Analysis of Gene Expression Data for Glioma Grade Classification and Survival Time Prediction



Muhammad Mokhlesur Rahman (ID: 012163022)

Department of Computer Science and Engineering

United International University

A thesis submitted for the degree of

MSc in Computer Science & Engineering

April 2020

Abstract

A glioma is a sort of tumor that starts in brain or spine glial cells. Gliomas comprise about 30% of all brain tumors and tumors in the focal nervous system, and 80% of all injurious brain tumors. Brain tumors occur when there is a defect in the DNA of normal brain cells. To understand the genetic basis of the tumor, we need to identify the candidate genes. Genes are segments of DNA that contain the code for a specific protein that functions in one or more types of cells in the body. Genes are high dimensional and vary in size, depending on the sizes of the proteins for which they code. Due to the complexity and high dimensionality of genes, the classification of tumor samples remains a challenge. In this work, we have focused on a comparative study of different feature selection methods and proposed a new methodological approach to identify patterns of gene expression effectively that is useful to classify unknown samples. If the tumor types are classified correctly and predict survival time, the plan of treatment can be improved. We propose a novel pipeline framework for glioma analysis that uses several feature selection algorithms followed by effective classifier selection to predict the tumor type based on genes expressions data and find the top n number of genes as features for tumor type and also find the possible survival time in days. We applied twelve well-known machine learning algorithms such as Extra Trees Classifier (ET), AdaBoost (AB), Bagging (BAG), Gradient Boosting (GB), Gaussian Naïve Bayes (NB), Support Vector Machines (SVM), Multi-Layer Perception (MLP), Logistic Regression (LR), K-nearest Neighbors (KNN), Random Forrest (RF), Linear Discriminant Analysis (LDA), and Decision Trees (DT), with five different feature selection methods, including Univariate Feature Selection, Principal component analysis (PCA), Factor Analysis (FA), Kernel

Principal component analysis (KPCA), and Independent Component Analysis (ICA) on two datasets. Datasets are collected from the National Center for Biotechnology Information to analyze the accuracy of tumor type known as Grade classification and Survival time in days. The best performance was achieved by using Univariate Feature Selection for both datasets comparing with other feature selection methods. It is observed that using the Freije dataset, the best classification accuracy achieved by the AdaBoost classifier (98.75%) for grade classification, whereas using the Phillips dataset, Extra tree classifier has the best accuracy(95.67%). For survival classification, the best accuracy achieved by SVM for both the Freije dataset(94.5%) and the Phillips dataset(85.75%). The Median Survival time for the Freije dataset is 1098 days, and for the Phillips dataset, Median Survival time is 2275 days.

Acknowledgements

First I express our heartiest thanks and thankfulness to Almighty God for His celestial gift that makes me conceivable to finish this postulation effectively.

I would like to express my gratitude to my supervisor Dr. Swakkhar Shatabda, Associate Professor and Undergraduate Program Coordinator, Department of CSE, United International University for his generous guidance, patience, and support throughout this research. His valuable suggestions and strict guidance made it possible to prepare a well-organized thesis report.

I am also grateful to Head Examiner Dr. Md. Abul Kashem Mia, Professor, Department of Computer Science and Engineering, Bangladesh University of Engineering and Technology, for his helpful suggestions and for critically reviewing this thesis report.

I would like to thank Dr. Dewan Md. Farid, Associate Professor, Department of Computer Science and Engineering at United International University for reviewing my thesis book and his valuable comments.

My deepest gratitude and respect to Dr. Mohammad Nurul Huda, Professor and MSCSE Director, Department of Computer Science and Engineering, United International University for reviewing my thesis. Special thanks to him for his kind and extraordinary support throughout the completion of my thesis and journey at UIU.

Finally, my heartfelt gratitude is extended to my parents for their encouragement and support.

Contents

List of Figures viii

List of Tables xi

1 Introduction 1

1.1 Problem Statement . . . 1
1.2 Motivation . . . 1
1.3 Objectives of the Thesis . . . 3
1.4 Thesis Contributions . . . 4
1.5 Organization of the Thesis . . . 4

2 Background 6

2.1 Preliminaries . . . 6
2.2 Gene Expression Data . . . 6
2.3 Feature Selection Method . . . 8
2.3.1 Filter method . . . 9
2.3.2 Principal Component Analysis (PCA) . . . 10
2.3.3 Kernel Principal Component Analysis (Kernel PCA) . . . 10
2.3.4 Independent Component Analysis (ICA) . . . 11
2.3.5 Factor Analysis (FA) . . . 11
2.4 Classification Methods . . . 12
2.4.1 Decision Tree (DT) . . . 12
2.4.2 Random forest(RF) . . . 13
2.4.3 Bagging (BAG) . . . 13
2.4.4 Gradient Boosting (GB) . . . 13
2.4.5 Naïve Bayes (NB) . . . 14
2.4.6 Multi-Layer Perceptron (MLP) . . . 15
2.4.7 Support Vector Machines (SVM) . . . 15
2.4.8 Logistic Regression (LR) . . . 16
2.4.9 K-nearest Neighbors (KNN) . . . 16
2.4.10 AdaBoost (AB) . . . 17
2.4.11 Linear Discriminant Analysis (LDA) . . . 17

2.4.12 Extra Tree Classifier (ET) 17
2.5 Survival Prediction 18
2.5.1 Cox's proportional hazard model 18
2.5.2 Convergence 18
2.5.3 Prediction 18
2.6 Related Work 18
2.7 Summary 29

3 Materials and Methods **30**
3.1 System Overview 30
3.2 Datasets 33
3.3 Data preprocessing 34
3.4 Feature Selection 34
3.5 Description of Classifier 34
3.6 Performance Evaluation 35
3.7 Summary 35

4 Experimental Analysis **39**
4.1 Experimental Setup 39
4.1.1 Parameter for Feature Selection Method 40
4.1.2 Parameter for Classifier 41
4.1.3 Parameter for Survival Time Prediction 44
4.2 Results 44
4.2.1 Grade Classification 45
4.2.2 Survival Classification 59
4.2.3 Survival Time Prediction 74
4.2.4 Comparison with Previous Works 78
4.3 Summary 80

5 Summary, Limitations, and Future Work **84**
5.1 Summary 84
5.2 Limitations 84
5.3 Future Work 85

Bibliography **86**

A Top 20 selected features **94**

A.1 Freije Dataset 94

A.2 Phillips Dataset 95

B Dataset Sample **96**

List of Figures

2.1 Central Dogma of Molecular Biology. 6
2.2 An illustration of gene expression profile. 8

3.1 System flow diagram for grade classification. 31
3.2 System flow diagram for survival classification. 31
3.3 System flow diagram for survival time prediction. 32

4.1 Grade Classification for Freije: Accuracy heatmap of 5-fold for top 15 features. 46
4.2 Grade Classification for Freije: Accuracy heatmap of 10-fold for top 15 features. 47
4.3 Grade Classification for Freije: Accuracy heatmap of 5-fold for top 20 features. 48
4.4 Grade Classification for Freije: Accuracy heatmap of 10-fold for top 20 features. 49
4.5 Grade Classification for Freije: Accuracy heatmap of 5-fold for top 25 features. 50
4.6 Grade Classification for Freije: Accuracy heatmap of 10-fold for top 25 features. 51
4.7 Grade Classification for Freije: Accuracy heatmap of 5-fold for top 30 features. 52
4.8 Grade Classification for Freije: Accuracy heatmap of 10-fold for top 30 features. 53
4.9 Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 15 features. 54
4.10 Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 15 features. 55
4.11 Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 20 features. 56
4.12 Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 20 features. 57

4.13 Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 25 features. 58
4.14 Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 25 features. 59
4.15 Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 30 features. 60
4.16 Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 30 features. 61
4.17 Survival Classification for Freije: Accuracy heatmap of 5-fold for top 15 features. 62
4.18 Survival Classification for Freije: Accuracy heatmap of 10-fold for top 15 features. 63
4.19 Survival Classification for Freije: Accuracy heatmap of 5-fold for top 20 features. 64
4.20 Survival Classification for Freije: Accuracy heatmap of 10-fold for top 20 features. 65
4.21 Survival Classification for Freije: Accuracy heatmap of 5-fold for top 25 features. 66
4.22 Survival Classification for Freije: Accuracy heatmap of 10-fold for top 25 features. 67
4.23 Survival Classification for Freije: Accuracy heatmap of 5-fold for top 30 features. 68
4.24 Survival Classification for Freije: Accuracy heatmap of 10-fold for top 30 features. 69
4.25 Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 15 features. 70
4.26 Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 15 features. 71
4.27 Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 20 features. 72
4.28 Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 20 features. 73

4.29 Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 25 features. 74
4.30 Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 25 features. 75
4.31 Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 30 features. 76
4.32 Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 30 features. 77
4.33 Lifelines for Survival of Glioma Patients. 77
4.34 Freije : Kaplan-Meier survival plot of the glioma patients. 78
4.35 Freije : Cox's Proportional Hazard Model Curves. 80
4.36 Freije : Survival predictor for 5 patients. 80
4.37 Phillips : Kaplan-Meier survival plot of the glioma patients. 81
4.38 Phillips : Cox's Proportional Hazard Model Curves. 81
4.39 Phillips : Survival predictor for 5 patients. 82

List of Tables

2.1 Summery of the related works 29

3.1 Summary of the information of Freije and Phillips datasets. 34

4.1 Freije : Cox's Proportional Hazard Model 79
4.2 Phillips : Cox's Proportional Hazard Model. 82
4.3 Grade Classification : Comparison of the performance of our experiment with L. Zaakaria et al.[1] and H. Abusamra[2] work. 83
4.4 Survival Classification : Highest accuracy of the proposed model. 83
4.5 Survival Prediction : Comparison of the performance of our experiment. 83

A.1 The 20 selected gene name for Freije dataset. 94
A.2 The 20 selected gene name for Phillips dataset. 95

B.1 Freije dataset sample. 96

List of Algorithms

1 The Random Forest algorithm for Classification (RFC) 14
2 K-nearest Neighbors (KNN) Algorithm 16
3 AdaBoost Algorithm 17
4 Grade Classification Algorithm 36
5 Survival Classification Algorithm 37
6 Survival Prediction Algorithm 38

Chapter 1

Introduction

Gene expression data gained enormous significance in the latest years due to its importance in diagnoses and prognoses. In this chapter, we will discuss how genes played a significant role to occur a tumor in biological aspects and how machine learning tools can help us to overcome this.

1.1 Problem Statement

The cancer diagnosis relies on physical assessment and clinical examination, which also applies to medical records. But as it may, this analysis takes quite a while. If a tumor is found in its critical stage, it may be too late to cure the patient. To develop the diagnostic procedure, we need to know the tumor type first. A brain tumor is a group of brain cells. The cells in the body continuously divide and die, which in turn creates another cell. In many cases, new cells are created, but old cells are not destroyed. As a result, these cells are likely to become solid tumors. As our cells expand, some genes regulate them, split them into new cells and die. There are several types of brain tumors, each having various sets of gene changes. Because of the large number of genes compared to the sample count, the classification of gene expression data is challenging. Because they are either irrelevant or redundant, or a large number of genes are not informative for taxonomy. For that reason, It is essential to find out if a small number of genes are sufficient for the classification of gene expression data. Effective feature selection techniques are therefore often needed in this case to help correctly identify different types of tumors and thus contribute to a better understanding of genetic signatures and improved treatment strategies[2].

1.2 Motivation

In Terms of Biological Aspects: A brain tumor is a group of abnormal brain cells, or masses. The Skull encompasses the very solid spine. Some development within

such a confined space can cause problems. There may be cancerous (malignant) or noncancerous (benign) brain tumors. As benign or malignant tumors grow, they may cause increased pressure within the skull. This can do damage to the brain, which can be life-threatening. Primary brain tumors originate in the brain. They can develop from (1) brain cells, (2) the membranes that surround the brain, which are called meninges (3) nerve cells, and (4) glands. Gliomas, and meningiomas are the most common forms of brain tumors. Gliomas are cancers formed from glial cells. Usually these cells: help the development of the central nervous system, supply the central nervous system with nutrients, clean cellular waste, break down damaged neurons. The diagnosis of a brain tumor begins with a physical examination and a look at the medical history of a patient. The physical exam includes a very detailed neurological analysis. Doctors will conduct a test to see if your cranial nerves are intact. These are the nerves that originate in the brain. Treatment for brain tumors depends on a variety of factors, including the type, size of the tumor and the location of he tumor, as well as the age and overall health of the patient. With infants and adults, the treatment methods and schedules are different.

In terms of Machine learning aspects: Machine learning is an intelligent technique which is used to solve complex real-world problems in bioinformatics. The techniques allow the algorithm to use automated feature learning, meaning that the algorithm will learn how to integrate various features of input data into a more complex set of features from which to perform more learning, based on the dataset alone. This multi-layered approach to learning patterns in the input data helps these systems to make predictions very complex when trained on large datasets. The size and number of available biological datasets have skyrocketed in recent years, allowing bioinformatics researchers to make use of these algorithms in machine learning. Machine learning has been applied to six biological fields: microarrays, genomics, Proteomics, Systems biology, treatment of strokes, and mining of documents.

Microarrays, a kind of lab-on-a-chip, are used to collect data automatically on large amounts of biological material. Machine learning can help in the analysis of this data and has been applied to the classification, identification, and induction of genetic networks for expression patterns. This technique is helpful in tracking gene expression within a genome, helping to detect various types of cancer-based genes that are expressed on it. One of the main issues in this area is the identification of which

genes are expressed based on the data collected. Machine learning provides a possible solution to this issue, as specific methods of classification can be used to carry out this detection and prediction.

1.3 Objectives of the Thesis

This research focuses on a comparative study of the feature selection methods with different types of classification algorithms. It proposes an effective way to find out the Glioma type and predict the survival time based on gene expression data. Gene expression is the mechanism by which the genetic code-a gene's nucleotide sequence is used to guide protein synthesis and to generate cell structures. A microarray is a laboratory tool which is used simultaneously to detect the expression of thousands of genes. DNA microarrays are microscope slides which are printed in defined positions with thousands of tiny spots. Each spot includes several copies of the same sequence of DNA and is unique to a gene. Since of its "high dimensionality with a low sample," it can be difficult to figure out the tumor grade for an accurate classification in behavior.

We work on three machine learning tasks i.e., classification of tumor type, classification of survival or deceased, and prediction of survival period in days, based on gene expression data to find out the appropriate treatment plan for patients. Different types of experiments were conducted to compare the efficiency of the techniques of multiple classification methods with feature selection methods. Results disclosed the significant role of choice of features in the classification of information on gene expression. By using top n genes, the classification precision can be considerably enhanced by performing feature selection methods.

From the above motivation, the following objectives are established to reach the aim:

1. To find out and prepare a microarray-based data set.
2. To extract an adequate number of features from the dataset.
3. To design and run different machine learning classification algorithms
4. To analyze data and results to find the best algorithm for two data set.
5. To reduce the feature set to improve accuracy than the state-of-the-art.
6. Find out which feature selection methods have the highest score

7. Classify the tumor type whether Grade III or Grade IV
8. Classify whether a patient survived or not
9. Predict the survival time in Days
10. Build a tool using a different type of library using Python.

1.4 Thesis Contributions

Here in this work, we proposed a model to compare the performance of different machine learning algorithms on grade classification and survival prediction as well as finds out the possible survival time in days. Our contribution in this thesis is mentioned below.

1. We have provided a list of related works, where we summarized several research's related to machine learning algorithms.
2. We also done comprehensive analysis on benchmark datasets using wide range of parameters and algorithms
3. Five well known feature selection methods are contemplated and compared
4. We have reduced the feature by using various feature selection method.
5. We have proposed a novel pipeline framework for glioma analysis that uses several feature selection algorithms followed by effective classifier selection
6. The performances are evaluated by using standard evaluation metrics.
7. To the best of our knowledge, we solved a new problem survival time prediction using microarray gene expression data

1.5 Organization of the Thesis

The thesis is divided into five chapters. We are now in Chapter 1, which contains Abstract, Acknowledgments, the Table of Contents, and Introduction. Other chapters are designed as follows:

Chapter 2 gives a brief idea about the background work that motivated me to work with this topic.

Chapter 3 described the methodology of this research, proposed model, represents the dataset, how the datasets have been preprocessed, reduces the feature, selects the n best feature, and finally, the selective algorithm that gives the best result.

Chapter 4 discusses the outcomes and exploratory examination.

Chapter 5 presents the findings of our thesis, thesis participation and contributions, and discusses about the future works.

Chapter 2

Background

In this chapter, we present the necessary preliminaries to understand the gene expression data, feature selection method, classification method, etc. Previous work-related to gene expression data and glioma also presented in this chapter.

2.1 Preliminaries

2.2 Gene Expression Data

Gene, known as the basic genetic component in every living organism. Genes are parts of DNA (deoxyribonucleic acid) that contains genetic information for the encoding of particular RNA (cellular ribonucleic acid) and proteins. The central dogma of molecular biology explains how proteins are produced from DNA that is divided into two major stages, shown in Figure 2.1. The first stage is called transcription, which describes that a gene in DNA is expressed through shifting its coded records into mRNA(messenger ribonucleic acid). The second stage, called translation, describes the proteins produced after ribosomes have decoded information from mRNA. This two-step process that allows a gene to be expressed as proteins is known as gene expression[3].

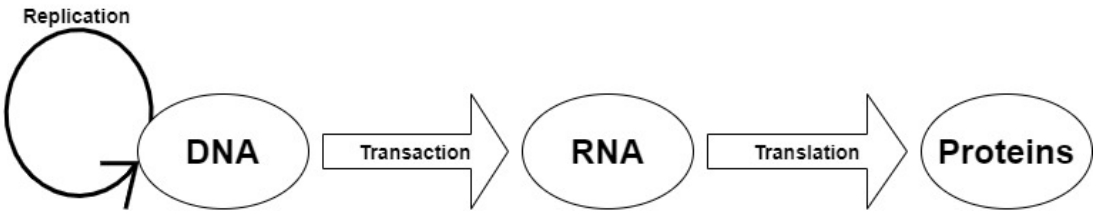


Figure 2.1: Central Dogma of Molecular Biology.

Even though almost every cell inside the body of an organism includes a specific same copy of the DNA, the mRNA level varies over time, and different cell types as well as varies within cells under different conditions. The amount of mRNA expression plays a vital role because protein production depends on mRNA production, and mRNA

levels are used as a measure of gene expression. Thus, during protein synthesis, the level of gene expression can be used as an indicator of the amount of mRNA produced in cells. Some tumors occur because of mutation of specific genes, and it is reflected in the change of the expression level of these particular genes, which means the genes are expressed abnormally in particular cells, either being up-regulated (express in a higher amount), down-regulated (express in a lower amount) or not being expressed[4]. The difference between the levels of gene expression creates a particular profile for each gene and can be achieved by calculating levels of gene expression using different techniques such as vector expression, fluorescent hybridization, and microarray[**?**]NABabu2016.

DNA microarray technology is one of the fastest-growing emerging innovations that has enabled gene expression research in such a way that scientists can now calculate the rates of expression of vast numbers of genes in a single experiment rather than running experiments and obtaining data for a single gene at a time[5]. This helps to discover the role that specific genes play in the development of the disease. In particular, the DNA microarray permits scientists and physicians to understand the pathophysiological mechanisms in diagnosis and prognosis and to choose the appropriate treatment plan for patients at the gene expression level.

A large dataset with expression values of hundreds of genes is generated using microarray technology, however, with a small number of varieties of samples. The gene expression data is usually organized in a matrix of n rows and m columns, known as a gene expression profile. Rows correspond to genes, and samples are presented in Figure 2.2 shows an illustration of a gene expression profile.

Many researchers have observed in recent years that data on microarray gene expression has a vital role in diagnosing disease, helping physicians to choose the appropriate treatment plan for patients in advance. When analyzing cancerous tumors, researchers want to differentiate and select genes, which might be responsible for the growth of tumor cells from samples. As gene expression data presented in the n x m matrix, computationally, it is impossible to analyze manually. So we need a computational method that can parse the gene expression level easily. The classification method is one of the computational methods that can classify the sample that is a disease or no disease. But the problem is, gene expression profile contains very fewer numbers of samples, but very high amounts of genes.

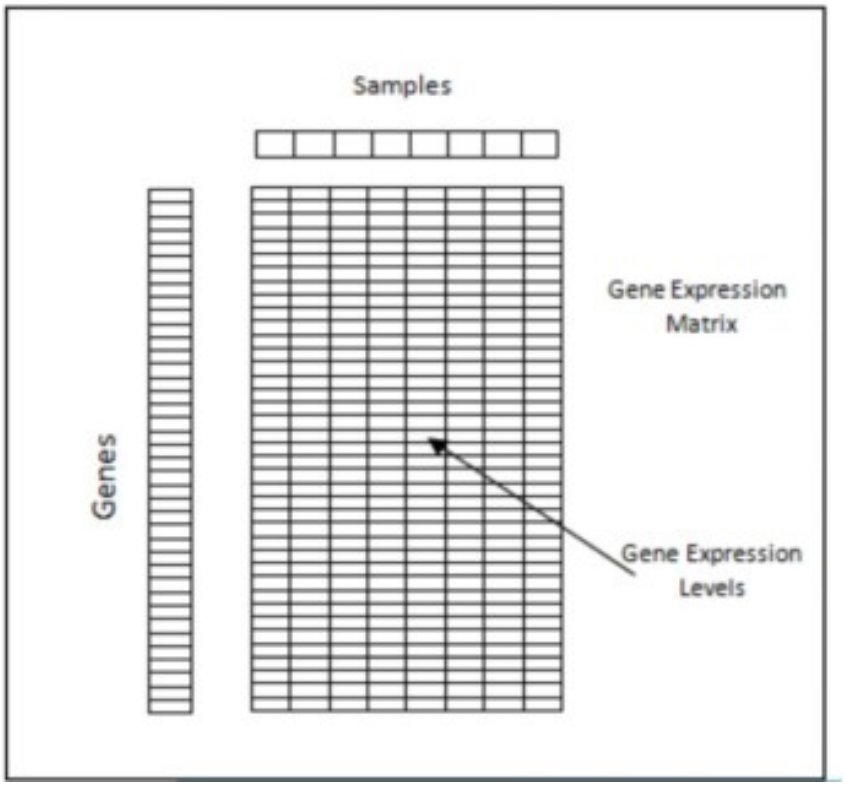


Figure 2.2: An illustration of gene expression profile.

It is also noted that a large number of genes are either redundant or irrelevant, and are not useful for classifying cancer. Only a small number of genes that can be relevant[6]. Obviously, feature/gene selection strategies are needed to build a simple method for accurate cancer classification[7]. Over-fitting problems also arise because the data is crowded with redundant features, so there are a limited number of features there. Therefore collection of features will greatly reduce the classification problem 's computational burden[8].

2.3 Feature Selection Method

Embedded methods, wrapper method and filter method are three popular approaches for feature selection. For feature selection, the filter approach and the wrapper approach are widely employed[9]. Filter methods are those which select features as a phase of preprocessing. That is, they choose the features without taking into account

the accuracy of classification. On the other hand, the wrapper methods use an algorithm's predictive accuracy to determine the possible subset of features and select the sub-set of features that provides the highest precision. Thus, although wrapper methods are usually more accurate than filter methods, they are much more expensive in computational terms and have a higher risk of overfitting than filter methods. Wrapper methods are often infeasible in problems with a large number of features, such as problems with gene expression, and filter methods are often adopted because of their computational efficacy[10].

Surprisingly, past studies have found filter methods that ignore interactions between variables can be just as effective as multivariate methods that consider these interactions[11]. A possible alternative to feature selection is dimensionality reduction, whereby a high-dimensional space is transformed into an area of fewer dimensions via linear or non-linear mappings, often taking into account complex interactions between variables[12]. It was recently suggested that unsupervised dimensionality reduction techniques could have higher predictive performance then filter methods[13]. In this study, four unsupervised dimensionality reduction methods: Principal Component Analysis (PCA), Independent Component Analysis (ICA), Kernel Principal Component Analysis (Kernel PCA), and Factor Analysis (FA) were used due to their simplicity, computational performance, and readily available to implement. So, we want to compare these five-technique and want to know which technique has the highest accuracy.

2.3.1 Filter method

For any machine learning algorithm, filter methods select features independently from a dataset. Such methods depend only on the characteristics of these variables, and before learning starts, features are filtered out of the data. There are two types of methods to filter: Univariate and Multivariate.

2.3.1.1 Univariate Filter Method

Each feature is assessed independently by univariate filters. Univariate filter methods assess a single feature and rank it according to specific criteria. They treat each feature of the feature space individually and separately. In practice this is how it works: It

ranks features according to certain rules. Then select the features with the highest ranking according to certain criteria.

2.3.1.2 Multivariate filter methods

Multivariate filter approaches measure the entire feature space. They consider features in relation to other features in the dataset. Such methods can accommodate duplicated, overlapping, and correlated features.

2.3.2 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a statistical technique used to reduce the dimensionality of the data (reduce the number of features in the dataset) by selecting the most important features capturing maximum data set information. Original data set features are converted into the main components that are the linear combinations of existing features. PCA breaks down the multivariate dataset into a set of orthogonal linear unrelated components, known as "key components," which explain the maximum amount of variation. The significant difference is defined by the first principal component, followed by the second, and so on. We can say that PCA generates new orthogonal variables, which might be linear combinations of the unique variables. So the basic idea when using PCA as a tool for selecting a feature is to select variables according to the magnitude of their coefficients (from the largest to the smallest in absolute values). PCA seeks to replace p variables (more or less correlated) by $k<p$ uncorrelated linear combinations of the original variables. Let's ignore how an optimal p can be selected for the problem. Those main components k are ranked by importance through their explained variance, and each variable contributes to each component with varying degrees. Use the highest criterion for variation will be equivalent to abstraction of functions, where the main factor is used as additional features instead of the original variables.

2.3.3 Kernel Principal Component Analysis (Kernel PCA)

Kernel PCA is an extension of Principal Component Analysis (PCA) using the kernel methods technique for non-linear data. Kernel PCA is a popular linear PCA that allows extraction of nonlinear features. In Kernel PCA, data in the input space is mapped

to larger (usually) dimensional feature space where the data can be modeled linearly. Typically, a kernel function induces the feature space implicitly, and linear PCA in the feature space is executed via the kernel trick. However, due to the implicitness of the feature space, some extensions of PCA such as robust PCA cannot be directly generalized to Kernel PCA[14]. PCA can be used for selection of features, because each main component is a linear combination of original features, and so one can see which original features contribute most to the most influential key components. But each main component in the kernel PCA is a linear combination of target space features, and the target space is infinite-dimensional.

2.3.4 Independent Component Analysis (ICA)

ICA is a computational method to identify hidden factors underlying random variables, measurements, or sets of codes. The ICA describes a generative model for the observed multivariate data, normally given as a large sample dataset. Within model , it is assumed that the data variables are linear mixtures of some unknown latent variables, and therefore the mixing system is also unknown. The latent variables are believed to be non-Gussian and mutually independent, and are considered the discrete components of the data being observed. These independent components are often discovered by ICA, also called sources or factors. If PCA is used as feature selection methods, that work well for normal data, but do not perform adequately for non-normal data[15]. In contrast, ICA works better, the more non-Gaussian the data is[16]. ICA transforms the original feature vectors or directions onto statistically independent directions by maximising the degree of non-Gaussianity of the new directions[15].

