects the fewest features is considered the best, as it enables a simpler model and more efficient classification. A similar comparison scheme is performed in [123, 136, 183].

Tables 7.1 and 7.2 demonstrates the highest accuracy achieved at different gene set sizes for each feature selection method across RF, kNN, SVM, and XGBoost classifiers. Each row displays the gene set size (along with its corresponding maximum classification accuracy, shown in brackets) obtained by all methods for a specific dataset, as reported in the first column. Additionally, the classification accuracies for the corresponding classifier using the full set of features, without feature selection, are presented in the seventh and last columns of Tables 7.1 and 7.2.

The performance of the compared techniques varies with different gene set sizes, datasets, and classifiers. According to Random Forest (RF), Support Vector Machine (SVM), and Extreme Gradient Boosting (XGBoost) classifiers across separate datasets, mPOS outperforms mRMR, Wilcoxon, Kruskal, while LASSO consistenly achives the highest classification accuracy across majority of datasets using kNN classifier. mPOS also achives the best classification accuracy, especially for small and moderate set sizes of informative features, compared to other feature selection techniques. Consequently, the mPOS feature selection approach is more adaptable to different data patterns and classifier types than the other techniques. mPOS can be implemented with an unlimited number of genes, even when working with small sample sizes, thereby positioning mPOS as a feature selection technique without inherent limitations. In contrast, the performance of the alternative techniques is more sensitive to variations in data characteristics and the choice of classifier.

Table 7.1: The maximum classification accuracies yielded by Random Forest and k-Nearest Neighbor classifiers with feature selection methods along-with the classification accuracy without selection

	RF						k-NN					
Datasets	LASSO	mRMR	Wilcoxon	Kruskal	mPOS	Full set	LASSO	mRMR	Wilcoxon	Kruskal	mPOS	Full set
GSE6861	19(0.53)		14(0.59)		2(0.51)	0.52	19(0.64)		1(0.64)		1(0.55)	0.52
GSE10780	14(0.97)		16(0.92)		11(0.97)	0.97	4(0.97)		17(0.91)		8(0.97)	0.85
GSE19615	19(0.93)		14(0.84)		1(0.87)	0.87	18(0.93)		1(0.86)		1(0.87)	0.87
GSE22513			19(0.84)		2(0.93)	0.86			13(0.82)		6(0.92)	0.69
GSE24514	15(0.92)	10(0.94)	6(0.86)		17(0.93)	0.96	13(0.96)	8(0.96)	6(0.90)		9(0.95)	0.95
GSE4045		17(0.85)	20(0.88)		18(0.83)	0.79		18(0.85)	18(0.89)		19(0.86)	0.87
Leukaemia	10(1.00)	17(1.00)	6(0.82)		16(1.00)	0.95	12(0.97)	13(0.96)	14(0.94)		7(0.94)	0.87
Carcinoma	14(0.97)	16(0.98)	19(0.89)		12(0.97)	0.98	4(0.97)	7(0.97)	19(0.94)		16(0.97)	0.86
Lung(1)	2(0.97)	19(0.88)	15(0.83)		20(0.83)	0.91	2(0.98)	19(0.86)	20(0.82)		19(0.83)	0.81
GSE21029	20(0.83)			15(0.64)	16(0.89)	0.79	17(0.84)			20(0.68)	17(0.89)	0.59
GSE22093		20(0.67)		3(0.66)	15(0.67)	0.52		12(0.72)		4(0.67)	6(0.65)	0.53
GSE23938		15(0.81)		11(0.84)	11(0.85)	0.79		12(0.79)		14(0.84)	9(0.85)	0.78
GSE102079	12(0.91)			20(0.88)	11(0.91)	0.92	13(0.92)			20(0.84)	5(0.91)	0.78
GSE21510	6(1.00)			20(0.97)	2(0.99)	1	7(1.00)			9(0.98)	2(0.99)	0.99
MLL	16(0.90)	20(0.91)		20(0.63)	20(0.91)	0.94	19(0.91)	19(0.87)		19(0.68)	17(0.86)	0.80
GSE15852		18(0.62)		16(0.61)	14(0.56)	0.63		19(0.62)		17(0.62)	19(0.59)	0.50
GSE27854(2)	18(0.35)			2(0.29)	9(0.33)	0.32	18(0.29)			1(0.30)	1(0.32)	0.26
GSE27651				12(0.78)	17(0.77)	0.88				20(0.72)	19(0.74)	0.70
GSE38666				19(0.81)	13(0.84)	0.83				19(0.75)	13(0.80)	0.72
GSE40595(2)				20(0.88)	15(0.94)	0.96				20(0.87)	19(0.92)	0.90
Srbct	20(0.96)	20(0.89)		10(0.83)	20(0.92)	1	7(0.97)	20(0.85)		10(0.77)	20(0.90)	0.81
GSE162228(2)				5(0.39)	16(0.36)	0.38				5(0.38)	12(0.36)	0.36
Brain Tumour		18(0.57)		20(0.66)	20(0.59)	0.69		19(0.54)		18(0.57)	15(0.54)	0.49
Lung(2)		19(0.88)		15(0.83)	20(0.83)	0.91		19(0.86)		20(0.82)	19(0.83)	0.81

The numbers outside brackets represent the size of the gene set that corresponding to the maximum classification accuracy. The boldface numbers in brackets indicate the the highest classification accuracy among the compared methods for the corresponding datasets, while blank spaces indicate where no analysis or implementation was performed.

Table 7.2: The maximum classification accuracies yielded by Support Vector Machine and Extreme Gradient Boost classifiers with feature selection methods along-with the classification accuracy without selection

	SVM						XGBoost					
Datasets	LASSO	mRMR	Wilcoxon	Kruskal	mPOS	Full set	LASSO	mRMR	Wilcoxon	Kruskal	mPOS	Full set
GSE6861	19(0.59)		1(0.65)		1(0.59)	0.56	18(0.62)		1(0.60)		3(0.52)	0.60
GSE10780	11(0.97)		16(0.93)		6(0.97)	0.97	2(0.94)		17(0.89)		6(0.95)	0.95
GSE19615	20(0.91)		1(0.87)		1(0.87)	0.87	19(0.84)		8(0.81)		1(0.87)	0.80
GSE22513			16(0.89)		2(0.93)	0.92			7(0.79)		2(0.92)	0.92
GSE24514	16(0.96)	14(0.95)	7(0.89)		17(0.95)	0.95	3(0.86)	4(0.88)	5(0.84)		1(0.87)	0.82
GSE4045		17(0.89)	19(0.90)		19(0.86)	0.96		4(0.79)	20(0.88)		4(0.78)	0.95
Leukaemia	16(0.97)	15(0.97)	5(0.81)		11(0.98)	0.97	9(0.99)	18(0.99)	6(0.81)		8(0.96)	0.98
Carcinoma	14(0.98)	7(0.97)	13(0.95)		19(0.96)	0.97	14(0.98)	3(0.89)	2(0.88)		2(0.88)	0.84
Lung(1)	6(0.99)	18(0.88)	19(0.82)		19(0.86)	0.94	2(0.96)	20(0.77)	16(0.64)		17(0.66)	0.88
GSE21029	19(0.88)			19(0.66)	16(.090)	0.74	17(0.68)			8(0.48)	14(0.79)	0.74
GSE22093		1(0.66)		3(0.67)	1(0.66)	0.67		16(0.59)		4(0.59)	10(0.59)	0.55
GSE23938		15(0.82)		11(0.86)	12(0.90)	0.86		6(0.68)		18(0.75)	9(0.80)	0.87
GSE102079	19(0.90)			11(0.86)	4(0.91)	0.94	11(0.86)			20(0.79)	5(0.85)	0.84
GSE21510	7(1.00)			10(0.97)	9(1.00)	1	3(0.98)			8(0.97)	2(0.98)	0.99
MLL	18(0.94)	17(0.86)		11(0.63)	14(0.89)	0.96	17(0.86)	20(0.89)		2(0.50)	15(0.85)	0.87
GSE15852		2(0.59)		3(0.60)	14(0.57)	0.58		10(0.53)		15(0.49)	12(0.47)	0.49
GSE27854(2)	1(0.30			1(0.31)	7(0.34)	0.34	10(0.33)			14(0.31)	17(0.33)	0.33
GSE27651				9(0.76)	17(0.78)	0.89				10(0.73)	19(0.61)	0.75
GSE38666				19(0.71)	9(0.81)	0.88				4(0.65)	5(0.70)	0.62
GSE40595(2)				19(0.94)	13(0.96)	0.99				20(0.69)	13(0.85)	0.83
Srbct	11(0.98)	17(0.92)		18(0.88)	15(0.92)	1	20(0.80)	16(0.76)		10(0.72)	13(0.81)	0.81
GSE162228(2)				1(0.38)	1(0.39)	0.44				20(0.38)	16(0.36)	0.36
Brain Tumour		19(0.58)		19(0.65)	20(0.61)	0.76		19(0.58)		15(0.43)	13(0.34)	0.31
Lung(2)		18(0.88)		19(0.82)	19(0.86)	0.94		20(0.77)		16(0.64)	17(0.66)	0.80

The numbers outside brackets represent the size of the gene set that corresponding to the maximum classification accuracy. The boldface numbers in brackets indicate the the highest classification accuracy among the compared methods for the corresponding datasets, while blank spaces indicate where no analysis or implementation was performed.

7.5.2 Performance Analysis for Stability

An effective feature selection method is expected to provide consistent results across multiple dataset sub-samples of the same dataset, particularly in high-dimentional gene expression data where the number of genes measures the small number of samples. Based on biomarker selection, identifying a stable feature subset depends on prioritising biological markers that are consistently selected across multiple analyzes. It is also crucial to take randomly selected features into account. The stability index is used to assess the stability of comparative methods at different set sizes of features, shown in Chapter 5. As presented in Figures 7.31, 7.32, and 7.32, the stability index varies across twenty-four different gene expression datasets, and several feature selection methods. This indicates that stability gets siginificantly influenced by both algorithm designs and dataset characteristics.

For datasets such as GSE6861, GSE19615, GSE22513, and GSE4045 datasets, seen in Figures 7.31 (a), (c), (d) and (f), the mPOS method achieves higher stability across most feature set sizes. Furthermore, For GSE27854(2) dataset, seen in Figures 7.33 (a), mPOS achieves higher stability across most feature set sizes. These datasets are known to be extremely sensitive because of small samples sizes in relation to dimensionality and various degrees of class imbalance (Table 4.3 in Chapter 4). Methods that enhance robustness or reduce sensitivity to noise and sample fluctuations are likely to present higher stability in such settings. The studies of [160] and [98] revealed that high dimensional data with limited samples conducts multiple genes having comparable discriminative power. This causes genes ranking to be fluctuated across different training splits unless stabilising mechanisms are utilised.

In contrast, for GSE10780, GSE24514, Leukaemia, and Carcinoma datasets, Wilcoxon deminstrates superior stability across most feature set sizes as seen in Figures 7.31 (b), (e), (g), and (h). Similarly, for Lung(1) dataset, Wilcoxon achieves higher stability across most feature set sizes, as seen in Figures 7.32 (a). Consistent with these observations, Kruskal shows superior stability across most feature set sizes in the GSE21029, GSE22093, GSE102079, GSE21510, MLL, and GSE15852 datasets, as presented in Figures 7.32 (b), (c), (e), (f), (g), and (h). Furthermore, Krukal achieves greater stability in GSE27651, GSE38666, GSE40595(2), Srbct, GSE162228(2), brain tumors and lung (2) datasets, as illustrated in Figure 7.33 (b) -

(h). The existence of a few highly discriminative genes that reliably distinguish classes across resampled subsets is responsible for this behavior. In clearer or balanced class distribution, univariate filter methods can reliably select the same top-ranked genes. This results in high stability [160]. This phenomenon has been documented in prior gene expression studies, wherein the presence of dominant biomarkers results in highly reproducible univariate feature rankings across varying subsampling realizations [53].

Class imbalance further interacts with stability outcomes. Imbalanced datasets amplify instability because minority-class samples are underrepresented, making feature selection highly sensitive to how these samples are distributed across sub-samples. Feature rankings can be significantly changed by small changes in minority-class composition, especially for algorithms that do not specifically account for imbalance. As studied by [78, 125], Robust or imbalance-aware feature selection techniques yield more consistent feature subsets and are more resistant to these fluctuations. These findings are supported by the observed stability patterns throughout the examined datasets, where more stability is maintained by techniques better suited to managing imbalance and noise.

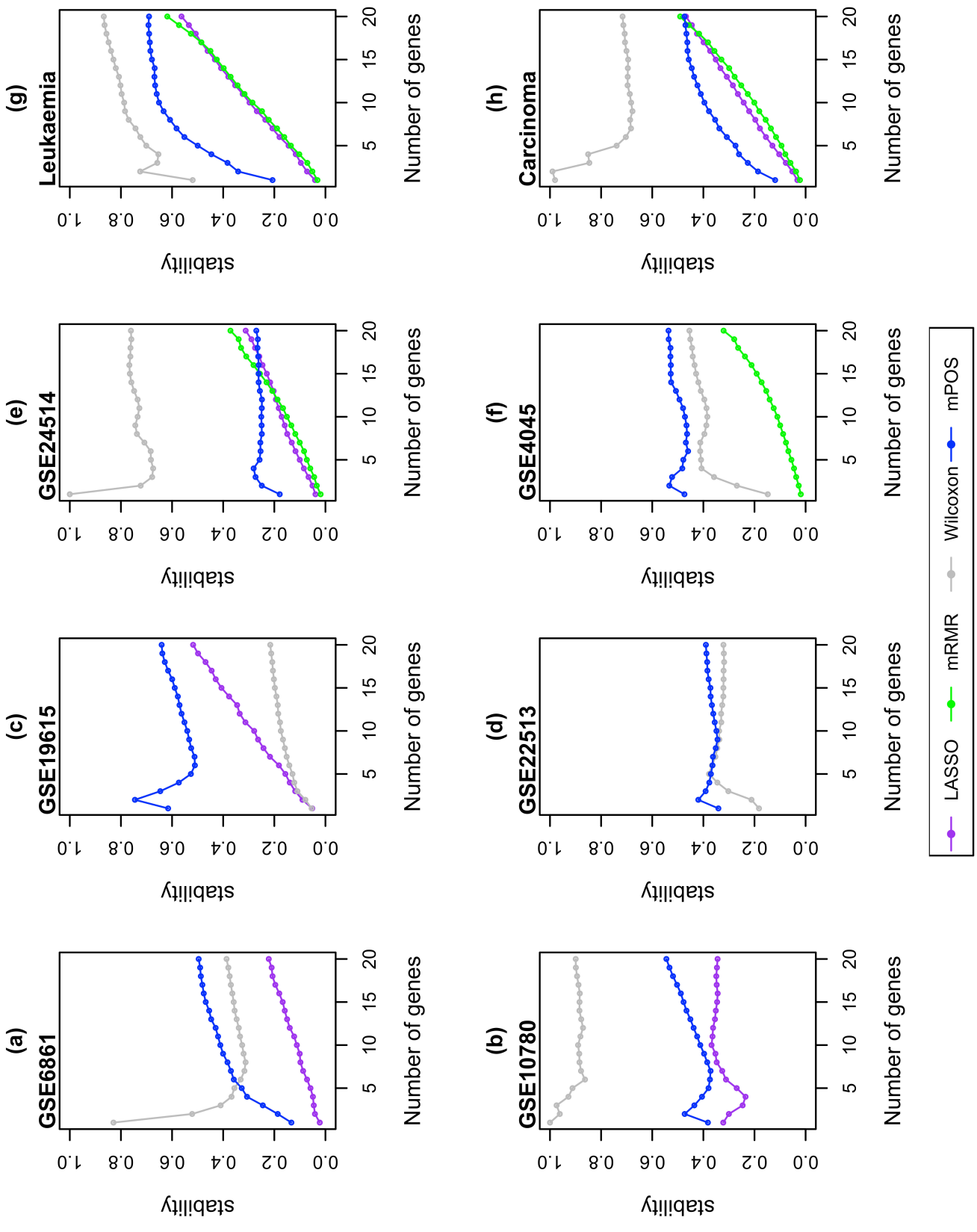

Figure 7.31: Stability scores for 8 datasets at different set sizes that selected by LASSO, mRMR, Wilcoxon, Kruskal, and mPOS: (a) GSE6861 dataset, (b) GSE10780 dataset, (c) GSE19615 dataset, (d) GSE22513 dataset, (e) GSE24514 dataset, (f) GSE4045 dataset, (g) Leukaemia dataset, and (h) Carcinoma dataset.

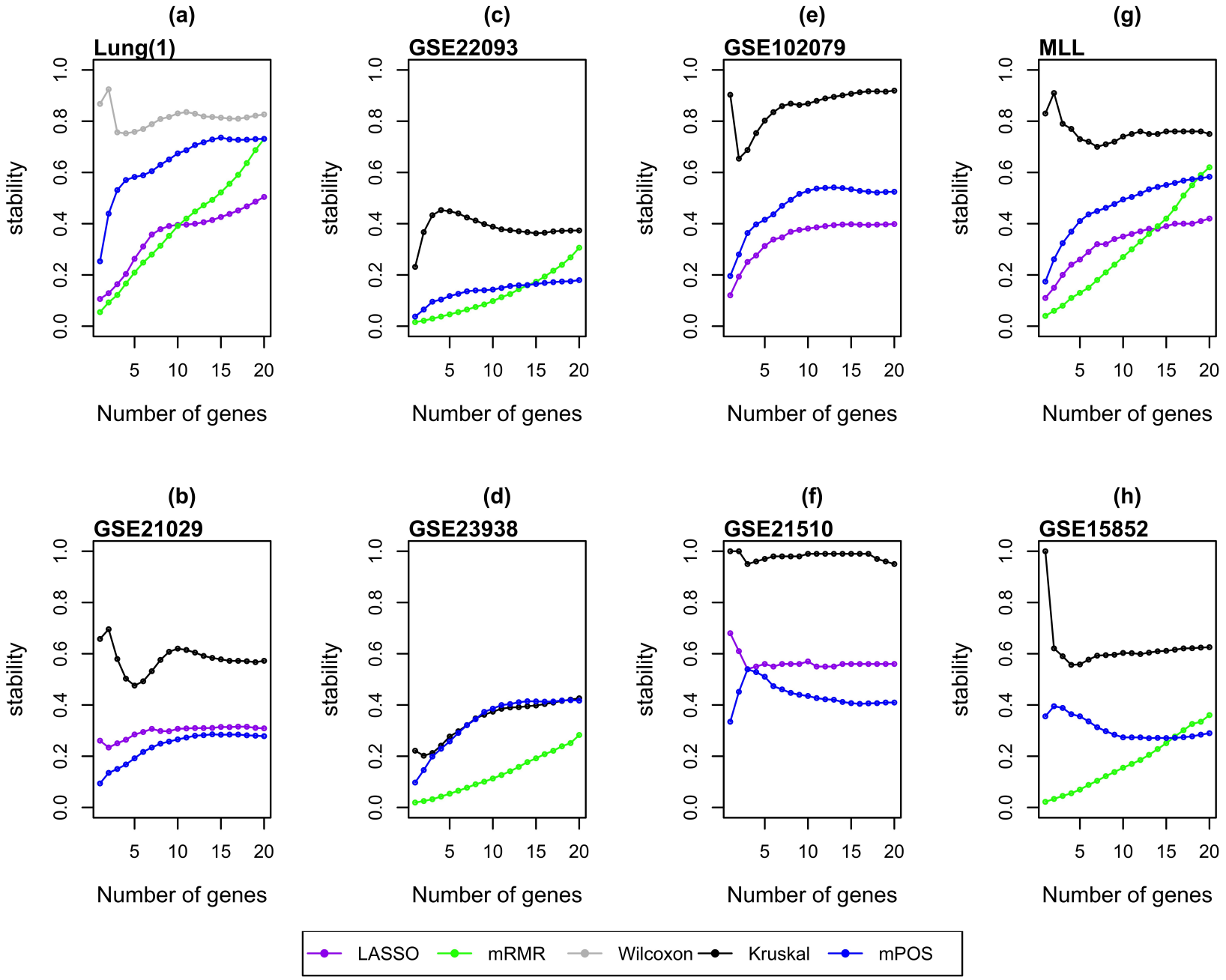

Figure 7.32: Stability scores for 8 datasets at different set sizes that selected by LASSO, mRMR, Wilcoxon, Kruskal, and mPOS: (a) Lung(1) dataset, (b) GSE21029 dataset, (c) GSE22093 dataset, (d) GSE23928 dataset, (e) GSE102079 dataset, (f) GSE21510 dataset, (g) MLL dataset, and (h) GSE15852 dataset.

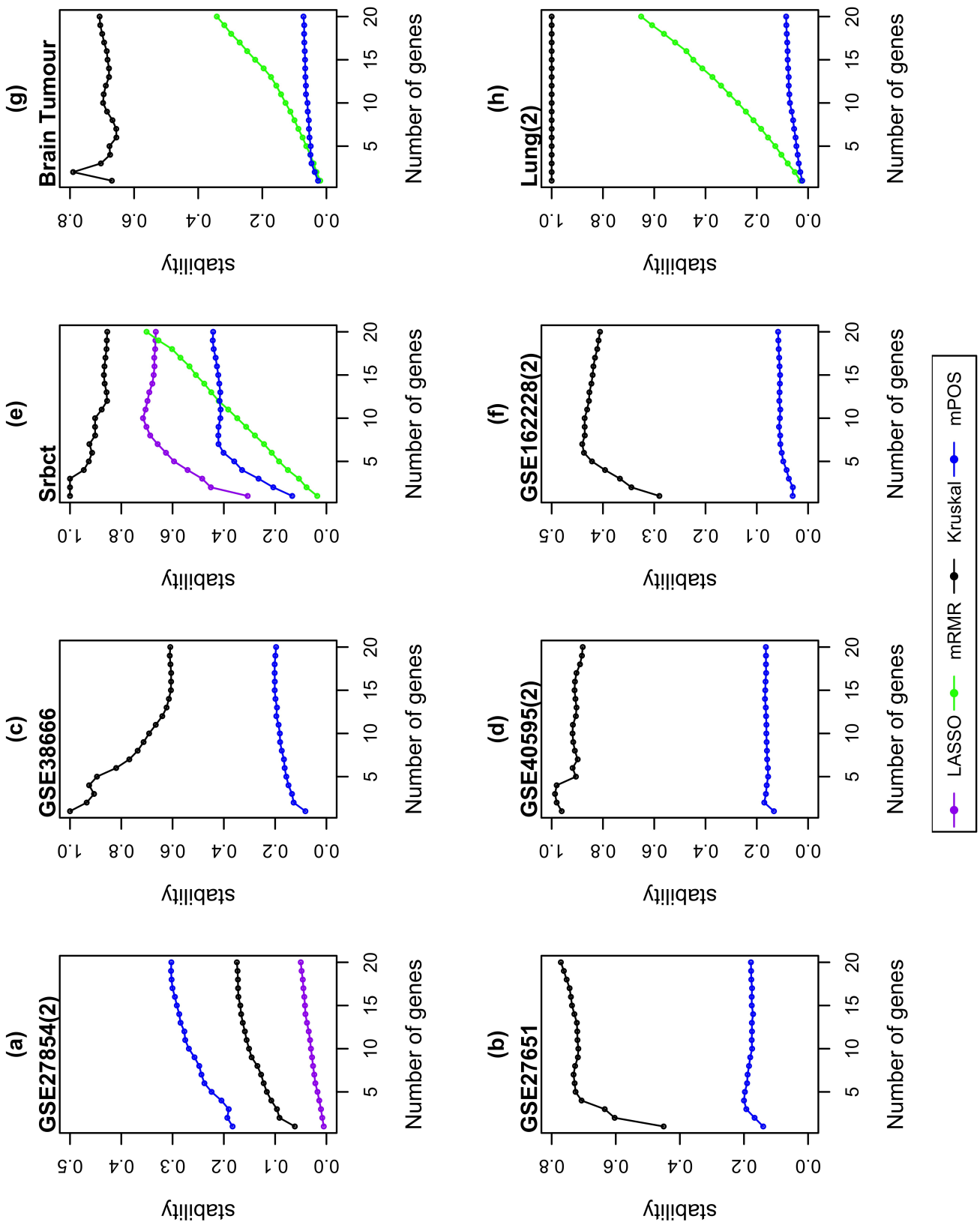

Figure 7.33: Stability scores for 8 datasets at different set sizes that selected by LASSO, mRMR, Kruskal, and mPOS: (a) GSE27854(2) dataset, (b) GSE27651 dataset, (c) GSE38666 dataset, (d) GSE40595(2) dataset, (e) Srbct dataset, (f) GSE162228(2) dataset, (g) Brain Tumour dataset, and (h) Lung(2) dataset.

7.5.3 Performance Analysis for Trade-off between Classification Accuracy and Stability

Although stability in feature selection is significant, it does not inherently ensure the relevance of the selected features to the target-class labels. Therefore, evaluating the predictive performance of a classifier using the selected features is crucial to guarantee their practical effectiveness in classification tasks. The relationship between classification accuracy and stability is considered. The stability scores were combined with the corresponding classification accuracy obtained by the RF, kNN, SVM, and XGBoost classifiers. Different set sizes of selected features are represented by different dots for the same feature selection technique. The optimal method is represented by dots located in the upper right corner of the plot, where stability scores grow along the vertical axis, and classification accuracy increases along the horizontal axis.

For the relationship between accuracy and stability, the entire datasets have been assessed in Figure 7.34 to Figure 7.57.

mPOS achieves a good trade-off between stability score and classification accuracy for GSE10780, GSE22513, Leukaemia, GSE23938 and GSE27854(2) datasets across all classifiers, see Figure 7.35, 7.37, 7.40, 7.45, and 7.50. Both mPOS and the Wilcoxon show comparable performance in achieving a good trade-off between stability and classification accuracy on the GSE4045 and Lung(1) datasets across four classifiers, as demonstrated in Figure 7.39 and 7.42. Our proposed method and the LASSO show comparable performance in providing a good trade-off between stability and classification accuracy on the GSE19615 dataset, see Figure in 7.36, while our proposed method and the mRMR show comparable performance in providing a good trade-off between stability and classification accuracy on the MLL dataset, see Figure in 7.48

In contrast, Wilcoxon generates a good trade-off between stability and classification accuracy on the GSE6861, GSE24514, and Caricinoma datasets across four classifiers, as demonstrated in Figure 7.34, 7.38, and 7.41. Kruskal generates a good trade-off between stability and classification accuracy on the GSE21029, GSE22093, GSE102079, GSE21510, GSE15852, GSE27651, GSE38666, GSE40595(2), Srbct, GSE162228(2), Brain Tumour, and Lung(2)

datasets across four classifiers, as demonstrated in Figure 7.43, 7.44, 7.46, 7.47, 7.49, 7.51, 7.52, 7.53, 7.54, 7.55, 7.56, and 7.57

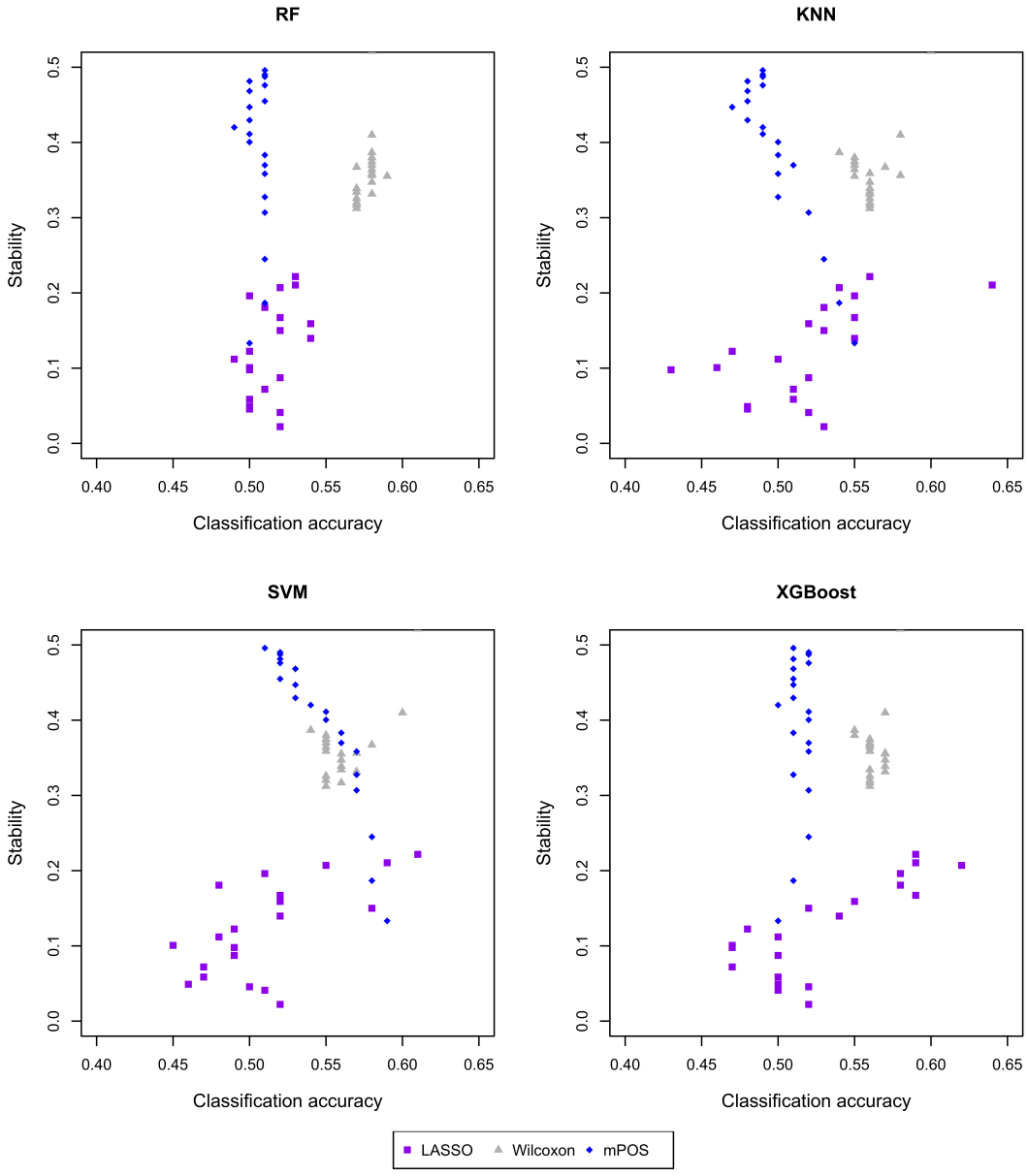

Figure 7.34: Stability - accuracy plot for GSE6861 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE6861 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

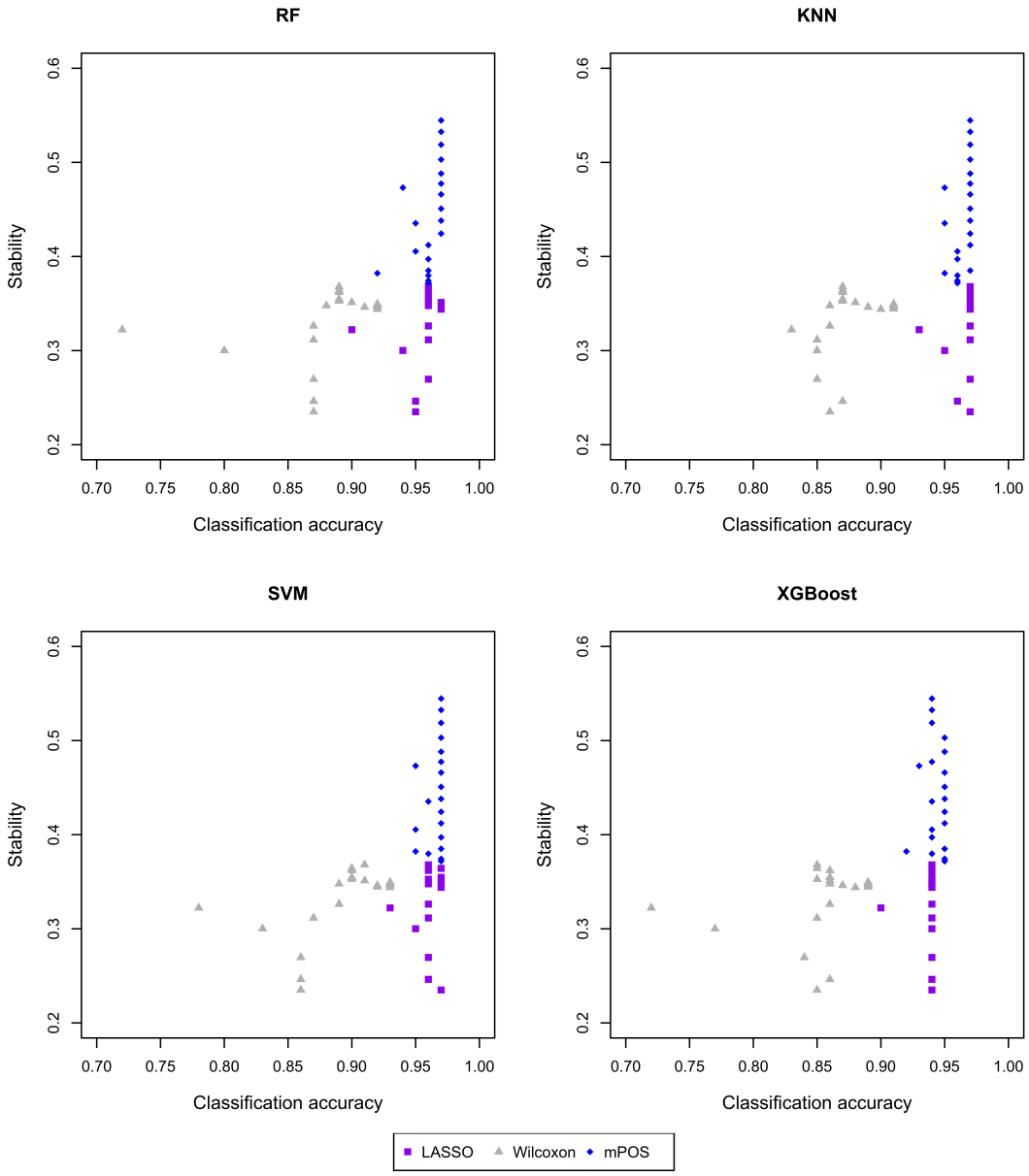

Figure 7.35: Stability - accuracy plot for GSE10780 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE10780 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

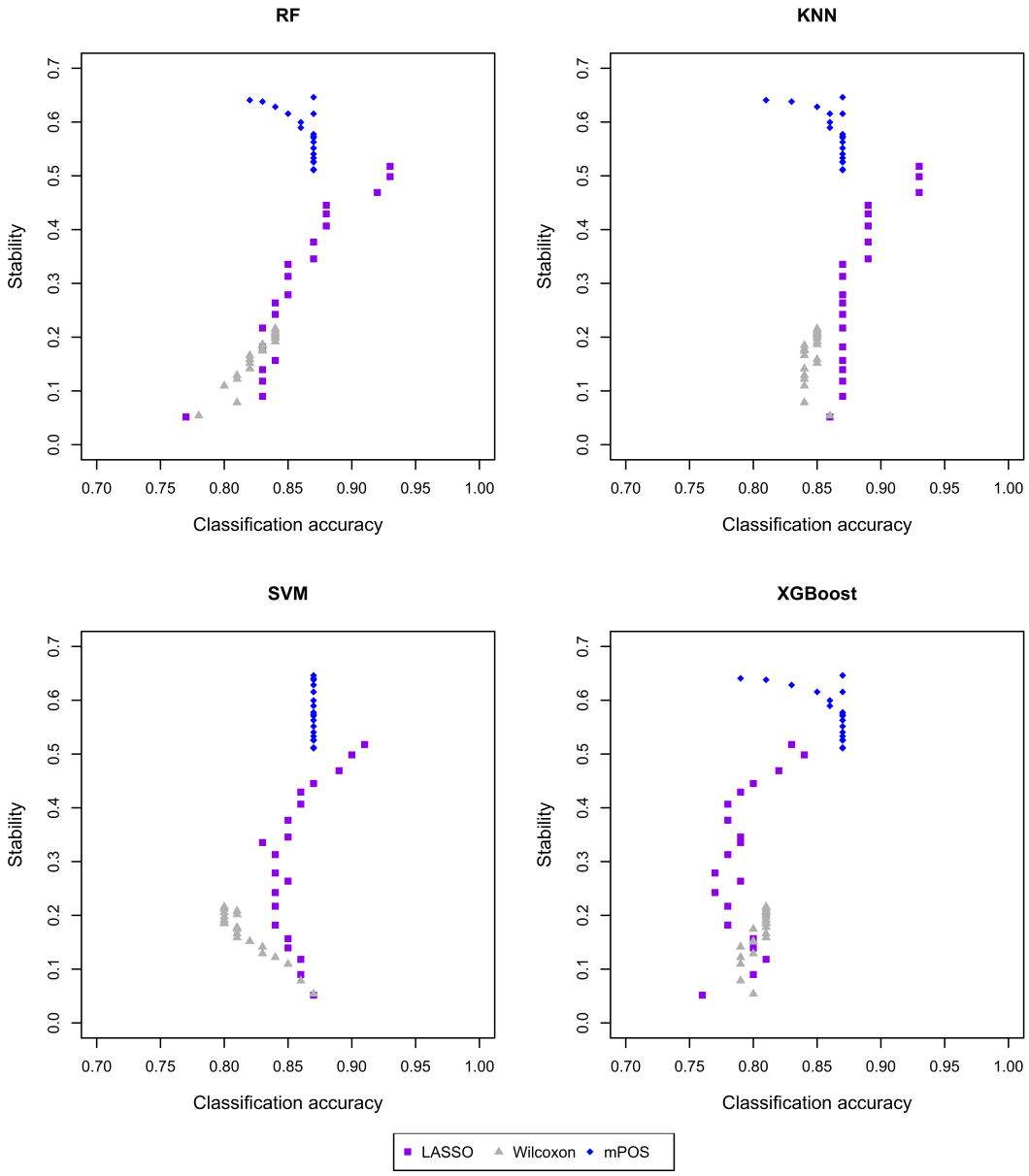

Figure 7.36: Stability - accuracy plot for GSE19615 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE19615 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

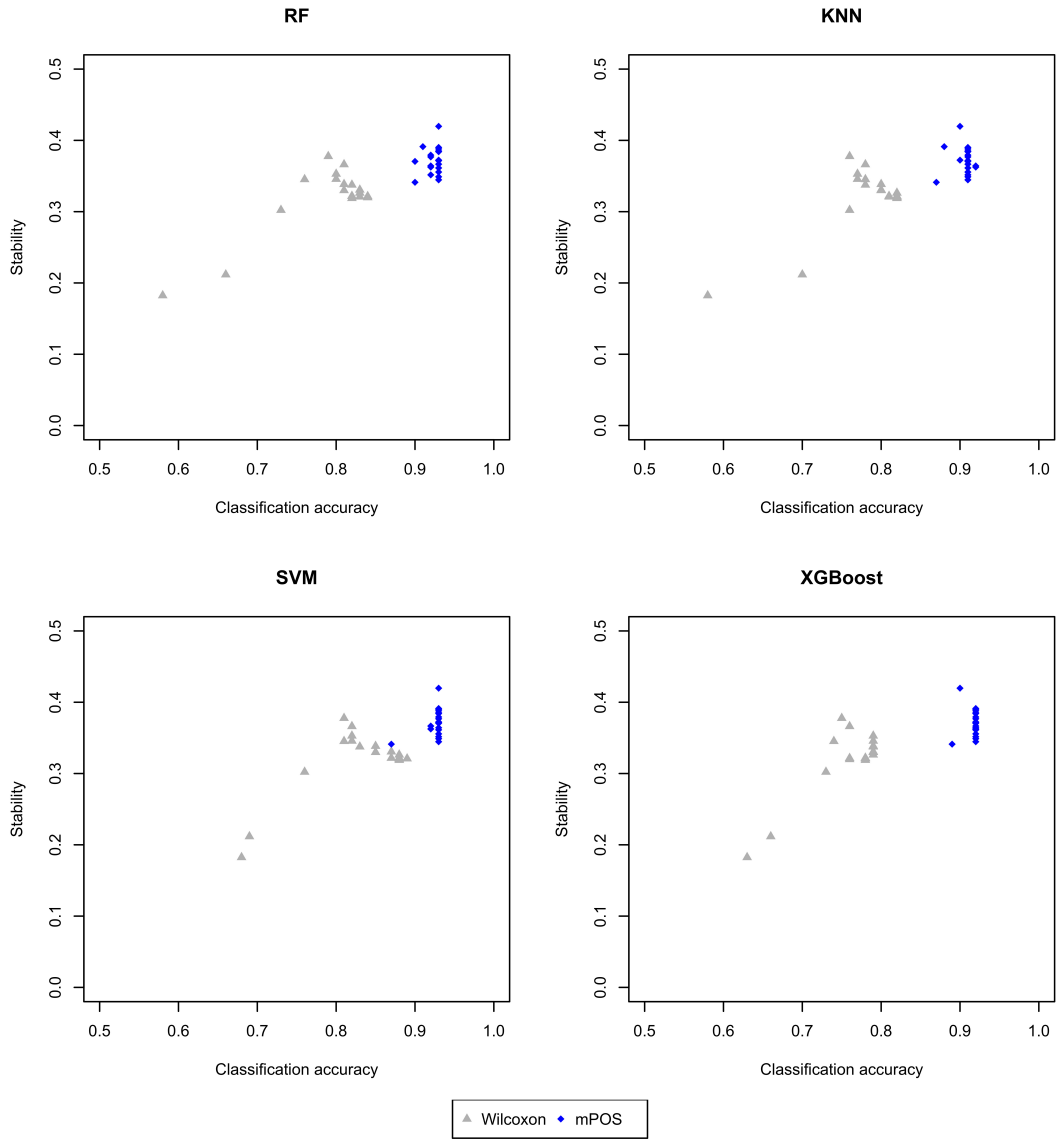

Figure 7.37: Stability - accuracy plot for GSE22513 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE22513 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

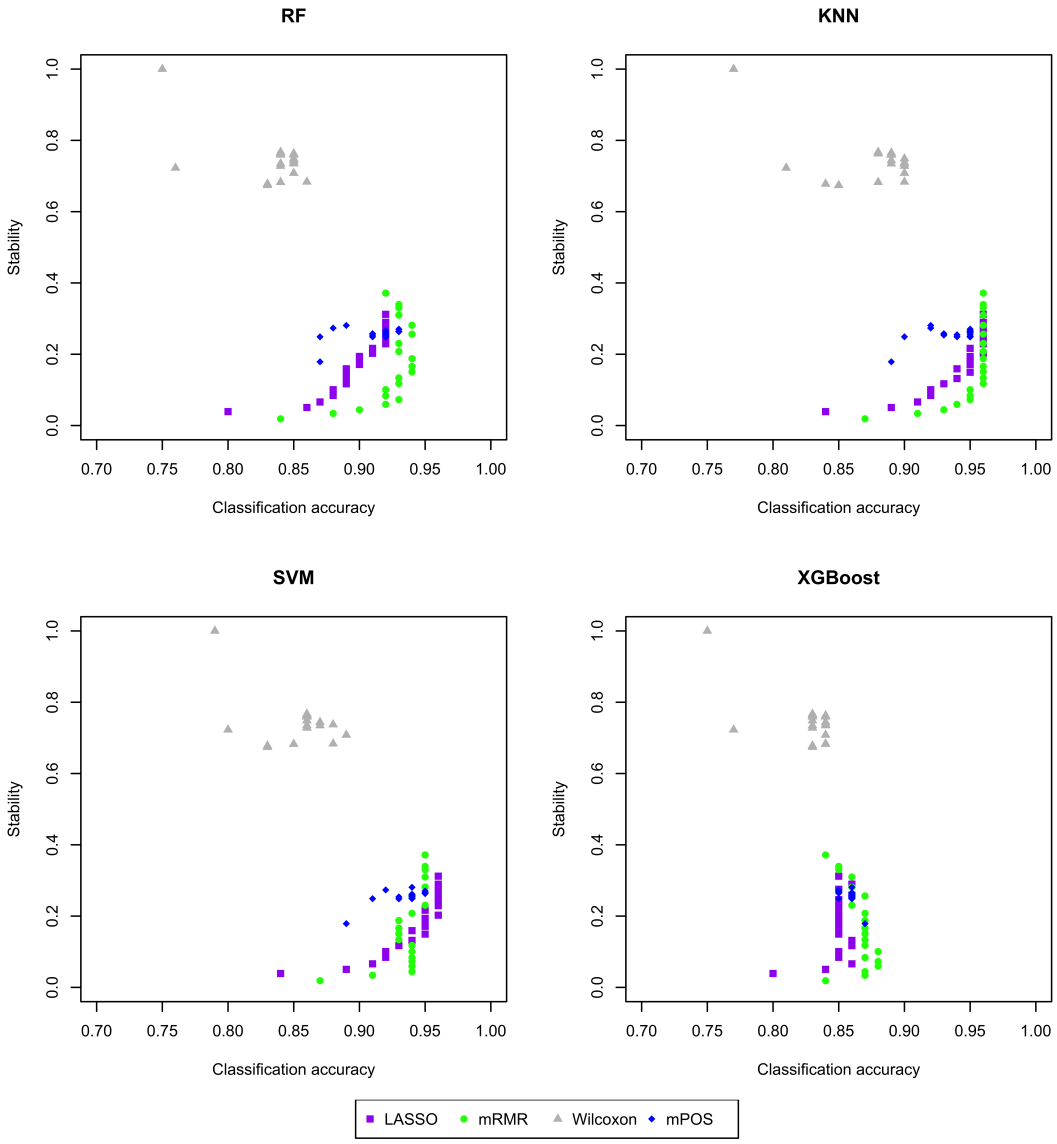

Figure 7.38: Stability - accuracy plot for GSE24514 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE24514 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

