GENE SELECTION FOR CANCER CLASSIFICATION BASED ON XGBoost CLASSIFIER

TEO VOON CHUAN

Bachelor of Computer Science (Software Engineering) with Honours

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : TEO VOON CHUAN__________________________________

Date of Birth : 15 NOVEMBER 1999__________________________

Title : GENE SELECTION FOR CANCER CLASSIFICATION

BASED ON XGBoost CLASSiFIER

____

Academic Session : SEMESTER 2 2021/2022_________________________

I declare that this thesis is classified as:

□	CONFIDENTIAL	(Contains confidential information under the Official Secret Act 1997)*
□	RESTRICTED	(Contains restricted information as specified by the organization where research was done)*
☑	OPEN ACCESS	I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

____
(Student's Signature)

____991115-08-5383____
New IC/Passport Number
Date: 1/6/2022

____
(Supervisor's Signature)

TS. DR. KOHBALAN A/L MOORTHY
SENIOR LECTURER
FACULTY OF COMPUTING
COLLEGE OF COMPUTING & APPLIED SCIENCES
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG DARUL MAKMUR
TEL : 09-424 4681 FAX : 09-424 4666

Name of Supervisor
Date: 16/02/2023

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

____
(Supervisor's Signature)

Date:

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

UMP Universiti Malaysia PAHANG
Engineering • Technology • Creativity

SUPERVISOR'S DECLARATION

I/We* hereby declare that I/We* have checked this thesis/project* and in my/our* opinion, this thesis/project* is adequate in terms of scope and quality for the award of the degree of *Doctor of Philosophy/ Master of Engineering/ Master of Science in …………………………..

____

(Supervisor's Signature)

Full Name : TS. DR. KOHBALAN A/L MOORTHY
SENIOR LECTURER
FACULTY OF COMPUTING
COLLEGE OF COMPUTING & APPLIED SCIENCES
Position : UNIVERSITI MALAYSIA PAHANG
26600 PEKAN, PAHANG DARUL MAKMUR
TEL : 09-424 4681 FAX : 09-424 4666
Date : 16/02/2023

____

(Co-supervisor's Signature)

Full Name :
Position :
Date :

UMP
Universiti
Malaysia
PAHANG
Engineering • Technology • Creativity

STUDENT’S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

(Student’s Signature)

Full Name : TEO VOON CHUAN

ID Number : 991115-08-5383

Date : 1 June 2022

GENE SELECTION FOR CANCER CLASSIFICAITON BASED ON XGBoost CLASSIFIER

TEO VOON CHUAN

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Bachelor of Computer Science (Software Engineering) with Honours

Faculty of Computing

UNIVERSITI MALAYSIA PAHANG

JUNE 2022

ACKNOWLEDGEMENTS

First of all, I would like to express my deep gratitude to my research supervisor, TS. DR. KOHBALAN A/L MOORTHY, for his patient guidance, valuable suggestions and assistance in helping me on planning and development for this research work. The time that he willingness to offer for me has been greatly appreciated.

I would also like to thank to my friends, who willing to provide assistance, share their experience and offer their help for me during the development of the research work.

Finally, I would also like to thank to my parents for their encouragement and support throughout my research work.

ABSTRAK

Pemilihan gen ialah teknik yang digunakan pada dataset pemilihan gen, seperti DNA microarray, yang dibangunkan untuk mengurangkan gen yang kurang bermaklumat, supaya gen yang dipilih berkaitan dengan diagnosis penyakit. Manakala klasifikasi kanser pula adalah satu proses mengenal pasti jenis kanser, dan sejauh mana tumor telah membesar dan merebak. Pengelas XGBoost digunakan dalam penyelidikan ini, yang merupakan pelaksanaan sumber terbuka yang cekap bagi algoritma pepohon yang dirangsang kecerunan. Peningkatan kecerunan ialah algoritma pembelajaran yang diselia, yang cuba meramalkan pembolehubah sasaran dengan tepat dengan menggabungkan anggaran set model yang lebih mudah dan lemah. Dalam dunia hari ini, penyakit kanser masih menjadi punca utama kematian kepada manusia. Masalah mempunyai halangan untuk membuat pengesahan awal penyakit kanser masih memberi kesukaran kepada pengkaji. Disebabkan dengan keadaan ini, pembanguanna kaedah pemilihan gen telah menjadi lebih penting dalam mendapatkan maklumat yang berguna untuk klasifikasi penyakit kanser, dan diagnosis untuk penyakit lain. Oleh itu, Pengelas XGBoost dicadangkan dalam penyelidikan ini, untuk membantu memileh subset gen minimum yang relevan untuk klasifikasi penyakit kanser. Dengan mengunakan Pengelas XGBoost, ketepatan dan prestasi pemilihan gen dan klasifikasi penyakit kanser boleh dipertingkatkan dengan sangat baik, dan mengurangkan masa dan kos untuk diagnosis penyakit. Kesimpulannya, Pengelas XGBoost meningkatkan prestasi dan ketepatan dalam pemilihan gen untuk klasifikasi kanser.

ABSTRACT

Gene selection is the technique that applied to the gene selection dataset, such as DNA microarray, which is develop to reduce the less informative gene, so that the selected gene is related to the disease diagnosis. While the cancer classification is a process of identifying the type of cancer, and the extent to which a tumor has grown and spread. XGBoost Classifier is applied in this research, which it is an efficient open-source implementation of the gradient boosted trees algorithm. Gradient boosting is a supervised learning algorithm, which attempts to accurately predict a target variable by combining the estimates of a set of simpler, weaker models. In today world, cancer is still as a leading cause to death. The problem of having obstacle to making early detection for cancer is still a difficulty for the researcher. Due to this situation, development of the gene selection method has become more important in obtain useful information for cancer classification, and diagnoses for other disease. Thus, a XGBoost Classifier is proposed in this research, to help to select minimum gene subset that are giving relevant information for cancer classification. By applied the XGBoost Classifier, the accuracy and the performance of the gene selection and cancer classification can be highly improved, and reduce the time and cost for the disease diagnoses. In conclusion, XGBoost Classifier is increasing the performance and accuracy in gene selection for cancer classification.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

x	Normalized value in the dataset.

LIST OF ABBREVIATIONS

WHO	World Health Organization
ABC	Artificial Bee Colony
DNA	Deoxyribonucleic Acid
MNN	Mutual Nearest Neighbor
GEP	Gene Expression Programming
NBC	Naïve Bayesian Classifiers
KNN	K-Nearest Neighbors
SVM	Support Vector Machines
GAABC	Genetic Algorithm with Artificial Bee Colony
GA	Genetic Algorithm
LOOCV	Leave One Out Cross-Validation
RFE	Recursive Feature Elimination
RFECV	Recursive Feature Elimination with Cross-Validation
PCA	Principle Component Analysis
2D	Two-Dimensional
3D	Three-Dimensional
SVD	Singular Value Decomposition

CHAPTER 1

INTRODUCTION

1.1 Introduction

In this era of globalization, cancer still ranks as a leading cause of death, no matter in what country. According to the data from World Health Organization (2022), there are nearly 10 million deaths in 2020 because of the cancer. The most common cancer in 2020 are breast, lung, and prostate cancers (World Health Organization, 2022). Although there are many types of cancer, but it can be cure if the type of the cancer is detected and treated early. Thus, gene selection can be important in helping the researcher to make classification on cancer. Other than that, with the assist of XGBoost Classifier, the accuracy of the cancer classification can be higher, so that the type of the cancer can be detected with more accuracy, and the patient can receive the most effective treatment to be cure as much as possible.

Gene selection is the technique for reducing redundant and less expressive or informative genes in a gene expression dataset, such as a DNA microarray, so that the selected gene is directly related to the disease diagnosis. (Mahendran, N., Durai Raj Vincent, P. M., Srinivasan, K., & Chang, C. Y., 2020). Generally, microarray data typically contains thousands of genes (sometimes more than 10,000 genes) and a limited number of samples (usually less than 100 samples). (Liu, JX., Wang, YT., Zheng, CH. et al., 2013). Thus, it is important to using gene selections methods to find a set of relevant gene that are related with the disease. In simple word to explain about it, gene selection is applied for finding the informative and expressive genes that are relevant and significant to a clinical diagnosis, such as cancer. Essentially, understanding the differences between cancer gene expression and normal gene expression can disclose more useful and relevant information. (Alanni, R., Hou, J., Azzawi, H. et al., 2019).

Cancer classification is the process of identifying the type of cancer, and the extent to which a tumour has grown and spread. Basically, cancers are classified by the type of tissue from which it arises, which mean that it is based on histological type (Cancer Classification | SEER Training, n.d.). Based on the histology or the tissue type, there are hundreds of distinct cancers that can be categorised into six broad categories based on histology or tissue type, such as Carcinoma, Leukaemia, Lymphoma, Mixed Types, Myeloma, and Sarcoma (Nall, 2018). Each type of cancer cell has a particular look that help the researcher to make differentiation for tumours or cancer (Nall, 2018). Thus, cancer classification can help researcher in developing an efficient treatment plan for the patient.

XGBoost Classifier is a gradient boosted decision tree implementation created for speed and performance. Briefly Extreme Gradient Boosting (XGBoost) is a distributed, scalable gradient-boosted decision tree (GBDT) machine learning framework. It offers parallel tree boosting and it is the top machine learning library for regression, classification, and ranking issues. Then, the algorithm can assist in the selection of the smallest number of genes that are relevant to cancer tissues, as well as increase prediction accuracy. Thus, the performance in gene selection for cancer classification can be improve, and be more accurate in identify the certain type of cancer, so that the patient can receive an effective treatment as soon as possible.

1.2 Problem Background

In this era, there are some of the problems that regarding the gene selection for cancer classification, and the most important problems that always occur is the accuracy and the early detection of the cancer classification. Traditionally, cancer nomenclature has been focused mostly on organ location; for example, “lung cancer” refers to a tumour that originates in the lung tissues (Song et al., 2015). Thus, most of the cancer is detected in its later stages due to accuracy problem, which cause the function of one or more organ systems compromised or malfunction, and hard to achieve a cure even after receive treatment.

Gene selection is intended to enable us to select the most relevant genes that help in diagnose tumours precisely and systematically. However, the task of cancer classification has become particularly tough in this situation due to the high-dimensional noise in gene expression profiles and the problem of small sample size. This means that a dataset will always have thousands of genes but just a few samples, and the majority of these genes will be irrelevant to the classification task. Essentially, including all genes during analysis could hinder classification performance by concealing the contribution of informative genes. (Alanni, R., Hou, J., Azzawi, H. et al., 2019). Thus, decrease the accuracy of the cancer classification.

1.3 Problem Statement

Based on the problem background, the problem statement of this research is:

1. The researcher having obstacle on making early detection on the type of cancer due to the higher amount of type of cancer.

2. The researcher always chose genes that are irrelevant to the cancer classification because of the gene or microarray data has its own high dimensionality issue.

3. The researcher is having difficulty dealing with irrelevant and misleading gene expression samples, which complicates the cancer classification process and increases the time and cost of finding the most associated genes.

1.4 Aim & Objective

The aim for this research is to minimize the irrelevant and less informative genes that are selected through the gene selection method. Thus, it can increase the accuracy of the cancer and reduce the time and cost of cancer classification and gene selection.

The objective for this research is:

1. To study the cancer classification based on gene selection approaches.

2. To implement XGBoost Classifier in the gene selection process for cancer classification.

3. To verify the performance of the gene selection for cancer classification based on XGBoost Classifier.

1.5 Research Scope

The scope of this research is:

1. The purpose of the study is only focus on selected the most relevant and informative gene to improve the accuracy of the cancer classification based on XGBoost Classifier.

2. The cancer sample or dataset that are studying in this research will be focus only on breast cancer.

3. The topics that will discuss in this research are limited on cancer classification with gene selection, and based on XGBoost Classifier.

1.6 Significant of Research

The importance of the research and the contribution of the research are:

1. Discover a new possible way to classify cancer using gene selection, along with XGBoost Classifier, so that cancer patient can get the best treatment available as soon as feasible.

2. Assist researchers in reducing the time and cost of selecting genes for cancer classification, as well as increasing the accuracy of cancer classification.

3. Assist the researcher in improving the performance of the gene selection for cancer classification, so that gene expression data may provide more accurate and relevant information, and treatment for cancer patients can be predicted.

1.7 Summary

In conclusion, this research is about the gene selection for cancer classification based on XGBoost Classifier. Its goal is to enhance the performance of gene selection for cancer classification, so that cancer patients can obtain appropriate treatments as soon as feasible, since early identification is one of the most significant factors in ensuring a patient's full recovery. This research also intends to reduce the time and expense of gene selection for cancer classification for researchers, as well as increase the accuracy of cancer classification. Thus, it allows to predict the most suitable treatment for the patient, and increase the cancer diagnosis accuracy, since every cancer type requires a specific treatment.

CHAPTER 2

LITERATURE REVIEW

2.1 Overview

In this chapter, we will be going to review the existing case studies that are related with the title that proposed before, which is Gene Selection for Cancer Classification Based on XGBoost Classifier. First, there are three keywords that will be discuss detail in this chapter based on the proposed title, which is Gene Selection, Cancer Classification and XGBoost Classifier. Lastly, there are a comparative analysis that will compare three existing case studies that are related to the topic. The three existing case studies that will be used for review and comparative are mainly focus on the gene selection for cancer classification.