2.3.5 Factor Analysis (FA)

Factor Analysis (FA) is a similar method to PCA since both methods involve linear combinations of original variables to create new variables. However, Factor Analysis (FA) aims to reproduce inter-correlations between variables, where non-observed factors are common variances excluding unique variances. Factor Analysis consists of techniques used to evaluate the association between a series of random variables detected or counted or calculated within each particular entity. The purpose of the factor analysis is to account for the inter-relationships between the n variables, by applying a set of common factors which are considerably less in number than the n

number of these variables. Factors are random variables that can not be observed, counted or measured directly, but are presumed to exist in the population and are therefore present in the experimental sample[17]. They, as well as the manifest variables, are presumed to vary from person to person. Factors are also sometimes referred to as latent variables. Factors are, in fact, abstract or causal constructions. Throughout factor analysis, the primary focus of concern is on specific factors, which are defined by comparison to the variables detected or counted or calculated. The unique factors in each variable are just whatever part of that variable is unrelated to all other variables in the total collection of observed variables, including its measurement error[18].

2.4 Classification Methods

In recent years, different classification algorithms in machine learning have been used to predict and classify different types of tumors using gene expression data. We compared twelve different classification methods: Decision Trees (DT), Random Forrest (RF), Bagging (BAG), Gradient Boosting (BST), Gaussian Naïve Bayes (GaussianNB), Multi-Layer Perceptron (MLP), Support Vector Machines (SVM), Logistic Regression (LR), K-nearest Neighbors (KNN), AdaBoost, Linear Discriminant Analysis, and Extra Tree Classifier. These methods were used because of their popularity.

2.4.1 Decision Tree (DT)

Decision tree (DT) is a decision support tool that uses a model such as a decision tree and its possible consequences, including event cost, source costs, and utilities. One way to demonstrate the algorithm is to include only conditional control statements. The purpose of using the decision tree is to develop a training model that can be used to predict the class or value of a target variable by studying simple decision rules derived from previous data. In the decision tree to predict the class label to write, we start with the root of the tree. We compare the root attribute values with a record attribute. By comparison, we follow the branch associated with this value and move on to the next node.

2.4.2 Random forest(RF)

Random Forest (RF) is a classifier using a collection of decision trees, each of which is a simple classifier that is learned from the data on a random sample. The classification of a new sample is carried out by majority vote of the trees of decision. Given N training samples, each having M features, each tree is grown as follows: First, n instances from the training set are sampled at random (with replacement). This specimen is the tree's training set. Then m¡¡M of the features is selected at random at each node. On these m features the best split is used to branch to the node. The random forest classifier uses the "Gini index" as the splitting function to determine which feature to split on during the tree learning phase. The same value of m is used for all trees in the forest. Each tree is grown to the most substantial extent possible, without pruning.

While each tree grown by the random forest algorithm will significantly overfit the data, the whole random forest is very resistant to over-fitting, due to the impact of averaging over several different trees. In this point of view the greater the number of trees, the better. Random forest derives other benefits over decision trees and produces improved outcomes by using sample bagging, random subsets of variables, and a majority voting system. Random forests are a very well-suited classification algorithm for microarray data, and they report superior performance even when most predictive variables are noisy[19]

2.4.3 Bagging (BAG)

Bagging (BAG), additionally referred to as Bootstrap aggregating, is designed to overcome two problems: overfitting and decreases variance to improve the stability and efficiency of machine learning algorithms for statistical classification and regression. Bagging is used with decision trees, but it also similar to the Random forest (RF). If a training dataset is given, Bagging (BAG) generate multiple sets from training dataset and train each model for each set. The accuracy can be measured by averaging all models.

2.4.4 Gradient Boosting (GB)

Gradient Boosting (GB) generates a model that consists of providing sequentially many weak models to fix the mistakes of the models. Once the model is created, then the

Algorithm 1 The Random Forest algorithm for Classification (RFC)

1. **for** $b \leftarrow 1, B$ **do**
   (a) From the training data draw a bootstrap sample $\mathbb{Z}^*$ in dimension N.
   (b) Grow a T_b random forest tree to the bootstrapped data by repeating the following steps recursively for each tree terminal node, until the minimum node size $n$min is reached.
   (c) From the variables of p, select m randomly.
   (d) Select the best variable / split-point between the m.
   (e) Split the node into two daughter nodes.
2. Return the ensemble tree $\{T_b\}_1^B$
3. Predict at new x point:
4. Let $\hat{C}_b(x)$ be the class predictor of a random forest tree bth. Then $\hat{C}_r f^B(x) =$ majority vote $\{\hat{C}_b(x)_1^B\}$.

model makes predictions that may be weighted by their precision, and the summation of the results is the final output. Boosting is similar to Random forest (RF) and Bagging (BAG) in that it is an ensemble method built from a Decision tree (DT).

2.4.5 Naïve Bayes (NB)

The Naïve Bayes Classifier is a finite set of probabilistic machine learning models designed to prepare, examine and characterize data. Naïve Bayes is based on the theorem of the Bayes. However, there is no specific algorithm for the training of these classifiers; therefore, Naïve Bayes pretends that the value of a particular feature is independent of the value of any other feature given the class variable. For example, a vehicle may be called a truck because it has four wheels and a rectangular shape. The classifier of Naïve Bayes will find any of these properties as a distinct contribution to the probability that the object is a tractor, regardless of the association between the number of wheels or the shape.

Bayes theorem provides a way of measure posterior probability $P(c|x)$ from

$P(c), P(x)$ and $P(x|c)$. Look at the equations below:

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} \tag{2.1}$$

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \cdots \times P(x_n|c) \times P(c) \tag{2.2}$$

Here:

- $P(c|x)$ is the posterior probability of class $(c, target)$ given predictor $(x, attributes)$.
- $P(c)$ is the class's prior likelihood.
- $P(x|c)$ is the probability of a given class predictor.
- $P(x)$ is the prior probability of predictor.

2.4.6 Multi-Layer Perceptron (MLP)

To classify a linearly distinct data sets, the perceptron is very helpful. A Multilayer Perceptron (MLP) can classify datasets that are not linearly separable. It is a network of perceptrons (artificial neurons) arranged in a feed-forward-only topology. An input layer, a hidden layer, and an output layer are needed to design an MLP network. The connection between the input and output is directed.

2.4.7 Support Vector Machines (SVM)

A support vector machine (SVM) is a classifier for differentials formally described by a separate hyperplane. That is, the algorithm generates an optimal hyperplane which classifies new instances, provided the labeled training data. This two-dimensional space hyperplane is a line dividing a plane into two parts, in which each class lay on either side. Support vector machines are constructed using training data and hyperplane display in test data.

2.4.8 Logistic Regression (LR)

Logistic regression is a linear algorithm to assume a linear correlation between the input variables with the output. For example, to find out whether the tumor is malignant or benign. Either the result will be only one. By using linear regression for this problem, a threshold must be set on the basis of which classification can be defined. When the actual class is malignant, the expected constant value is 0.4, and the threshold value is 0.5, the data point is categorized as benign, and can lead to significant effects in real time.

2.4.9 K-nearest Neighbors (KNN)

K-nearest neighbor is an algorithm that is used to store all possible cases and classify new events based on distance. A case is classified by its neighbors' majority vote, with the case being assigned by a distance function to the most common class among its nearest K neighbors. If K = 1, then the case is simply assigned to its closest neighbor's class.

Algorithm 2 K-nearest Neighbors (KNN) Algorithm

1. At first, load datasets
2. Initialize K for the number of neighbors you want
3. **for** every single element in the datasets **do**
   (a) Calculate the distance between the example query and the current example from the data.
   (b) Add the distance and the example index to the ordered collection
4. Sort the ordered distance and indice collection from the smallest to the largest (in ascending order) by distance
5. Choose the first K entries from the sorted list
6. Get the labels for the chosen K entries the K Label mode

2.4.10 AdaBoost (AB)

Freund and Schapire propose the AdaBoost method for multi-class classification. It may be used combined with other learning algorithms to enhance precision. It produces a robust classifier from the collection of different weak classifiers.

Algorithm 3 AdaBoost Algorithm

1. Initialize weights for observation $w_i = 1/N$, $i = 1, 2, \ldots, N$
2. For $m = 1$ to M:
   (a) Using weights w_i, fit a classifier $G_m(x)$ into the training data.
   (b) Compute
   $$\mathrm{err}_m = \frac{\sum_{i=1}^{N} w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^{N} w_i}.$$
   (c) Calculate $\alpha_m = \log((1 - \mathrm{err}_m)/\mathrm{err}_m)$.
   (d) Assign $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))]$, $i = 1, 2, \ldots, N$.
3. Return $G(x) = [\sum_{m=1}^{M} \alpha_m G_m(x)]$.

2.4.11 Linear Discriminant Analysis (LDA)

Linear discriminant analysis (LDA) is a technique that can find a linear series of features that discriminate or groups two or more classes of objects or events. Logistic regression is used only for two-class classification problems, but LDA can classify more than two types.

2.4.12 Extra Tree Classifier (ET)

Extra Trees Classifier (ET) is a type of learning strategy for the ensemble that aggregates the outcomes of several de-correlated decision trees gathered in a "forest" to produce the classification outcome. In nature it is very similar to a Random Forest Classifier, which only differs from it in the way decision trees are constructed in the forest.

2.5 Survival Prediction

We often have extra data besides the duration we want to use. The technique is called survival regression – the name implies that we regress covariates against another variable (e.g., country, age, etc.) – in this case, duration.

2.5.1 Cox's proportional hazard model

The concept behind Cox's proportional hazard model is that an individual's log hazard is a linear function of their static covariates, and a baseline hazard at population level that changes over time.

2.5.2 Convergence

It involves the use of iterative methods to fit the Cox model to the data. In addition, correcting the warnings would tend to simplify and decrease the amount of iterative steps needed in the Cox proportional hazard model.

2.5.3 Prediction

Predicting the event time of censored subjects is a common use case. It is simple to do but first we need to measure an basic probability of condition.Let T be the (random) event time for some subject, and S(t)P(T¿t) be their survival function.

2.6 Related Work

Throughout this chapter, we give descriptions of various methods of information that is used in the literature to propose a predictive model. This literature review will assist in representing the difference between the previous model to our proposed method.

In 2004, Freije et al.[20] analyzed 85 diffuse infiltrating gliomas from four histological types of probe sets and identified 595 different expressed genes that correlated with overall survival. Hierarchical clustering showed four molecular subtypes (labeled HC2B, HC2A, HC1A, and HC1B) which separated into two distinct survival clusters (P=0.00011). Patients with grade III tumors had a median survival of 4.8 years, whereas patients with grade IV tumors had a median survival of 293 days[20]

In 2006, Phillips et al. [21] suggested subtyping of gliomas into three subtypes based on gene expression profiling: proneural, proliferative and mesenchymal. They found a strong association between tumor grade and subtypes regardless of oligodendroglial or astrocytic morphology[22].

Heba Abusamra[2] presented a comparative study of eight different feature selection methods and three different classifiers on two datasets (Freije et al., 2004[20], Phillips et al., 2006[21]) of microarray gene expression data. The feature selection methods are: Gini index, information gain, the sum of variances, towing rule, sum minority, t-statistics, one-dimension support vector machine and max minority and the Classification method are k-nearest neighbor, random forest and support vector machines. Rankgene Software selected the top 20 genes that have the highest score in feature selection. Then three different classification methods SVM, KNN, and the random forest were applied and obtained the accuracy on the test set. The top 20 genes that have the best score in feature selection were selected by Rankgene Software. Then three separate classification methods (KNN, random forest, and SVM) were applied and obtained the accuracy on the test set. SVM reported the highest accuracy of 94.59% and 90.81% respectively in Freije and Phillips. When K is set to 7 in both datasets KNN achieved its best performance. Random forest reported 90.54% accuracy in the Freije data set, and 87.96% accuracy in the Phillips data set[2].

Lamiaa Zakaria et al.[1] presented a comparative study on the gene expression data using two different datasets, Frieje[20], Phillips[21], that SVM performed better than the other two classification methods, K-Nearest Neighbors (KNN) and Random Forest. These data sets shared precisely the same genes for the same disease named Glioma. In the Freije data set, the SVM classification had better accuracy (91.8%) than the Philips data set (86.7%). KNN achieved the best performance(91.5% on Frieje Dataset, 84.6% on Phillips datasets) when K equals 3. Random forest reports the minimum accuracy in the Freije dataset(87.9%) and Phillips datasets (49.1%). Selection of features was only done on the training set. Many gene numbers (20, 50 , 100 , 150, 200 and 250) were checked and the top 20 genes with the highest score were chosen because they performed better, took less time and required less memory relative to other genes..[1].

Huiqing Liu et al.[23] presented a comparative study using two datasets (Leukemia and Ovarian Cancer) of five feature selection methods. Entropy-based,

signal-to-noise statistic, correlation-based, t-statistics, and Chi-squared are used as feature selection methods. Author picked the top 20 genes with the highest score in t-statistics, signal-to-noise, chi-squared and chose all the features suggested by correlation-based functions. For entropy, features having an entropy value less than 0.1 were selected if existed, or the 20 features with the lowest entropy values were selected otherwise. The effectiveness of these features was evaluated using SVM, naïve Bayes, C4.5, and KNN classifiers. Without feature selection, SVM reported the least error rate among the other classification methods. When applying feature selection on the datasets, the accuracy performance of the four classifiers was significantly improved in most cases[2].

Pirooznia et al.[24] evaluated and compared the efficiency of eight different classifiers, including RBF Neural Nets, Bayesian, MLP Neural Nets, SVM, Random Forrest methods and Decision Tree (J84, ID3). Several clustering approaches, including density-based clustering, expectation-maximization clustering and K-means, were also applied to eight separate binary (two-class) microarray datasets. In addition, feature selection methods efficiency was compared, including correlation, recursive feature elimination (SVM-RFE), supporting vector machine, and chi-squared-based feature selection. Ten-fold cross-validation has been used to calculate the classifiers' accuracy. First, various types of classification methods were applied to all datasets, without any selection of features. Neural networks and SVM were performing better in most datasets than other methods of classification. Then the effect of feature selection methods was examined on the different classification methods. A number of genes (50, 100 , 200, and 500) were tested and the top 50 genes were selected because they gave a good accuracy, consumed less processing time, and required fewer memory configurations compared to others. The accuracy performance of classifiers was improved after applying feature selection methods to the datasets. In all cases, SVM-RFE performed very well when it was used with SVM classification methods[2].

Li et al.[25]'s studied and compared multi-class classification results using many feature selection and classification methods on nine datasets of multi-class gene expression. With eight methods supported in this software, the "RankGene" software was used to select the informative and related genes on the training set: Gini index, t-statistics, one-dimensional SVM, twoing rule, max minority, the sum of variances,

sum minority, and information gain. Author selected the top 150 ranked genes in each dataset. The multi-class classifiers used to assess the genes selected were: KNN, SVM, decision tree and Naïve Bayes. In the experiments, the original partition of the datasets into training and test sets was used whenever information about the data split was available. Otherwise, four-fold cross-validation was applied. They found that SVM has the highest results on all datasets. The KNN classifier provided a fairly reasonable result for certain datasets, which indicates it is not problem-dependent. Some fascinating aspects in their study is that it was impossible to chose the right method in selection of features, so it was quite confusing how the selection methods interacted. Due to the separation of the gene selection part from the classification part, there is no learning mechanism to learn how that component interacts with each other[2].

Eric P. Xing et al.[26] evaluated and compared the efficiency of three different classifiers, including Gaussian Classifier, Logistic Regression, and K Nearest Neighbor Classification for high dimensional microarray data. Also, three feature selection methods, such as Markov Blanket filtering, Unconditional Mixture Modeling, and Information Gain Ranking, have compared on leukemia microarray-based dataset. The dataset contains 72 samples from leukemia patients, with 7130 genes on each sample giving the expression levels. Then feature selection methods are applied to different classification methods. All three classifiers use the same 2-100 genes selected by the three stages of feature selection[26].

Mundra and Rajapakse[27] have done a comparative analysis on three feature selection such as SVM-RFE (Support Vector Machine Recursive Feature Elimination), MRMR (Minimum Redundancy–Maximum Relevance), SVM-RFE (SVM-RFE with MRMR Filter) with filter methods on four microarray gene expression cancer datasets, namely, leukemia, Colon Tumor, Hepato, and Prostate contains 7129, 2000, 7129 and 12600 genes respectively. Ten-fold cross-validation used to calculate the accuracy of the classifiers. The efficiency of SVM-RFE with MRMR filter was greatly improved in most accuracy tests for hepato, leukemia, and prostate datasets.

Jayger et al. [28] developed a framework for testing and developing strategies for the collection of sensitive genes from microarray results. Genes are chosen by classifying the genes according to the test equation and then picking the top k genes. Jayger et al. proposed three pre-filter methods where one is based on correlation, and the other two

are based on clustering to retrieve groups of similar genes. Five different test statistics: Fisher, Golub, Wilcoxon, TNoM, and t-test was used to compare classification on three different publicly available microarray datasets such as Alon (40 Adenocarcinoma and 22 normal samples), Golub (47 ALL and 25 AML leukemia samples) and Notterman(18 tumors and 18 normal samples). At first, Five different test statistics applied to three datasets. Then, three different filtering algorithms: Correlation, Clustering, Masked out Clustering was applied on the datasets. SVM and LOOCV scores used to evaluate the performance of the feature selection. To assess the performance, Jayger et al. varying the cluster between 1 to 30 and the number of selected features between 2 and 100. LOOCV performance is calculated, and the result showed that LOOCV produced six false classifications in Alon's colon dataset, resulting in an error of 9.7%. In Golub's leukemia dataset LOOCV produced two erroneous classifications, resulting in an error of 2.8% and Notterman's carcinoma dataset LOOCV made one false classification, resulting in an error of 2.8%[28]

Chanho et al.[29] performed a comparative study of seven methods of gene selection and six classifiers using Lymphoma and Colon Cancer data[30]. They proposed a classification system. The system architecture is as follows: at first normalized the gene expression profile. Then important features need to be selected using statistical correlation analysis, Distance measure approach, and Information-theoretic approach, etc. and classification was performed simultaneously. The classifiers are described as follows: k-nearest neighbor (k-NN), Multi-layer perceptron (MLP), Support vector machine (SVM), and Structure ' adaptive' self-organizing map (SASOM), etc. This process named Feature-Classifier pairs. Then fitness evaluation performed using the Genetic Algorithm to Search Optimal ensemble. The accuracy of the ensemble evaluates the fitness. The proposed method applied to lymphoma (24 samples of germinal center) and colon cancer (62 samples of colon epithelial cells) datasets. Each sample contains 2000 gene expression levels. Based on feature score, 25 high score genes are selected to search for the optimal ensemble of classifiers. Experimental results showed that IC yields good performance in gene selection, and KNN yields superior performance in classification. The best accuracy of single feature-classifier pair is 83.9%, but the ensemble searched by GA yields 93.5% accuracy by majority voting.

Hala Alshamlan et al.[31] proposed an Artificial Bee Colony (ABC) algorithm to analyzed microarray-based gene expression profile and an innovative feature selection

algorithm called minimum redundancy maximum relevance (mRMR) was proposed to select informative genes from microarray profile. They were combining both algorithms, and mRMR-ABC is proposed to select high score genes from gene expression profiles. The proposed algorithm performance was evaluated on six binary and multi-class gene expression microarray datasets (colon, leukemia, lung, and prostate).

The algorithm is divided into three phases: pre-processing, gene selection, and classification. LOOCV (Leave-One-Out Cross-Validation) used to assess the performance of the proposed algorithm. Preprocessing: the MRMR filter technique is used to identify the relative and informative gene subset of the candidate microarray dataset. When choosing a gene, the ABC algorithm is used to pick the predictive genes from the mRMR gene subset. Eventually, the SVM classifier was educated and checked using the chosen genes and the accuracy of the classification was returned. The findings showed that the new algorithm achieved greater improvement as compared to the other algorithms previously proposed[31].

Manish Babu et al.[30] present a comparative study of Gene Selection Methods for Cancer Classification Using Microarray Data using two different datasets such as Colon Tumor Data and Leukemia ALL-AML Data. Manish Babu et al. studied the impact of changing the number of used features in the classification methods, and how the change in the numbers of features affects the classification accuracy. At first, the classification methods without any feature selection were applied to both datasets[30]. SVM classification had the best accuracy(65%) in the Colon dataset with accuracy better than the Leukemia dataset (58.8%). KNN achieved the best performance(75% on the Colon dataset, 67.6% on Leukemia dataset). Then, the classification methods with six feature selection were applied to both datasets, and several numbers of genes were tested like (10, 20, 30, 400, 50, and 60) and top 50 genes that have the highest score was selected for colon data and 40 genes for Leukemia data. It showed that KNN performs well better than SVM. KNN had the best accuracy(99%) in the Leukemia dataset with accuracy better than the Colon dataset (89%) with feature selection using Support Vector Machine Recursive Feature Elimination (SVM-RFE). SVM performed well if the feature is selected by Information gain, and the accuracy is 81% on colon data and 98% on Leukemia data if Support Vector Machine Recursive Feature Elimination used for feature selection.

Frank G. Zöllnera[32] compared three feature reduction techniques (i) principal component analysis (PCA), (ii) independent component analysis (ICA), and (iii) Pearson's correlation coefficients (PCC) in support vector machine (SVM)- based classification of glioma grade. Among 101 patients, 63 were diagnosed with high-grade glioma (14 grade III and 49 grade IV), and 38 patients were diagnosed with low-grade glioma (6 grade I and 32 grade II), three analytic approaches were evaluated. Tumor cerebral blood volume (CBV) histograms and patient age was used for tumor grading By the SVM approach. The best performance was achieved by PCA at 85% using three principal components from a 101-dimensional feature vector. PCC classification accuracy was 82% (89%, 77%, 2 dimensions) and 79% ICA classification. In order to improve speed (up to 30%) and simplicity, the three-method reduction feature provided similar classification accuracy to the literature values (87%) while reducing the number of features by up to 98%.[32].

Hwan-ho Cho[33] evaluated and compared the efficiency of three different classifiers, including SVM, Logistic Regression, and Random Forrest methods. Also, the top five feature was selected using Minimum redundancy maximum relevance (mRMR) with mutual information. Five-fold cross-validation was used to calculate the accuracy of the classifiers for the dataset. Datasets contain 285 samples where 210 sample is high-Grade Glioma, and other 75 is considered as low-grade glioma. Each sample was multi-modal based on T1-weighted, FLAIR images, T2-weighted, and T1-contrast enhanced. To separate the training and test data, five-fold cross-validation was adopted. Four hundred sixty-eight radionics features were calculated, and using Minimum redundancy maximum relevance (mRMR) top five features were selected. Model performance was measured using accuracy, specificity, area under the curve (AUC) and sensitivity of the receiver operating characteristic curve. Best classification accuracy was 0.8877 for Random forest and 0.8807, 0.8877 accuracies for SVM, and Logistic Regression respectively. The average accuracy for the predictive model was 0.8854. An average AUC was 0.9030 (Support vector machine 0.8866, logistic regression 0.9010 and random forest 0.9213).

Supriya Patil et al.[34] suggest a combination of ratio of mean values of a particular gene of both the classes, t-statistics, and standard deviation to obtain the optimal subset of genes. With the help of the wavelet transform, the size of this data is further cut down. Finally, the classification is performed using neural network algorithms.

Resilient Back Propagation Algorithm (RPROP) algorithm with Bior1.3 and Bior2.4 wavelets gives 100% classification accuracy[34] using 25 genes.

Siqi Wang et el.[35] merged 34 microarray glioma datasets. The datasets contains 1893 glioma samples and conducted genetic data mining through bioclustering, pathway analysis, and statistical analysis. Studies have been confirmed in TCGA, CGGA and internal cohorts. Eight differentially expressed glioblastoma-specific genes were tested using GEO. Patients with high expression BARD1,IFRD1, CBX3, or EGFR have a slightly reduced survival in the TCGA and CGGA cohorts. Still, patients with high expression of MOBP or GUCY1A3 had significantly longer survival than those with lower expression of these genes. We picked the CBX3 gene for further studies, after reviewing the literature. Authors confirmed that the immunohistochemistry analysis of tissue microarrays and the qPCR analysis of surgical specimens have overexpressed CBX3 in glioblastoma. Functional assay results have shown that silencing CBX3 stops the cell cycle in the G2 / M phase, thus weakening the ability of cell proliferation[35].

Kaplan et al.[36] proposed a new scheme to categorize three types of brain tumors from MRI images, namely the Meningioma, Glioma and Pituitary tumours. First, preprocessing is applied on MRI images. Then, LBP images were generated by feature extraction method of LBP, nLBP and LBP functions from the images. Suggested methods are fundamental, feasible, and low-cost methods. Histograms of the generated images of LBP, nLBP, and LBP tumors were then formed, and matrices of features were obtained. The classification process was performed by using Artificial Neural Networks (ANN), Random Forest (RF), K-Nearest Neighbor (Knn), Linear Discriminant Analysis (LDA), and A1DE with the obtained feature matrices. The highest success rate in brain tumor classification was 95.56% with the nLBPd=1 feature extraction method and Knn model[36].

Muhammad Sharif et al.[37] presented a new methodology based on an extreme learning model. Experiments are carried out on two datasets i.e. Leader board databases and BRATS challenge. The method achieves 0.99 DSC, 1.00 PPV, 0.98 SE, 0.02 FNR, 0.00 FPR, 0.98 JSI,on BRATS 2012, 0.00 FPR, 0.99 DSC, 0.98 JSI, 0.98 SE, 1.00 PPV, 0.00 FPR, 0.02 FNR, on BRATS 2013, 86.7 SE, 89.3 DSC, 0.12 FPR, 80.7 JSI, 90.8 PPV, 0.14 FNR, on 2013 Leader board, 91.2 PPV, 0.12 FNR, 89.3 DSC, 81.4 JSI, 88.0 SE, 0.13 FPR on BRATS 2014 and 0.05 FNR, 95.5 PPV, 0.07 FPR, 95.0 SE, 0.90 JSI, 95.3 DSC, on BRATS 2015 databases[37].Muhammad sharif

et al used brain magnetic resonance imaging based on automated segmentation and tumor evaluation classification methods. They said the presented scheme works out in less time and shows improved outcome.

Bing Niu et al. intended to evaluate glioma molecular pathways based on protein-protein interaction networks, together with machine learning techniques. Using protein-protein interaction (PPI) networks, primary differentially expressed genes (DEGs) were scanned and identified. As a result, 19 genes in Grade I through Grade II, 21 genes in Grade II through Grade III, and 20 genes in Grade III and Grade IV. Five methods of machine learning were then used to predict the stages of gliomas based on selected primary genes. Upon reference, the Complement Naïve Bayes classifier was used to construct a predictive model for grade II-III with 72.8 percent precision. And Random Forest was used to generate a forecast model for grade I-II and grade III-VI with precision of 97.1% and 83.2%, respectively. Finally, selected genes were analyzed using PPI networks, Gene Ontology (GO) terminology, and Kyoto Encyclopedia of Genes and Genomes (KEGG) pathways, and the results enhance our understanding of the biological roles of selected DEGs involved in glioma development. The author expects the expressed key genes to have a guiding significance for the occurrence of gliomas or, at the very least, to be useful for tumor researchers.[38].

Golub et al.[39] has shown an effective method for identifying the predictive gene subset for cancer classification. This work provides a neighborhood study in the identification of a gene subset that can differentiate between the two types of cancer: AML and ALL. The proposed neighborhood analysis is similar to t-statistic, which has been applied on a leukemia dataset. The leukemia dataset contained 72 samples with 6,817 gene expression levels. A gene subset is selected via a training sample that can distinguish between the two cancer types: AML and ALL. Golub et al. identified fifty of the most sensitive genes by testing samples and approved training samples to classify the test samples.