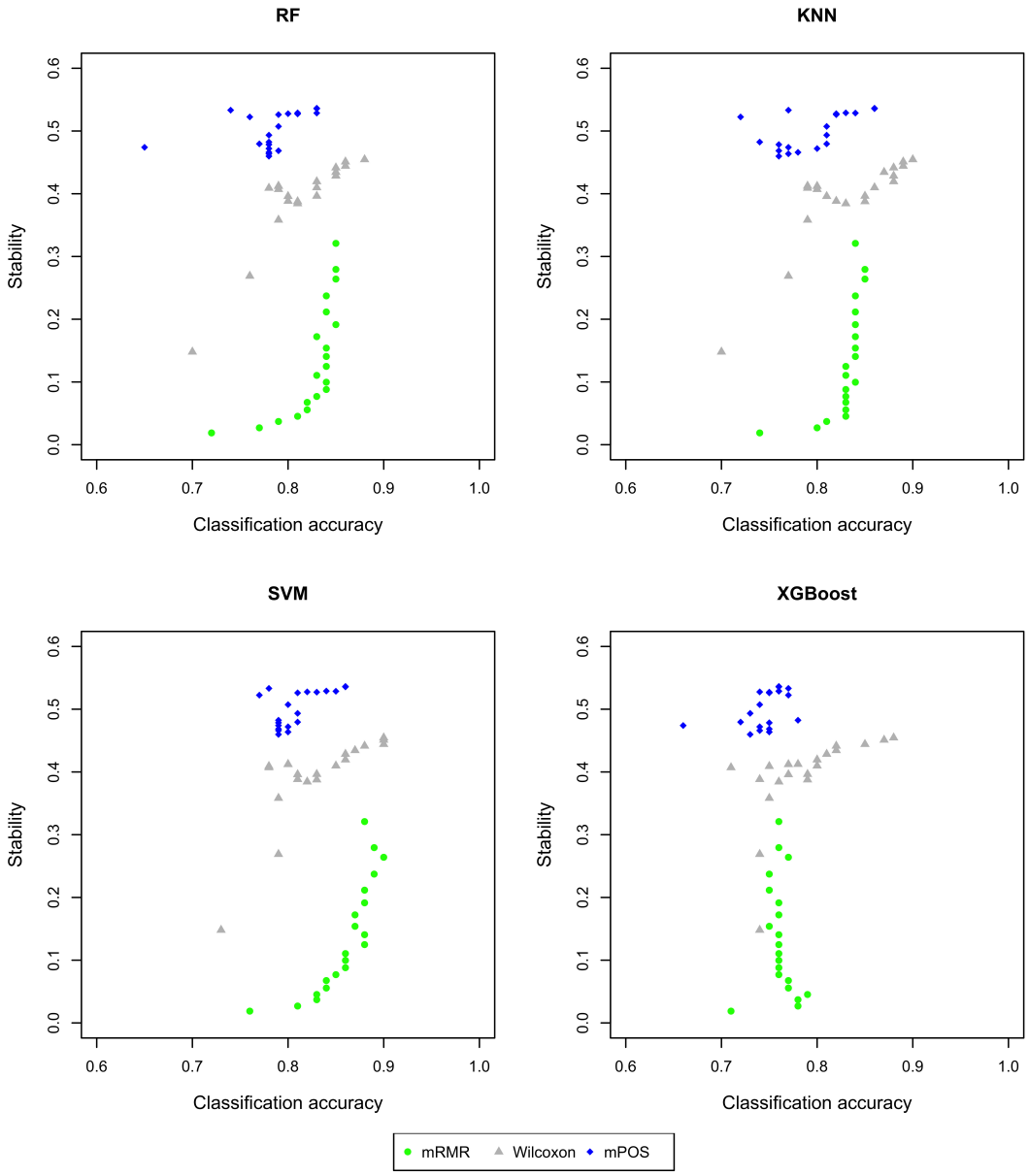

Figure 7.39: Stability - accuracy plot for GSE4045 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE4045 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

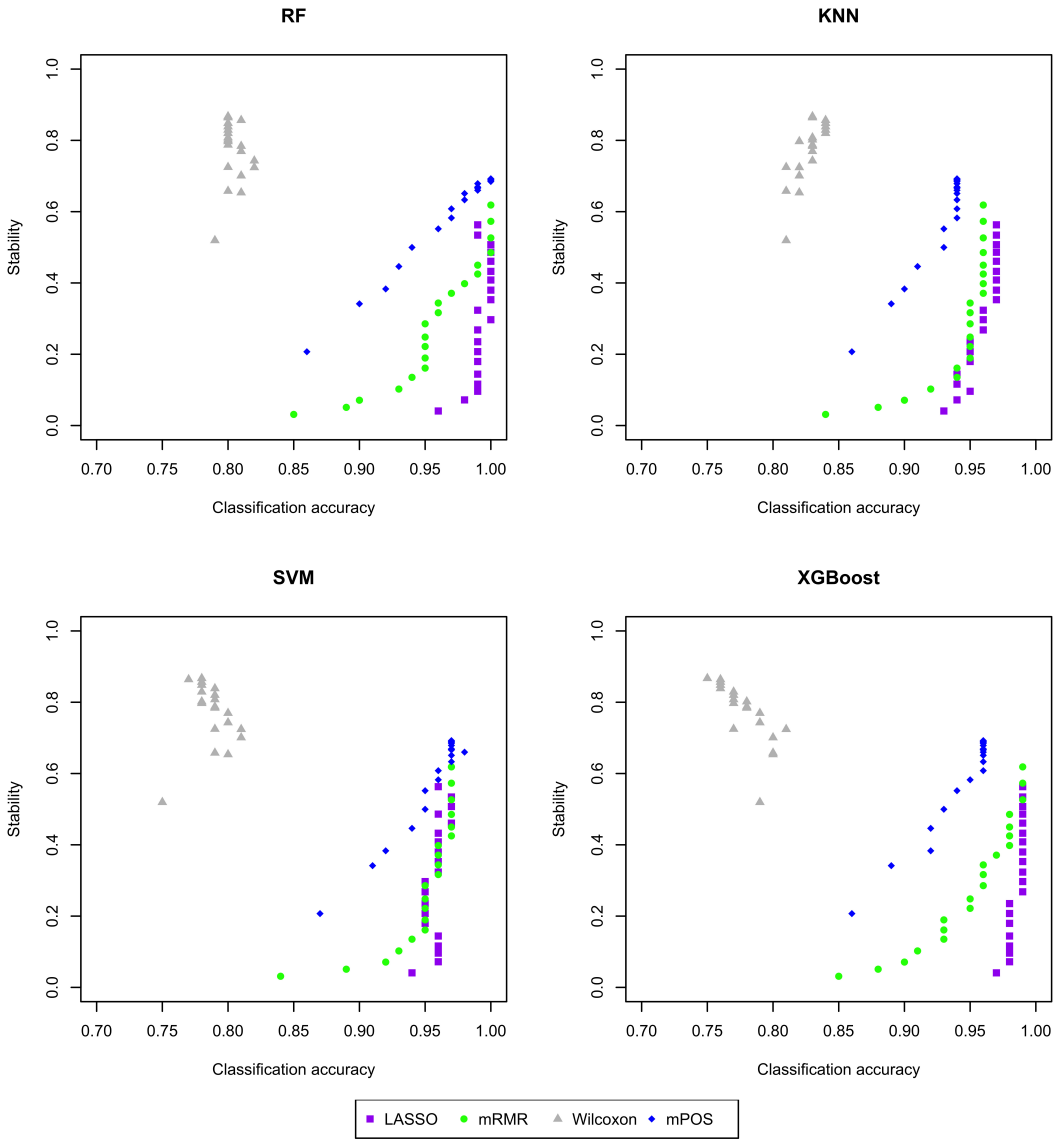

Figure 7.40: Stability - accuracy plot for Leukaemia dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on Leukaemia dataset by 20 iterations of 5-fold cross validation for four different classifiers.

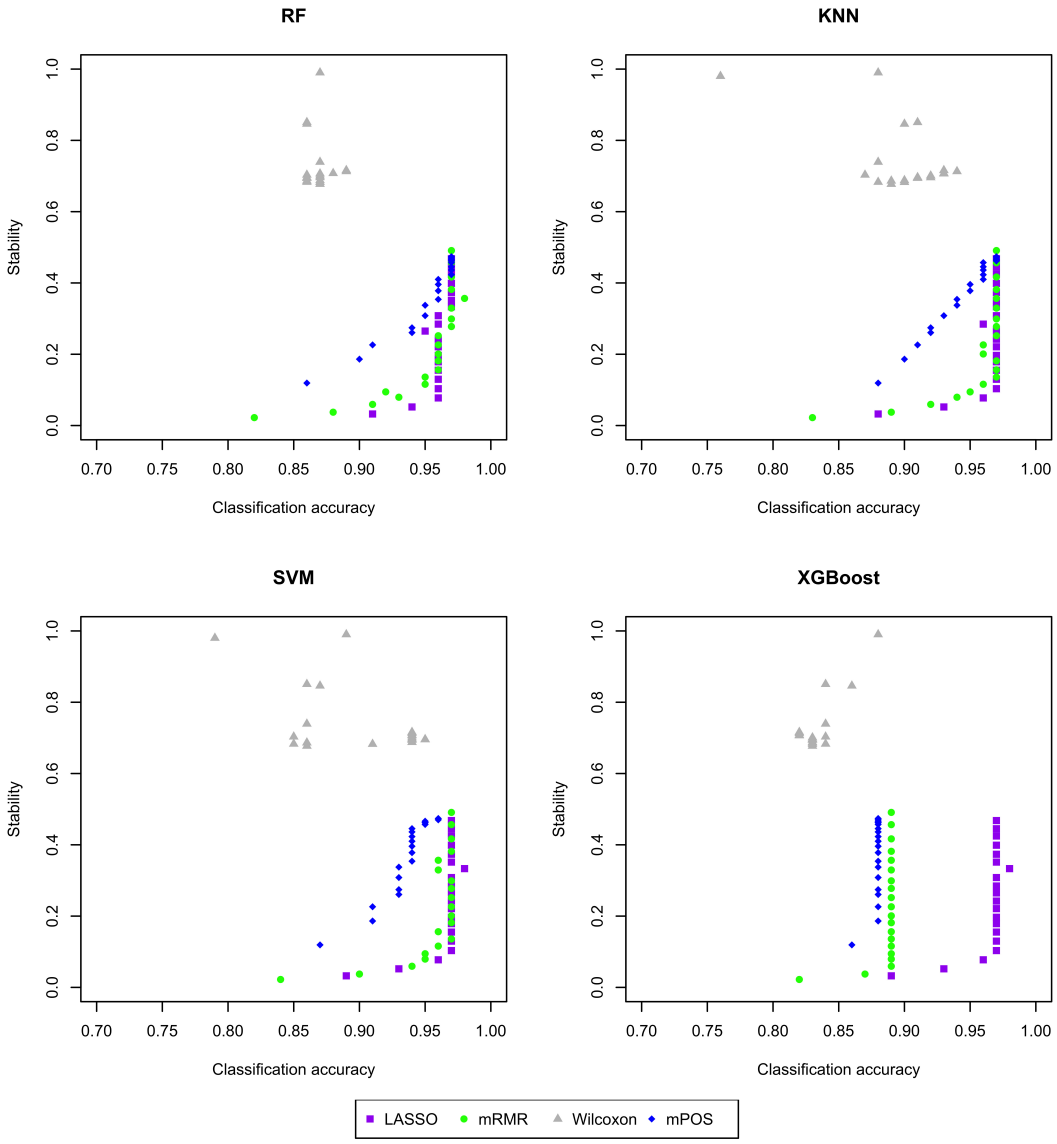

Figure 7.41: Stability - accuracy plot for Carcinoma dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on Carcinoma dataset by 20 iterations of 5-fold cross validation for four different classifiers.

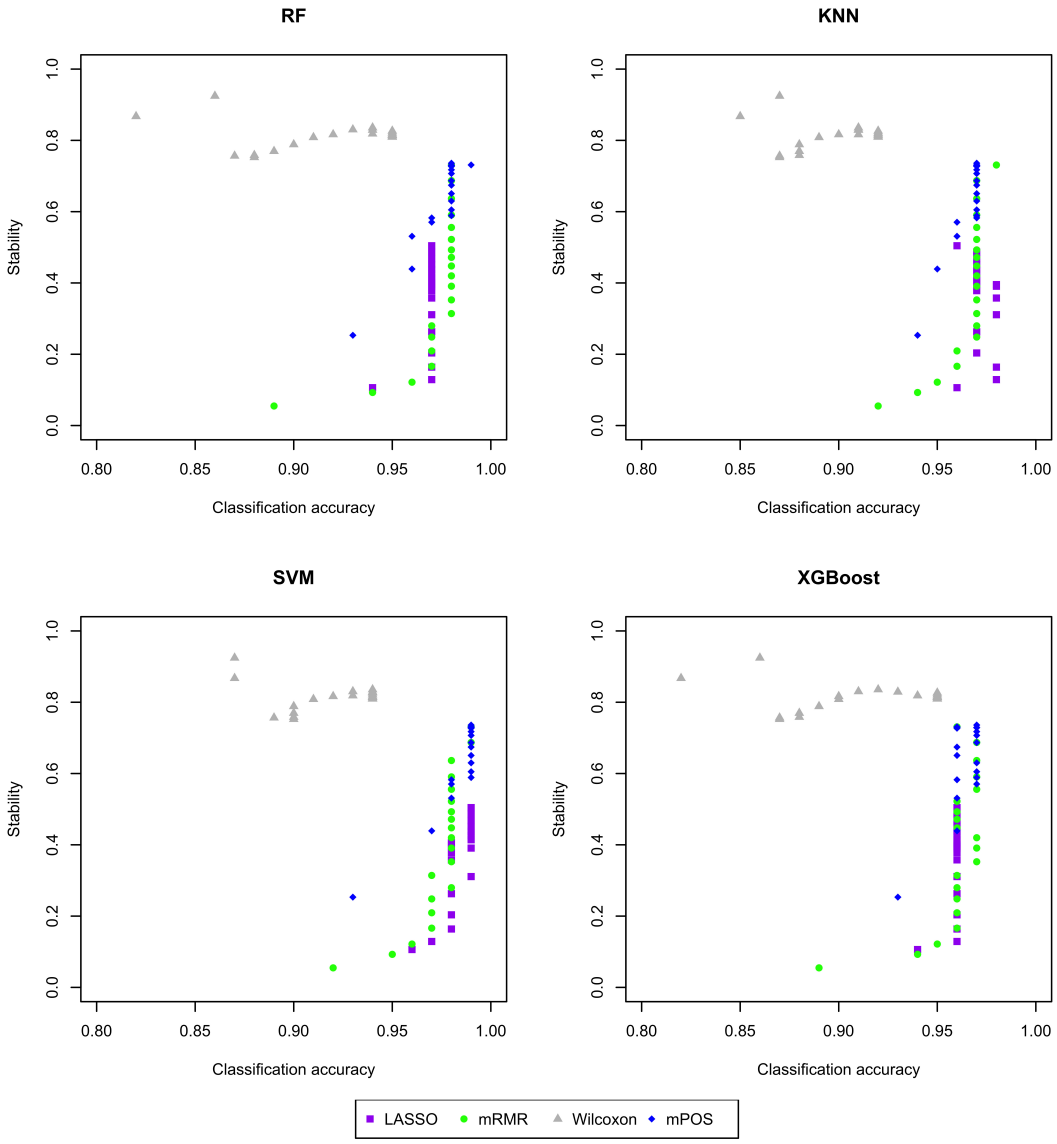

Figure 7.42: Stability - accuracy plot for Lung(1) dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on Lung(1) dataset by 20 iterations of 5-fold cross validation for four different classifiers.

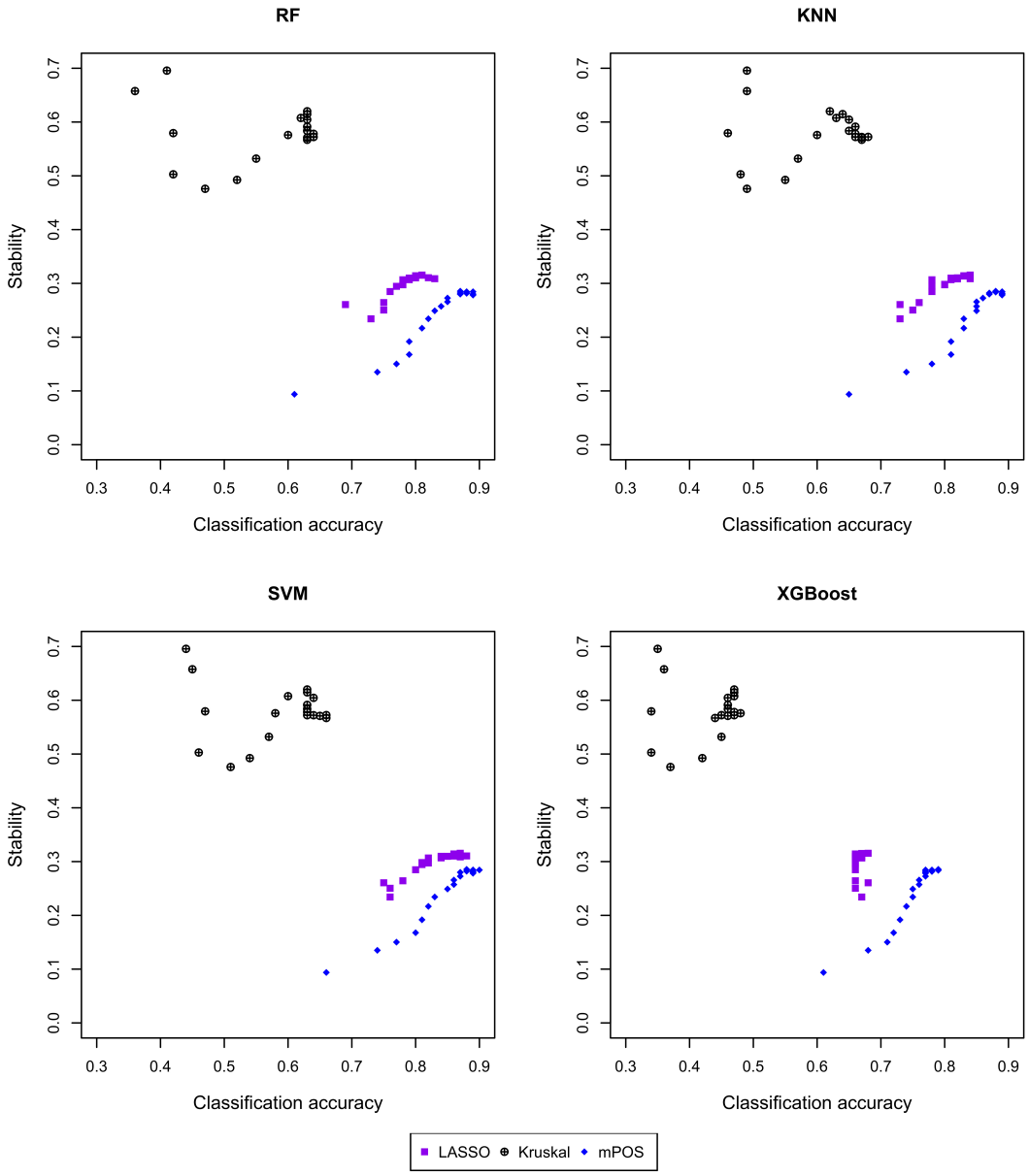

Figure 7.43: Stability - accuracy plot for GSE21029 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE21029 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

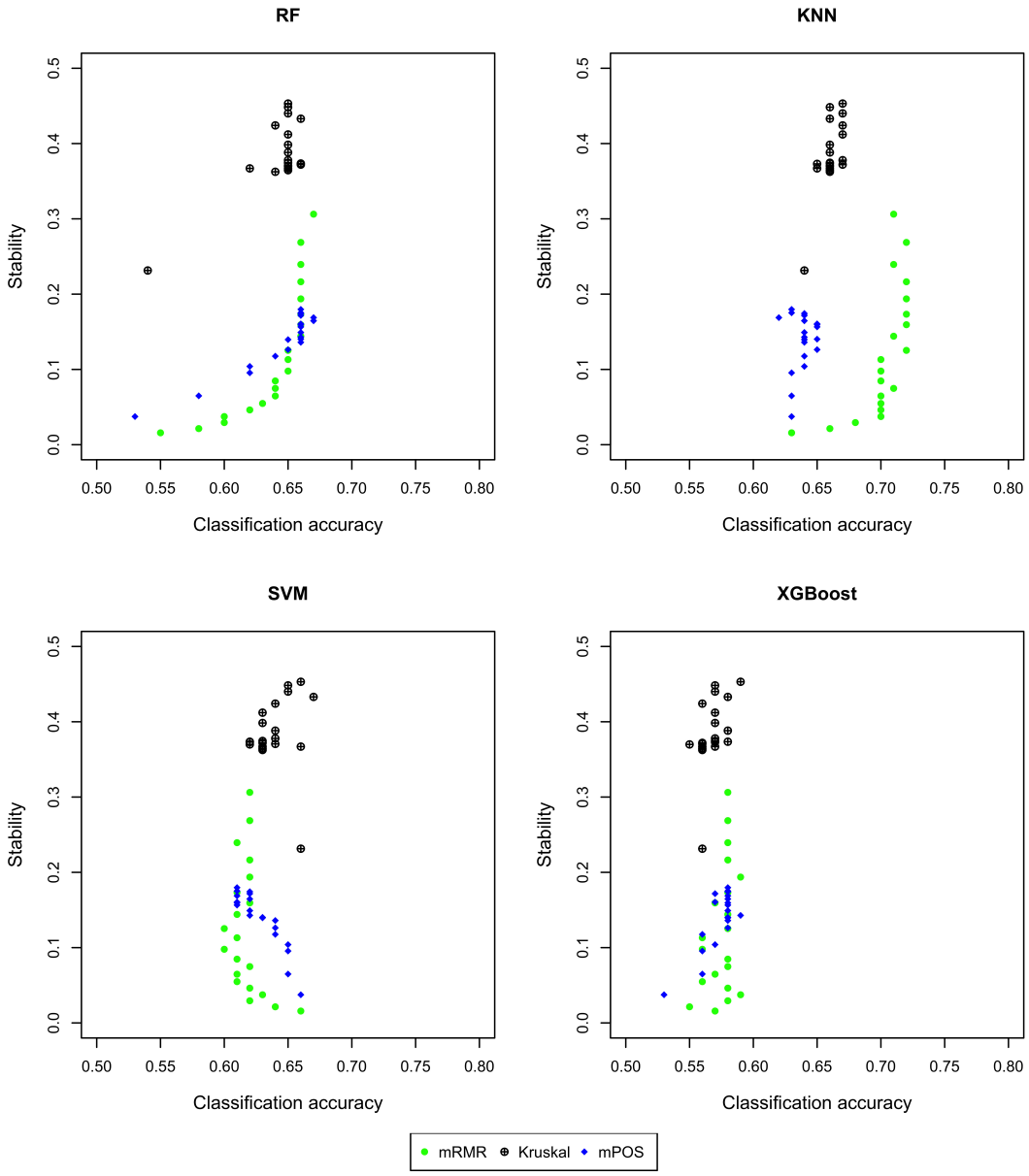

Figure 7.44: Stability - accuracy plot for GSE22093 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE22093 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

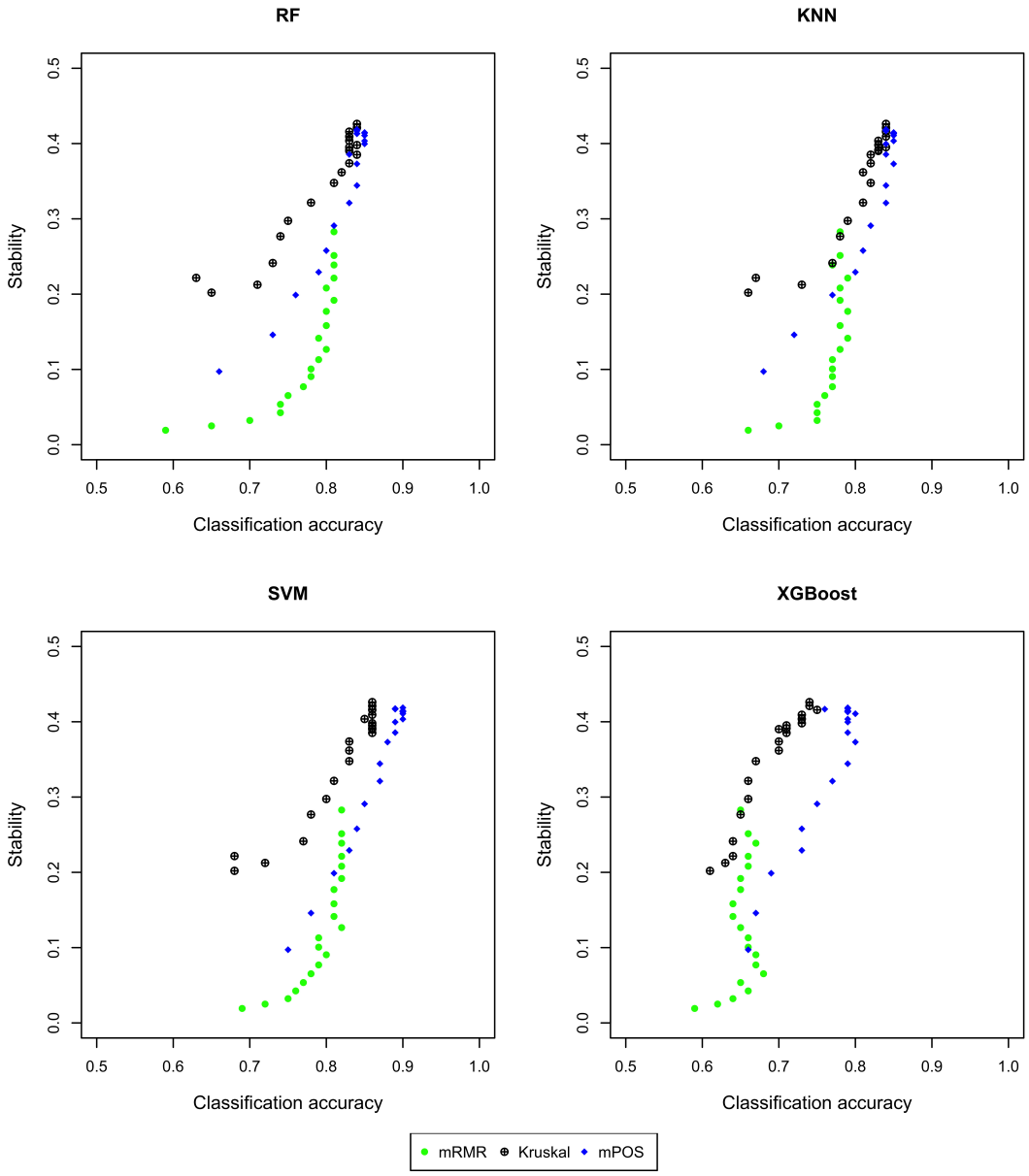

Figure 7.45: Stability - accuracy plot for GSE23938 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE23938 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

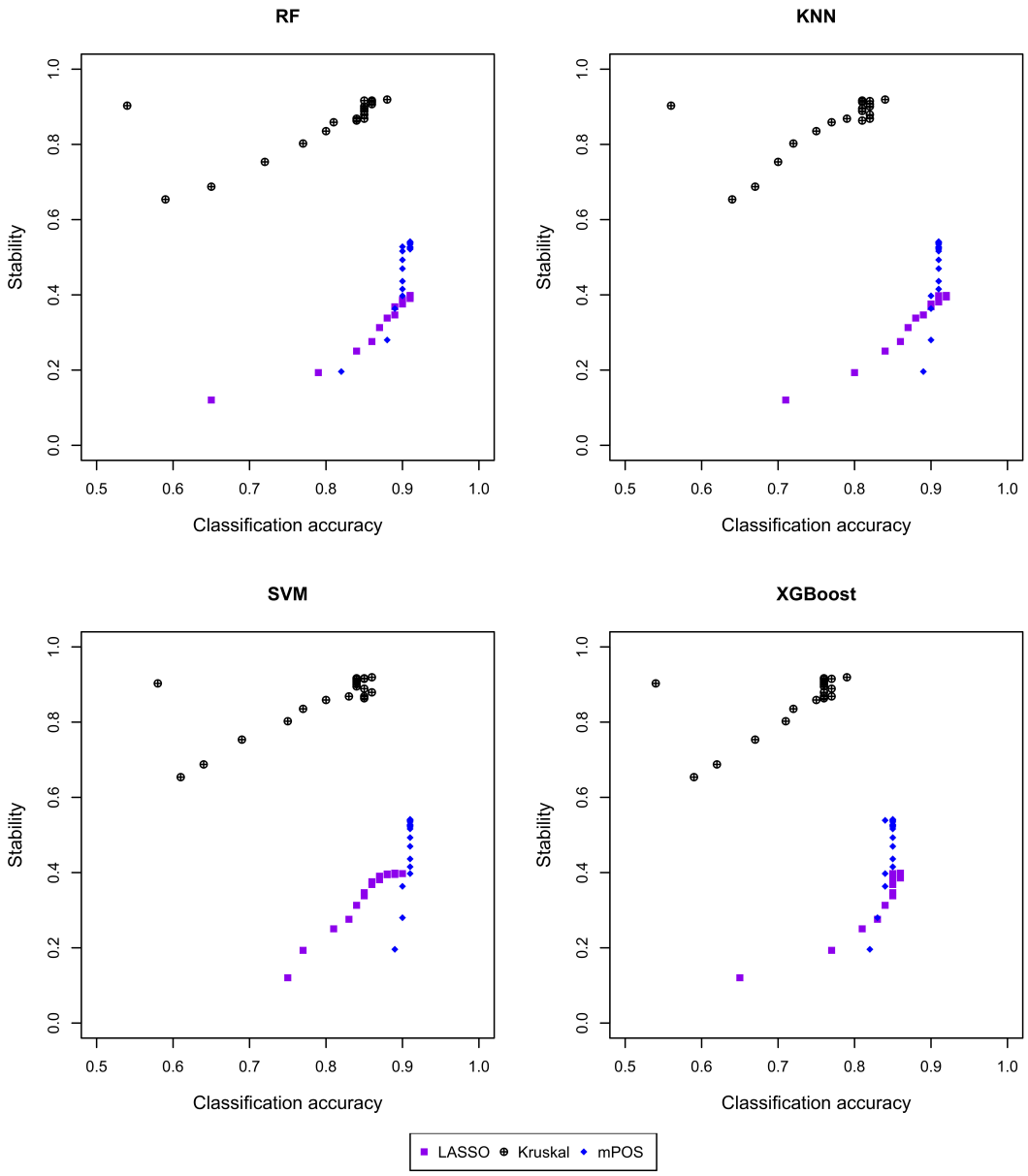

Figure 7.46: Stability - accuracy plot for GSE102079 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE102079 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

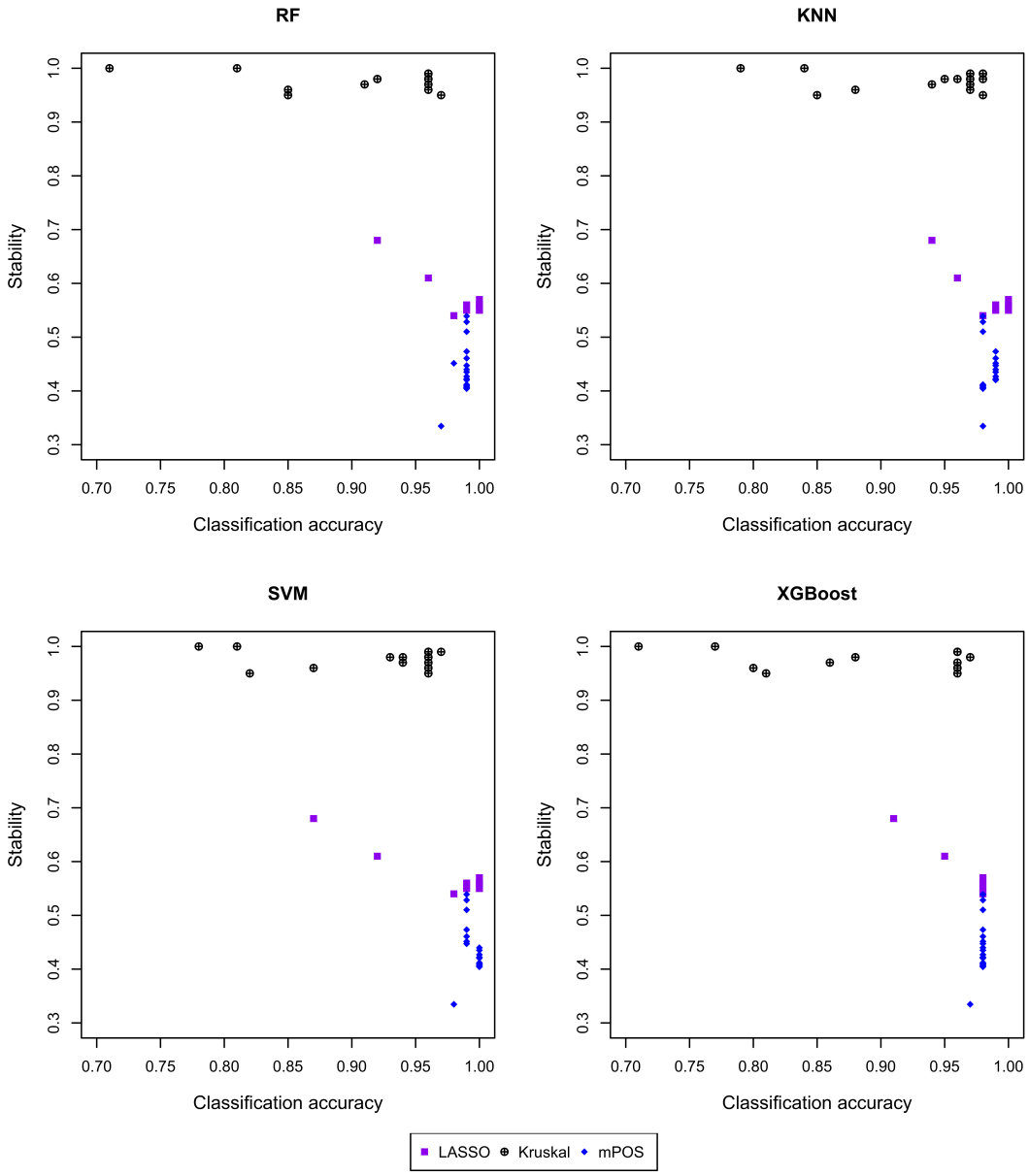

Figure 7.47: Stability - accuracy plot for GSE21510 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE21510 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

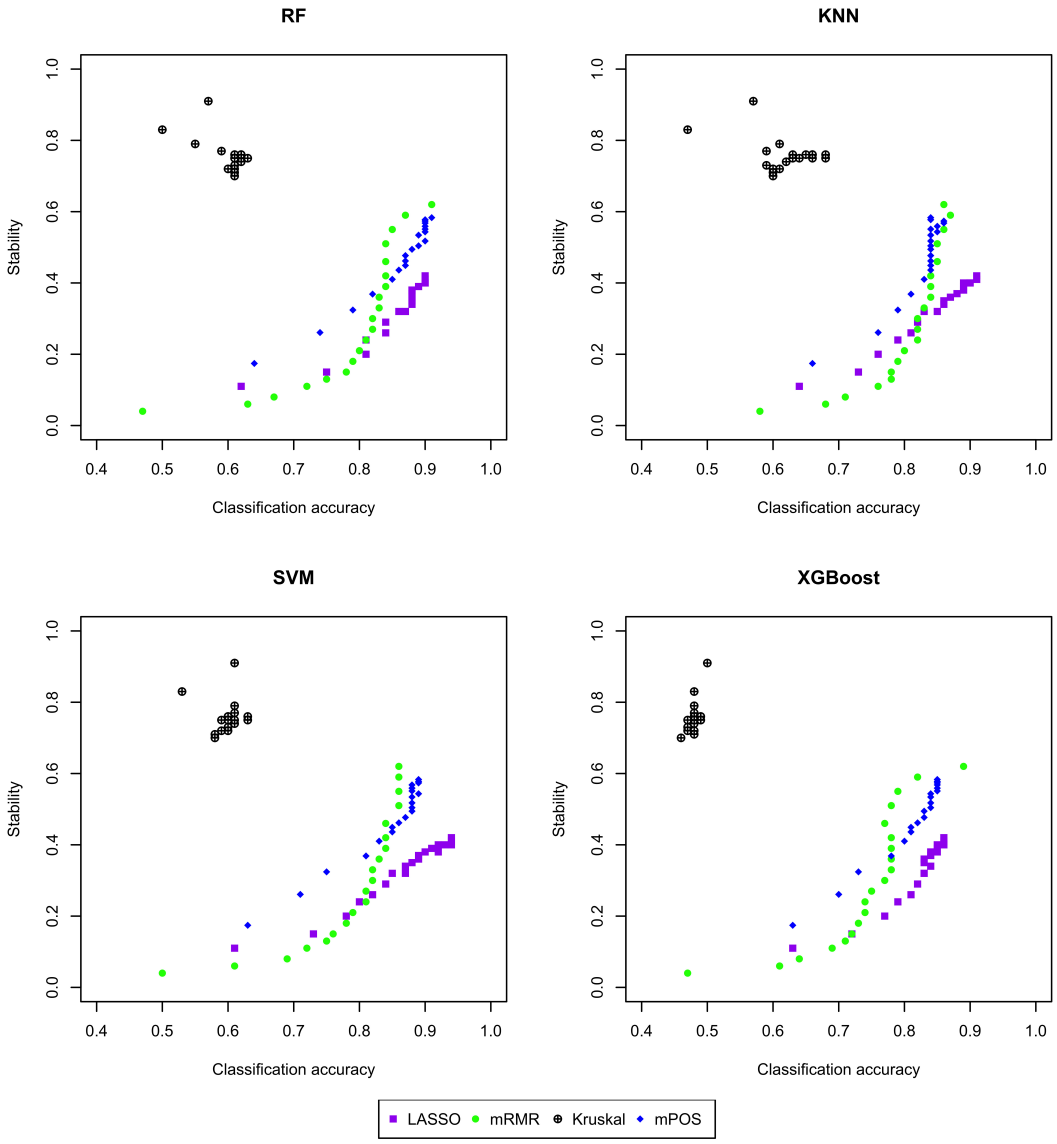

Figure 7.48: Stability - accuracy plot for MLL dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on MLL dataset by 20 iterations of 5-fold cross validation for four different classifiers.

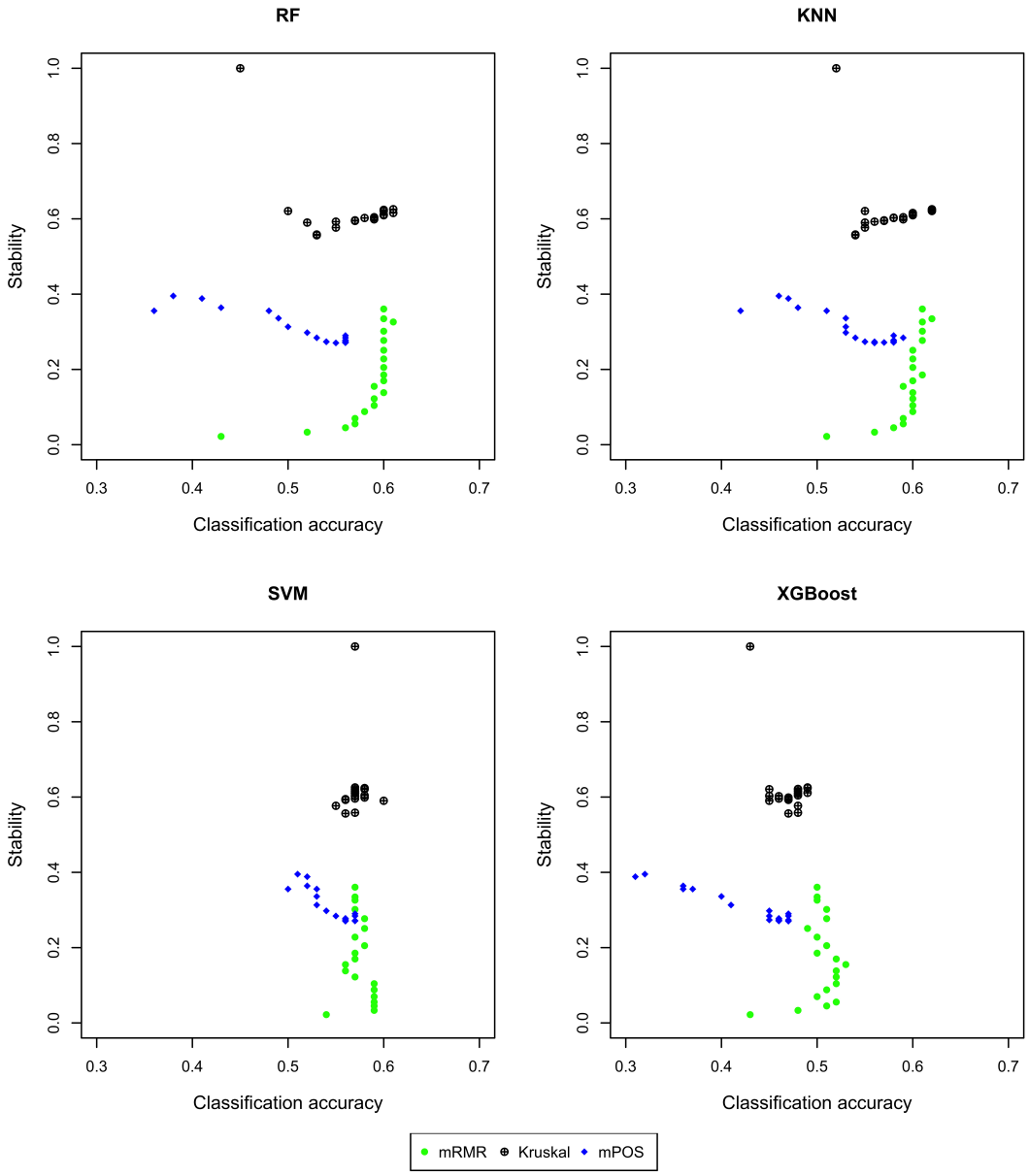

Figure 7.49: Stability - accuracy plot for GSE15852 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE15852 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

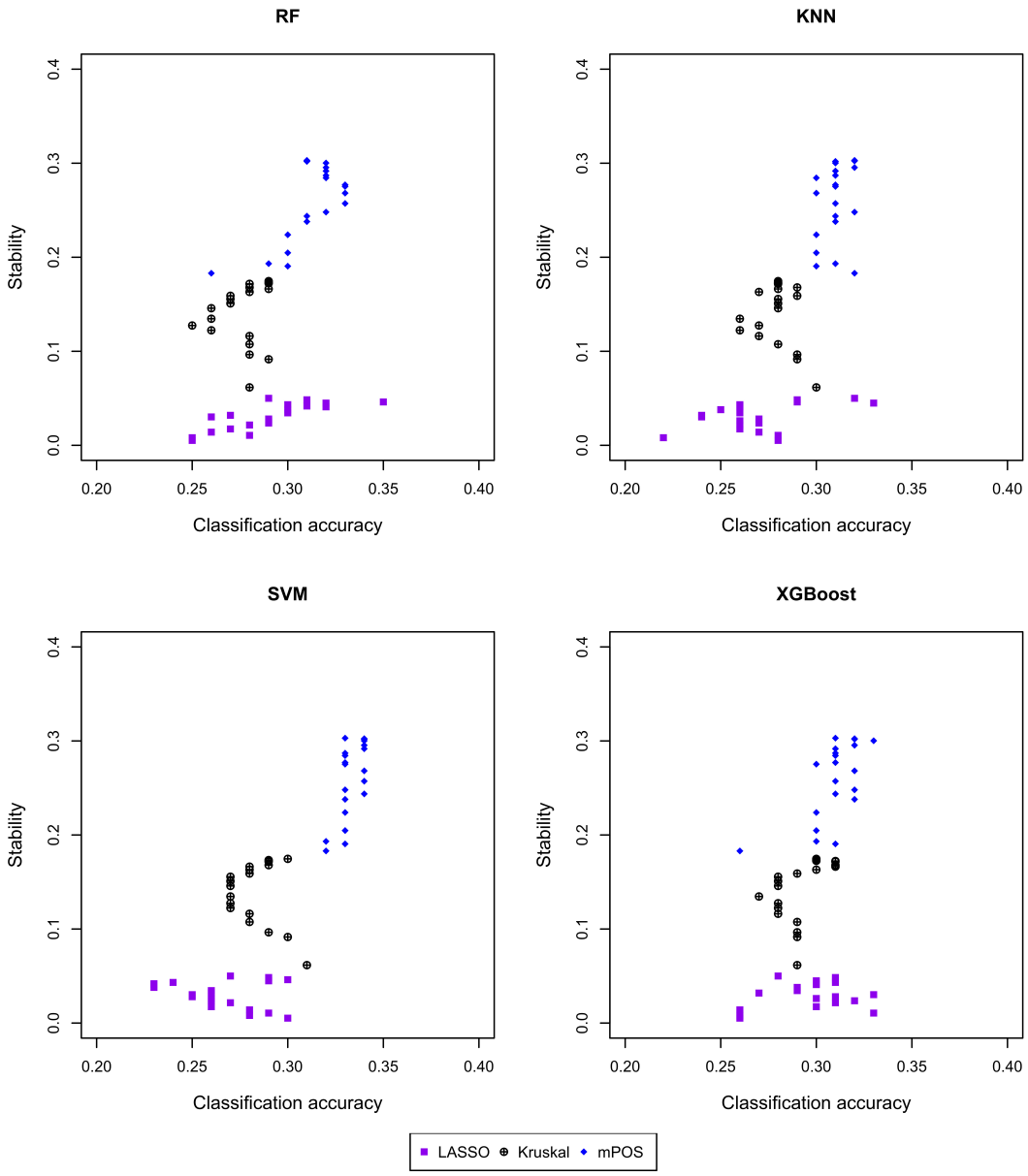

Figure 7.50: Stability - accuracy plot for GSE27854(2) dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE27854(2) dataset by 20 iterations of 5-fold cross validation for four different classifiers.