2.2 Gene Selection

Gene selection is the approach that are used on gene expression datasets, for generating a related, or nearly related populations of differentiated cell types (Mahendran, N., Durai Raj Vincent, P. M., Srinivasan, K., & Chang, C. Y., 2020). In another word, gene selection is basically a method that reduce the number of genes, such as DNA microarray, which are redundant and not relevant. This methods or methodology has the capability of identifying the most relevant and informative genes, as well as assisting in the diagnosis of cancer based on the expression of the selected genes. It is useful in discover specific genes, and analyse the gene expression with specific diagnosis to determine the disease on the patients, so that the most effective treatment can be given to the patient.

Basically, gene selection technique is only applied for finding the informative genes, and removing the redundant genes, so that the most relevant gene can be find out. It really helps the researcher to analyse thousands of genes in an experiment quickly and in an efficient way. However, gene expression data, on the other hand, typically contains a large number of genes and a small number of samples to serve as references. (Alanni, R., Hou, J., Azzawi, H. et al., 2019). This has reduced the performance of the classification accuracy, and consume more time and cost.

D. Pavithra and B. Lakshmanan (2017) has been proposed feature selection in gene expression cancer data. Feature selection is a method for reducing irrelevant and redundant characteristics and selecting the optimal collection of features for better categorization of patterns belonging to different classes. Briefly, filter and wrapper methods are two types of feature selection approaches. Filter approaches rely on common aspects of the training records to select a small number of features without requiring the use of another Learning Algorithm. The other is about wrapper model, which requires a single predetermined learning algorithm, and its performance is utilized to compute and forecast which features are elected.

H. L. Shashirekha and A. H. Wani (2015) has been proposed gene selection by mutual nearest neighbour approach. In this study, for selecting the most important genes from a huge set of gene expression data, the Mutual Nearest Neighbour (MNN) algorithm is proposed. The MNN is a filter strategy that selects a subset of features using distance metrics. Aside from that, the Mean test is used as a filter for identifying important genes based on their mean value, with genes ranked by mean value and genes with lower mean values being removed.

Alanni, R., Hou, J., Azzawi, H. et al. (2019) has presented a deep gene selection method to select genes from microarray datasets to make classification on disease. For this research studies, its goal is to develop an efficient gene selection strategy for identifying the most relevant genes that are sensitive to the existing sample and producing the most accurate cancer classification results. In this research studies, the Deep Gene Selection is developed based on the gene expression programming (GEP) algorithm for solving the problem of microarray gene expression data selection.

J. Ge, X. Zhang, G. Liu and Y. Sun (2019) has been proposed a novel feature selection algorithm based on Artificial Bee Colony Algorithm and Genetic Algorithm. The research studies are aim to solve the high dimensionality problem in microarray data classification, by improving the variety of bee populations. Thus, a feature selection algorithm based on Artificial Bee Colony Algorithm and Genetic Algorithm is presented. This have result in improvement of the classification performance for the high dimensional microarray data and small sample data.

2.3 Cancer Classification

Cancer classification aim to find out and detect the type of cancer, or tumours for a specific patient, so that they can receive the correct treatment as soon as possible. Since early detection on cancer result in more possibility to survival and cure, then the cancer classification process needs to be more efficient and effective. Thus, cancer classification is important to determine the suitable treatment for the patient based on the result.

Usually there are involve many steps to make a classification on the type of cancer. For example, it needs nine characteristics or criteria to make a breast cancer classification, such as determine the layered structures, analyse the sample and so on (M. Amrane, S. Oukid, I. Gagaoua and T. Ensarİ, 2018). Some of this step is complicated, and some of these criteria cover very large of the scope. Thus, a suitable method along with cancer classification is needed in order to reduce the time needed. Compare with traditional cancer classification method, some of the method such as gene expression profiles can be used for making a more accurate and reliable cancer classification.

M. Amrane, S. Oukid, I. Gagaoua and T. Ensarİ (2018) has been proposed the breast cancer classification using machine learning. For the breast cancer classification, it is aim to find the best treatment for the cancer, which might be aggressive or less aggressive depending on the type of cancer. There are two types of machine learning approaches, which is supervised learning and unsupervised learning. In this research, Naïve Bayesian Classifiers (NBC) and K-Nearest Neighbours (KNN) are two supervised learning classifiers that are used in this study. In conclusion for the research, NBC has a best accuracy, while the KNN achieved a higher level of efficiency. However, if the dataset is too big, the running time for the KNN will increase.

W. Luo, L. Wang and J. Sun (2009) has been proposed the feature selection for cancer classification based on support vector machine. The mixed two-step feature selection method is proposed in this study, which uses a support vector machine for cancer classification. The first step is to pick discriminatory characteristics using a modified t-test method, and the second step is to extract primary components from the top-ranked genes using a modified t-test method. As a result, the two-step technique can achieve higher accuracy while using fewer genes.

L. Wang, F. Chu and W. Xie (2007) has been proposed an accurate cancer classification with very few of the gene’s expressions. Basically, the research studies, its goal is to use supervised machine learning techniques to find the smallest set of genes that can deliver highly accurate cancer classification from microarray data. The case studies provided a simple two-step strategy for discovering the smallest set of genes that may accurately classify cancer from microarray data. In the first step, a feature importance ranking technique is used to select a few key genes, and then a classifier is used to test the classification ability of all simple combinations of those significant genes in the second step. As a result, those method is highly reducing the number of genes that needed to making a reliable diagnosis on cancer.

H. Wang, H. Yu, Q. Zhang, S. Cang, W. Liao and F. Zhu (2016) has been proposed the parameters optimization of classifier and feature selection based on improved artificial bee colony algorithm. The research is aim to optimize the feature subset and the parameters of support vector machines (SVM), in order to ensure the optimal classification performance. Meanwhile, the initialization and scout bee phases have been enhanced to increase the ABC algorithm's optimizing performance. As a result, two dataset is included for verification, and the simulation results show that the proposed strategy has a greater classification accuracy and a smaller feature subset than other approaches.

Chen et al. (2021) has been proposed A Novel XGBoost Method to Infer the Primary Lesion of 20 Solid Tumour Types from Gene Expression Data. Chen et al. first employs two feature selection methods to pick genes with high expression levels from the dataset, and then use XGBoost for classification to be more precise and increase the diagnostic effectiveness of primary lesions. Even if the experimental results were reached, there were still some drawbacks. The procedure must include a parameter specifying the number of chosen genes.

In conclusion, the XGBoost Classifier have shown the better performance among the other method, such as Genetic Algorithm and so on. Based on the research from Deng (2021), the mean average execution time of the suggested methodology was faster than that of the majority of the gene selection techniques evaluated, and it outperformed all of the gene selection techniques by a sizeable margin.

2.4 A Comparative Analysis

For the comparative analysis, there are three related research studies that will be using for comparisons, which is:

1. Deep Gene Selection Method to Select Genes from Microarray Datasets for Cancer Classification (Alanni, R., Hou, J., Azzawi, H. et al., 2019).

2. Gene Selection by Mutual Nearest Neighbor approach (H. L. Shashirekha and A. H. Wani, 2015).

3. A Novel XGBoost Method to Infer the Primary Lesion of 20 Solid Tumor Types from Gene Expression Data. Frontiers. (Chen S, Zhou W, Tu J, 2021).

Below is the table that compare the three different research studies for the pros and cons.

Table 2.1 A Comparative Analysis For Three Research Studies

Comparison	**Deep Gene Selection Method**	**Mutual Nearest Neighbor Algorithm**	**A Novel XGBoost Method**
Aim	- Provide an effective approach for selecting the most relevant genes that are sensitive to the present sample.	- Attempt to improve the performance by eliminating redundant and unnecessary gene in order to locate a	- For primary metastatic tumours, identify its primary lesions using an integrated learning approach.

	- Produce the most accurate result on cancer classification.	very small number of significant genes.	- Making it more precise to increase the effectiveness of primary lesions' diagnostics.
Pros	- Provide the small number of relevant genes and have higher accuracy for cancer classification in less processing time. - Remove unrelated genes in each generation to increase the possibility of choosing related genes.	- Noted that the MNN algorithm performs better than that Mean Test, and other technique. - Show that the MNN algorithm provide more accurate classification result.	- Each malignancy has a corresponding classification accuracy that can be used to forecast the location of primary metastatic tumours by integrating tumour data with machine learning techniques. - Can be employed as an orthogonal diagnostic technique to evaluate the processing of a machine learning model and clinically relevant pathological situations.
Cons	- If the generation size is too large, the computational size will increase. - Generation size is too small will lead to the generation not cover all attributes.	- When compared with some of the other algorithm, MNN algorithm take more time for gene selection.	- Must include as a parameter the number of chosen genes.

As a result, XGBoost Classifier is used in this research, since the XGBoost Classifier is already showing the best performance among the other method when comparing with the other research studies.

XGBoost Classifier is an open-source software library which it is an implementation of gradient boosting machines. Basically, when it comes to classification and regression predictive modelling issues, XGBoost dominates structured or tabular datasets. To conclude, XGboost Classifier offers parallel tree boosting, and it is the top machine learning library for regression, classification, and ranking issues.

Decision trees are generated sequentially in this approach. Weights are significant in XGBoost. Each independent variable is given a weight before being fed into the decision tree that forecasts outcomes. Variables that the tree incorrectly predicted are given more weight before being placed into the second decision tree. These distinct classifiers/predictors are then combined to produce a robust and accurate model. Regression, classification, ranking, and user-defined prediction issues are all applicable.

As a conclusion, the XGBoost Classifier is applied in this research to find the most relevant gene that bring the high accuracy for the cancer classification.

2.5 Summary

In conclusion, this chapter is making the discussion in detail based on the title, such as gene selection, cancer classification and XGBoost Classifier. The discussion helps to make us to have a better understanding on the proposed title, and understand the subject of the topic or title. Then, the literature review continues on making comparison on the three existing research studies that related to the title or topic. As a result, those research studies have quiet similar aim, which is aim to provide more accurate classification on certain disease, or solve the high dimensional problem for the microarray data.

CHAPTER 3

METHODOLOGY

3.1 Overview

The research investigates how the XGBoost Classifier can assist in the selection of genes for cancer classification. Gene expression profiles always has great potential for disease diagnostic. However, the number of genes that needed to be compare always larger than the datasets, which mean that the datasets contain only small sample size for compare and making the disease classification. As a result, some unimportant and redundant genes always lower the classification quality, raising the false positive rates.

In this research, XGBoost Classifier is used to locate useful genes that can provide more relevant information for cancer classification. By going through the pre-processing, the most informative genes are left for cancer classification. The most informative genes are then processed again using the search approach to find a smaller group of informative genes that yield the highest accurate cancer classification result.

3.2 Research Framework

The research framework will be act as a research plan, and will be used as a guide for the researcher to focus on the scope of the research or study.

Below of the table is showing the research framework of this research. There is total five phase for the research framework in this research.

Table 3.1 Research Framework With Five Phase

Phase	Research Content
Phase 1	- Explore on the related work for the research, such as gene selection, cancer classification, XGBoost Classifier and the other related method and algorithm. - Have detail understanding on how the other researcher work on their selected algorithm and method. - Explore on how the researcher implementing, testing, and record the result for their selected method or algorithm.
Phase 2	- Explore on the methodology of the research. - Explore on the flow of the whole research. - Explore on how the XGBoost Classifier is working in this research. - Explore on the dataset that will be used in the research.
Phase 3	- Design on implement the XGBoost Classifier on the gene selection for cancer classification.
Phase 4	- Implement the XGBoost Classifier on the gene selection for cancer classification. - Testing the performance of the algorithm on difference dataset.
Phase 5	- Keep track of all the results and the classifier's performance in terms of gene selection for cancer classification. - Compare the performance of the classifier with the other relevant methods and algorithms. - Keep track all the record and make documentation.

3.3 Proposed Methodology

In this research, the XGboost Classifier is applied to find the most relevant and informative gene for cancer classification.

Below is the flowchart for the overall research flow based on the XGBoost Classifier.

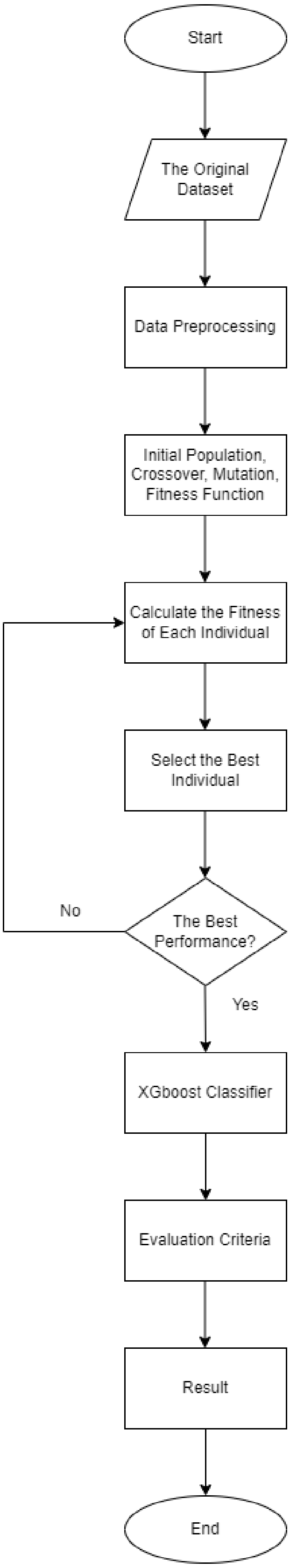

Start
The Original Dataset
Data Preprocessing
Initial Population, Crossover, Mutation, Fitness Function
Calculate the Fitness of Each Individual
Select the Best Individual
The Best Performance?
No
Yes
XGboost Classifier
Evaluation Criteria
Result
End

Figure 3.1 XGBoost Classifier

First is for the original dataset, where the dataset or initial feature subset is input, and the classifier is going to initialization.