Ashok Kumar Dwivedi[40] presents a comparative study of Artificial Neural Network (ANN) with the other five traditional machine learning techniques. Eight different performance measures attribute used to analyzed the six machine learning-based classification using acute lymphoblastic leukemia (ALL) and acute myeloid leukemia (AML) datasets. The datasets generated based on microarray gene expression profile data. The datasets contained 46 samples of microarray gene

expression profiles of 7129 genes for the classification of AML and ALL. Ten-fold cross-validation and leave-one-out methods used to validate the results. Using ANN without an error, 98% accuracy achieved to identify acute lymphoblastic leukemia and only one error to identify acute myeloid leukemia.

Pablo Guillen et al.[41] presented an application of supervised machine learning for the classification of several types of cancer via deep learning[41]. At first, Pablo Guillen et al. presented a network that can learn well. The network can categorize the discrete specimens with high efficiency, and high accuracy is obtained in the task of divining the distinct classes. The repercussion of this work is, molecular techniques like gene expressions microarray data with deep learning has the capability to capture gene expression models, which may increase the classification performance.

Zhong et al.[42] compared their proposed method Indexes B/SVM to other methods of classification such as SWKC/SVM and SVM-RFE[1]. Zhong et al. implemented a system of gene selection method using SVMs. Dataset of leukemia and Colon cancer used to test the new model. In the proposed method, All the genes were rank according to the magnitude of Bhattacharyya distances between the two distinct classes. Then the best gene subset is chosen as the one which presents the lowest misclassification rate in the constructed SVMs following a forward selection algorithm. Later, to find the best parameters for SVM, the ten-fold cross-validation is applied with the final optimal gene subset so that the predictive classification model is built and trained. Finally, the classification model is evaluated by its prediction performance for testing set[42]. The outcome of their work is, Proposed B/SVM model performs well rather than the other two in terms of small average misclassification rate (1.1%) and high average recovery rate (95.7%). This indicates that B/SVM seems to be more efficient and has more influence on finding out the marker genes rather than SVM-RFE and SWKC/SVM.

Yuanyuan et al.[43] presented the Genetic algorithm/k-nearest neighbors (GA/KNN), a supervised classification method for pan-cancer classification. The genetic algorithm (GA) is the gene/feature selection engine, and K-Nearest Neighbors (KNN) algorithm is the classification tool for the GA/KNN method. They download their datasets from the TCGA data portal (https://tcga-data.nci.nih.gov), which contains UNC RNASeqV2 level 3 expression data for 9096 patients representing 31 tumor types. They compare GA/KNN tools with another supervised method named as XGBoost (gradient boosting machines). The outcome of their work is, they were

able to identify many sets of 20 genes that could correctly classify more than 90% of the samples from 31 different tumor types in a validation set[43]. They identified a few genes that might play a role in sexual dimorphism in certain cancers.

Huawen, Lei et al.[44] have proposed a new EGSG (Ensemble Gene Selection by Grouping) gene selection method for selecting multiple subsets of genes for classification. The approach is compared on six public microarray datasets with 5 known gene selection methods. Two classification methods k-nearest neighbor (KNN) and Naïve Bayes classifier (NBC) used to construct a classification model and a cross-validation method (leave-one-out) used to evaluate the classification model performance.

Thomas J Chrysman is here[45] introducing a new genetic algorithm-based random forest (GARF) modeling approach. This allows the complexity of large-scale genetic disease signatures to be reduced to a highly accurate and highly simplified gene panel. This method was applied to 803 glioblastoma polymorphism specimens. The gene panel can be reduced to 48 gene classifiers while maintaining a classification accuracy of 90.91 %, improving the best alternative methods available. Besides, using this approach, a 32-gene panel was constructed to improve the stability between RNA-seq and microarray-based classification. Cross-platform classification retention ranges from 69.67% to 86.07%.

From Table 2.1 it is observe that different researchers proposed different approaches to predict glioma grade type with different type of dataset but no one working on survival classification and survival prediction before. Also, most of them working on MRI images dataset. Therefore, we select Microarray gene expression data to work on. Another point in the selection of microarray dataset is the fact that data on microarray gene expression have become very important in recent years due to its role in disease diagnostics and predictions that help to choose the appropriate treatment plan for patients[2]. So we are highly motivated to develop a suitable model for grade classification, survival classification and survival time prediction in days with higher accuracy using microarray based gene expression dataset.

Author's Name	Dataset type	Working Principle	Used ML Algorithm	Highest Accuracy Achieved By	Accuracy	Year
L. Zakaria et al.	Microarray data	Classify tumor types based on glioma grade	Support Vector Machine (SVM), k-Nearest Neighbor (KNN), and Random Forest (RF)	SVM	91.80%	2019
Heba Abusamra	Microarray data	Classify tumor types based on glioma grade	Support Vector Machine (SVM), k-Nearest Neighbor (KNN), and Random Forest (RF)	SVM	94.59%	2013
Frank G. Zollnera et el.	MRI Images	Predict glioma grade	SVM with Pearson's correlation coefficients (PCC), Principal Component Analysis (PCA), Independent Component Analysis (ICA)	SVM	85%	2012
Hwan-ho Cho et el.	Radiomics features from MRI Images	Predict glioma grade	Support Vector Machine (SVM), Logistics Regression (LR), and Random Forest (RF)	LR, RF	88%	2018
Supriya Patil et al.	Microarray data	Predict cancer subtype using wavelet transform	Resilient Back Propagation Algorithm, Conjugate gradient back propagation with Polak-Ribiére updates, Conjugate gradient back propagation with Powell-Beale restarts, and Conjugate gradient back propagation with FletcherReeves updates	Resilient Back Propagation Algorithm	100%	2016
Kaplan et al.	MRI Images	Classify three types of brain tumors	K-Nearest Neighbor (KNN), Artificial Neural Networks (ANN), Random Forest (RF), A1DE, Linear Discriminant Analysis (LDA)	KNN	95.56%	2020
Muhammad Sharif et al.	MRI Images	Brain tumor detection based on extreme learning	Extreme Learning Machine (ELM), Regression ELM leaves one out	ELM	99%	2020
Bing Niu et al.	Gene expression data	Glioma stages prediction	Complement Naive Bayes (CNB), support vector machine (SVM), K-nearest neighbor (KNN), Random forest (RF), and Artificial Neural Network(ANN)	RF	97.00%	2019

Table 2.1: Summery of the related works

2.7 Summary

From our study on the literature review, we can conclude that different types of feature selection methods and classification methods have been done on the various datasets of gene expression profile data. Still, there was no analysis of the comparison between feature selection methods for selecting informative genes. We will try to focus on this comparative study.

Chapter 3

Materials and Methods

When it is related to biological elements such as microarray, it is difficult to predict the result. We started by collecting two datasets to predict the classification and survival time. Then we extracted features from the dataset then find out the best n features. The different methods we used in our study will be discussed in this chapter to provide a clearer understanding of how they were used to conduct the experiments. We'll explain the datasets, system overview, feature selection method, classifier description, and performance assessment.

3.1 System Overview

To make a good predictor, we first collected two sets of data based on gene expression, extracted features from the data sets based on feature score, then shortlisted the top n feature to improve prediction accuracy, developed and selected effective classification algorithms to perform prediction, performed cross-validation tests to assess the accuracy of the predictor. The framework of our proposed model for grade classification, survival classification and survival prediction are shown in Figure 3.1, Figure 3.2 and Figure 3.3 respectively. Figure 3.1 illustrate the system overview for grade classification. From Figure 3.1, we can see that there are two phases in the framework for grade classification: training phase and testing phase. At first, each dataset randomly split into two parts for training and testing that is 80% and 20%, respectively. Then In the training Phase, synthetic minority upsampling (SMOTE) is applied to prevent class imbalances from affecting classifiers[46].

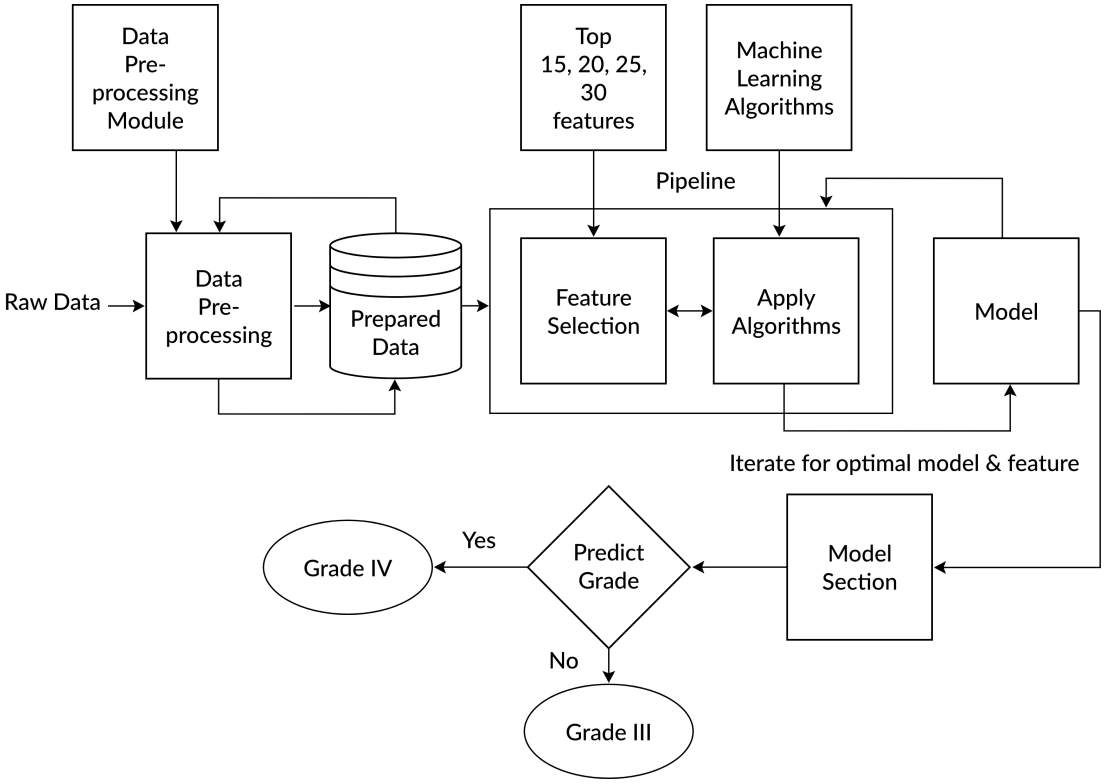


Figure 3.1: System flow diagram for grade classification.

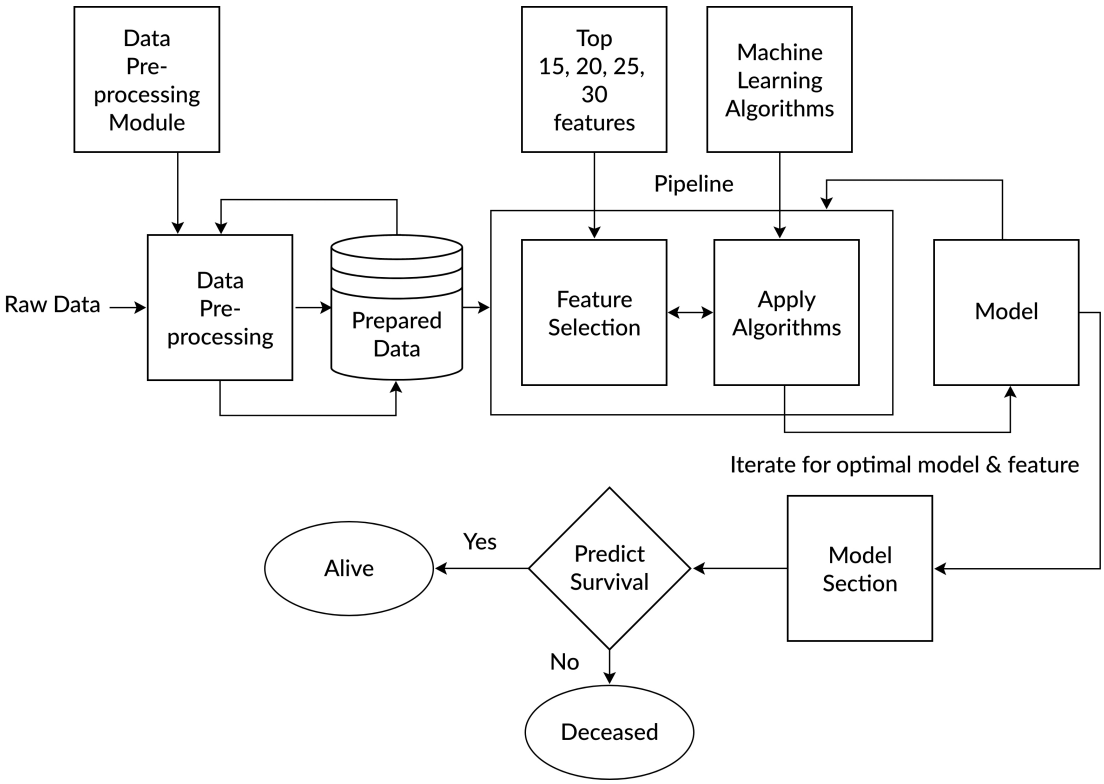


Figure 3.2: System flow diagram for survival classification.

Then data preprocessing step carried on for the training set. Next, in order to investigate and compare different feature selection methods, the prepared training datasets passed for feature extraction. Then, feature selection and classification are done at the same time. The model is trained using twelve well-known classifiers to find out the optimal model and features and stored later for prediction. In the testing phase: All the step is carried out for testing sample as like a training phase, except SMOTE. The testing sample is used to the previously saved model in the training phase to predict the classification. The model is evaluated using K-fold cross-validation, where the k-value is 3,5,7 and 10. The model evaluated based on the value of accuracy, sensitivity, specificity, recall, precision, F1-score, and area under the receiver operating characteristic.

Figure 3.2 illustrate the system overview for survival classification. Except synthetic minority upsampling (SMOTE), a similar scheme used for the survival classification task.

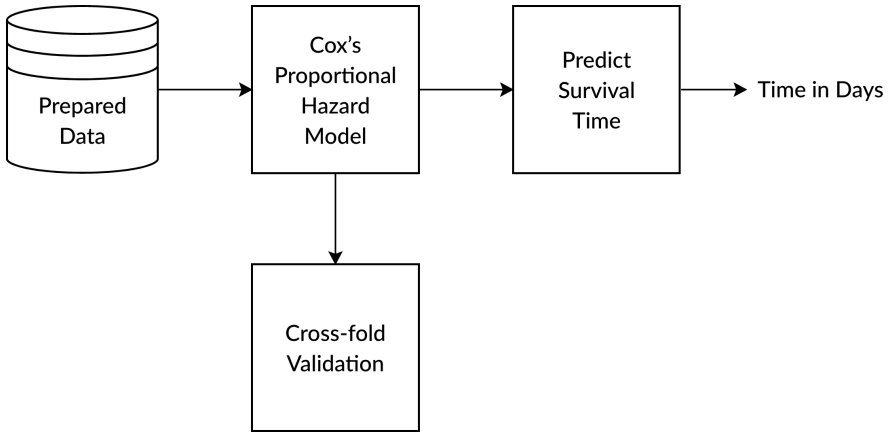


Figure 3.3: System flow diagram for survival time prediction.

Figure 3.3 illustrate the system overview for survival time prediction in days. At first, a dataset was prepared from survival classification. After that, the Kaplan-Meier estimator has been applied for the median survival time. Then, Cox's Proportional Hazard Model applied for individual prediction. The model was evaluated using cross-fold validation. The following sections describe the system architecture and experimental setup in detail.

3.2 Datasets

Glioma is a type of primary central nervous system (CNS) tumor that develops from glial cells[2]. The type of glioma can be classified by its grade. Low grades (grades I and II) of glioma are benign and high grades (grades III and IV) are malignant tumors. There are three types of glioma, such as Astrocytomas, Ependymomas, and Oligodendrogliomas. Astrocytomas tumors are known as anaplastic astrocytoma (AA) is grade III of astrocytoma, and glioblastoma multiforme (GBM) is grade IV. Oligodendrogliomas are responsible for grade III type glioma: Anaplastic oligodendroglioma (AO). Ependymomas make up less than 2% of all brain tumors.

Two glioma data sets of gene expression have been used in the experiments labeled as 'Freije'[20] and 'Phillips'[21], according to their author names. Both data sets are downloaded from GEO website (Gene Expression Omnibus) [47].

The Freije dataset includes 85 expression profiles for seven cases with a total of 11 repeat samples and comprises 26 grade III samples and 59 grade IV samples with a total of 22,280 features.

As like as Freije data set, the Phillips data has the same genes, but 98 grade III samples (23 samples) and grade IV samples (75 samples).

The two data sets Freije and Philips are summarized in Table 3.1.

Freije dataset constructed between 1996 and 2003. At that time, patients with primary brain tumors were undergone surgical treatment at the University of California. For the dataset, freshly frozen samples were obtained during the initial surgical treatment. There were 74 patients (46 women and 28 men), and their ages at diagnosis period varied from eighteen to eighty-two years. In this study, only grade III and grade IV gliomas were included for experiment. The dataset has been downloaded from http:/www.ncbi.nlm.nih.gov / geo / query / acc.cgi? AC = GSE4412.

In the Phillips dataset, at the time of initial surgical resection from patients with ages higher than twenty without prior therapy, fresh-frozen samples were obtained at the University of California, San Francisco, and M.D. Anderson Cancer Center. Dataset has been downloaded from http:/www.ncbi.nlm.nih.gov / geo / query / acc.cgi? AC = GSE4271.

Table 3.1: Summary of the information of Freije and Phillips datasets.

Dataset	Samples	Tumor Type	Samples	Features	Platform
Freije	85	Grade III	26	22283	U133A
		Grade IV	59		
Phillips	98	Grade III	23	22283	U133A
		Grade IV	75		

3.3 Data preprocessing

Data preprocessing is the first step of our system architecture. Sometimes, Data preprocessing is also known as data wrangling (e.g., data in CSV format, noise reduction, normalization, etc.), where data-wrangling refers to be a set of procedures which transforms raw data into a suitable environmental data format. Data generated using microarray technology has noisy nature, so before starting the experiment, we need to preprocess the data. At first, synthetic minority upsampling (SMOTE) is applied to balance the data. Min-max normalization is used to standardize data sets. Gene expression values are scaled to the point where the smallest amount for each gene becomes zero, and the largest value becomes one.[2].

3.4 Feature Selection

There are two phases in the methodology: (i) the feature (genes) selection that will be used for training and testing using Univariate Filters and (ii) applying different feature (genes) reduction technique to generate reduced features set. In order to investigate and compare different feature selection methods, we use the number ranging from 15 to 30 (15, 20, 25, 30).

3.5 Description of Classifier

After Feature Selection, we got a set of reduced features. Then, we applied twelve different classification methods: Extra Trees Classifier (ET), AdaBoost (AB), Bagging (BAG), Gradient Boosting (GB), Gaussian Naïve Bayes (NB), Support Vector Machines (SVM), Multi-Layer Perception (MLP), Logistic Regression (LR), K-nearest Neighbors

(KNN), Random Forrest (RF), Linear Discriminant Analysis (LDA), and Decision Trees (DT), and obtained the accuracy on the test set. + Classification outcomes are then used to compare the effectiveness of the various feature selection methods.

3.6 Performance Evaluation

Five-fold and Ten-fold cross-validation are used to evaluate the model. In each fold, feature selection is applied to the training set, and the reduced feature sets are tested on the testing set. The average accuracy, sensitivity, and specificity are reported. In the literature, many comparison matrices are used to compare the performance of different algorithms to predict. Some important metrics are accuracy (ACC), sensitivity (SNS), specificity (SPF), area under the precision-recall curve (auPR) and the area under the curve (AUC).

3.7 Summary

The main contributions of this thesis are primarily based on three topics. At first, to the best of our knowledge, this is the first comparative study on glioma data for gene expression of feature selection methods in machine-learning methods. Second, on two sets of data i.e. Freije and Phillips on gene expression, we applied a machine learning strategy that used the same genes for the same glioma tumor disease. We will also test the reliability of various processes. Third, we suggested a way of determining the survival period in days after selecting the genes.

Algorithm 4 Grade Classification Algorithm

Input: classifier, fsm, dimension
Output: grade type

1. $clf \leftarrow [DT, RF, BAG, BST, NB, MLP, SVM, LR, KNN, AB, LDA, ET]$
2. $fsm \leftarrow [KANOVA, PRCN, GUS, PCA, KPCA, ICA, FA]$
3. $dimension \leftarrow [15, 20, 25, 30]$
4. **function** PIPELINE($classifier, fsm, dimension$)
5. $predict = [\ ]$
6. $predict \leftarrow$ *Predict classifier accuracy with top n dimension features*
7. **return** $predict$
8. **end function**
9. **procedure** GRADECLASSIFICATION($classifier, fsm, dimension$)
10. System Initialization
11. $result = [\ ]$
12. **for** $classifier \leftarrow DT$ to ET **do**
13. **for** $fsm \leftarrow KANOVA$ to FA **do**
14. **for** $dimension \leftarrow 15$ to 30 **do**
15. $result \leftarrow$ PIPELINE*(*$classifier, fsm, dimension$*)*
16. **end for**
17. **end for**
18. **end for**
19. **return** $result$
20. **end procedure**

Algorithm 5 Survival Classification Algorithm

Input: classifier, fsm, dimension
Output: survival type

1. $clf \leftarrow [DT, RF, BAG, BST, NB, MLP, SVM, LR, KNN, AB, LDA, ET]$
2. $fsm \leftarrow [KANOVA, PRCN, GUS, PCA, KPCA, ICA, FA]$
3. $dimension \leftarrow [15, 20, 25, 30]$
4. **function** PIPELINE($classifier, fsm, dimension$)
5. $predict = [\,]$
6. $predict \leftarrow$ *Predict classifier accuracy with top n dimension features*
7. **return** $predict$
8. **end function**
9. **procedure** SURVIVALCLASSIFICATION($classifier, fsm, dimension$)
10. System Initialization
11. $result = [\,]$
12. **for** $classifier \leftarrow DT$ to ET **do**
13. **for** $fsm \leftarrow KANOVA$ to FA **do**
14. **for** $dimension \leftarrow 15$ to 30 **do**
15. $result \leftarrow$ PIPELINE*(*$classifier, fsm, dimension$*)*
16. **end for**
17. **end for**
18. **end for**
19. **return** $result$
20. **end procedure**

Algorithm 6 Survival Prediction Algorithm

Input: dataset, k=integer number

Output: survival time in days

1. $hazard_model \leftarrow [Coxś - model, Kaplan - Meier]$
2. **function** LIFELINE($dataset, hazard_model, k$)
3. $predict = [\ ]$
4. Use Hazards model to fit the dataset
5. $predict \leftarrow$ *predict the survival curve for the selected k entries*
6. **return** *predict*
7. **end function**
8. **procedure** SURVIVALPREDICTION($dataset, k$)
9. System Initialization
10. $result = [\ ]$
11. $result \leftarrow$ LIFELINE*(*$dataset, hazard_model, k$*)*
12. **return** *result*
13. **end procedure**

Chapter 4

Experimental Analysis

This chapter presents the results with experimental setup and metrics obtained from experiments. The experiments carried out to investigate and compare different dimensionality reduction and feature selection methods on grade classification, survival classification, and survival time prediction. All the experimental results are conducted here on two different datasets. Several modeling techniques are evaluated to arrive at a decision. The results and analysis are summarized in tabular form.

4.1 Experimental Setup

All analysis was performed using Python 3.7; the scikit-learn version is 0.22.2, and lifelines version is 0.24.3. For each dataset, the data was randomly divided into training and testing sets with train size = 0.8 and test size = 0.2 respectively. SMOTE is performed to prevent the sample imbalances. Then Min-Max normalization has been applied to the training set to normalize the data. Then, feature selection is applied to the training set to generate reduce feature sets. Reduced feature sets generated with a number ranging from 15 to 30 (feature_number = 15, 20, 25, 30). These sets of features evaluated using each of the twelve machine learning classifiers and training data to build a predictive machine learning model. The model was then evaluated on the test set by calculating the accuracy and the area under the receiver operating curve (AUC) score. There are three main experimental factors in our study which can affect the experiment: Feature selection method, Classification algorithms, and a number of features selected (feature_number = 15, 20, 25, 30). Five-fold and Ten-fold cross-validation are used to evaluate the model. Grade Classification and Survival Classification can be done using this model. We use another python library named as lifeline v0.24.3 to predict the survival time in days.

To find out machine learning approaches for glioma Grade Type Classification and Survival Classification, a sum of 2283 features were extracted. For the Grade datasets,

the output classes were Grade III and Grade IV; for the Survival datasets, the output classes were Living and Deceased while for the prediction of survival time is in days.

4.1.1 Parameter for Feature Selection Method

We investigate two types of feature selection techniques: (1) Univariate Feature Selection (2) Dimensions Reduction Technique. The parameter to use these feature selection method are shown below:

4.1.1.1 Univariate feature selection

Univariate feature selection operates by choosing the right traits based on univariate statistical tests. This can be regarded as a pre-processing step into an estimator. Univariate feature selection is three types: SelectKBest removes all but the k highest scoring features (denoted as K_ANOVA_F), SelectPercentile removes all but the user-specific highest score percentage of features (denoted as PRCN_ANOVA_F) and GenericUnivariateSelect (denoted as GUS_ANOVA_F) allows to perform univariate feature selection with a configurable strategy.

