MACHINE LEARNING CLASSIFICATION OF PRIMARY TISSUE ORIGIN OF CANCER FROM DNA METHYLATION MARKERS

By

Sravani Gannavarapu Surya Naga

Bachelor of Technology, Computer Science
Jawaharlal Nehru Technological University, Hyderabad
2014

A thesis submitted in partial fulfillment
of the requirements for the

Master of Science in Computer Science

Department of Computer Science
Howard R. Hughes College of Engineering
The Graduate College

University of Nevada, Las Vegas
May 2019

UNLV | GRADUATE COLLEGE

Thesis Approval

The Graduate College
The University of Nevada, Las Vegas

April 16, 2019

This thesis prepared by

Sravani Gannavarapu Surya Naga

entitled

Machine Learning Classification of Primary Tissue Origin of Cancer from DNA Methylation Markers

is approved in partial fulfillment of the requirements for the degree of

Master of Science in Computer Science
Department of Computer Science

Fatma Nasoz, Ph.D.
Examination Committee Chair

Kathryn Hausbeck Korgan, Ph.D.
Graduate College Dean

Ajoy Datta, Ph.D.
Examination Committee Member

Kazem Taghva, Ph.D.
Examination Committee Member

Mira Han, Ph.D.
Graduate College Faculty Representative

Abstract

Cancer is one of the leading causes of death globally and was responsible for approximately 9.6 million deaths in 2018. One of the main reason for deaths from cancer is late-stage presentation and inaccessible diagnosis and treatment. Cancer often spreads from the part of the body where it started (primary site) to a different part of the body (metastatic site). Identifying the primary site of cancer plays a key role as it directs the appropriate treatment. Cancer which spreads needs the same treatment as its origin. Having this knowledge can help doctors to decide the type of treatment.

All cancers begin when one or more genes in a cell mutate and create abnormal proteins which cause cells to multiply uncontrollably. Genes are present in the DNA of each cell in human body, and research shows that distinct and abnormal patterns in methylation of DNA are observed in case of cancers. DNA methylation is also considered as an early and fundamental step where normal tissue undergoes transformations. Since DNA methylation is tissue-specific and change with cell differentiation, methylation sites are good markers for identifying tissues of origin.

In this thesis, we propose the use of machine learning techniques to identify the primary sites of cancers to increase the accuracy of diagnosis and treatment.For this purpose, we implemented various classification algorithms in machine learning like support vector machines, random forests classifier, decision trees, and K nearest neighbor classifier to classify the tumor samples into their tissue origin and compared these models using traditional machine learning metrics. The models are trained and tested on features extracted from the DNA methylation datasets maintained by The Cancer Genome Atlas (TCGA). The experimental results showed that support vector machines could predict the primary sites with 95% training accuracy. The model gave 86% accuracy when tested on a completely independent dataset collected from Gene Expression Omnibus (GEO).

Acknowledgements

I would like to express my special thanks and gratitude to my advisor Dr. Fatma Nasoz, for all the guidance and motivation throughout my research. She is a great mentor who always showed me a right direction and also a person whom I look upto.

I would like to extend my thanks to Dr. Mira Han, for the continuous guidance and support throughout my thesis. You had been the backbone and guided me all the way from understanding domain, to gathering data to producing results. I would also like to thank Travis Mize, Richard Van, Nikolay Stoynov, Xiaogang Wu and Nevada Institute of Personal Medicine Department for all the support provided whenever needed.

I would like to express my gratitude to Dr. Kazem Taghva and Dr. Ajoy Datta for their guidance and support throughout my Master's program at UNLV and also for being part of my thesis committee.

My deep sense of gratitude to my family Rajeshwar Kothuri, Satya Prakash Gannavarapu, Lakshmi Gannavarapu, Ram Rohit Gannavarapu, Lakshmi Kothuri and Sarma Kothuri for being my strength and motivation. I would also like to thank Lohitha Chintham Reddy, Anusha Challa, Aditya Rajuladevi and Sai Phani Krishna Parsa for their guidance and support throughout my Master's program. Finally, I would like to thank all my friends, seniors and juniors who made my time here at UNLV very memorable.

Sravani Gannavarapu Surya Naga

University of Nevada, Las Vegas

May 2019

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Cancer is among the leading causes of death worldwide. Every year, new cases of cancer are reported[PdMV+16]. Figure 1.1 shows the statistics published by the International Agency for Cancer Research (IARC) of number of new cases in 2018 and number of deaths in 2018. IARC is an intergovernmental agency which is part of the World Health Organization of the United Nations. Early detection of cancer is extremely important as it greatly increases the chances of successful treatment. Identifying the exact tissue of origin also plays a vital role in successful treatment. Many researchers in this field are focusing on finding ways for early identification of the tissue origin of cancer using various ways of gene expression and various epigenetic markers.

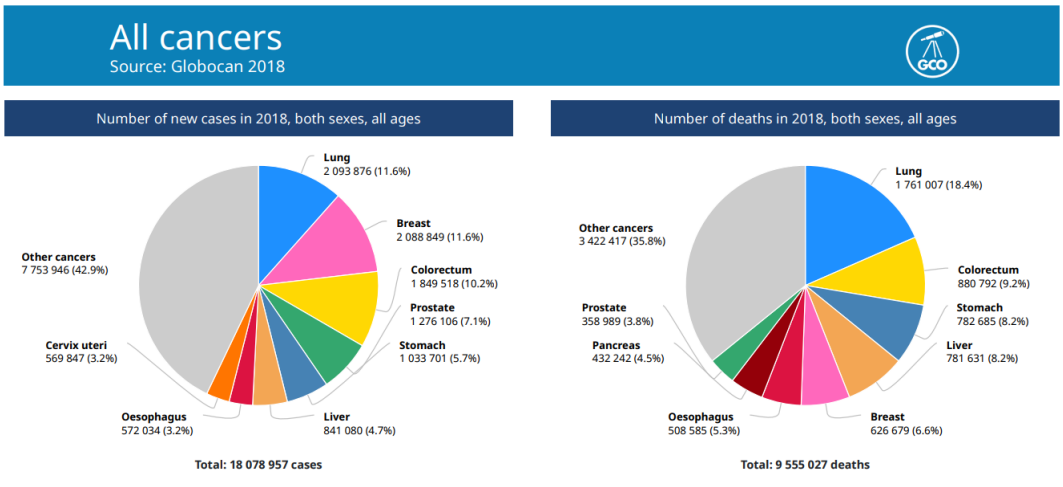


Figure 1.1: Cancer statistics

Cancer is the name given to a collection of related diseases. In every type of cancer, cells in a part of the body divide uncontrollably and spread to surrounding tissues. It is natural for cells in human body to grow, divide and form new cells as required by our body. Cells die when they grow old or damaged and new cells replace them. In case of cancer, cells become abnormal where old and damaged cells survive and new cells are created even when they are not required. These extra cells divide uncontrollably and form tumors [Ins15].