Next is the data processing. Basically, the data processing is aim to reduce the training error, and providing the most accurate result for the classification problem. As a result, going through data processing ensures that the fitness metric provides equal weight to each variable.

The next step is to initial the population, crossover, mutation and fitness function. In this step, all the necessary element is set up and initialization for the next step.

Next is to calculate the fitness of each individual. In this step, calculates each individual's genes opportunity by weighing their advantages and drawbacks in relation to their particular fitness value. High fitness individuals have a better chance of surviving into the next generation.

The next is to select the best individual, where aim to reduce the irrelevant gene subset. Prefilter is used in this step to pre-select a small number of informative genes based on a set of filtering parameters.

Next is the step where to find the optimal gene subset using XGBoost Classifier. In this step, only the genes with scores larger than 0 were kept after the genes were rated using XGBoost Classifier. This step can successfully exclude unimportant genes and produce a collection of genes that are most pertinent to the class.

Next step is the evaluation criteria, which include using the Support Vector Machines (SVM) to evaluate the accuracy of the gene subset. SVM are used in this step to classify the gene subset's accuracy and access the fitness of a food position. SVM is well suited to classification tasks involving high-dimensional and small-sample data.

Finally, the result of the cancer classification is generated, such as the accuracy rate and graph.

3.4 Dataset

There is only one dataset is included in this research. The dataset that is included in this research is Breast Cancer Dataset.

There are total of 151 columns which represent sample and 54676 rows which represent genes. There is total 6 classes in the dataset, which is basal, HER, luminal_B, luminal_A, cell_line and normal.

The summary of the dataset is presented through the below table.

Table 3.2 Datasets

Name of the Dataset	**Sample Size**	**Number of Genes**	**Number of Classes**
Breast_GSE45827	151	54676	6

Source: Grisci, B. (2020).

3.5 Performance Measurement

The expected evaluation methods for the proposed XGboost Classifier in this research for verification or validation purpose include Accuracy, Precision, Recall, and F1 Score.

Before the explanation for performance measurement, there are some keywords needed to be explained. The keyword is True Positive, False Negative, False Positive and True Negative. Below is the explanation for each keyword.

1. True Positive mean that there are how often did the model accurately categorise a Positive sample as Positive.

2. False Negative mean that there are how often did the model inaccurately categorise a Positive sample as Negative.

3. False Positive mean that there are how often did the model inaccurately categorise a Negative sample as Positive.

4. True Negative mean that there are how often did the model accurately categorise a Negative sample as negative.

For the accuracy, it is the most logical way to assess the success of any classification algorithm by calculating the percentage of right predictions. Basically, it is the indicator that includes the percentage of accurate predictions made by a model. Additionally, it uses a single value to assess the model's performance.

Below is the Accuracy score formula.

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Number\ of\ Predictions} \qquad 3.1$$

Below is the more detail formula for the Accuracy score.

$$Accuracy = \frac{TrueNegatives + TruePositive}{TruePositive + FalsePostive + TrueNegative + FalseNegative} \qquad 3.2$$

For the precision, it is to measures the proportion of predictions in the Positive class that are confirmed by ground truth to be Positive. Basically, precision assesses a classifier's capacity not to classify a negative sample as positive.

Below is the Precision formula.

$$Precision = \frac{TruePostive}{TruePostive + FalsePostive} \qquad 3.3$$

For the recall, it is to measures the proportion of Positive class predictions that match the ground truth to all Positive samples. In a simple word, it is to assesses a classifier's capacity to identify positive samples.

Below is the Recall formula.

$$Recall = \frac{TruePostive}{TruePostive + FalseNegative} \qquad 3.4$$

For the F1 score, it is represented that the percentage of the positive prediction that are correct. In other word, it is a weighted harmonic mean of recall and precision, with 1.0 representing the best result and 0.0 the lowest.

Below is the F1 score formula.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{3.5}$$

3.6 Hardware & Software Requirement

In this research, the hardware requirement for the experimental setup is listed below:

Table 3.3 Hardware Requirement

Central Processing Unit (CPU)	Intel Core i5 6th Generation processor or higher.
RAM	Minimum 8 GB. Recommended 16 GB or higher.
Graphics Processing Unit (GPU)	NVIDIA GeForce GTX 960 or higher.
Operating System	Ubuntu or Microsoft Windows 10.

3.7 Summary

In conclusion, this chapter is having a detail discussion on the methodology for the research. This chapter help the researcher to have a detail understanding about the way to perform the experimental setup, and how the algorithm is working in this research. As a result, this chapter are focus on explaining the methodological approach, which is XGBoost Classifier, and data collection, and expected evaluate for the methodological choice.

CHAPTER 4

DESIGN AND IMPLEMENTATION

4.1 Introduction

Chapter 4 will discuss about the implementation of the Gene Selection for Cancer Classification with XGBoost Classifier. This chapter contains the discussion of the finding based on implantation of the method, and testing that have been done. It also includes the explanation for the method used, which is the XGBoost classifier.

4.2 Implementation Process

XGBoost classifier was implement in this research, and the breast cancer gene expression dataset are used in this research. For the dataset, there are 6 classes, 54676 genes and 151 samples for the breast cancer.

The 6 classes that are include in the breast cancer gene expression dataset is:

1. Basal
2. HER
3. luminal_B
4. luminal_A
5. cell_line
6. normal.

All the code are writing in python and running on the Google Collaboratory platform.

4.2.1 Implementation of The XGBoost Classifier

There are several steps include for implement the XGBoost classifier with the gene selection for cancer classification. Below is the flowchart that represent the implementation of the XGBoost with gene selection for cancer classification. The box that highlights as blue is the contribution that are newly added.

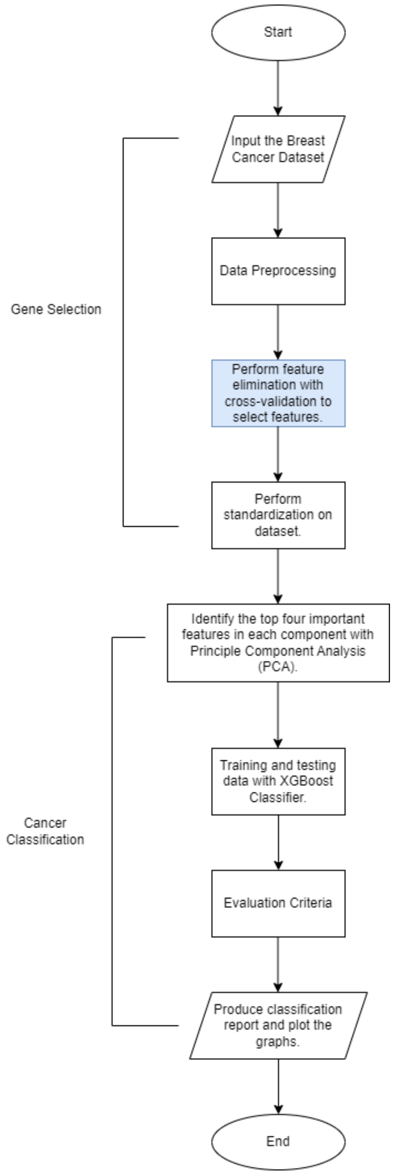

Start
Input the Breast Cancer Dataset
Data Preprocessing
Gene Selection
Perform feature elimination with cross-validation to select features.
Perform standardization on dataset.
Identify the top four important features in each component with Principle Component Analysis (PCA).
Training and testing data with XGBoost Classifier.
Cancer Classification
Evaluation Criteria
Produce classification report and plot the graphs.
End

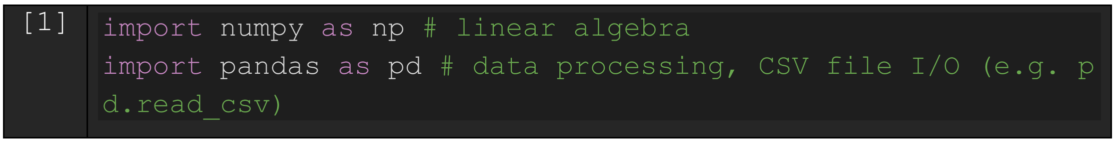
Figure 4.1 Gene Selection for Cancer Classification Based on XGBoost and Recursive Features Elimination with Cross-Validation

The detail explanation of the flow and python code are explained below.

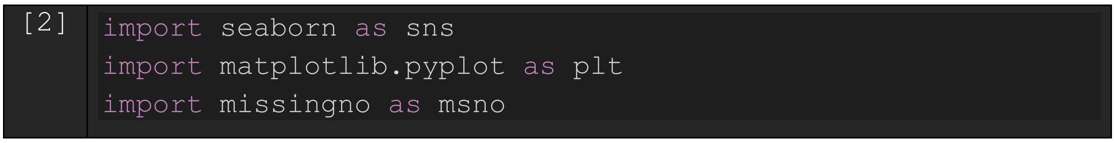
4.2.1.1 Input the breast cancer dataset and initialize the population.

In this step, the breast cancer gene expression dataset is input for the classification process.

For row 1 and row 2, there are two module is import which is numpy and pandas. This is use for linear algebra and data processing such as input the dataset. The other module is, the seaborn, matplotlib, and missingno module which is import for plotting the graphs that will be produce in the end of the code.

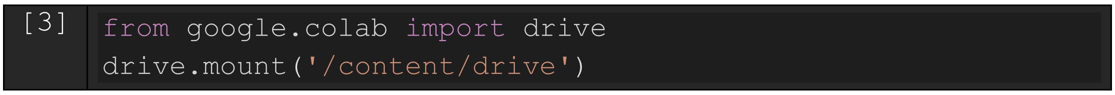


```
[1]  import numpy as np # linear algebra
     import pandas as pd # data processing, CSV file I/O (e.g. p
     d.read_csv)
```

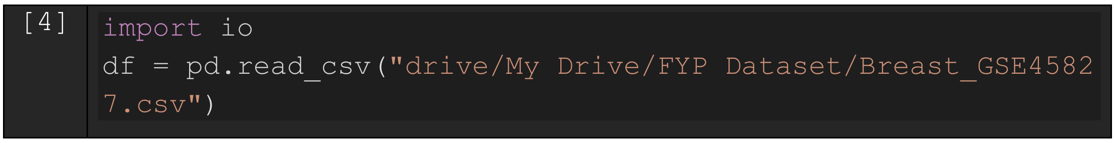


```
[2]  import seaborn as sns
     import matplotlib.pyplot as plt
     import missingno as msno
```

For row 3 and row 4, this is to given permission for the Google Colab notebook to access the Google Drive, to input the dataset that are save in the Google Drive. In here, the Breast_GSE45827.csv dataset is input for classification.

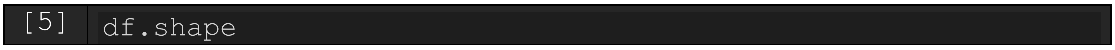


```
[3]  from google.colab import drive
     drive.mount('/content/drive')
```



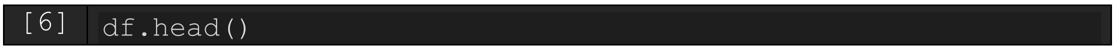
```
[4]  import io
     df = pd.read_csv("drive/My Drive/FYP Dataset/Breast_GSE4582
     7.csv")
```

For row 5, the number of row and columns is display. This indicates that there are 151 sample for row, 54676 genes expression for column, and 1 row for the label of the breast cancer type.

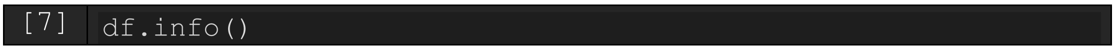


```
[5]  df.shape
```

For row 6, display the information for the first five row in the dataset. While for row 7, display the full summary of the dataset, which include the range index, columns, data types and so on.

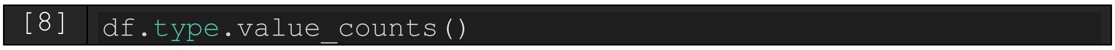


```
[6]  df.head()
```



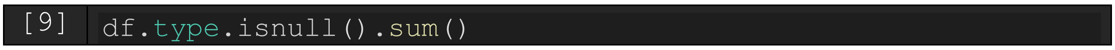
```
[7]  df.info()
```

For row 8, identify and calculate the type of breast cancer in the dataset. For the above image, there are 41 breast cancer which is basal, 30 for HER, 30 for luminal_B, 30 for luminal 29, 14 for cell_line, and 7 for normal which stand for healthy tissue.