4.1.1.2 Dimensionality Reduction Methods

The following unsupervised dimensionality reduction methods were utilized in this study:

Principal Component Analysis (PCA): The following default hyperparameter settings are used for the experiment:

PCA(random_state=0, n_components=feature_number, whiten=False, copy=True, tol=0.0, svd_solver='auto', iterated_power='auto')[48]

Kernel Principal Component Analysis (KPCA): The following default hyperparameter settings are used for the experiment:

KernelPCA(kernel='linear',n_components=feature_number, coef0=1, degree=3, alpha=1.0, kernel_params=None, eigen_solver='auto', tol=0, fit_inverse_transform=False, max_iter=None, random_state=0, copy_X=True, remove_zero_eig=False, n_jobs=None, gamma=None)[49]

Independent Component Analysis (ICA): The following default hyperparameter settings are used for the experiment:

FastICA(algorithm='parallel', n_components=feature_number, fun='logcosh', whiten=True, tol=0.0001, max_iter=200, fun_args=None, random_state=0, w_init=None)[50]

Factor Analysis (FA): The following default hyperparameter settings are used for the experiment:

FactorAnalysis(n_components=feature_number, tol=0.01, copy=True, noise_variance_init=None, svd_method='randomized', random_state=0, iterated_power=3)[51]

4.1.2 Parameter for Classifier

Decision Tree (DT): The following default hyperparameter settings are used for the experiment:

DT(splitter='best', criterion='gini', max_depth=None, min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, random_state=0, max_leaf_nodes=None, min_impurity_split=None, min_impurity_decrease=0.0, class_weight=None, ccp_alpha=0.0, presort='deprecated, max_features=None)[52]

Random Forrest (RF): The following default hyperparameter settings are used for the experiment:

RF(criterion='gini', max_depth=None, n_estimators=100, min_samples_split=2, min_samples_leaf=1, max_features='auto', min_weight_fraction_leaf=0.0, max_leaf_nodes=None, min_impurity_split=None, min_impurity_decrease=0.0, bootstrap=True, oob_score=False, random_state=0, class_weight=None, n_jobs=None, verbose=0, ccp_alpha=0.0, warm_start=False, max_samples=None)[53]

Bagging (BAG): The following default hyperparameter settings are used for the experiment:

BAG(n_estimators=10, max_samples=1.0, base_estimator=None, max_features=1.0, bootstrap=True, oob_score=False, bootstrap_features=False, n_jobs=None, warm_start=False, verbose=0, random_state=0)[54]

Gradient Boosting (GB): The following default hyperparameter are used for the experiment:

GB(learning_rate=0.1, loss='deviance', n_estimators=100, criterion='friedman_mse', subsample=1.0, tol=0.0001, min_samples_split=2, min_weight_fraction_leaf=0.0, min_samples_leaf=1, max_depth=3, min_impurity_split=None, min_impurity_decrease=0.0, init=None, random_state=0, verbose=0, max_features=None, max_leaf_nodes=None, presort='deprecated', warm_start=False, n_iter_no_change=None, validation_fraction=0.1, ccp_alpha=0.0)[55]

Gaussian Naïve Bayes (NB): The following default hyperparameter settings are used for the experiment:

NB(var_smoothing=1e-09, priors=None)[56]

Multi-Layer Perceptron (MLP): The following default hyperparameter settings are used for the experiment:

MLP(activation='relu', hidden_layer_sizes=(100,), solver='adam', batch_size='auto', alpha=0.0001, learning_rate_init=0.001, learning_rate='constant', max_iter=200, power_t=0.5, random_state=0, shuffle=True, tol=0.0001, warm_start=False, verbose=False, momentum=0.9, early_stopping=False, nesterovs_momentum=True, validation_fraction=0.1, beta_2=0.999, beta_1=0.9, epsilon=1e-08, max_fun=15000, n_iter_no_change=10,)[57]

Support Vector Machines (SVM): The following default hyperparameter settings are used for the experiment:

SVM(kernel='linear', C=1.0, degree=3, coef0=0.0, gamma='scale', shrinking=True, tol=0.001, probability=False, cache_size=200, verbose=False, class_weight=None, max_iter=-1, break_ties=False, decision_function_shape='ovr', random_state=0)[58]

Logistic Regression (LR): The following default hyperparameter settings are used for the experiment:

LR(dual=False, penalty='l2', C=1.0, tol=0.0001, intercept_scaling=1, fit_intercept=True, class_weight=None, solver='lbfgs', random_state=0, max_iter=100, verbose=0, warm_start=False, l1_ratio=None, multi_class='auto', n_jobs=None)[59]

K-nearest Neighbors (KNN): The following default hyperparameter settings are used for the experiment:

KNN(weights='uniform', n_neighbors=5, leaf_size=30, algorithm='auto', metric='minkowski', p=2, **kwargs, n_jobs=None, metric_params=None)[60]

AdaBoost: The following default hyperparameter settings are used for the experiment:

AB(n_estimators=50, base_estimator=None, learning_rate=1.0, random_state=0, algorithm='SAMME.R')[61]

Linear Discriminant Analysis: The following default hyperparameter settings are used for the experiment:

LDA(shrinkage=None, solver='svd', n_components=None, priors=None,tol=0.0001, store_covariance=False,)[62]

Extra Tree Classifier: The following default hyperparameter settings are used for the experiment:

ET(criterion='gini', max_depth=None, n_estimators=100, min_samples_split=2, min_samples_leaf=1, min_impurity_decrease=0.0, min_weight_fraction_leaf=0.0, max_features='auto', ccp_alpha=0.0, min_impurity_split=None, oob_score=False, n_jobs=None, random_state=None, max_leaf_nodes=None, verbose=0, max_samples=None, bootstrap=False, warm_start=False, class_weight=None)[63]

4.1.3 Parameter for Survival Time Prediction

Kaplan Meier Estimator: The Kaplan-Meier method estimates survival probability from observed survival times. It results in a step function that changes value only at the time of each event, and confidence intervals can be computed for the survival probabilities. A valuable overview of the data is given by the KM survival curve, which is a plot of KM survival probability over time. This can be used for calculating factors like median survival time. It CANNOT account for risk factors and is NOT regression. It is non-parametric (does not involve parameters). However, it is a excellent way to visualize a survival dataset and can be useful to compare the effects of a single categorical variable.

4.2 Results

In our study, there are three experimental factors: (1) Grade Classification (II) Survival Classification (III) Survival Time Prediction. Grade classification and Survival classification has three stages: Applying Feature selection method (Univariate Feature selection, Principal Component Analysis (PCA), Kernel PCA, ICA, FA), Applying classifier method such as Decision Tree (DT), Random Forest (RF), etc., and evaluate the performance based on selected feature_number are 15, 20, 25, 30. For survival time prediction, our experiment has two stages: (i) Estimator with Hazard model and (ii) Model evaluation based on K-fold cross validation. Cross-validation is a technique in which we train our model using the subset of the data-set and then evaluate using the complementary subset of the data-set. The purpose of cross validation is to assess how our prediction model performs with an unknown dataset. This technique involves randomly dividing the dataset into k groups or folds of approximately equal size. The first fold is kept for testing and the model is trained on k-1 folds. The process is repeated k times and each time different fold or a different group of data points are used for validation where k can be 1,2,3,4....n. From our literature review we found that Heba Abusamra[2] and Pirooznia et al.[24] both work on microarray gene expression data, whereas Heba Abusamra[2] used both our datasets. Heba Abusamra[2] evaluated predictive model performance using five-fold cross validation and Pirooznia et al.[24] evaluated using ten-fold cross validation. Using five-fold and ten-fold has its own advantage,

computation time is reduced as we repeated the process only 5 or 10 times when the value of k is 5 or 10. Every data points get to be tested exactly once and is used in training k-1 times. As the variance of the resulting estimate is reduced as k increases, for experimental purpose we use 3-fold and 7-fold validation to find out the more accurate accuracy. Kaplan Meier Estimator and Cox's Proportional Hazards Model used in this experiment. For grade classification: the output class will be Grade III and Grade IV, for survival classification: the output class will be Living and Deceased and for survival time prediction output result will be in days. For both classification tasks, feature selection and classification performance were done using the testing test, whereas the training set was used for model building and training. For survival time prediction a reduced feature set used for evaluation.

4.2.1 Grade Classification

At first, Twelve different classification methods were applied to both datasets with five different feature selection method (feature_number are 15, 20, 25, 30) for Grade classification. The performance was evaluated using 5-fold cross validation and 10-fold cross validation. comparison between different feature selection method has been reported below:

4.2.1.1 Results on the Freije data set

For Top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.1 for 5-fold cross-validation. From the results shown in the figure, we notice that the highest accuracy is achieved by AdaBoost (AB) using Univariate feature selection (K_ANOVA_F), which is 96.25%. Decision Tree (DT) and Gradient Boosting (GB) using Univariate feature selection (K_ANOVA_F) have the same accuracy 95%. Logistic Regression (LR) and SVM classifier had better results after performing the Factor Analysis (FA) method. Factor Analysis (FA) method had 93.57% accuracy score. While Decision Tree (DT) with Kernel PCA has the worst result of 73.21%. Table S4 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.2 for 10-fold cross-validation. From the results shown in the figure, we notice that the highest accuracy is achieved by AdaBoost (AB) using

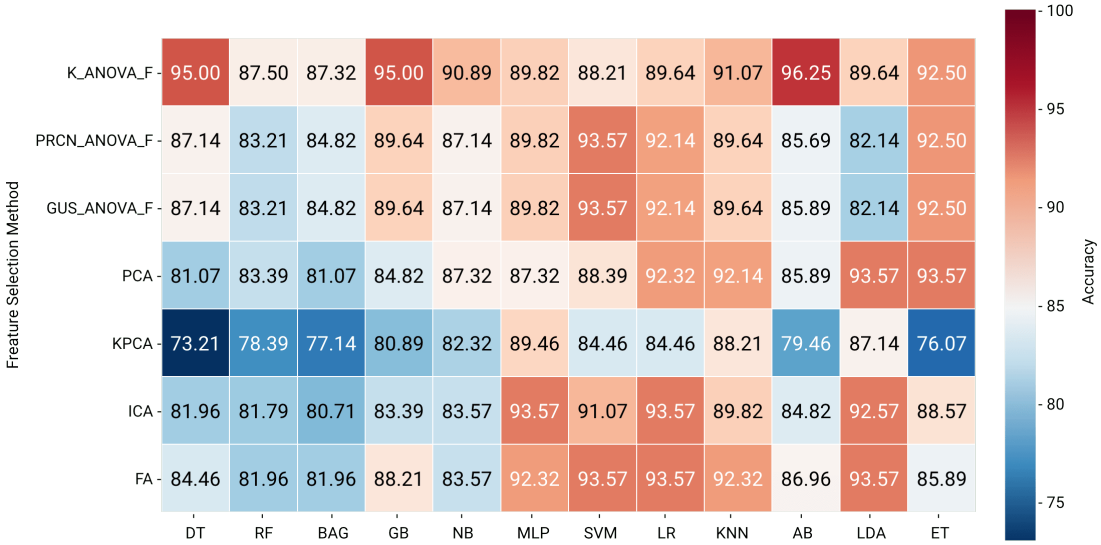


Figure 4.1: Grade Classification for Freije: Accuracy heatmap of 5-fold for top 15 features.

Univariate feature selection (K_ANOVA_F), which is 98.75%. SVM and Logistic Regression (LR) using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) have the same accuracy 95%. MLP has better results after performing the Independent Component Analysis (ICA) method. Independent Component Analysis (ICA) method has 95% accuracy too. While Decision Tree (DT) with Kernel PCA has the worst result of 74.17%. Table S6 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For top 15 features at grade classification, the best results among the feature selection methods are achieved by Univariate feature selection. In contrast, the worst results are often made by Kernel PCA. Moreover, Factor analysis (FA) has comparable results to Univariate feature selection, which usually has excellent predictive performance. In terms of classifiers, AB, SVM, and Logistic Regression (LR) classifier methods show similar results except for Decision Tree (DT), which is noticeably lower. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For top 20 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.3 for 5-fold cross-validation. From the results shown in the figure, we notice that the highest accuracy is achieved by AdaBoost (AB) and



Figure 4.2: Grade Classification for Freije: Accuracy heatmap of 10-fold for top 15 features.

Gradient Boosting (GB) using Univariate feature selection (K_ANOVA_F) which is 96.25%. Decision Tree (DT) using Univariate feature selection (K_ANOVA_F) has the same accuracy 95%. MLP classifier had better results after performing the Factor Analysis (FA) method. Factor Analysis (FA) method had a 94.82% accuracy score, while Naïve Bayes (NB) with Factor Analysis (FA) has the worst result of 66.79%. Table S8 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 20 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.4 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by AdaBoost (AB) using Univariate feature selection (K_ANOVA_F), which is 96.25%. SVM using Univariate feature selection (PRCN_ANOVA_F) has the same accuracy 96.25%. Logistic Regression (LR) has better results after performing the Univariate feature selection (GUS_ANOVA_F). The accuracy of Univariate feature selection (GUS_ANOVA_F) is 95%, while Decision Tree (DT) with Independent Component Analysis (ICA) has the worst result of 66.67%. Table S10 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 20 features at grade classification, the best results among the feature

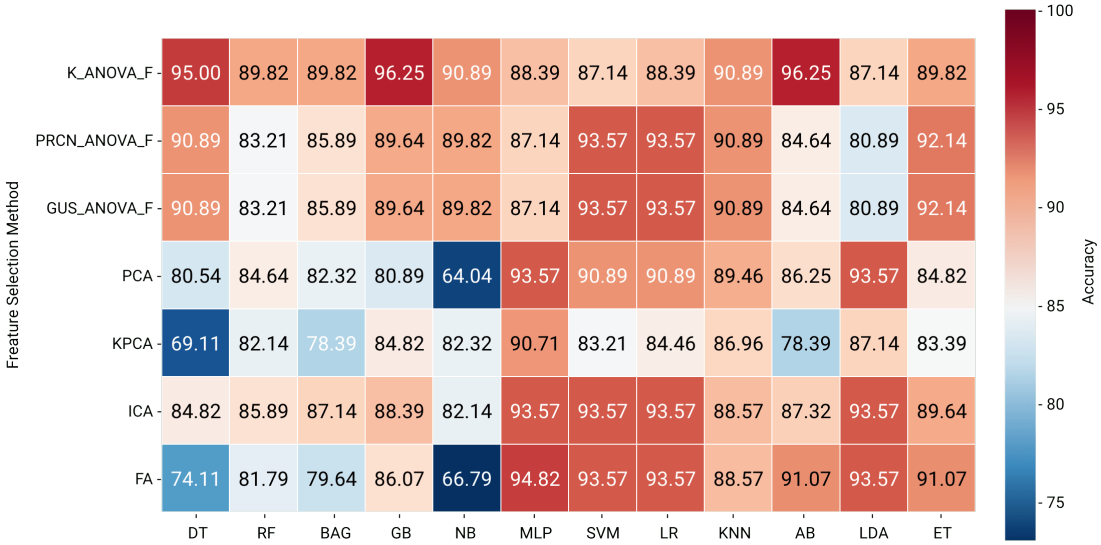


Figure 4.3: Grade Classification for Freije: Accuracy heatmap of 5-fold for top 20 features.

selection methods are achieved by Univariate feature selection, while ICA often obtains the worst results. Moreover, Factor analysis (FA) has comparable results to Univariate feature selection, which usually has good predictive performance. In terms of classifiers, AB, SVM, DT, and Logistic Regression (LR) classifier methods show similar results except for Naïve Bayes (DT), which is noticeably lower. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.5 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Decision Tree (DT) using Univariate feature selection (K_ANOVA_F), which is 97.5%. Gradient Boosting (GB) and AdaBoost (AB) using Univariate feature selection (K_ANOVA_F) have the same accuracy 96.25%. MLP classifier had better results after performing the Factor Analysis (FA) method. Factor Analysis (FA) method had 94.82% accuracy score. While Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 61.79%. Table S12 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.6 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by AdaBoost (AB) using

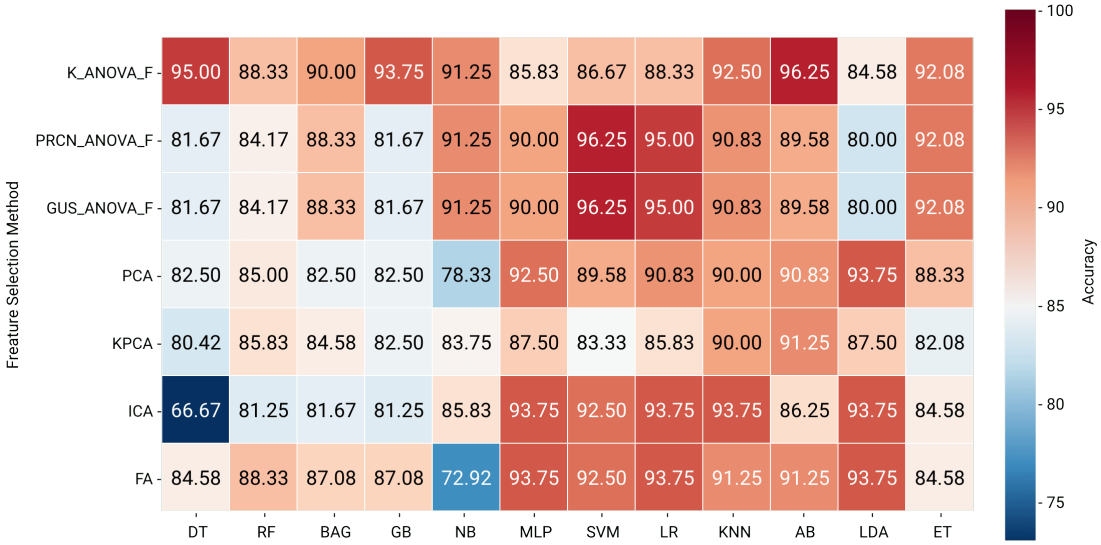


Figure 4.4: Grade Classification for Freije: Accuracy heatmap of 10-fold for top 20 features.

Univariate feature selection (K_ANOVA_F), which is 98.75%. Logistic Regression (LR) and SVM using Factor analysis have the same accuracy 95%; also SVM with Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) and Decision Tree (DT) with Univariate feature selection (K_ANOVA_F) performs same. Linear Discriminant Analysis (LDA), Logistic Regression (LR), and MLP have better results after performing Independent Component Analysis (ICA) and Factor analysis (FA). The accuracy of this experiment is 93.75%. While Principal Component Analysis (PCA) and Factor analysis (FA) with Naïve Bayes (NB) classifier have the worst result 66.67%. Table S14 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 25 features at grade classification, the best results among the feature selection methods is achieved by Univariate feature selection, while the worst results are often achieved by Principal Component Analysis (PCA) and sometimes Independent Component Analysis (ICA). Moreover, FA and Independent Component Analysis (ICA) has comparable results to Univariate feature selection, which often has the good predictive performance. In terms of classifiers, AB, DT, Logistic Regression (LR) and sometimes LDA classifier methods show similar results with the exception of Naïve Bayes (DT) which is noticeably lower. Also, we perform 3-fold and

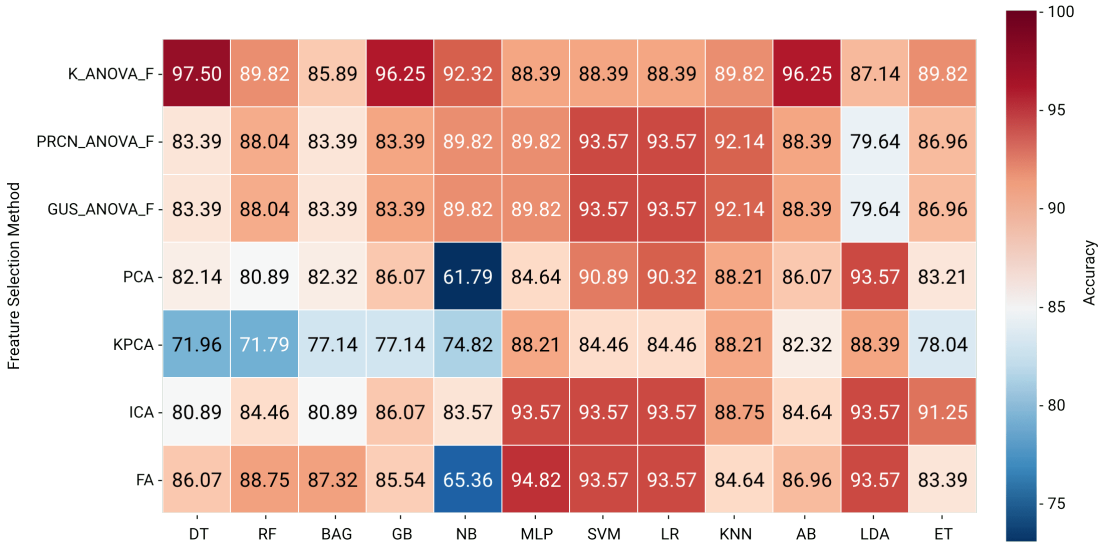


Figure 4.5: Grade Classification for Freije: Accuracy heatmap of 5-fold for top 25 features.

7-fold cross-validation which can be found in Supplementary B.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.7 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Decision Tree (DT) using Univariate feature selection (K_ANOVA_F), which is 97.5%. Gradient Boosting (GB) and AdaBoost (AB) using Univariate feature selection (K_ANOVA_F) have the same accuracy 96.25%. MLP and SVM classifiers had better results after performing the Factor Analysis (FA) method. Factor Analysis (FA) method had a 94.82% accuracy score, while Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result 59.29%. Table S16 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.8 for 10-fold cross-validation. From the results shown in the figure, we notice that the highest accuracy is achieved by AdaBoost (AB) using Univariate feature selection (K_ANOVA_F), which is 96.25%. However, Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) with SVM classifier has the same accuracy. MLP and Logistic Regression (LR) using Factor analysis have the same accuracy 95%. SVM has better results after performing Independent Component Analysis (ICA) and Factor analysis (FA). The accuracy of this

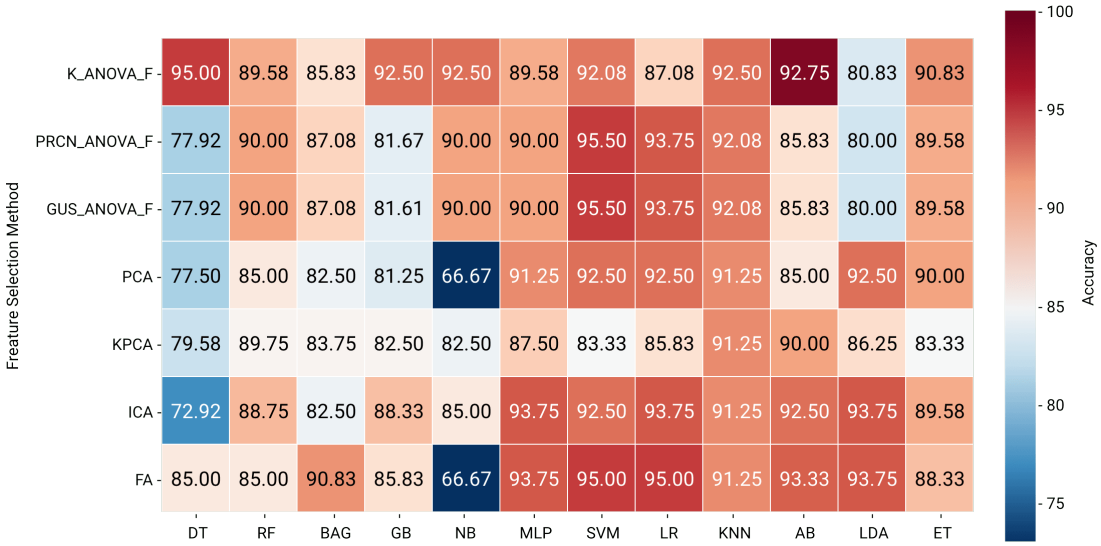


Figure 4.6: Grade Classification for Freije: Accuracy heatmap of 10-fold for top 25 features.

experiment is 93.75%. Also, MLP with Independent Component Analysis (ICA), Logistic Regression (LR) with Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) and Principal Component Analysis (PCA), Naïve Bayes (NB), Gradient Boosting (GB), and Decision Tree (DT) with Univariate feature selection (K_ANOVA_F) has the same accuracy 93.75% while Factor analysis (FA) with Naïve Bayes (NB) classifier has the worst result 60.42%. Table S18 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 30 features at grade classification, the best results among the feature selection methods are achieved by Univariate feature selection; however the worst results are often achieved by FA and sometimes PCA. Moreover, FA and PCA have comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, AdaBoost (AB) and SVM classifier methods show similar results except for Naïve Bayes (DT), which is noticeably lower. But, the problem is Factor analysis (FA) performance has a good and bad impact. So there is no clear decision fro this experiment. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

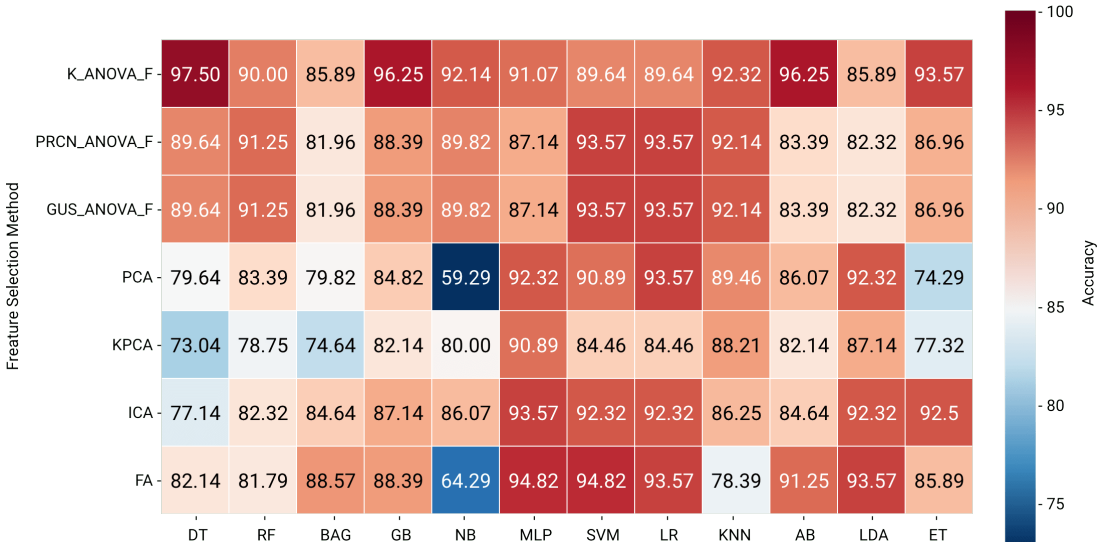


Figure 4.7: Grade Classification for Freije: Accuracy heatmap of 5-fold for top 30 features.

4.2.1.2 Results on the Phillips data set

For Top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.9 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by AdaBoost (AB) using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 94.27%. MLP using Univariate feature selection (K_ANOVA_F) has the same accuracy 94.27%. AdaBoost (AB) and Extra Trees (ET) classifier had better results 93.45% after performing Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F). Factor Analysis (FA) method had a 91.55% accuracy score, which is a good result too. While Decision Tree (DT) with Independent Component Analysis (ICA) has the worst outcome of 82.55%. Table S20 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.10 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Random Forest (RF) classifier using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) which is 95.5%. However, Univariate feature selection (K_ANOVA_F, PRCN_ANOVA_F, GUS_ANOVA_F) with Naïve Bayes (NB) classifier has 93.67% accuracy. AdaBoost (AB) using Factor analysis (FA) has the same precision, 93.5%.

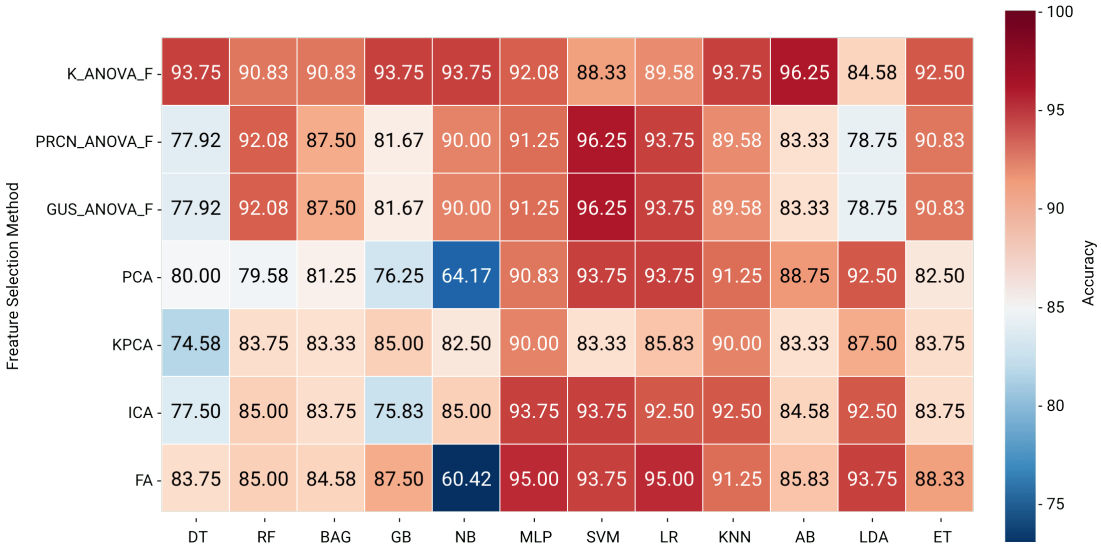


Figure 4.8: Grade Classification for Freije: Accuracy heatmap of 10-fold for top 30 features.