Tumors can be benign or malignant. Unlike benign tumors, malignant tumors invade nearby tissues and spread to different parts of the body. Sometimes, cancer cells break off and travel through blood and lymph system to distant places in the body and form new tumors. The place where cancer initially started is called the primary site and the place where it spreads is called metastatic site. Metastatic cancer cells look similar to original cancer cells under a microscope. Moreover, they have some common features like specific chromosome changes. Name of cancer is determined based on its primary site. For example, brain cancer which spread to the lungs is still classified as brain cancer. In some cases, determining the primary site of cancer is difficult. When cancer is found in one or more metastatic sites, but its primary site is not known, it is called cancer of unknown primary(CUP)[mect18].

Many cancers have the high chance of being cured if diagnosed and treated adequately. When cancer spreads to different parts of the body, it needs the same treatment as that of the primary site. Hence, knowing where the cancer started will direct the treatment in proper direction. Determining the primary site is very important as it helps doctors to determine the type of treatment. This becomes extremely important for those types of cancers that respond only to a specific treatment [mect18][DHR09].

All cancers begin when one or more genes in a cell mutate and create abnormal proteins that cause cells to multiply uncontrollably. Genes are composed of pieces of DNA and are present inside our cells. DNA, present in each cell is considered to be the genetic blueprint. Any change to DNA is called mutation. These mutations play an important role in cancer. Mutations bring change in the process of making protein by the cells, which affects cell's growth and division into new cells. Certain mutations will cause the cells to grow uncontrollably, which lead to cancer [DHR09]. [WLD01] states that DNA methylation is an alternative way in cancer.

Research shows that DNA methylation markers can be used for diagnosis of common cancers [ZZH$^+$17][LLK$^+$18]. With the advancement of technology in the field of medicine, large amount of

cancer data could be easily collected and made available to the research community. Researchers also suggest that artificial intelligence is better and faster in detecting cancer than clinicians [Tuc18]. Machine learning techniques can identify patterns in complex datasets which are able to accurately predict cancers. Several studies are based on applying machine learning algorithms to microaray gene expression data to classify the cancer types [RTR+02] [FCD+00]. Probabilistic approaches are used on genome-wide DNA methylation data in order to find the primary origin of cancer [KLC+17].

1.1 Objective

The main objective of this thesis is to apply machine learning techniques on genome-wide DNA methylation data inorder to identify the tissue origin of cancer. Different approaches are followed to select important features and apply machine learning models like Support Vector Machines, random forest classifiers, decision trees classifiers and k-nearest neighbours classifiers. Different models are trained and the best model is tested on an independent dataset, different from the dataset used for training. The performance of different models are evaluated on various metrics and the results are reported.

1.2 Outline

Chapter 1 gives a brief introduction to cancer, importance of methylation markers in cancer diagnosis and the proposed approach to identify the tissue origin of cancer.

In Chapter 2, we will discuss the previous research studies related to use of methylation markers in identifying tissue origin of cancer and the proposed approach using machine learning. It also includes popular algorithms of machine learning.

In Chapter 3, we will discuss in detail the methodology used in the analysis, different approaches used for selection of features is also discussed.

In Chapter 4, we will discuss the process in which experiments were performed using different models and compare the results.

In Chapter 5, we will summarize the results and provide insights about future work.

Chapter 2

Background and Preliminaries

2.1 Background

There has been a lot of research in the field of cancer in last few decades.[CW06] shows that use of machine learning methods have increased the accuracy of predicting cancer susceptibility, recurrence and mortality by 15%-20%. According to [KEE+15], papers related to application of machine learning methods in the field of Cancer prognosis and prediction had been on a rise. In [KEE+15], different machine learning methods were summarized. Research in [KEE+15] is mostly focused on validating different research studies related to using machine learning methods on classification of low or high risk groups.

In [BDH+97], an artificial neural network was used to predict the survival of patients suffering from colorectal cancer. [BDH+97] also states that neural networks could better predict the outcome than existing clinicopathological methods.

While [KEE+15] states that majority of algorithms used in classifying high-risk and low-risk tumors [HTW+18] states different approaches of deep learning, a subset of machine learning, is also used in cancer detection and diagnosis.

[KEE+15] also states that majority of the studies in the area of cancer prognosis and prediction were based on gene expression profiles, clinical variables as well as histological parameters. Research in [Tu18] was focussed on miRNA expression profiles and DNA methylation expression profiles, where they have adopted Pearson's correlation analysis and principal component analysis as the two feature selection methods on a combination of miRNA and DNA methylation expression profiles. However, there are other feature selection methods like Chi-squared feature extraction method discussed in [CD04] where they applied Chi-squared feature extraction method on microarrays. In

this thesis, we applied Chi-squared feature extraction method and principal component analysis (PCA) for selecting the features on DNA Methylation expression profiles with a motive to accurately predict the tissue origin in cancer.

In [TWY+18], they focused on machine learning models like random forest classifiers, support vector machines and k-nearest neighbors classifier on the features extracted from Pearson's correlation analysis and PCA on a combination of miRNA and DNA methylation expression profiles. In this thesis, we applied machine learning models support vector machines, decision trees classifier, random forest classifiers, k-nearest neighbors classifier on features extracted using Chi-squared method and PCA on just the DNA methylation expression profiles.

2.2 Preliminaries

Before discussing about the specific application in depth, lets discuss the basic concepts used in this thesis. This section gives a basic overview of the proposed models for identification of tissue origin in cancer.

2.2.1 Machine Learning

Arthur Samuel coined the term machine learning in 1959 [Sam00] as "Field of study that gives computers the ability to learn without being explicitly programmed". In general, computers can be trained to learn automatically without human intervention. Machine learning algorithms build complex mathematical models based on the training data, which can be used to make predictions for new or unseen data. Computers can be trained to learn complex patterns from high-dimensional data, which could be otherwise very difficult for human to process and identify patterns. With the advancement in technology and computation resources, training machine learning models on large sets of data is possible. Hence, there is a rise in use of machine learning algorithms to solve various real world problems in many fields.

There are different categories in machine learning like supervised learning, unsupervised learning and reinforcement learning. In supervised learning, data samples are labeled, and models are trained to accurately predict the trained labels. For example, trying to recognize hand written numbers, where models are trained on different hand written images that are labeled with corresponding number. But in unsupervised learning, data sample are unlabeled and models automatically try to figure out patterns in data itself or cluster them. On the other hand, reinforcement learning aims

to take suitable action so that reward for the given situation is maximized.Figure 2.1 from [Gra17] shows different categories in machine learning. In this thesis, we applied supervised learning methods, where training samples where labelled with the corresponding tissue of origin. Models try to recognize patterns in these labeled samples and accurately predict on unseen data.