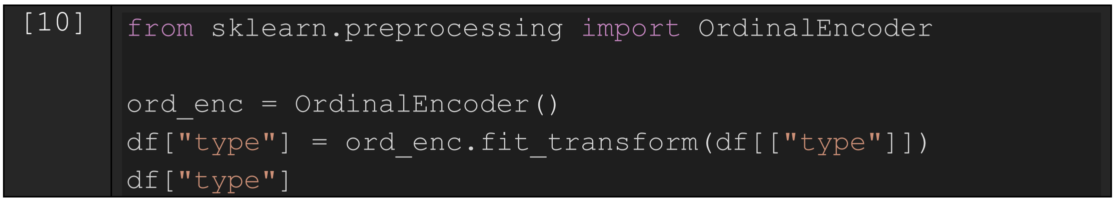
```
[8]  df.type.value_counts()
```

For row 9, count the missing value in the type column. From the above image, there are no missing value in the type column.



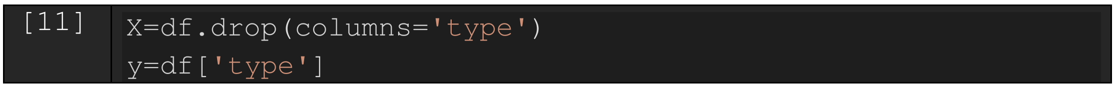
```
[9]  df.type.isnull().sum()
```

For row 10, the categories type is replaced by ordinal numbers such as one, two and so on.

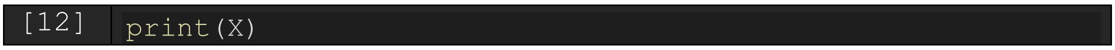


```
[10]  from sklearn.preprocessing import OrdinalEncoder

      ord_enc = OrdinalEncoder()
      df["type"] = ord_enc.fit_transform(df[["type"]])
      df["type"]
```

For row 11, the type columns are removed, and the remain data is store in variable X. This is to avoid confuse in calculating and making classification on the number of gene. While the variable y stores the information from type column. After this, in row 12, output the variable X which store the dataset information that are without type column.

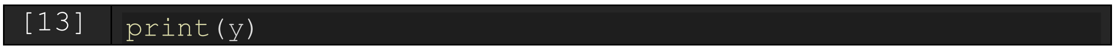


```
[11]  X=df.drop(columns='type')
      y=df['type']
```



```
[12]  print(X)
```

For row 13, output the variable y which store the type column information for each row of the sample.



```
[13]  print(y)
```

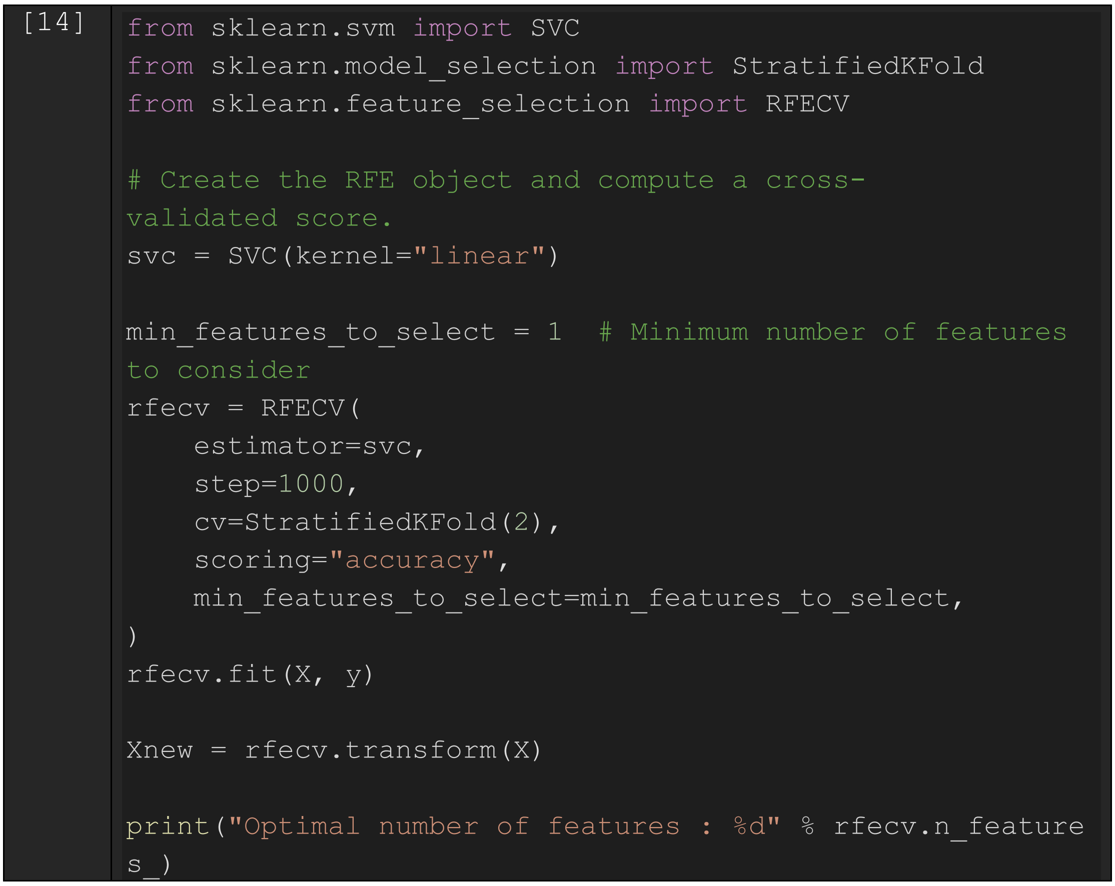
4.2.1.2 Perform recursive feature elimination with cross-validation (RFECV) on dataset.

For this step, this is the part that are added for contribution, which mean that this is the part there are not existing in the previous work. The purpose for adding the recursive feature elimination with cross-validation (RFECV) is to try for increase the cancer classification accuracy with the gene selection.

In this step, recursive feature elimination with cross validation (RFECV) is perform on the dataset to minimize the number of genes. Since there are 54676 genes in the dataset, then it is important to filter out some of the not important genes to improve the performance of the cancer classification process. Thus, the goal of RFECV is to select the important features by performs recursive feature elimination (RFE) in a cross-validation loop.

For row 14, there are three modules import for perform the RFECV, which is Support Vector Classification (SVC), Stratified K-Folds cross-validator

(StratifiedKFold), and RFECV. This is the necessary module for performing the RFECV on the breast cancer dataset.



```
from sklearn.svm import SVC
from sklearn.model_selection import StratifiedKFold
from sklearn.feature_selection import RFECV

# Create the RFE object and compute a cross-
validated score.
svc = SVC(kernel="linear")

min_features_to_select = 1  # Minimum number of features
to consider
rfecv = RFECV(
    estimator=svc,
    step=1000,
    cv=StratifiedKFold(2),
    scoring="accuracy",
    min_features_to_select=min_features_to_select,
)
rfecv.fit(X, y)

Xnew = rfecv.transform(X)

print("Optimal number of features : %d" % rfecv.n_feature
s_)
```

Then, create the RFE object, which is the estimator for minimize the number of genes. The estimator here means the fitted estimator used to select features. After this, the minimum number to be selected and consider as important features are remain the default setting, which is 1.

In the RFECV model, the estimator is defined. The step is defined as 1000 to increase the speed of the process due to the high amount of the number of genes. The step greater than the default setting which is 1, mean that the step corresponds to the integer number of features to remove at each iteration. Thus, the speed of the process for RFECV will become faster.

Next, the CV is to determine the cross-validation splitting strategy. In here, the Stratified K-Folds cross-validator is used to provide train/test indices to split data in train/test sets. The number of splits in the function is defined as 2.

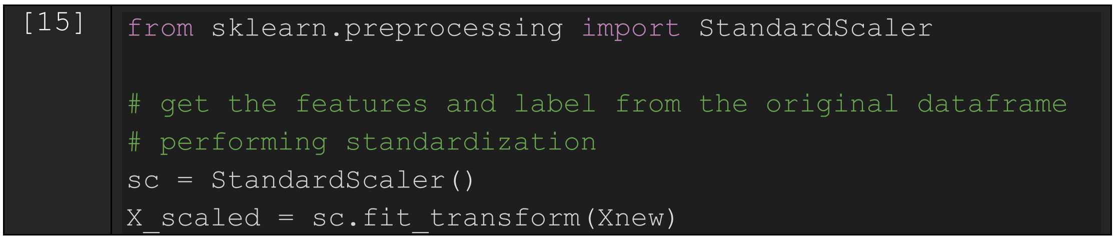
Lastly, defined the number of minimum features to select as 1.

After the function, fit the RFECV model and automatically tune the number of features. Then, transform the result, mean that reduce the X to the selected features and return the input sample with only the selected features into Xnew. Lastly, output the number of selected features. In here, there are 2676 genes is selected from the total of 54676 genes.

4.2.1.3 Perform the standardization on dataset.

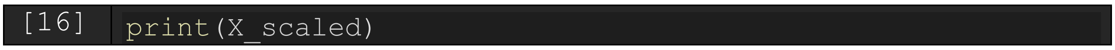
Standardization is a pre-processing method used to transform continuous data to make it look normally distributed. In another word, it is mean to rescaling the distribution of values so that the mean of the observed value is 0 and the standard deviation is 1.

For row 15, the module StandardScaler is import to perform standardization. In here the StandardScaler is defined as sc, the fit_transform is performed for the Xnew result that are get from the previous step, and the result is stored in X_scaled. fit_transform here mean that fits transformer to Xnew with optional parameters and returns a transformed version of Xnew.



```
[15]  from sklearn.preprocessing import StandardScaler

      # get the features and label from the original dataframe
      # performing standardization
      sc = StandardScaler()
      X_scaled = sc.fit_transform(Xnew)
```

For row 16, the output the result that return to X_scaled.

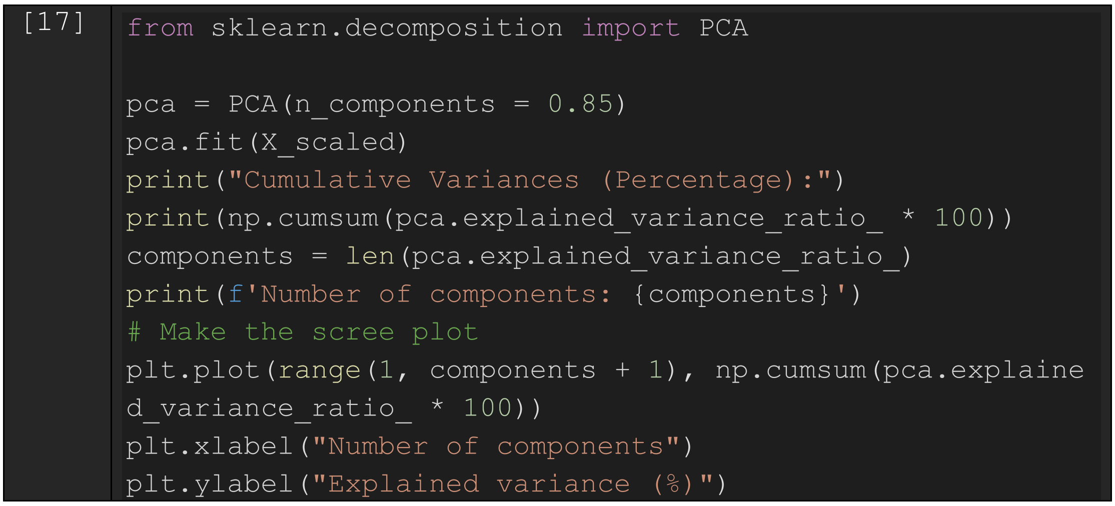


```
[16]  print(X_scaled)
```

4.2.1.4 Identify the top four most important features in each component with principal component analysis (PCA).

Principal component analysis (PCA) is a linear dimensionality reduction using Singular Value Decomposition (SVD) of the data to project it to a lower dimensional space. The input data is centered but not scaled for each feature before applying the SVD.

For row 17, the PCA module is import to perform the PCA. PCA can be used when the dimensions of the input features are high. For example, there are a lot of variables. In another word, PCA reduce the number of dimensions in a dataset while retaining the most information features.



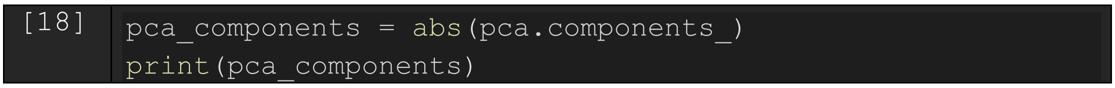
```
[17] from sklearn.decomposition import PCA

     pca = PCA(n_components = 0.85)
     pca.fit(X_scaled)
     print("Cumulative Variances (Percentage):")
     print(np.cumsum(pca.explained_variance_ratio_ * 100))
     components = len(pca.explained_variance_ratio_)
     print(f'Number of components: {components}')
     # Make the scree plot
     plt.plot(range(1, components + 1), np.cumsum(pca.explained_variance_ratio_ * 100))
     plt.xlabel("Number of components")
     plt.ylabel("Explained variance (%)")
```

Next, the PCA is defied and the n_component is set to 0.85, which mean that the number of components to keep is 85%. Then, fit the PCA model with the X-scaled that get from the previous step.