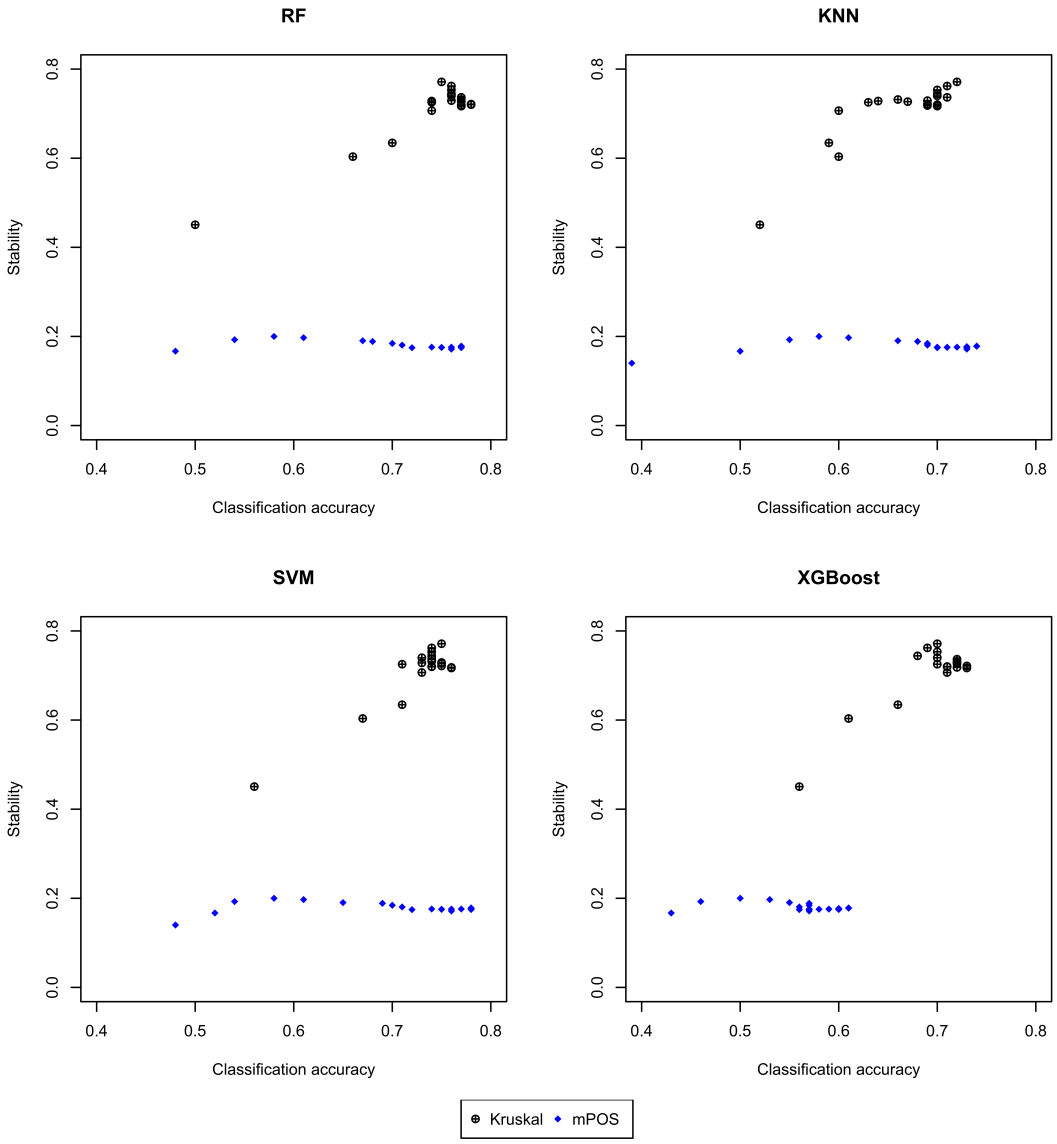

Figure 7.51: Stability - accuracy plot for GSE27651 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE27651 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

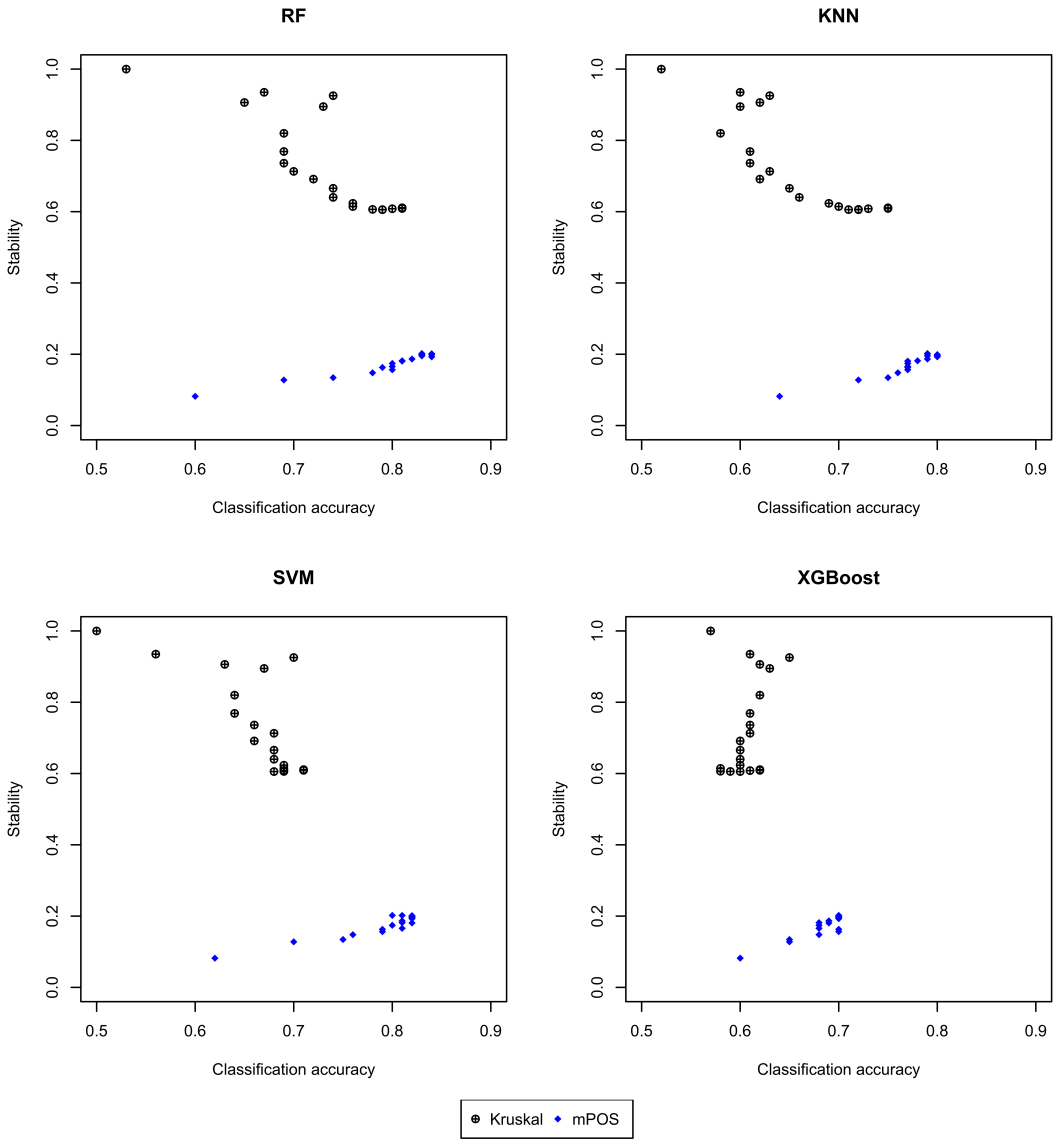

Figure 7.52: Stability - accuracy plot for GSE38666 dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE38666 dataset by 20 iterations of 5-fold cross validation for four different classifiers.

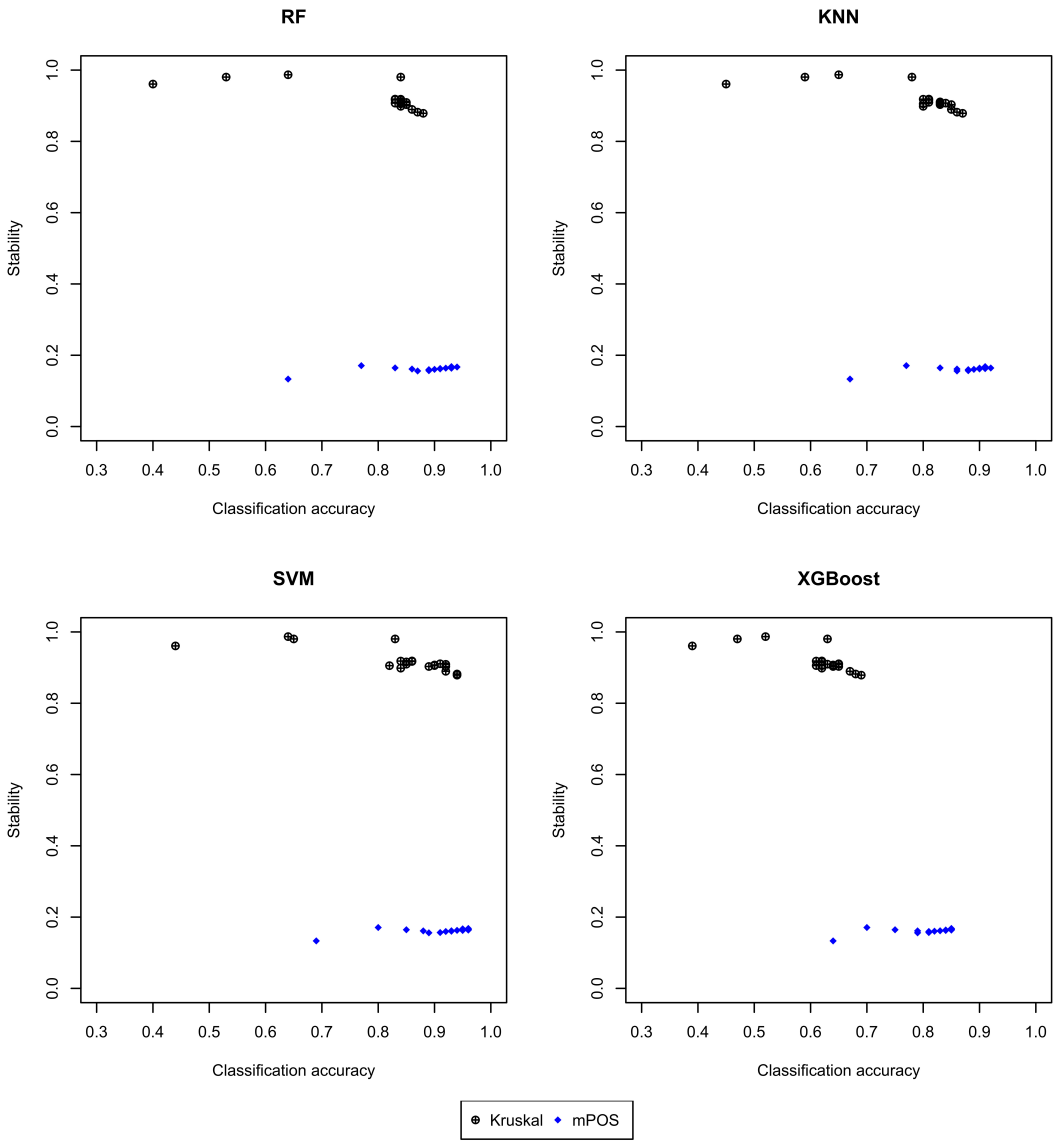

Figure 7.53: Stability - accuracy plot for GSE40595(2) dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE40595(2) dataset by 20 iterations of 5-fold cross validation for four different classifiers.

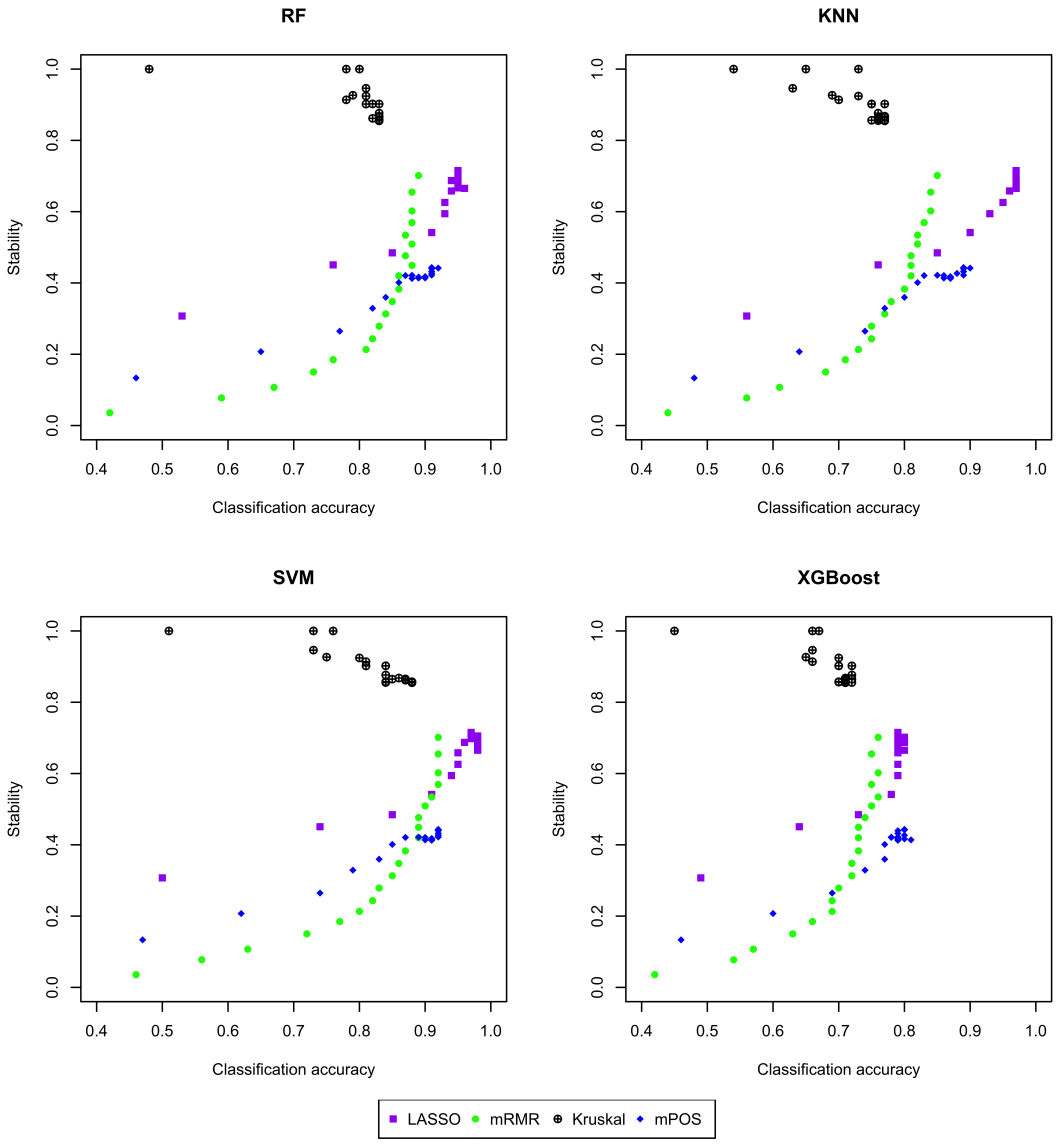

Figure 7.54: Stability - accuracy plot for Srbct dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on Srbct dataset by 20 iterations of 5-fold cross validation for four different classifiers.

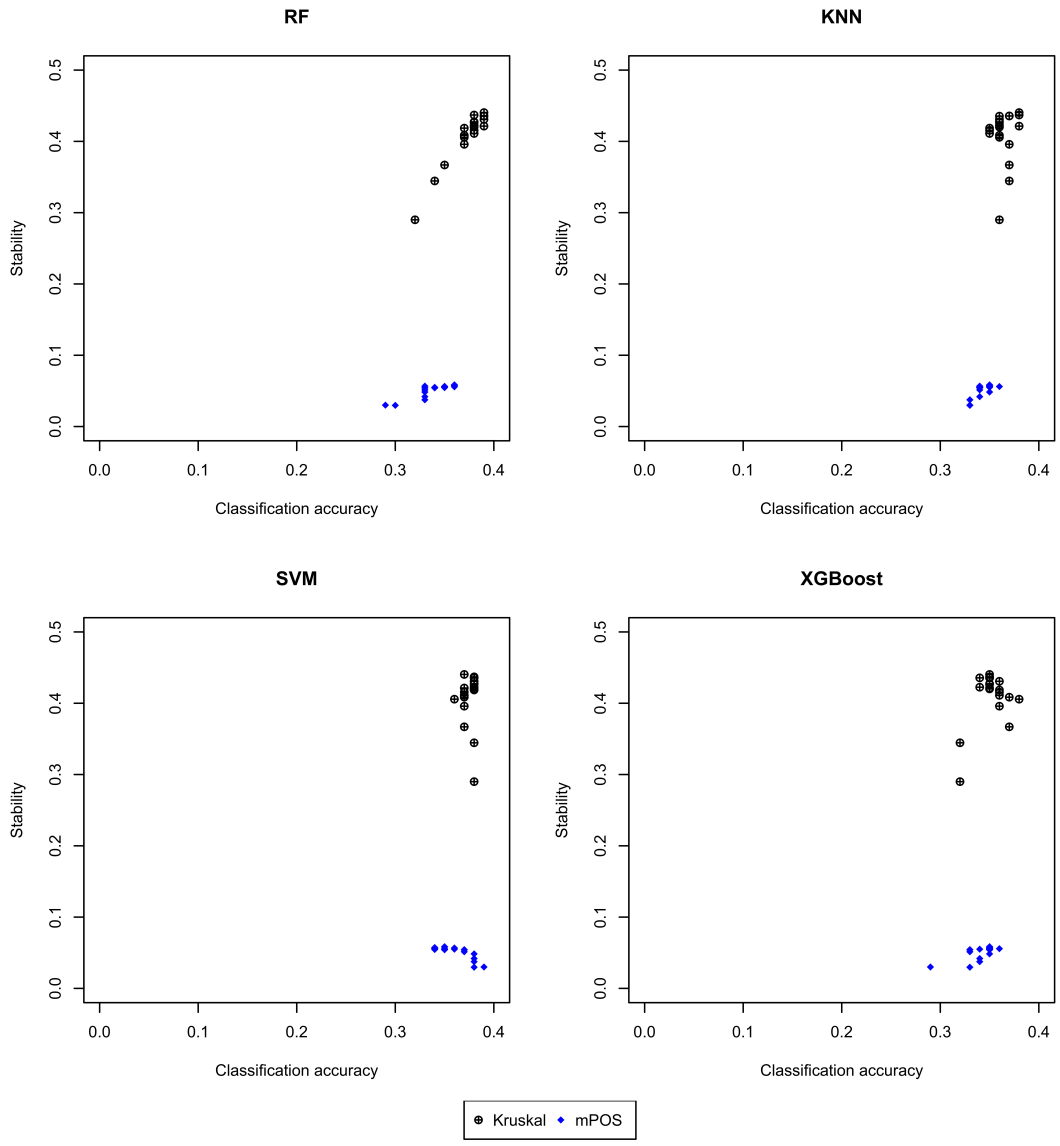

Figure 7.55: Stability - accuracy plot for GSE162228(2) dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on GSE162228(2) dataset by 20 iterations of 5-fold cross validation for four different classifiers.

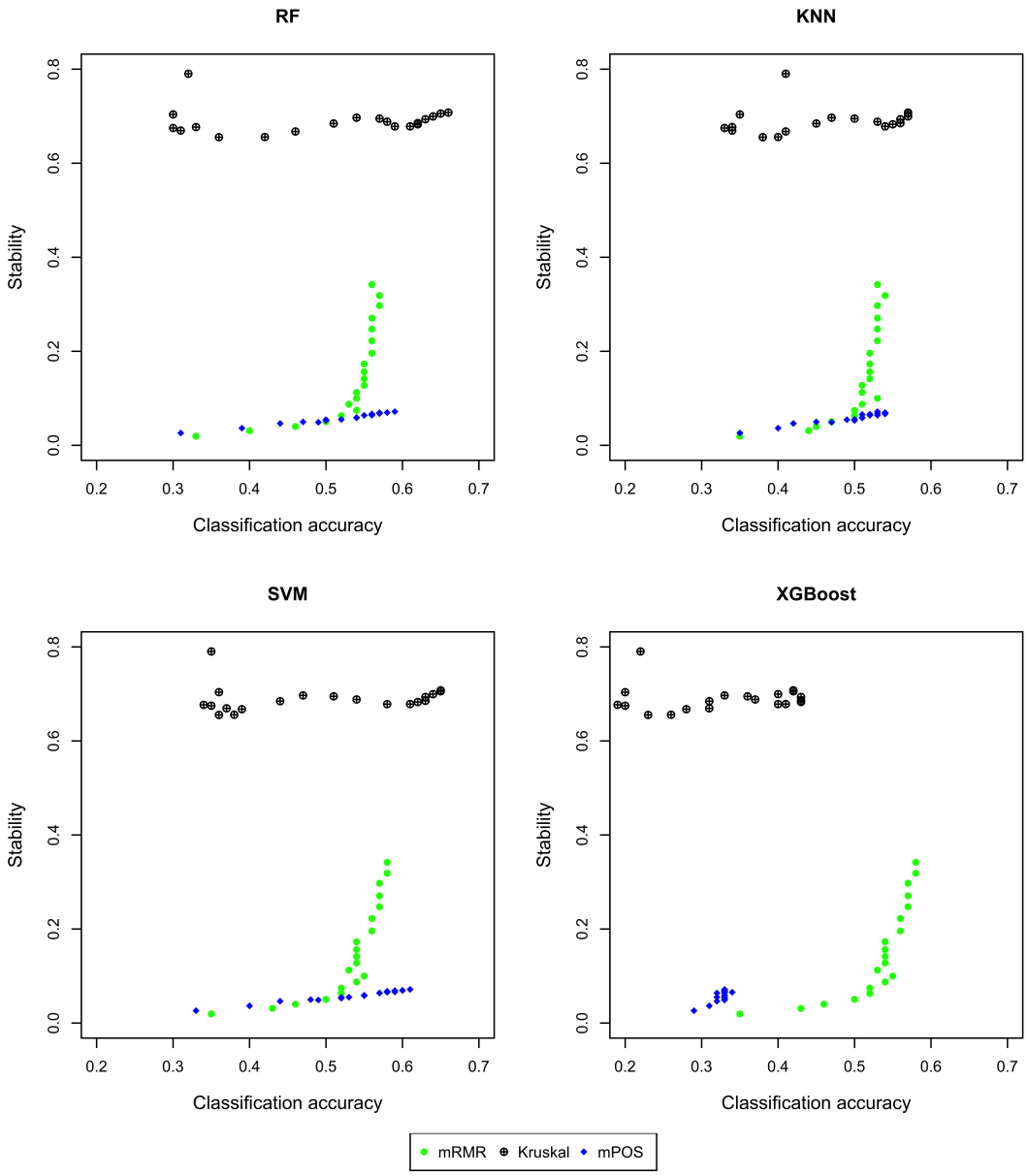

Figure 7.56: Stability - accuracy plot for Brain Tumour dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on Brain Tumour dataset by 20 iterations of 5-fold cross validation for four different classifiers.

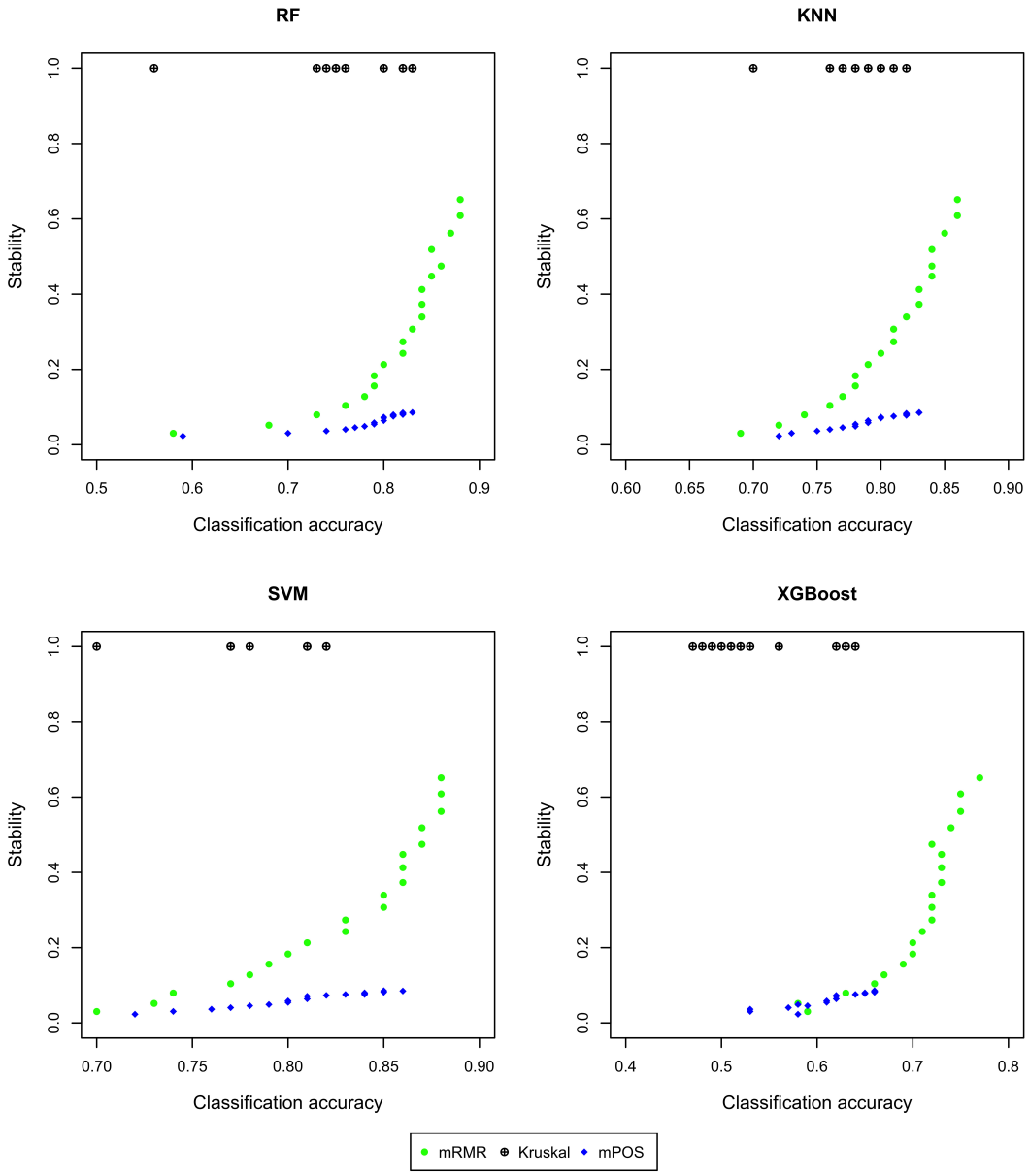

Figure 7.57: Stability - accuracy plot for Lung(2) dataset. The stability of the feature selection techniques versus the corresponding estimated classification accuracy on Lung(2) dataset by 20 iterations of 5-fold cross validation for four different classifiers.

7.5.4 Computational Complexity Analysis

The mPOS algorithm evaluates the class-separability of each gene by computing class-wise intervals, overlap regions, and scoring them. Let m be the number of genes (features) in the dataset, and N be the number of samples per gene in the dataset. The mPOS feature selection method operates on c classes, where c is the number of classes. The time complexity of each step of the mPOS algorithm, as outlined in Algorithm 3 and Section 7.2 of this thesis, is evaluated using Big O notation. The overall time complexity of the mPOS algorithm is then included as follows.

1. **Z-score Standardisation**: The z-score standardization is used to transform expression values into standardised expression values by calculating the mean and standard deviation of each gene (lines 2-4 in Algorithm 3). Each gene contributes $O(N)$, and the total cost of this step results in $O(m \cdot N)$ across all m genes.

2. **Class-Based Core Interval Computation**: The class-specific mean and standard deviation are calculated for each gene i over samples j in class c (lines 5–7 in Algorithm 3). When the class's core interval is determined, contributing $O(N)$ for each gene. Consequently, the total time cost of this step results in $O(m \cdot N)$ across all m genes.

3. **Non-Outlier Sample Count**: The number of inlier observations is considered for each gene by checking each of the N samples and determining if its value lies within its own class core interval (line 8 in Algorithm 3). This contributes $O(N)$ per gene, and the total time cost of this step results in $O(m \cdot N)$ across all m genes.

4. **Multi-Class Overlap Region Calculation**: All possible overlap regions between class intervals are computed for each gene i (lines 10–12 in Algorithm 3), e.g., two-way, three-way, $\cdots$, up to ω-way overlaps . The number of class subsets of size $c \geq 2$ is expressed as follows:

$$\sum_{k=2}^{c} \binom{c}{k} = 2^c - c - 1 \tag{7.10}$$

   To compute the intersection, it requires $O(2^c - c - 1)$ overlap subsets per gene. Therefore, the total time cost of this step results in $O(m(2^c - c - 1))$ across all m genes.

5. **mPOS Score Calculation**: mPOS score is calculated for each gene by plugging the values into the final scoring formula (Equation 7.8) This computation results in only a constant number of arithmetic operations for each gene (line 13 in Algorithm 3). Therefore, the mPOS score is $O(1)$ per gene and results in $O(m)$ across all m genes as the total time cost of this step.

6. **Final Selection**: The mPOS scores are sorted in ascending order to rank informative genes (line 14 in Algorithm 3). Moreover, the top r genes from the sorted list are selected (line 15 in Algorithm 3). Both sorting and selecting genes result in $O(m)$ across all m genes as the total time cost of this step.

Overall time complexity, by taking into account Z-score standardization, class-wise core interval computation, as well as non-outlier sample count, the dominant cost of the mPOS algorithm is scaled on the order of $O(m \cdot N)$. The multi-class overlap region computation is denoted as $O(m(2^c - c - 1))$ when the number of classes c is varied. Furthermore, mPOS score calculation and final selection contribute $O(m)$. Therefore, for the worst-case scenario, the time complexity of the mPOS method can be expressed as

$$O(m(N + 2^c - c)) \tag{7.11}$$

To provide a clear assessment of computational efficiency, [191, 25] examined time complexity to evaluate the efficiency of feature selection algorithms. By following this scheme, a comparison of the time complexity of the mPOS alongside LASSO, mRMR, and the Wilcoxon/Kruskal methods using Big O notation is evaluated, as seen in Table 7.3. Table 7.3 demonstrates that the relative scalability of mPOS in relation to sample size, feature dimensionality, as well as class sizes.

Table 7.3: Comparison of theoretical time complexity for different feature selection methods

Methods	**Theoretical Time Complexity**
LASSO	$O(N \cdot m \cdot I)$, where I denotes the number of iterations
mRMR	$O(m^2 \cdot N)$
Wilcoxon/Kruskal	$O(m \cdot N \log N)$
mPOS	$O(m(N + 2^c - c))$

7.6 Summary

This chapter discusses the concept of an extended version of POS [123] and 3cPOS (in Chapter 5) using overlapping analysis. Overlapping analysis is commonly utilised to validate the relevance of genes in various aspects. For identifying genes with significant relevance, this approach can boost predictive accuracy, learning performance, and decision-making across multiple applications such as machine learning, pattern recognition, and bioinformatics.

We proposed a novel feature selection algorithm, called the multiple Proportional Overlapping Score (mPOS). This aims at estimating the overlapping degree for each gene by considering class intervals, overlapping between intervals, and mPOS measure. The class intervals is determined to alleviate the effects of outliers. The overlap between classes is analysed and a novel mPOS measure is derived to identify the ability of a gene to distinguish the correct target class. Genes with lower mPOS scores indicate higher discriminative power.

A total of twenty-four publicly available gene expression datasets were used to evaluate the performance of mPOS method, in comparison with four other well-known feature selection techniques: Wilcoxon, Kruskal, mRMR, and LASSO. The informative gene sets of different sizes, up to 20 genes, are selected using these feature selection techniques to construct predictive models. Random Forest, k Nearest Neighbor, Support Vector Machine, Extreme Gradient Boost were employed to construct classification models. The average classification accuracy given by the considered classifiers was used for assessing the classification performance over 20 repetitions of 5-fold cross-validation.

Experimental results demonstrate that mPOS either outperforms or exhibits comparable performance to the four representative competing feature selection techniques in 14 out of the 24 datasets when evaluated with the Random Forest classifier. Our proposed approach is better than, or comparably well to, four representative competing feature selection algorithms in 19 out of the 24 datasets when using the k Nearest Neighbor classifier. mPOS is superior, or comparably well to, other competing feature selection techniques in 17 out of the 24 datasets using Support Vector Machine classifier. By evaluated with the Extreme Gradient Boost classifier, mPOS is superior, or comparably well to, other competing feature selection techniques in 16 out of the 24

datasets.

Overall, mPOS achieves either outperforms or demonstrates comparable performance to the four representative competing feature selection techniques using RF, KNN, SVM, and XGBoost classifiers. It also maintains a stable performance across different set sizes of selection features and an effective trade-off good trade-off between stability and classification accuracy. A key benefit of mPOS is its capability to accommodate an unlimited number of genes, even when dealing with small sample sizes. This positions mPOS as a feature selection technique that operates without limitations, making it highly adaptable in a wide range of settings and diverse datasets, and robust for various applications in genomic studies and related research fields. Moreover, a comparative evaluation of computational complexity reveals that mPOS incurs a relatively low computational cost when compared with other feature selection methods, making mPOS as a computationally efficient and powerful method.

CHAPTER 8

Simulation Studies

8.1 Introduction

In epidemiology and biostatistics, statistical methods are widely employed to address various research questions. However, most statistical methods are developed under specific assumptions, which can be challenging to verify in practical applications. For instance, common issues such as imbalanced class distributions, missing data, measurement errors, unmeasured confounders, and insufficiently accurate information on event timings can significantly impact the accuracy and validity of proposed analyses [20]. To address these challenges, simulation studies are suggested to get insight into the ability of statistical methods in various scenarios.

Simulation studies have become a significant tool for statistical research or other related fields, particularly for the process of generating the data to consider properties of methods, for the evaluation of new methods and the comparison of alternative methods [134]. [185] have conducted a simulation study to assess the performances of several combinations of classifiers and feature selection methods and their dependence on the class distribution, dimensionality, and the training sample size. Some studies have compared some basic feature selection methods using both model-based simulated data and real data such as gene expression data, as found by [68] and [86]. [72] generated simulated datasets to validate the performance of an ensemble of a subset of kNN classifier (ESkNN) under different setups.

This chapter discusses simulation studies to enhance the understanding of the 3cPOS and mPOS methods across various setups/scenarios. Two distinct simulation models: Simulation models 1 and 2, are exploited to form datasets for investigating properties of methods, as detailed in Sections 8.2.1 and 8.2.2. The experimental setups for the simulation studies are detailed in Section 8.2.3, where several scenarios are presented to generate datasets with balanced class distributions and varying degrees of overlap between the classes. These scenarios are designed to evaluate and assess the performance of the 3cPOS and mPOS methods, while also comparing their performance against other feature selection techniques; LASSO, mRMR, Wilcoxon, and Kruskal, across multiple classifiers, including Random Forest (RF), k-nearest Neighbors (KNN), Support Vector Machines (SVM), and Extreme gradient boost (XGBoost), so as to assess the predictive performance of the 3cPOS and mPOS methods.

8.2 Data Simulation for Main Simulation Experiments

In Chapters 5 and 7, it is demonstrated that the 3cPOS and mPOS techniques outperformed all other feature selection methods in terms of classification accuracy and stability. However, it is important to note that the majority of the datasets, which correspond to the target classes, are characterised by imbalanced class distributions. To address these gaps, we aim to implement simulation studies to get insight into abilities of the 3cPOS and mPOS methods under balanced class distribution.

We exploited two simulation models to provide four different experiments as follows:

8.2.1 Simulation Model 1

In this model, identity matrix is determined for the covariance matrix to generate informative features in the first two experiments, $\Sigma = \omega I$. The variance - covariance matrix, I, which is a $d \times d$ matrix, is:

$$I = \begin{bmatrix} \sigma_{1,1} & 0 & ,\cdots, & 0 \\ 0 & \sigma_{2,2} & ,\cdots, & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & ,\cdots, & \sigma_{d,d} \end{bmatrix} \tag{8.1}$$

where $\sigma_{i,j}$, on the diagonal of I, is the variance, $\sigma_{i,j} = 1$, when $\omega = 1$. The first experiment maintains ω at 1 to examine noise input features while ω is varied in the second experiment as so to investigate the effect of an increased difference in variance among classes.

8.2.2 Simulation Model 2

For the second simulation model, we exploited a simulation setup which was proposed by [72]. We used a model to generate informative features for the third and fourth experiments, $\Sigma = \omega\psi$. The variance - covariance matrix, ψ, which is a d × d matrix, is:

$$\psi = \begin{bmatrix} \sigma_{1,1} & \vartheta_{1,2} & ,\cdots, & \vartheta_{1,d} \\ \vartheta_{2,1} & \sigma_{2,2} & ,\cdots, & \vartheta_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ \vartheta_{d,1} & \vartheta_{d,2} & ,\cdots, & \sigma_{d,d} \end{bmatrix} \tag{8.2}$$

where $\vartheta_{i,j}$ are the covariance given by

$$\vartheta_{i,j} = (0.5)^{|i-j|}, i, j = 1, \cdots, d \tag{8.3}$$

where $\sigma_{i,j}$, on the diagonal of ψ, is the variance, $\sigma_{i,j} = 1$, when $\omega = 1$. The third experiment fixes ω at 1 to analyze noisy input features, whilst ω is varied in the fourth experiment to assess the impact of a greater variance difference across classes.

8.2.3 Experimental Setups

For the initial setup, data generation for the simulated experiments is conducted including both non-informative and informative features. Non-informative features are generated using a standard normal distribution, while informative features are generated based on a multivariate

normal distribution with a varying covariance structure using Simulation model 1 or 2, as Equation 8.1 or 8.2, respectively. The inclusion of a non-informative feature tests a model's robustness with respect to noise input feature. Meanwhile, adjusting the covariance structure of informative features allows for the examination of the effect of maintaining constant variance and increasing difference in variances among classes, as examined in Experiments 1 to 4. In each experiment, three distinct scenarios are designed to simulate varying degrees of overlap between the distributions of the classes. This overlap offers different levels of difficulty for feature selection methods, providing a more comprehensive evaluation of the ability of the 3cPOS and mPOS method to handle complex classification tasks. Therefore, each classification task is generated based on 4 experiments across 3 distinct scenarios, resulting in a total of 12 scenarios.

For evaluation of the 3cPOS method, 200 non-informative features and 30 informative features are generated. Besides, 900 samples are distributed evenly throughout the three classes, ensuring a balanced class distribution. The specific details of the scenarios, including the overlap characteristics, are summarized in Table 8.1.

Table 8.1: Simulation setup for the evaluation of the 3cPOS method, involving the generation of three-class classification problems.

Simulation models	Experiments	Scenarios	Class 1	Class 2	Class 3
1	1	1	$N(1,I)$	$N(3,I)$	$N(5,I)$
		2	$N(1,I)$	$N(2,I)$	$N(3,I)$
		3	$N(1,I)$	$N(1.5,I)$	$N(2,I)$
	2	4	$N(1,0.5I)$	$N(3,2I)$	$N(5,3I)$
		5	$N(1,0.5I)$	$N(2,I)$	$N(3,1.5I)$
		6	$N(1,0.5I)$	$N(1.5,0.75I)$	$N(2,I)$
2	3	7	$N(1,\varphi)$	$N(3,\varphi)$	$N(5,\varphi)$
		8	$N(1,\varphi)$	$N(2,\varphi)$	$N(3,\varphi)$
		9	$N(1,\varphi)$	$N(1.5,\varphi)$	$N(2,\varphi)$
	4	10	$N(1,0.5\varphi)$	$N(3,2\varphi)$	$N(5,3\varphi)$
		11	$N(1,0.5\varphi)$	$N(2,\varphi)$	$N(3,1.5\varphi)$
		12	$N(1,0.5\varphi)$	$N(1.5,0.75\varphi)$	$N(2,\varphi)$

For the evaluation of the mPOS method, the ambition is to generate data that accounts for different classification problems. In the case of binary classification, 100 non-informative features and 20 informative features are generated. This setup ensures that the data consists of

both irrelevant and relevant features, simulating a typical feature selection challenge. A total of 600 samples are distributed evenly across the two classes, with 300 samples in each class, to ensure balanced data, which mitigates any potential bias from class imbalances. Table 8.2 shows the specific details of the scenarios, including the overlap characteristics.