Extra Trees (ET) classifier has better results after performing Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F). The accuracy of this experiment is 92.83%. Also, Logistic Regression (LR) and Linear Discriminant Analysis (LDA) with Independent Component Analysis (ICA) have the same precision, 92.5%. While Univariate feature selection (K_ANOVA_F) with Decision Tree (DT) classifier has the worst result 81.67%. Table S22 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 15 features at grade classification, the best results among the feature selection methods are achieved by Univariate feature selection, while ICA often produces the worst results. Moreover, Principal Component Analysis (PCA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, AdaBoost (AB) performs well on 5-fold, but Random Forest (RF) classifier performs well on 10-fold. AdaBoost (AB) and Decision Tree (DT), which performance noticeably lower. The problem is AdaBoost (AB) perform well on 5-Fold and bad on 10-fold. So there is no clear decision in terms of the classifier from this experiment. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 20 features, the accuracy of different classifiers with feature selection

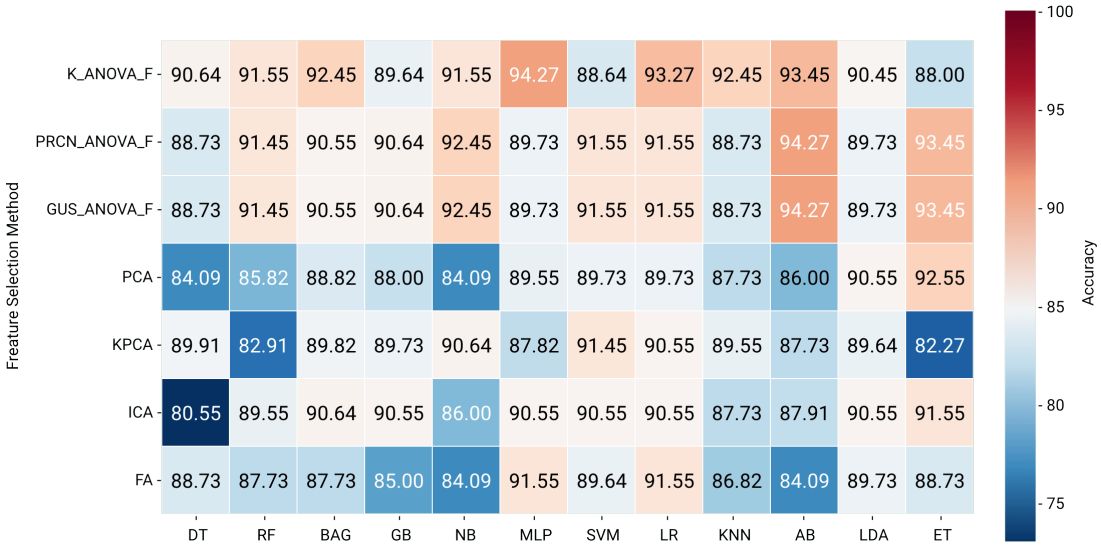


Figure 4.9: Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 15 features.

methods are shown in Figure 4.11 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Bagging (BAG) using Univariate feature selection (K_ANOVA_F), which is 94.36%. Bagging (BAG) using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) has the same accuracy 93.36%. Random Forest (RF) classifier had better results 93.27% after performing Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F). Factor Analysis (FA) method had a 92.45% accuracy score, while Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 79.55%. Table S24 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 20 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.12 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Extra Trees (ET) classifier using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) which is 95.67%, though Univariate feature selection (K_ANOVA_F, PRCN_ANOVA_F, GUS_ANOVA_F) with Naïve Bayes (NB) classifier has 93.67% accuracy. MLP using Univariate feature selection (K_ANOVA_F) has an accuracy of 92.83%. SVM classifier has better results after performing Principal Component

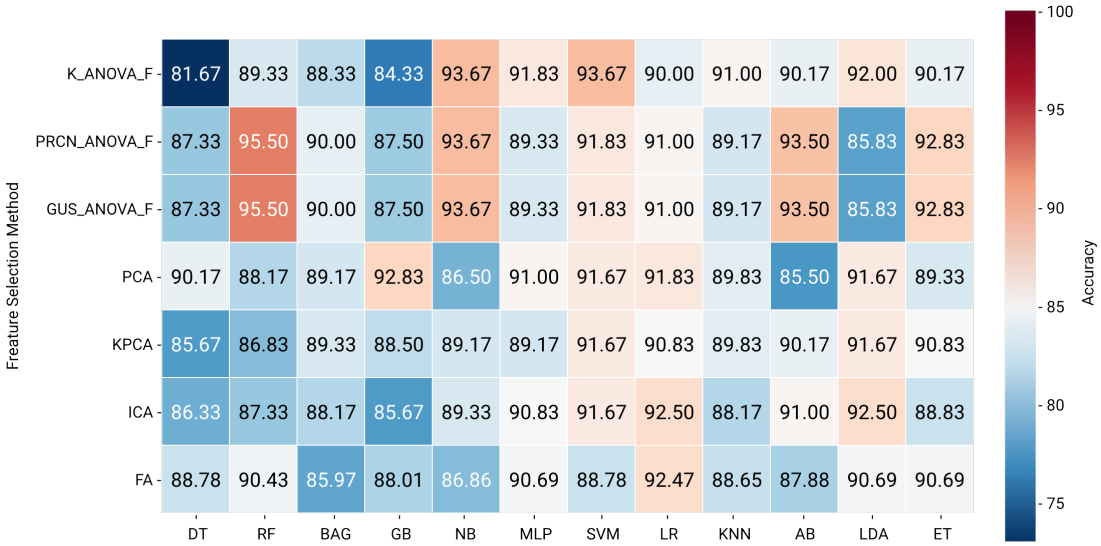


Figure 4.10: Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 15 features.

Analysis (PCA). The accuracy of this experiment is 92.17%. Factor Analysis (FA) has god predictive performance, which is 92% while Factor Analysis (FA) with Decision Tree (DT) classifier has the worst result 84.33%. Table S26 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 20 features at grade classification, the best results among the feature selection methods are achieved by Univariate feature selection, while Principal Component Analysis (PCA) and Factor analysis (FA) often achieve the worst results. Moreover, Factor analysis (FA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, Extra Trees (ET) and Bagging (BAG) classifier methods show similar results except for Decision Tree (DT), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.13 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Extra Trees (ET) using Independent Component Analysis (ICA), which is 94.36%. Random Forest (RF)

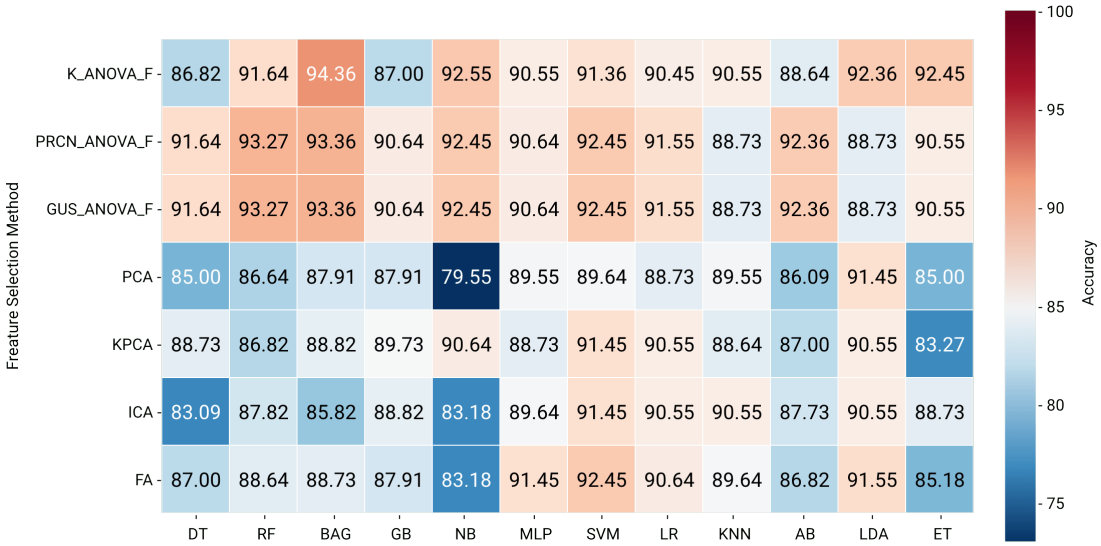


Figure 4.11: Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 20 features.

using Univariate feature selection (GUS_ANOVA_F) has an accuracy of 94.27%. Gradient Boosting (GB) classifier has better results 93.64% after performing Independent Component Analysis (ICA). Independent Component Analysis (ICA) method had a 92.64% accuracy score. While Naïve Bayes (NB) with Factor Analysis (FA) has the worst result of 75.45%. Table S28 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.14 for 10-fold cross-validation. From the results shown in the figure, we notice that the highest accuracy is achieved by Random Forest (RF) classifier using Univariate feature selection (PRCN_ANOVA_F), which is 95.5%, though Univariate feature selection (K_ANOVA_F) with Extra Trees (ET) classifier has 94.5% accuracy. MLP using Univariate feature selection (K_ANOVA_F), Extra Trees (ET), Gradient Boosting (GB) with Independent Component Analysis (ICA) has the accuracy 93.67%. Naïve Bayes (NB) classifier has better results after performing Univariate feature selection (K_ANOVA_F). The accuracy of this experiment is 92.83%. Independent Component Analysis (ICA) has a good predictive performance, which is 92%. While Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 75.17%. Table S30 at supplementary shows

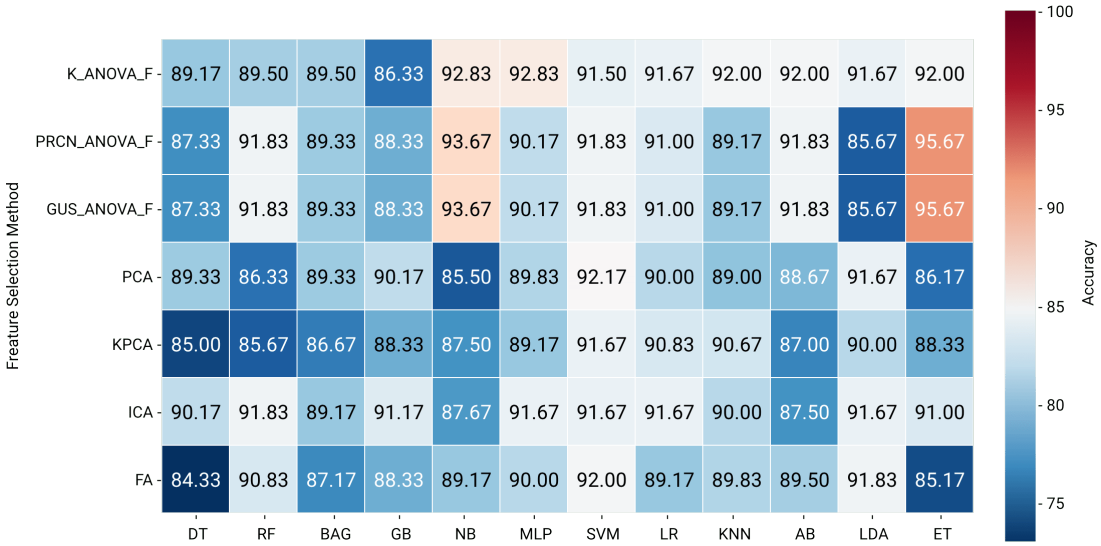


Figure 4.12: Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 20 features.

the results of cross-fold validation for different classifiers with feature selection methods.

For the top 25 features at grade classification, the best results among the feature selection methods are achieved by Univariate feature selection, while PCA often produces the worst results. Moreover, Independent Component Analysis (ICA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, Extra Trees (ET) and Random Forest (RF) classifier methods show similar results except for Naïve Bayes (NB), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.15 for 5-fold cross-validation. From the results shown in the figure, we notice that the highest accuracy is achieved by Extra Trees (ET) classifier using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) which is 94.27%, though Independent Component Analysis (ICA) with Gradient Boosting (GB) classifier has 93.64% accuracy. The performance of both classifiers almost the same. MLP using Univariate feature selection has an accuracy of 93.45%.

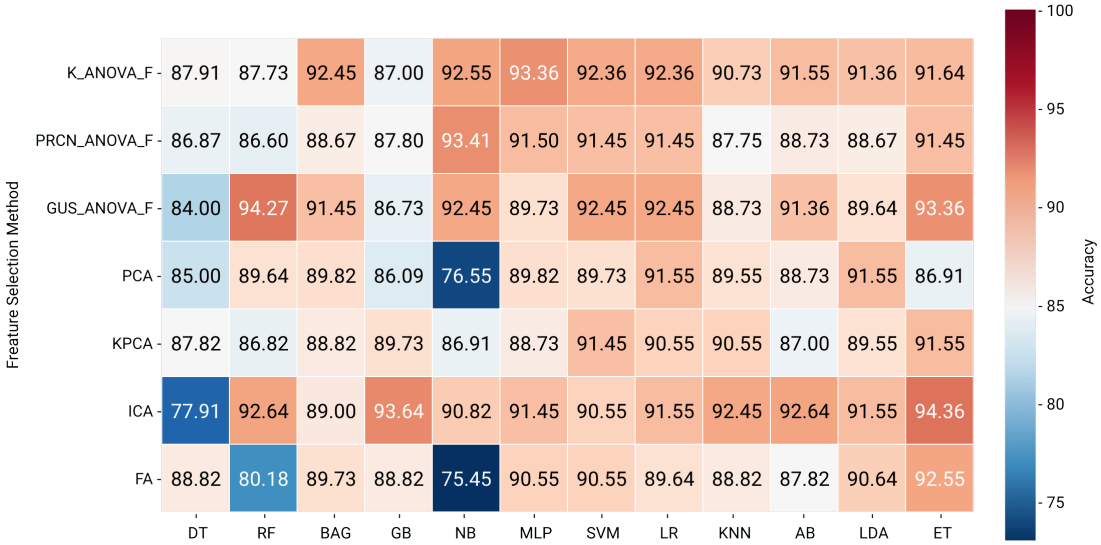


Figure 4.13: Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 25 features.

Logistic Regression (LR) classifier has better results after performing Univariate feature selection (K_ANOVA_F). The accuracy of this experiment is 93.36%. Independent Component Analysis (ICA) with Gradient Boosting (GB) and Bagging (BAG) has a good predictive performance, which is 93.55%, while Naïve Bayes (NB) with Factor analysis (FA) has the worst result 75.17%. Table S32 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.16 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Extra Trees (ET) classifier using Independent Component Analysis (ICA), which is 95.67%. Random Forest (RF) using Univariate feature selection (K_ANOVA_F) has an accuracy of 94.5%. MLP and Extra Trees (ET) classifiers have better results after performing Univariate feature selection (K_ANOVA_F). The accuracy of this experiment is 93.67%. Independent Component Analysis (ICA) with K Nearest Neighbor (KNN) has a good predictive performance, which is 92.67% while Naïve Bayes (NB) with Factor analysis (FA) has the worst result 71.5%. Table S34 at supplementary shows the results of cross-fold validation for different classifiers with feature selection

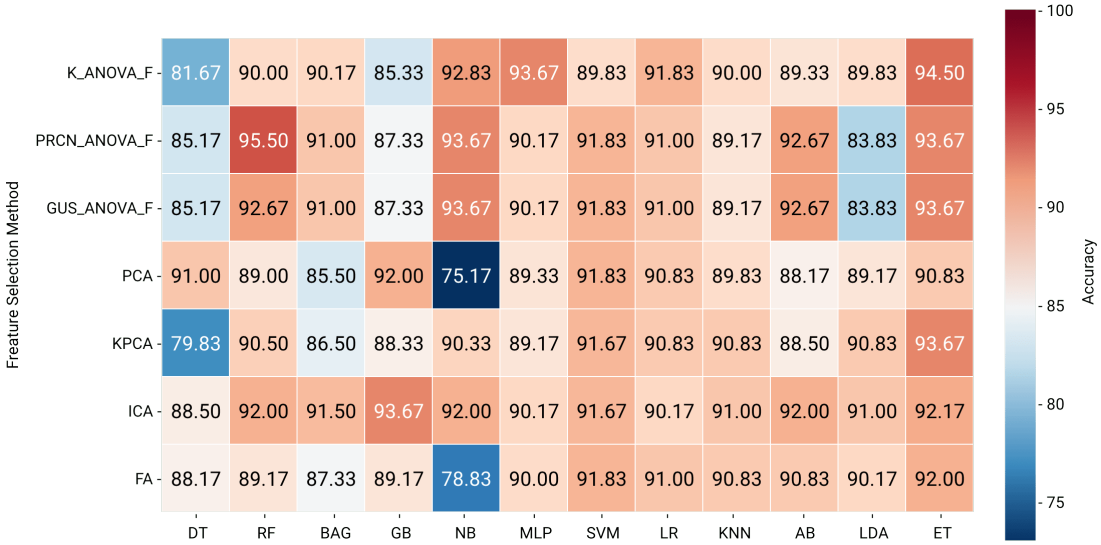


Figure 4.14: Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 25 features.

methods.

For the top 30 features at grade classification, the best results among the feature selection methods are achieved by Univariate feature selection. In contrast, the worst results are often achieved by Univariate feature selection. Moreover, Independent Component Analysis (ICA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, Extra Trees (ET) classifier methods show the best results except for Naïve Bayes (NB), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

4.2.2 Survival Classification

At first, Twelve different classification methods were applied to both datasets with five different feature selection methods (feature_number are 15, 20, 25, 30) for Survival classification. The performance was evaluated using 5-fold cross-validation and 10-fold cross-validation. Comparison between different feature selection method has been reported below:

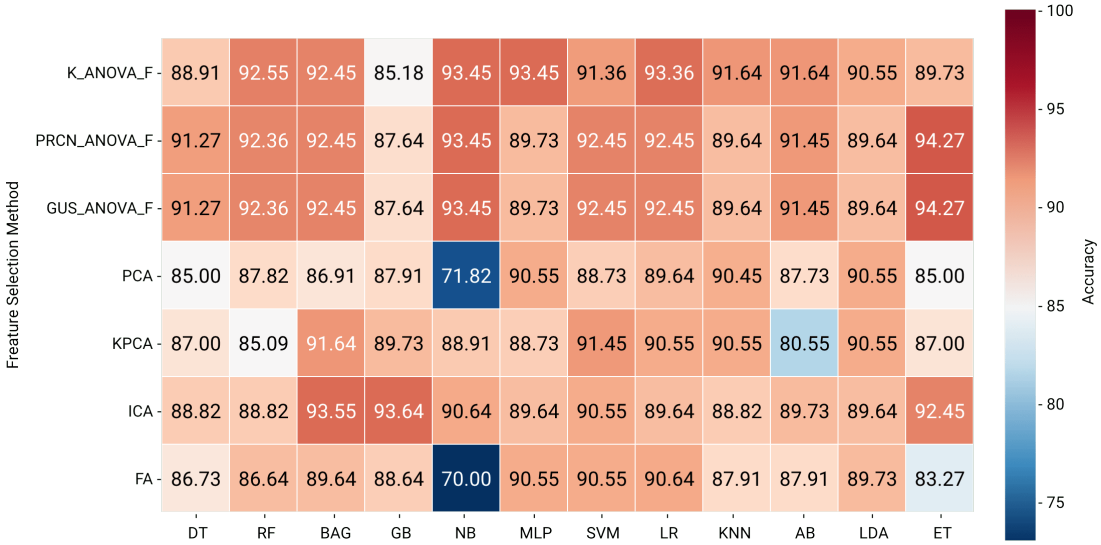


Figure 4.15: Grade Classification for Phillips: Accuracy heatmap of 5-fold for top 30 features.

4.2.2.1 Results on the Freije data set

For Top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.17 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Logistic Regression (LR) and SVM classifier using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 93.19%. Linear Discriminant Analysis (LDA) using Kernel PCA has an accuracy of 92.08%. MLP classifier has better results after performing Kernel PCA. The accuracy of this experiment is 90.69%. Independent Component Analysis (ICA) with MLP has a good predictive performance, which is 88.47%. While Naïve Bayes (NB) with Principal Component Analysis (PCA) classifier has the worst result of 73.06%. Table S36 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.18 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by MLP classifier using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 92.25%. Logistic Regression (LR) and SVM using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) have the accuracy 92%. MLP has better results

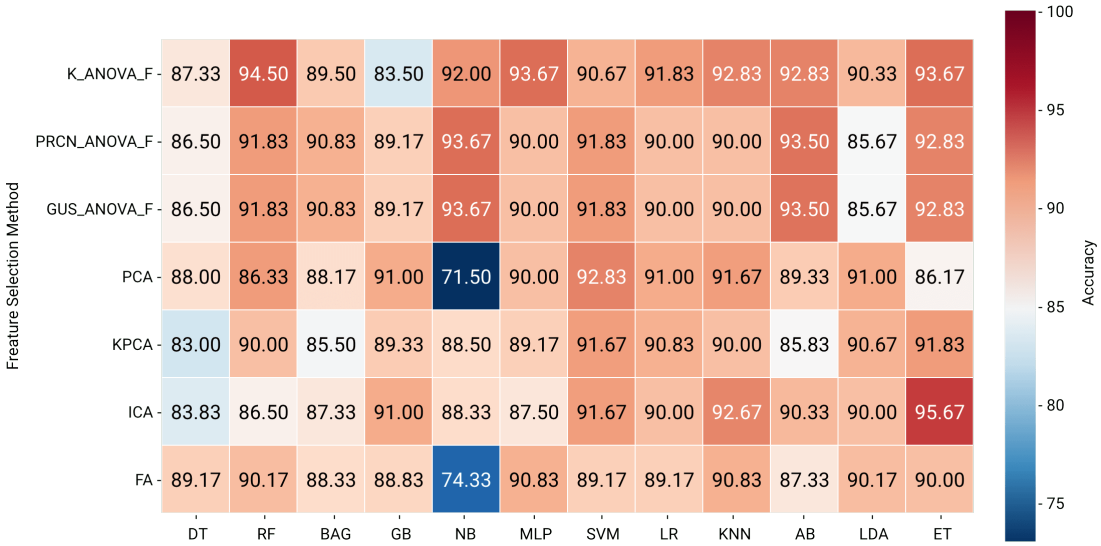


Figure 4.16: Grade Classification for Phillips: Accuracy heatmap of 10-fold for top 30 features.

after performing Kernel PCA. The accuracy of this experiment is 91.75%. Independent Component Analysis (ICA) with Extra Trees (ET) has a good predictive performance, which is 90%. While Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 74.25%. Table S38 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 15 features at survival classification, the best results among the feature selection methods are achieved by Univariate feature selection, while PCA often produces the worst results. Moreover, Kernel PCA has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, SVM, MLP and Logistic Regression (LR) classifier methods show the best results except for Naïve Bayes (NB), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in y Supplementary B.

For Top 20 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.19 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Logistic Regression (LR) and SVM classifier using Univariate feature selection (PRCN_ANOVA_F,

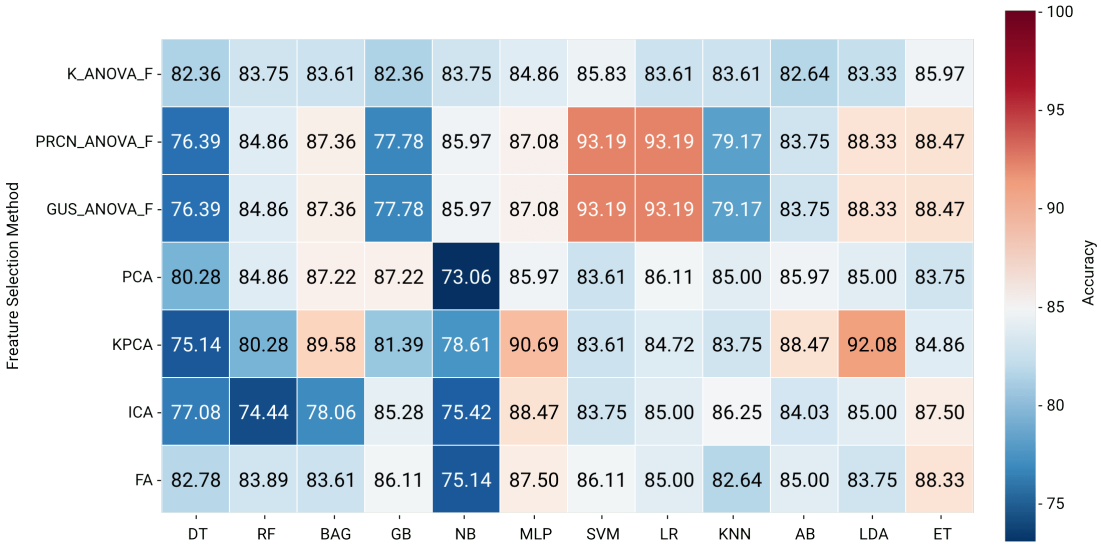


Figure 4.17: Survival Classification for Freije: Accuracy heatmap of 5-fold for top 15 features.

GUS_ANOVA_F), which is 93.19%. SVM using Factor analysis (FA) has an accuracy of 90.97%. MLP classifier has better results after performing Principal Component Analysis (PCA). The accuracy of this experiment is 90.69%. Independent Component Analysis (ICA) with MLP has a good predictive performance, which is 89.72%, while Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 57.36%. Table S40 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 20 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.20 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the SVM classifier using Principal Component Analysis (PCA), which is 94.5%. Logistic Regression (LR) and SVM using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F) have the accuracy 92%. Logistic Regression (LR) has better results after performing Principal Component Analysis (PCA). The accuracy of this experiment is 91.75%. Kernel PCA with Linear Discriminant Analysis (LDA) has a good predictive performance, which is 91%, while Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 63.5%. Table S42 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

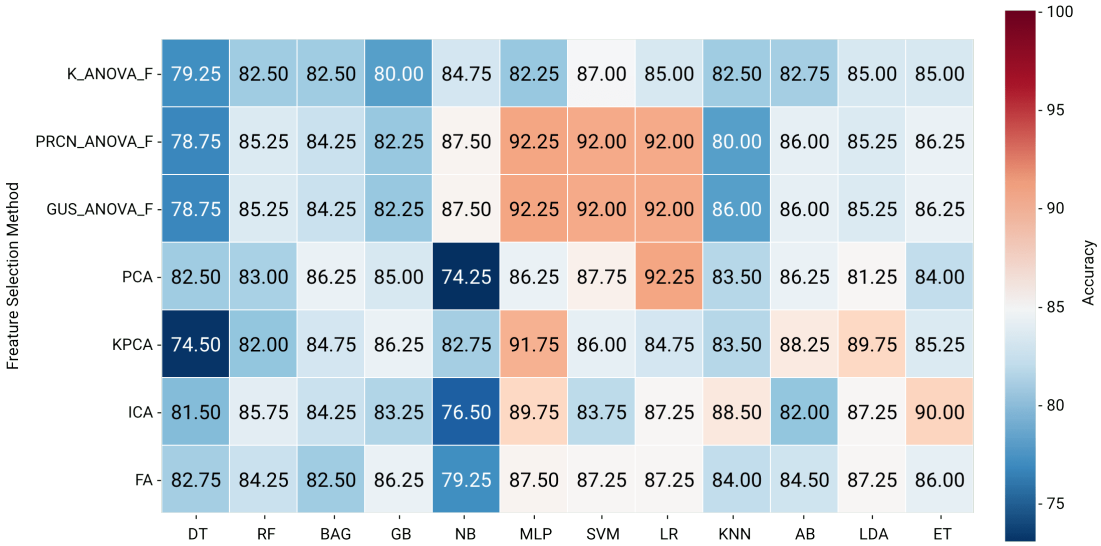


Figure 4.18: Survival Classification for Freije: Accuracy heatmap of 10-fold for top 15 features.