Then, print out the cumulative variances in percentage based on the percentage of variance explained by each of the selected component. Next, output the number of the selected component. In here, the selected component is 63.

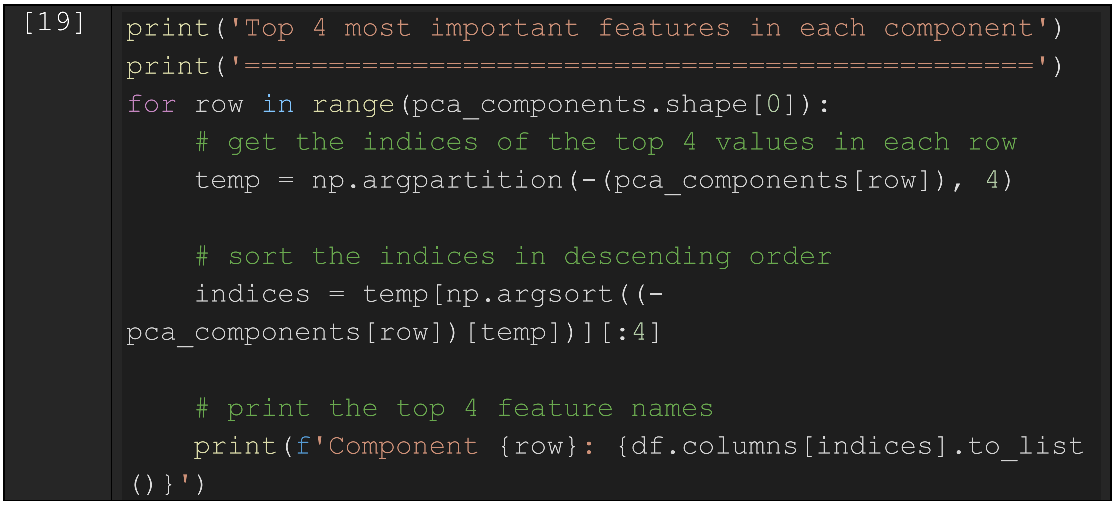
Lastly, plot the graphs based on the number of component and explained variance. In here, explained variance tells how much information (variance) can be attributed to each of the principal components. For the graphs, the variance is remained same after reach the number of components which is 63.

For row 18, output all the components that get from the previous code in detail. In here, there are total 63 component are listed in detail.



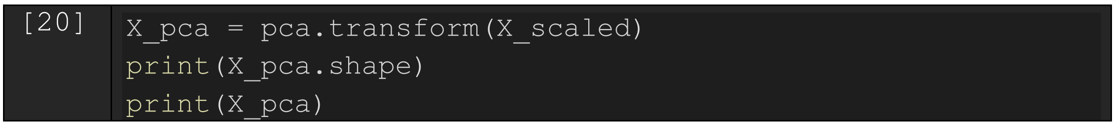
```
[18]  pca_components = abs(pca.components_)
      print(pca_components)
```

For row 19, select the top four most important features in each of the component for the total of 63 component that get from previous code.



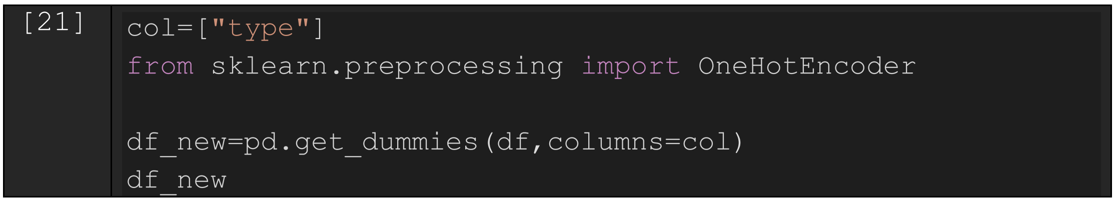
```
[19]  print('Top 4 most important features in each component')
      print('==============================================')
      for row in range(pca_components.shape[0]):
          # get the indices of the top 4 values in each row
          temp = np.argpartition(-(pca_components[row]), 4)

          # sort the indices in descending order
          indices = temp[np.argsort((-
      pca_components[row])[temp])][:4]

          # print the top 4 feature names
          print(f'Component {row}: {df.columns[indices].to_list
      ()}')
```

For row 20, the X_scaled is transform, mean that apply the dimensionality reduction to X_scaled, and return the projection of X_scaled in the first principal components, where 151 is the number of samples and 63 is the number of the components.



```
[20]  X_pca = pca.transform(X_scaled)
      print(X_pca.shape)
      print(X_pca)
```

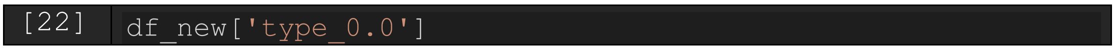
For row 21, the OneHotEncoder is import. This is to encode the categorical features as a one-hot numeric array. In here, the type of breast cancer is converted from categorical variable into dummy/indicator variables. A dummy variable is a numeric variable that encodes categorical information. Usually, dummy variables have two possible values which is 0 or 1. A 1 encode the presence of a category, and a 0 encodes the absence of a category. There are 6 class in the dataset, and the category is divided into type_0.0, type_1.0, type_2.0, type_3.0, type_4.0, type_5.0 and type_6.0. Lastly, the whole dataset with type label is store in df_new variable.



[21]

```
col=["type"]
from sklearn.preprocessing import OneHotEncoder

df_new=pd.get_dummies(df,columns=col)
df_new
```

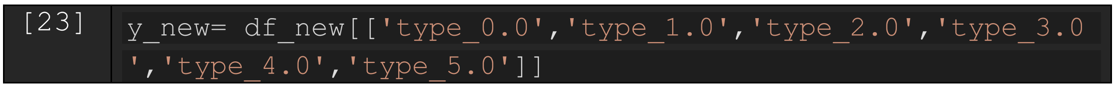
For row 22, output the detail info for class 0, which is type_0.0 in the dataset. The present of the category will label with 1, while the absence of the category is label with 0.



[22]

```
df_new['type_0.0']
```

For row 23, all the type of breast cancer with new category label is store into y_new.



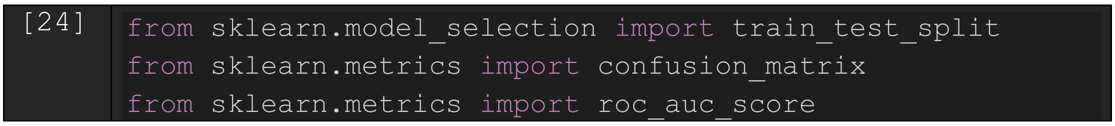
[23]

```
y_new= df_new[['type_0.0','type_1.0','type_2.0','type_3.0','type_4.0','type_5.0']]
```

4.2.1.5 Training and testing data. (60% Training, 40% Testing) with XGBoost Classifier.

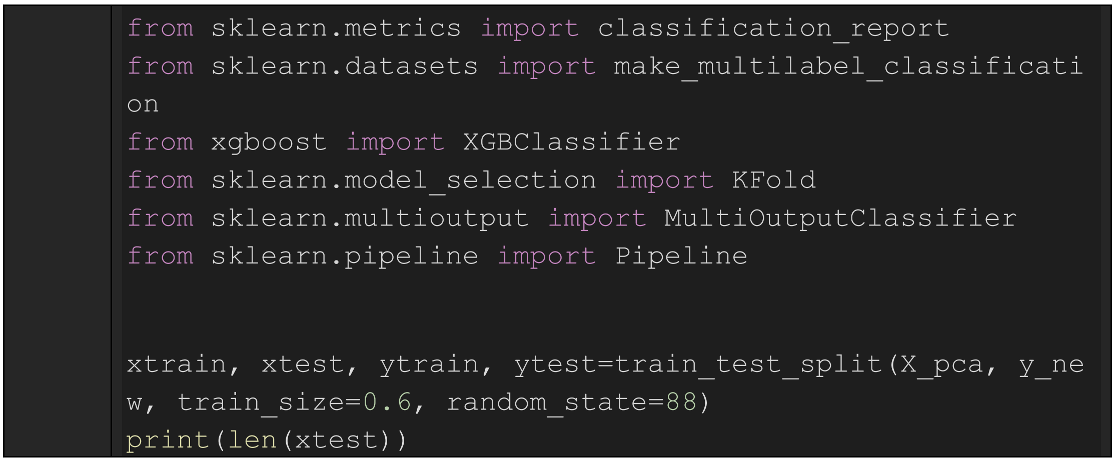
In this step, the XGB classifier is used for the training and testing for the dataset, and also to perform gene selection for cancer classification.

For row 24, there are several modules import for the cancer classification with gene selection, which is train_test_split, confusion_matrix, roc_auc_score, classification_report, make_multilable_classification, XGBClassifier, KFold, MultiOutputClassifier and Pipeline. This is the module that need for training, testing the dataset, defiend the classfier, calculate the accuracy, and output the classification report.



[24]

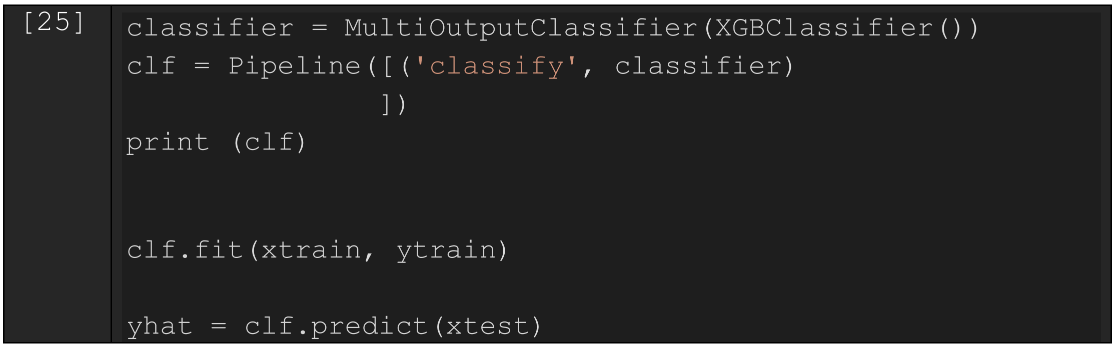
```
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
from sklearn.metrics import roc_auc_score
```



```
from sklearn.metrics import classification_report
from sklearn.datasets import make_multilabel_classificati
on
from xgboost import XGBClassifier
from sklearn.model_selection import KFold
from sklearn.multioutput import MultiOutputClassifier
from sklearn.pipeline import Pipeline


xtrain, xtest, ytrain, ytest=train_test_split(X_pca, y_ne
w, train_size=0.6, random_state=88)
print(len(xtest))
```

After import the module, the dataset is divided into training and testing set. Training size here is 0.6 (60%), while the size for testing is 0.4 (40%). The arrays here using the X_pca that output from the previous code, and y_new that store the new category label. While the random_state is set to 88, which is for controls the shuffling applied to the data before applying the split. Lastly, output the number size that are gong to testing, which is 61.

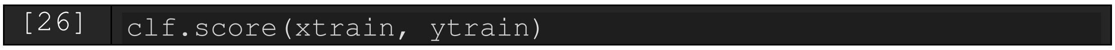
For row 25, the XGBoost classifier is defined for classifier. Then, the pipeline used as the estimator that avoid leaking the test set into the train set. After this, output the detail of the clf where the pipeline is defined.



```
[25]  classifier = MultiOutputClassifier(XGBClassifier())
      clf = Pipeline([('classify', classifier)
                      ])
      print (clf)


      clf.fit(xtrain, ytrain)

      yhat = clf.predict(xtest)
```

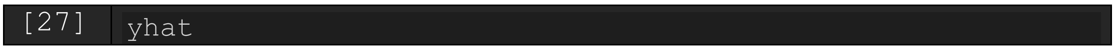
Next, fit the model with xtrain and ytrain, which is the training sets. xtrain and y train here are the data and target variable that used to train the model on. After this, predict the target variable for xtest, and the result is store in yhat.

For row 26, return the coefficient of determination of the prediction. The best possible score is 1.0 and it can be negative because of the model can be arbitrarily worse. In here, the score is 1.0 where indicate the model is best.



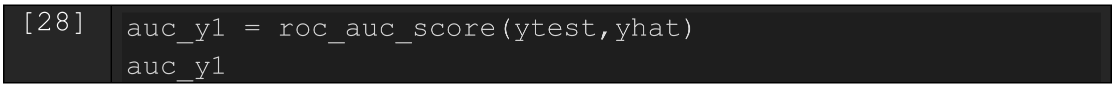
```
[26]  clf.score(xtrain, ytrain)
```

For row 27, print the predict result in numpy array.



```
[27]  yhat
```

For row 28, evaluate the performance by comparing the ytest and ypred, then output the accuracy score. The accuracy score is bounded between 0 (worst possible ranking) and 1 (best possible ranking), with 0.5 indicating random ranking. For the result here, the accuracy score is 0.9289125922321123 which is identified as accuracy. Originally, the accuracy score before adding the recursive feature elimination with cross-validation (RFECV) is 0.8755930492849098. After adding the RFECV, there are slightly improve in the accuracy about gene selection for cancer classification.