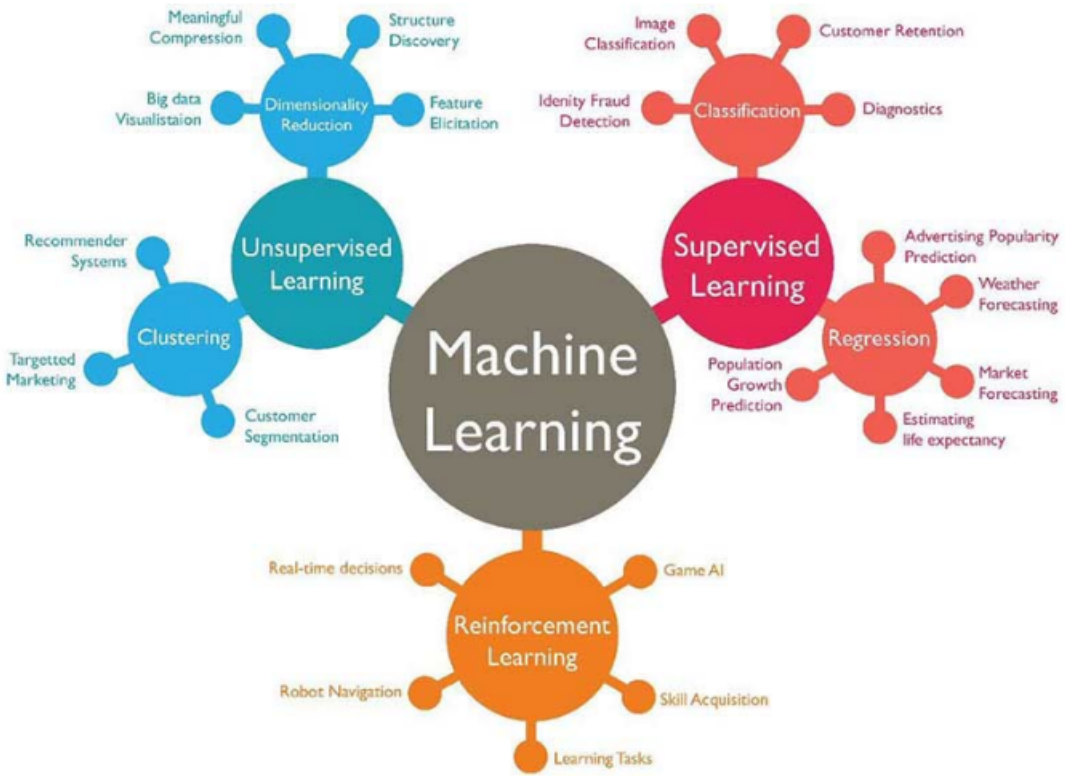


Figure 2.1: Categories in machine learning

2.2.2 Classification

Classification is the process of categorizing the data samples into different classes. Classification is considered an instance of supervised learning, For example, detection of spam emails is a classification problem. In this example, it is considered as a binary classification where there are two class labels, spam and not spam. Classification problems can also be multi-class where there are more

than two classes.For example, identifying the species of Iris flower (Iris setosa, Iris virginica and Iris versicolor), given features like length and width of sepals and petals is a multi-class classification problem. This thesis also deals with multi-class classification.

2.2.3 Selected Models

In this section, we will discuss the different models used for predictive analysis. In this thesis, we choose four popular models, support vector machines, random forests classifier, decision trees classifier and k-nearest neighbour classifier.

2.2.3.1 Decision Trees

Decision tree builds the model in the form of a tree, utilizing if-then rule set [Mur12a]. The rules are learned sequentially using the training data one at a time. Figure 2.2 from [Le18b] shows an overview of a decision tree. The tree is constructed in top-down manner where each node represents a condition, and its branches represent its outcomes. All the leaves represent the output labels. Decision trees can easily overfit the data by generating too many branches and a small change in the data can lead to a large change in structure of the decision tree.

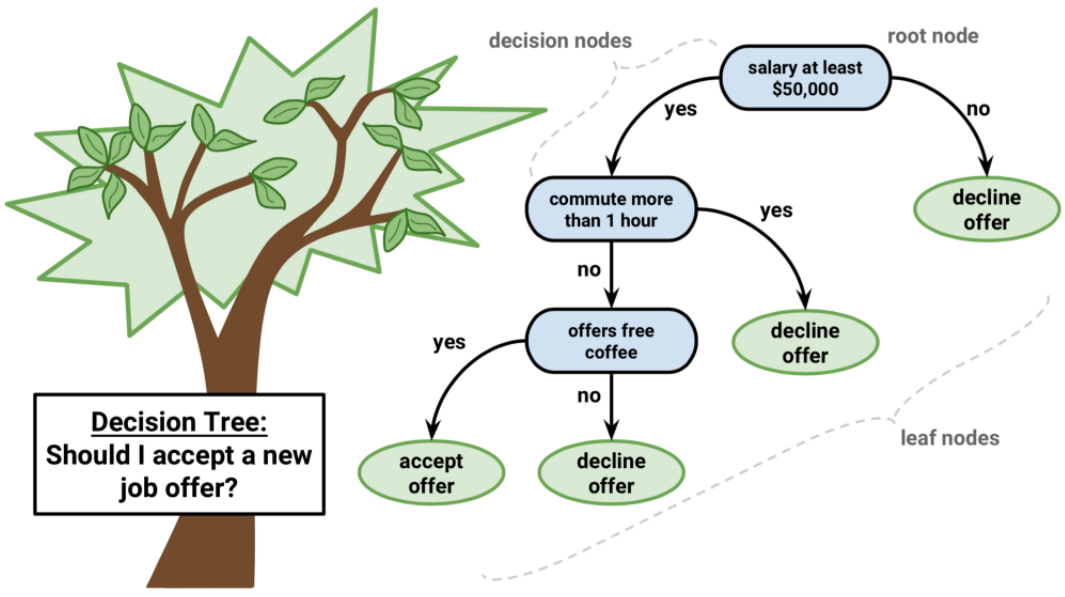


Figure 2.2: Overview of decision Tree

2.2.3.2 Random Forest Classifiers

Random Forest is a supervised learning algorithm in which multiple decision trees are built and then merged together in order to get stable and accurate prediction [Mur12a]. While building the trees and splitting the nodes, a random subset of features are taken into consideration. Unlike searching for the best features as in decision trees, by randomly choosing the thresholds for each feature, trees can be made more random. Figure 2.3 from [Koe17] gives an overview of random forest algorithm.