Table 8.2: Simulation setup for the mPOS method across two classification

Simulation models	Experiments	Scenarios	Class 1	Class 2
1	1	1	$N(1,I)$	$N(3,I)$
		2	$N(1,I)$	$N(2,I)$
		3	$N(1,I)$	$N(1.5,I)$
	2	4	$N(1,0.5I)$	$N(3,2I)$
		5	$N(1,0.5I)$	$N(2,I)$
		6	$N(1,0.5I)$	$N(1.5,0.75I)$
2	3	7	$N(1,\varphi)$	$N(3,\varphi)$
		8	$N(1,\varphi)$	$N(2,\varphi)$
		9	$N(1,\varphi)$	$N(1.5,\varphi)$
	4	10	$N(1,0.5\varphi)$	$N(3,2\varphi)$
		11	$N(1,0.5\varphi)$	$N(2,\varphi)$
		12	$N(1,0.5\varphi)$	$N(1.5,0.75\varphi)$

In the case of three classification, 200 non-informative features and 30 informative features are generated to confirm that the data contains both irrelevant and relevant features. A total of 900 samples are distributed evenly across the three classes, with 300 samples in each class, to ensure balanced data. Table 8.3 demonstrates the specific details of the scenarios, including the overlap characteristics.

Table 8.3: Simulation setup for the mPOS method across three classification

Simulation models	Experiments	Scenarios	Class 1	Class 2	Class 3
1	1	1	$N(1,I)$	$N(3,I)$	$N(5,I)$
		2	$N(1,I)$	$N(2,I)$	$N(3,I)$
		3	$N(1,I)$	$N(1.5,I)$	$N(2,I)$
	2	4	$N(1,0.5I)$	$N(3,2I)$	$N(5,3I)$
		5	$N(1,0.5I)$	$N(2,I)$	$N(3,1.5I)$
		6	$N(1,0.5I)$	$N(1.5,0.75I)$	$N(2,I)$
2	3	7	$N(1,\varphi)$	$N(3,\varphi)$	$N(5,\varphi)$
		8	$N(1,\varphi)$	$N(2,\varphi)$	$N(3,\varphi)$
		9	$N(1,\varphi)$	$N(1.5,\varphi)$	$N(2,\varphi)$
	4	10	$N(1,0.5\varphi)$	$N(3,2\varphi)$	$N(5,3\varphi)$
		11	$N(1,0.5\varphi)$	$N(2,\varphi)$	$N(3,1.5\varphi)$
		12	$N(1,0.5\varphi)$	$N(1.5,0.75\varphi)$	$N(2,\varphi)$

For four classification, 300 non-informative features and 40 informative features are generated to ensure that the data contains both irrelevant and relevant features. A total of 1200 samples are distributed evenly across the four classes, with 300 samples in each class, to guarantee balanced class distribution. The specific details of the scenarios, including the overlap characteristics is shown in Table 8.4 .

Table 8.4: Simulation setup for the mPOS method across four classification

Simulation models	Experiments	Scenarios	Class 1	Class 2	Class 3	Class 4
1	1	1	$N(1,I)$	$N(3,I)$	$N(5,I)$	$N(7,I)$
		2	$N(1,I)$	$N(2,I)$	$N(3,I)$	$N(4,I)$
		3	$N(1,I)$	$N(1.5,I)$	$N(2,I)$	$N(2.5,I)$
	2	4	$N(1,0.5I)$	$N(3,2I)$	$N(5,3I)$	$N(7,4I)$
		5	$N(1,0.5I)$	$N(2,I)$	$N(3,1.5I)$	$N(4,2I)$
		6	$N(1,0.5I)$	$N(1.5,0.75I)$	$N(2,I)$	$N(2.5,1.25I)$
2	3	7	$N(1,\varphi)$	$N(3,\varphi)$	$N(5,\varphi)$	$N(7,\varphi)$
		8	$N(1,\varphi)$	$N(2,\varphi)$	$N(3,\varphi)$	$N(4,\varphi)$
		9	$N(1,\varphi)$	$N(1.5,\varphi)$	$N(2,\varphi)$	$N(2.5,\varphi)$
	4	10	$N(1,0.5\varphi)$	$N(3,2\varphi)$	$N(5,3\varphi)$	$N(7,4\varphi)$
		11	$N(1,0.5\varphi)$	$N(2,\varphi)$	$N(3,1.5\varphi)$	$N(4,2\varphi)$
		12	$N(1,0.5\varphi)$	$N(1.5,0.75\varphi)$	$N(2,\varphi)$	$N(2.5,1.25\varphi)$

For five classification, 400 non-informative features and 50 informative features are generated to ensure that the data includes both irrelevant and relevant features. A total of 1500 samples are distributed evenly across the four classes, with 300 samples in each class, to a guarantee balanced class distribution. The specific details of the scenarios, including the overlap characteristics is shown in Table 8.5.

For each scenario, 10 replications of 5-fold cross-validation are performed to assess the feature selection performance. This evaluation is conducted across Random Forest (RF), k-Nearest Neighbors (k-NN), Support Vector Machines (SVM), and Extreme Gradient Boosting (XGBoost) to ensure robust performance metrics by averaging the results across multiple replications and folds. This approach achieves a comprehensive evaluation of each feature selection method under different conditions.

Table 8.5: Simulation setup for the mPOS method across five classification

Simulation models	Experiments	Scenarios	Class 1	Class 2	Class 3	Class 4	Class 5
1	1	1	$N(1,I)$	$N(3,I)$	$N(5,I)$	$N(7,I)$	$N(9,I)$
		2	$N(1,I)$	$N(2,I)$	$N(3,I)$	$N(4,I)$	$N(5,I)$
		3	$N(1,I)$	$N(1.5,I)$	$N(2,I)$	$N(2.5,I)$	$N(3,I)$
		4	$N(1,0.5I)$	$N(3,2I)$	$N(5,3I)$	$N(7,4I)$	$N(9,5I)$
	2	5	$N(1,0.5I)$	$N(2,I)$	$N(3,1.5I)$	$N(4,2I)$	$N(5,2.5I)$
		6	$N(1,0.5I)$	$N(1.5,0.75I)$	$N(2,I)$	$N(2.5,1.25I)$	$N(3,1.5I)$
2	3	7	$N(1,\varphi)$	$N(3,\varphi)$	$N(5,\varphi)$	$N(7,\varphi)$	$N(9,\varphi)$
		8	$N(1,\varphi)$	$N(2,\varphi)$	$N(3,\varphi)$	$N(4,\varphi)$	$N(5,\varphi)$
		9	$N(1,\varphi)$	$N(1.5,\varphi)$	$N(2,\varphi)$	$N(2.5,\varphi)$	$N(3,\varphi)$
	4	10	$N(1,0.5\varphi)$	$N(3,2\varphi)$	$N(5,3\varphi)$	$N(7,4\varphi)$	$N(9,5\varphi)$
		11	$N(1,0.5\varphi)$	$N(2,\varphi)$	$N(3,1.5\varphi)$	$N(4,2\varphi)$	$N(5,2.5\varphi)$
		12	$N(1,0.5\varphi)$	$N(1.5,0.75\varphi)$	$N(2,\varphi)$	$N(2.5,1.25\varphi)$	$N(3,1.5\varphi)$

8.3 Results

8.3.1 Simulation Performance of the 3cPOS Method

Based on the previously described experimental setup for simulation studies, datasets are simulated across various configurations which results in balanced class distribution and different degrees of overlaps under uncorrelated and correlated structures, Model 1 and Model 2. To evaluate its performance, a comparative analysis of feature selection performance is performed across twelve scenarios, with classification accuracy employed as the metric for evaluating the models. This analysis helps understand the behavior of the 3cPOS method. The performance is assessed across four different classifiers, as shown in Tables 8.6 - 8.17.

The average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 1** for three-class classification problems, is shown in Table 8.6. In the context of the RF classifier, 3cPOS outperforms LASSO, mRMR, and Kruskal in terms of classification accuracy across both small and moderate sets of informative genes. For the KNN classifier, 3cPOS achieves a classification accuracy of 100%, comparable to LASSO, mRMR, and Kruskal. For the SVM classifier, 3cPOS demonstrates superior performance with 100% accuracy in both small and large sets of informative features. Notably, 3cPOS exclusively outperforms all other feature selection methods in the small set of informative features.

Table 8.7 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 2** for three-class classification problems. The results indicate that the 3cPOS method outperforms all other feature selection techniques for both small and moderate sets of informative genes when evaluated with the RF and KNN classifiers. Furthermore, 3cPOS demonstrates superior performance in the small set of informative features compared to all other feature selection methods across SVM and XGBoost classifiers.

Table 8.8 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 3** for three-class classification problems. The 3cPOS method consistently outperforms all other feature selection techniques for the small set of informative genes across RF, KNN, SVM, and XGBoost classifiers.

Table 8.6: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 1** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.71	0.87	0.93	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	mRMR	0.69	0.87	0.93	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1
	Kruskal	0.72	0.88	0.93	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1
	3cPOS	0.77	0.89	0.93	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1
	full set	1															
KNN	LASSO	0.78	0.89	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	mRMR	0.79	0.89	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	
	Kruskal	0.80	0.89	0.95	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	3cPOS	0.81	0.90	0.95	0.97	0.98	0.99	1	1	1	1	1	1	1	1	1	1
	full set	1															
SVM	LASSO	0.78	0.89	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	mRMR	0.79	0.89	0.94	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	Kruskal	0.79	0.89	0.95	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	3cPOS	0.81	0.9	0.95	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	full set	1															
XGBoost	LASSO	0.71	0.86	0.92	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.97
	mRMR	0.69	0.85	0.91	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.97	0.96	0.96	0.96	0.98
	Kruskal	0.72	0.87	0.92	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	3cPOS	0.77	0.87	0.92	0.93	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.97
	full set	0.97															

Table 8.7: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 2** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.49	0.63	0.70	0.75	0.79	0.82	0.84	0.86	0.87	0.88	0.90	0.90	0.91	0.92	0.92	0.95
	mRMR	0.48	0.62	0.70	0.75	0.79	0.81	0.84	0.85	0.87	0.89	0.90	0.92	0.92	0.93	0.94	0.95
	Kruskal	0.52	0.65	0.72	0.77	0.80	0.83	0.83	0.86	0.87	0.88	0.89	0.90	0.91	0.92	0.92	0.95
	3cPOS	0.54	0.66	0.72	0.77	0.80	0.83	0.84	0.86	0.87	0.88	0.89	0.90	0.91	0.92	0.93	0.95
	full set	0.99															
KNN	LASSO	0.58	0.68	0.72	0.77	0.81	0.84	0.87	0.89	0.90	0.91	0.93	0.94	0.94	0.95	0.96	0.98
	mRMR	0.56	0.66	0.72	0.77	0.81	0.84	0.87	0.88	0.90	0.91	0.92	0.94	0.94	0.95	0.96	0.98
	Kruskal	0.58	0.69	0.75	0.80	0.83	0.87	0.88	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.98
	3cPOS	0.60	0.68	0.74	0.79	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.96	0.96	0.98
	full set	0.97															
SVM	LASSO	0.58	0.68	0.74	0.78	0.82	0.85	0.87	0.90	0.90	0.91	0.93	0.94	0.94	0.95	0.95	0.97
	mRMR	0.59	0.67	0.73	0.78	0.82	0.84	0.87	0.88	0.90	0.91	0.93	0.94	0.94	0.95	0.95	0.97
	Kruskal	0.59	0.69	0.75	0.81	0.84	0.86	0.88	0.90	0.91	0.91	0.92	0.94	0.95	0.95	0.95	0.97
	3cPOS	0.62	0.69	0.75	0.79	0.83	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	full set	0.98															
XGBoost	LASSO	0.49	0.61	0.68	0.73	0.76	0.79	0.81	0.82	0.83	0.84	0.85	0.86	0.86	0.87	0.87	0.88
	mRMR	0.48	0.60	0.68	0.73	0.76	0.79	0.81	0.82	0.83	0.84	0.85	0.86	0.87	0.88	0.88	0.88
	Kruskal	0.52	0.62	0.68	0.74	0.77	0.79	0.79	0.80	0.82	0.84	0.84	0.86	0.86	0.86	0.87	0.89
	3cPOS	0.54	0.64	0.71	0.74	0.77	0.79	0.8	0.82	0.83	0.83	0.85	0.85	0.86	0.87	0.87	0.88
	full set	0.88															

Table 8.8: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 3** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.41	0.48	0.52	0.55	0.58	0.60	0.61	0.63	0.65	0.65	0.66	0.68	0.69	0.70	0.70	0.75
	mRMR	0.38	0.45	0.50	0.54	0.57	0.60	0.61	0.63	0.65	0.67	0.68	0.69	0.70	0.71	0.72	0.75
	Kruskal	0.39	0.47	0.49	0.54	0.56	0.60	0.61	0.63	0.65	0.67	0.67	0.68	0.69	0.70	0.72	0.75
	3cPOS	0.43	0.49	0.51	0.54	0.57	0.60	0.62	0.63	0.65	0.66	0.68	0.69	0.69	0.70	0.71	0.75
	full set	0.79															
KNN	LASSO	0.45	0.50	0.54	0.57	0.60	0.62	0.64	0.65	0.67	0.68	0.69	0.71	0.72	0.73	0.74	0.79
	mRMR	0.44	0.49	0.53	0.56	0.59	0.61	0.62	0.65	0.67	0.68	0.69	0.70	0.71	0.72	0.74	0.79
	Kruskal	0.45	0.50	0.54	0.60	0.60	0.63	0.64	0.65	0.66	0.70	0.71	0.72	0.74	0.75	0.76	0.81
	3cPOS	0.46	0.50	0.54	0.56	0.59	0.61	0.63	0.65	0.67	0.68	0.70	0.71	0.72	0.73	0.74	0.79
	full set	0.73															
SVM	LASSO	0.46	0.52	0.56	0.59	0.62	0.65	0.66	0.68	0.69	0.71	0.73	0.73	0.75	0.76	0.76	0.80
	mRMR	0.46	0.51	0.55	0.58	0.61	0.63	0.65	0.67	0.68	0.70	0.72	0.73	0.74	0.75	0.76	0.81
	Kruskal	0.45	0.51	0.56	0.62	0.63	0.66	0.66	0.68	0.70	0.73	0.73	0.74	0.76	0.76	0.78	0.81
	3cPOS	0.48	0.52	0.55	0.58	0.61	0.63	0.66	0.67	0.69	0.70	0.72	0.73	0.74	0.76	0.77	0.81
	full set	0.76															
XGBoost	LASSO	0.41	0.46	0.49	0.52	0.56	0.58	0.60	0.61	0.61	0.63	0.64	0.64	0.66	0.67	0.67	0.69
	mRMR	0.38	0.42	0.47	0.50	0.53	0.56	0.58	0.59	0.60	0.63	0.64	0.64	0.65	0.65	0.66	0.69
	Kruskal	0.39	0.43	0.47	0.50	0.52	0.57	0.58	0.58	0.60	0.61	0.62	0.64	0.64	0.65	0.67	0.68
	3cPOS	0.43	0.46	0.49	0.52	0.54	0.57	0.59	0.60	0.62	0.62	0.64	0.65	0.65	0.66	0.66	0.68
	full set	0.67															

The average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 4** for three-class classification problems, is shown in Table 8.9. The results reveal that 3cPOS demonstrates superior performance compared to other techniques, achieving the highest accuracy with a single informative feature across the RF, kNN, and SVM classifiers. Furthermore, 3cPOS outperformed all other feature selection methods with a small set of informative features using the XGBoost classifier.

Table 8.10 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 5** for three-class classification problems. 3cPOS demonstrates notable effectiveness, outperforming other feature selection techniques with a small set of informative features for the RF, SVM, and XGBoost classifiers. Furthermore, 3cPOS performed better than other methods with a single informative feature when using the kNN classifier.

Table 8.11 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 6** for three-class classification problems. The 3cPOS method continues to excel, demonstrating the best performance for both small and moderate sets of informative features across all classifiers.

The average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 7** for three-class classification problems, is shown in Table 8.12. The results indicate that 3cPOS outperforms all other feature selection techniques when using a single informative feature with the RF classifier.

Table 8.13 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 8** for three-class classification problems. The 3cPOS method outperformed all other feature selection methods across RF, kNN, SVM, and XGBoost classifiers.

Table 8.14 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 9** for three-class classification problems. 3cPOS shows superior performance compared to all other feature selection techniques when using a large set of informative genes with RF, kNN, and SVM classifiers. However, its performance is comparable to other feature selection methods when used with XGBoost classifiers.

Table 8.9: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 4** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.62	0.81	0.87	0.90	0.92	0.93	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.98	0.98	0.99
	mRMR	0.61	0.79	0.86	0.90	0.92	0.93	0.94	0.96	0.96	0.97	0.97	0.98	0.98	0.98	0.99	0.99
	Kruskal	0.63	0.82	0.87	0.90	0.93	0.95	0.95	0.96	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99
	3cPOS	0.66	0.81	0.87	0.90	0.93	0.94	0.95	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99
	full set	1															
KNN	LASSO	0.72	0.83	0.88	0.90	0.92	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99	0.99
	mRMR	0.71	0.82	0.87	0.90	0.92	0.93	0.95	0.96	0.96	0.97	0.98	0.98	0.98	0.98	0.99	1
	Kruskal	0.72	0.84	0.89	0.92	0.94	0.96	0.96	0.96	0.98	0.97	0.98	0.98	0.98	0.98	0.99	1
	3cPOS	0.73	0.83	0.88	0.91	0.93	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99	1
	full set	0.98															
SVM	LASSO	0.72	0.82	0.88	0.91	0.93	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1
	mRMR	0.71	0.81	0.87	0.90	0.93	0.94	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	1
	Kruskal	0.72	0.84	0.89	0.92	0.95	0.96	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1
	3cPOS	0.75	0.83	0.88	0.92	0.94	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1
	full set	1															
XGBoost	LASSO	0.62	0.78	0.85	0.89	0.91	0.92	0.92	0.93	0.94	0.94	0.94	0.95	0.95	0.95	0.95	0.95
	mRMR	0.61	0.77	0.84	0.88	0.90	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.95	0.95	0.95
	Kruskal	0.63	0.80	0.85	0.88	0.91	0.92	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.95	0.95	0.95
	3cPOS	0.66	0.80	0.86	0.89	0.91	0.92	0.92	0.93	0.94	0.94	0.94	0.94	0.94	0.94	0.94	0.95
	full set	0.95															

Table 8.10: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 5** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.67	0.74	0.79	0.82	0.85	0.86	0.88	0.89	0.90	0.92	0.92	0.93	0.93	0.94	0.96
	mRMR	0.49	0.64	0.72	0.78	0.82	0.84	0.86	0.88	0.89	0.90	0.91	0.92	0.93	0.93	0.94	0.96
	Kruskal	0.51	0.67	0.74	0.79	0.84	0.86	0.86	0.88	0.89	0.90	0.91	0.92	0.93	0.93	0.94	0.96
	3cPOS	0.55	0.68	0.75	0.79	0.82	0.85	0.87	0.88	0.89	0.91	0.92	0.92	0.93	0.94	0.94	0.96
	full set	0.98															
KNN	LASSO	0.61	0.70	0.76	0.80	0.83	0.85	0.87	0.88	0.89	0.90	0.90	0.91	0.92	0.92	0.92	0.94
	mRMR	0.59	0.68	0.74	0.79	0.82	0.84	0.86	0.87	0.88	0.89	0.90	0.91	0.92	0.92	0.93	0.94
	Kruskal	0.59	0.71	0.77	0.81	0.84	0.86	0.88	0.89	0.89	0.90	0.91	0.92	0.92	0.93	0.93	0.95
	3cPOS	0.61	0.71	0.76	0.80	0.82	0.85	0.86	0.87	0.88	0.89	0.90	0.91	0.92	0.92	0.93	0.94
	full set	0.89															
SVM	LASSO	0.60	0.70	0.75	0.80	0.83	0.86	0.88	0.90	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	mRMR	0.60	0.68	0.74	0.79	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.96	0.97
	Kruskal	0.61	0.70	0.76	0.80	0.84	0.87	0.89	0.90	0.91	0.92	0.93	0.95	0.95	0.95	0.95	0.98
	3cPOS	0.63	0.71	0.76	0.80	0.83	0.86	0.88	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	full set	0.97															
XGBoost	LASSO	0.51	0.64	0.71	0.77	0.80	0.82	0.83	0.84	0.85	0.86	0.86	0.87	0.88	0.88	0.89	0.89
	mRMR	0.49	0.61	0.70	0.75	0.80	0.81	0.83	0.84	0.85	0.86	0.86	0.87	0.88	0.88	0.89	0.89
	Kruskal	0.51	0.63	0.71	0.76	0.80	0.82	0.82	0.83	0.84	0.86	0.86	0.86	0.87	0.87	0.88	0.89
	3cPOS	0.55	0.66	0.72	0.77	0.80	0.81	0.83	0.84	0.85	0.86	0.86	0.87	0.88	0.88	0.88	0.89
	full set	0.88															

Table 8.11: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 6** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.43	0.50	0.57	0.61	0.64	0.67	0.69	0.70	0.72	0.73	0.75	0.76	0.77	0.78	0.79	0.83
	mRMR	0.39	0.49	0.55	0.60	0.63	0.66	0.68	0.70	0.72	0.73	0.74	0.75	0.76	0.78	0.79	0.83
	Kruskal	0.40	0.49	0.54	0.60	0.64	0.66	0.67	0.69	0.71	0.74	0.74	0.75	0.76	0.77	0.78	0.82
	3cPOS	0.47	0.51	0.58	0.62	0.64	0.66	0.69	0.71	0.72	0.74	0.75	0.76	0.77	0.78	0.79	0.83
	full set	0.88															
KNN	LASSO	0.47	0.55	0.6	0.63	0.66	0.68	0.69	0.70	0.72	0.73	0.74	0.75	0.75	0.76	0.76	0.77
	mRMR	0.47	0.54	0.58	0.61	0.64	0.66	0.68	0.69	0.70	0.72	0.73	0.73	0.74	0.74	0.75	0.76
	Kruskal	0.47	0.54	0.59	0.65	0.66	0.67	0.68	0.69	0.71	0.73	0.72	0.73	0.74	0.75	0.76	0.76
	3cPOS	0.48	0.55	0.59	0.61	0.64	0.66	0.69	0.70	0.71	0.72	0.73	0.73	0.74	0.74	0.75	0.76
	full set	0.66															
SVM	LASSO	0.49	0.56	0.60	0.63	0.66	0.69	0.70	0.72	0.74	0.76	0.77	0.79	0.80	0.80	0.82	0.86
	mRMR	0.48	0.54	0.59	0.61	0.65	0.67	0.70	0.72	0.74	0.75	0.76	0.78	0.79	0.80	0.82	0.86
	Kruskal	0.48	0.54	0.6	0.66	0.67	0.69	0.71	0.73	0.75	0.77	0.78	0.78	0.80	0.81	0.83	0.86
	3cPOS	0.51	0.56	0.59	0.62	0.65	0.67	0.69	0.72	0.73	0.75	0.77	0.78	0.79	0.80	0.82	0.85
	full set	0.83															
XGBoost	LASSO	0.43	0.49	0.55	0.59	0.62	0.65	0.66	0.67	0.68	0.69	0.70	0.72	0.73	0.73	0.74	0.77
	mRMR	0.39	0.46	0.51	0.55	0.60	0.62	0.64	0.67	0.68	0.69	0.70	0.72	0.73	0.74	0.74	0.76
	Kruskal	0.40	0.48	0.52	0.58	0.60	0.64	0.65	0.67	0.68	0.69	0.70	0.71	0.72	0.72	0.73	0.75
	3cPOS	0.47	0.50	0.54	0.58	0.61	0.64	0.66	0.67	0.69	0.70	0.71	0.71	0.72	0.73	0.73	0.76
	full set	0.74															

Table 8.12: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 7** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.71	0.88	0.93	0.95	0.96	0.97	0.98	0.98	0.98	0.99	0.99	1	0.99	0.99	0.99	0.99
	mRMR	0.70	0.87	0.92	0.94	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	Kruskal	0.70	0.78	0.86	0.89	0.91	0.93	0.94	0.95	0.95	0.96	0.96	0.96	0.97	0.97	0.98	0.99
	3cPOS	0.71	0.86	0.91	0.94	0.95	0.97	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99
	full set	1															
KNN	LASSO	0.79	0.89	0.94	0.97	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	mRMR	0.79	0.89	0.94	0.96	0.98	0.98	0.99	0.99	0.99	0.99	1	0.99	1	1	1	1
	Kruskal	0.79	0.82	0.87	0.9	0.92	0.95	0.95	0.96	0.97	0.97	0.97	0.98	0.98	0.99	0.99	1
	3cPOS	0.76	0.88	0.93	0.96	0.97	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	full set	1															
SVM	LASSO	0.79	0.90	0.94	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	mRMR	0.79	0.89	0.94	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	Kruskal	0.70	0.76	0.84	0.87	0.90	0.92	0.92	0.93	0.93	0.93	0.94	0.94	0.94	0.94	0.94	0.95
	3cPOS	0.78	0.88	0.93	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1
	full set	1															
XGBoost	LASSO	0.79	0.90	0.94	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	mRMR	0.70	0.86	0.91	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.97	0.97	0.97	0.96
	Kruskal	0.70	0.76	0.84	0.87	0.90	0.92	0.92	0.93	0.93	0.93	0.94	0.94	0.94	0.94	0.94	0.95
	3cPOS	0.71	0.84	0.90	0.93	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96
	full set	0.96															

Table 8.13: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 8** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.65	0.72	0.77	0.8	0.83	0.85	0.86	0.87	0.87	0.88	0.88	0.89	0.89	0.89	0.91
	mRMR	0.49	0.61	0.70	0.75	0.78	0.80	0.82	0.84	0.85	0.86	0.86	0.87	0.87	0.88	0.88	0.90
	Kruskal	0.51	0.57	0.61	0.65	0.67	0.71	0.72	0.73	0.74	0.77	0.77	0.78	0.79	0.80	0.81	0.85
	3cPOS	0.54	0.66	0.72	0.77	0.80	0.83	0.84	0.86	0.87	0.88	0.89	0.90	0.91	0.92	0.93	0.95
	full set	0.90															
KNN	LASSO	0.58	0.68	0.74	0.79	0.83	0.85	0.86	0.87	0.88	0.89	0.89	0.90	0.91	0.91	0.91	0.92
	mRMR	0.56	0.66	0.73	0.77	0.80	0.82	0.84	0.86	0.87	0.88	0.88	0.89	0.90	0.90	0.90	0.92
	Kruskal	0.57	0.59	0.65	0.66	0.68	0.71	0.73	0.75	0.76	0.77	0.79	0.80	0.81	0.82	0.83	0.86
	3cPOS	0.60	0.68	0.74	0.79	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.96	0.96	0.98
	full set	0.88															
SVM	LASSO	0.59	0.69	0.75	0.8	0.83	0.85	0.86	0.88	0.89	0.89	0.9	0.9	0.91	0.91	0.91	0.92
	mRMR	0.59	0.67	0.74	0.78	0.81	0.83	0.85	0.86	0.87	0.88	0.89	0.89	0.9	0.9	0.91	0.91
	Kruskal	0.59	0.61	0.66	0.67	0.7	0.73	0.74	0.76	0.77	0.79	0.8	0.81	0.81	0.82	0.84	0.86
	3cPOS	0.62	0.69	0.75	0.79	0.83	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	full set	0.84															
XGBoost	LASSO	0.51	0.63	0.69	0.75	0.77	0.79	0.81	0.82	0.83	0.83	0.84	0.84	0.84	0.84	0.84	0.84
	mRMR	0.49	0.60	0.67	0.72	0.74	0.77	0.78	0.79	0.81	0.81	0.82	0.82	0.82	0.83	0.83	0.83
	Kruskal	0.51	0.54	0.59	0.63	0.65	0.68	0.69	0.70	0.71	0.73	0.74	0.75	0.74	0.76	0.76	0.79
	3cPOS	0.54	0.64	0.71	0.74	0.77	0.79	0.80	0.82	0.83	0.83	0.85	0.85	0.86	0.87	0.87	0.88
	full set	0.81															

Table 8.14: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 9** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.40	0.47	0.51	0.53	0.55	0.57	0.59	0.59	0.60	0.60	0.61	0.62	0.63	0.63	0.64	0.65
	mRMR	0.39	0.45	0.49	0.53	0.56	0.58	0.59	0.60	0.61	0.61	0.62	0.63	0.63	0.64	0.65	0.66
	Kruskal	0.37	0.38	0.43	0.45	0.48	0.49	0.49	0.50	0.54	0.55	0.55	0.56	0.58	0.59	0.60	0.63
	3cPOS	0.38	0.43	0.48	0.51	0.53	0.55	0.57	0.58	0.59	0.59	0.60	0.62	0.63	0.63	0.63	0.66
	full set	0.66															
KNN	LASSO	0.44	0.52	0.53	0.55	0.57	0.58	0.59	0.60	0.60	0.61	0.62	0.62	0.63	0.64	0.64	0.64
	mRMR	0.44	0.50	0.53	0.55	0.57	0.58	0.60	0.61	0.62	0.62	0.63	0.64	0.63	0.64	0.65	0.65
	Kruskal	0.44	0.45	0.48	0.47	0.50	0.49	0.52	0.51	0.54	0.55	0.55	0.56	0.57	0.59	0.59	0.63
	3cPOS	0.43	0.47	0.5	0.52	0.54	0.56	0.57	0.59	0.59	0.60	0.61	0.62	0.62	0.62	0.63	0.65
	full set	0.62															
SVM	LASSO	0.46	0.52	0.54	0.57	0.58	0.60	0.61	0.62	0.63	0.64	0.64	0.64	0.65	0.65	0.65	0.66
	mRMR	0.44	0.50	0.53	0.55	0.57	0.58	0.60	0.61	0.62	0.62	0.63	0.64	0.63	0.64	0.65	0.65
	Kruskal	0.47	0.47	0.49	0.50	0.51	0.51	0.52	0.53	0.56	0.56	0.56	0.57	0.58	0.58	0.60	0.64
	3cPOS	0.44	0.48	0.52	0.54	0.56	0.58	0.59	0.60	0.61	0.61	0.62	0.63	0.63	0.63	0.64	0.66
	full set	0.57															
XGBoost	LASSO	0.40	0.43	0.47	0.51	0.52	0.53	0.54	0.55	0.56	0.56	0.56	0.57	0.57	0.58	0.58	0.59
	mRMR	0.39	0.43	0.45	0.49	0.52	0.53	0.54	0.55	0.56	0.57	0.57	0.58	0.58	0.59	0.59	0.59
	Kruskal	0.37	0.38	0.41	0.42	0.44	0.45	0.45	0.46	0.49	0.51	0.5	0.52	0.53	0.54	0.55	0.57
	3cPOS	0.38	0.42	0.44	0.47	0.49	0.52	0.53	0.54	0.54	0.54	0.56	0.57	0.57	0.57	0.58	0.58
	full set	0.57															

The average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 10** for three-class classification problems, is shown in Table 8.15. The results show that 3cPOS performs the best with a single set of informative features in the RF and XGBoost classifiers. In contrast, 3cPOS shows comparable performance to LASSO and mRMR when applied to the kNN and SVM classifiers.

Table 8.16 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 11** for three-class classification problems. 3cPOS is found to be comparable to other feature selection methods across all four classifiers.

Table 8.17 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 12** for three-class classification problems. The 3cPOS method is comparable to other feature selection techniques across all four classifiers.

Table 8.15: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 10** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.61	0.82	0.87	0.90	0.92	0.93	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.97
	mRMR	0.61	0.79	0.85	0.89	0.91	0.92	0.93	0.95	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.98
	Kruskal	0.62	0.73	0.78	0.84	0.85	0.89	0.89	0.90	0.90	0.90	0.92	0.92	0.93	0.93	0.94	0.96
	3cPOS	0.64	0.78	0.84	0.87	0.90	0.91	0.92	0.92	0.93	0.94	0.94	0.95	0.95	0.95	0.96	0.97
	full set	0.98															
KNN	LASSO	0.70	0.84	0.88	0.91	0.93	0.94	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.97	0.97	0.97
	mRMR	0.70	0.81	0.86	0.89	0.92	0.93	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.97
	Kruskal	0.70	0.75	0.79	0.82	0.85	0.86	0.87	0.87	0.89	0.90	0.90	0.91	0.91	0.91	0.92	0.95
	3cPOS	0.70	0.80	0.85	0.87	0.89	0.90	0.91	0.92	0.92	0.93	0.94	0.94	0.94	0.94	0.95	0.96
	full set	0.94															
SVM	LASSO	0.72	0.84	0.88	0.91	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.98
	mRMR	0.71	0.81	0.86	0.90	0.93	0.94	0.95	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.98	0.98
	Kruskal	0.71	0.74	0.78	0.82	0.84	0.87	0.89	0.90	0.90	0.91	0.92	0.93	0.93	0.94	0.94	0.96
	3cPOS	0.71	0.8	0.85	0.88	0.90	0.91	0.92	0.93	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.97
	full set	0.97															
XGBoost	LASSO	0.61	0.80	0.86	0.89	0.91	0.92	0.92	0.92	0.92	0.93	0.93	0.93	0.93	0.93	0.93	0.93
	mRMR	0.61	0.76	0.84	0.87	0.89	0.90	0.91	0.92	0.92	0.92	0.93	0.92	0.93	0.93	0.94	0.93
	Kruskal	0.62	0.70	0.75	0.82	0.83	0.86	0.87	0.87	0.88	0.88	0.89	0.89	0.90	0.90	0.90	0.91
	3cPOS	0.64	0.76	0.83	0.85	0.88	0.89	0.89	0.90	0.90	0.91	0.91	0.91	0.91	0.92	0.92	0.92
	full set	0.92															

Table 8.16: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 11** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.64	0.72	0.77	0.80	0.81	0.83	0.84	0.85	0.86	0.87	0.87	0.88	0.88	0.89	0.90
	mRMR	0.48	0.64	0.72	0.76	0.79	0.82	0.84	0.85	0.87	0.88	0.89	0.89	0.90	0.90	0.90	0.91
	Kruskal	0.51	0.57	0.64	0.68	0.70	0.74	0.75	0.76	0.79	0.80	0.81	0.81	0.82	0.83	0.84	0.88
	3cPOS	0.51	0.61	0.69	0.74	0.76	0.79	0.80	0.82	0.83	0.84	0.84	0.85	0.86	0.87	0.87	0.89
	full set	0.93															
KNN	LASSO	0.60	0.68	0.74	0.78	0.80	0.81	0.83	0.83	0.84	0.84	0.85	0.85	0.85	0.85	0.86	0.86
	mRMR	0.59	0.69	0.75	0.78	0.81	0.82	0.83	0.84	0.85	0.85	0.86	0.86	0.86	0.86	0.86	0.85
	Kruskal	0.59	0.62	0.66	0.68	0.70	0.72	0.73	0.74	0.75	0.78	0.79	0.79	0.80	0.80	0.80	0.83
	3cPOS	0.58	0.66	0.71	0.75	0.77	0.78	0.79	0.80	0.81	0.82	0.82	0.82	0.83	0.83	0.84	0.85
	full set	0.81															
SVM	LASSO	0.61	0.68	0.73	0.78	0.8	0.82	0.83	0.85	0.86	0.86	0.87	0.88	0.88	0.88	0.89	0.90
	mRMR	0.60	0.69	0.75	0.78	0.81	0.83	0.85	0.85	0.86	0.87	0.88	0.89	0.89	0.89	0.89	0.89
	Kruskal	0.61	0.62	0.64	0.67	0.70	0.70	0.72	0.73	0.77	0.77	0.78	0.78	0.80	0.81	0.82	0.86
	3cPOS	0.59	0.66	0.71	0.74	0.77	0.79	0.81	0.82	0.83	0.84	0.85	0.85	0.86	0.87	0.87	0.89
	full set	0.84															
XGBoost	LASSO	0.51	0.62	0.69	0.74	0.76	0.77	0.79	0.8	0.81	0.82	0.82	0.82	0.83	0.83	0.83	0.84
	mRMR	0.50	0.63	0.71	0.74	0.77	0.79	0.81	0.81	0.82	0.83	0.83	0.83	0.84	0.84	0.83	0.84
	Kruskal	0.51	0.54	0.60	0.64	0.68	0.70	0.72	0.73	0.74	0.77	0.77	0.76	0.77	0.78	0.79	0.82
	3cPOS	0.51	0.60	0.67	0.71	0.73	0.75	0.76	0.78	0.79	0.79	0.80	0.81	0.81	0.81	0.82	0.83
	full set	0.83															

Table 8.17: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 12** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.41	0.50	0.55	0.59	0.62	0.64	0.67	0.67	0.68	0.69	0.70	0.71	0.71	0.71	0.71	0.73
	mRMR	0.40	0.49	0.55	0.58	0.62	0.64	0.66	0.67	0.68	0.70	0.70	0.71	0.72	0.72	0.72	0.73
	Kruskal	0.42	0.46	0.52	0.54	0.54	0.56	0.58	0.58	0.60	0.60	0.61	0.62	0.65	0.66	0.67	0.7
	3cPOS	0.41	0.48	0.54	0.57	0.6	0.62	0.64	0.65	0.66	0.67	0.67	0.68	0.68	0.69	0.70	0.73
	full set	0.75															
KNN	LASSO	0.48	0.53	0.57	0.60	0.63	0.65	0.66	0.67	0.67	0.67	0.68	0.68	0.68	0.67	0.68	0.67
	mRMR	0.47	0.53	0.58	0.61	0.63	0.64	0.65	0.66	0.66	0.67	0.68	0.68	0.68	0.68	0.68	0.69
	Kruskal	0.50	0.50	0.53	0.53	0.56	0.55	0.57	0.58	0.58	0.59	0.59	0.61	0.61	0.62	0.62	0.65
	3cPOS	0.46	0.54	0.57	0.59	0.61	0.62	0.64	0.64	0.65	0.66	0.67	0.66	0.67	0.66	0.67	0.67
	full set	0.62															
SVM	LASSO	0.48	0.55	0.58	0.61	0.63	0.65	0.67	0.68	0.69	0.70	0.70	0.71	0.71	0.71	0.71	0.71
	mRMR	0.48	0.55	0.58	0.62	0.63	0.65	0.66	0.68	0.69	0.71	0.72	0.72	0.72	0.73	0.73	0.73
	Kruskal	0.50	0.50	0.52	0.53	0.54	0.56	0.57	0.58	0.60	0.60	0.61	0.62	0.64	0.65	0.65	0.68
	3cPOS	0.47	0.53	0.57	0.59	0.62	0.64	0.65	0.66	0.66	0.68	0.68	0.69	0.69	0.70	0.71	0.72
	full set	0.64															
XGBoost	LASSO	0.40	0.47	0.53	0.56	0.59	0.61	0.62	0.63	0.65	0.66	0.66	0.65	0.66	0.66	0.66	0.67
	mRMR	0.40	0.46	0.52	0.56	0.58	0.60	0.62	0.63	0.64	0.66	0.67	0.67	0.67	0.68	0.68	0.68
	Kruskal	0.42	0.45	0.47	0.50	0.51	0.53	0.54	0.55	0.56	0.56	0.58	0.59	0.60	0.61	0.62	0.65
	3cPOS	0.41	0.46	0.51	0.54	0.57	0.58	0.60	0.61	0.62	0.63	0.63	0.64	0.65	0.65	0.65	0.67
	full set	0.66															

8.3.2 Simulation Performance of the mPOS Method

Balanced class distribution and varying levels of overlap under uncorrelated and correlated structures, Model 1 and Model 2, are achieved through the use of different configurations, as described in Section 8.2.3. To understand the behavior of the mPOS method in comparison to alternative feature selection techniques across several scenarios, a comparative analysis of feature selection performance is conducted across four different classifiers.

For the evaluation of the mPOS method in binary classification problems, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers across Scenario 1 to Scenario 12 is shown in Tables 8.18 - Table 8.29, respectively. For **Scenario 1**, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers for binary-class classification problems, is shown in Table 8.18. mPOS achieves comparable performance to alternative feature selection techniques across four classifiers and it provides classification accuracy up 100% when considering larger set sizes of informative features.

Table 8.19 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 2** for binary-class classification problems. mPOS remains comparative with LASSO and mRMR methods at a single informative feature. It shows comparable performance to LASSO, mRMR, and Wilcoxon techniques at the moderate and larger set sizes of informative feature across RF classifier. For KNN and SVM classifier, mPOS achieves comparable performance to alternative feature selection techniques when considering higher set sizes of informative features. Although the performance of mPOS are comparable to LASSO and Wilcoxon at a single informative feature, mRMR performs better than other feature selection techniques at the small set sizes of informative features across the XGBoost classifier.

Table 8.20 shows the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 3** for binary-class classification problems. mPOS demonstrates superior performance in comparison to other feature selection techniques across varying set sizes of informative features. This superiority is consistent across four different classifiers, with the mPOS approach yielding classification accuracy rates ranging from 88% to 97%.