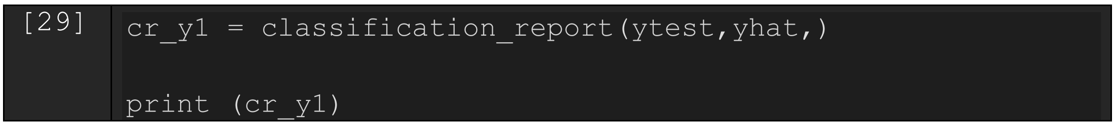


```
[28]  auc_y1 = roc_auc_score(ytest,yhat)
      auc_y1
```

4.2.1.6 Produce classification report, and plot the graphs.

In this section, the classification report and graph about the gene selection for cancer classification based on XGBoost classifier is produced. The detail explanation will be in Chapter 5 which is Testing and Result Discussion.

For row 29, the classification report about the gene selection for cancer classification based on XGBoost classifier is produced. The classification report is a build text report that showing the main classification metrics.



```
[29]  cr_y1 = classification_report(ytest,yhat,)

      print (cr_y1)
```

For the other row, related graphs are produce based on the classification result.

Lastly, the detail explanation of the result will be in Chapter 5.

4.3 Summary

In summary, chapter 4 is focus on design and implementation of the algorithm and coding. In this chapter, the implementation of the XGBoost classifier is explained in detail. After implement the XGBoost classifier with adding the recursive feature elimination with cross-validation (RFECV), there are some slightly improve in the accuracy compare to the previous work which is without the implementation of RFECV. Lastly, the chapter 5 which is Testing and Result Discussion will explain deeply about the result that produce from the code.

CHAPTER 5

TESTING AND RESULT DISSCUSSION

5.1 Introduction

Chapter 5 will discuss about the result and finding of the research, which is Gene Selection for Cancer Classification with XGBoost Classifier. This chapter contain the result of the finding based on the experiment and testing that has been done. Other than that, it also includes the explanation for each of the result that produce after running the classifier.

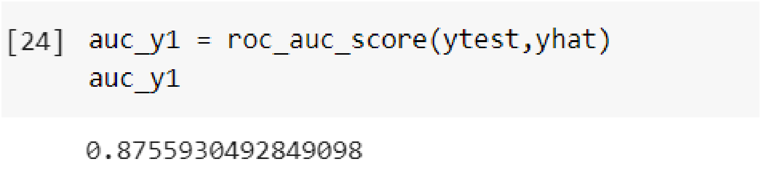
5.2 Testing and Result Discussion

In this section will going through discussion on testing and result, which cover the tested data, measurement method used in the research, and also the result related to the research. The result will be compare based on the previous work, and the proposed method, which the new code has adding contribution with recursive feature elimination with cross validation (RFECV).

5.2.1 Previous Work with XGBoost Classifier

After the python code is run based on the Gene Selection for Cancer Classification with XGBoost Classifier, there are an accuracy score will be generate based on the classifier.

Below is the image that show the accuracy score based on the XGBoost classifier.

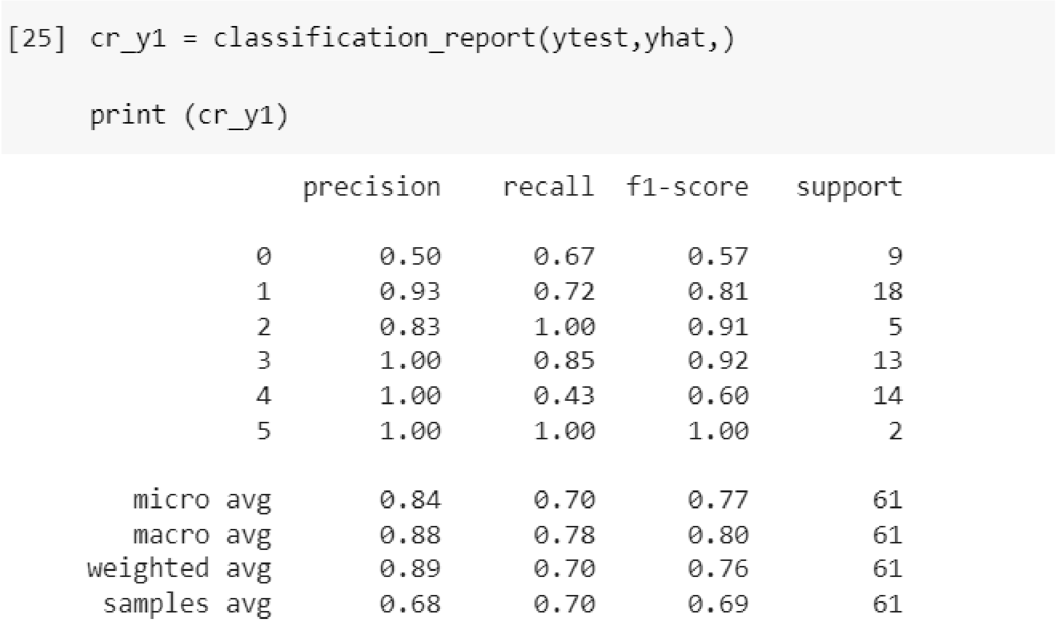


```
[24] auc_y1 = roc_auc_score(ytest,yhat)
     auc_y1

     0.8755930492849098
```

Figure 5.1 Accuracy Score for the Previous Work.

The accuracy score indicates that the accuracy for the classifier, which evaluate the performance by comparing y_test and y_pred. The score is bounded between 0 (worst possible ranking) and 1 (best possible ranking), with 0.5 indicating random ranking. For the accuracy score that get after running the code, the accuracy of the XGBoost classifier is considered as high accuracy.

After this, there are a classification report produce based on the classifier. Below is the image that show the classification report based on the XGboost classifier. 0 to 5 indicate the total of 6 classes that are contain in the dataset.



```
[25] cr_y1 = classification_report(ytest,yhat,)

     print (cr_y1)

                  precision    recall  f1-score   support

               0       0.50      0.67      0.57         9
               1       0.93      0.72      0.81        18
               2       0.83      1.00      0.91         5
               3       1.00      0.85      0.92        13
               4       1.00      0.43      0.60        14
               5       1.00      1.00      1.00         2

       micro avg       0.84      0.70      0.77        61
       macro avg       0.88      0.78      0.80        61
    weighted avg       0.89      0.70      0.76        61
     samples avg       0.68      0.70      0.69        61
```

Figure 5.2 Classification Report for the Previous Work.

In machine learning, a classification report is a performance evaluation metric. It is used to display the trained classification model's precision, recall, F1 Score, and support. Basically, it gives a clearer picture of the trained model's overall performance.

From the above image, precision is the capacity of a classifier to avoid classifying an instance positive that is in fact negative. In another word, it is indicating the accuracy of positive prediction. When looking into the macro average, the average score for the precision is consider as high.

The capacity of a classifier to locate every positive instance is known as recall. It is described as the proportion of true positives to the total of true positives and false negatives for each class. In another word, it is described as fraction of positives that were correctly identified.

The weighted harmonic mean of recall and precision is known as the F1-score. The projected performance of the model is higher when the closer the F1 score value is near 1.0. In other word, the best score is 1.0 and the worst is 0.0. When looking into the macro average, the average score for the F1 score is consider as high.

Last is the support, it is the number of instances of the class that actually occur in the dataset. It only diagnoses the performance evaluation procedure and does not differ between models. For example, the support value of 9 in class 0 mean that there are only 9 observations with actual occurrences of class 0.

After the classification report, several graphs are produce based on the result which is Principal Component Analysis (PCA). Basically, Principal Component Analysis (PCA) is performed in order to decrease the number of dimensions or features in a dataset. In this research, there are total 90 components in the previous work. For the result discussion, only top 3 of the PCA components is discuss.

Below is the graph that produce from previous work, where it is the comparison of the PCA between PCA 1 and PCA 2.

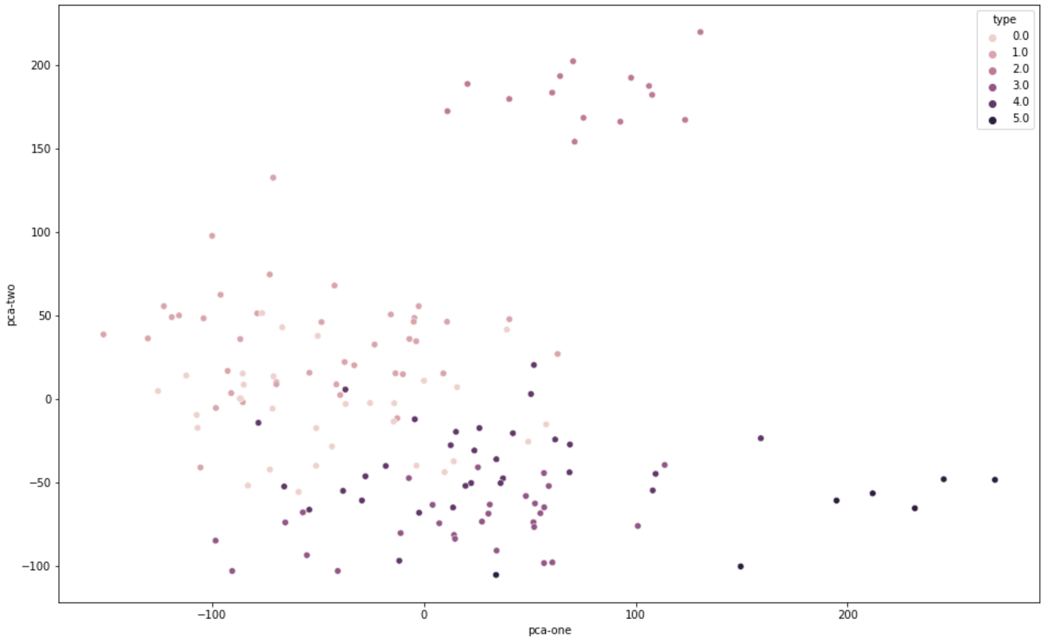


Figure 5.3 2D PCA Grpah from Previous Work, Compare between PCA 1 and PCA 2.

While the next image is the comparison of the PCA between PCA 2 and PCA 3.

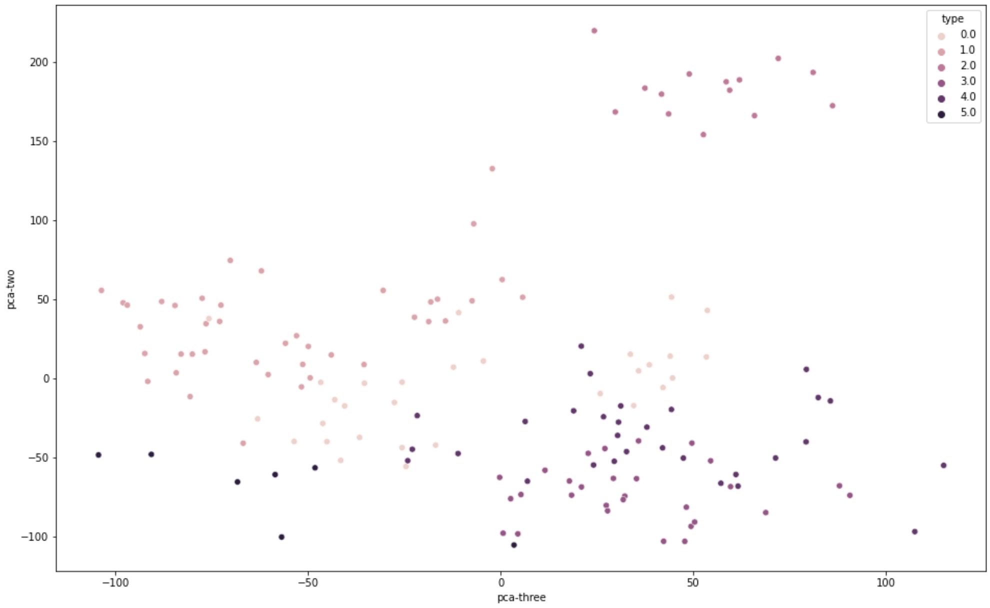


Figure 5.4 2D PCA Grpah from Previous Work, Compare between PCA 2 and PCA 3.

Lastly is the image of the comparison of the PCA between PCA 1and PCA 3.