For the top 20 features at survival classification, the best results among the feature selection methods are achieved by Univariate feature selection, while the worst results are often achieved by PCA. Moreover, Kernel PCA has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, SVM, MLP and Logistic Regression (LR) classifier methods show the best results except for Naïve Bayes (NB), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.21 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Logistic Regression (LR) and SVM classifier using Univariate feature selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 93.19%. MLP using Factor analysis (FA) has an accuracy of 90.97%. AB classifier has better results after performing KPCA. The accuracy of this experiment is 90.69%. Independent Component Analysis (ICA) and Kernel PCA have a good predictive performance, which is 89.72%. While Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 51.25%. Table S44 at supplementary shows the results of cross-fold validation for different classifiers with

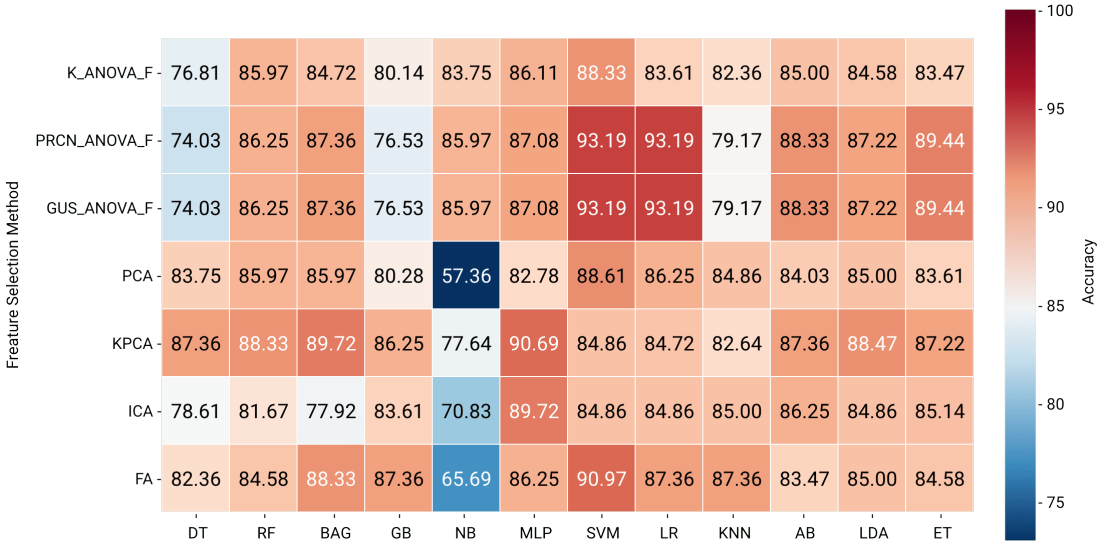


Figure 4.19: Survival Classification for Freije: Accuracy heatmap of 5-fold for top 20 features.

feature selection methods.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.22 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the SVM classifier using Principal Component Analysis (PCA), which is 93.25%. Logistic Regression (LR) and SVM using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has the accuracy 92%, though Linear Discriminant Analysis (LDA) with Kernel PCA and SVM with Factor analysis (FA) has the same accuracy. Logistic Regression (LR) and Gradient Boosting (GB) classifier has better results after performing Factor analysis (FA). The accuracy of this experiment is 90.75%. Kernel PCA has a good predictive performance, which is 89.75%. While Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 57.5%. Table S46 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 25 features at survival classification, overall, the best results among the feature selection methods are achieved by Univariate feature selection. Though PCA often produces the worst results. Moreover, Kernel PCA has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, SVM, MLP and Logistic Regression (LR) classifier methods show the best



Figure 4.20: Survival Classification for Freije: Accuracy heatmap of 10-fold for top 20 features.

results except for Naïve Bayes (NB), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.23 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the Extra Trees (ET) classifier using Independent Component Analysis (ICA), which is 93.33%. Logistic Regression (LR) and SVM using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has the accuracy 93.19%, though SVM using Principal Component Analysis (PCA) performs the same. Gradient Boosting (GB) using Independent Component Analysis (ICA) and SVM using Factor analysis (FA) has better results. The accuracy of this experiment is 92.08%. Factor analysis (FA) has a good predictive performance, which is 90.97%, while Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result 51.39%. Table S48 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.24 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the SVM classifier using

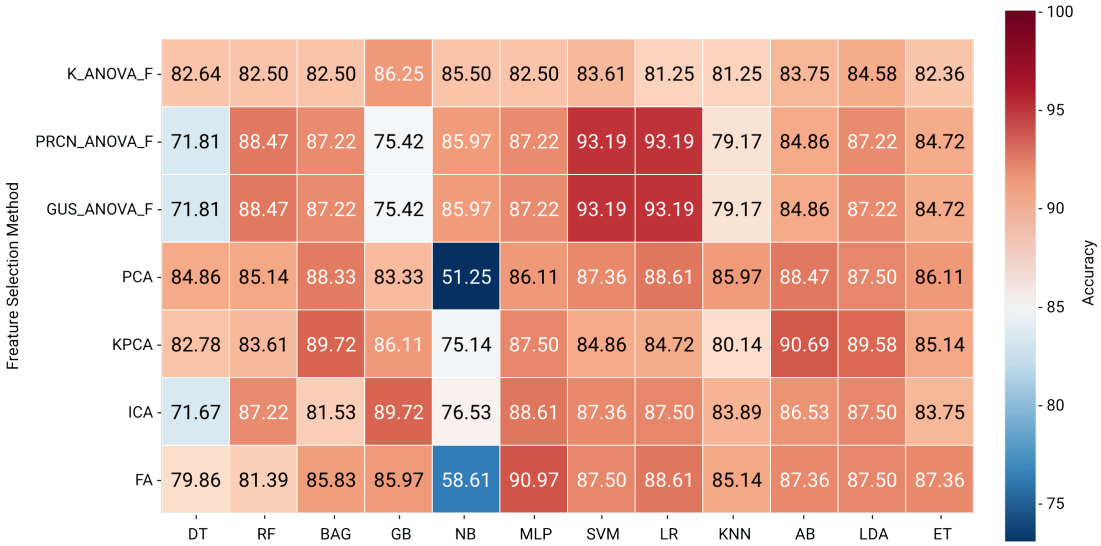


Figure 4.21: Survival Classification for Freije: Accuracy heatmap of 5-fold for top 25 features.

Factor analysis (FA), which is 93.25%. MLP using Factor analysis (FA) has an accuracy of 93%. Linear Discriminant Analysis (LDA) has better results after performing Kernel PCA. The accuracy of this experiment is 92.25%. Univariate Feature Selection has a good predictive performance, which is 92%, while Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result 51.75%. Table S50 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 30 features at survival classification, overall, the best results among the feature selection methods are achieved by Univariate feature selection. In contrast, the worst results are often achieved by PCA. Moreover, Factor analysis (FA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, SVM, MLP and Logistic Regression (LR) classifier methods show the best results except for Naïve Bayes (NB), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

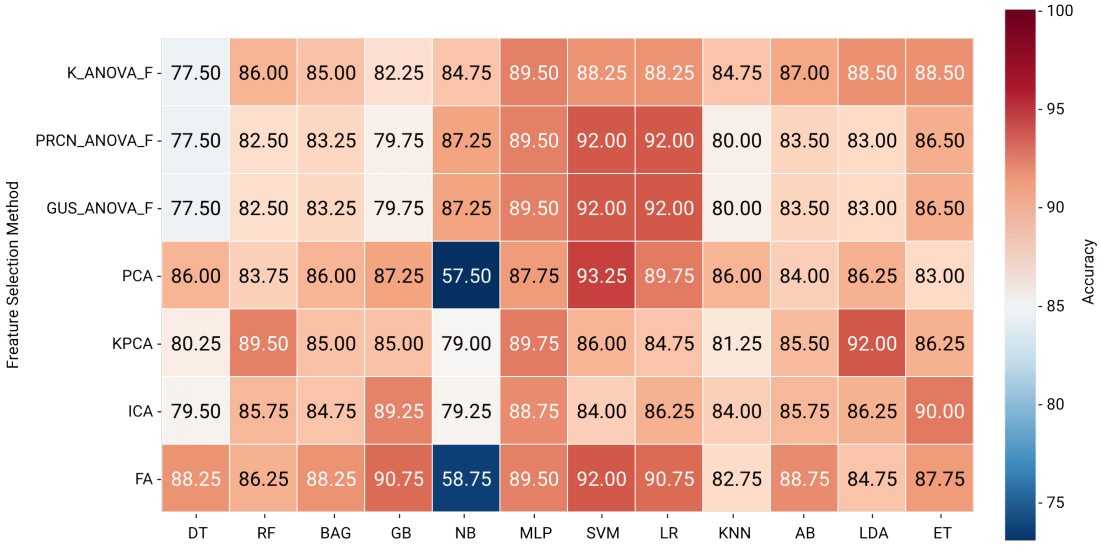


Figure 4.22: Survival Classification for Freije: Accuracy heatmap of 10-fold for top 25 features.

4.2.2.2 Results on the Phillips data set

For Top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.25 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 85.14%. Logistic Regression (LR) using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has an accuracy of 85%. MLP using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has good results too. The accuracy of this experiment is 84.02%. Principal Component Analysis (PCA) has a good predictive performance, which is 82.08% for the Extra Trees (ET) classifier. While Naïve Bayes (NB) with Principal Component Analysis (PCA) has the worst result of 59.03%. Table S52 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 15 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.26 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 85.75%. MLP using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has an

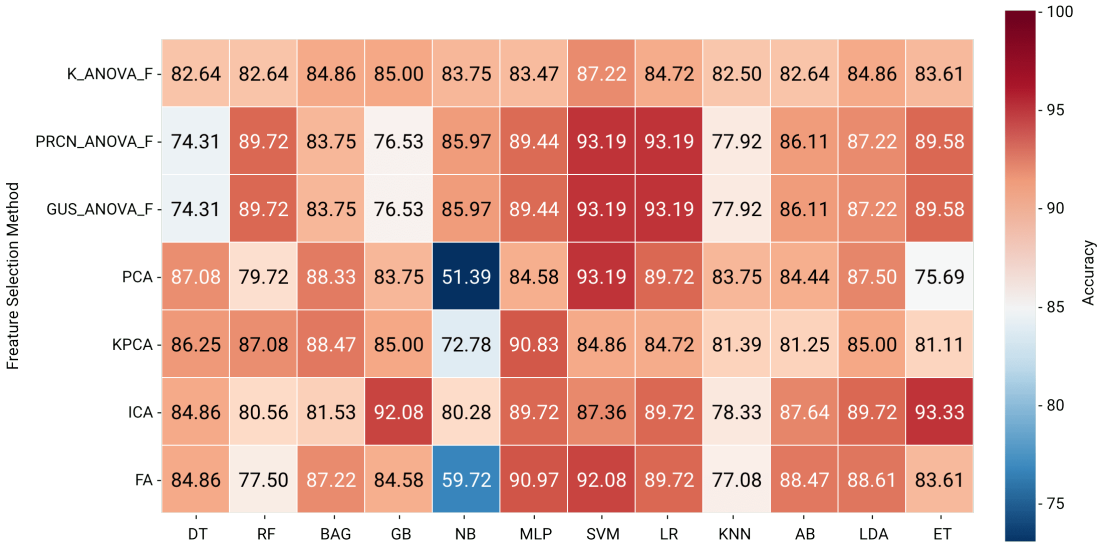


Figure 4.23: Survival Classification for Freije: Accuracy heatmap of 5-fold for top 30 features.

accuracy of 85.5%. Logistic Regression (LR) using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has good results too. The accuracy of this experiment is 85.25%. Factor analysis (FA) has a good predictive performance, which is 83% for the Extra Trees (ET) classifier. While Decision Tree (DT) with Kernel PCA has the worst result of 59%. Table S54 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 15 features at survival classification, overall, the best results among the feature selection methods are achieved by Univariate feature selection. In contrast, the worst results are often achieved by Principal Component Analysis (PCA) and Kernel PCA. Moreover, Factor analysis (FA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, SVM, MLP and Logistic Regression (LR) classifier methods show the best results except for Naïve Bayes (NB) and Decision Tree (DT), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 20 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.25 for 5-fold cross-validation. From the results shown

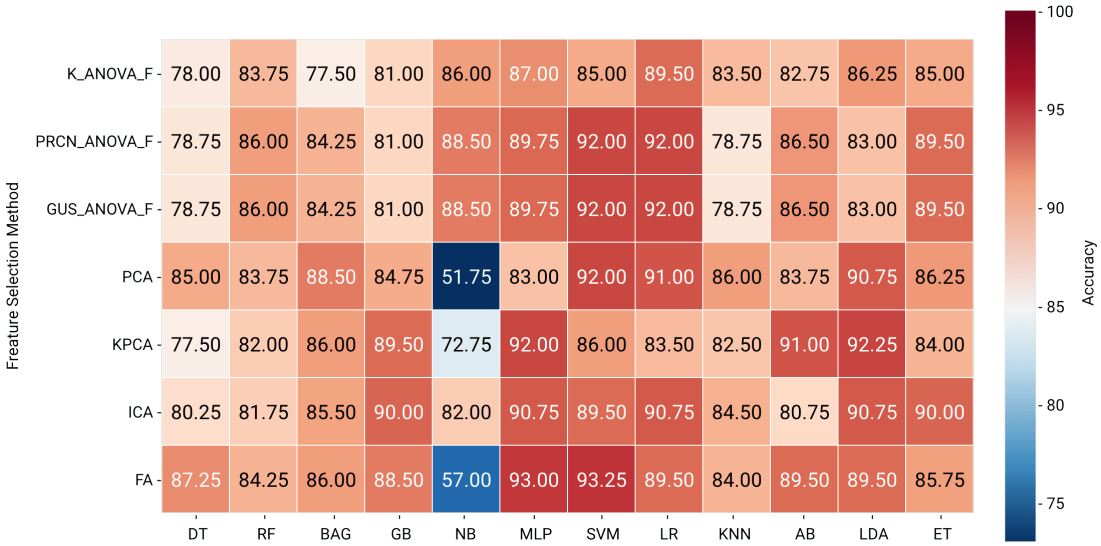


Figure 4.24: Survival Classification for Freije: Accuracy heatmap of 10-fold for top 30 features.

in the figure, we notice that highest accuracy is achieved by MLP classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 84.03%. Logistic Regression (LR) using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has an accuracy of 83.75%. SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has good results too. The accuracy of this experiment is 82.78%. Factor analysis (FA) has a good predictive performance, which is 80.56% for MLP classifier, while Decision Tree (DT) with Principal Component Analysis (PCA) has the worst result of 60.28%. Table S56 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 20 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.26 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 85.5%. Gradient Boosting (GB) using Univariate Feature Selection (K_ANOVA_F) has an accuracy of 85.25%. MLP using Univariate Feature Selection (K_ANOVA_F) has good results too. The accuracy of this experiment is 84.25%. Independent Component Analysis (ICA) has a good predictive performance, which is 81.25% while Decision

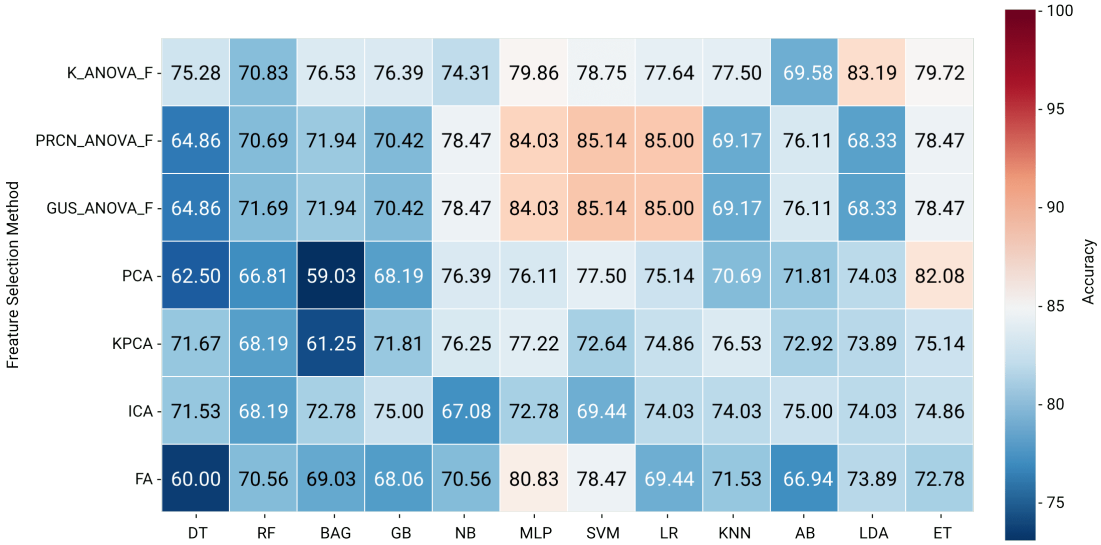


Figure 4.25: Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 15 features.

Tree (DT) with Kernel PCA and Independent Component Analysis (ICA) classifier has the worst result 62.5%. Table S58 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 20 features at survival classification, overall, the best results among the feature selection methods are achieved by Univariate feature selection. In contrast, the worst results are often achieved by Principal Component Analysis (PCA), KPCA. Moreover, Independent Component Analysis (ICA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, SVM, MLP and Logistic Regression (LR) classifier methods show the best results except for Decision Tree (DT), which is noticeably lower. Principal Component Analysis (PCA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.29 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by the SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F), which is 85%. MLP using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has an

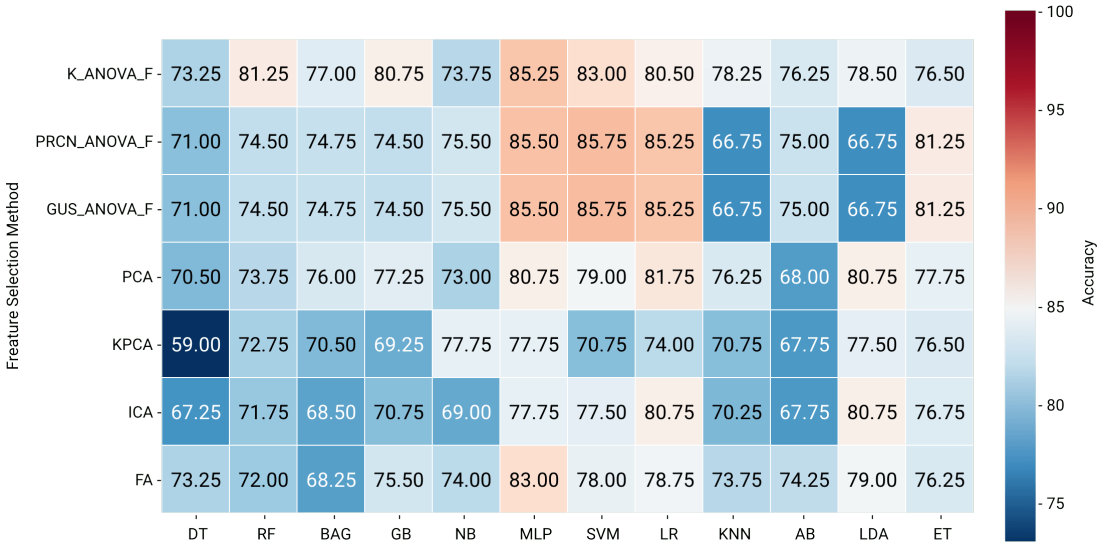


Figure 4.26: Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 15 features.

accuracy of 84.03%. Logistic Regression (LR) classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has good results too. The accuracy of this experiment is 83.89%. Independent Component Analysis (ICA) has a good predictive performance, which is 81.81% for the Extra Trees (ET) classifier. While Decision Tree (DT) with Kernel PCA classifier has the worst result of 61.81%. Table S60 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 25 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.30 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Logistic Regression (LR) and SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) which is 85.25%. MLP using Univariate Feature Selection (K_ANOVA_F) has an accuracy of 85%. Logistic Regression (LR) using Univariate Feature Selection (K_ANOVA_F) has good results too. The accuracy of this experiment is 84.25%. Factor analysis (FA) has a good predictive performance, which is 82.5%. While Decision Tree (DT) with Independent Component Analysis (ICA) classifier has the worst result of 60%. Table S62 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

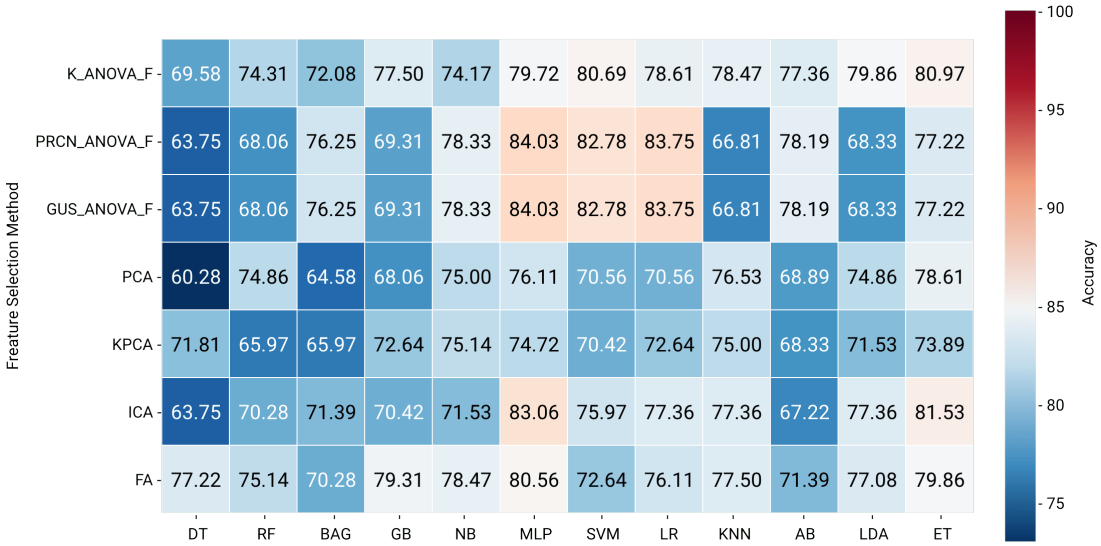


Figure 4.27: Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 20 features.

For the top 25 features at survival classification, overall, the best results among the feature selection methods are achieved by Univariate feature selection. However, the worst results are often achieved by Kernel PCA and Independent Component Analysis (ICA). Moreover, Factor analysis (FA) has comparable results to Univariate feature selection, which often has good predictive performance. In terms of classifiers, SVM, MLP and Logistic Regression (LR) classifier methods show the best results except for Decision Tree (DT), which is noticeably lower. Kernel PCA has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.31 for 5-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Logistic Regression (LR) and SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) which is 83.89%. MLP using Univariate Feature Selection (K_ANOVA_F) has an accuracy of 83.06%. MLP classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has good results too. The accuracy of this experiment is 82.78%. Factor analysis (FA) has a good predictive performance, which is 81.67% for MLP classifier while Decision Tree (DT) with Kernel PCA and

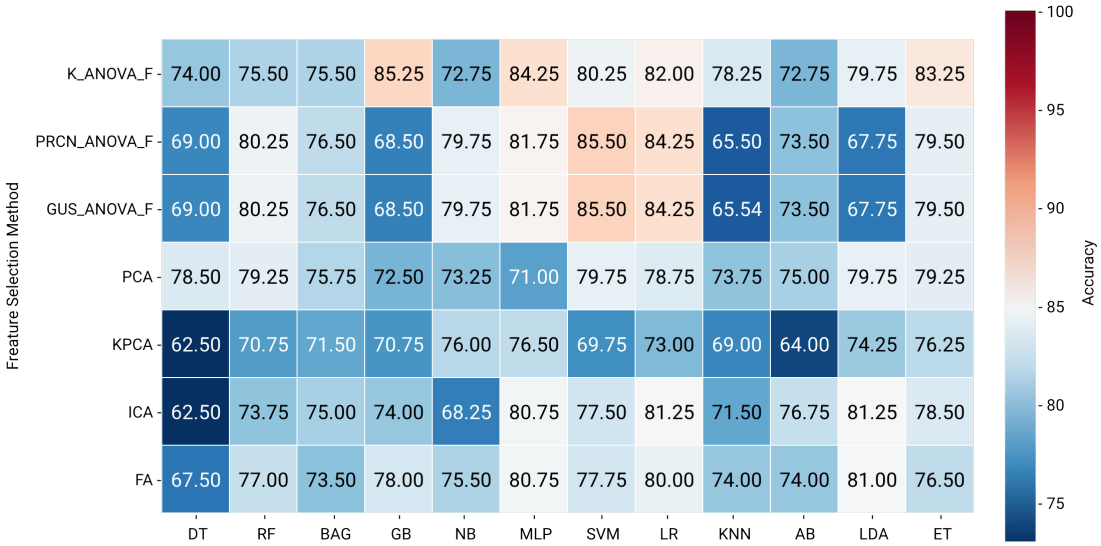


Figure 4.28: Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 20 features.

Naïve Bayes (NB) with Factor analysis (FA) has the worst result 60.42%. Table S64 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For Top 30 features, the accuracy of different classifiers with feature selection methods are shown in Figure 4.32 for 10-fold cross-validation. From the results shown in the figure, we notice that highest accuracy is achieved by Logistic Regression (LR) and SVM classifier using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) which is 85.25%. Extra Trees (ET) using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has an accuracy of 84.5%. MLP using Univariate Feature Selection (PRCN_ANOVA_F, GUS_ANOVA_F) has good results too. The accuracy of this experiment is 83.5%. Factor analysis (FA) has a good predictive performance, which is 83% while Decision Tree (DT) with Factor analysis (FA) has the worst result 58.5%. Table S66 at supplementary shows the results of cross-fold validation for different classifiers with feature selection methods.

For the top 30 features at survival classification, overall, the best results among the feature selection methods are achieved by Univariate feature selection. In contrast, the worst results are often achieved by Factor analysis (FA). Moreover, FA performs good and bad for the different classifiers. In terms of classifiers, SVM, MLP and

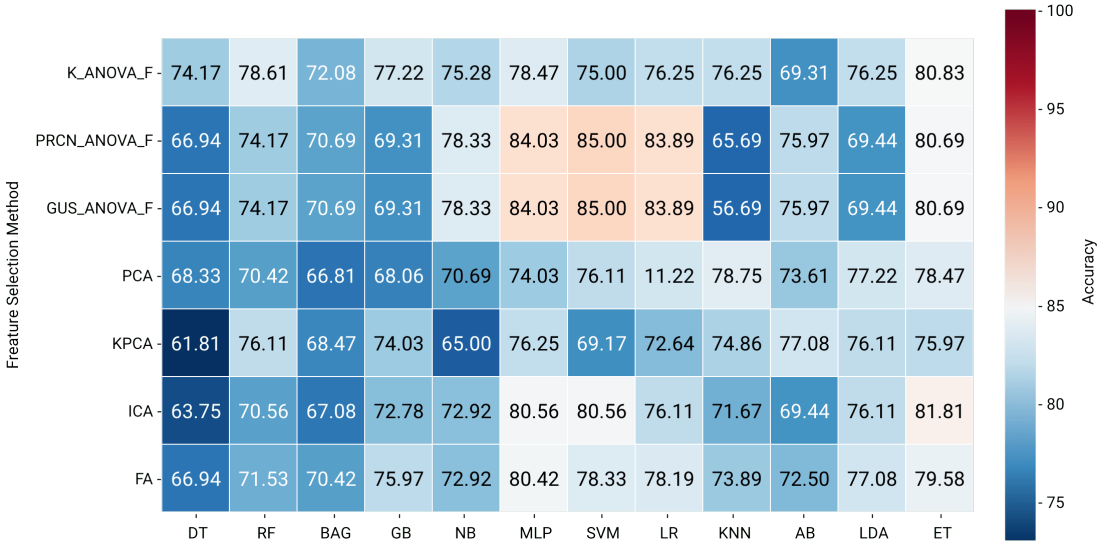


Figure 4.29: Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 25 features.

Logistic Regression (LR) classifier methods show the best results except for Decision Tree (DT), which is noticeably lower. Factor analysis (FA) has lower accuracy in terms of the feature selection method. Also, we perform 3-fold and 7-fold cross-validation, which can be found in Supplementary B.