Table 8.18: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 1** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.77	0.91	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	mRMR	0.77	0.91	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	Wilcoxon	0.77	0.89	0.94	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	mPOS	0.77	0.89	0.94	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	full set	1															
KNN	LASSO	0.83	0.92	0.95	0.97	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1	1	1
	mRMR	0.84	0.92	0.96	0.98	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	Wilcoxon	0.81	0.91	0.95	0.97	0.99	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	mPOS	0.81	0.9	0.95	0.97	0.99	0.99	1	1	1	1	1	1	1	1	1	1
	full set	1															
SVM	LASSO	0.83	0.92	0.95	0.97	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	mRMR	0.85	0.92	0.96	0.98	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	Wilcoxon	0.82	0.91	0.95	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	mPOS	0.82	0.91	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	full set	1															
XGBoost	LASSO	0.77	0.90	0.94	0.95	0.96	0.96	0.97	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	mRMR	0.77	0.90	0.94	0.95	0.96	0.96	0.97	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	Wilcoxon	0.77	0.88	0.93	0.94	0.95	0.95	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	mPOS	0.77	0.88	0.92	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.97
	full set	0.97															

Table 8.19: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 2** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.62	0.7	0.78	0.82	0.84	0.86	0.88	0.90	0.91	0.92	0.93	0.93	0.94	0.95	0.95	0.97
	mRMR	0.58	0.73	0.79	0.83	0.85	0.87	0.89	0.90	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.97
	Wilcoxon	0.62	0.72	0.78	0.80	0.81	0.84	0.87	0.88	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.97
	mPOS	0.62	0.70	0.76	0.80	0.83	0.86	0.88	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.95	0.97
	full set	0.98															
KNN	LASSO	0.67	0.74	0.8	0.83	0.85	0.87	0.90	0.92	0.93	0.94	0.94	0.95	0.95	0.96	0.96	0.98
	mRMR	0.70	0.78	0.82	0.85	0.87	0.88	0.90	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.96	0.98
	Wilcoxon	0.66	0.77	0.79	0.82	0.84	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.95	0.96	0.98
	mPOS	0.65	0.72	0.78	0.82	0.85	0.88	0.90	0.91	0.93	0.94	0.94	0.95	0.96	0.96	0.96	0.98
	full set	0.96															
SVM	LASSO	0.68	0.75	0.80	0.83	0.85	0.87	0.90	0.91	0.93	0.93	0.94	0.95	0.95	0.96	0.96	0.97
	mRMR	0.72	0.78	0.82	0.86	0.88	0.89	0.91	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.96	0.97
	Wilcoxon	0.68	0.77	0.8	0.83	0.85	0.87	0.90	0.91	0.92	0.94	0.94	0.95	0.95	0.95	0.96	0.97
	mPOS	0.66	0.73	0.79	0.82	0.86	0.88	0.90	0.92	0.93	0.94	0.95	0.95	0.96	0.96	0.96	0.97
	full set	0.95															
XGBoost	LASSO	0.62	0.68	0.76	0.8	0.82	0.84	0.85	0.86	0.87	0.87	0.88	0.88	0.89	0.89	0.89	0.90
	mRMR	0.58	0.69	0.76	0.81	0.83	0.84	0.85	0.86	0.87	0.88	0.88	0.89	0.88	0.89	0.89	0.89
	Wilcoxon	0.62	0.69	0.74	0.77	0.79	0.81	0.84	0.85	0.87	0.88	0.88	0.88	0.88	0.88	0.89	0.90
	mPOS	0.62	0.68	0.74	0.78	0.82	0.82	0.84	0.86	0.86	0.87	0.88	0.88	0.88	0.89	0.89	0.90
	full set	0.89															

Table 8.20: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 3** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.54	0.57	0.60	0.62	0.64	0.65	0.67	0.68	0.70	0.71	0.71	0.71	0.72	0.74	0.75
	mRMR	0.51	0.56	0.60	0.64	0.67	0.69	0.70	0.72	0.73	0.75	0.76	0.77	0.78	0.79	0.79	0.82
	Wilcoxon	0.53	0.59	0.62	0.64	0.64	0.68	0.70	0.71	0.74	0.75	0.77	0.77	0.77	0.77	0.79	0.82
	mPOS	0.60	0.68	0.74	0.78	0.81	0.84	0.85	0.87	0.88	0.89	0.91	0.92	0.92	0.93	0.93	0.95
	full set	0.96															
KNN	LASSO	0.55	0.60	0.63	0.66	0.67	0.69	0.71	0.73	0.75	0.76	0.77	0.78	0.79	0.80	0.81	0.82
	mRMR	0.56	0.61	0.64	0.68	0.70	0.71	0.71	0.72	0.74	0.75	0.76	0.77	0.78	0.78	0.79	0.84
	Wilcoxon	0.54	0.64	0.66	0.65	0.66	0.70	0.72	0.74	0.75	0.76	0.78	0.78	0.78	0.77	0.79	0.84
	mPOS	0.63	0.71	0.77	0.80	0.83	0.85	0.87	0.89	0.90	0.91	0.91	0.92	0.92	0.92	0.92	0.94
	full set	0.91															
SVM	LASSO	0.58	0.62	0.66	0.68	0.70	0.71	0.73	0.75	0.76	0.77	0.79	0.80	0.81	0.82	0.83	0.84
	mRMR	0.61	0.65	0.68	0.7	0.72	0.74	0.75	0.75	0.76	0.78	0.79	0.79	0.79	0.80	0.81	0.85
	Wilcoxon	0.59	0.66	0.67	0.67	0.70	0.71	0.74	0.75	0.75	0.78	0.78	0.79	0.79	0.80	0.81	0.85
	mPOS	0.65	0.71	0.76	0.80	0.84	0.86	0.88	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.95	0.97
	full set	0.93															
XGBoost	LASSO	0.57	0.58	0.60	0.62	0.64	0.66	0.68	0.70	0.71	0.72	0.72	0.72	0.74	0.74	0.75	0.75
	mRMR	0.51	0.54	0.57	0.60	0.62	0.64	0.65	0.67	0.68	0.70	0.71	0.71	0.71	0.72	0.74	0.75
	Wilcoxon	0.53	0.56	0.57	0.60	0.61	0.62	0.67	0.69	0.70	0.71	0.70	0.72	0.73	0.72	0.74	0.75
	mPOS	0.60	0.67	0.72	0.76	0.79	0.81	0.82	0.82	0.84	0.85	0.86	0.86	0.86	0.87	0.87	0.88
	full set	0.86															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 4** for binary-class classification problems, is shown in Table 8.21. mPOS shows comparable performance to alternative feature selection techniques at the different set sizes of informative feature across RF and XGBoost classifiers. However, for KNN and SVM classifiers, mRMR performs better than other feature selection techniques at the small set sizes of informative features. Despite this, mPOS remains comparative with other feature selection techniques when considering larger set sizes of informative feature.

Table 8.22 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 5** for binary-class classification problems. For a RF classifier, mPOS remains comparative with mRMR and Wilcoxon at the moderate and large set sizes of informative features. For the KNN and SVM classifier, even though mRMR achieves superior performance at the small set sizes of informative features, mPOS shows comparable performance to other feature selection techniques when considering larger set sizes of informative features. According to a XGBoost classifier, mPOS achieves comparable performance to alternative feature selection techniques at the different set sizes of informative feature.

Table 8.23 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 6** for binary-class classification problems. mRMR outperforms other feature selection techniques at the smaller set sizes of informative features across four classifiers. However, mPOS shows comparable performance to mRMR, Wilcoxon and other feature selection techniques when evaluated with RF and KNN, SVM, and XGBoost classifiers, respectively.

Table 8.21: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 4** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.77	0.91	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	mRMR	0.75	0.92	0.95	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1	1
	Wilcoxon	0.75	0.90	0.95	0.97	0.97	0.99	0.99	0.99	0.99	1	1	1	1	1	1	1
	mPOS	0.77	0.89	0.94	0.96	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1
	full set	1															
KNN	LASSO	0.84	0.92	0.95	0.96	0.97	0.98	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	mRMR	0.85	0.93	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	Wilcoxon	0.83	0.92	0.94	0.96	0.97	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1	0.99
	mPOS	0.80	0.90	0.94	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	full set	0.99															
SVM	LASSO	0.84	0.91	0.95	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1
	mRMR	0.85	0.92	0.95	0.97	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	Wilcoxon	0.84	0.91	0.94	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	mPOS	0.81	0.9	0.94	0.96	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1
	full set	1															
XGBoost	LASSO	0.77	0.89	0.94	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	mRMR	0.75	0.90	0.95	0.97	0.97	0.98	0.98	0.97	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.97
	Wilcoxon	0.75	0.88	0.94	0.96	0.96	0.96	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.98	0.97
	mPOS	0.77	0.88	0.93	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	full set	0.96															

Table 8.22: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 5** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.63	0.75	0.83	0.87	0.88	0.89	0.91	0.91	0.92	0.93	0.93	0.93	0.93	0.94	0.94	0.93
	mRMR	0.63	0.79	0.86	0.89	0.90	0.92	0.93	0.94	0.95	0.95	0.96	0.97	0.97	0.97	0.98	0.99
	Wilcoxon	0.62	0.78	0.82	0.87	0.89	0.90	0.92	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.98
	mPOS	0.63	0.76	0.82	0.85	0.88	0.90	0.92	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.98
	full set	0.99															
KNN	LASSO	0.72	0.8	0.85	0.88	0.90	0.91	0.92	0.93	0.93	0.94	0.95	0.95	0.95	0.96	0.96	0.97
	mRMR	0.74	0.83	0.88	0.90	0.91	0.91	0.92	0.93	0.93	0.93	0.94	0.95	0.95	0.95	0.96	0.97
	Wilcoxon	0.72	0.81	0.85	0.87	0.88	0.90	0.92	0.93	0.94	0.94	0.94	0.95	0.95	0.95	0.95	0.97
	mPOS	0.70	0.78	0.83	0.86	0.89	0.90	0.92	0.93	0.93	0.95	0.95	0.96	0.96	0.96	0.96	0.97
	full set	0.95															
SVM	LASSO	0.72	0.79	0.85	0.87	0.90	0.91	0.93	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.98	0.98
	mRMR	0.76	0.82	0.86	0.89	0.90	0.92	0.93	0.94	0.95	0.95	0.96	0.97	0.97	0.98	0.98	0.98
	Wilcoxon	0.72	0.80	0.83	0.86	0.88	0.91	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.98	0.98	0.98
	mPOS	0.70	0.78	0.83	0.86	0.89	0.91	0.93	0.94	0.95	0.96	0.97	0.97	0.97	0.98	0.98	0.98
	full set	0.98															
XGBoost	LASSO	0.62	0.76	0.81	0.85	0.87	0.88	0.89	0.90	0.91	0.91	0.91	0.92	0.92	0.92	0.93	0.93
	mRMR	0.63	0.75	0.83	0.87	0.88	0.89	0.91	0.91	0.92	0.93	0.93	0.93	0.93	0.94	0.94	0.93
	Wilcoxon	0.62	0.75	0.80	0.85	0.87	0.87	0.89	0.90	0.91	0.92	0.92	0.92	0.92	0.92	0.92	0.92
	mPOS	0.63	0.74	0.80	0.83	0.85	0.87	0.88	0.90	0.90	0.91	0.92	0.92	0.92	0.92	0.93	0.93
	full set	0.91															

Table 8.23: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 6** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.56	0.62	0.66	0.69	0.71	0.73	0.75	0.75	0.76	0.78	0.79	0.79	0.80	0.80	0.81	0.82
	mRMR	0.54	0.62	0.69	0.72	0.75	0.77	0.79	0.80	0.81	0.83	0.83	0.85	0.86	0.87	0.88	0.90
	Wilcoxon	0.55	0.66	0.70	0.72	0.73	0.74	0.77	0.78	0.81	0.83	0.84	0.85	0.85	0.85	0.87	0.90
	mPOS	0.56	0.63	0.67	0.71	0.74	0.76	0.78	0.79	0.8	0.81	0.83	0.84	0.85	0.86	0.87	0.90
	full set	0.90															
KNN	LASSO	0.59	0.65	0.70	0.73	0.75	0.77	0.78	0.79	0.80	0.82	0.82	0.83	0.82	0.83	0.83	0.82
	mRMR	0.62	0.68	0.73	0.74	0.77	0.78	0.78	0.78	0.80	0.80	0.81	0.81	0.81	0.81	0.82	0.82
	Wilcoxon	0.59	0.69	0.70	0.72	0.73	0.75	0.78	0.78	0.80	0.81	0.81	0.81	0.80	0.80	0.82	0.82
	mPOS	0.58	0.66	0.69	0.72	0.75	0.77	0.78	0.79	0.80	0.81	0.82	0.82	0.82	0.82	0.82	0.82
	full set	0.78															
SVM	LASSO	0.62	0.65	0.69	0.73	0.75	0.77	0.79	0.81	0.82	0.84	0.85	0.87	0.87	0.88	0.89	0.89
	mRMR	0.64	0.69	0.73	0.75	0.77	0.79	0.80	0.81	0.82	0.83	0.84	0.85	0.86	0.87	0.88	0.90
	Wilcoxon	0.62	0.70	0.72	0.72	0.74	0.76	0.80	0.80	0.82	0.84	0.85	0.86	0.86	0.86	0.86	0.90
	mPOS	0.61	0.65	0.69	0.73	0.76	0.78	0.80	0.81	0.82	0.83	0.85	0.85	0.86	0.87	0.88	0.90
	full set	0.83															
XGBoost	LASSO	0.56	0.62	0.66	0.69	0.71	0.73	0.75	0.75	0.76	0.78	0.79	0.79	0.80	0.80	0.81	0.82
	mRMR	0.54	0.59	0.65	0.68	0.71	0.72	0.74	0.75	0.77	0.78	0.79	0.79	0.80	0.81	0.81	0.82
	Wilcoxon	0.55	0.62	0.67	0.70	0.70	0.71	0.74	0.76	0.77	0.79	0.8	0.81	0.81	0.80	0.82	0.82
	mPOS	0.56	0.61	0.65	0.69	0.71	0.72	0.74	0.75	0.76	0.78	0.78	0.79	0.79	0.81	0.82	0.82
	full set	0.79															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 7** for binary-class classification problems, is shown in Table 8.24. For RF and XGBoost classifier, both mPOS and Wilcoxon show superior performance at a single informative feature. According to KNN and SVM classifiers, Wilcoxon performs better than alternative feature selection techniques at a single informative feature. In contrast, mPOS shows comparable performance to other feature techniques when larger set sizes of informative features are considered across all four classifiers.

Table 8.25 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 8** for binary-class classification problems. mPOS shows comparable performance to LASSO and Wilcoxon at a single informative feature. However, mRMR, Wilcoxon, and mPOS show superior performance when considering larger set sizes of informative features across a RF classifier. For the KNN classifier, mPOS maintains a comparative performance with alternative feature selection techniques at the different set sizes of informative features. Even though LASSO show superior performance at the smaller set sizes of informative feature, it achieves a comparable performance to alternative feature selection techniques when considering larger set sizes of informative features across the SVM classifier. LASSO outperforms all other feature selection techniques at the small set sizes of informative features, but its performance remains comparative with other feature selection techniques when larger set sizes of informative features are applied across the XGBoost classifier.

Table 8.26 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 9** for binary-class classification problems. For four classifiers, Wilcoxon outperforms all other feature selection techniques at smaller set sizes of informative features. Despite this, mPOS shows comparable performance to mRMR and Wilcoxon when considering larger set sizes of informative features.

Table 8.24: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 7** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.76	0.87	0.93	0.95	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.96	0.97	0.96	0.96	0.96
	mRMR	0.76	0.89	0.93	0.96	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
	Wilcoxon	0.8	0.85	0.92	0.92	0.93	0.95	0.95	0.96	0.97	0.97	0.97	0.98	0.98	0.99	0.99	1
	mPOS	0.79	0.87	0.92	0.94	0.95	0.97	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1
	full set	1															
KNN	LASSO	0.84	0.92	0.95	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	mRMR	0.84	0.91	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1	1
	Wilcoxon	0.87	0.88	0.93	0.93	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.99	0.99	0.99	1
	mPOS	0.84	0.89	0.93	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1
	full set	1															
SVM	LASSO	0.84	0.92	0.95	0.97	0.98	0.99	0.99	0.99	0.99	1	1	1	0.99	1	1	1
	mRMR	0.85	0.91	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
	Wilcoxon	0.87	0.89	0.93	0.93	0.95	0.95	0.95	0.96	0.97	0.97	0.97	0.97	0.98	0.99	0.99	1
	mPOS	0.84	0.89	0.93	0.95	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
	full set	1															
XGBoost	LASSO	0.78	0.9	0.93	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96
	mRMR	0.76	0.87	0.93	0.95	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.96	0.97	0.96	0.96	0.96
	Wilcoxon	0.80	0.84	0.91	0.92	0.92	0.93	0.94	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96
	mPOS	0.79	0.86	0.91	0.93	0.94	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96
	full set	0.96															

Table 8.25: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 8** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.62	0.70	0.77	0.81	0.83	0.84	0.85	0.84	0.85	0.85	0.85	0.85	0.85	0.85	0.85	0.86
	mRMR	0.58	0.70	0.76	0.79	0.82	0.84	0.86	0.87	0.88	0.89	0.89	0.89	0.90	0.90	0.90	0.90
	Wilcoxon	0.63	0.69	0.76	0.78	0.79	0.80	0.82	0.82	0.83	0.84	0.84	0.85	0.86	0.86	0.88	0.90
	mPOS	0.63	0.7	0.74	0.77	0.79	0.82	0.83	0.84	0.86	0.86	0.87	0.88	0.89	0.89	0.89	0.90
	full set	0.91															
KNN	LASSO	0.67	0.76	0.81	0.84	0.86	0.87	0.88	0.89	0.89	0.90	0.90	0.90	0.91	0.91	0.91	0.92
	mRMR	0.69	0.74	0.79	0.81	0.84	0.86	0.87	0.88	0.89	0.90	0.91	0.91	0.91	0.91	0.92	0.92
	Wilcoxon	0.69	0.75	0.77	0.78	0.79	0.81	0.83	0.84	0.85	0.85	0.85	0.86	0.87	0.88	0.89	0.92
	mPOS	0.67	0.74	0.76	0.78	0.80	0.83	0.84	0.85	0.86	0.87	0.88	0.89	0.90	0.90	0.91	0.92
	full set	0.90															
SVM	LASSO	0.68	0.76	0.82	0.85	0.86	0.88	0.89	0.90	0.90	0.90	0.90	0.91	0.91	0.91	0.91	0.92
	mRMR	0.70	0.75	0.79	0.82	0.84	0.87	0.88	0.89	0.90	0.90	0.91	0.91	0.91	0.91	0.92	0.92
	Wilcoxon	0.71	0.75	0.77	0.78	0.79	0.81	0.83	0.83	0.85	0.85	0.85	0.86	0.86	0.86	0.88	0.92
	mPOS	0.69	0.74	0.76	0.79	0.81	0.83	0.84	0.86	0.87	0.88	0.89	0.90	0.90	0.91	0.91	0.92
	full set	0.90															
XGBoost	LASSO	0.62	0.70	0.77	0.81	0.83	0.84	0.85	0.84	0.85	0.85	0.85	0.85	0.85	0.85	0.85	0.86
	mRMR	0.58	0.67	0.73	0.77	0.80	0.82	0.83	0.84	0.85	0.85	0.86	0.86	0.86	0.85	0.85	0.86
	Wilcoxon	0.63	0.65	0.73	0.74	0.76	0.78	0.80	0.80	0.81	0.82	0.82	0.83	0.83	0.83	0.84	0.85
	mPOS	0.63	0.68	0.72	0.75	0.77	0.79	0.81	0.82	0.83	0.83	0.84	0.85	0.85	0.85	0.85	0.86
	full set	0.84															

Table 8.26: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 9** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.53	0.55	0.58	0.60	0.63	0.64	0.64	0.65	0.67	0.67	0.68	0.68	0.68	0.68	0.69
	mRMR	0.51	0.54	0.57	0.60	0.63	0.65	0.66	0.67	0.69	0.71	0.71	0.71	0.72	0.72	0.73	0.73
	Wilcoxon	0.56	0.57	0.62	0.64	0.65	0.65	0.67	0.67	0.69	0.70	0.69	0.71	0.71	0.71	0.71	0.74
	mPOS	0.53	0.57	0.60	0.63	0.64	0.66	0.68	0.69	0.69	0.70	0.71	0.72	0.73	0.73	0.73	0.73
	full set	0.73															
KNN	LASSO	0.57	0.61	0.64	0.65	0.67	0.68	0.69	0.70	0.71	0.71	0.71	0.71	0.71	0.72	0.72	0.73
	mRMR	0.53	0.57	0.60	0.63	0.64	0.66	0.66	0.67	0.68	0.69	0.70	0.70	0.71	0.70	0.72	0.72
	Wilcoxon	0.59	0.63	0.65	0.64	0.65	0.66	0.68	0.68	0.69	0.69	0.69	0.70	0.72	0.71	0.72	0.73
	mPOS	0.57	0.6	0.62	0.64	0.66	0.66	0.68	0.69	0.70	0.70	0.71	0.72	0.72	0.73	0.72	0.73
	full set	0.69															
SVM	LASSO	0.57	0.62	0.65	0.67	0.69	0.70	0.71	0.72	0.73	0.73	0.74	0.74	0.74	0.74	0.73	0.74
	mRMR	0.54	0.59	0.63	0.65	0.67	0.68	0.69	0.70	0.71	0.72	0.72	0.72	0.73	0.73	0.74	0.73
	Wilcoxon	0.61	0.64	0.66	0.66	0.65	0.66	0.67	0.70	0.7	0.70	0.71	0.71	0.72	0.72	0.71	0.75
	mPOS	0.58	0.61	0.63	0.65	0.66	0.68	0.69	0.70	0.71	0.72	0.73	0.73	0.74	0.74	0.74	0.75
	full set	0.67															
XGBoost	LASSO	0.51	0.53	0.55	0.58	0.60	0.63	0.64	0.64	0.65	0.67	0.67	0.68	0.68	0.68	0.68	0.69
	mRMR	0.51	0.53	0.55	0.58	0.60	0.63	0.64	0.64	0.65	0.67	0.67	0.68	0.68	0.68	0.68	0.69
	Wilcoxon	0.56	0.58	0.59	0.60	0.62	0.63	0.64	0.66	0.66	0.66	0.65	0.66	0.66	0.67	0.67	0.68
	mPOS	0.53	0.56	0.58	0.60	0.62	0.63	0.65	0.65	0.66	0.67	0.67	0.68	0.69	0.69	0.69	0.69
	full set	0.67															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 10** for binary-class classification problems, is shown in Table 8.27. Across RF classifier, mPOS remains comparative performance with other feature selection techniques at different set sizes of informative features, except a set of single informative feature. For KNN, SVM, and XGBoost classifiers, mPOS achieves comparable performance to alternative feature selection techniques at different set sizes of informative features.

Table 8.28 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 11** for binary-class classification problems. For RF classifier, mPOS, mRMR, and Wilcoxon show comparable performance across different set sizes of informative features, with the exception of smaller set sizes. For KNN and SVM classifier, mPOS maintains a comparative performance with alternative feature selection techniques across various set sizes of informative features. Furthermore, when evaluated with the XGBoost classifier, mPOS achieves comparable performance to other feature selection techniques at different set sizes of informative feature, except a single informative feature.

Table 8.29 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 12** for binary-class classification problems. mPOS performs better than other feature selection techniques at smaller and larger set sizes of informative features across RF classifier. For KNN, SVM, XGBoost classifiers, both LASSO and mPOS outperform all other feature selection techniques at smaller set sizes of informative features. However, as larger set sizes of informative features are considered, mPOS demonstrates performance comparable to that of other feature selection techniques.

Table 8.27: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 10** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.79	0.92	0.96	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	mRMR	0.72	0.88	0.94	0.96	0.97	0.98	0.98	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1
	Wilcoxon	0.85	0.89	0.93	0.93	0.95	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.98	0.98	0.99
	mPOS	0.79	0.9	0.93	0.95	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1
	full set	1															
KNN	LASSO	0.84	0.92	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98
	mRMR	0.83	0.90	0.94	0.96	0.97	0.97	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.98	0.99
	Wilcoxon	0.85	0.90	0.93	0.93	0.94	0.94	0.96	0.96	0.96	0.96	0.96	0.97	0.97	0.97	0.98	0.99
	mPOS	0.84	0.91	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.98	0.97	0.98	0.98	0.98	0.99
	full set	0.97															
SVM	LASSO	0.84	0.92	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	mRMR	0.82	0.89	0.94	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	Wilcoxon	0.85	0.89	0.93	0.93	0.95	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.98	0.98	0.99
	mPOS	0.85	0.90	0.92	0.95	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	full set	0.99															
XGBoost	LASSO	0.79	0.91	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.96	0.96	0.97	0.97	0.97	0.97	0.97
	mRMR	0.72	0.86	0.93	0.95	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	Wilcoxon	0.80	0.87	0.92	0.92	0.93	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.97	0.97
	mPOS	0.79	0.89	0.92	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.96
	full set	0.96															

Table 8.28: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 11** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.59	0.74	0.8	0.84	0.85	0.86	0.87	0.88	0.89	0.90	0.89	0.90	0.90	0.90	0.90	0.90
	mRMR	0.59	0.78	0.83	0.86	0.88	0.89	0.90	0.91	0.92	0.93	0.94	0.94	0.94	0.95	0.95	0.95
	Wilcoxon	0.70	0.75	0.80	0.82	0.85	0.86	0.87	0.88	0.89	0.89	0.90	0.91	0.92	0.92	0.93	0.95
	mPOS	0.65	0.76	0.81	0.84	0.86	0.88	0.89	0.90	0.91	0.92	0.93	0.93	0.94	0.94	0.94	0.95
	full set	0.95															
KNN	LASSO	0.71	0.81	0.85	0.89	0.90	0.91	0.91	0.91	0.92	0.92	0.92	0.92	0.92	0.92	0.93	0.93
	mRMR	0.71	0.80	0.84	0.87	0.88	0.89	0.89	0.89	0.9	0.91	0.91	0.92	0.91	0.92	0.91	0.92
	Wilcoxon	0.74	0.79	0.81	0.83	0.84	0.85	0.86	0.87	0.89	0.89	0.89	0.89	0.90	0.90	0.90	0.92
	mPOS	0.71	0.77	0.81	0.84	0.86	0.88	0.89	0.89	0.90	0.91	0.92	0.92	0.92	0.92	0.92	0.92
	full set	0.91															
SVM	LASSO	0.73	0.81	0.86	0.89	0.90	0.91	0.92	0.93	0.93	0.94	0.94	0.94	0.94	0.94	0.94	0.94
	mRMR	0.72	0.79	0.84	0.86	0.89	0.90	0.91	0.92	0.93	0.94	0.94	0.94	0.94	0.95	0.95	0.95
	Wilcoxon	0.75	0.79	0.81	0.83	0.84	0.85	0.86	0.87	0.89	0.89	0.89	0.90	0.90	0.91	0.92	0.95
	MPOS	0.73	0.77	0.81	0.83	0.86	0.88	0.89	0.90	0.91	0.91	0.92	0.93	0.94	0.94	0.94	0.95
	full set	0.92															
XGBoost	LASSO	0.67	0.76	0.83	0.86	0.87	0.88	0.89	0.89	0.89	0.89	0.90	0.90	0.90	0.90	0.90	0.90
	mRMR	0.59	0.74	0.80	0.84	0.85	0.86	0.87	0.88	0.89	0.90	0.89	0.90	0.90	0.90	0.90	0.90
	Wilcoxon	0.70	0.71	0.77	0.79	0.82	0.84	0.84	0.86	0.86	0.87	0.87	0.87	0.88	0.88	0.88	0.90
	mPOS	0.65	0.73	0.78	0.82	0.85	0.86	0.86	0.88	0.88	0.88	0.89	0.90	0.90	0.90	0.90	0.90
	full set	0.89															

Table 8.29: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 12** for binary-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.57	0.66	0.7	0.74	0.75	0.77	0.78	0.79	0.80	0.80	0.81	0.81	0.81	0.81	0.81	0.81
	mRMR	0.52	0.55	0.61	0.67	0.70	0.72	0.74	0.74	0.75	0.77	0.78	0.79	0.79	0.80	0.80	0.82
	Wilcoxon	0.55	0.59	0.64	0.67	0.70	0.71	0.72	0.73	0.72	0.72	0.73	0.76	0.77	0.77	0.77	0.82
	mPOS	0.59	0.67	0.72	0.74	0.75	0.76	0.77	0.78	0.78	0.78	0.79	0.79	0.80	0.80	0.80	0.82
	full set	0.82															
KNN	LASSO	0.63	0.69	0.71	0.74	0.76	0.77	0.78	0.78	0.78	0.78	0.78	0.78	0.78	0.78	0.77	0.77
	mRMR	0.53	0.57	0.64	0.68	0.70	0.72	0.73	0.74	0.75	0.76	0.76	0.76	0.76	0.77	0.77	0.77
	Wilcoxon	0.60	0.63	0.66	0.67	0.70	0.72	0.71	0.71	0.71	0.72	0.72	0.72	0.74	0.74	0.75	0.77
	mPOS	0.63	0.69	0.72	0.74	0.75	0.75	0.77	0.76	0.77	0.77	0.77	0.77	0.77	0.77	0.77	0.77
	full set	0.74															
SVM	LASSO	0.64	0.69	0.71	0.74	0.76	0.78	0.78	0.79	0.80	0.80	0.81	0.80	0.80	0.81	0.80	0.79
	mRMR	0.53	0.60	0.66	0.70	0.72	0.74	0.76	0.77	0.77	0.78	0.79	0.79	0.79	0.80	0.80	0.81
	Wilcoxon	0.61	0.64	0.67	0.69	0.70	0.72	0.72	0.72	0.72	0.71	0.73	0.74	0.76	0.76	0.77	0.81
	mPOS	0.63	0.69	0.73	0.74	0.76	0.76	0.78	0.78	0.79	0.79	0.78	0.79	0.79	0.80	0.80	0.81
	full set	0.73															
XGBoost	LASSO	0.57	0.62	0.68	0.71	0.73	0.75	0.75	0.76	0.77	0.77	0.77	0.77	0.77	0.78	0.77	0.76
	mRMR	0.52	0.55	0.59	0.63	0.66	0.68	0.69	0.71	0.71	0.73	0.74	0.74	0.74	0.75	0.74	0.78
	Wilcoxon	0.55	0.59	0.61	0.63	0.66	0.67	0.68	0.69	0.67	0.68	0.68	0.70	0.72	0.71	0.72	0.77
	mPOS	0.59	0.65	0.69	0.72	0.73	0.73	0.74	0.74	0.75	0.75	0.75	0.75	0.76	0.76	0.76	0.77
	full set	0.74															

For the evaluation of mPOS method in three-class classification problems, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers across Scenario 1 to Scenario 12 is shown in Tables 8.30 - Table 8.41, respectively. For **Scenario 1**, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers for three-class classification problems, is shown in Table 8.30. The mPOS achieves superior performance with a single informative feature. However, mPOS shows comparable performance to other feature selection techniques when considering larger sets of informative features across the RF and XGBoost classifiers. Furthermore, mPOS remains competitive with alternative feature selection techniques at different set sizes of informative features when evaluated with the KNN and SVM classifiers.

Table 8.31 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 2** for three-class classification problems. mPOS performs better than other feature selection techniques at the small set sizes of informative features across four classifiers. However, as the size of the informative feature set increases, the performance of mPOS becomes comparable to that of other feature selection techniques.

Table 8.32 shows the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 3** for three-class classification problems. The mPOS outperforms all other feature selection techniques when considering small sets of informative features, while mPOS shows comparable performance to other feature selection techniques with larger set sizes of informative features across RF and KNN classifiers. Furthermore, mPOS achieves superior performance with a single informative feature when applied to SVM and XGBoost classifiers.

Table 8.30: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 1** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.71	0.87	0.93	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	mRMR	0.69	0.87	0.93	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1
	Kruskal	0.72	0.88	0.93	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1
	mPOS	0.75	0.87	0.92	0.95	0.97	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1	1
	full set	1															
KNN	LASSO	0.78	0.89	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	mRMR	0.78	0.89	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	Kruskal	0.8	0.89	0.95	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	mPOS	0.79	0.89	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	full set	1															
SVM	LASSO	0.78	0.89	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	mRMR	0.79	0.89	0.94	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	Kruskal	0.79	0.89	0.95	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	mPOS	0.79	0.89	0.94	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	full set	1															
XGBoost	LASSO	0.71	0.86	0.92	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.97	0.97	0.96	0.97	0.97	0.97
	mRMR	0.69	0.85	0.91	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.98	0.96	0.97	0.96	0.97
	Kruskal	0.72	0.87	0.92	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	mPOS	0.75	0.87	0.91	0.93	0.95	0.95	0.95	0.96	0.96	0.96	0.96	0.97	0.96	0.96	0.96	0.96
	full set	0.97															

Table 8.31: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 2** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.49	0.63	0.7	0.75	0.79	0.82	0.84	0.86	0.87	0.88	0.9	0.9	0.91	0.92	0.92	0.95
	mRMR	0.48	0.62	0.7	0.75	0.79	0.81	0.84	0.85	0.87	0.89	0.9	0.92	0.92	0.93	0.94	0.95
	Kruskal	0.52	0.65	0.72	0.77	0.8	0.83	0.83	0.86	0.87	0.89	0.89	0.9	0.91	0.92	0.92	0.95
	mPOS	0.54	0.66	0.72	0.76	0.8	0.82	0.84	0.86	0.88	0.88	0.89	0.9	0.91	0.92	0.92	0.95
	full set	0.99															
KNN	LASSO	0.59	0.67	0.73	0.77	0.81	0.84	0.87	0.89	0.9	0.91	0.93	0.94	0.94	0.95	0.96	0.98
	mRMR	0.57	0.66	0.72	0.77	0.81	0.84	0.86	0.88	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.98
	Kruskal	0.58	0.69	0.74	0.8	0.83	0.86	0.88	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.98
	mPOS	0.6	0.69	0.74	0.79	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.96	0.96	0.98
	full set	0.97															
SVM	LASSO	0.58	0.68	0.74	0.78	0.82	0.85	0.87	0.9	0.9	0.91	0.93	0.94	0.94	0.95	0.95	0.97
	mRMR	0.59	0.67	0.73	0.78	0.82	0.84	0.87	0.88	0.9	0.91	0.93	0.94	0.94	0.95	0.95	0.97
	Kruskal	0.59	0.69	0.75	0.81	0.84	0.86	0.88	0.9	0.91	0.91	0.92	0.94	0.95	0.95	0.95	0.97
	mPOS	0.62	0.69	0.75	0.79	0.83	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.95	0.97
	full set	0.98															
XGBoost	LASSO	0.49	0.61	0.68	0.73	0.76	0.79	0.81	0.82	0.83	0.84	0.85	0.86	0.86	0.87	0.87	0.88
	mRMR	0.48	0.6	0.68	0.73	0.76	0.79	0.81	0.82	0.83	0.84	0.85	0.86	0.87	0.88	0.88	0.88
	Kruskal	0.52	0.62	0.68	0.74	0.77	0.79	0.79	0.8	0.82	0.84	0.84	0.86	0.86	0.86	0.87	0.89
	mPOS	0.54	0.64	0.7	0.74	0.77	0.79	0.8	0.82	0.83	0.84	0.84	0.85	0.86	0.87	0.87	0.88
	full set	0.88															

Table 8.32: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 3** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.41	0.48	0.52	0.55	0.58	0.6	0.61	0.63	0.65	0.65	0.66	0.68	0.69	0.7	0.7	0.75
	mRMR	0.38	0.45	0.5	0.54	0.57	0.6	0.61	0.63	0.65	0.66	0.69	0.69	0.7	0.71	0.72	0.75
	Kruskal	0.39	0.47	0.49	0.54	0.56	0.6	0.61	0.63	0.65	0.67	0.67	0.68	0.69	0.7	0.72	0.75
	mPOS	0.43	0.49	0.52	0.54	0.57	0.58	0.6	0.62	0.63	0.65	0.67	0.68	0.69	0.7	0.7	0.74
	full set	0.79															
KNN	LASSO	0.45	0.5	0.54	0.57	0.6	0.62	0.64	0.65	0.67	0.68	0.69	0.71	0.72	0.73	0.74	0.79
	mRMR	0.44	0.49	0.53	0.57	0.59	0.61	0.62	0.65	0.67	0.68	0.69	0.7	0.71	0.72	0.74	0.79
	Kruskal	0.45	0.5	0.54	0.6	0.6	0.63	0.64	0.65	0.66	0.7	0.71	0.72	0.74	0.75	0.76	0.81
	mPOS	0.46	0.51	0.54	0.56	0.58	0.6	0.62	0.65	0.66	0.68	0.69	0.71	0.72	0.73	0.74	0.79
	full set	0.73															
SVM	LASSO	0.46	0.52	0.56	0.59	0.62	0.65	0.66	0.68	0.69	0.71	0.73	0.73	0.75	0.76	0.76	0.8
	mRMR	0.46	0.51	0.55	0.58	0.61	0.63	0.65	0.67	0.68	0.7	0.72	0.73	0.74	0.75	0.76	0.81
	Kruskal	0.45	0.51	0.56	0.62	0.63	0.66	0.66	0.68	0.7	0.73	0.73	0.74	0.76	0.76	0.78	0.81
	mPOS	0.48	0.52	0.55	0.58	0.6	0.62	0.64	0.66	0.69	0.7	0.71	0.73	0.75	0.75	0.77	0.81
	full set	0.76															
XGBoost	LASSO	0.41	0.46	0.49	0.52	0.56	0.58	0.6	0.61	0.61	0.63	0.64	0.64	0.66	0.67	0.67	0.69
	mRMR	0.38	0.42	0.47	0.5	0.53	0.56	0.58	0.59	0.6	0.63	0.64	0.64	0.65	0.65	0.66	0.69
	Kruskal	0.39	0.43	0.47	0.5	0.52	0.57	0.58	0.58	0.6	0.61	0.62	0.64	0.64	0.65	0.67	0.68
	mPOS	0.43	0.46	0.49	0.52	0.53	0.56	0.58	0.6	0.61	0.62	0.63	0.64	0.65	0.66	0.66	0.69
	full set	0.67															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 4** for three-class classification problems, is shown in Table 8.33. mPOS shows comparable performance to other feature selection techniques at different set sizes of informative features across the RF, KNN, SVM, and XGBoost classifiers.

Table 8.34 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 5** for three-class classification problems. mPOS performs better than other feature selection techniques with a single informative feature across four classifiers. However, as the size of the informative feature set increases, the performance of mPOS becomes comparable to that of other feature selection techniques.

Table 8.35 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 6** for three-class classification problems. mPOS demonstrates superior performance with a single informative feature when applied to the RF and XGBoost classifiers. Furthermore, mPOS remains competitive with alternative feature selection techniques at different set sizes of informative features across the KNN and SVM classifiers.

Table 8.33: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 4** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.62	0.81	0.87	0.9	0.92	0.93	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.98	0.98	0.99
	mRMR	0.61	0.79	0.86	0.9	0.92	0.93	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.99	0.99
	Kruskal	0.63	0.82	0.87	0.9	0.93	0.95	0.95	0.96	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99
	mPOS	0.62	0.81	0.88	0.91	0.92	0.94	0.95	0.95	0.96	0.97	0.97	0.97	0.97	0.98	0.98	0.99
	full set	1															
KNN	LASSO	0.71	0.82	0.88	0.9	0.92	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99	0.99
	mRMR	0.71	0.82	0.87	0.9	0.92	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.99	1
	Kruskal	0.72	0.84	0.89	0.92	0.94	0.96	0.96	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.99	0.99
	mPOS	0.7	0.83	0.89	0.91	0.93	0.94	0.95	0.96	0.97	0.97	0.97	0.98	0.98	0.99	0.99	0.99
	full set	0.98															
SVM	LASSO	0.72	0.82	0.88	0.91	0.93	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1
	mRMR	0.71	0.81	0.87	0.9	0.93	0.94	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	1
	Kruskal	0.72	0.84	0.89	0.92	0.95	0.96	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1
	mPOS	0.71	0.83	0.89	0.92	0.94	0.95	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	1
	full set	1															
XGBoost	LASSO	0.62	0.78	0.85	0.89	0.91	0.92	0.92	0.93	0.94	0.94	0.94	0.95	0.95	0.95	0.95	0.95
	mRMR	0.61	0.77	0.84	0.88	0.9	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.95	0.95	0.95
	Kruskal	0.63	0.8	0.85	0.88	0.91	0.92	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.95	0.95	0.95
	mPOS	0.62	0.79	0.87	0.9	0.91	0.92	0.93	0.93	0.93	0.94	0.94	0.94	0.94	0.94	0.95	0.95
	full set	0.95															

Table 8.34: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 5** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.67	0.74	0.79	0.82	0.85	0.86	0.88	0.89	0.9	0.92	0.92	0.93	0.93	0.94	0.96
	mRMR	0.49	0.64	0.72	0.78	0.82	0.84	0.86	0.88	0.89	0.9	0.91	0.92	0.93	0.93	0.94	0.96
	Kruskal	0.51	0.67	0.74	0.79	0.84	0.85	0.87	0.88	0.89	0.9	0.91	0.92	0.93	0.93	0.94	0.96
	mPOS	0.55	0.67	0.74	0.79	0.82	0.84	0.86	0.88	0.89	0.9	0.91	0.92	0.93	0.93	0.94	0.96
	full set	0.99															
KNN	LASSO	0.61	0.7	0.76	0.8	0.83	0.85	0.87	0.88	0.89	0.9	0.9	0.91	0.92	0.92	0.92	0.94
	mRMR	0.59	0.68	0.74	0.78	0.82	0.84	0.86	0.87	0.88	0.89	0.9	0.91	0.92	0.92	0.93	0.94
	Kruskal	0.59	0.71	0.77	0.81	0.85	0.86	0.88	0.89	0.9	0.9	0.91	0.92	0.92	0.93	0.93	0.95
	mPOS	0.62	0.7	0.76	0.8	0.83	0.84	0.86	0.87	0.88	0.89	0.9	0.91	0.91	0.92	0.92	0.94
	full set	0.89															
SVM	LASSO	0.6	0.7	0.75	0.8	0.83	0.86	0.88	0.9	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	mRMR	0.6	0.68	0.74	0.79	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.96	0.97
	Kruskal	0.61	0.7	0.76	0.8	0.84	0.87	0.89	0.9	0.91	0.92	0.93	0.95	0.95	0.95	0.95	0.98
	mPOS	0.63	0.71	0.75	0.8	0.83	0.85	0.88	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	full set	0.97															
XGBoost	LASSO	0.51	0.64	0.71	0.77	0.8	0.82	0.83	0.84	0.85	0.86	0.86	0.87	0.88	0.88	0.89	0.89
	mRMR	0.49	0.61	0.7	0.75	0.8	0.81	0.83	0.84	0.85	0.86	0.86	0.87	0.88	0.88	0.89	0.89
	Kruskal	0.51	0.63	0.71	0.76	0.8	0.82	0.82	0.83	0.84	0.86	0.86	0.86	0.87	0.87	0.88	0.89
	mPOS	0.55	0.66	0.72	0.76	0.79	0.81	0.83	0.84	0.85	0.86	0.86	0.87	0.87	0.88	0.88	0.89
	full set	0.88															

Table 8.35: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 6** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.43	0.5	0.57	0.61	0.64	0.67	0.69	0.7	0.72	0.73	0.75	0.76	0.77	0.78	0.79	0.83
	mRMR	0.39	0.48	0.55	0.6	0.63	0.66	0.68	0.7	0.72	0.73	0.74	0.75	0.76	0.78	0.79	0.83
	Kruskal	0.4	0.49	0.55	0.6	0.64	0.66	0.67	0.69	0.71	0.74	0.74	0.75	0.77	0.77	0.78	0.82
	mPOS	0.45	0.5	0.56	0.61	0.64	0.65	0.68	0.69	0.71	0.73	0.74	0.76	0.77	0.78	0.78	0.83
	full set	0.88															
KNN	LASSO	0.47	0.55	0.6	0.63	0.66	0.68	0.69	0.7	0.72	0.73	0.74	0.75	0.75	0.76	0.76	0.77
	mRMR	0.47	0.53	0.58	0.61	0.64	0.66	0.68	0.69	0.71	0.72	0.73	0.73	0.74	0.74	0.75	0.76
	Kruskal	0.47	0.54	0.59	0.65	0.66	0.67	0.68	0.69	0.71	0.73	0.72	0.73	0.74	0.75	0.76	0.76
	mPOS	0.47	0.54	0.59	0.62	0.64	0.66	0.68	0.69	0.7	0.7	0.71	0.71	0.73	0.73	0.73	0.76
	full set	0.66															
SVM	LASSO	0.49	0.56	0.6	0.63	0.66	0.69	0.7	0.72	0.74	0.76	0.77	0.79	0.8	0.8	0.82	0.86
	mRMR	0.48	0.54	0.59	0.61	0.65	0.67	0.7	0.72	0.74	0.75	0.76	0.78	0.79	0.8	0.82	0.86
	Kruskal	0.48	0.54	0.6	0.66	0.67	0.69	0.71	0.73	0.75	0.77	0.78	0.78	0.8	0.81	0.83	0.86
	mPOS	0.5	0.55	0.59	0.62	0.65	0.67	0.69	0.71	0.73	0.75	0.76	0.78	0.79	0.8	0.81	0.86
	full set	0.83															
XGBoost	LASSO	0.43	0.49	0.55	0.59	0.62	0.65	0.66	0.67	0.68	0.69	0.7	0.72	0.73	0.73	0.74	0.77
	mRMR	0.39	0.46	0.51	0.55	0.6	0.62	0.64	0.67	0.68	0.69	0.7	0.72	0.73	0.74	0.74	0.76
	Kruskal	0.4	0.48	0.52	0.58	0.6	0.64	0.65	0.67	0.68	0.69	0.7	0.71	0.72	0.72	0.73	0.75
	mPOS	0.45	0.48	0.53	0.57	0.61	0.63	0.65	0.67	0.68	0.69	0.7	0.72	0.72	0.72	0.73	0.75
	full set	0.74															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 7** for three-class classification problems, is shown in Table 8.36. mPOS shows comparable performance to the LASSO and mRMR techniques at different set sizes of informative features with a RF classifier. In the case of the KNN classifier, mPOS and LASSO perform better than other feature selection techniques with a single informative feature, while Kruskal shows inferior performance compared to other feature selection techniques when considering the larger set sizes of informative features. For SVM and XGBoost classifier, mPOS remains comparable to the LASSO and mRMR techniques.

Table 8.37 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 8** for three-class classification problems. For the RF classifier, LASSO outperforms all other feature selection techniques at the small set sizes of informative features. However, mPOS shows comparable performance to LASSO and mRMR techniques when considering larger set sizes of informative features. For KNN, SVM, and XGBoost classifiers, LASSO and mRMR achieve superior performance at the smaller set sizes of informative features. As the size of informative feature set increases, the performance of mPOS becomes comparable to that of LASSO and mRMR techniques.

Table 8.38 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 9** for three-class classification problems. For RF and XGBoost classifiers, mPOS shows comparable performance to the LASSO and mRMR technqies. In contrast, both LASSO and mRMR outperform all other feature selection techniques across KNN and SVM classifiers.