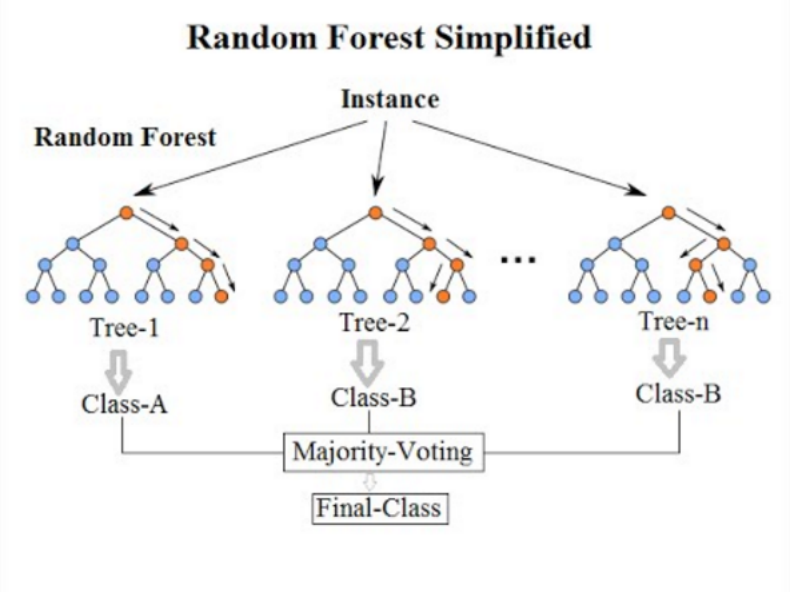


Figure 2.3: Overview of Random Forest

2.2.3.3 Support Vector Machines

A support vector machine (SVM) is a discriminative classifier which tries to figure out a hyperplane that segregates different classes. It is also called large margin classifier. In other words, given training data, it identifies a hyperplane which is at maximum distance from all the clusters [Mur12b]. In case of binary classification, where number of target classes are two, we can visualize this hyperplane in a two-dimensional space as a line dividing the space into 2 parts where each part

corresponds to a class. Figure 2.4 from [Dab18] gives an overview of an SVM.

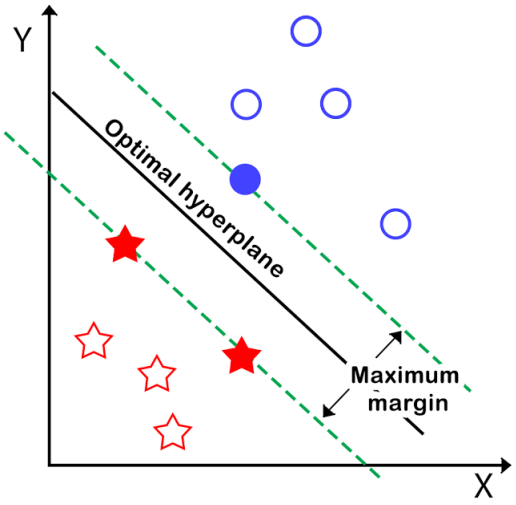


Figure 2.4: Overview of SVM

2.2.3.4 K Nearest Neighbors

K-nearest neighbor (KNN) is a lazy learning algorithm which stores all instances corresponding to training data in n-dimensional space. For an unseen data point it analyzes the closest k number of instances and returns the most common class as prediction. In distance-weighted nearest neighbour algorithm, it weighs the contribution of each of the k neighbors according to their distance an gives greater weight to closest neighbors. Figure 2.5 from [Le18a] gives an overview of KNN.

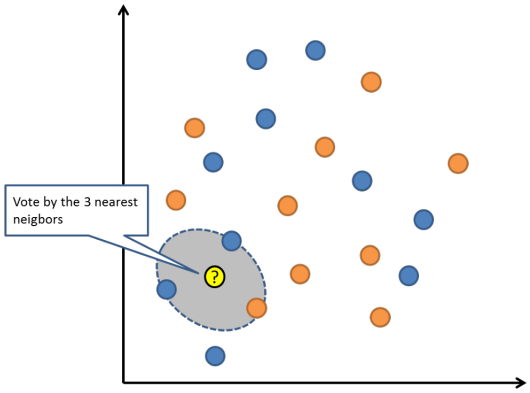


Figure 2.5: Overview of KNN

2.2.4 Evaluation Metrics

In machine learning, the main goal is to make predictions on unseen data. We should be very sure that the model makes accurate predictions on unseen data before using it in real world applications. Especially when we are dealing with treatment of patients based on the model predictions, we should ensure that the model has given the accurate results so that it does not cause any adverse impact. Hence, evaluating different machine learning models based on metrics is extremely important. Various metrics are available and these metrics depend on the type of the problem we are addressing: classification or regression. In this thesis, we consider metrics used in classification.

Before applying machine learning techniques , available data is divided into two categories, training set and test set. Training set is used to train the model and test set is used as unseen data and trained model is used to predict the labels of the test set. We can then evaluate the performance of the model based on the predicted labels and actual labels. Below sections discuss in detail about various evaluation metrics.

2.2.4.1 Confusion Matrix

In classification problems, there can be two or more output labels. Confusion matrix is a table with 4 different combinations of predicted and actual values as shown in figure 2.6 . It comprises

of true positives, true negatives, false positives and false negatives which gives some useful insights on how many samples where correctly classified and how many were not.

True Positive (TP): Samples which are predicted positive and they were actually positive. For example, a women is predicted to be pregnant and she actually is.

True Negative (TN): Samples which are predicted negative and they were actually negative. For example, a man is predicted as not pregnant and he actually is not.

False Positive (FP): Samples which are predicted positive and they were actually negative. For example, a man is predicted as pregnant but he is actually not.

False Negative (FN): Samples which are predicted negative and they were actually positive. For example, a women is predicted as not pregnant but she is actually is. Confusion matrix help to calculate more advanced classification metrics such as precision, recall, specificity and sensitivity.

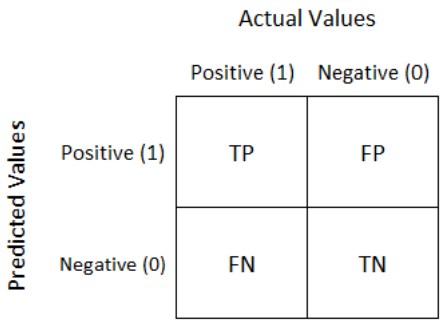


Figure 2.6: Predicted labels vs Actual labels

Figure 2.7 shows the confusion matrix for multi-class classification, where the diagonal elements represent the number of samples where the predicted label was equal to actual label. Off diagonal entries shows the samples that are miss-labelled.

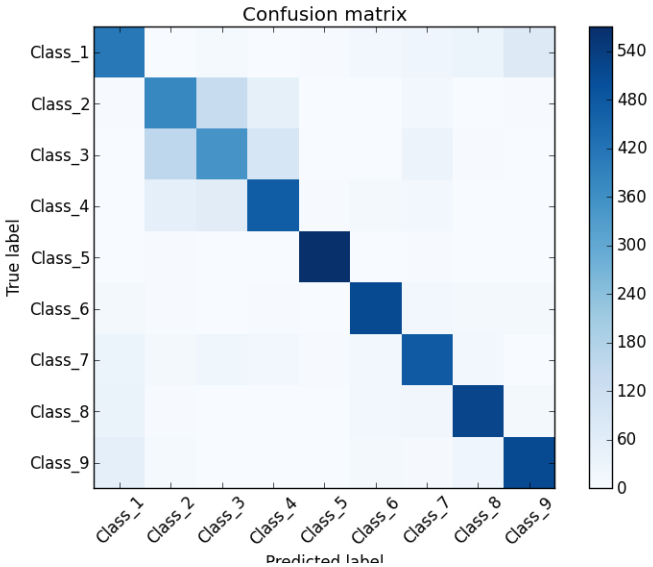


Figure 2.7: Confusion matrix

2.2.4.2 Precision

Of all the samples which are predicted as positive, how many are actually positive is defined as precision.

$$Precision = \frac{TruePositives}{TruePositives + FalsePositives} \tag{2.1}$$

2.2.4.3 Recall

Of all the positive samples, how many are predicted positive is defined as recall.