4.2.3 Survival Time Prediction

Survival Time is defined as the time starting from a predefined point to the occurrence of the event of interest[64]. Kaplan-Meier Estimators are used to measure the fraction of subjects that survived under the same circumstances for a certain amount of survival time t. The Survival Curve of Kaplan-Meier is the probability of survival in a given length of time when time is considered in small intervals. Figure 4.33a and Figure 4.33b show the lifeline curve for survival time of glioma patient of Freije dataset and Phillips dataset, respectively.

In order to determine the relationship between grade and survival, we plotted the Kaplan-Meier survival curves of patients involved in this study categorized by grade.[20]. A critical aspect of survival analysis is that both the training data and the test data are subject to censoring because we are unable to observe the exact time of an event no matter how the data was split. Therefore, performance measures need to

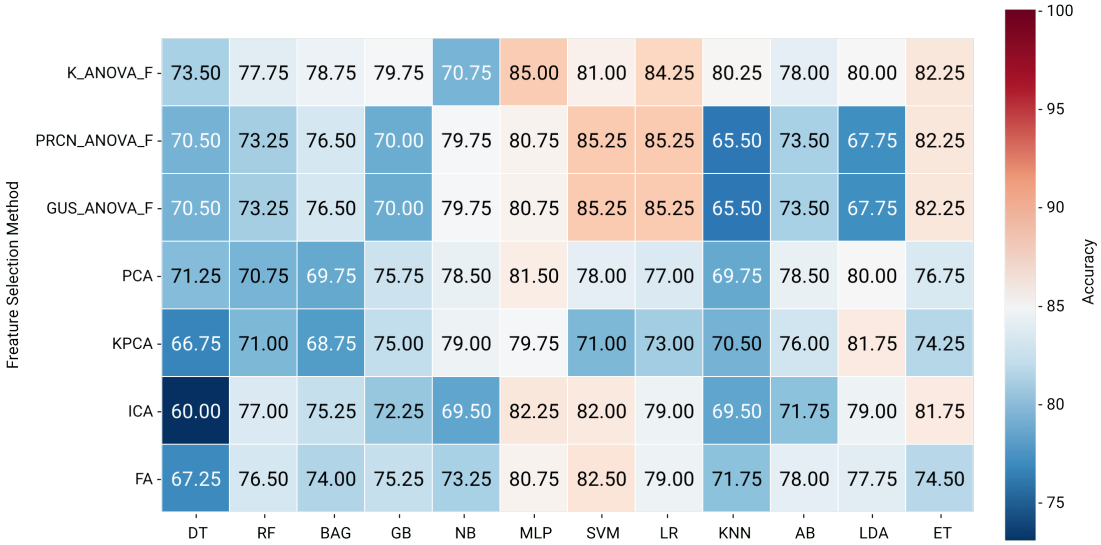


Figure 4.30: Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 25 features.

account for censoring. The most widely used performance measure is Harrell's concordance index. Given a set of (predicted) risk scores and observed times, it checks whether the ordering by risk scores is concordant with the order by actual survival time. So we need to censor the data. By far, the most widely used model to learn from censored survival data is Cox's proportional hazards model. The experimental results are described below.

4.2.3.1 Results on the Freije data set

Figure 4.34 shows the Kaplan-Meier survival plot. We observed that grade III tumors patients had 1830 median survival days in Freije datasets after plotting the Kaplan-Meier survival estimator. In contrast, the median survival of patients with grade IV tumors was 1022 days. Overall the median survival time is 1098 days.

To learn from censored survival data, we used the Cox Proportional Hazards model. Table 4.1 shows the experimental result. From our experiment, we found that the concordance is 0.99. Concordance is a measure of the predictive accuracy of the model being tested, ranging from 0.5 (completely random prediction) to 1 (perfect prediction)[65]. Our fitted models have an almost perfect concordance.

The model is fitted with 85 observations, 59 censored, and 26 has been observed.

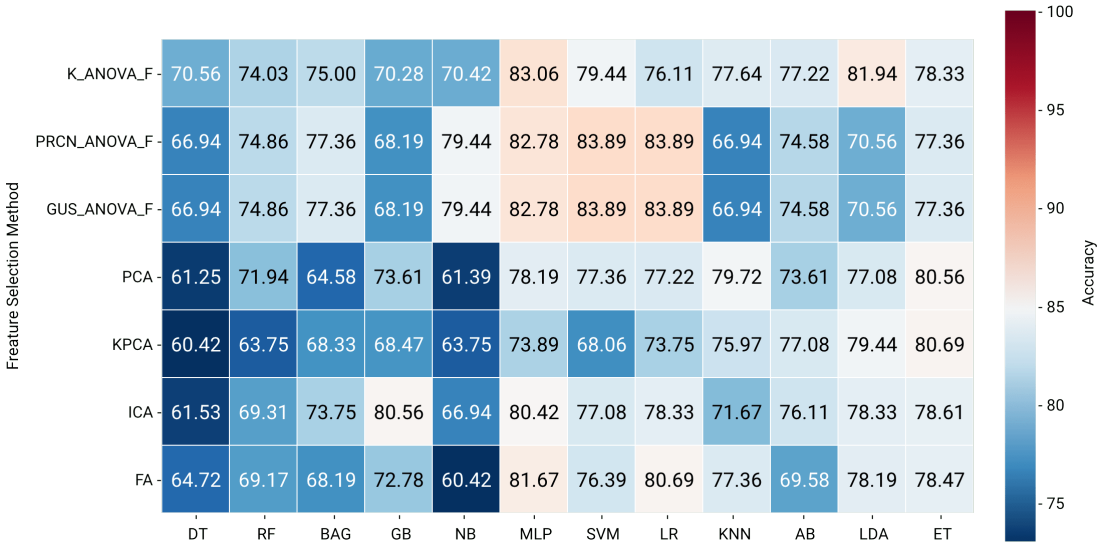


Figure 4.31: Survival Classification for Phillips: Accuracy heatmap of 5-fold for top 30 features.

If we check the model using assumptions, then it returned the Proportional hazard assumption looks okay. Figure 4.35 shows covariates of Cox's proportional Hazard Model.

If we evaluate the model using cross-fold validation for 5-fold and 10-fold, the accuracy is 72.61% and 77.5%, respectively. After that, we randomly choose five patients and predict their survival time in days, and the results are as follows: Patient number 26 survived for 320 days, Patient number 45 survived for 693 days, Patient number 34 survived for 2508 days, Patient number 14 survived for 1782 days, and Patient number 31 survived for 288 days. Figure 4.36 shows the illustration of the predicted result.

4.2.3.2 Results on the Phillips data set

Figure 4.37 shows the Kaplan-Meier survival plot. We found that grade III tumors patients had a 2681 median survival days in the Phillips datasets after plotting the Kaplan-Meier survival estimator. In contrast, the median survival of patients with grade IV tumors was 1939 days. Overall the median life time is 2275 days.

To learn from censored survival data, we used the Cox Proportional Hazards model. Table 4.2 shows the experimental result. From our experiment, we found that

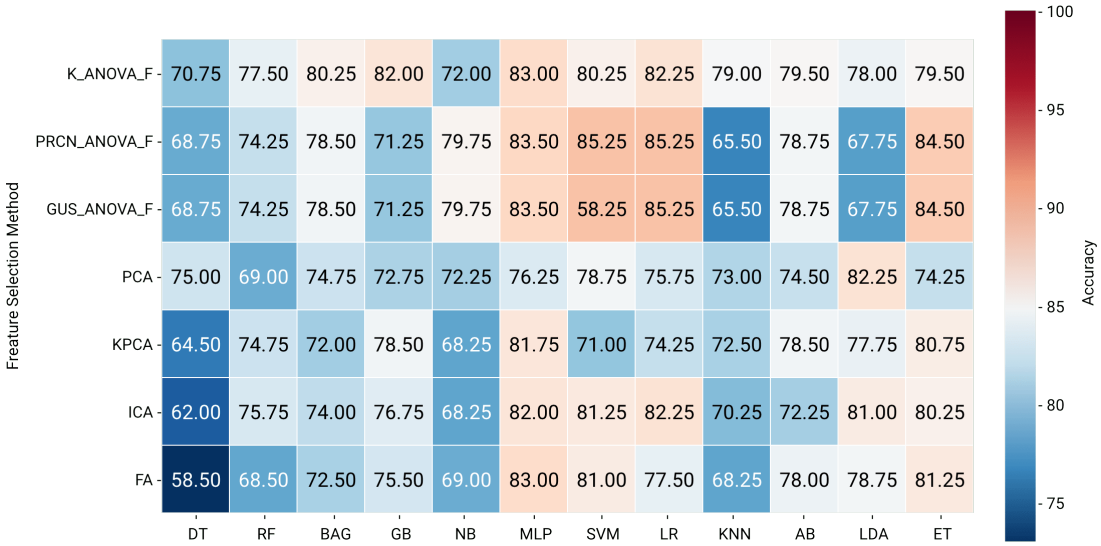
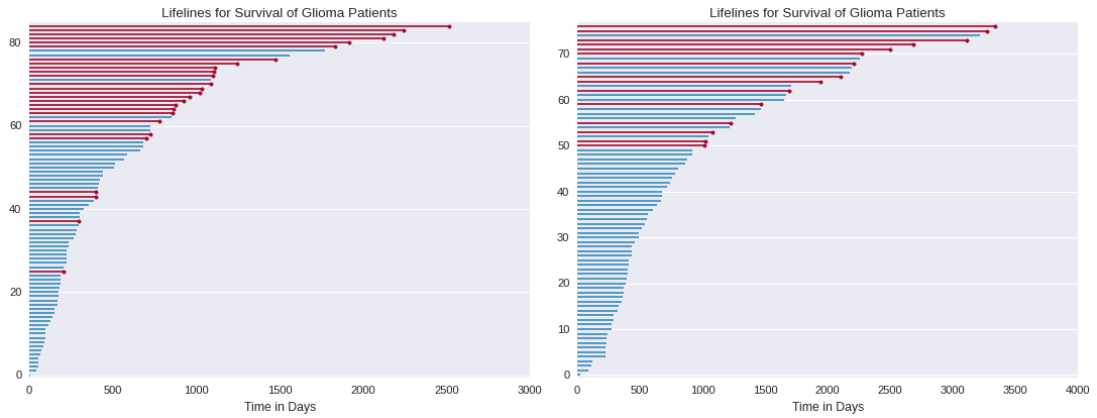


Figure 4.32: Survival Classification for Phillips: Accuracy heatmap of 10-fold for top 30 features.



(a) Freije Dataset **(b)** Phillips Dataset

Figure 4.33: Lifelines for Survival of Glioma Patients.

the concordance is 0.89. Concordance is a measure of the predictive accuracy of the model being tested, ranging from 0.5 (completely random prediction) to 1 (perfect prediction)[65]. Our fitted models have an almost perfect concordance.

The model is fitted with 77 observations, 62 censored, and 15 has been observed. If we check the model using assumptions, then it returned the Proportional hazard assumption looks okay. Figure 4.38 show the result of covariates of Cox's proportional

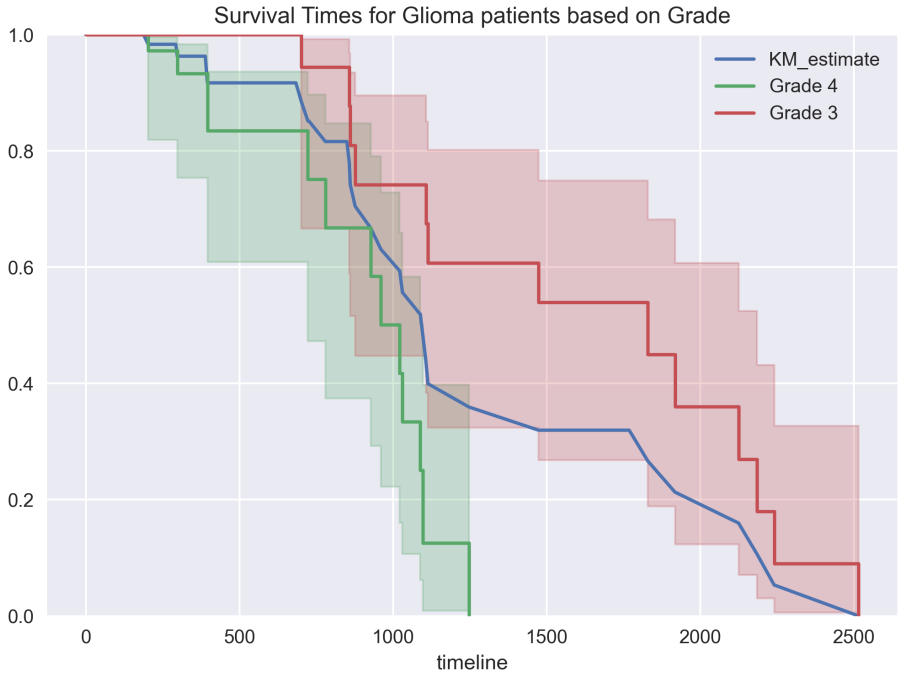


Figure 4.34: Freije : Kaplan-Meier survival plot of the glioma patients.

Hazard Model.

If we evaluate the model using cross-fold validation for 5-fold and 10-fold, the accuracy is 73.78% and 76.19%, respectively. After that, we randomly choose five patients and predict their survival time in days, and the results are as follows: Patient number 27 survived for 2240 days, Patient number 38 survived for 2457 days, Patient number 4 survived for 1984 days, Patient number 28 survived for 2233 days, and Patient number 43 survived for 2499 days. Figure 4.39 shows the illustration of the predicted result.

4.2.4 Comparison with Previous Works

For grade classification, We also compared our proposed method's performance with two other paperwork done by L. Zaakaria et al.[1] and H. Abusamra[2]. The cross-validation results achieved by all these works and their variations are reported in Table 4.3. There are a couple of blank spaces in the table since the literature did not mention those.

model	lifelines.CoxPHFitter									
duration col	'survival_time'									
event col	'living'									
baseline estimation	breslow									
number of observations	85									
partial log-likelihood	-12.8									
	coef	exp(coef)	se(coef)	coef lower 95%	coef upper 95%	exp(coef) lower 95%	exp(coef) upper 95%	z	p	-log2(p)
age	1.04	2.83	0.39	0.27	1.81	1.32	6.1	2.66	0.01	7
gender	-0.77	0.46	2.57	-5.81	4.27	0	71.36	-0.3	0.76	0.39
grade	3.01	20.36	3.03	-2.93	8.95	0.05	7728.24	0.99	0.32	1.64
201668_x_at	0	1	0	0	0	1	1	-2.47	0.01	6.22
202152_x_at	0	1	0	-0.01	0	0.99	1	-2.61	0.01	6.8
202259_s_at	0	1	0	0	0.01	1	1.01	2.55	0.01	6.53
202596_at	0	1	0	0	0	1	1	-1.51	0.13	2.93
206107_at	0.01	1.01	0	0	0.01	1	1.01	2.46	0.01	6.18
206734_at	0	1	0	0	0	1	1	-0.64	0.52	0.94
208334_at	0	1	0	-0.01	0.01	0.99	1.01	-0.14	0.89	0.17
210875_s_at	0	1	0	0	0	1	1	-0.13	0.9	0.15
211090_s_at	0	1	0	0	0.01	1	1.01	2.68	0.01	7.08
211094_s_at	0	1	0	0	0.01	1	1.01	0.6	0.55	0.87
212678_at	0	1	0	-0.01	0	0.99	1	-1.88	0.06	4.06
214250_at	0	1	0	0	0.01	1	1.01	1.86	0.06	4.01
215991_s_at	0.03	1.03	0.01	0	0.05	1	1.05	2.01	0.04	4.51
217142_at	-0.01	0.99	0.01	-0.03	0	0.98	1	-2.24	0.02	5.33
217196_s_at	0	1	0	-0.01	0	0.99	1	-2.77	0.01	7.5
217987_at	0	1	0	0	0	1	1	0.13	0.9	0.15
219312_s_at	-0.02	0.98	0.01	-0.03	-0.01	0.97	0.99	-2.83	¡0.005	7.76
220055_at	-0.01	0.99	0	-0.01	0	0.99	1	-1.55	0.12	3.05
220606_s_at	0.01	1.01	0	0	0.02	1	1.02	2.21	0.03	5.21
49306_at	0	1	0	0	0	1	1	3.17	¡0.005	9.34
Concordance	0.99									
Log-likelihood ratio test	108.46 on 23 df									
-log2(p) of ll-ratio test	40.95									

Table 4.1: Freije : Cox's Proportional Hazard Model

From the findings recorded in this table, we can conclude that our proposed method achieved higher accuracy compared to other methods. We examined five different feature selection methods but in this Table 4.3 we only presented the result which has the highest accuracy. Table 4.3 clearly states that, our model reported the highest accuracy which is 98.57%, whereas H. Abusamra[2] reported 94.59% and L. Zaakaria et al.[1] reported 91.8% accuracy.

For the survival Classification task, there is no previous work we found based on gene expression data for Freije and Phillips dataset. So we present our highest accuracy in table 4.4 for reference. Table 4.4 reported that highest accuracies in Freije and Phillips of 94.5% and 85.75%, respectively

For survival prediction, we have also compared the predicted result with the actual dataset. After running the survival prediction model on the Freije dataset, we randomly select five patients. The results are as follows: (i) patient number 26 survived for 320 days though, the patient survived for 279 days actually, (ii) Patient number 45 survived for 693 days (actual survival 702 days), (iii) Patient number 34 survived for 2508 days (actual survival 850 days), (iv) Patient number 14 survived for 1782 days

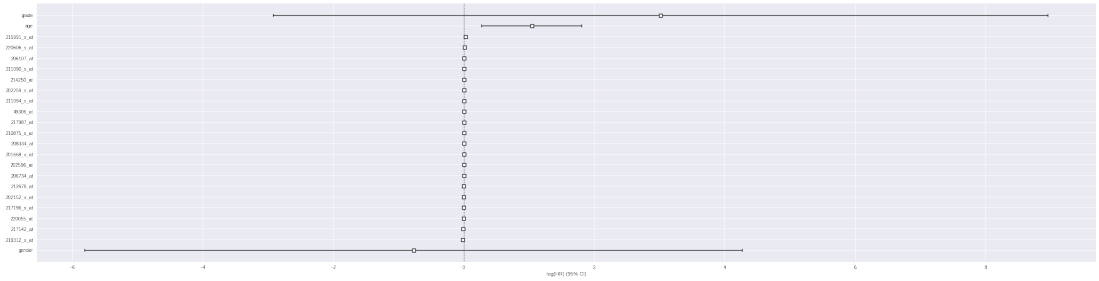


Figure 4.35: Freije : Cox's Proportional Hazard Model Curves.

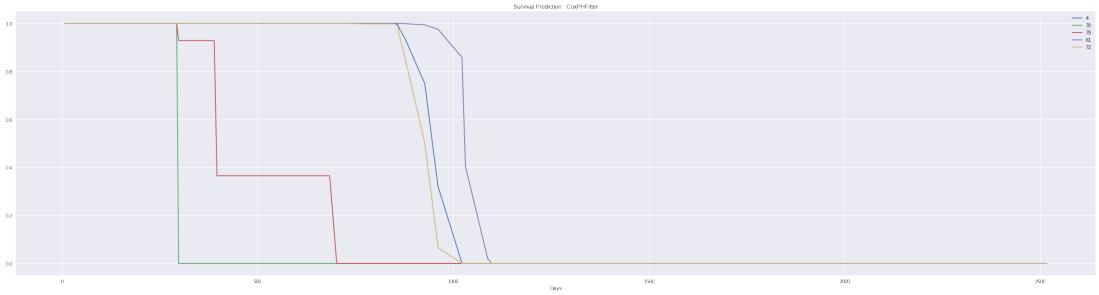


Figure 4.36: Freije : Survival predictor for 5 patients.

(actual survival 1782 days), and (v) Patient number 31 survived for 288 days (actual survival 153 days). For Phillips dataset the results are as follows: (i) Patient number 27 survived for 2240 days (actual survival 665 days), (ii) Patient number 38 survived for 2457 days (actual survival 399 days), (iii) Patient number 4 survived for 1984 days (actual survival 499 days), (iv) Patient number 28 survived for 2233 days (actual survival 1694 days), and (v) Patient number 43 survived for 2499 days (actual survival 875 days). Table 4.5 shows the comparative result of Freije and Phillips dataset.

4.3 Summary

From the result, we can summarize as follows, for grade classification, the best result 98.75% is achieved for the Freije dataset using AdaBoost classifier, and 95.67% is the accuracy for the phillips dataset using Extra tree classifier for univariate feature selection. For survival classification, the best result 94.5% is achieved by PCA using SVM for the freije dataset, and 85.75% is the accuracy for the phillips dataset using SVM for univariate feature selection. Median survival time for the freije dataset 1098 days and 2275 days for the phillips dataset.

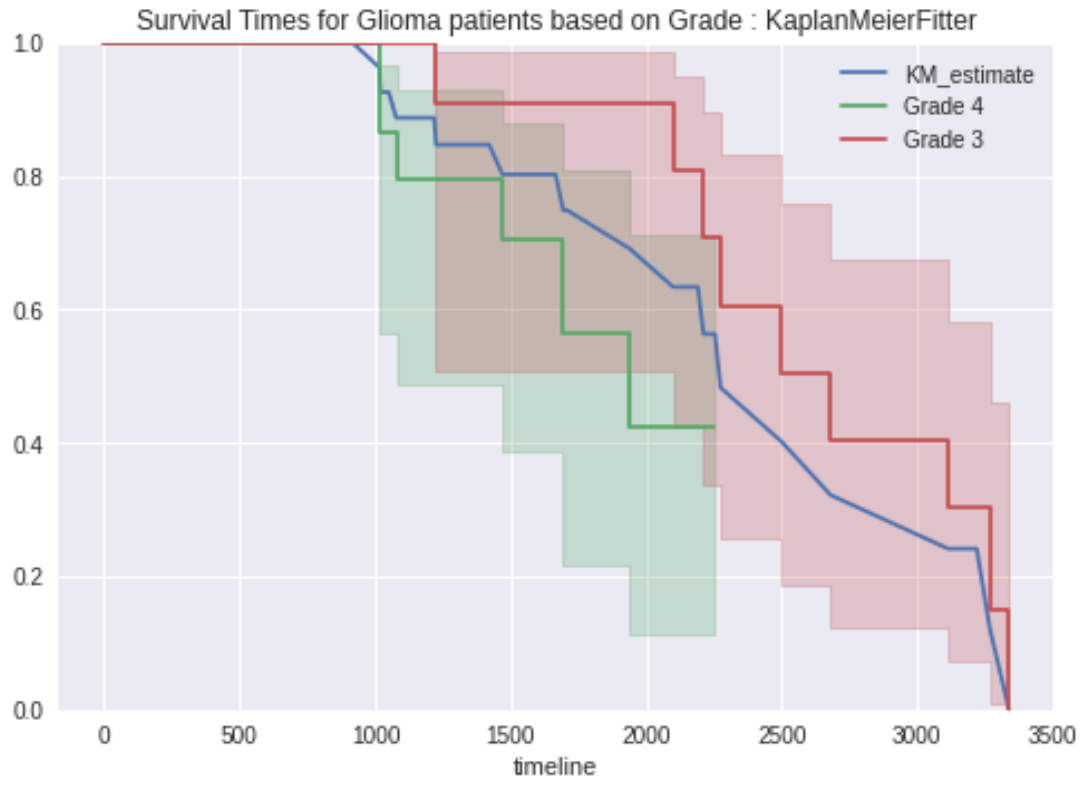


Figure 4.37: Phillips : Kaplan-Meier survival plot of the glioma patients.

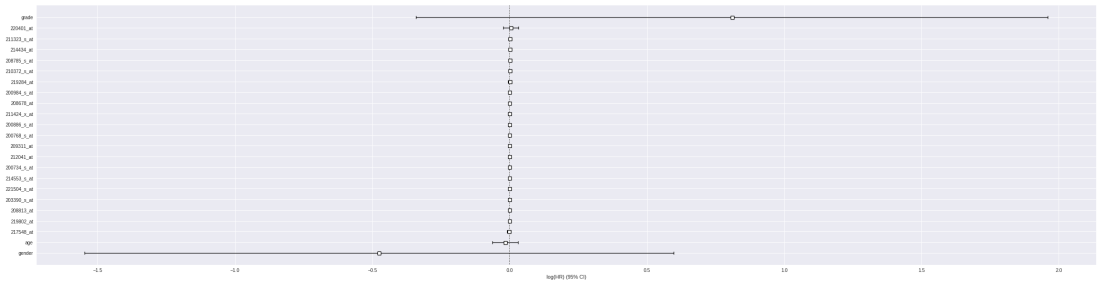


Figure 4.38: Phillips : Cox's Proportional Hazard Model Curves.

model	lifelines.CoxPHFitter									
duration col	'survival_time'									
event col	'living'									
number of observations	77									
partial log-likelihood	-27.64									
	coef	exp(coef)	se(coef)	coef lower 95%	coef upper 95%	exp(coef) lower 95%	exp(coef) upper 95%	z	p	-log2(p)
age	-0.02	0.98	0.02	-0.06	0.03	0.94	1.03	-0.66	0.51	0.97
gender	-0.48	0.62	0.55	-1.55	0.6	0.21	1.82	-0.87	0.39	1.38
grade	0.81	2.25	0.59	-0.34	1.96	0.71	7.1	1.38	0.17	2.58
200734_s_at	0	1	0	0	0	1	1	-0.33	0.74	0.43
200768_s_at	0	1	0	0	0	1	1	-0.13	0.9	0.16
200886_s_at	0	1	0	0	0	1	1	0.04	0.97	0.05
200984_s_at	0	1	0	0	0	1	1	0.5	0.62	0.7
203390_s_at	0	1	0	0	0	1	1	-0.74	0.46	1.12
208678_at	0	1	0	0	0	1	1	0.31	0.75	0.41
208785_s_at	0	1	0	0	0	1	1	1.21	0.23	2.15
208813_at	0	1	0	0	0	1	1	-0.55	0.58	0.78
209311_at	0	1	0	0	0	1	1	-0.07	0.94	0.08
210372_s_at	0	1	0	0	0	1	1	1.16	0.25	2.03
211323_s_at	0	1	0	0	0.01	1	1.01	0.48	0.63	0.66
211424_x_at	0	1	0	-0.01	0.01	0.99	1.01	0.02	0.99	0.02
212041_at	0	1	0	0	0	1	1	-0.24	0.81	0.3
214434_at	0	1	0	0	0	1	1	1.3	0.19	2.38
214553_s_at	0	1	0	0	0	1	1	-0.25	0.8	0.31
217548_at	0	1	0	-0.01	0	0.99	1	-0.56	0.57	0.8
219284_at	0	1	0	-0.01	0.01	0.99	1.01	0.2	0.84	0.25
219802_at	0	1	0	0	0	1	1	-0.5	0.62	0.69
220401_at	0	1	0.01	-0.02	0.03	0.98	1.03	0.3	0.76	0.39
221504_s_at	0	1	0	0	0	1	1	-0.31	0.75	0.41
Concordance	0.89									
Log-likelihood ratio test	11.01 on 23 df									
-log2(p) of ll-ratio test	0.02									

Table 4.2: Phillips : Cox's Proportional Hazard Model.