Table 8.36: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 7** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.72	0.87	0.92	0.94	0.96	0.97	0.97	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.99	0.99
	mRMR	0.7	0.86	0.92	0.94	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	Kruskal	0.68	0.76	0.81	0.86	0.89	0.92	0.93	0.93	0.94	0.95	0.96	0.97	0.97	0.97	0.97	0.99
	mPOS	0.73	0.86	0.91	0.93	0.95	0.96	0.96	0.97	0.97	0.98	0.98	0.98	0.98	0.98	0.99	0.99
	full set	1															
KNN	LASSO	0.79	0.89	0.94	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1
	mRMR	0.77	0.88	0.93	0.96	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1
	Kruskal	0.77	0.81	0.84	0.88	0.91	0.92	0.94	0.94	0.96	0.97	0.97	0.98	0.98	0.99	0.99	1
	mPOS	0.79	0.88	0.92	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1
	full set	1															
SVM	LASSO	0.8	0.89	0.94	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
	mRMR	0.77	0.89	0.93	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	Kruskal	0.78	0.81	0.84	0.88	0.91	0.92	0.94	0.94	0.95	0.96	0.97	0.97	0.98	0.98	0.98	1
	mPOS	0.79	0.89	0.93	0.95	0.96	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99
	full set	1															
XGBoost	LASSO	0.72	0.86	0.91	0.93	0.94	0.95	0.95	0.95	0.95	0.95	0.96	0.96	0.96	0.95	0.96	0.96
	mRMR	0.7	0.85	0.91	0.93	0.94	0.95	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.97	0.96	0.96
	Kruskal	0.68	0.75	0.79	0.84	0.87	0.89	0.9	0.91	0.92	0.93	0.93	0.94	0.94	0.94	0.95	0.95
	mPOS	0.73	0.85	0.9	0.92	0.93	0.94	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.96
	full set	0.96															

Table 8.37: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 8** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.64	0.7	0.74	0.77	0.78	0.8	0.82	0.82	0.83	0.84	0.84	0.85	0.85	0.86	0.87
	mRMR	0.5	0.62	0.69	0.73	0.77	0.79	0.81	0.82	0.83	0.84	0.85	0.85	0.86	0.86	0.87	0.87
	Kruskal	0.52	0.53	0.59	0.62	0.64	0.67	0.68	0.7	0.74	0.75	0.76	0.78	0.78	0.79	0.8	0.83
	mPOS	0.49	0.6	0.68	0.72	0.75	0.77	0.79	0.8	0.81	0.82	0.83	0.83	0.84	0.84	0.85	0.87
	full set	0.90															
KNN	LASSO	0.59	0.67	0.72	0.76	0.78	0.8	0.82	0.84	0.84	0.85	0.86	0.87	0.87	0.88	0.88	0.89
	mRMR	0.56	0.67	0.72	0.76	0.79	0.81	0.83	0.84	0.85	0.86	0.87	0.88	0.88	0.89	0.9	0.9
	Kruskal	0.57	0.6	0.63	0.64	0.66	0.68	0.69	0.72	0.74	0.76	0.77	0.78	0.8	0.8	0.8	0.87
	mPOS	0.55	0.64	0.7	0.74	0.77	0.79	0.81	0.82	0.83	0.84	0.85	0.86	0.86	0.87	0.87	0.89
	full set	0.88															
SVM	LASSO	0.6	0.68	0.73	0.76	0.79	0.81	0.83	0.84	0.85	0.86	0.86	0.87	0.87	0.87	0.88	0.89
	mRMR	0.58	0.68	0.72	0.77	0.79	0.82	0.84	0.85	0.85	0.86	0.87	0.88	0.88	0.89	0.89	0.89
	Kruskal	0.59	0.6	0.63	0.66	0.69	0.69	0.71	0.72	0.76	0.77	0.77	0.78	0.8	0.8	0.8	0.85
	mPOS	0.57	0.65	0.72	0.74	0.78	0.8	0.81	0.83	0.84	0.84	0.85	0.86	0.86	0.87	0.87	0.89
	full set	0.84															
XGBoost	LASSO	0.51	0.61	0.68	0.72	0.74	0.76	0.77	0.78	0.78	0.79	0.79	0.79	0.8	0.8	0.8	0.82
	mRMR	0.5	0.6	0.67	0.71	0.74	0.76	0.77	0.78	0.79	0.8	0.8	0.8	0.8	0.8	0.81	0.81
	Kruskal	0.52	0.52	0.56	0.59	0.62	0.64	0.66	0.67	0.7	0.71	0.72	0.73	0.74	0.75	0.75	0.78
	mPOS	0.49	0.58	0.66	0.69	0.72	0.74	0.76	0.77	0.77	0.78	0.79	0.79	0.79	0.8	0.8	0.81
	full set	0.81															

Table 8.38: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 9** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.4	0.47	0.51	0.53	0.55	0.57	0.59	0.59	0.6	0.6	0.61	0.62	0.63	0.63	0.64	0.65
	mRMR	0.39	0.45	0.49	0.53	0.56	0.58	0.59	0.6	0.6	0.61	0.62	0.63	0.63	0.64	0.64	0.66
	Kruskal	0.37	0.38	0.43	0.46	0.48	0.48	0.49	0.5	0.54	0.55	0.55	0.56	0.58	0.59	0.6	0.62
	mPOS	0.39	0.44	0.48	0.5	0.53	0.55	0.56	0.58	0.59	0.6	0.6	0.61	0.62	0.62	0.63	0.65
	full set	0.66															
KNN	LASSO	0.44	0.52	0.53	0.55	0.57	0.58	0.59	0.6	0.6	0.61	0.62	0.62	0.63	0.64	0.64	0.64
	mRMR	0.44	0.5	0.53	0.56	0.57	0.58	0.6	0.61	0.61	0.62	0.63	0.64	0.63	0.64	0.65	0.65
	Kruskal	0.44	0.45	0.48	0.47	0.49	0.49	0.52	0.51	0.54	0.56	0.55	0.56	0.57	0.58	0.59	0.62
	mPOS	0.43	0.47	0.51	0.53	0.55	0.56	0.57	0.59	0.6	0.6	0.61	0.61	0.62	0.62	0.63	0.65
	full set	0.63															
SVM	LASSO	0.46	0.52	0.54	0.57	0.58	0.6	0.61	0.62	0.63	0.64	0.64	0.64	0.65	0.65	0.65	0.66
	mRMR	0.46	0.52	0.54	0.57	0.59	0.59	0.62	0.62	0.63	0.64	0.65	0.66	0.65	0.66	0.67	0.67
	Kruskal	0.47	0.47	0.49	0.5	0.51	0.51	0.52	0.53	0.56	0.56	0.56	0.57	0.58	0.58	0.6	0.64
	mPOS	0.44	0.49	0.52	0.55	0.56	0.57	0.58	0.6	0.6	0.61	0.62	0.63	0.63	0.63	0.64	0.66
	full set	0.57															
XGBoost	LASSO	0.4	0.43	0.47	0.51	0.52	0.53	0.54	0.55	0.56	0.56	0.56	0.57	0.57	0.58	0.58	0.59
	mRMR	0.39	0.43	0.45	0.49	0.52	0.53	0.54	0.55	0.56	0.57	0.57	0.58	0.58	0.59	0.59	0.59
	Kruskal	0.37	0.38	0.41	0.42	0.44	0.45	0.45	0.46	0.49	0.51	0.5	0.52	0.53	0.54	0.55	0.57
	mPOS	0.39	0.42	0.44	0.48	0.49	0.51	0.52	0.53	0.55	0.56	0.55	0.56	0.57	0.56	0.57	0.58
	full set	0.57															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 10** for three-class classification problems, is shown in Table 8.39. For RF and XGBoost classifiers, mPOS demonstrates superior performance with a single informative feature. However, mPOS show comparable performance to the LASSO and mRMR techniques when considering larger set sizes of informative features. In term of KNN and SVM classifiers, mPOS remains comparative with LASSO and mRMR techniques at the different set sizes of informative features.

Table 8.40 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 11** for three-class classification problems. mPOS achieves comparable performance to alternative feature selection techniques with a single informative feature and it also remains competitive with LASSO and mRMR at the larger set sizes of informative features across four different classifiers. In contrast, both LASSO and mRMR perform better than other feature selection techniques at the small and moderate set sizes of informative features.

Table 8.41 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 12** for three-class classification problems. For RF and XGBoost classifiers, mPOS shows comparable performance to other feature selection techniques with a single informative feature across four different classifiers. As a size of informative feature increases, both LASSO and mRMR outperform all other feature selection techniques when evaluated with the RF classifier. mRMR achieves the best performance at larger set sizes of informative features when using KNN classifier. Both LASSO and mRMR outperform all other feature selection techniques at moderate set sizes of informative features when using the SVM classifier, while LASSO shows superior performance at different set sizes of informative feature, except for a single informative feature, when evalued with the XGBoost classifier.

Table 8.39: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 10** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.62	0.78	0.86	0.89	0.91	0.93	0.94	0.94	0.95	0.95	0.95	0.95	0.96	0.96	0.96	0.97
	mRMR	0.62	0.79	0.86	0.89	0.91	0.93	0.93	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.97
	Kruskal	0.62	0.69	0.77	0.8	0.82	0.85	0.87	0.88	0.9	0.91	0.91	0.92	0.93	0.93	0.93	0.96
	mPOS	0.66	0.8	0.85	0.88	0.89	0.9	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.95	0.96	0.97
	full set	0.98															
KNN	LASSO	0.72	0.81	0.86	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.95	0.95	0.95	0.95	0.96	0.96
	mRMR	0.71	0.82	0.87	0.9	0.91	0.92	0.93	0.94	0.94	0.94	0.95	0.95	0.95	0.95	0.95	0.96
	Kruskal	0.7	0.75	0.79	0.8	0.82	0.84	0.85	0.86	0.88	0.88	0.89	0.89	0.9	0.91	0.91	0.94
	mPOS	0.73	0.81	0.85	0.87	0.89	0.9	0.9	0.92	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.96
	full set	0.93															
SVM	LASSO	0.72	0.8	0.87	0.89	0.91	0.93	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.98
	mRMR	0.72	0.82	0.87	0.9	0.92	0.94	0.94	0.95	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97
	Kruskal	0.72	0.74	0.79	0.81	0.81	0.84	0.86	0.86	0.89	0.9	0.91	0.91	0.92	0.93	0.94	0.96
	mPOS	0.73	0.81	0.85	0.88	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.97
	full set	0.96															
XGBoost	LASSO	0.62	0.77	0.84	0.87	0.89	0.91	0.91	0.92	0.92	0.92	0.92	0.92	0.92	0.93	0.92	0.93
	mRMR	0.62	0.77	0.85	0.88	0.89	0.91	0.91	0.91	0.92	0.92	0.93	0.93	0.93	0.93	0.93	0.92
	Kruskal	0.62	0.67	0.74	0.78	0.82	0.84	0.85	0.86	0.87	0.88	0.89	0.89	0.9	0.9	0.9	0.92
	mPOS	0.66	0.78	0.83	0.86	0.87	0.89	0.89	0.9	0.91	0.91	0.92	0.92	0.92	0.92	0.92	0.92
	full set	0.92															

Table 8.40: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 11** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.51	0.64	0.72	0.77	0.8	0.81	0.83	0.84	0.85	0.86	0.87	0.87	0.88	0.88	0.89	0.9
	mRMR	0.5	0.65	0.73	0.78	0.8	0.83	0.84	0.85	0.86	0.87	0.88	0.88	0.89	0.89	0.89	0.9
	Kruskal	0.51	0.57	0.64	0.68	0.7	0.74	0.75	0.76	0.79	0.8	0.81	0.81	0.82	0.83	0.84	0.88
	mPOS	0.52	0.63	0.7	0.73	0.76	0.78	0.8	0.81	0.83	0.84	0.85	0.85	0.86	0.87	0.87	0.89
	full set	0.92															
KNN	LASSO	0.6	0.68	0.74	0.78	0.8	0.81	0.83	0.83	0.84	0.84	0.85	0.85	0.85	0.85	0.86	0.86
	mRMR	0.59	0.69	0.75	0.78	0.81	0.82	0.83	0.84	0.84	0.85	0.86	0.86	0.86	0.86	0.86	0.85
	Kruskal	0.59	0.62	0.66	0.68	0.7	0.72	0.73	0.74	0.75	0.78	0.79	0.79	0.8	0.8	0.8	0.83
	mPOS	0.59	0.67	0.71	0.74	0.76	0.78	0.79	0.8	0.81	0.81	0.82	0.82	0.83	0.83	0.83	0.84
	full set	0.81															
SVM	LASSO	0.61	0.68	0.73	0.78	0.8	0.82	0.83	0.85	0.86	0.86	0.87	0.88	0.88	0.88	0.89	0.9
	mRMR	0.6	0.69	0.75	0.78	0.81	0.83	0.85	0.85	0.86	0.87	0.88	0.89	0.89	0.89	0.89	0.89
	Kruskal	0.61	0.62	0.64	0.67	0.7	0.7	0.72	0.73	0.77	0.77	0.78	0.78	0.8	0.81	0.82	0.86
	mPOS	0.6	0.66	0.71	0.74	0.77	0.8	0.81	0.82	0.83	0.84	0.85	0.86	0.86	0.87	0.87	0.88
	full set	0.84															
XGBoost	LASSO	0.51	0.62	0.69	0.74	0.76	0.77	0.79	0.8	0.81	0.82	0.82	0.82	0.83	0.83	0.83	0.84
	mRMR	0.5	0.63	0.71	0.74	0.77	0.79	0.81	0.81	0.82	0.83	0.83	0.83	0.84	0.84	0.83	0.84
	Kruskal	0.51	0.54	0.6	0.64	0.68	0.7	0.72	0.73	0.74	0.77	0.77	0.76	0.77	0.78	0.79	0.82
	mPOS	0.52	0.61	0.68	0.72	0.74	0.75	0.77	0.78	0.79	0.79	0.8	0.81	0.82	0.82	0.82	0.83
	full set	0.83															

Table 8.41: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 12** for three-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.42	0.5	0.57	0.61	0.64	0.66	0.68	0.69	0.7	0.71	0.72	0.72	0.72	0.73	0.73	0.75
	mRMR	0.4	0.49	0.55	0.58	0.62	0.64	0.65	0.67	0.68	0.7	0.7	0.71	0.72	0.72	0.72	0.73
	Kruskal	0.42	0.46	0.52	0.54	0.54	0.56	0.58	0.58	0.6	0.6	0.61	0.62	0.65	0.66	0.67	0.7
	mPOS	0.4	0.49	0.54	0.57	0.6	0.62	0.63	0.64	0.65	0.66	0.67	0.68	0.69	0.7	0.7	0.73
	full set	0.75															
KNN	LASSO	0.46	0.55	0.59	0.62	0.64	0.65	0.65	0.65	0.66	0.65	0.65	0.65	0.63	0.63	0.63	0.63
	mRMR	0.47	0.53	0.58	0.61	0.63	0.64	0.65	0.66	0.66	0.67	0.68	0.68	0.68	0.68	0.69	0.69
	Kruskal	0.5	0.5	0.53	0.53	0.55	0.55	0.57	0.58	0.58	0.59	0.59	0.61	0.61	0.62	0.62	0.65
	mPOS	0.47	0.54	0.57	0.59	0.61	0.63	0.63	0.64	0.64	0.65	0.65	0.66	0.66	0.67	0.66	0.67
	full set	0.62															
SVM	LASSO	0.49	0.54	0.58	0.61	0.63	0.65	0.67	0.68	0.7	0.7	0.7	0.7	0.71	0.71	0.71	0.71
	mRMR	0.48	0.55	0.58	0.62	0.63	0.65	0.66	0.68	0.69	0.71	0.72	0.72	0.72	0.73	0.73	0.73
	Kruskal	0.5	0.5	0.52	0.53	0.54	0.56	0.57	0.58	0.6	0.6	0.61	0.62	0.64	0.65	0.65	0.68
	mPOS	0.48	0.54	0.57	0.6	0.62	0.64	0.65	0.65	0.66	0.67	0.67	0.68	0.69	0.69	0.7	0.72
	full set	0.64															
XGBoost	LASSO	0.43	0.49	0.55	0.59	0.62	0.65	0.66	0.67	0.68	0.69	0.7	0.72	0.73	0.73	0.74	0.77
	mRMR	0.4	0.46	0.52	0.56	0.58	0.6	0.62	0.63	0.64	0.66	0.67	0.67	0.67	0.68	0.68	0.68
	Kruskal	0.42	0.45	0.47	0.5	0.51	0.53	0.54	0.55	0.56	0.56	0.58	0.59	0.6	0.61	0.62	0.64
	mPOS	0.4	0.46	0.51	0.55	0.57	0.58	0.6	0.61	0.62	0.62	0.62	0.64	0.64	0.65	0.66	0.67
	full set	0.66															

For the evaluation of mPOS method in four-class classification problems, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers across Scenario 1 to Scenario 12 is shown in Tables 8.42 - Table 8.53, respectively. For **Scenario 1**, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers for four-class classification problems, is shown in Table 8.42. mPOS shows comparable performance to other feature selection techniques at the different set sizes of informative features across the RF, KNN, SVM, and XGBoost classifiers.

Table 8.43 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 2** for four-class classification problems. mPOS remains comparative with alternative feature selection technique at the different set sizes of informative features across the RF, KNN, SVM, and XGBoost classifiers.

Table 8.44 shows the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 3** for four-class classification problems. mPOS demonstrates comparable performance to other feature selection technique at the different set sizes of informative features across the RF, KNN, SVM, and XGBoost classifiers.

Table 8.42: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 1** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.66	0.85	0.92	0.95	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	mRMR	0.67	0.86	0.92	0.95	0.96	0.97	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1
	Kruskal	0.64	0.86	0.91	0.95	0.97	0.97	0.98	0.98	0.98	0.98	0.99	0.99	0.99	0.99	1	1
	mPOS	0.68	0.86	0.92	0.95	0.97	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1
	full set	1															
KNN	LASSO	0.75	0.87	0.93	0.96	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	mRMR	0.76	0.88	0.93	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	Kruskal	0.74	0.89	0.93	0.97	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1
	mPOS	0.76	0.88	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	full set	1															
SVM	LASSO	0.75	0.87	0.93	0.96	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	mRMR	0.76	0.88	0.93	0.96	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	Kruskal	0.73	0.89	0.93	0.97	0.98	0.99	0.99	0.99	0.99	1	1	1	1	1	1	1
	mPOS	0.77	0.88	0.94	0.96	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	full set	1															
XGBoost	LASSO	0.66	0.84	0.9	0.93	0.95	0.95	0.96	0.96	0.97	0.96	0.97	0.97	0.97	0.97	0.97	0.97
	mRMR	0.67	0.84	0.91	0.93	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.98	0.97	0.98	0.98
	Kruskal	0.64	0.84	0.9	0.93	0.95	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.98	0.97
	mPOS	0.68	0.85	0.91	0.94	0.95	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.98	0.97	0.98
	full set	0.98															

Table 8.43: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 2** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.44	0.58	0.66	0.72	0.77	0.79	0.82	0.84	0.86	0.88	0.89	0.9	0.91	0.91	0.92	0.95
	mRMR	0.43	0.57	0.66	0.72	0.77	0.8	0.83	0.85	0.86	0.88	0.89	0.9	0.91	0.92	0.92	0.95
	Kruskal	0.42	0.53	0.65	0.71	0.75	0.79	0.82	0.84	0.86	0.87	0.88	0.89	0.9	0.91	0.91	0.95
	mPOS	0.43	0.58	0.66	0.72	0.76	0.79	0.82	0.84	0.86	0.87	0.89	0.9	0.91	0.91	0.92	0.95
	full set	0.99															
KNN	LASSO	0.53	0.62	0.69	0.75	0.8	0.83	0.85	0.88	0.9	0.91	0.92	0.94	0.94	0.95	0.96	0.98
	mRMR	0.52	0.63	0.7	0.75	0.79	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.97
	Kruskal	0.5	0.61	0.69	0.73	0.77	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.96	0.98
	mPOS	0.53	0.62	0.69	0.74	0.78	0.82	0.85	0.87	0.89	0.9	0.91	0.93	0.94	0.94	0.95	0.98
	full set	0.96															
SVM	LASSO	0.54	0.63	0.7	0.76	0.8	0.83	0.85	0.88	0.9	0.91	0.92	0.93	0.94	0.95	0.95	0.97
	mRMR	0.54	0.64	0.71	0.76	0.8	0.83	0.85	0.88	0.89	0.91	0.92	0.93	0.93	0.94	0.95	0.96
	Kruskal	0.52	0.62	0.7	0.74	0.78	0.82	0.85	0.88	0.9	0.91	0.92	0.93	0.94	0.94	0.95	0.97
	mPOS	0.53	0.63	0.7	0.75	0.79	0.82	0.85	0.87	0.89	0.9	0.91	0.93	0.93	0.94	0.94	0.97
	full set	0.99															
XGBoost	LASSO	0.44	0.56	0.64	0.69	0.74	0.77	0.79	0.8	0.81	0.83	0.84	0.84	0.85	0.86	0.86	0.89
	mRMR	0.43	0.56	0.64	0.7	0.74	0.77	0.78	0.8	0.82	0.83	0.83	0.84	0.84	0.85	0.86	0.88
	Kruskal	0.42	0.54	0.63	0.68	0.72	0.75	0.78	0.8	0.81	0.82	0.83	0.84	0.85	0.85	0.86	0.88
	mPOS	0.43	0.56	0.64	0.69	0.73	0.76	0.77	0.8	0.81	0.82	0.83	0.83	0.85	0.85	0.86	0.88
	full set	0.90															

Table 8.44: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 3** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.31	0.39	0.45	0.48	0.52	0.55	0.56	0.58	0.59	0.62	0.63	0.64	0.65	0.66	0.67	0.71
	mRMR	0.31	0.39	0.45	0.49	0.51	0.54	0.56	0.59	0.61	0.62	0.63	0.65	0.66	0.67	0.69	0.72
	Kruskal	0.31	0.39	0.46	0.48	0.51	0.55	0.58	0.59	0.6	0.61	0.63	0.63	0.65	0.65	0.66	0.7
	mPOS	0.31	0.4	0.44	0.48	0.52	0.54	0.56	0.58	0.59	0.61	0.62	0.64	0.65	0.66	0.67	0.71
	full set	0.77															
KNN	LASSO	0.38	0.43	0.48	0.51	0.54	0.57	0.59	0.61	0.63	0.64	0.66	0.68	0.7	0.7	0.72	0.77
	mRMR	0.38	0.44	0.48	0.51	0.53	0.56	0.59	0.61	0.63	0.65	0.66	0.68	0.69	0.71	0.72	0.76
	Kruskal	0.36	0.43	0.48	0.51	0.53	0.55	0.58	0.6	0.63	0.63	0.66	0.66	0.67	0.68	0.7	0.75
	mPOS	0.38	0.43	0.48	0.51	0.54	0.56	0.58	0.6	0.62	0.64	0.65	0.66	0.68	0.7	0.71	0.76
	full set	0.70															
SVM	LASSO	0.41	0.45	0.5	0.53	0.56	0.59	0.61	0.64	0.65	0.67	0.69	0.71	0.72	0.73	0.74	0.78
	mRMR	0.4	0.45	0.5	0.53	0.56	0.59	0.61	0.63	0.66	0.67	0.68	0.7	0.71	0.73	0.74	0.78
	Kruskal	0.38	0.44	0.5	0.51	0.55	0.58	0.61	0.63	0.65	0.66	0.67	0.68	0.7	0.71	0.72	0.76
	mPOS	0.4	0.45	0.48	0.52	0.55	0.58	0.6	0.62	0.64	0.66	0.67	0.69	0.7	0.71	0.72	0.77
	full set	0.80															
XGBoost	LASSO	0.31	0.37	0.42	0.46	0.49	0.53	0.53	0.55	0.56	0.58	0.59	0.61	0.62	0.63	0.63	0.66
	mRMR	0.31	0.38	0.42	0.47	0.49	0.52	0.54	0.56	0.57	0.59	0.59	0.61	0.62	0.62	0.63	0.65
	Kruskal	0.31	0.37	0.42	0.45	0.49	0.52	0.55	0.56	0.58	0.59	0.59	0.6	0.61	0.62	0.63	0.65
	mPOS	0.31	0.38	0.42	0.46	0.48	0.51	0.53	0.55	0.56	0.57	0.58	0.59	0.6	0.61	0.61	0.64
	full set	0.67															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 4** for four-class classification problems, is shown in Table 8.45. mPOS achieves comparable performance to LASSO and mRMR techniques with a single informative feature across RF and XGBoost classifiers. Furthermore, as the sizes of the informative feature set increase, mPOS remains a comparative performance with alternative feature selection techniques. According to the KNN and SVM classifiers, mPOS shows a comparable performance to other feature selection techniques at the different set sizes of informative features.

Table 8.46 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 5** for four-class classification problems. mPOS shows a performance comparable to that of LASSO and mRMR techniques at the smaller set sizes of informative features across RF and XGBoost classifiers. For KNN and SVM classifiers, mPOS demonstrates superior performance at a single informative feature.

Table 8.47 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 6** for four-class classification problems. mPOS achieves a comparable performance to alternative feature selection techniques across the RF, KNN, SVM, XGBoost classifiers.

Table 8.45: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 4** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.53	0.72	0.79	0.84	0.87	0.89	0.91	0.92	0.93	0.94	0.95	0.96	0.96	0.96	0.97	0.98
	mRMR	0.53	0.71	0.8	0.84	0.88	0.9	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97	0.97	0.98
	Kruskal	0.51	0.71	0.79	0.85	0.88	0.89	0.9	0.92	0.93	0.94	0.94	0.95	0.95	0.96	0.97	0.98
	mPOS	0.53	0.72	0.8	0.84	0.87	0.9	0.91	0.92	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.98
	full set	1															
KNN	LASSO	0.64	0.75	0.81	0.85	0.88	0.9	0.92	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.99
	mRMR	0.62	0.75	0.81	0.86	0.88	0.9	0.92	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.97	0.98
	Kruskal	0.61	0.75	0.81	0.85	0.88	0.9	0.92	0.93	0.94	0.94	0.95	0.96	0.97	0.97	0.97	0.99
	mPOS	0.64	0.75	0.82	0.85	0.88	0.9	0.92	0.92	0.94	0.94	0.95	0.96	0.96	0.97	0.97	0.98
	full set	0.97															
SVM	LASSO	0.64	0.75	0.81	0.86	0.89	0.91	0.93	0.94	0.95	0.96	0.97	0.97	0.97	0.98	0.98	0.99
	mRMR	0.63	0.75	0.82	0.86	0.89	0.91	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.98	0.99
	Kruskal	0.62	0.75	0.81	0.86	0.88	0.91	0.93	0.94	0.95	0.96	0.97	0.97	0.97	0.97	0.98	0.99
	mPOS	0.64	0.75	0.82	0.86	0.89	0.91	0.93	0.94	0.95	0.96	0.97	0.97	0.97	0.98	0.98	0.99
	full set	1															
XGBoost	LASSO	0.53	0.69	0.77	0.82	0.84	0.86	0.88	0.89	0.9	0.9	0.91	0.91	0.91	0.92	0.92	0.93
	mRMR	0.53	0.7	0.78	0.82	0.85	0.87	0.88	0.89	0.89	0.9	0.9	0.91	0.91	0.92	0.92	0.94
	Kruskal	0.51	0.69	0.76	0.82	0.84	0.86	0.87	0.88	0.89	0.9	0.91	0.91	0.92	0.92	0.93	0.93
	mPOS	0.53	0.7	0.78	0.82	0.84	0.86	0.88	0.89	0.9	0.91	0.91	0.91	0.92	0.92	0.92	0.94
	full set	0.94															

Table 8.46: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 5** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.42	0.57	0.65	0.71	0.75	0.79	0.81	0.83	0.84	0.86	0.87	0.88	0.89	0.9	0.9	0.93
	mRMR	0.42	0.57	0.66	0.72	0.76	0.79	0.81	0.83	0.85	0.86	0.87	0.88	0.88	0.89	0.9	0.93
	Kruskal	0.4	0.53	0.64	0.7	0.75	0.78	0.8	0.81	0.83	0.85	0.86	0.87	0.88	0.89	0.9	0.92
	mPOS	0.42	0.58	0.67	0.72	0.77	0.79	0.82	0.84	0.85	0.86	0.88	0.89	0.89	0.9	0.91	0.94
	full set	0.98															
KNN	LASSO	0.51	0.61	0.68	0.74	0.77	0.79	0.81	0.84	0.85	0.86	0.87	0.88	0.88	0.89	0.89	0.91
	mRMR	0.51	0.62	0.69	0.74	0.77	0.79	0.81	0.83	0.84	0.85	0.86	0.86	0.87	0.88	0.89	0.91
	Kruskal	0.49	0.6	0.68	0.72	0.76	0.79	0.82	0.83	0.85	0.86	0.87	0.86	0.87	0.88	0.88	0.9
	mPOS	0.53	0.62	0.7	0.74	0.77	0.8	0.81	0.83	0.84	0.85	0.86	0.87	0.88	0.88	0.89	0.9
	full set	0.87															
SVM	LASSO	0.53	0.62	0.69	0.74	0.77	0.8	0.83	0.85	0.87	0.88	0.9	0.91	0.92	0.92	0.93	0.96
	mRMR	0.52	0.62	0.69	0.74	0.78	0.81	0.83	0.85	0.87	0.88	0.89	0.9	0.91	0.91	0.92	0.95
	Kruskal	0.5	0.6	0.68	0.71	0.75	0.8	0.83	0.85	0.87	0.88	0.9	0.91	0.92	0.92	0.93	0.95
	mPOS	0.54	0.63	0.69	0.74	0.78	0.8	0.83	0.85	0.87	0.88	0.9	0.91	0.91	0.92	0.93	0.95
	full set	0.98															
XGBoost	LASSO	0.42	0.55	0.64	0.69	0.72	0.75	0.77	0.79	0.8	0.81	0.82	0.82	0.83	0.83	0.84	0.86
	mRMR	0.42	0.55	0.64	0.69	0.73	0.75	0.77	0.78	0.8	0.8	0.81	0.82	0.82	0.82	0.83	0.85
	Kruskal	0.4	0.52	0.62	0.67	0.71	0.74	0.76	0.77	0.8	0.8	0.81	0.82	0.83	0.83	0.83	0.86
	mPOS	0.42	0.55	0.64	0.69	0.73	0.76	0.78	0.79	0.8	0.82	0.82	0.83	0.83	0.83	0.84	0.85
	full set	0.86															

Table 8.47: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 6** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.32	0.43	0.49	0.53	0.57	0.59	0.61	0.63	0.65	0.66	0.68	0.69	0.71	0.72	0.73	0.77
	mRMR	0.33	0.42	0.48	0.53	0.56	0.59	0.61	0.63	0.65	0.67	0.68	0.7	0.71	0.72	0.73	0.77
	Kruskal	0.31	0.4	0.49	0.54	0.56	0.59	0.63	0.64	0.65	0.67	0.68	0.69	0.7	0.71	0.71	0.75
	mPOS	0.31	0.42	0.48	0.52	0.56	0.59	0.6	0.63	0.65	0.66	0.68	0.69	0.7	0.71	0.72	0.77
	full set	0.85															
KNN	LASSO	0.4	0.46	0.52	0.55	0.59	0.61	0.63	0.65	0.65	0.67	0.68	0.69	0.7	0.7	0.71	0.71
	mRMR	0.39	0.46	0.51	0.55	0.57	0.6	0.62	0.62	0.64	0.66	0.66	0.67	0.68	0.69	0.69	0.71
	Kruskal	0.38	0.46	0.53	0.54	0.57	0.6	0.63	0.64	0.66	0.66	0.66	0.67	0.69	0.69	0.7	0.7
	mPOS	0.41	0.47	0.52	0.55	0.58	0.6	0.61	0.63	0.64	0.65	0.66	0.67	0.67	0.68	0.68	0.7
	full set	0.64															
SVM	LASSO	0.42	0.48	0.53	0.56	0.59	0.62	0.65	0.67	0.68	0.7	0.72	0.74	0.75	0.76	0.77	0.82
	mRMR	0.4	0.47	0.52	0.56	0.58	0.62	0.63	0.65	0.67	0.69	0.71	0.72	0.74	0.75	0.76	0.81
	Kruskal	0.4	0.46	0.53	0.53	0.58	0.62	0.65	0.66	0.68	0.69	0.71	0.72	0.74	0.75	0.76	0.79
	mPOS	0.42	0.48	0.51	0.55	0.58	0.61	0.62	0.65	0.67	0.69	0.7	0.72	0.73	0.75	0.76	0.8
	full set	0.83															
XGBoost	LASSO	0.32	0.4	0.46	0.5	0.54	0.57	0.59	0.61	0.62	0.63	0.64	0.66	0.66	0.67	0.68	0.7
	mRMR	0.33	0.4	0.46	0.5	0.53	0.56	0.58	0.61	0.61	0.63	0.63	0.65	0.65	0.67	0.67	0.69
	Kruskal	0.31	0.4	0.46	0.5	0.53	0.55	0.59	0.59	0.61	0.62	0.64	0.64	0.65	0.66	0.67	0.7
	mPOS	0.31	0.4	0.45	0.49	0.53	0.55	0.57	0.59	0.61	0.62	0.63	0.64	0.65	0.66	0.67	0.7
	full set	0.70															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 7** for four-class classification problems, is shown in Table 8.48. The results indicate that mPOS maintains a comparable performance with alternative feature selection techniques when a single informative feature is used. Furthermore, mPOS shows a comprable performance to both LASSO and mRMR, particularly when the number of informative features is smaller, across the RF, KNN, SVM, and XGBoost classifiers.

Table 8.49 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 8** for four-class classification problems. The results reveal that mPOS achieves a performance comparable to that of both LASSO and mRMR when a smaller number of informative features is considered, particularly across RF, KNN, and XGBoost classifiers. However, mPOS only gives a comparable performance to other feature selection techniques when using a single informative feature with a SVM classifier.

Table 8.50 shows the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **scenario 9** for four-class classification problems. The results show that mPOS achieves a performance comparable to that of the LASSO technique when a single informative feature is investigated with the RF classifier. Additionally, mPOS maintains a comparable performance to both LASSO and mRMR when smaller sets of informative features are used across the kNN, SVM, and XGBoost classifiers.

Table 8.48: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 7** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.68	0.86	0.92	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
	mRMR	0.68	0.86	0.92	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1
	Kruskal	0.67	0.77	0.82	0.86	0.87	0.89	0.91	0.93	0.94	0.95	0.96	0.96	0.97	0.98	0.98	0.99
	mPOS	0.68	0.84	0.9	0.92	0.95	0.96	0.97	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99
	full set	1															
KNN	LASSO	0.76	0.88	0.93	0.96	0.98	0.98	0.99	0.99	1	1	1	1	1	1	1	1
	mRMR	0.76	0.88	0.94	0.96	0.98	0.98	0.99	0.99	1	1	1	1	1	1	1	1
	Kruskal	0.74	0.79	0.84	0.87	0.89	0.9	0.92	0.94	0.95	0.96	0.97	0.98	0.98	0.98	0.99	0.99
	mPOS	0.75	0.86	0.91	0.94	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1
	full set	1															
SVM	LASSO	0.76	0.88	0.93	0.96	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1
	mRMR	0.76	0.88	0.94	0.96	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1
	Kruskal	0.75	0.8	0.84	0.87	0.89	0.9	0.92	0.94	0.95	0.96	0.97	0.97	0.97	0.98	0.99	0.99
	mPOS	0.76	0.86	0.91	0.94	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1
	full set	1															
XGBoost	LASSO	0.68	0.84	0.9	0.94	0.94	0.95	0.96	0.96	0.97	0.96	0.97	0.97	0.97	0.97	0.97	0.97
	mRMR	0.68	0.84	0.91	0.93	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97
	Kruskal	0.67	0.75	0.8	0.84	0.85	0.87	0.89	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.96
	mPOS	0.68	0.82	0.88	0.91	0.93	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.97	0.96	0.97	0.97
	full set	0.97															

Table 8.49: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 8** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.44	0.57	0.66	0.71	0.75	0.78	0.8	0.82	0.83	0.85	0.86	0.87	0.87	0.88	0.89	0.9
	mRMR	0.43	0.57	0.66	0.71	0.75	0.78	0.8	0.82	0.84	0.85	0.86	0.87	0.88	0.89	0.89	0.9
	Kruskal	0.42	0.51	0.56	0.6	0.64	0.66	0.68	0.7	0.72	0.73	0.74	0.75	0.77	0.79	0.8	0.84
	mPOS	0.45	0.56	0.63	0.68	0.71	0.74	0.76	0.79	0.8	0.81	0.82	0.83	0.84	0.85	0.85	0.89
	full set	0.94															
KNN	LASSO	0.52	0.62	0.69	0.74	0.77	0.8	0.83	0.85	0.86	0.88	0.89	0.9	0.9	0.91	0.91	0.93
	mRMR	0.52	0.63	0.69	0.74	0.78	0.81	0.83	0.85	0.86	0.88	0.89	0.9	0.9	0.91	0.91	0.93
	Kruskal	0.51	0.55	0.6	0.63	0.65	0.67	0.7	0.72	0.74	0.74	0.75	0.75	0.77	0.79	0.8	0.86
	mPOS	0.53	0.6	0.66	0.7	0.73	0.75	0.78	0.8	0.81	0.83	0.84	0.85	0.86	0.87	0.87	0.91
	full set	0.90															
SVM	LASSO	0.53	0.63	0.69	0.74	0.78	0.81	0.83	0.84	0.87	0.88	0.89	0.9	0.9	0.91	0.91	0.93
	mRMR	0.54	0.64	0.7	0.75	0.79	0.81	0.83	0.85	0.87	0.88	0.89	0.9	0.9	0.91	0.91	0.92
	Kruskal	0.53	0.56	0.61	0.64	0.66	0.68	0.71	0.72	0.73	0.73	0.74	0.75	0.78	0.79	0.8	0.85
	mPOS	0.54	0.61	0.67	0.71	0.73	0.76	0.79	0.81	0.82	0.83	0.84	0.85	0.86	0.87	0.87	0.9
	full set	0.91															
XGBoost	LASSO	0.44	0.56	0.64	0.69	0.72	0.75	0.77	0.78	0.8	0.8	0.81	0.82	0.82	0.83	0.83	0.84
	mRMR	0.43	0.55	0.64	0.69	0.72	0.75	0.75	0.78	0.79	0.8	0.81	0.82	0.82	0.82	0.83	0.83
	Kruskal	0.42	0.48	0.53	0.57	0.6	0.62	0.65	0.66	0.68	0.69	0.7	0.71	0.72	0.74	0.75	0.78
	mPOS	0.45	0.54	0.61	0.65	0.68	0.7	0.72	0.74	0.75	0.76	0.77	0.78	0.79	0.79	0.8	0.82
	full set	0.84															

Table 8.50: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 9** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.33	0.39	0.47	0.49	0.52	0.55	0.56	0.58	0.59	0.6	0.61	0.62	0.62	0.63	0.64	0.65
	mRMR	0.3	0.4	0.45	0.48	0.5	0.53	0.55	0.57	0.58	0.6	0.6	0.61	0.62	0.64	0.64	0.65
	Kruskal	0.31	0.34	0.36	0.41	0.42	0.44	0.46	0.47	0.48	0.5	0.5	0.51	0.52	0.53	0.54	0.6
	mPOS	0.33	0.37	0.41	0.44	0.47	0.49	0.5	0.52	0.54	0.55	0.56	0.57	0.58	0.58	0.59	0.63
	full set	0.69															
KNN	LASSO	0.37	0.44	0.49	0.51	0.53	0.55	0.57	0.59	0.6	0.61	0.63	0.63	0.64	0.64	0.65	0.68
	mRMR	0.38	0.44	0.47	0.5	0.53	0.55	0.57	0.58	0.59	0.61	0.62	0.64	0.65	0.66	0.66	0.68
	Kruskal	0.38	0.37	0.41	0.43	0.45	0.46	0.48	0.49	0.5	0.5	0.5	0.5	0.52	0.53	0.54	0.6
	mPOS	0.39	0.42	0.45	0.46	0.49	0.5	0.52	0.53	0.55	0.55	0.57	0.57	0.58	0.59	0.6	0.65
	full set	0.62															
SVM	LASSO	0.39	0.45	0.49	0.52	0.55	0.57	0.59	0.61	0.62	0.64	0.65	0.66	0.67	0.67	0.67	0.69
	mRMR	0.38	0.45	0.49	0.52	0.54	0.57	0.59	0.61	0.62	0.64	0.65	0.66	0.67	0.68	0.69	0.7
	Kruskal	0.38	0.4	0.43	0.45	0.46	0.48	0.49	0.5	0.5	0.51	0.51	0.52	0.53	0.53	0.54	0.61
	mPOS	0.4	0.43	0.45	0.48	0.5	0.52	0.54	0.55	0.57	0.58	0.59	0.6	0.61	0.62	0.63	0.67
	full set	0.62															
XGBoost	LASSO	0.33	0.37	0.43	0.46	0.5	0.52	0.54	0.54	0.55	0.56	0.57	0.57	0.58	0.59	0.59	0.6
	mRMR	0.3	0.37	0.42	0.46	0.48	0.5	0.52	0.53	0.54	0.55	0.56	0.57	0.58	0.58	0.58	0.6
	Kruskal	0.31	0.32	0.34	0.36	0.38	0.39	0.41	0.43	0.44	0.45	0.45	0.46	0.47	0.49	0.5	0.54
	mPOS	0.33	0.36	0.4	0.42	0.44	0.46	0.47	0.49	0.5	0.5	0.51	0.52	0.53	0.54	0.55	0.57
	full set	0.59															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 10** for four-class classification problems, is shown in Table 8.51. The results show that mPOS achieves comparable performance to both LASSO and mRMR techniques at the different set sizes of informative features across four classifiers.

Table 8.52 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 11** for four-class classification problems. The results reveal that mPOS shows comprable performance to alternative feature selection techniques when a single informative feature is considered with all different classifiers. Moreover, mPOS maintains comparative performance with both the LASSO and mRMR techniques at smaller set sizes of informative features across four classifiers.

Table 8.53 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 12** for four-class classification problems. The results indicate that mPOS shows comparable performance to both the LASSO and mRMR at the different set sizes of informative feature across four classifiers.