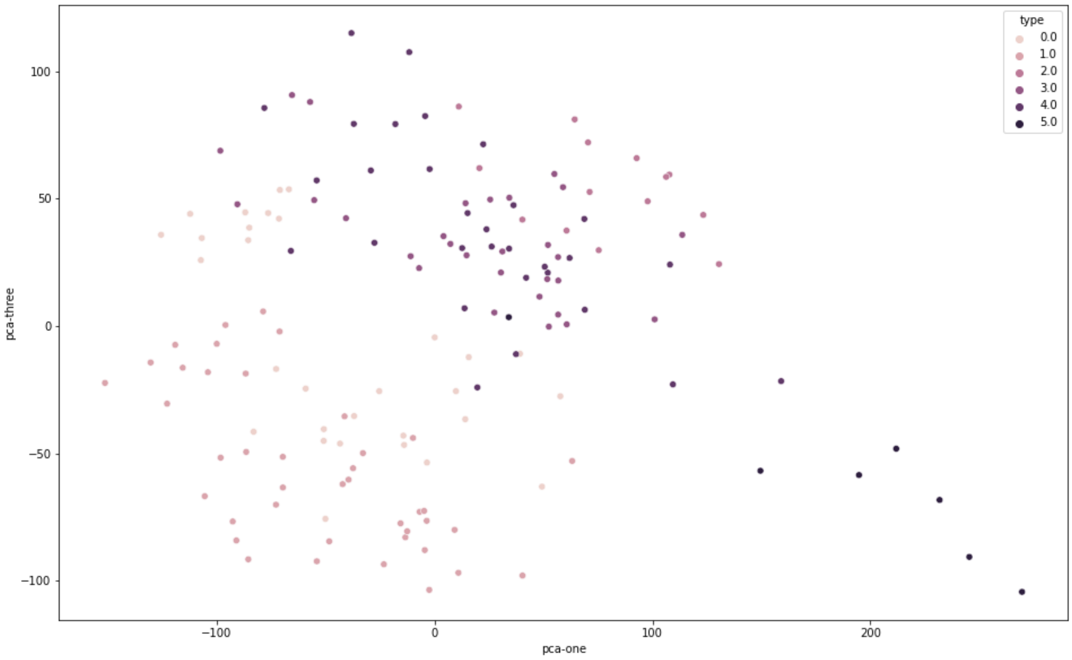


Figure 5.5 2D PCA Grpah from Previous Work, Compare between PCA 1 and PCA 3.

Form the above image that are produce based on the Principal Component Analysis (PCA), A PCA plot shows clusters of samples based on their similarity. PCA does not eliminate any samples or traits (variables). Instead, it creates principal components to decrease the overwhelming number of dimensions (PCs). PCA plot make it much easier to compare the similar genes. The genes will group together if their expression profiles are comparable.

After observe the three of the PCA graph from above, the some of the gene expressions are not in a cluster, which mean some of the gene expression is not group together. This mean that the gene expression that is not cluster is defined as outlier, which mean that it is the gene expression that needed to be focus on and observe.

Next, the code produces the 3D PCA graph which compare between PCA 1, PCA 2, and PCA 3. A PCA plot is often a 2D scatter plot in which the data is presented with the two principal components that are best descriptive of the data. Instead, select to plot using three PCs, which will result in a 3D scatter plot, also known as a 3D PCA.

Below is the 3D PCA that produce from the previous work.

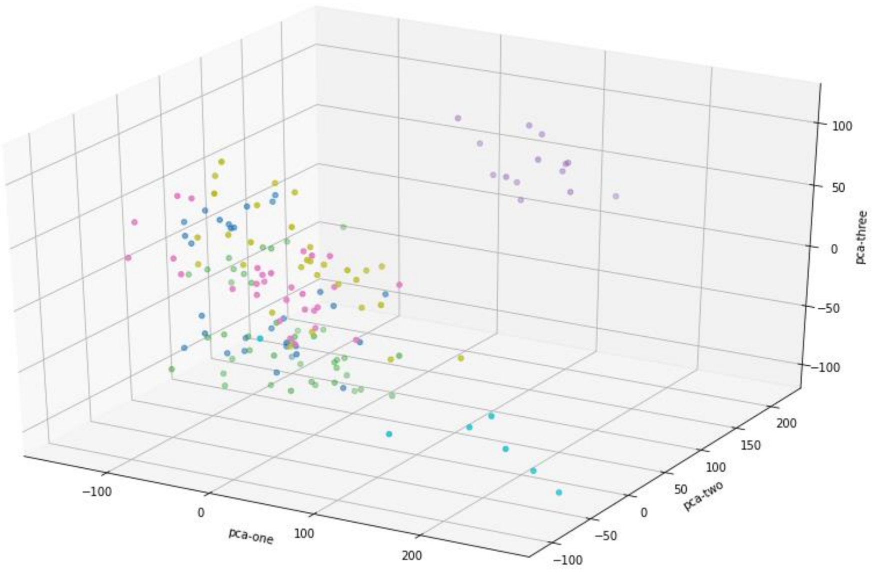


Figure 5.6 3D PCA Grpah from Previous Work, Compare between PCA 1, PCA 2 and PCA 3.

The number of principle components used for visualization is the main distinction between 2D PCA and 3D PCA. Principle component analysis (PCA) builds its principal components to capture the dataset's greatest variance: PC1 depicts the greatest variation, PC2 reflects the second-greatest variation, and so on. The top two or three PCs can thus catch the majority of the variation, while the remaining ones can be eliminated without losing much data.

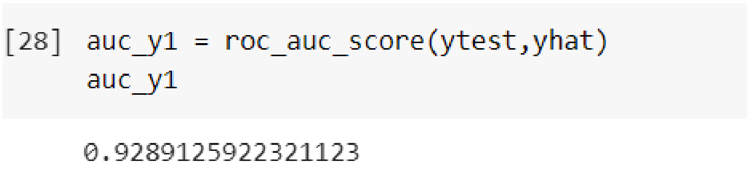
Although PCA is not a clustering technique, it can aid in the visualization of patterns by reducing dimensionality, such as clusters of expression profiles with comparable characteristics. These patterns might be difficult to see on a 2D PCA plot but are more obvious in 3D.

After the observation for the 3D PCA, it is clearly that the gene expression is not in cluster, and not easy to identifiable based on cluster or classes. This mean that the 3D PCA show more messy gene that are not group together, and it is harder to identify the outlier that need to observe.

5.2.2 Proposed Method with XGBoost Classifier and Recursive Feature Elimination with Cross-Validation

Same as the previous, most of the result and graph are produce, but the performance after adding the recursive elimination with cross-validation, with XGBoost classifier are significantly improve compare with the previous work. The detail explanation is elaborate at below.

After the python code is run, the accuracy score is generated. Below is the image that showing the accuracy score.

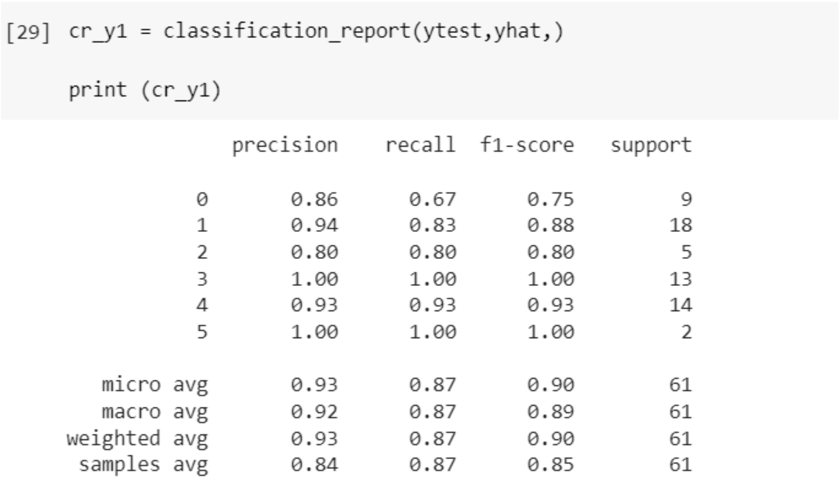


```
[28] auc_y1 = roc_auc_score(ytest,yhat)
     auc_y1

     0.9289125922321123
```

Figure 5.7 Accuracy Score for the Proposed Method.

The score is bound between 0 and 1, and getting the result as above image is considered the accuracy of the model is high. When compare with the previous work, the proposed method with contribution has a higher accuracy score.

After this, there are a classification report produce based on the classifier. Below is the image that show the classification report based on the classifier.



```
[29] cr_y1 = classification_report(ytest,yhat,)

     print (cr_y1)

                  precision    recall  f1-score   support

               0       0.86      0.67      0.75         9
               1       0.94      0.83      0.88        18
               2       0.80      0.80      0.80         5
               3       1.00      1.00      1.00        13
               4       0.93      0.93      0.93        14
               5       1.00      1.00      1.00         2

       micro avg       0.93      0.87      0.90        61
       macro avg       0.92      0.87      0.89        61
    weighted avg       0.93      0.87      0.90        61
     samples avg       0.84      0.87      0.85        61
```

Figure 5.8 Classification Report for Proposed Method.

When compare the macro average F1 score with the previous work, it is clearly show that the proposed method produces more higher F1 score. This mean that the projected performance of the model is higher, since the F1 score value is more near 1.0.

After the classification report, several PCA graph is produce. In the proposed method, there are 63 components for the PCA.

Below is the graph that produce based on the comparison of the PCA between PCA 1 and PCA 2.

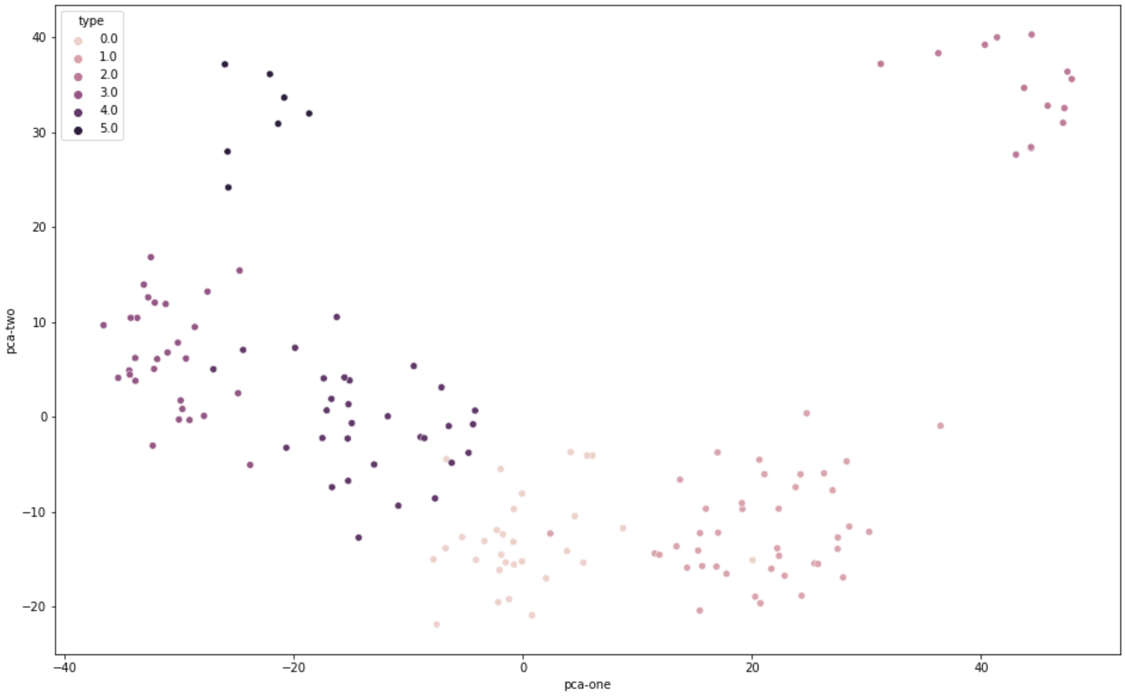


Figure 5.9 2D PCA Grpah from Proposed Method, Compare between PCA 1 and PCA 2.

While the next image is the comparison of the PCA between PCA 2 and PCA 3.

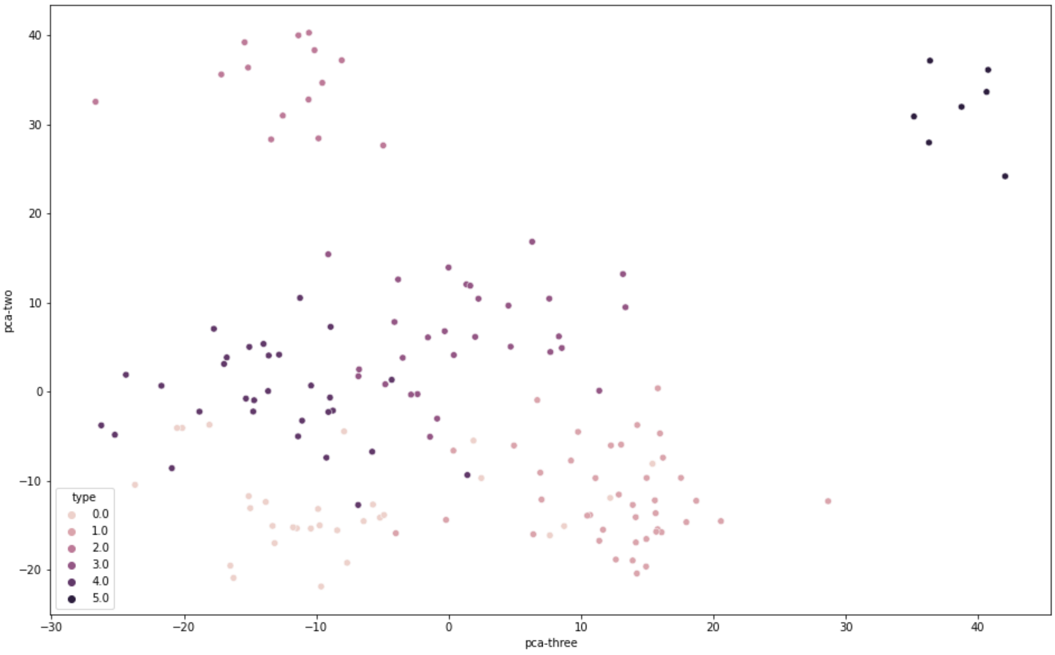


Figure 5.10 2D PCA Grpah from Proposed Method, Compare between PCA 2 and PCA 3.