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{2.2}$$

2.2.4.4 F1 Score

F1 Score is harmonic mean of precision and recall. F1 score tells us how many instances the classifier predicted correctly and also tells how robust it is. The range for F1 Score is [0, 1].

$$F1Score = \frac{2 * (Precision * Recall)}{Precision + Recall} \tag{2.3}$$

2.2.4.5 Classification Accuracy

Classification accuracy is the percentage of samples which were correctly predict of all the samples. It is the most important metric in classification problems. It can be calculated from the values in confusion matrix.

$$ClassificationAccuracy = \frac{TruePositives + TrueNegatives}{Sizeofpredictedpopulation} \tag{2.4}$$

Chapter 3

Methodology

3.1 Data Collection

The Cancer Genome Atlas (TCGA) is a collaboration between the National Cancer Institute (NCI) and the National Human Genome Research Institute (NHGRI). TCGA has generated genomic, epigenomic, transcriptomic, and proteomic data of primary cancer and normal sample types related to 33 different cancer types. TCGA has made this data publicly available for research purposes. The dataset used for this thesis was DNA methylation expression profiles collected from The Cancer Genome Atlas (TCGA), consisting of 9756 samples representing 33 types of cancer. This dataset was used for training and validation of different machine learning models.

DNA methylation is an epigenetic mark which is frequently associated with transcriptional activity of genes[JLR11]. TCGA DNA methylation data is generated by 450k methylation arrays (HymmanMethylation450 containing 485512 probes covering 99 percent of RefSeq genes). Epigenetics is the study of heritable changes in gene activity that do not involve alterations to the genetic code a process by which methyl groups are added to the DNA molecule [LSB10]. Methylation can change the activity of a DNA segment without changing the sequence. Gene Expression Omnibus (GEO) is a public functional genomics data repository supporting MIAME- compliant data submissions. DNA methylation expression profiles collected from this repository is used for testing the best model obtained based on training with the TCGA dataset.

3.2 Data Preparation

In this section we will discuss the steps taken to combine and clean the dataset collected from TCGA and GEO to obtain the input and output features for applying machine learning models.

3.3 Data Description

The DNA methylation dataset is downloaded from TCGA data portal (https://portal.gdc.cancer.gov/). This dataset was used for training and validation of different machine learning models. There are 9756 .txt files each of which comprise of DNA methylation expression profiles of a single patient. Each file stores composite, beta value, chromosome details, coordinate details etc. Composite and beta values are extracted from each of these files and a matrix is created which includes beta values of all the composites of 9756 patients. The dimensions of this matrix is 485577 rows by 9756 columns. This matrix is too large to load into a computer memory. So, it was divided into multiple files by grouping the composites according to chromosomes. There are 24 chromosomes, chromosome 1 to chromosome 22 and then chromosomes X and Y. The complete DNA methylation TCGA data is split into 24 different files, each containing beta values of probes which belong to single chromosome for all the 9756 patients. Table 3.1 gives details about total number of probes present in each of the chromosomes. Since X and Y chromosomes are related to sex, these two chromosomes are not considered in the analysis.

Each Cancer type has a specific label and description of these labels are indicated in Table 3.2 [(NC]. Figure 3.1 shows the frequency distribution of each of the class labels.

.

Chromosome Name	**Total Probes**
Chromosome 1	46850
Chromosome 2	34815
Chromosome 3	25163
Chromosome 4	20469
Chromosome 5	24331
Chromosome 6	36614
Chromosome 7	30016
Chromosome 8	20958
Chromosome 9	9871
Chromosome 10	24390
Chromosome 11	28796
Chromosome 12	24543
Chromosome 13	12285
Chromosome 14	15078
Chromosome 15	15261
Chromosome 16	21970
Chromosome 17	27879
Chromosome 18	5923
Chromosome 19	25521
Chromosome 20	10381
Chromosome 21	4245
Chromosome 22	8562

Table 3.1: Total probes in each chromosome in TCGA dataset

Label	Description
LAML	Acute Myeloid Leukemia
ACC	Adrenocortical carcinoma
BLCA	Bladder Urothelial Carcinoma
LGG	Brain Lower Grade Glioma
BRCA	Breast invasive carcinoma
CESC	Cervical squamous cell carcinoma and endocervical adenocarcinoma
CHOL	Cholangiocarcinoma
LCML	Chronic Myelogenous Leukemia
COAD	Colon adenocarcinoma
CNTL	Controls
ESCA	Esophageal carcinoma
FPPP	FFPE Pilot Phase II
GBM	Glioblastoma multiforme
HNSC	Head and Neck squamous cell carcinoma
KICH	Kidney Chromophobe
KIRC	Kidney renal clear cell carcinoma
KIRP	Kidney renal papillary cell carcinoma
LIHC	Liver hepatocellular carcinoma
LUAD	Lung adenocarcinoma
LUSC	Lung squamous cell carcinoma
DLBC	Lymphoid Neoplasm Diffuse Large B-cell Lymphoma
MESO	Mesothelioma
MISC	Miscellaneous
OV O	varian serous cystadenocarcinoma
PAAD	Pancreatic adenocarcinoma
PCPG	Pheochromocytoma and Paraganglioma
PRAD	Prostate adenocarcinoma
READ	Rectum adenocarcinoma
SARC	Sarcoma
SKCM	Skin Cutaneous Melanoma
STAD	Stomach adenocarcinoma
TGCT	Testicular Germ Cell Tumors
THYM	Thymoma
THCA	Thyroid carcinoma
UCS	Uterine Carcinosarcoma
UCES	Uterine Corpus Endometrial Carcinoma
UVM	Uveal Melanoma