Table 8.51: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 10** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.52	0.72	0.8	0.84	0.86	0.88	0.89	0.9	0.91	0.92	0.92	0.93	0.93	0.94	0.94	0.95
	mRMR	0.53	0.72	0.79	0.84	0.87	0.89	0.9	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.96
	Kruskal	0.54	0.62	0.69	0.75	0.76	0.78	0.79	0.82	0.83	0.85	0.85	0.86	0.87	0.89	0.9	0.93
	mPOS	0.55	0.71	0.77	0.82	0.84	0.86	0.87	0.88	0.89	0.9	0.91	0.92	0.92	0.93	0.93	0.95
	full set	0.98															
KNN	LASSO	0.64	0.75	0.81	0.85	0.88	0.89	0.91	0.92	0.92	0.93	0.93	0.94	0.94	0.94	0.94	0.95
	mRMR	0.64	0.75	0.81	0.85	0.88	0.9	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.95	0.96
	Kruskal	0.62	0.67	0.71	0.73	0.76	0.78	0.8	0.82	0.83	0.83	0.84	0.85	0.86	0.87	0.87	0.91
	mPOS	0.64	0.74	0.78	0.82	0.84	0.86	0.87	0.89	0.89	0.9	0.91	0.91	0.92	0.92	0.92	0.94
	full set	0.92															
SVM	LASSO	0.64	0.75	0.81	0.85	0.88	0.9	0.91	0.93	0.93	0.94	0.95	0.95	0.95	0.96	0.96	0.96
	mRMR	0.64	0.75	0.81	0.85	0.88	0.9	0.92	0.93	0.94	0.94	0.95	0.95	0.96	0.96	0.96	0.97
	Kruskal	0.64	0.67	0.73	0.74	0.76	0.79	0.81	0.83	0.84	0.85	0.85	0.86	0.88	0.88	0.89	0.92
	mPOS	0.65	0.74	0.78	0.83	0.85	0.87	0.88	0.9	0.91	0.92	0.92	0.93	0.94	0.94	0.95	0.96
	full set	0.96															
XGBoost	LASSO	0.52	0.69	0.78	0.82	0.84	0.86	0.87	0.88	0.88	0.89	0.89	0.89	0.89	0.9	0.9	0.9
	mRMR	0.53	0.7	0.77	0.81	0.84	0.86	0.87	0.88	0.89	0.89	0.89	0.9	0.9	0.9	0.91	0.91
	Kruskal	0.54	0.61	0.67	0.73	0.74	0.75	0.77	0.79	0.81	0.81	0.82	0.82	0.83	0.84	0.85	0.87
	mPOS	0.55	0.69	0.75	0.8	0.81	0.83	0.84	0.85	0.87	0.87	0.87	0.88	0.88	0.88	0.89	0.9
	full set	0.91															

Table 8.52: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 11** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.44	0.58	0.66	0.72	0.75	0.77	0.79	0.81	0.82	0.83	0.84	0.85	0.85	0.86	0.87	0.88
	mRMR	0.42	0.57	0.65	0.71	0.75	0.77	0.8	0.81	0.82	0.84	0.85	0.85	0.86	0.87	0.87	0.89
	Kruskal	0.42	0.49	0.55	0.62	0.64	0.66	0.68	0.7	0.72	0.73	0.74	0.74	0.76	0.77	0.78	0.82
	mPOS	0.43	0.55	0.62	0.67	0.7	0.73	0.75	0.77	0.79	0.8	0.81	0.82	0.83	0.84	0.85	0.88
	full set	0.93															
KNN	LASSO	0.51	0.62	0.69	0.73	0.76	0.78	0.8	0.81	0.82	0.82	0.83	0.84	0.85	0.85	0.85	0.85
	mRMR	0.51	0.62	0.69	0.73	0.76	0.78	0.8	0.81	0.82	0.83	0.83	0.84	0.84	0.85	0.85	0.86
	Kruskal	0.51	0.53	0.58	0.62	0.64	0.65	0.67	0.69	0.7	0.7	0.71	0.72	0.73	0.74	0.75	0.78
	mPOS	0.52	0.59	0.65	0.69	0.71	0.73	0.75	0.76	0.77	0.79	0.79	0.8	0.8	0.81	0.81	0.83
	full set	0.80															
SVM	LASSO	0.52	0.63	0.68	0.73	0.75	0.78	0.8	0.82	0.83	0.85	0.86	0.87	0.87	0.88	0.88	0.9
	mRMR	0.53	0.62	0.68	0.73	0.77	0.79	0.81	0.83	0.84	0.85	0.86	0.87	0.88	0.88	0.89	0.9
	Kruskal	0.52	0.54	0.59	0.62	0.64	0.66	0.69	0.69	0.71	0.71	0.71	0.72	0.74	0.76	0.77	0.81
	mPOS	0.52	0.59	0.65	0.69	0.71	0.73	0.75	0.76	0.77	0.79	0.79	0.8	0.8	0.81	0.81	0.83
	full set	0.86															
XGBoost	LASSO	0.44	0.55	0.63	0.69	0.73	0.74	0.76	0.77	0.78	0.79	0.79	0.8	0.8	0.81	0.81	0.82
	mRMR	0.42	0.54	0.63	0.69	0.71	0.74	0.75	0.77	0.78	0.79	0.79	0.8	0.8	0.8	0.81	0.82
	Kruskal	0.42	0.46	0.5	0.57	0.59	0.61	0.64	0.67	0.69	0.7	0.7	0.71	0.72	0.72	0.74	0.77
	mPOS	0.43	0.53	0.6	0.64	0.67	0.69	0.72	0.73	0.75	0.76	0.77	0.77	0.78	0.78	0.78	0.8
	full set	0.82															

Table 8.53: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 12** for four-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.34	0.41	0.48	0.54	0.57	0.6	0.62	0.64	0.65	0.66	0.67	0.68	0.68	0.69	0.7	0.72
	mRMR	0.33	0.42	0.48	0.52	0.55	0.59	0.61	0.62	0.64	0.65	0.66	0.68	0.69	0.69	0.7	0.72
	Kruskal	0.33	0.39	0.43	0.46	0.48	0.49	0.52	0.52	0.53	0.54	0.56	0.56	0.59	0.6	0.6	0.65
	mPOS	0.34	0.43	0.48	0.51	0.55	0.57	0.58	0.59	0.61	0.62	0.62	0.64	0.64	0.65	0.65	0.68
	full set	0.76															
KNN	LASSO	0.39	0.46	0.51	0.55	0.58	0.61	0.63	0.63	0.64	0.65	0.65	0.66	0.66	0.67	0.67	0.67
	mRMR	0.39	0.46	0.51	0.55	0.57	0.59	0.61	0.62	0.63	0.64	0.65	0.66	0.67	0.67	0.67	0.67
	Kruskal	0.4	0.43	0.46	0.47	0.48	0.49	0.5	0.51	0.52	0.53	0.53	0.54	0.56	0.57	0.58	0.59
	mPOS	0.41	0.47	0.51	0.54	0.56	0.58	0.59	0.59	0.6	0.6	0.61	0.62	0.62	0.62	0.63	0.64
	full set	0.61															
SVM	LASSO	0.41	0.46	0.51	0.55	0.58	0.61	0.63	0.65	0.66	0.67	0.68	0.69	0.7	0.71	0.72	0.73
	mRMR	0.4	0.46	0.52	0.55	0.58	0.6	0.62	0.64	0.66	0.67	0.68	0.7	0.71	0.72	0.72	0.74
	Kruskal	0.41	0.43	0.46	0.48	0.48	0.49	0.51	0.52	0.53	0.53	0.54	0.55	0.57	0.57	0.58	0.63
	mPOS	0.41	0.47	0.51	0.54	0.56	0.58	0.59	0.59	0.6	0.6	0.61	0.62	0.62	0.62	0.63	0.64
	full set	0.65															
XGBoost	LASSO	0.34	0.4	0.45	0.5	0.54	0.56	0.59	0.6	0.62	0.62	0.63	0.63	0.64	0.64	0.64	0.65
	mRMR	0.33	0.39	0.45	0.5	0.53	0.55	0.58	0.59	0.61	0.61	0.63	0.64	0.64	0.64	0.65	0.66
	Kruskal	0.33	0.36	0.39	0.42	0.44	0.45	0.47	0.49	0.51	0.52	0.51	0.52	0.55	0.56	0.57	0.6
	mPOS	0.34	0.41	0.46	0.5	0.52	0.53	0.54	0.56	0.56	0.58	0.58	0.6	0.6	0.61	0.61	0.63
	full set	0.64															

For the evaluation of mPOS method in five-class classification problems, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers across Scenario 1 to Scenario 12 is shown in Tables 8.54 - Table 8.65, respectively. For **Scenario 1**, the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers is shown in Table 8.54. mPOS shows comparable performance to mRMR and Kruskal at different set sizes of informative genes across four classifiers. Specifically, when considering sets composed of 1-2 informative features, mPOS demonstrates comparable performance to alternative feature selection techniques.

Table 8.55 demonstrates the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 2** for five-class classification problems. mPOS achieves comparable performance to mRMR and Kruskal methods across four different classifiers.

Table 8.56 shows the average classification accuracy for the RF, kNN, SVM, and XGBoost classifiers on **Scenario 3** for five-class classification problems. mPOS shows comparable performance to LASSO, mRMR, and Kruskal at a single informative feature. However, mPOS performs similarly to mRMR and Kruskal when larger set sizes of informative feature are considered across all classifiers.

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 4** for five-class classification problems is shown in Table 8.57. mPOS shows comparable performance to competitive feature selection techniques when selecting a single informative feature. In contrast, LASSO demonstrates inferior performance, as the set size of informative features increases across all classifiers.

Table 8.58 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 5** for five-class classification problems. mPOS demonstrates comparable performance to all other feature selection techniques when selecting a single informative feature. However, as larger sets of informative features are considered, mPOS shows performance on par with mRMR and Kruskal.

Table 8.59 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 6** for five-class classification problems. mPOS shows a comparable performance to mRMR and Kruskal when selecting larger set sizes of informative features across all classifiers.

Table 8.54: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 1** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.65	0.68	0.68	0.72	0.76	0.78	0.80	0.81	0.83	0.84	0.84	0.86	0.87	0.88	0.88	0.92
	mRMR	0.65	0.85	0.91	0.95	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	1	1	1	1
	Kruskal	0.67	0.86	0.91	0.94	0.96	0.97	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1
	mPOS	0.67	0.86	0.92	0.95	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	1	1	1	1
	full set	1															
KNN	LASSO	0.72	0.75	0.75	0.77	0.81	0.82	0.84	0.84	0.85	0.87	0.87	0.88	0.88	0.89	0.89	0.92
	mRMR	0.74	0.87	0.93	0.96	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	Kruskal	0.74	0.87	0.92	0.96	0.98	0.98	0.99	0.99	1	1	1	1	1	1	1	1
	mPOS	0.74	0.88	0.94	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	full set	0.98															
SVM	LASSO	0.73	0.75	0.75	0.78	0.81	0.82	0.84	0.85	0.86	0.87	0.87	0.88	0.89	0.90	0.90	0.93
	mRMR	0.74	0.87	0.93	0.96	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	Kruskal	0.74	0.87	0.92	0.95	0.97	0.98	0.99	0.99	1	1	1	1	1	1	1	1
	mPOS	0.75	0.88	0.94	0.97	0.98	0.99	0.99	0.99	1	1	1	1	1	1	1	1
	full set	1															
XGBoost	LASSO	0.65	0.68	0.68	0.71	0.75	0.77	0.79	0.80	0.82	0.83	0.83	0.85	0.86	0.86	0.87	0.90
	mRMR	0.65	0.83	0.90	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.98	0.98	0.98	0.97	0.98
	Kruskal	0.67	0.84	0.89	0.93	0.94	0.95	0.96	0.96	0.97	0.97	0.97	0.98	0.97	0.98	0.98	0.98
	mPOS	0.67	0.84	0.91	0.94	0.95	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.98	0.98	0.98
	full set	0.98															

Table 8.55: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 2** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.40	0.42	0.43	0.44	0.46	0.47	0.48	0.50	0.51	0.52	0.53	0.54	0.56	0.56	0.58	0.63
	mRMR	0.40	0.55	0.64	0.70	0.74	0.78	0.80	0.83	0.85	0.86	0.87	0.89	0.90	0.91	0.91	0.94
	Kruskal	0.41	0.55	0.64	0.70	0.74	0.78	0.81	0.83	0.85	0.86	0.87	0.89	0.89	0.90	0.91	0.95
	mPOS	0.41	0.55	0.65	0.71	0.75	0.78	0.81	0.83	0.85	0.87	0.89	0.90	0.91	0.91	0.92	0.94
	full set	1															
KNN	LASSO	0.49	0.50	0.52	0.53	0.54	0.55	0.55	0.57	0.58	0.58	0.59	0.60	0.61	0.61	0.63	0.66
	mRMR	0.48	0.60	0.67	0.72	0.77	0.80	0.83	0.86	0.88	0.90	0.91	0.92	0.93	0.94	0.94	0.97
	Kruskal	0.50	0.61	0.68	0.73	0.78	0.80	0.83	0.85	0.87	0.89	0.90	0.91	0.92	0.94	0.95	0.97
	mPOS	0.48	0.6	0.68	0.74	0.77	0.81	0.84	0.87	0.89	0.90	0.92	0.93	0.93	0.94	0.95	0.97
	full set	0.91															
SVM	LASSO	0.51	0.52	0.53	0.54	0.55	0.56	0.56	0.58	0.59	0.59	0.6	0.61	0.62	0.62	0.64	0.68
	mRMR	0.5	0.61	0.68	0.73	0.77	0.81	0.84	0.86	0.88	0.89	0.91	0.92	0.93	0.93	0.94	0.96
	Kruskal	0.5	0.62	0.68	0.73	0.78	0.8	0.84	0.86	0.88	0.89	0.9	0.91	0.92	0.93	0.94	0.96
	mPOS	0.5	0.61	0.69	0.74	0.78	0.81	0.85	0.87	0.88	0.9	0.91	0.92	0.93	0.94	0.95	0.97
	full set	0.99															
XGBoost	LASSO	0.40	0.41	0.43	0.44	0.45	0.46	0.47	0.49	0.50	0.51	0.51	0.53	0.54	0.55	0.56	0.60
	mRMR	0.40	0.54	0.62	0.67	0.71	0.74	0.76	0.78	0.80	0.81	0.82	0.83	0.84	0.85	0.85	0.87
	Kruskal	0.41	0.55	0.63	0.67	0.71	0.74	0.76	0.78	0.80	0.80	0.81	0.82	0.84	0.84	0.85	0.87
	mPOS	0.41	0.54	0.63	0.68	0.72	0.75	0.78	0.8	0.81	0.82	0.83	0.84	0.85	0.85	0.86	0.88
	full set	0.91															

Table 8.56: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 3** for five class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.30	0.30	0.30	0.32	0.33	0.33	0.34	0.34	0.35	0.35	0.35	0.36	0.36	0.37	0.37	0.40
	mRMR	0.28	0.35	0.41	0.45	0.49	0.51	0.54	0.55	0.57	0.59	0.60	0.62	0.62	0.64	0.65	0.70
	Kruskal	0.28	0.36	0.44	0.47	0.49	0.53	0.55	0.57	0.58	0.60	0.60	0.61	0.62	0.64	0.66	0.69
	mPOS	0.27	0.34	0.41	0.45	0.49	0.51	0.53	0.55	0.57	0.59	0.61	0.62	0.63	0.64	0.65	0.70
	full set	0.80															
KNN	LASSO	0.34	0.35	0.35	0.36	0.37	0.38	0.38	0.39	0.39	0.39	0.40	0.40	0.41	0.41	0.42	0.44
	mRMR	0.34	0.40	0.44	0.48	0.51	0.53	0.55	0.57	0.59	0.61	0.63	0.65	0.66	0.67	0.69	0.75
	Kruskal	0.33	0.40	0.45	0.48	0.52	0.54	0.58	0.59	0.6	0.62	0.62	0.63	0.66	0.67	0.69	0.75
	mPOS	0.34	0.39	0.44	0.47	0.5	0.53	0.55	0.58	0.6	0.62	0.64	0.65	0.67	0.69	0.70	0.76
	full set	0.69															
SVM	LASSO	0.36	0.36	0.37	0.38	0.39	0.40	0.40	0.41	0.41	0.41	0.41	0.42	0.43	0.43	0.44	0.46
	mRMR	0.36	0.42	0.46	0.50	0.53	0.56	0.58	0.59	0.62	0.63	0.65	0.66	0.67	0.69	0.71	0.76
	Kruskal	0.35	0.41	0.46	0.49	0.54	0.56	0.59	0.62	0.63	0.65	0.66	0.67	0.68	0.69	0.71	0.76
	mPOS	0.35	0.41	0.45	0.49	0.52	0.55	0.58	0.61	0.62	0.64	0.66	0.67	0.69	0.7	0.72	0.77
	full set	0.81															
XGBoost	LASSO	0.30	0.30	0.30	0.32	0.33	0.33	0.34	0.34	0.34	0.35	0.35	0.35	0.36	0.36	0.37	0.39
	mRMR	0.28	0.34	0.39	0.43	0.46	0.49	0.51	0.52	0.54	0.56	0.56	0.57	0.59	0.59	0.60	0.62
	Kruskal	0.28	0.35	0.41	0.44	0.47	0.49	0.52	0.54	0.55	0.56	0.56	0.58	0.58	0.59	0.60	0.63
	mPOS	0.27	0.34	0.39	0.42	0.46	0.49	0.51	0.53	0.55	0.56	0.57	0.59	0.60	0.61	0.61	0.63
	full set	0.68															

Table 8.57: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 4** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.47	0.48	0.53	0.54	0.55	0.56	0.59	0.61	0.61	0.63	0.64	0.64	0.66	0.67	0.68	0.73
	mRMR	0.48	0.66	0.75	0.8	0.83	0.86	0.88	0.9	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	Kruskal	0.5	0.66	0.74	0.79	0.83	0.86	0.88	0.89	0.91	0.92	0.93	0.94	0.94	0.95	0.95	0.97
	mPOS	0.49	0.66	0.75	0.8	0.84	0.87	0.88	0.9	0.91	0.93	0.93	0.94	0.94	0.95	0.95	0.97
	full set	1															
KNN	LASSO	0.58	0.58	0.61	0.62	0.63	0.63	0.65	0.66	0.67	0.68	0.69	0.69	0.70	0.71	0.72	0.75
	mRMR	0.58	0.70	0.77	0.82	0.84	0.87	0.89	0.90	0.92	0.92	0.93	0.94	0.95	0.95	0.96	0.97
	Kruskal	0.58	0.71	0.77	0.82	0.85	0.86	0.88	0.90	0.91	0.93	0.93	0.94	0.94	0.95	0.96	0.97
	mPOS	0.58	0.70	0.77	0.82	0.86	0.88	0.89	0.91	0.92	0.93	0.94	0.95	0.95	0.96	0.96	0.98
	full set	0.92															
SVM	LASSO	0.58	0.59	0.62	0.62	0.63	0.64	0.65	0.67	0.67	0.69	0.7	0.7	0.71	0.71	0.72	0.76
	mRMR	0.58	0.7	0.77	0.82	0.85	0.88	0.9	0.92	0.93	0.94	0.95	0.95	0.96	0.96	0.97	0.98
	Kruskal	0.59	0.71	0.76	0.82	0.86	0.88	0.89	0.91	0.93	0.93	0.94	0.95	0.95	0.96	0.96	0.98
	mPOS	0.59	0.7	0.78	0.82	0.86	0.89	0.91	0.92	0.93	0.94	0.95	0.96	0.96	0.96	0.97	0.98
	full set	1															
XGBoost	LASSO	0.47	0.48	0.52	0.53	0.54	0.55	0.58	0.59	0.6	0.62	0.63	0.63	0.65	0.65	0.67	0.71
	mRMR	0.48	0.64	0.72	0.78	0.8	0.83	0.85	0.86	0.87	0.87	0.89	0.89	0.9	0.9	0.9	0.92
	Kruskal	0.5	0.64	0.72	0.76	0.8	0.83	0.84	0.85	0.87	0.87	0.88	0.89	0.89	0.9	0.89	0.91
	mPOS	0.49	0.64	0.73	0.78	0.81	0.84	0.85	0.86	0.87	0.88	0.89	0.89	0.89	0.9	0.91	0.92
	full set																

Table 8.58: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 5** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.37	0.40	0.41	0.42	0.43	0.45	0.46	0.46	0.47	0.49	0.50	0.51	0.53	0.53	0.55	0.59
	mRMR	0.37	0.51	0.60	0.65	0.69	0.73	0.75	0.77	0.79	0.81	0.83	0.84	0.85	0.86	0.87	0.91
	Kruskal	0.38	0.51	0.59	0.65	0.70	0.74	0.76	0.78	0.8	0.81	0.82	0.84	0.85	0.86	0.87	0.90
	mPOS	0.38	0.51	0.6	0.66	0.70	0.73	0.76	0.77	0.79	0.81	0.83	0.84	0.85	0.86	0.87	0.90
	full set	0.98															
KNN	LASSO	0.47	0.48	0.49	0.50	0.51	0.52	0.53	0.53	0.54	0.55	0.56	0.57	0.57	0.58	0.59	0.62
	mRMR	0.46	0.55	0.63	0.67	0.71	0.74	0.76	0.78	0.80	0.81	0.82	0.83	0.84	0.84	0.85	0.88
	Kruskal	0.48	0.57	0.64	0.68	0.72	0.74	0.76	0.78	0.79	0.80	0.81	0.81	0.82	0.84	0.84	0.87
	mPOS	0.46	0.55	0.63	0.67	0.71	0.74	0.76	0.78	0.80	0.82	0.83	0.83	0.85	0.85	0.86	0.89
	full set	0.81															
SVM	LASSO	0.48	0.49	0.50	0.50	0.51	0.53	0.53	0.54	0.54	0.55	0.56	0.57	0.59	0.59	0.60	0.63
	mRMR	0.47	0.56	0.63	0.68	0.72	0.75	0.78	0.80	0.82	0.84	0.85	0.87	0.87	0.89	0.90	0.92
	Kruskal	0.48	0.57	0.64	0.68	0.72	0.75	0.78	0.80	0.81	0.83	0.85	0.86	0.88	0.89	0.90	0.94
	mPOS	0.47	0.56	0.63	0.68	0.72	0.75	0.78	0.80	0.82	0.84	0.86	0.87	0.88	0.89	0.90	0.93
	full set	0.97															
XGBoost	LASSO	0.37	0.39	0.41	0.41	0.43	0.44	0.45	0.45	0.46	0.48	0.49	0.50	0.52	0.52	0.53	0.57
	mRMR	0.37	0.49	0.58	0.63	0.67	0.70	0.72	0.73	0.75	0.76	0.77	0.79	0.79	0.80	0.80	0.83
	Kruskal	0.38	0.48	0.58	0.63	0.67	0.71	0.72	0.74	0.75	0.77	0.77	0.78	0.79	0.79	0.80	0.82
	mPOS	0.38	0.50	0.58	0.63	0.66	0.69	0.72	0.74	0.75	0.76	0.77	0.79	0.79	0.8	0.80	0.83
	full set	0.86															

Table 8.59: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 6** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.37	0.40	0.41	0.42	0.43	0.45	0.46	0.46	0.47	0.49	0.50	0.51	0.53	0.53	0.55	0.59
	mRMR	0.28	0.37	0.43	0.47	0.51	0.53	0.56	0.57	0.59	0.60	0.62	0.64	0.65	0.67	0.68	0.73
	Kruskal	0.30	0.37	0.44	0.48	0.51	0.54	0.57	0.59	0.60	0.62	0.63	0.64	0.65	0.67	0.68	0.72
	mPOS	0.27	0.39	0.44	0.49	0.52	0.55	0.57	0.59	0.60	0.62	0.64	0.65	0.66	0.68	0.69	0.73
	full set	0.84															
KNN	LASSO	0.36	0.37	0.38	0.39	0.39	0.40	0.41	0.41	0.42	0.42	0.42	0.43	0.44	0.44	0.44	0.46
	mRMR	0.34	0.42	0.46	0.49	0.52	0.55	0.57	0.58	0.59	0.60	0.61	0.62	0.63	0.64	0.65	0.67
	Kruskal	0.34	0.42	0.47	0.50	0.54	0.55	0.58	0.58	0.60	0.60	0.61	0.62	0.63	0.63	0.62	0.66
	mPOS	0.36	0.43	0.46	0.50	0.53	0.55	0.58	0.59	0.61	0.62	0.63	0.63	0.65	0.65	0.66	0.68
	full set	0.61															
SVM	LASSO	0.37	0.38	0.39	0.39	0.40	0.41	0.42	0.42	0.43	0.43	0.43	0.44	0.45	0.45	0.45	0.47
	mRMR	0.36	0.42	0.46	0.50	0.53	0.56	0.58	0.60	0.62	0.64	0.66	0.67	0.69	0.70	0.71	0.77
	Kruskal	0.36	0.42	0.47	0.51	0.55	0.57	0.61	0.63	0.64	0.65	0.66	0.68	0.70	0.71	0.72	0.77
	mPOS	0.37	0.43	0.47	0.51	0.54	0.57	0.60	0.62	0.64	0.66	0.68	0.69	0.70	0.72	0.73	0.78
	full set	0.81															
XGBoost	LASSO	0.29	0.30	0.31	0.31	0.32	0.33	0.33	0.34	0.35	0.35	0.36	0.36	0.37	0.37	0.37	0.40
	mRMR	0.28	0.35	0.40	0.45	0.49	0.51	0.53	0.55	0.56	0.58	0.59	0.60	0.61	0.62	0.62	0.65
	Kruskal	0.30	0.36	0.39	0.43	0.47	0.51	0.54	0.56	0.57	0.59	0.60	0.60	0.61	0.62	0.63	0.67
	mPOS	0.27	0.36	0.41	0.46	0.49	0.52	0.54	0.56	0.58	0.59	0.60	0.61	0.62	0.63	0.63	0.67
	full set	0.70															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 7** for five-class classification problems is shown in Table 8.60. mPOS shows comparable performance to alternative feature selection techniques when selecting a single informative feature across all classifiers. However, mRMR achieves superior performance at smaller set sizes of informative features. As the set size of informative features increases, mPOS demonstrates comparable performance to mRMR and Kruskal.

Table 8.61 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 8** for five-class classification problems. mPOS demonstrates comparable performance to other feature selection techniques at a single informative feature across all classifiers. However, both mPOS and mRMR show superior performance when considering larger set sizes of informative features across RF, KNN, and SVM classifiers.

Table 8.62 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 9** for five-class classification problems. mPOS shows comparable performance to alternative feature selection at a single informative feature. In contrast, mRMR outperforms all other feature selection techniques when considering larger set sizes of informative features across all classifiers.

Table 8.60: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 7** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.64	0.67	0.68	0.68	0.69	0.69	0.70	0.70	0.71	0.72	0.73	0.75	0.77	0.79	0.80	0.85
	mRMR	0.66	0.85	0.92	0.95	0.96	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99	1	1	1
	Kruskal	0.64	0.77	0.82	0.86	0.89	0.91	0.92	0.94	0.94	0.95	0.96	0.96	0.96	0.97	0.97	0.98
	mPOS	0.65	0.83	0.89	0.92	0.94	0.95	0.96	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	1
	full set	1															
KNN	LASSO	0.74	0.76	0.76	0.76	0.77	0.77	0.77	0.78	0.78	0.79	0.79	0.81	0.82	0.83	0.84	0.88
	mRMR	0.74	0.88	0.93	0.96	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	Kruskal	0.74	0.81	0.84	0.88	0.91	0.93	0.93	0.95	0.95	0.97	0.97	0.97	0.97	0.98	0.98	0.99
	mPOS	0.74	0.85	0.90	0.93	0.95	0.96	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1
	full set	0.97															
SVM	LASSO	0.75	0.76	0.77	0.77	0.77	0.77	0.78	0.78	0.79	0.79	0.80	0.82	0.82	0.84	0.85	0.88
	mRMR	0.75	0.88	0.93	0.96	0.98	0.99	0.99	1	1	1	1	1	1	1	1	1
	Kruskal	0.75	0.81	0.84	0.88	0.91	0.93	0.93	0.95	0.95	0.96	0.97	0.97	0.97	0.98	0.98	0.99
	mPOS	0.74	0.86	0.90	0.93	0.95	0.97	0.98	0.98	0.99	0.99	0.99	1	1	1	1	1
	full set	1															
XGBoost	LASSO	0.64	0.67	0.67	0.67	0.69	0.69	0.69	0.70	0.71	0.71	0.72	0.75	0.76	0.78	0.79	0.84
	mRMR	0.66	0.83	0.91	0.93	0.95	0.96	0.96	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.98
	Kruskal	0.64	0.75	0.80	0.84	0.87	0.89	0.90	0.92	0.93	0.93	0.94	0.94	0.94	0.95	0.95	0.96
	mPOS	0.65	0.81	0.87	0.90	0.92	0.93	0.94	0.95	0.95	0.96	0.96	0.96	0.96	0.96	0.97	0.97
	full set	0.98															

Table 8.61: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 8** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.42	0.43	0.44	0.45	0.45	0.47	0.48	0.48	0.50	0.50	0.51	0.52	0.53	0.53	0.54	0.57
	mRMR	0.40	0.55	0.64	0.70	0.74	0.77	0.80	0.82	0.84	0.85	0.86	0.87	0.88	0.89	0.89	0.92
	Kruskal	0.42	0.48	0.52	0.58	0.61	0.64	0.66	0.70	0.72	0.73	0.75	0.77	0.78	0.78	0.80	0.83
	mPOS	0.41	0.54	0.62	0.67	0.71	0.74	0.76	0.78	0.79	0.81	0.83	0.84	0.85	0.86	0.87	0.90
	full set	0.96															
KNN	LASSO	0.48	0.49	0.50	0.50	0.51	0.52	0.53	0.54	0.55	0.55	0.56	0.57	0.58	0.58	0.59	0.61
	mRMR	0.49	0.60	0.68	0.72	0.77	0.80	0.83	0.85	0.86	0.88	0.89	0.90	0.91	0.92	0.92	0.95
	Kruskal	0.48	0.54	0.55	0.59	0.62	0.65	0.67	0.71	0.73	0.75	0.76	0.78	0.79	0.80	0.81	0.85
	mPOS	0.50	0.60	0.67	0.71	0.74	0.76	0.78	0.8	0.81	0.83	0.84	0.86	0.87	0.87	0.88	0.91
	full set	0.85															
SVM	LASSO	0.50	0.51	0.51	0.52	0.53	0.54	0.55	0.56	0.57	0.57	0.58	0.58	0.59	0.60	0.61	0.63
	mRMR	0.50	0.61	0.68	0.73	0.77	0.8	0.84	0.85	0.87	0.88	0.89	0.90	0.91	0.91	0.92	0.94
	Kruskal	0.50	0.54	0.58	0.59	0.62	0.66	0.68	0.72	0.73	0.75	0.76	0.77	0.78	0.79	0.81	0.83
	mPOS	0.51	0.61	0.67	0.71	0.74	0.77	0.79	0.80	0.82	0.84	0.85	0.86	0.87	0.88	0.89	0.91
	full set	0.92															
XGBoost	LASSO	0.42	0.43	0.44	0.44	0.45	0.47	0.47	0.48	0.49	0.50	0.51	0.51	0.52	0.52	0.53	0.56
	mRMR	0.40	0.53	0.61	0.67	0.71	0.73	0.76	0.78	0.79	0.80	0.81	0.81	0.83	0.83	0.84	0.86
	Kruskal	0.50	0.54	0.58	0.59	0.62	0.66	0.68	0.72	0.73	0.75	0.76	0.77	0.78	0.79	0.81	0.83
	mPOS	0.41	0.53	0.61	0.65	0.68	0.71	0.72	0.74	0.75	0.76	0.78	0.78	0.79	0.80	0.81	0.83
	full set	0.86															

Table 8.62: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 9** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.27	0.28	0.29	0.3	0.3	0.32	0.33	0.33	0.34	0.33	0.34	0.34	0.34	0.35	0.36	0.39
	mRMR	0.28	0.36	0.41	0.45	0.48	0.51	0.53	0.55	0.56	0.58	0.59	0.6	0.62	0.63	0.63	0.65
	Kruskal	0.27	0.34	0.36	0.38	0.39	0.42	0.43	0.45	0.46	0.48	0.49	0.49	0.51	0.51	0.52	0.56
	mPOS	0.26	0.33	0.38	0.42	0.45	0.47	0.49	0.5	0.52	0.53	0.54	0.55	0.56	0.57	0.58	0.6
	full set	0.69															
KNN	LASSO	0.33	0.34	0.35	0.35	0.35	0.36	0.37	0.37	0.37	0.37	0.38	0.38	0.38	0.39	0.4	0.42
	mRMR	0.34	0.4	0.45	0.48	0.51	0.53	0.55	0.57	0.59	0.6	0.61	0.62	0.64	0.64	0.65	0.68
	Kruskal	0.32	0.37	0.38	0.4	0.4	0.42	0.44	0.46	0.48	0.49	0.49	0.5	0.51	0.52	0.54	0.57
	mPOS	0.33	0.39	0.42	0.45	0.48	0.49	0.51	0.52	0.54	0.55	0.56	0.57	0.57	0.58	0.58	0.62
	full set	0.61															
SVM	LASSO	0.35	0.35	0.36	0.36	0.37	0.38	0.39	0.39	0.39	0.39	0.4	0.4	0.4	0.41	0.42	0.44
	mRMR	0.35	0.42	0.46	0.5	0.53	0.55	0.57	0.59	0.6	0.62	0.63	0.64	0.65	0.66	0.67	0.69
	Kruskal	0.34	0.37	0.39	0.41	0.41	0.44	0.45	0.48	0.48	0.49	0.5	0.5	0.52	0.52	0.54	0.58
	mPOS	0.35	0.4	0.44	0.47	0.49	0.51	0.53	0.54	0.55	0.56	0.57	0.58	0.59	0.6	0.6	0.63
	full set	0.62															
XGBoost	LASSO	0.27	0.28	0.29	0.29	0.3	0.31	0.32	0.33	0.33	0.33	0.33	0.33	0.34	0.35	0.35	0.38
	mRMR	0.28	0.34	0.39	0.42	0.46	0.48	0.5	0.51	0.53	0.54	0.55	0.56	0.57	0.57	0.58	0.59
	Kruskal	0.27	0.3	0.32	0.34	0.36	0.38	0.4	0.42	0.43	0.43	0.44	0.45	0.46	0.46	0.47	0.52
	mPOS	0.26	0.32	0.36	0.4	0.43	0.44	0.46	0.47	0.48	0.5	0.51	0.51	0.52	0.53	0.54	0.55
	full set	0.60															

The average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 10** for five-class classification problems, is shown in Table 8.63. mPOS achieves comparable performance to other feature selection techniques at a single informative feature. However, as the set sizes of informative feature increases, both mRMR and mPOS show superior performance across all classifiers.

Table 8.64 demonstrates the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 11** for five-class classification problems. mPOS performs similarly to other feature selection techniques when selecting a single informative feature. However, as the number of informative features increases, both mRMR and mPOS exhibit superior performance across all classifiers.

Table 8.65 shows the average classification accuracy for RF, kNN, SVM, and XGBoost classifiers on **Scenario 12** for five-class classification problems. Both mRMR and mPOS perform better than alternative feature selection techniques when applied to RF and XGboost classifiers. However, mRMR show superior performance when selecting larger set sizes of informative feature across KNN and SVM classifiers.

Table 8.63: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 10** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.49	0.49	0.49	0.50	0.50	0.51	0.51	0.52	0.54	0.55	0.56	0.57	0.58	0.60	0.60	0.63
	mRMR	0.48	0.65	0.75	0.80	0.83	0.85	0.87	0.89	0.90	0.91	0.91	0.92	0.93	0.93	0.93	0.95
	Kruskal	0.49	0.58	0.62	0.68	0.70	0.73	0.77	0.79	0.81	0.82	0.83	0.83	0.84	0.85	0.86	0.89
	mPOS	0.46	0.64	0.73	0.77	0.81	0.84	0.85	0.87	0.88	0.89	0.90	0.90	0.91	0.92	0.92	0.94
	full set	0.98															
KNN	LASSO	0.56	0.56	0.57	0.57	0.57	0.58	0.58	0.59	0.61	0.61	0.62	0.62	0.64	0.64	0.65	0.67
	mRMR	0.58	0.69	0.77	0.82	0.84	0.87	0.88	0.90	0.91	0.92	0.92	0.93	0.93	0.94	0.94	0.95
	Kruskal	0.56	0.64	0.66	0.70	0.72	0.74	0.77	0.79	0.81	0.80	0.82	0.83	0.83	0.84	0.85	0.88
	mPOS	0.57	0.69	0.75	0.79	0.82	0.84	0.86	0.87	0.88	0.89	0.90	0.91	0.91	0.92	0.92	0.93
	full set	0.87															
SVM	LASSO	0.58	0.58	0.58	0.59	0.59	0.59	0.60	0.60	0.62	0.63	0.63	0.64	0.65	0.66	0.66	0.68
	mRMR	0.59	0.70	0.77	0.82	0.85	0.87	0.89	0.91	0.92	0.93	0.93	0.94	0.94	0.95	0.95	0.96
	Kruskal	0.58	0.63	0.66	0.70	0.72	0.75	0.77	0.80	0.81	0.83	0.84	0.85	0.85	0.86	0.86	0.89
	mPOS	0.58	0.69	0.76	0.79	0.83	0.85	0.87	0.88	0.89	0.90	0.91	0.92	0.92	0.93	0.93	0.95
	full set	0.95															
XGBoost	LASSO	0.49	0.49	0.49	0.5	0.5	0.51	0.51	0.52	0.54	0.55	0.56	0.56	0.57	0.58	0.59	0.62
	mRMR	0.48	0.63	0.72	0.77	0.8	0.82	0.84	0.85	0.86	0.87	0.87	0.87	0.87	0.88	0.88	0.89
	Kruskal	0.49	0.56	0.61	0.67	0.69	0.71	0.72	0.76	0.77	0.78	0.8	0.8	0.81	0.82	0.82	0.84
	mPOS	0.46	0.62	0.7	0.75	0.78	0.81	0.82	0.83	0.84	0.85	0.85	0.86	0.86	0.87	0.87	0.88
	full set	0.89															

Table 8.64: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 11** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.37	0.38	0.40	0.41	0.43	0.44	0.46	0.46	0.48	0.48	0.48	0.50	0.51	0.51	0.52	0.55
	mRMR	0.38	0.52	0.60	0.65	0.70	0.72	0.74	0.76	0.78	0.80	0.81	0.82	0.83	0.84	0.84	0.87
	Kruskal	0.39	0.45	0.50	0.55	0.57	0.60	0.62	0.66	0.67	0.69	0.70	0.72	0.72	0.73	0.74	0.78
	mPOS	0.37	0.50	0.58	0.64	0.67	0.70	0.72	0.74	0.76	0.78	0.79	0.79	0.80	0.81	0.82	0.84
	full set	0.91															
KNN	LASSO	0.45	0.46	0.47	0.48	0.50	0.51	0.51	0.52	0.53	0.53	0.54	0.54	0.55	0.55	0.56	0.58
	mRMR	0.46	0.56	0.63	0.67	0.71	0.73	0.75	0.76	0.78	0.79	0.80	0.81	0.81	0.82	0.82	0.83
	Kruskal	0.45	0.52	0.53	0.56	0.59	0.61	0.63	0.66	0.67	0.67	0.68	0.69	0.70	0.71	0.72	0.76
	mPOS	0.46	0.55	0.61	0.65	0.68	0.70	0.72	0.74	0.75	0.76	0.77	0.78	0.79	0.79	0.8	0.81
	full set	0.75															
SVM	LASSO	0.46	0.47	0.48	0.5	0.51	0.52	0.53	0.53	0.54	0.54	0.55	0.56	0.56	0.57	0.57	0.59
	mRMR	0.48	0.57	0.64	0.68	0.71	0.74	0.77	0.78	0.80	0.82	0.83	0.84	0.85	0.86	0.86	0.90
	Kruskal	0.46	0.51	0.54	0.56	0.58	0.61	0.62	0.66	0.67	0.68	0.69	0.71	0.72	0.73	0.74	0.78
	mPOS	0.47	0.55	0.61	0.65	0.69	0.71	0.74	0.76	0.78	0.79	0.8	0.81	0.82	0.82	0.83	0.86
	full set	0.85															
XGBoost	LASSO	0.37	0.38	0.39	0.41	0.42	0.43	0.44	0.45	0.46	0.46	0.47	0.48	0.49	0.49	0.50	0.52
	mRMR	0.38	0.51	0.58	0.63	0.66	0.69	0.71	0.73	0.74	0.75	0.76	0.76	0.77	0.77	0.77	0.79
	Kruskal	0.39	0.44	0.49	0.53	0.55	0.58	0.59	0.63	0.64	0.65	0.67	0.68	0.68	0.69	0.69	0.73
	mPOS	0.37	0.48	0.56	0.61	0.64	0.67	0.69	0.71	0.72	0.73	0.74	0.75	0.76	0.76	0.76	0.78
	full set	0.79															

Table 8.65: Average classification accuracy of Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting on **Scenario 12** for five-class classification problems, computed across 10 repetitions of 5-fold cross-validation.

		Number of genes															
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20
RF	LASSO	0.46	0.47	0.48	0.50	0.51	0.52	0.53	0.53	0.54	0.54	0.55	0.56	0.56	0.57	0.57	0.59
	mRMR	0.29	0.37	0.44	0.49	0.51	0.54	0.56	0.58	0.59	0.60	0.62	0.63	0.64	0.64	0.65	0.68
	Kruskal	0.27	0.37	0.37	0.41	0.43	0.45	0.47	0.48	0.50	0.50	0.52	0.53	0.53	0.54	0.54	0.58
	mPOS	0.30	0.37	0.43	0.47	0.50	0.52	0.54	0.55	0.57	0.58	0.59	0.60	0.61	0.62	0.63	0.66
	full set	0.72															
KNN	LASSO	0.36	0.36	0.37	0.37	0.38	0.39	0.4	0.4	0.4	0.41	0.42	0.42	0.43	0.44	0.44	0.46
	mRMR	0.36	0.42	0.46	0.51	0.53	0.55	0.56	0.58	0.59	0.6	0.61	0.61	0.61	0.62	0.63	0.63
	Kruskal	0.34	0.39	0.41	0.43	0.44	0.45	0.46	0.48	0.48	0.5	0.5	0.52	0.53	0.52	0.53	0.55
	mPOS	0.38	0.43	0.46	0.49	0.51	0.52	0.53	0.54	0.55	0.56	0.57	0.57	0.58	0.58	0.58	0.6
	full set	0.57															
SVM	LASSO	0.37	0.37	0.38	0.39	0.40	0.40	0.41	0.41	0.42	0.42	0.43	0.43	0.43	0.44	0.44	0.46
	mRMR	0.37	0.42	0.47	0.51	0.53	0.56	0.58	0.60	0.62	0.63	0.64	0.66	0.66	0.67	0.68	0.71
	Kruskal	0.35	0.38	0.41	0.42	0.42	0.45	0.46	0.48	0.48	0.50	0.51	0.52	0.54	0.54	0.55	0.59
	mPOS	0.38	0.42	0.46	0.49	0.51	0.53	0.55	0.57	0.58	0.60	0.61	0.61	0.62	0.63	0.64	0.67
	full set	0.63															
XGBoost	LASSO	0.28	0.29	0.29	0.30	0.31	0.32	0.32	0.33	0.34	0.35	0.35	0.36	0.36	0.37	0.37	0.40
	mRMR	0.29	0.36	0.41	0.46	0.49	0.51	0.53	0.56	0.56	0.57	0.58	0.59	0.60	0.60	0.61	0.62
	Kruskal	0.27	0.33	0.33	0.35	0.38	0.41	0.43	0.45	0.46	0.47	0.49	0.49	0.51	0.50	0.52	0.56
	mPOS	0.30	0.35	0.40	0.44	0.46	0.49	0.51	0.52	0.54	0.55	0.56	0.57	0.57	0.57	0.59	0.60
	full set	0.62															

8.4 Summary

This chapter focuses on simulation studies designed to enhance the understanding of the properties of 3cPOS and mPOS methods under conditions of balanced class distributions under uncorrelated and correlated structures. Four experiments were conducted using two simulation models; Simulation model 1 and 2, to generate data distributions. The experiments 1 and 3 were specifically aimed at investigating the impact of noise in input features, while the experiments 2 and 4 were conducted to examine the effects of increased variance differences among classes. For each experiment, three scenarios are considered, in which the data distributions are configured with varying degrees of overlap. Therefore, the performance of the 3cPOS and mPOS methods is evaluated across 12 scenarios for each classification, with a total of 12 and 48 scenarios considered, respectively. These scenarios are used to thoroughly assess the effectiveness of both methods in various classification settings.

For evaluation of 3cPOS and mPOS methods, simulated datasets are performed with 10 iterations of 5-fold cross-validation, resulting in 50 runs. For each run of binary classification, features are selected up to 20, $r = 1, 2, 3, \ldots, 20$, using LASSO, mRMR, Wilcoxon, and mPOS methods. Unlike binary classification, LASSO, mRMR, Kruskal, 3cPOS and mPOS methods are applied to datasets with multi-class classification in order to select top 20 informative features. The various sets of selected informative features are used to construct classification models including the Random Forest (RF), k-Nearest Neighbors (KNN), Support Vector Machine (SVM), and XGBoost classifiers, providing the average classification accuracy.

The 3cPOS method outperforms other feature selection techniques in most classifiers, especially with small and moderate sets of informative features. It achieves 100% accuracy with the KNN classifier and outperforms others in small feature sets for SVM and RF classifiers. While its performance is generally superior in Scenarios 1 to 7, it is comparable to other methods in Scenarios 9 and 10 when applied to XGBoost and KNN classifiers.

The mPOS method demonstrates comparable performance to other feature selection techniques across multiple classifiers in binary classification problems, achieving high accuracy at both small and large set sizes of informative features. It excels in several scenarios, notably

performing well with larger feature sets, though in some cases, methods like mRMR or Wilcoxon perform better at smaller feature sets.

The mPOS method demonstrates superior performance in various classifiers in three-classification scenarios, especially in smaller sets of informative features. As the size of the informative feature set increases, mPOS performs comparably to other feature selection techniques like LASSO and mRMR.