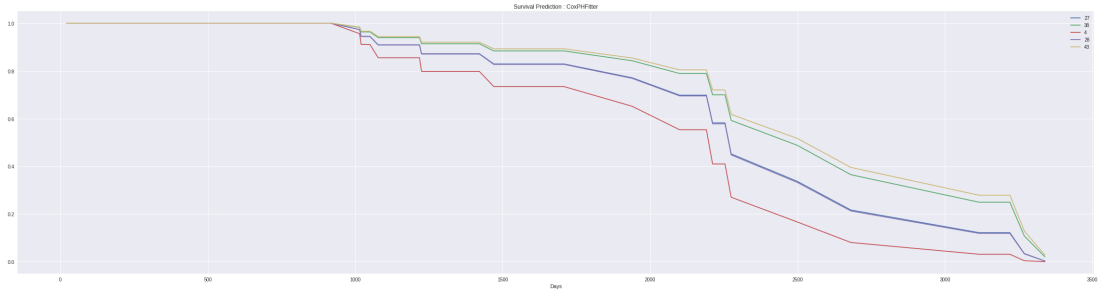


Figure 4.39: Phillips : Survival predictor for 5 patients.

Author	Dataset	Selected Features	Best performing	Fold	ACC	auROC	auPR	F1	Recall	SNS	SPF
L. Zakaria et al.	Freije	-	SVM	-	91.8	0.9334	-	-	-	-	-
L. Zakaria et al.	Philips	-	SVM	-	86.7	0.8519	-	-	-	-	-
Heba Abusamra	Freije	20	SVM,KNN	5-Fold	94.59	-	-	-	-	96.07	91.3
Heba Abusamra	Philips	20	SVM	5-Fold	90.81	-	-	-	-	93.33	82.6
Heba Abusamra	Freije	40	SVM	5-Fold	94.59	-	-	-	-	96.07	91.3
Heba Abusamra	Philips	40	SVM	5-Fold	90.81	-	-	-	-	94.66	78.26
Proposed	Freije	25	DT	5-Fold	97.5	97.5	96.53	97.49	97.5	95.83	95.83
Proposed	Freije	30	DT	5-Fold	97.5	97.5	96.53	97.49	97.5	96.25	96.25
Proposed	Freije	15	AB	10-Fold	98.75	100	100	98.57	97.5	84.26	90.76
Proposed	Freije	25	AB	10-Fold	98.75	100	100	98.57	97.5	84.36	90.82
Proposed	Phillips	20	BAG	5-Fold	94.36	97.3	97.28	94.42	94.18	84.61	92.26
Proposed	Phillips	25	ET	5-Fold	94.36	97.88	97.13	94.22	94.18	84.91	94.24
Proposed	Phillips	20	ET	10-Fold	95.67	99.04	98.91	95.35	94.67	85.82	93.5
Proposed	Phillips	30	ET	10-Fold	95.67	96.67	94.79	95.59	96.67	85.9	93.61

Table 4.3: Grade Classification : Comparison of the performance of our experiment with L. Zaakaria et al.[1] and H. Abusamra[2] work.

Author	Dataset	Selected Features	Best performing	Fold	ACC	auROC	auPR	F1	Recall	SNS	SPF
Proposed	Freije	30	ET	5-Fold	93.33	96.54	97.07	91.79	86.67	84.71	83.95
Proposed	Freije	20	SVM	10-Fold	94.5	95.28	95.68	94.37	96	84.72	83.32
Proposed	Phillips	15	SVM	5-Fold	85.14	93.09	95.24	85.8	88.89	74.33	71.04
Proposed	Phillips	15	SVM	10-Fold	85.75	93.62	95.77	84.59	85	75.57	74.58

Table 4.4: Survival Classification : Highest accuracy of the proposed model.

Dataset	Patients ID	Actual Time	Predicted Time	Dataset	Patients ID	Actual Time	Predicted Time
Freije	26	279 days	320 days	Phillips	27	665 days	2240 days
Freije	45	702 days	693 days	Phillips	38	399 days	2457 days
Freije	34	850 days	2508 days	Phillips	4	499 days	1984 days
Freije	14	173 days	1782 days	Phillips	28	1694 days	2233 days
Freije	31	153 days	288 days	Phillips	43	875 days	2499 days

Table 4.5: Survival Prediction : Comparison of the performance of our experiment.

Chapter 5

Summary, Limitations, and Future Work

5.1 Summary

We presented a novel pipeline framework for glioma analysis that uses several feature selection algorithms followed by effective classifier selection. The efficiency of twelve different classification methods had been evaluated. Also we have solve a new problem survival time prediction, Kaplan-Meier Estimate and Cox Proportional Hazard Model used to predict the survival time in days.

5.2 Limitations

To present a comparative study on Grade classification and Survival classification, different types of feature selection methods with classification methods applied on gene expression data. For survival prediction, we used Kaplan-Meier Estimate and Cox Proportional Hazard Model. The accuracy of AdaBoost (AB) AND Gradient Boost (GB) out of twelve different classification methods including Extra Trees Classifier (ET), AdaBoost (AB), Bagging (BAG), Gradient Boosting (GB), Gaussian Naïve Bayes (NB), Multi-Layer Perception (MLP), Support Vector Machines (SVM), Logistic Regression (LR), k-nearest neighbors (k-NN), Random Forrest (RF), Linear Discriminant Analysis (LDA), and Decision Trees (DT), and univariate filter methods With five different unsupervised dimensionality reduction methods including Principal Component Analysis (PCA), Kernel PCA (KPCA), Factor Analysis (FA), and Independent Component Analysis (ICA) was compared for grade classification and survival classification. These methods applied on Freije and Philips datasets of glioma gene expression. Five-fold and Ten-fold cross-validation is used to measure the performance of classifier for grade & survival. For grade classification task on Freije, the best results achieved by Univariate filter methods, while PCA often obtains the

worst results. Although we have performed many experiments to obtain the best result, there are some limitations in our work. These limitations are described below:

- Global variations in gene expression can be detected in microarray experiments, but these experiments are constrained by consumption and expense.
- The variability of the microarray results can be important , especially for genes with low levels of expression.
- The quality and quantity of RNA remains a major challenge for microarray experiments. The complexity of the tissue sample and amount of tissue obtained limit the quantity and quality of the RNA which can be extracted.
- Another drawback of microarray technologies is that only knowledge about the genes used in the array can be given by each microarray.
- In an ideal world, prediction of the survival time yield meaningful results. However, the world is not perfect. We never know what kind of disease a patient can be suffered.

5.3 Future Work

In the future, we want to check our model performance based on computational pathology. Computational pathology is a diagnostic methodology that incorporates several data points such as: anatomy, radiology, surgical, molecular, and laboratory operations; statistical models are used to produce medical inferences; and clinically actionable information.So that the right prognosis method for the benign tumor will reduce the risk of developing malignant tumors. Also, if we can find out the gene expression pattern for an individual person, then we can predict the possibility of a brain tumor for a specific person.

Bibliography

[1] L. Zakaria, H. M. Ebeid, S. Dahshan, and M. F. Tolba, "Analysis of classification methods for gene expression data," in *Advances in Intelligent Systems and Computing*. Springer International Publishing, Mar. 2019, pp. 190–199. [Online]. Available: https://doi.org/10.1007/978-3-030-14118-9_19 xi, 19, 27, 78, 79, 83

[2] H. Abusamra, "A comparative study of feature selection and classification methods for gene expression data of glioma," *Procedia Computer Science*, vol. 23, pp. 5–14, 2013. [Online]. Available: https://doi.org/10.1016/j.procs.2013.10.003 xi, 1, 19, 20, 21, 28, 33, 34, 44, 78, 79, 83

[3] S. for experimental biology (GB). Symposium, *The Biological Replication of Macromolecules*. Published for the Company of Biologists on behalf on the Society for . . . , 1958. 6

[4] T. K. Paul and H. Iba, "Extraction of informative genes from microarray data," in *Proceedings of the 2005 conference on Genetic and evolutionary computation - GECCO 05*. ACM Press, 2005. [Online]. Available: https://doi.org/10.1145/1068009.1068081 7

[5] M. Schena, D. Shalon, R. W. Davis, and P. O. Brown, "Quantitative monitoring of gene expression patterns with a complementary DNA microarray," *Science*, vol. 270, no. 5235, pp. 467–470, Oct. 1995. [Online]. Available: https://doi.org/10.1126/science.270.5235.467 7

[6] A. Osareh and B. Shadgar, "Microarray data analysis for cancer classification," in *2010 5th International Symposium on Health Informatics and Bioinformatics*. IEEE, 2010. [Online]. Available: https://doi.org/10.1109/hibit.2010.5478893 8

[7] J. K. Nurminen, "Using software complexity measures to analyze algorithms—an experiment with the shortest-paths algorithms," *Computers & Operations Research*, vol. 30, no. 8, pp. 1121–1134, Jul. 2003. [Online]. Available: https://doi.org/10.1016/s0305-0548(02)00060-6 8

[8] J.-G. Zhang and H.-W. Deng, "Gene selection for classification of microarray data based on the bayes error," *BMC Bioinformatics*, vol. 8, no. 1, Oct. 2007. [Online]. Available: https://doi.org/10.1186/1471-2105-8-370 8

[9] R. Kohavi and G. H. John, "The wrapper approach," in *Feature Extraction, Construction and Selection.* Springer US, 1998, pp. 33–50. [Online]. Available: https://doi.org/10.1007/978-1-4615-5725-8_3 8

[10] P. Dunn-Rankin, G. A. Knezek, S. R. Wallace, and S. Zhang, *Scaling methods.* Psychology Press, 2014. 9

[11] C. Parmar, P. Grossmann, D. Rietveld, M. M. Rietbergen, P. Lambin, and H. J. W. L. Aerts, "Radiomic machine-learning classifiers for prognostic biomarkers of head and neck cancer," *Frontiers in Oncology*, vol. 5, Dec. 2015. [Online]. Available: https://doi.org/10.3389/fonc.2015.00272 9

[12] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning.* Springer, 2013, vol. 112. 9

[13] Y. Zhang, A. Oikonomou, A. Wong, M. A. Haider, and F. Khalvati, "Radiomics-based prognosis analysis for non-small cell lung cancer," *Scientific Reports*, vol. 7, no. 1, Apr. 2017. [Online]. Available: https://doi.org/10.1038/srep46349 9

[14] M. H. Nguyen and F. Torre, "Robust kernel principal component analysis," in *Advances in Neural Information Processing Systems*, 2009, pp. 1185–1192. 11

[15] M. Prasad, A. Sowmya, and I. Koch, "Efficient feature selection based on independent component analysis," in *Proceedings of the 2004 Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004.* IEEE. [Online]. Available: https://doi.org/10.1109/issnip.2004.1417499 11

[16] A. Hyvärinen, J. Karhunen, and E. Oja, "al. independent component analysis. john willey ans sons," *Inc., New York*, 2001. 11

[17] N. Kambhatla and T. K. Leen, "Dimension reduction by local principal component analysis," *Neural computation*, vol. 9, no. 7, pp. 1493–1516, 1997. 12

[18] R. A. Johnson, D. W. Wichern *et al.*, *Applied multivariate statistical analysis*. Prentice hall Upper Saddle River, NJ, 2002, vol. 5, no. 8. 12

[19] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001. 13

[20] W. A. Freije, F. E. Castro-Vargas, Z. Fang, S. Horvath, T. Cloughesy, L. M. Liau, P. S. Mischel, and S. F. Nelson, "Gene expression profiling of gliomas strongly predicts survival," *Cancer Research*, vol. 64, no. 18, pp. 6503–6510, Sep. 2004. [Online]. Available: https://doi.org/10.1158/0008-5472.can-04-0452 18, 19, 33, 74

[21] H. S. Phillips, S. Kharbanda, R. Chen, W. F. Forrest, R. H. Soriano, T. D. Wu, A. Misra, J. M. Nigro, H. Colman, L. Soroceanu, P. M. Williams, Z. Modrusan, B. G. Feuerstein, and K. Aldape, "Molecular subclasses of high-grade glioma predict prognosis, delineate a pattern of disease progression, and resemble stages in neurogenesis," *Cancer Cell*, vol. 9, no. 3, pp. 157–173, Mar. 2006. [Online]. Available: https://doi.org/10.1016/j.ccr.2006.02.019 19, 33

[22] J. Behnan, G. Finocchiaro, and G. Hanna, "The landscape of the mesenchymal signature in brain tumours," *Brain*, vol. 142, no. 4, pp. 847–866, Mar. 2019. [Online]. Available: https://doi.org/10.1093/brain/awz044 19

[23] H. Liu, J. Li, and L. Wong, "A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns," *Genome informatics*, vol. 13, pp. 51–60, 2002. 19

[24] M. Pirooznia, J. Y. Yang, M. Q. Yang, and Y. Deng, "A comparative study of different machine learning methods on microarray gene expression data," *BMC Genomics*, vol. 9, no. Suppl 1, p. S13, 2008. [Online]. Available: https://doi.org/10.1186/1471-2164-9-s1-s13 20, 44

[25] T. Li, C. Zhang, and M. Ogihara, "A comparative study of feature selection and multiclass classification methods for tissue classification based on gene expression," *Bioinformatics*, vol. 20, no. 15, pp. 2429–2437, Apr. 2004. [Online]. Available: https://doi.org/10.1093/bioinformatics/bth267 20

[26] E. P. Xing, M. I. Jordan, and R. M. Karp, "Feature selection for high-dimensional genomic microarray data," in *ICML*, 2001. 21

[27] P. Mundra and J. Rajapakse, "SVM-RFE with MRMR filter for gene selection," *IEEE Transactions on NanoBioscience*, vol. 9, no. 1, pp. 31–37, Mar. 2010. [Online]. Available: https://doi.org/10.1109/tnb.2009.2035284 21

[28] J. Jäger, R. Sengupta, and W. L. Ruzzo, "Improved gene selection for classification of microarrays," in *Biocomputing 2003*. World Scientific, 2002, pp. 53–64. 21, 22

[29] C. Park and S.-B. Cho, "Evolutionary ensemble classifier for lymphoma and colon cancer classification," in *The 2003 Congress on Evolutionary Computation, 2003. CEC 03.* IEEE, 2003. [Online]. Available: https://doi.org/10.1109/cec.2003.1299385 22

[30] M. Babu and K. Sarkar, "A comparative study of gene selection methods for cancer classification using microarray data," in *2016 Second International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*. IEEE, Sep. 2016. [Online]. Available: https://doi.org/10.1109/icrcicn.2016.7813657 22, 23

[31] H. Alshamlan, G. Badr, and Y. Alohali, "A comparative study of cancer classification methods using microarray gene expression profile," in *Lecture Notes in Electrical Engineering*. Springer Singapore, Dec. 2013, pp. 389–398. [Online]. Available: https://doi.org/10.1007/978-981-4585-18-7_44 22, 23

[32] F. G. Zöllner, K. E. Emblem, and L. R. Schad, "SVM-based glioma grading: Optimization by feature reduction analysis," *Zeitschrift für Medizinische Physik*, vol. 22, no. 3, pp. 205–214, Sep. 2012. [Online]. Available: https://doi.org/10.1016/j.zemedi.2012.03.007 24

[33] H. ho Cho, S. hak Lee, J. Kim, and H. Park, "Classification of the glioma grading using radiomics analysis," *PeerJ*, vol. 6, p. e5982, Nov. 2018. [Online]. Available: https://doi.org/10.7717/peerj.5982 24

[34] S. Patil, G. Naik, K. Pai, and R. Gad, "Neural network based classification of glioma grade iii and grade iv," *International Conference on Engineering and Technology*, 2016. 24, 25

[35] S. Wang, F. Liu, Y. Wang, W. Fan, H. Zhao, L. Liu, C. Cen, X. Jiang, M. Sun, and P. Han, "Integrated analysis of 34 microarray datasets reveals CBX3 as a diagnostic and prognostic biomarker in glioblastoma," *Journal of Translational Medicine*, vol. 17, no. 1, May 2019. [Online]. Available: https://doi.org/10.1186/s12967-019-1930-3 25

[36] K. Kaplan, Y. Kaya, M. Kuncan, and H. M. Ertunç, "Brain tumor classification using modified local binary patterns (LBP) feature extraction methods," *Medical Hypotheses*, vol. 139, p. 109696, Jun. 2020. [Online]. Available: https://doi.org/10.1016/j.mehy.2020.109696 25

[37] M. Sharif, J. Amin, M. Raza, M. A. Anjum, H. Afzal, and S. A. Shad, "Brain tumor detection based on extreme learning," *Neural Computing and Applications*, Jan. 2020. [Online]. Available: https://doi.org/10.1007/s00521-019-04679-8 25

[38] B. Niu, C. Liang, Y. Lu, M. Zhao, Q. Chen, Y. Zhang, L. Zheng, and K.-C. Chou, "Glioma stages prediction based on machine learning algorithm combined with protein-protein interaction networks," *Genomics*, vol. 112, no. 1, pp. 837–847, Jan. 2020. [Online]. Available: https://doi.org/10.1016/j.ygeno.2019.05.024 26

[39] T. R. Golub, D. K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J. P. Mesirov, H. Coller, M. L. Loh, J. R. Downing, M. A. Caligiuri, C. D. Bloomfield, and E. S. Lander, "Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring," *Science*, vol. 286, no. 5439, pp. 531–537, Oct. 1999. [Online]. Available: https://doi.org/10.1126/science.286.5439.531 26

[40] A. K. Dwivedi, "Artificial neural network model for effective cancer classification using microarray gene expression data," *Neural Computing and Applications*, vol. 29, no. 12, pp. 1545–1554, Nov. 2016. [Online]. Available: https://doi.org/10.1007/s00521-016-2701-1 26

[41] P. Guillen and J. Ebalunode, "Cancer classification based on microarray gene expression data using deep learning," in *2016 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE, Dec. 2016. [Online]. Available: https://doi.org/10.1109/csci.2016.0270 27

[42] J. Wu, "Feature selection for cancer classification using microarray gene expression data," *Biostatistics and Biometrics Open Access Journal*, vol. 1, no. 2, Apr. 2017. [Online]. Available: https://doi.org/10.19080/bboaj.2017.01.555557 27

[43] Y. Li, K. Kang, J. M. Krahn, N. Croutwater, K. Lee, D. M. Umbach, and L. Li, "A comprehensive genomic pan-cancer classification using the cancer genome atlas gene expression data," *BMC Genomics*, vol. 18, no. 1, Jul. 2017. [Online]. Available: https://doi.org/10.1186/s12864-017-3906-0 27, 28

[44] H. Liu, L. Liu, and H. Zhang, "Ensemble gene selection for cancer classification," *Pattern Recognition*, vol. 43, no. 8, pp. 2763–2772, Aug. 2010. [Online]. Available: https://doi.org/10.1016/j.patcog.2010.02.008 28

[45] T. J. Crisman, I. Zelaya, D. R. Laks, Y. Zhao, R. Kawaguchi, F. Gao, H. I. Kornblum, and G. Coppola, "Identification of an efficient gene expression panel for glioblastoma classification," *PLOS ONE*, vol. 11, no. 11, p. e0164649, Nov. 2016. [Online]. Available: https://doi.org/10.1371/journal.pone.0164649 28

[46] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002. [Online]. Available: https://doi.org/10.1613/jair.953 30

[47] T. Barrett, D. B. Troup, S. E. Wilhite, P. Ledoux, D. Rudnev, C. Evangelista, I. F. Kim, A. Soboleva, M. Tomashevsky, K. A. Marshall, K. H. Phillippy, P. M. Sherman, R. N. Muertter, and R. Edgar, "NCBI GEO: archive for high-throughput functional genomic data," *Nucleic Acids Research*, vol. 37, no. Database, pp. D885–D890, Jan. 2009. [Online]. Available: https://doi.org/10.1093/nar/gkn764 33

[48] S. learn Developers, *Principal component analysis (PCA)*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html 40

[49] ——, *Kernel Principal component analysis (KPCA)*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.KernelPCA.html 40

[50] ——, *Independent Component Analysis*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.FastICA.html 41

[51] ——, *Factor Analysis*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.FactorAnalysis.html 41

[52] ——, *Decision Tree lassifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html#sklearn.tree.DecisionTreeClassifier 41

[53] ——, *Random Forest Classifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html 41

[54] ——, *Bagging classifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingClassifier.html 42

[55] ——, *Gradient Boosting for classification*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html 42

[56] ——, *Gaussian Naive Bayes (GaussianNB)*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.naive_bayes.GaussianNB.html#sklearn.naive_bayes.GaussianNB 42

[57] ——, *Multi-layer Perceptron classifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html 42

[58] ——, *Support Vector Classifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html#sklearn.svm.SVC 42

[59] ——, *Logistic Regression*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html 43

[60] ——, *KNeighbors Classifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html 43

[61] ——, *AdaBoost classifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html 43

[62] ——, *Linear Discriminant Analysis*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.discriminant_analysis.LinearDiscriminantAnalysis.html 43

[63] ——, *Extra-Trees Classifier*. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.ExtraTreesClassifier.html 43

[64] V. Bewick, L. Cheek, and J. Ball, "Statistics review 12: Survival analysis," *Critical Care*, vol. 8, no. 5, p. 389, 2004. [Online]. Available: https://doi.org/10.1186/cc2955 74

[65] H. Gittleman, A. E. Sloan, and J. S. Barnholtz-Sloan, "An independently validated survival nomogram for lower-grade glioma," *Neuro-Oncology*, Oct. 2019. [Online]. Available: https://doi.org/10.1093/neuonc/noz191 75, 77

[66] U. of Tartu, *Bioinformatics, Algorithmics and Data Mining group BIIT*. [Online]. Available: https://biit.cs.ut.ee/funcexplorer/user/83dfcffcba4067a43dc53025dbb1620a/GSE11121_funcExpl?corr=a&cut=first&dist=p&gpversion=r1741_e90_eg37&max=1000&max_term_size=700&min=5&min_term_size=5&sparse=0&terms=BP-CC-MF&thr=1e-07 94, 95

Appendix A

Top 20 selected features

A.1 Freije Dataset

Affymetrix Probe ID	Gene Symbol	Gene name
201668_x_at	MARCKS	Myristoylated Alanine Rich Protein Kinase C Substrate[66]
202152_x_at	USF2	Upstream Transcription Factor 2, C-fos Interacting
202259_s_at	N4BP2L2	NEDD4 binding protein 2 like 2
202596_at	ENSA	Endosulfine Alpha
206107_at	RGS11	Regulator Of G Protein Signaling 11
206734_at	JRKL	JRK like
208334_at	NDST4	N-deacetylase And N-sulfotransferase 4[66]
210875_s_at	ZEB1	Zinc Finger E-box Binding Homeobox 1
211090_s_at	PRPF4B	Pre-mrna Processing Factor 4b
211094_s_at	NF1	Neurofibromin 1
212678_at	NF1	Neurofibromin 1
214250_at	NUMA1	nuclear mitotic apparatus protein 1[66]
215991_s_at	EMC1	ER membrane protein complex subunit 1[66]
217142_at		
217196_s_at	CAMSAP2	calmodulin regulated spectrin associated protein family member 2[66]
217987_at	ASNSD1	asparagine synthetase domain containing 1[66]
219312_s_at	ZBTB10	Zinc Finger And Btb Domain Containing 10
220055_at	ZNF287	zinc finger protein 287
220606_s_at	ADPRM	ADP-ribose/CDP-alcohol diphosphatase, manganese dependent[66]
49306_at	RASSF4	Ras association domain family member 4[66]

Table A.1: The 20 selected gene name for Freije dataset.

A.2 Phillips Dataset

Affymetrix Probe ID	Gene Symbol	Gene name
200734_s_at	ARF3	ADP ribosylation factor 3[66]
200768_s_at	MAT2A	methionine adenosyltransferase 2A
200886_s_at	PGAM1	phosphoglycerate mutase 1
200984_s_at	CD59	CD59 molecule (CD59 blood group)[66]
203390_s_at	KIF3C	kinesin family member 3C[66]
208678_at	ATP6V1E1	ATPase H+ transporting V1 subunit E1[66]
208785_s_at	MAP1LC3B	microtubule associated protein 1 light chain 3 beta[66]
208813_at	GOT1	glutamic-oxaloacetic transaminase 1
209311_at	BCL2L2	BCL2 like 2
210372_s_at	TPD52L1	TPD52 like 1
211323_s_at	ITPR1	inositol 1,4,5-trisphosphate receptor type 1[66]
211424_x_at	METTL7A	methyltransferase like 7A
212041_at	ATP6V0D1	ATPase H+ transporting V0 subunit d1[66]
214434_at	HSPA12A	heat shock protein family A (Hsp70) member 12A[66]
214553_s_at	ARPP19	cAMP regulated phosphoprotein 19[66]
217548_at	ARPIN	actin related protein 2/3 complex inhibitor[66]
219284_at	HSPBAP1	HSPB1 associated protein 1
219802_at	PYROXD1	pyridine nucleotide-disulphide oxidoreductase domain 1
220401_at	FLJ21369	
221504_s_at	ATP6V1H	ATPase H+ transporting V1 subunit H[66]

Table A.2: The 20 selected gene name for Phillips dataset.

Appendix B

Dataset Sample

Sample_title	brain, Right Frontal Lobe: Extract58_le1	brain, Right Frontal Lobe: Extract59_le1	brain, Right Frontal Lobe: Extract60_le1
Sample_geo_accession	GSM99432	GSM99434	GSM99436
Sample_status	Public on Mar 08 2006	Public on Mar 08 2006	Public on Mar 08 2006
Sample_submission_date	Mar 08 2006	Mar 08 2006	Mar 08 2006
Sample_last_update_date	Sep 11 2014	Sep 11 2014	Sep 11 2014
Sample_type	RNA	RNA	RNA
Sample_channel_count	1	1	1
Sample_source_name_ch1	brain, Right Frontal Lobe	brain, Right Frontal Lobe	brain, Right Frontal Lobe
Sample_organism_ch1	Homo sapiens	Homo sapiens	Homo sapiens
Sample_characteristics_ch1	tissue: brain, Right Frontal Lobe	tissue: brain, Right Frontal Lobe	tissue: brain, Right Frontal Lobe
Sample_characteristics_ch1	histology: GBM	histology: GBM	histology: GBM
Sample_characteristics_ch1	age: 49	age: 18	age: 64
Sample_characteristics_ch1	gender: MALE	gender: FEMALE	gender: FEMALE
Sample_characteristics_ch1	grade: 4	grade: 4	grade: 4
Sample_characteristics_ch1	living: DECEASED	living: DECEASED	living: DECEASED
Sample_characteristics_ch1	survival time: 188	survival time: 98	survival time: 356
Sample_characteristics_ch1	survival cluster: SC2	survival cluster: SC2	survival cluster: SC2
Sample_characteristics_ch1	hc_74_samples: HC2B	hc_74_samples: HC2A	hc_74_samples: HC2B
Sample_characteristics_ch1	hc_85_samples: HC2B	hc_85_samples: HC2A	hc_85_samples: HC2B
Sample_characteristics_ch1	array: 133A	array: 133A	array: 133A
Sample_molecule_ch1	total RNA	total RNA	total RNA
Sample_description	GBM	GBM	GBM
Sample_description	Treatment, type=remove	Treatment, type=remove	Treatment, type=remove
Sample_platform_id	GPL96	GPL96	GPL96
Sample_data_row_count	22283	22283	22283
ID_REF	GSM99432	GSM99434	GSM99436
1007_s_at	24052.068	25781.375	29665.607
1053_at	1426.0499	911.12646	1713.9501
117_at	1528.1367	1609.5969	1143.108
121_at	5264.1978	5782.814	4745.39
1255_g_at	301.99957	491.45572	273.6372
1294_at	2038.3907	1590.8643	2410.7683
1316_at	709.11957	818.83203	825.465
1320_at	580.4482	181.79169	522.3024
1405_i_at	58.270237	325.0617	58.962086
1431_at	294.53876	626.75934	256.80264
1438_at	2171.9753	1611.5458	1835.3738
1487_at	2547.4182	2630.8496	2268.2163
1494_f_at	1320.2832	1665.4812	1047.8127
1598_g_at	7864.3657	5536.349	4686.2954

Table B.1: Freije dataset sample.