While the last image is the comparison of the PCA between PCA 1 and PCA 3.

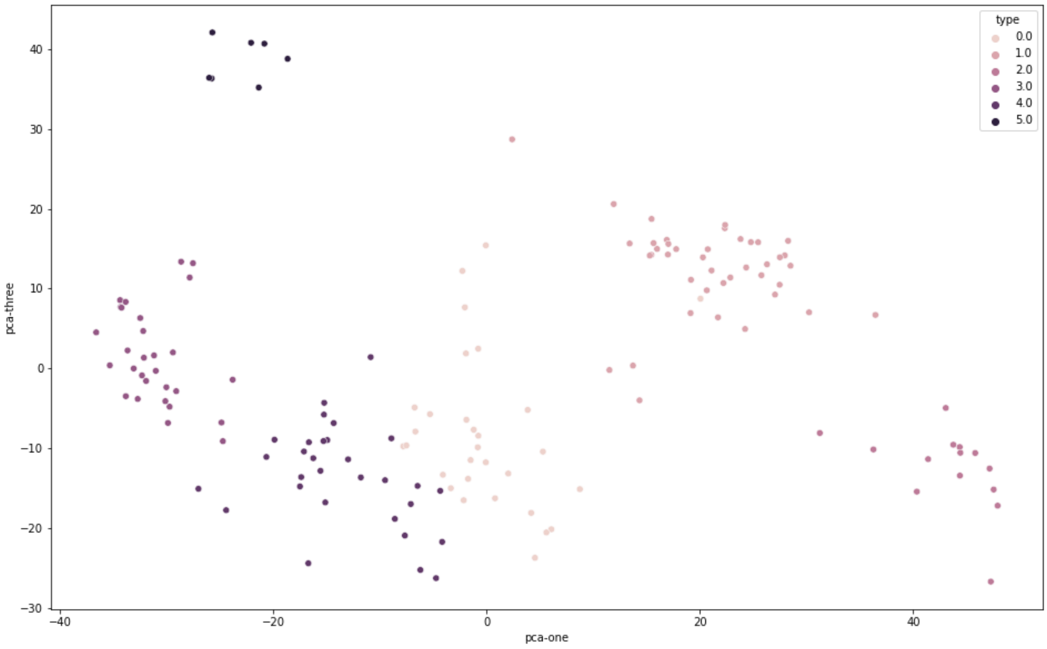


Figure 5.11 2D PCA Grpah from Proposed Method, Compare between PCA 1 and PCA 3.

After observe the PCA plot for the proposed method, the cluster for each of the classes indicate the higher relationship based on the gene expression profile. Examining the distance is more effective to identifying the outliers that looking at one variable at a time. It is clearly seen that the graph that produce based on new code have more cluster gene compare with the original code. The gene based on classes that have similar expression profiles are now clustered together. While the genes that is not cluster together is defined as outlier.

For example, the plot that compare based on PCA 1 and PCA 2, it is clearly seen that the cluster from the plot produce by the proposed method is more identifiable. Which mean that the gene based on the classes from the proposed method is group together, and there is less gene that are wrongly group to the other classes.

Next is the 3D PCA that produce by the proposed method. Below is the 3D PCA that produce.

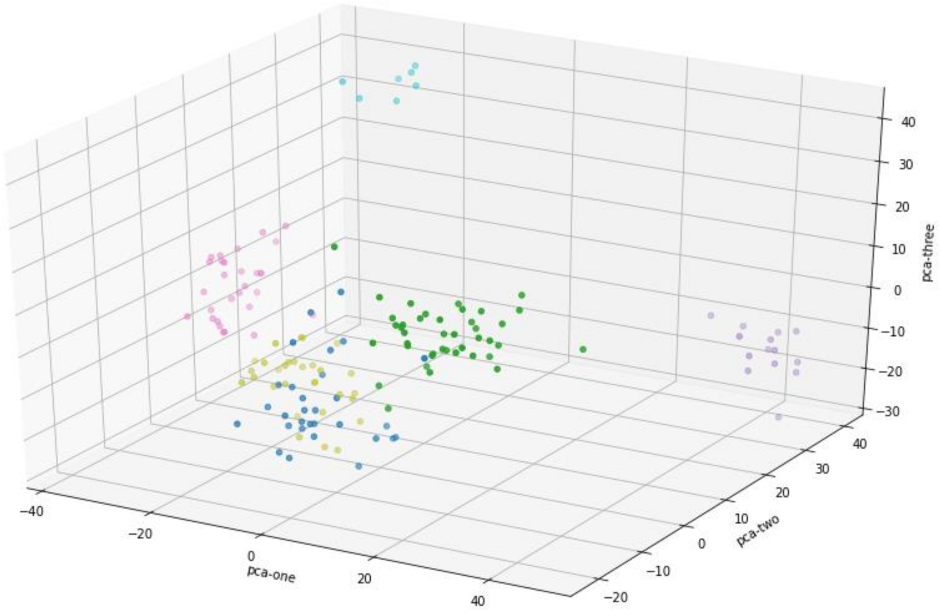


Figure 5.12 3D PCA Grpah from Proposed Method, Compare between PCA 1, PCA 2 and PCA 3.

Afte the observation, it is clearly show that the 3D PCA from proposed method showing more identifiable genes based on the cluster of the classes. Based on the plot, it can be identified that the proposed method with have higher accuracy compare with previous work.

5.2.3 Discussion After Observe Both Result

In this section, there are a discussion after observer the both of the result that produce.

First of all, the proposed method, where adding the recursive feature elimination with cross validation (RFECV), with the XGBoost classifier produce higher accuracy to the gene selection for cancer classification. This can be easily observed from the accuracy score that produce by two difference code.

When coming to observe the graph, it is clearly showing the plot from the proposed method are more cluster, which mean that most of the genes are group together. Outlier are easily to identify and observe compare with the previous work.

Thus, the adding of recursive feature elimination with cross validation (RFECV) is giving improvement for the XGBoost classifier in making the gene selection for cancer classification. The reason why the recursive feature elimination with cross validation (RFECV) is chosen the explain below.

The aim of the RFECV is to recursively delete traits that are of lesser relevance than others. Compare to the recursive feature elimination (RFE), cross-validation is added to the mix. Which mean that the RFECV only using the validation data to determine the score for feature importance. Although this process may require more resources depending on the quantity of data and the estimator utilised, but it provides more significant improvement for the XGBoost classifier when compare to the recursive feature elimination (RFE).

In conclusion, recursive feature elimination with cross validation (RFECV) are chosen in this research to maximize the performance for the XGBoost classifier in gene selection for cancer classification.

5.3 Summary

In conclusion, chapter 5 is more focus in explanation of the result and finding after going through the code. Most of the result always indicating that the proposed method with contribution have higher accuracy compare with the previous work. Lastly, it is indicating that the recursive feature elimination with cross-validation (RFECV) have improve the accuracy of the XGBoost classifier in gene selection for cancer classification.

CHAPTER 6

CONCLUSION

6.1 Introduction

In this chapter, it will focus in conclude the research that have been done, which include the conclusion of the research, data retrieve and observe, methodology implementation conclusion, and future suggestion and enhancement of the research.

In this research, Gene Selection for Cancer Classification with XGBoost Classifier have proposed. Microarray technology enables the creation of databases of cancerous tissues based on gene expression data. When compared to the number of genes involved, training datasets for cancer classification typically have a small sample size and consist of multiclass categories. For this research, breast cancer dataset is used. To select the most important genes, recursive feature elimination with cross-validation (RFECV) is used to select the features, and lastly the XGBoost classifier is used for cancer classification. For the result observation, the combination of RFECV with XGBoost given a higher accuracy for gene selection and cancer classification compare with the original method which only using the XGBoost.

6.2 Research Constraint

For the research constraint, there are two things that need to clarify, which is the limited time. Detail explanation is show below.

1. Limited Time

- Some of the algorithm take time to process, and some of it is too complex to process within a short time. Thus, some of the algorithm is not able to include in the research as an approach to increase the accuracy of the cancer classification.

6.3 Future Work

Even though the proposed methodology has given an effectiveness result, but it can be further increase by improved the features elimination method or the search approach. For example, include of the feature selection method search as Ant Colony Optimization. To conclude, the main aim is to improve the accuracy of the gene selection for cancer classification and reduce the unrelated genes before going for the next process.

REFERENCES

World Health Organization. (2022, February 3). *Cancer*. https://www.who.int/news-room/fact-sheets/detail/cancer

Mahendran, N., Durai Raj Vincent, P. M., Srinivasan, K., & Chang, C. Y. (2020). Machine Learning Based Computational Gene Selection Models: A Survey, Performance Evaluation, Open Issues, and Future Research Directions. *Frontiers in Genetics*, *11*.

Liu, JX., Wang, YT., Zheng, CH. et al. Robust PCA based method for discovering differentially expressed genes. BMC Bioinformatics 14, S3 (2013).

Cancer Classification | SEER Training. (n.d.). National Cancer Institute. https://training.seer.cancer.gov/disease/categories/classification.html#:%7E:text=Carcinomas%20are%20divided%20into%20two,thickened%20plaque%2Dlike%20white%20mucosa.

Alanni, R., Hou, J., Azzawi, H. et al. Deep gene selection method to select genes from microarray datasets for cancer classification. BMC Bioinformatics 20, 608 (2019).

Nall, R. M. (2018, November 16). *How does a doctor diagnose cancer?* Medical News Today. https://www.medicalnewstoday.com/articles/323708

Karaboga, D. (2010, March 30). *Artificial bee colony algorithm - Scholarpedia*. Scholarpedia. http://www.scholarpedia.org/article/Artificial_bee_colony_algorithm

A. Singh and D. Kumar, "Novel ABC based training algorithm for ovarian cancer detection using neural network," 2017 International Conference on Trends in Electronics and Informatics (ICEI), 2017, pp. 594-597

Song, Q., Merajver, S. D., & Li, J. Z. (2015). Cancer classification in the genomic era: five contemporary problems. *Human Genomics*, *9*(1).

D. Pavithra and B. Lakshmanan, "Feature selection and classification in gene expression cancer data," 2017 International Conference on Computational Intelligence in Data Science (ICCIDS), 2017, pp. 1-6.

H. L. Shashirekha and A. H. Wani, "Gene selection by Mutual Nearest Neighbor approach," 2015 International Conference on Emerging Research in Electronics, Computer Science and Technology (ICERECT), 2015, pp. 398-402.

M. Amrane, S. Oukid, I. Gagaoua and T. Ensarİ, "Breast cancer classification using machine learning," 2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT), 2018, pp. 1-4.

W. Luo, L. Wang and J. Sun, "Feature Selection for Cancer Classification Based on Support Vector Machine," 2009 WRI Global Congress on Intelligent Systems, 2009, pp. 422-426.

L. Wang, F. Chu and W. Xie, "Accurate Cancer Classification Using Expressions of Very Few Genes," in IEEE/ACM Transactions on Computational Biology and Bioinformatics, vol. 4, no. 1, pp. 40-53, Jan.-March 2007.

H. Wang, H. Yu, Q. Zhang, S. Cang, W. Liao and F. Zhu, "Parameters optimization of classifier and feature selection based on improved artificial bee colony algorithm," 2016 International Conference on Advanced Mechatronic Systems (ICAMechS), 2016, pp. 242-247.

J. Ge, X. Zhang, G. Liu and Y. Sun, "A Novel Feature Selection Algorithm Based on Artificial Bee Colony Algorithm and Genetic Algorithm," 2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS), 2019, pp. 131-135.

Moosa, J.M., Shakur, R., Kaykobad, M. et al. Gene selection for cancer classification with the help of bees. BMC Med Genomics 9, 47 (2016).

Grisci, B. (2020). Breast cancer gene expression - CuMiDa [Data File]. Kaggle. Retrieved from https://www.kaggle.com/datasets/brunogrisci/breast-cancer-gene-expression-cumida

What is XGBoost? (n.d.). NVIDIA Data Science Glossary. https://www.nvidia.com/en-us/glossary/data-science/xgboost/

Brownlee, J. (2016, August 17). A Gentle Introduction to XGBoost for Applied Machine Learning. Retrieved December 27, 2022, from https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/

Chen, S. (2021, February 3). A Novel XGBoost Method to Infer the Primary Lesion of 20 Solid Tumor Types From Gene Expression Data. Frontiers. https://www.frontiersin.org/articles/10.3389/fgene.2021.632761/full

Grisci, B. (2020). Breast cancer gene expression - CuMiDa [Data File]. Kaggle. Retrieved from https://www.kaggle.com/datasets/brunogrisci/breast-cancer-gene-expression-cumida