The mPOS method consistently demonstrates comparable performance to alternative feature selection techniques across different set sizes of informative features for the RF, kNN, SVM, and XGBoost classifiers in four-class classification problems. In scenarios with a smaller number of informative features, mPOS performs comparable performance to LASSO and mRMR, while in larger sets of informative features, mPOS remains competitive with these methods.

The evaluation of the mPOS method for five-class classification problems demonstrates its comparable performance to other feature selection techniques, particularly when selecting a single informative feature. Furthermore, both mPOS and mRMR show superior performance at larger set sizes of informative features in most scenarios.

Overall, both 3cPOS and mPOS demonstrate robust and effective performance as feature selection methods when evaluated on simulated datasets. These datasets were generated under conditions characterised by a balanced class distribution, with a focus on assessing the impact of noise in input features, increased variance differences among classes, and varying degrees of class overlap. The methods consistently show strong performance, highlighting their reliability and adaptability in various classification scenarios. These findings highlight the potential of 3cPOS and mPOS as reliable tools for feature selection in diverse, challenging data environments.

CHAPTER 9

Conclusions and Future Plans

9.1 Conclusions

Functional genomics experiments, such as gene expression microarrays, play a crucial role in identifying phenotypes and their links to various biological processes. A primary objective of utilising gene expression data is to detect multiple stages of diseases or to identify differentially expressed genes that can accurately predict the phenotypes of new samples. However, gene expression datasets consists of measurements for tens of thousands of genes across only a limited number of samples, leading to a high-dimensional feature space. This may result in poor model performance, model overfitting, or challenges in result interpretation. To address these issues, feature selection techniques are considered to reduce the dimensionality of the feature space, enhance computational efficiency, and improve the predictive accuracy of machine learning models.

Feature selection is employed to identify the most relevant features while eliminating redundant or irrelevant ones, offering a set of informative features. Evaluating the overlap between gene expressions of different classes has become a key criterion for assessing the discriminative capability of genes in classification tasks. The main idea is that gene i has the ability to correctly classify samples whose ith expressions lie within a region (interval) of a single class, i.e. not overlapping with ith expressions of other classes. This approach yields overlap scores, which

quantify the discriminatory power of a given gene *i*. In this thesis, we exploited this approach to propose an extension to the POS method [123] to work beyond binary class problems. Our approach incorporates additional considerations, including the length of overlapping regions, the number of overlapping samples, and the proportion of each class' contribution to the overlapped samples, to generate overlapping scores.

Due to the balance between the simplicity of binary classification and the complexity of a larger multiclass scenario, a novel feature selection technique, called the 3-class Proportional Overlapping Score (3cPOS), is proposed in Chapter 5. This method offers an understanding of class interactions, facilitating the development of sophisticated models for improved prediction accuracy and robustness, making three-class problems ideal benchmarks for testing algorithms and features. Firstly, the core expression intervals are assigned for each gene *i* and class to mitigate the effect of outliers. The overlap between the expression intervals of the gene for various classes are then considered to reflect the gene's discriminative characteristics. 3cPOS scores are derived to assess the capability of a gene to distinguish the correct target classes. Genes with smaller 3cPOS scores indicate higher discriminative power. To evaluate the performance of 3cPOS, we compare the results of the most informative features selected by 3cPOS with three other established feature selection techniques (LASSO, mRMR and Kruskal-Wallis Test), as well as with with the full set of features, across seven benchmark datasets. These selected features were used to construct several classification models including Random Forest, k-Nearest Neighbour, Support Vector Machine, and Extreme Gradient Boost. The average classification accuracies and stability of selected features are then compared across 20 repetitions of 5-fold cross validation, resulting 100 runs. Our experimental results demonstrate that our proposed method, 3cPOS, achieves superior performance in terms of classification accuracy and stability compared to alternative feature selection techniques. Furthermore, 3cPOS demonstrates greater adaptability to different data patterns and classifier types, which makes it a unique and effective feature selection method.

3cPOS achieves superior performance across seven gene expression datasets comparable to other feature selection techniques. However, considering a minimum subset of genes could help boost predictive accuracy and mitigate the effects of redundant information and imbalanced class

sizes. Standardised expressions and its core expressions are exploited to determine gene masks, representing the discriminative power of gene *i*. The minimum subset of genes is defined based on 3cPOS scores and gene masks where this subset refers to the minimum set of gene that can detect their correct target classes from the training phase. The Relative Dominant Class (RDC) is then considered to mitigate misleading assignments resulting from imbalanced class sizes by identifying the dominant class of each gene *i* based on its relative roles. Class with highest proportion indicates the RDC of gene *i*. To provide the final gene selection, the minimum subset of gene and gene ranking are incorporated. Genes that are not included in the minimum subset of gene are then considered for gene ranking. We proposed two distinct ideas to determine gene ranking: Idea 1 and Idea 2. For Idea 1, the remaining genes are categorised by RDC and sorted in ascending order according to their 3cPOS scores within each category of RDC. In contrast, Idea 2 focuses on directly sorting the remaining genes in ascending order based solely on their 3cPOS scores. To assess these approaches, we aimed at comparing the classification accuracy of Idea 1, Idea 2, and 3cPOS on seven gene expression datasets. The result demonstrates that Idea 1 and Idea 2 achieve a comparable performance to 3cPOS across RF, KNN, SVM, and XGBoost Classifiers.

Applying 3cPOS have improved significantly predictive power as well as interpretation. However, 3cPOS is only designed for three class problems. To address this restriction, we proposed a novel feature selection techniques, named multiple Proportional Overlapping Scores (mPOS), that can be implemented for multiple class problems. Initially, core intervals in gene expressions are determined to alleviate the influence of outliers prior to analysing overlapping between intervals. mPOS scores are generated to assess the ability of a gene to classify the correct target class. Genes with lower mPOS scores offer higher discriminative capability. Our experiments was performed using 20 iterations with 5-fold cross validation to evaluate the performance of feature selection methods. We compared the proposed mPOS approach with established techniques including LASSO, mRMR, Wilcoxon, and Kruskal, as well as using the full feature set. The sets of selected features from those techniques were used to construct classification models, including Random Forest (RF), k-Nearest Neighbours, Support Vector Machine, and Extreme Gradient Boosting. The average classification accuracy was computed

over 100 runs to ensure robustness and reliability of the results. Our experiments reveal that our proposed method, mPOS, is better than, or comparable to, four representative competing feature selection algorithms in terms of classification accuracy and stability. Furthermore, mPOS can be implemented with an unlimited number of genes, even when working with small sample sizes, thereby positioning mPOS as a feature selection technique without inherent limitations

The evaulation of the 3cPOS and mPOS methods is undertaken using real-world gene expression datasets. Most datasets are characterised by imbalanced class distributions. To get insight into the abilities of 3cPOS and mPOS methods under various setups, simulation studies are considered. We exploited simulation studies to generate data with balanced class distribution and varying degrees of overlaps between classes. Simulation models 1 and 2 are exploited to generate informative features using a multivariate normal distribution across four experiments. In addition, non-informative features are generated based on a standard normal distribution for each experiment to test a model's robustness with respect to noisy input features. For each experiment, three distinct scenarios are designed to simulate varying the degree of overlaps. These overlaps offer different levels of difficulty for feature selection methods, providing a comprehensive evaluation of the abilities of 3cPOS and mPOS method to deal with complex classification tasks. Therefore, in each case of classification tasks, data is generated based on 4 experiments across 3 distinct scenarios, resulting in a total of 12 scenarios. All scenarios are utilised to assess the performance of 3cPOS and mPOS in comparison with LASSO, mRMR, Wilcoxon, and Kruskal methods across four classifiers: Random Forest, k-Nearest Neighbors, Support Vector Machine, and Extreme Gradient Boosting. Regardless of maintaining or varying the variance-covariance, the 3cPOS and mPOS methods consistently show strong performance in most classifiers, highlighting their reliability and adaptability in various classification scenarios.

9.2 Future plans

Several ideas are briefly discussed here and are intended to provide directions for future research.

1. Constructing a framework for mPOS in which mutual information between genes are considered in the final gene set might be another useful direction. Such a framework could

be effective in selecting the discriminative genes with a lower degree of dependency.

2. The entirety of this thesis focuses on feature selection methods in the context of functional genomic experiments and their role in improving the performance of statistical learning models. Applying the proposal on datasets from different domains as well as different kinds of features is also another direction.

Bibliography

[1] Anthony J Alberg and Jonathan M Samet. “Epidemiology of lung cancer”. In: *Chest* 123.1 (2003), 21S–49S.

[2] Pia Alhopuro et al. “Candidate driver genes in microsatellite-unstable colorectal cancer”. In: *International Journal of Cancer* 130.7 (2012), pp. 1558–1566.

[3] Peshawa Jamal Muhammad Ali et al. “Data normalization and standardization: a technical report”. In: *Mach Learn Tech Rep* 1.1 (2014), pp. 1–6.

[4] Anshul. *Guide on Support Vector Machine (SVM) Algorithm*. Last accessed 30 December 2024. (2024). URL: https://www.analyticsvidhya.com/blog/2021/10/support-vector-machinessvm-a-complete-guide-for-beginners/.

[5] Anuj. *World Ovarian Cancer day : Symptoms, Treatment , Diagnosis All You Want To Know*. Last accessed 30 October 2024. (2023). URL: https://www.stackumbrella.com/world-ovarian-cancer-day-all-symptoms/.

[6] Daniele Apiletti et al. “Maskedpainter: feature selection for microarray data analysis”. In: *Intelligent Data Analysis* 16.4 (2012), pp. 717–737.

[7] Daniele Apiletti et al. “The painter’s feature selection for gene expression data”. In: *29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*. IEEE. (2007), pp. 4227–4230.

[8] Scott A Armstrong et al. “MLL translocations specify a distinct gene expression profile that distinguishes a unique leukemia”. In: *Nature Genetics* 30.1 (2002), pp. 41–47. DOI: https://doi.org/10.1038/ng765.

[9] Child Age Average. *Logistic Regression*. (2019).

[10] Hasseeb Azzawi et al. “Lung cancer prediction from microarray data by gene expression programming”. In: *IET Systems Biology* 10.5 (2016), pp. 168–178.

[11] Julia Bachman. “Reverse-transcription PCR (rt-PCR)”. In: *Methods in Enzymology*. Vol. 530. Elsevier, (2013), pp. 67–74.

[12] Elena Baralis, Giulia Bruno, and Alessandro Fiori. “Measuring gene similarity by means of the classification distance”. In: *Knowledge and Information Systems* 29 (2011), pp. 81–101.

[13] Joshua A Bauer et al. “Identification of markers of taxane sensitivity using proteomic and genomic analyses of breast tumors from patients receiving neoadjuvant paclitaxel and radiation”. In: *Clinical Cancer Research* 16.2 (2010), pp. 681–690.

[14] John R Benson. “The TNM staging system and breast cancer”. In: *The Lancet Oncology* 4.1 (2003), pp. 56–60.

[15] Viv Bewick, Liz Cheek, and Jonathan Ball. “Statistics review 14: Logistic regression”. In: *Critical Care* 9.1 (2005), pp. 1–7.

[16] Arindam Bhattacharjee et al. “Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses”. In: *Proceedings of the National Academy of Sciences* 98.24 (2001), pp. 13790–13795.

[17] Hervé Bonnefoi et al. “RETRACTED: Validation of gene signatures that predict the response of breast cancer to neoadjuvant chemotherapy: a substudy of the EORTC 10994/BIG 00-01 clinical trial”. In: *The Lancet Oncology* 8.12 (2007), pp. 1071–1078.

[18] Tomas Bonome et al. “A gene signature predicting for survival in suboptimally debulked patients with ovarian cancer”. In: *Cancer Research* 68.13 (2008), pp. 5478–5486.

[19] David G Bostwick, Robert P Myers, and Joseph E Oesterling. “Staging of prostate cancer”. In: *Seminars in surgical oncology*. Vol. 10. 1. Wiley Online Library. (1994), pp. 60–72.

[20] Anne-Laure Boulesteix et al. “Introduction to statistical simulations in health research”. In: *BMJ open* 10.12 (2020), e039921.

[21] Freddie Bray et al. “The ever-increasing importance of cancer as a leading cause of premature death worldwide”. In: *Cancer* 127.16 (2021), pp. 3029–3030.

[22] Leo Breiman. “Random forests”. In: *Machine Learning* 45.1 (2001), pp. 5–32.

[23] Peter Bühlmann and Sara Van De Geer. *Statistics for high-dimensional data: methods, theory and applications*. Springer Science & Business Media, (2011).

[24] William Burke et al. “Executive summary of the ovarian cancer evidence review conference”. In: *Obstetrics & Gynecology* 142.1 (2023), pp. 179–195.

[25] Jie Cai et al. “Feature selection in machine learning: A new perspective”. In: *Neurocomputing* 300 (2018), pp. 70–79.

[26] Melissa M Center et al. “Worldwide variations in colorectal cancer”. In: *CA: a Cancer Journal for Clinicians* 59.6 (2009), pp. 366–378.

[27] Nigel Chaffey. *Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K. and Walter, P. Molecular biology of the cell. 4th edn.* (2003).

[28] Bertram KC Chan and Bertram KC Chan. “Data analysis using R programming”. In: *Biostatistics for Human Genetic Epidemiology* (2018), pp. 47–122.

[29] B Chandra and Manish Gupta. “An efficient statistical feature selection approach for classification of gene expression data”. In: *Journal of Biomedical Informatics* 44.4 (2011), pp. 529–535.

[30] Dung-Tsa Chen et al. “Proliferative genes dominate malignancy-risk gene signature in histologically-normal breast tissue”. In: *Breast Cancer Research and Treatment* 119 (2010), pp. 335–346.

[31] Gang Chen and Jin Chen. “A novel wrapper method for feature selection and its applications”. In: *Neurocomputing* 159 (2015), pp. 219–226.

[32] Tianqi Chen and Carlos Guestrin. “Xgboost: A scalable tree boosting system”. In: *Proceedings of the 22nd acm sigkdd International Conference on Knowledge Discovery and Data Mining*. (2016), pp. 785–794.

[33] Tianqi Chen et al. “Xgboost: extreme gradient boosting”. In: *R package version 0.4-2* 1.4 (2015), pp. 1–4.

[34] Yen-Jen Chen et al. “Molecular subtyping of breast cancer intrinsic taxonomy with oligonucleotide microarray and NanoString nCounter”. In: *Bioscience Reports* 41.8 (2021), BSR20211428.

[35] Yuan Chen et al. “Informative gene selection and the direct classification of tumors based on relative simplicity”. In: *BMC Bioinformatics* 17 (2016), pp. 1–16.

[36] Adithya Chennamadhavuni et al. “Continuing education activity”. In: *National Library of Medicine* (2021), p. 2.

[37] Ian Chivers and Jane Sleightholme. “An introduction to Algorithms and the Big O Notation”. In: *Introduction to Programming with Fortran: With Coverage of Fortran 90, 95, 2003, 2008 and 77*. Springer, 2015, pp. 359–364.

[38] Norimichi Chiyonobu et al. “Fatty acid binding protein 4 (FABP4) overexpression in intratumoral hepatic stellate cells within hepatocellular carcinoma with metabolic risk factors”. In: *The American Journal of Pathology* 188.5 (2018), pp. 1213–1224.

[39] Carlton Chu et al. “Does feature selection improve classification accuracy? Impact of sample size and feature selection on classification using anatomical magnetic resonance images”. In: *Neuroimage* 60.1 (2012), pp. 59–70.

[40] Emily Clough and Tanya Barrett. “The gene expression omnibus database”. In: *Statistical Genomics: Methods and Protocols* (2016), pp. 93–110.

[41] Huntly Collins. *New Jersey Association for Biomedical Research. The Man Who Saved Your Life-Maurice R. Hilleman-Developer of Vaccines for Mumps and Pandemic Flu. Maurice Hilleman’s Vaccines Prevent Millions of Deaths Every Year.*

[42] Thomas A Cooper, Lili Wan, and Gideon Dreyfuss. “RNA and disease”. In: *Cell* 136.4 (2009), pp. 777–793.

[43] Mariarosaria D’Errico et al. “Genome-wide expression profile of sporadic gastric cancers with microsatellite instability”. In: *European Journal of Cancer* 45.3 (2009), pp. 461–469.

[44] Dennise D Dalma-Weiszhausz et al. “The Affymetrix GeneChip® Platform: An Overview”. In: *Methods in Enzymology* 410 (2006), pp. 3–28.

[45] Nicolas De Jay et al. “Package ‘mRMRe’”. In: *CRAN R Repositary* (2020).

[46] Lin Deng et al. “A rank sum test method for informative gene discovery”. In: *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (2004), pp. 410–419.

[47] Ramón Díaz-Uriarte and Sara Alvarez de Andrés. “Gene selection and classification of microarray data using random forest”. In: *BMC Bioinformatics* 7 (2006), pp. 1–13.

[48] Evgenia Dimitriadou et al. “The e1071 package”. In: *Misc Functions of Department of Statistics (e1071), TU Wien* (2006), pp. 297–304.

[49] Chris Ding and Hanchuan Peng. “Minimum redundancy feature selection from microarray gene expression data”. In: *Journal of Bioinformatics and Computational Biology* 3.02 (2005), pp. 185–205.

[50] Dreamstime. *Colon cancer*. Last accessed 30 October 2024. (2020). URL: https://www.dreamstime.com/colon-cancer-colorectal-oncology-stages-development-malignant-tumor-colon-cancer-colorectal-oncology-development-image176306034.

[51] Aditya Duneja and Thendral Puyalnithi. “Enhancing classification accuracy of k-nearest neighbours algorithm using gain ratio”. In: *Int. Res. J. Eng. Technol* 4.9 (2017), pp. 1385–1388.

[52] Anwesha Dutta. *Adding automated Statistical Analysis and Biological Evaluation modules to www. arrayanalysis. org*. 2011.

[53] Liat Ein-Dor et al. “Outcome signature genes in breast cancer: is there a unique set?” In: *Bioinformatics* 21.2 (2005), pp. 171–178.

[54] Encyclopedia. *Microarray*. Last accessed 8 January 2024. (2023). URL: https://www.pcmag.com/encyclopedia/term/micro-array.

[55] Xinyan Fan et al. “Assisted graphical model for gene expression data analysis”. In: *Statistics in Medicine* 38.13 (2019), pp. 2364–2380.

[56] Hongqing Fang, Pei Tang, and Hao Si. “Feature selections using minimal redundancy maximal relevance algorithm for human activity recognition in smart home environments”. In: *Journal of Healthcare Engineering* 2020 (2020).

[57] Stefano Ferilli et al. “K-nearest neighbor classification on first-order logic descriptions”. In: *2008 IEEE International Conference on Data Mining Workshops*. IEEE. (2008), pp. 202–210.

[58] Valeria Fonti and Eduard Belitser. “Feature selection using lasso”. In: *VU Amsterdam Research Paper in Business Analytics* 30 (2017), pp. 1–25.

[59] Ronald N Forthofer, Eun Sul Lee, and Mike Hernandez. *Biostatistics: a guide to design, analysis and discovery*. Elsevier, (2006).

[60] Steven A Frank. “Genetic predisposition to cancer—insights from population genetics”. In: *Nature Reviews Genetics* 5.10 (2004), pp. 764–772.

[61] Jerome Friedman et al. “Package ‘glmnet’”. In: *CRAN R Repositary* (2021).

[62] Vincent G. *Nucleic acids*. Last accessed 4 January 2024. (2017). URL: https://socratic.org/questions/593c9eba7c0149793d298820#438045.

[63] VV Galatenko et al. “Highly informative marker sets consisting of genes with low individual degree of differential expression”. In: *Scientific Reports* 5.1 (2015), p. 14967.

[64] Laurent Gautier et al. “affy—analysis of Affymetrix GeneChip data at the probe level”. In: *Bioinformatics* 20.3 (2004), pp. 307–315.

[65] Geeksforgeeks. *Difference Between Purines and Pyrimidines*. Last accessed 24 January 2024. (2024). URL: https://www.geeksforgeeks.org/difference-between-purines-and-pyrimidines/.

[66] International Society of Genetic Genealogy Wiki. *Affymetrix*. Last accessed 20 May 2022. (2012). URL: https://isogg.org/wiki/Affymetrix.

[67] STAT 555 Statistical Analysis of Genomics Data. *High Density (Affymetrix@) Microarrays and their Normalization*. Last accessed 7 January 2024. (2023). URL: https://online.stat.psu.edu/stat555/node/49/.

[68] Christopher E Gillies et al. “A simulation to analyze feature selection methods utilizing gene ontology for gene expression classification”. In: *Journal of Biomedical Informatics* 46.6 (2013), pp. 1044–1059.

[69] Federico M Giorgi et al. “Algorithm-driven artifacts in median polish summarization of microarray data”. In: *BMC Bioinformatics* 11 (2010), pp. 1–12.

[70] Glossary. *RNA*. Last accessed 22 November 2024. (2017). URL: https://rosalind.info/glossary/rna/.

[71] Todd R Golub et al. “Molecular classification of cancer: class discovery and class prediction by gene expression monitoring”. In: *Science* 286.5439 (1999), pp. 531–537.

[72] Asma Gul et al. “Ensemble of a subset of k NN classifiers”. In: *Advances in Data Analysis and Classification* 12 (2018), pp. 827–840.

[73] Isabelle Guyon and André Elisseeff. “An introduction to variable and feature selection”. In: *Journal of Machine Learning Research* 3.Mar (2003), pp. 1157–1182.

[74] Muhammad Hamraz et al. “Robust proportional overlapping analysis for feature selection in binary classification within functional genomic experiments”. In: *PeerJ Computer Science* 7 (2021), e562.

[75] Ahmad Basheer Hassanat et al. “Solving the problem of the K parameter in the KNN classifier using an ensemble learning approach”. In: *arXiv preprint arXiv:1409.0919* (2014).

[76] Trevor Hastie, Brad Efron, and Maintainer Trevor Hastie. “Package ‘lars’”. In: *CRAN R Repositary* (2022).

[77] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. “Overview of supervised learning”. In: *The elements of statistical learning*. Springer, (2009), pp. 9–41.

[78] Haibo He and Edwardo A Garcia. “Learning from imbalanced data”. In: *IEEE Transactions on knowledge and data engineering* 21.9 (2009), pp. 1263–1284.

[79] Marti A. Hearst et al. “Support vector machines”. In: *IEEE Intelligent Systems and Their Applications* 13.4 (1998), pp. 18–28.

[80] Henderi Henderi, Tri Wahyuningsih, and Efana Rahwanto. "Comparison of Min-Max normalization and Z-Score Normalization in the K-nearest neighbor (kNN) Algorithm to Test the Accuracy of Types of Breast Cancer". In: *International Journal of Informatics and Information Systems* 4.1 (2021), pp. 13–20.

[81] Yair Herishanu et al. "The lymph node microenvironment promotes B-cell receptor signaling, NF-κB activation, and tumor proliferation in chronic lymphocytic leukemia". In: *Blood, The Journal of the American Society of Hematology* 117.2 (2011), pp. 563–574. DOI: 10.1182/blood-2010-05-284984.

[82] Joseph M Hilbe. *Practical guide to logistic regression*. CRC Press, (2016).

[83] Myles Hollander, Douglas A Wolfe, and Eric Chicken. *Nonparametric statistical methods*. John Wiley & Sons, (2013).

[84] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*. Vol. 398. John Wiley & Sons, (2013).

[85] Torsten Hothorn et al. *Coin: a computational framework for conditional inference*. 2013.

[86] Jianping Hua, Waibhav D Tembe, and Edward R Dougherty. "Performance of feature-selection methods in the classification of high-dimension data". In: *Pattern Recognition* 42.3 (2009), pp. 409–424.

[87] Cai Huang et al. "Machine learning predicts individual cancer patient responses to therapeutic drugs with high accuracy". In: *Scientific Reports* 8.1 (2018), p. 16444.

[88] Kyu-Baek Hwang et al. "Applying machine learning techniques to analysis of gene expression data: cancer diagnosis". In: *Methods of Microarray Data Analysis: Papers from CAMDA'00* (2002), pp. 167–182.

[89] Muhammad Ali Imron and Budi Prasetyo. "Improving algorithm accuracy k-nearest neighbor using z-score normalization and particle swarm optimization to predict customer churn". In: *Journal of Soft Computing Exploration* 1.1 (2020), pp. 56–62.

[90] Rafael A Irizarry, Laurent Gautier, et al. "Package 'affy'". In: *CRAN R Repositary* (2013).

[91] Rafael A Irizarry et al. “Exploration, normalization, and summaries of high density oligonucleotide array probe level data”. In: *Biostatistics* 4.2 (2003), pp. 249–264.

[92] Takayuki Iwamoto et al. “Gene pathways associated with prognosis and chemotherapy sensitivity in molecular subtypes of breast cancer”. In: *Journal of the National Cancer Institute* 103.3 (2011), pp. 264–272. DOI: 10.1093/jnci/djq524.

[93] Gareth James et al. *An introduction to statistical learning*. Vol. 112. Springer, (2013).

[94] Andreas Janecek et al. “On the relationship between feature selection and classification accuracy”. In: *New challenges for feature selection in data mining and knowledge discovery*. PMLR. 2008, pp. 90–105.

[95] Jaemin Jeon et al. “Denoiseit: denoising gene expression data using rank based isolation trees”. In: *BMC Bioinformatics* 25.1 (2024), p. 271.

[96] Zhenyu Jia et al. “Diagnosis of prostate cancer using differentially expressed genes in stroma”. In: *Cancer Research* 71.7 (2011), pp. 2476–2487.

[97] SAHU K. *Microarray technology, biochip, DNA chip*. Last accessed 8 January 2024. (2020). URL: https://www.slideshare.net/slideshow/microarray-technology-biochip-dna-chip/233508467.

[98] Alexandros Kalousis, Julien Prados, and Melanie Hilario. “Stability of feature selection algorithms: a study on high-dimensional spaces”. In: *Knowledge and information systems* 12.1 (2007), pp. 95–116.

[99] Esra Mahsereci Karabulut, Selma Ayşe Özel, and Turgay Ibrikci. “A comparative study on the effect of feature selection on classification accuracy”. In: *Procedia Technology* 1 (2012), pp. 323–327.

[100] Aman Kataria and MD Singh. “A review of data classification using k-nearest neighbour algorithm”. In: *International Journal of Emerging Technology and Advanced Engineering* 3.6 (2013), pp. 354–360.

[101] Timothy J Key, Pia K Verkasalo, and Emily Banks. “Epidemiology of breast cancer”. In: *The Lancet Oncology* 2.3 (2001), pp. 133–140.

[102] Javed Khan et al. “Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks”. In: *Nature Medicine* 7.6 (2001), pp. 673–679.

[103] Akifumi Kikuchi et al. “Identification of NUCKS1 as a colorectal cancer prognostic marker through integrated expression and copy number analysis”. In: *International Journal of Cancer* 132.10 (2013), pp. 2295–2302.

[104] Erin R King et al. “The anterior gradient homolog 3 (AGR3) gene is associated with differentiation and survival in ovarian cancer”. In: *The American Journal of Surgical Pathology* 35.6 (2011), pp. 904–912.

[105] William H Kruskal and W Allen Wallis. “Use of ranks in one-criterion variance analysis”. In: *Journal of the American statistical Association* 47.260 (1952), pp. 583–621.

[106] BIOINFORMATICS LABORATORY. *Data set name: leukemia*. Last accessed 7 July 2023. (1999). URL: https://file.biolab.si/biolab/supp/bi-cancer/projections/info/leukemia.html.

[107] Päivi Laiho et al. “Serrated carcinomas form a subclass of colorectal cancer with distinct molecular basis”. In: *Oncogene* 26.2 (2007), pp. 312–320.

[108] Thomas Navin Lal et al. “Embedded methods”. In: *Feature extraction: Foundations and applications*. Springer, (2006), pp. 137–165.

[109] Ludwig Lausser et al. “Measuring and visualizing the stability of biomarker selection techniques”. In: *Computational Statistics* 28 (2013), pp. 51–65. DOI: 10.1007/s00180-011-0284-y.

[110] Tae-Hwy Lee, Aman Ullah, and Ran Wang. “Bootstrap aggregating and random forest”. In: *Macroeconomic forecasting in the era of big data*. Springer, (2020), pp. 389–429.

[111] Alan C Leonard and Marcel Méchali. “DNA replication origins”. In: *Cold Spring Harbor Perspectives in Biology* 5.10 (2013), a010116.

[112] Tao Li, Chengliang Zhang, and Mitsunori Ogihara. “A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression”. In: *Bioinformatics* 20.15 (2004), pp. 2429–2437.

[113] Yang Li et al. “Amplification of LAPTM4B and YWHAZ contributes to chemotherapy resistance and recurrence of breast cancer”. In: *Nature Medicine* 16.2 (2010), pp. 214–218.

[114] Andy Liaw and Matthew Wiener. “Package ‘randomforest’”. In: *University of California, Berkeley: Berkeley, CA, USA* (2018).

[115] Loukia N Lili et al. “Molecular profiling predicts the existence of two functionally distinct classes of ovarian cancer stroma”. In: *BioMed Research International* 2013.1 (2013), p. 846387.

[116] JS Lilleyman et al. “French American British (FAB) morphological classification of childhood lymphoblastic leukaemia and its clinical importance.” In: *Journal of Clinical Pathology* 39.9 (1986), pp. 998–1002.

[117] Hongfang Liu, Ionut Bebu, and Xin Li. “Microarray probes and probe sets”. In: *Frontiers in Bioscience (Elite Edition)* 2 (2010), p. 325.

[118] Huan Liu and Hiroshi Motoda. *Feature selection for knowledge discovery and data mining*. Vol. 454. Springer Science & Business Media, (2012).

[119] Huan Liu and Rudy Setiono. “Feature selection and classification–a probabilistic wrapper approach”. In: *Industrial and Engineering Applications of Artificial Intelligence and Expert Systems*. CRC Press, (1996), pp. 419–424.

[120] Binbin Lu et al. “The Minkowski approach for choosing the distance metric in geographically weighted regression”. In: *International Journal of Geographical Information Science* 30.2 (2016), pp. 351–368.

[121] HSLU Hochschule Luzern. *Logistic regression*. Last accessed 20 November 2022. URL: https://www.empirical-methods.hslu.ch/decisiontree/relationship/3145-2.

[122] McNulty M. *Gastric (Stomach) Cancer: Types, Symptoms, Diagnosis, Treatment - PMCC Denver Oncology*. Last accessed 30 October 2024. (2024). URL: https://www.mylungcancerteam.com/resources/inoperable-lung-cancer-what-to-expect.

[123] Osama Mahmoud et al. “A feature selection method for classification within functional genomics experiments based on the proportional overlapping score”. In: *BMC Bioinformatics* 15.1 (2014), pp. 1–20.

[124] Osama Mahmoud et al. “Minimizing redundancy among genes selected based on the overlapping analysis”. In: *Analysis of Large and Complex Data*. Springer. (2016), pp. 275–285.

[125] Sebastián Maldonado and Julio López. “Dealing with high-dimensional class-imbalanced datasets: Embedded feature selection for SVM classification”. In: *Applied Soft Computing* 67 (2018), pp. 94–105.

[126] Patrick E McKight and Julius Najab. “Kruskal-wallis test”. In: *The Corsini Encyclopedia of Psychology* (2010), pp. 1–1.

[127] Geoffrey J McLachlan, Kim-Anh Do, and Christophe Ambroise. “Analyzing microarray gene expression data”. In: *John Wiley & Sons* (2005).

[128] Preventive Medicine and Cancer Care. *Gastric (Stomach) Cancer: Types, Symptoms, Diagnosis, Treatment - PMCC Denver Oncology*. Last accessed 30 October 2024. (2020). URL: https://www.pmccdenver.com/explore-the-various-types-of-cancer/gastric-cancer-types-symptoms-diagnosis-treatment-denver.

[129] Michael. *Measures of distance between samples: Euclidean*. Last accessed 20 November 2022. (2022). URL: http://www.econ.upf.edu/~michael/stanford/maeb4.pdf.

[130] Khadijah A Mitchell et al. “Comparative transcriptome profiling reveals coding and noncoding RNA differences in NSCLC from African Americans and European Americans”. In: *Clinical Cancer Research* 23.23 (2017), pp. 7412–7425.

[131] Keiji Miura. “An introduction to maximum likelihood estimation and information geometry”. In: *Interdisciplinary Information Sciences* 17.3 (2011), pp. 155–174.

[132] Douglas C Montgomery, Elizabeth A Peck, and G Geoffrey Vining. *Introduction to linear regression analysis*. John Wiley & Sons, (2021).

[133] Dale P Mood and James R Morrow. *Introduction to statistics in human performance*. Routledge, (2017).

[134] Tim P Morris, Ian R White, and Michael J Crowther. "Using simulation studies to evaluate statistical methods". In: *Statistics in Medicine* 38.11 (2019), pp. 2074–2102.

[135] Shiva Kumar R Mukkamalla, Alejandro Recio-Boiles, and Hani M Babiker. "Gastric cancer". In: *National Library of Medicine* (2017).

[136] Christoph Müssel et al. "Multi-objective parameter selection for classifiers". In: *Journal of Statistical Software* 46 (2012), pp. 1–27.

[137] R Muthukrishnan and R Rohini. "LASSO: A feature selection technique in predictive modeling for machine learning". In: *IEEE International Conference on Advances in Computer Applications (ICACA)*. IEEE. (2016), pp. 18–20.

[138] Scitable by natureEDUCATION. *Microarray*. Last accessed 20 May 2022. (2014). URL: https://www.nature.com/scitable/definition/microarray-202/.

[139] Danh V Nguyen et al. "DNA microarray experiments: biological and technological aspects". In: *Biometrics* 58.4 (2002), pp. 701–717.

[140] Ivyna Bong Pau Ni et al. "Gene expression patterns distinguish breast carcinomas from normal breast tissues: the Malaysian context". In: *Pathology-Research and Practice* 206.4 (2010), pp. 223–228.

[141] Daniel A Notterman et al. "Transcriptional gene expression profiles of colorectal adenoma, adenocarcinoma, and normal tissue examined by oligonucleotide arrays". In: *Cancer Research* 61.7 (2001), pp. 3124–3130.

[142] Franco Odicino et al. "History of the FIGO cancer staging system". In: *International Journal of Gynecology & Obstetrics* 101.2 (2008), pp. 205–210.

[143] Hicham Omara, Mohamed Lazaar, and Youness Tabii. "Effect of feature selection on gene expression datasets classification accuracy". In: *International Journal of Electrical and Computer Engineering* 8.5 (2018), pp. 3194–3203.

[144] D Max Parkin et al. "Global cancer statistics, 2002". In: *CA: a Cancer Journal for Clinicians* 55.2 (2005), pp. 74–108.

[145] Iman Paryudi. “What Affects K Value Selection In K-Nearest Neighbor”. In: *International Journal Of Scientific & Technology Research* 8.07 (2019).

[146] Susmita Pathak. *Residual sum of squares*. Last accessed 20 November 2022. (2022). URL: https://www.wallstreetmojo.com/residual-sum-of-squares/.

[147] Topon Kumar Paul and Hitoshi Iba. “Extraction of informative genes from microarray data”. In: *Proceedings of the 7th Annual Conference on Genetic and Evolutionary Computation*. (2005), pp. 453–460.

[148] Hanchuan Peng, Fuhui Long, and Chris Ding. “Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy”. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.8 (2005), pp. 1226–1238.

[149] Free Learning Platform. *DNA Structure | Watson and Crick’s model of DNA*. Last accessed 5 January 2024. (2017). URL: https://www.javatpoint.com/dna-structure.

[150] Scott L Pomeroy et al. “Prediction of central nervous system embryonal tumour outcome based on gene expression”. In: *Nature* 415.6870 (2002), pp. 436–442.

[151] The Biology Projects and Molecular Biology. *Molecular Genetics of Prokaryotes Problem Set*. Last accessed 30 October 2024. (2023). URL: https://www3.med.unipmn.it/did/will/BiologyProject/09t-18.html.

[152] J. Ross Quinlan. “Learning decision tree classifiers”. In: *ACM Computing Surveys (CSUR)* 28.1 (1996), pp. 71–72.

[153] Kanti R Rai et al. “Clinical staging of chronic lymphocytic leukemia”. In: *Blood* (1975), pp. 219–234.

[154] Ramon Rami-Porta, John J Crowley, and Peter Goldstraw. “Review the revised TNM staging system for lung cancer”. In: *Ann Thorac Cardiovasc Surg* 15.1 (2009), p. 5.

[155] Sebastian Raschka. “Model evaluation, model selection, and algorithm selection in machine learning”. In: *arXiv preprint arXiv:1811.12808* (2018).

[156] Brian Ripley, William Venables, and Maintainer Brian Ripley. "Package 'class'". In: *The Comprehensive R Archive Network* 11 (2015).

[157] Érick O Rodrigues. "Combining Minkowski and Chebyshev: New distance proposal and survey of distance metrics using k-nearest neighbours classifier". In: *Pattern Recognition Letters* 110 (2018), pp. 66–71.

[158] Sophie Rousseaux et al. "Ectopic activation of germline and placental genes identifies aggressive metastasis-prone lung cancers". In: *Science Translational Medicine* 5.186 (2013), 186ra66–186ra66.

[159] Tamang S. *Gene Expression: Stages, Regulations, Methods*. Last accessed 5 January 2024. URL: https://microbenotes.com/gene-expression/.

[160] Yvan Saeys, Inaki Inza, and Pedro Larranaga. "A review of feature selection techniques in bioinformatics". In: *Bioinformatics* 23.19 (2007), pp. 2507–2517.

[161] Sujay Saha et al. "A novel gene ranking method using Wilcoxon rank sum test and genetic algorithm". In: *International Journal of Bioinformatics Research and Applications* 12.3 (2016), pp. 263–279.

[162] Claude Sammut and Geoffrey I Webb. *Encyclopedia of machine learning and data mining*. Springer Publishing Company, Incorporated, (2017).

[163] Noelia Sánchez-Maroño, Amparo Alonso-Betanzos, and María Tombilla-Sanromán. "Filter methods for feature selection–a comparative study". In: *International Conference on Intelligent Data Engineering and Automated Learning*. Springer. (2007), pp. 178–187.

[164] Thomas J Smith and Cornelius M McKenna. "A comparison of logistic regression pseudo R2 indices". In: *Multiple Linear Regression Viewpoints* 39.2 (2013), pp. 17–26.

[165] Henry M Sobell. "Actinomycin and DNA transcription." In: *Proceedings of the National Academy of Sciences* 82.16 (1985), pp. 5328–5331.

[166] Christos Sotiriou et al. "Gene expression profiling in breast cancer: understanding the molecular basis of histologic grade to improve prognosis". In: *Journal of the National Cancer Institute* 98.4 (2006), pp. 262–272.

[167] Michael Steinbach and Pang-Ning Tan. “kNN: k-nearest neighbors”. In: *The top ten algorithms in data mining*. Chapman and Hall/CRC, (2009), pp. 165–176.

[168] William P. Skelton Stephen W. Leslie Taylor L. Soon-Sutton. “Prostate Cancer”. In: *StatPearls [Internet]* (2024).

[169] Jill C Stoltzfus. “Logistic regression: a brief primer”. In: *Academic Emergency Medicine* 18.10 (2011), pp. 1099–1104.

[170] Shan Suthaharan. “Machine learning models and algorithms for big data classification”. In: *Integr. Ser. Inf. Syst* 36 (2016), pp. 1–12.

[171] A Szabo et al. “Variable selection and pattern recognition with gene expression data generated by the microarray technology”. In: *Mathematical Biosciences* 176.1 (2002), pp. 71–98.

[172] R Core Team. *R: A language and environment for statistical computing*. (2013).

[173] R Core Team et al. “Package stats”. In: *The R Stats Package* (2018).

[174] ThoughtCo. *What Are the 3 Parts of a Nucleotide? How Are They Connected?* Last accessed 4 January 2024. (2017). URL: https://www.thoughtco.com/what-are-the-parts-of-nucleotide-606385.

[175] Robert Tibshirani. “Regression shrinkage and selection via the lasso”. In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 58.1 (1996), pp. 267–288.

[176] Shunsuke Tsukamoto et al. “Clinical significance of osteoprotegerin expression in human colorectal cancer”. In: *Clinical Cancer Research* 17.8 (2011), pp. 2444–2450.

[177] Sante Tura, Michele Baccarani, and Giovanna Corbelli. “Staging of chronic myeloid leukaemia”. In: *British Journal of Haematology* 47.1 (1981), pp. 105–119.

[178] Cancer Research UK. *Breast Cancer*. Last accessed 7 July 2024. (2019). URL: https://www.cancerresearchuk.org/about-cancer/breast-cancer.

[179] Bayside Urology. *Prostate Cancer*. Last accessed 30 October 2024. (2023). URL: https://www.baysideurology.com.au/prostate-cancers.

[180] Vinod Vathipadiekal et al. “Creation of a human secretome: a novel composite library of human secreted proteins: validation using ovarian cancer gene expression data and a virtual secretome array”. In: *Clinical Cancer Research* 21.21 (2015), pp. 4960–4969.

[181] John Verzani. *Getting started with RStudio*. " O’Reilly Media, Inc.", (2011).

[182] Priya Wadgaonkar. “Environmental causes of cancer”. In: *Cancer Epigenetics and Nanomedicine*. Elsevier, (2024), pp. 69–92.

[183] Lianxi Wang, Shengyi Jiang, and Siyu Jiang. “A feature selection method via analysis of relevance, redundancy, and interaction”. In: *Expert Systems with Applications* 183 (2021), p. 115365.

[184] Ling Wang et al. “Application of relative entropy and gradient boosting decision tree to fault prognosis in electronic circuits”. In: *Symmetry* 10.10 (2018), p. 495.

[185] Ted W Way et al. “Effect of finite sample size on feature selection and classification: a simulation study”. In: *Medical Physics* 37.2 (2010), pp. 907–920.

[186] Eric W Weisstein. “Normal distribution”. In: *https://mathworld. wolfram. com/* (2002).

[187] Frank Wilcoxon. “Individual comparisons by ranking methods”. In: *Biometrics bulletin* 1.6 (1945), pp. 80–83.

[188] Suraj Yadav. *What is Kernel Trick in SVM ? Interview questions related to Kernel Trick*. Last accessed 16 January 2024. (2023). URL: https://medium.com/@Suraj_Yadav/what-is-kernel-trick-in-svm-interview-questions-related-to-kernel-trick-97674401c48d.

[189] Roi Yehoshua. *Random Forest*. Last accessed 30 March 2023. (2023). URL: https://medium.com/@roiyeho/random-forests-%98892261dc49.

[190] Tsz-Lun Yeung et al. “TGF-β modulates ovarian cancer invasion by upregulating CAF-derived versican in the tumor microenvironment”. In: *Cancer Research* 73.16 (2013), pp. 5016–5028.

[191] Lei Yu and Huan Liu. “Feature selection for high-dimensional data: A fast correlation-based filter solution”. In: *Proceedings of the 20th international conference on machine learning (ICML-03)*. 2003, pp. 856–863.

[192] Hongyan Zhang et al. “Improving accuracy for cancer classification with a new algorithm for genes selection”. In: *BMC Bioinformatics* 13 (2012), pp. 1–20.

[193] Jianguo Zhang et al. “Local features and kernels for classification of texture and object categories: A comprehensive study”. In: *International Journal of Computer Vision* 73 (2007), pp. 213–238.

[194] Shichao Zhang et al. “Learning k for knn classification”. In: *ACM Transactions on Intelligent Systems and Technology (TIST)* 8.3 (2017), pp. 1–19.

[195] Yaxing Zhao, Limsoon Wong, and Wilson Wen Bin Goh. “How to do quantile normalization correctly for gene expression data analyses”. In: *Scientific Reports* 10.1 (2020), p. 15534.

[196] Zhenyu Zhao, Radhika Anand, and Mallory Wang. “Maximum relevance and minimum redundancy feature selection methods for a marketing machine learning platform”. In: *IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE. (2019), pp. 442–452.

[197] Zhi-Hua Zhou. *Machine learning*. Springer Nature, (2019).

[198] Min Zhu et al. “Integrated miRNA and mRNA expression profiling of mouse mammary tumor models identifies miRNA signatures associated with mammary tumor lineage”. In: *Genome Biology* 12 (2011), pp. 1–17.