Table 3.2: Description of Labels for TCGA dataset

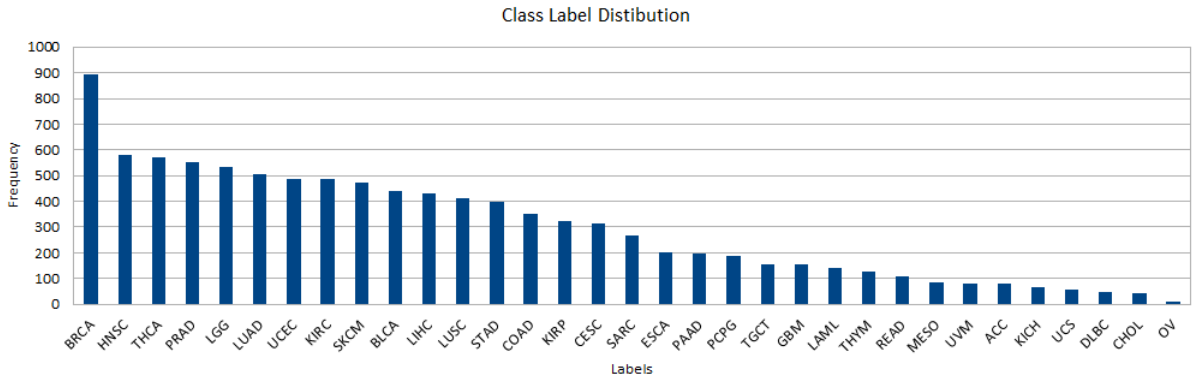


Figure 3.1: Class label distribution of TCGA dataset

Samples of the class labels whose count was less than 100 where removed from the study. Figure 3.2 shows the labels in this thesis.

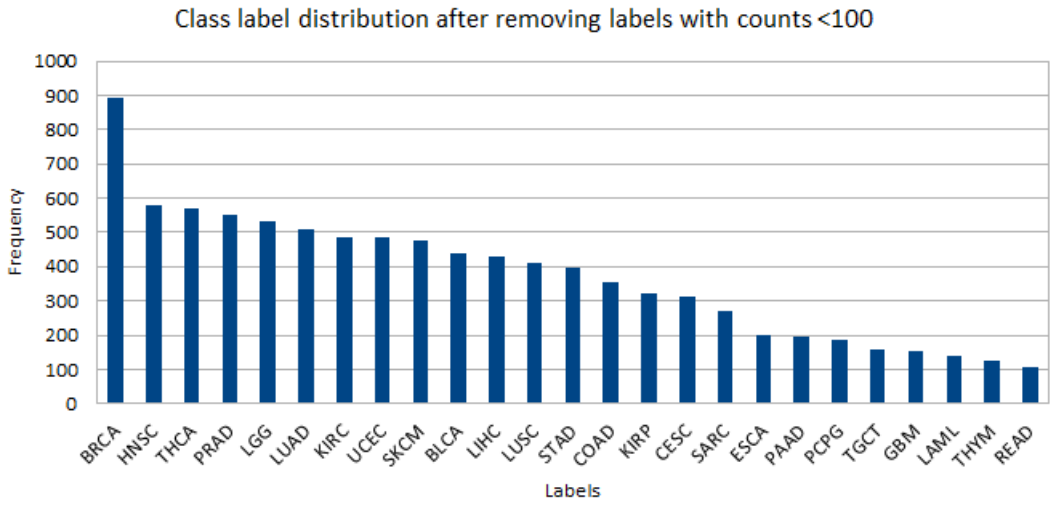


Figure 3.2: Class label distribution of TCGA dataset after removing the labels with counts less than 100.

Similarly, the DNA methylation dataset was downloaded from GEO data portal (https://www.ncbi.nlm.nih.gov/geo/). GEO DNA methylation dataset has 2052 samples and 13 cancer types. The downloaded files where in .txt format. Each file is specific to a particular type of cancer and have information of patients diagnosed with same type of cancer. Each file have patients information like age, sex, ethnicity, tumor type, etc. and also a series matrix comprising of the beta values of the probes. This matrix is extracted for all the files and combined together into a single matrix which have beta values of all the probes and all the cancer types. Then this matrix is sorted according to chromosomes and is subdivided into 24 different files. Each file have all the beta values pertaining to probes of a particular chromosome. Similar to TCGA, chromosome X and Y were not included in the study. Labels to these samples were assigned according to TCGA notation as mentioned in table 3.2. Frequency distribution of different class labels in the GEO dataset are displayed in figure 3.3. Only the samples whose labels were other than noTCGA were included in the study which turned out to be 1596 out of 2052 samples.

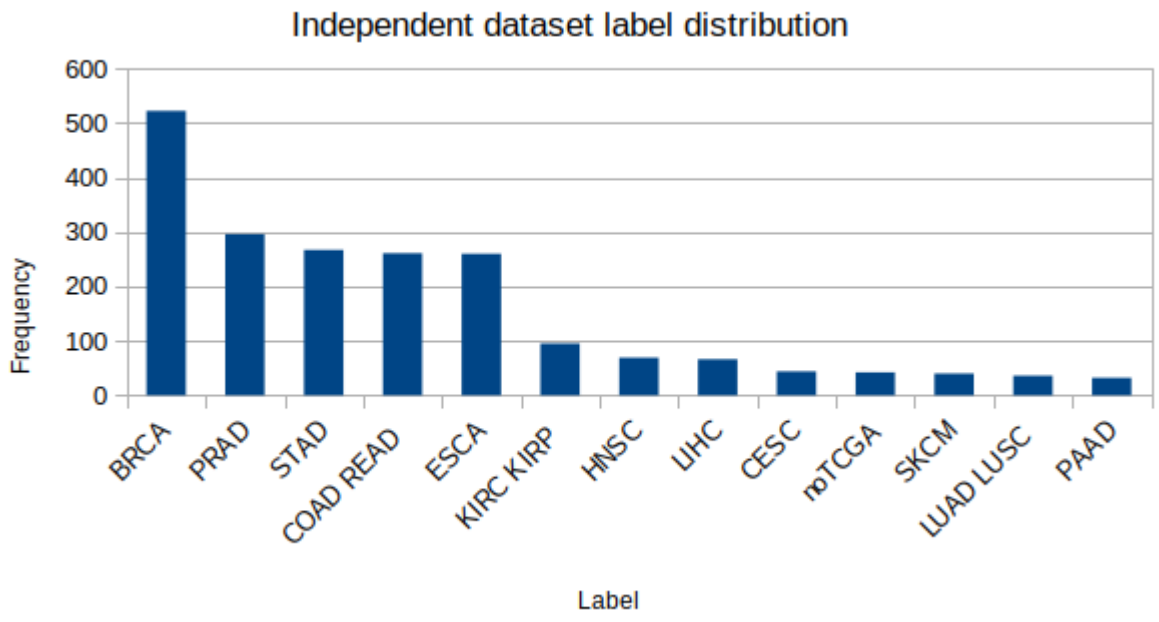


Figure 3.3: Frequency distribution of class labels of the independent dataset.

3.3.1 Feature Selection

Feature selection is the process of selecting relevant features from the raw data which can best classify the samples to their respective labels. In this thesis, we have applied two different kinds of feature selection methods, principal component analysis (PCA) and Chi-squared test. Further details about both of the methods are given in section 3.3.1.1 and 3.3.1.2. Machine learning models were applied to data obtained from these two feature selection methods separately.

3.3.1.1 Principal Component Analysis

Principal component analysis (PCA) is a statistical procedure to transform high dimensions of data into lower dimensions [Mur12c]. PCA reduces the complexity in high dimensional data by retaining the patterns in them. It converts a set of correlated features into a set of linearly uncorrelated variables called as principal components (PC). If there are n samples and f features, then PCA would form smaller of n-1 and f principal components. The first principal component would be a variable of maximum variance, second component would have the highest variance with the constraint that it is orthogonal to the preceding components and so on.

PCA cannot be applied on datasets with missing values. Both TCGA DNA Methylation dataset and GEO DNA Methylation dataset where had a few probe values missing. In TCGA all the probes that were missing in any of the samples were removed and also the probes whose mean for all the samples is less than 0.2 was removed to reduce the noise in the data. Only these set of features where considered in the GEO dataset, and any missing values for these features in GEO dataset was handled by filling in with the mean values of respective probe for all the samples. This process is discussed further in section 3.4.1.1.

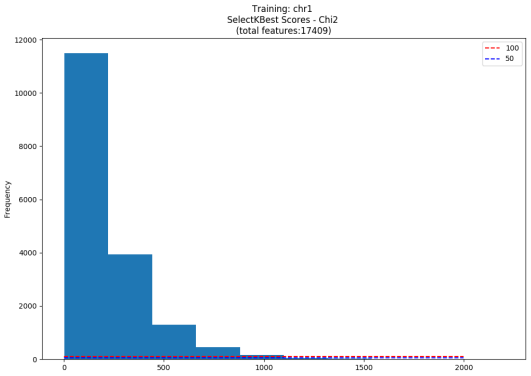
PCA was applied on this cleaned dataset to reduce the number of dimensions. In this thesis, the dataset is divided according to different chromosomes, we applied PCA to reduce dimensionality on each of the chromosomes separately. Table 3.1 shows the number of features in each chromosome. These where reduced to 100 principal components for each chromosome. This was achieved using Scikit-learn PCA module. Separate models were trained for different chromosomes and a model for a chromosome was fit with TCGA data to transform into 100 principal components. Same model was also used to transform the corresponding chromosome features in GEO dataset to 100 principal components. The reason for using the same model is to maintain consistency of the features used in both of the datasets. These 100 principal components formed for each chromosome are merged

together to form a training dataset of 9756 samples with 2200 features and an independent dataset of 1596 samples with 2200 features.

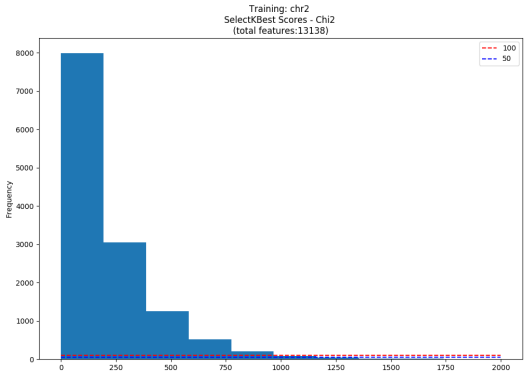
3.3.1.2 Chi-Squared Test

Pearson's Chi-squared test is often called as Chi-squared test. The Chi-squared test is used to determine whether there is a significant difference between the expected frequencies and the observed frequencies in one or more categories [GS12]. SelectKBest module of scikit-learn is used to achieve this purpose with Chi-squared test as scoring function. It measures the dependence between stochastic variables which weeds out the features that are most likely to be independent of class and therefore irrelevant for classification.

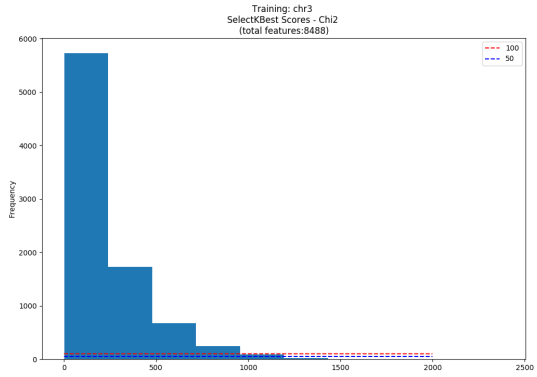
SelectKBest is applied to find scores of all the features in each chromosome. Figure 3.4- Figure 3.7 display the the frequency distribution of theses scores. All these graphs show that majority of the features have scores near to 0 and less than 50 features in each chromosome had very high scores in the range of 1500-2000. These top 50 features in each chromosome where extracted and combined to form 1100 features for 9756 samples in the training dataset. This dataset was used for training the machine learning models. Same features are extracted from the GEO dataset. If there were any values missing, these were imputed using the methods discussed in Section 3.4.1.1 and Section 3.4.1.2.



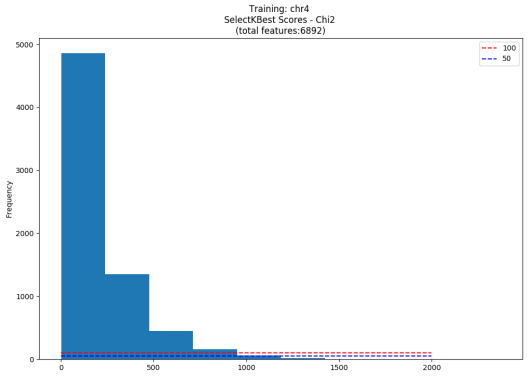
(a) Chromosome 1



(b) Chromosome 2

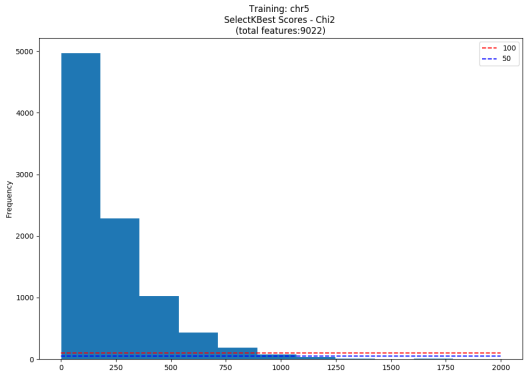


(c) Chromosome 3

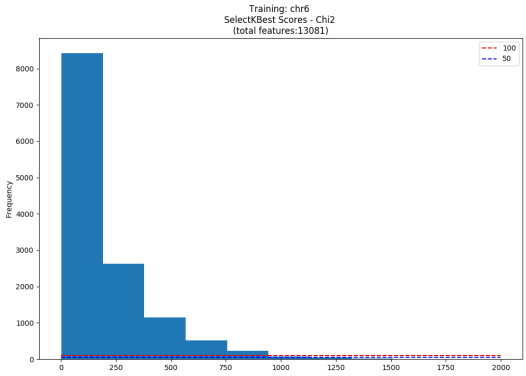


(d) Chromosome 4

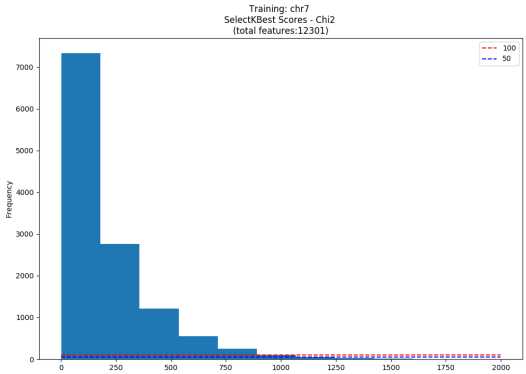
Figure 3.4: Chi-squared test scores of features in Chromosome 1-4.



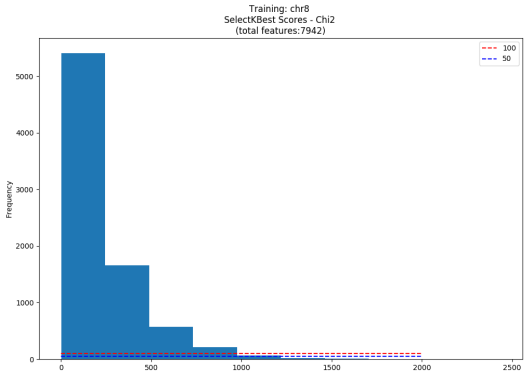
(a) Chromosome 5



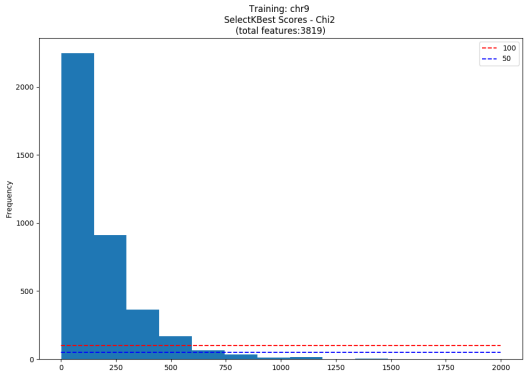
(b) Chromosome 6



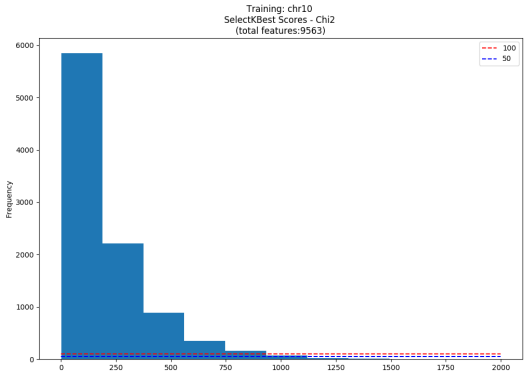
(c) Chromosome 7



(d) Chromosome 8

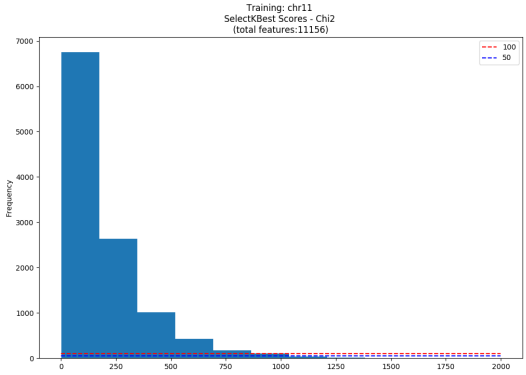


(e) Chromosome 9

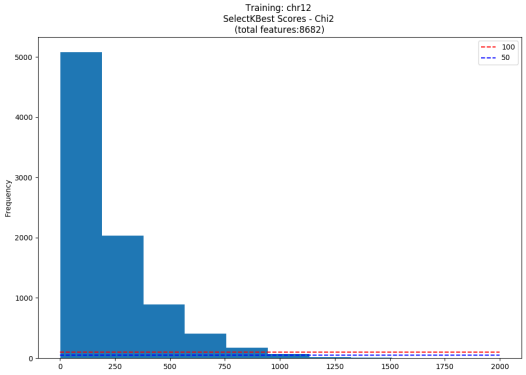


(f) Chromosome 10

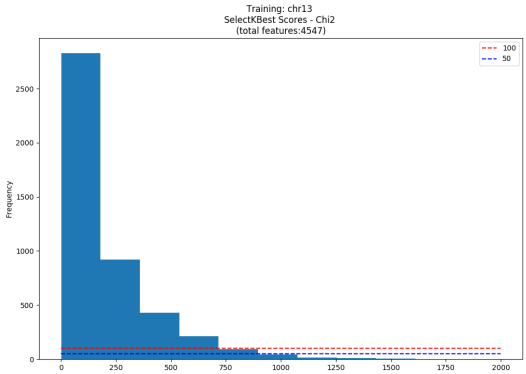
Figure 3.5: Chi-squared test scores of features in Chromosome 5-10.



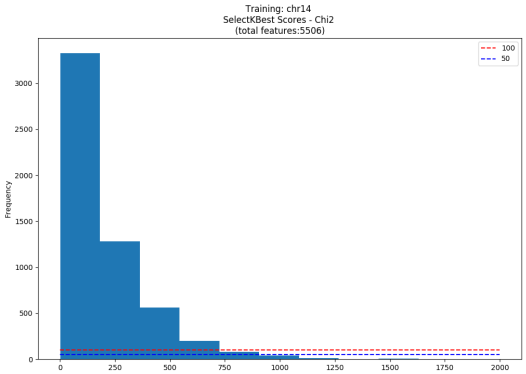
(a) Chromosome 11



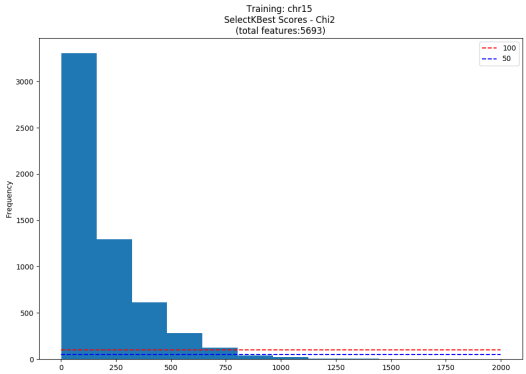
(b) Chromosome 12



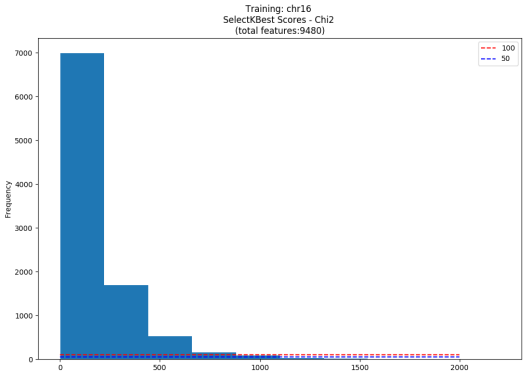
(c) Chromosome 13



(d) Chromosome 14

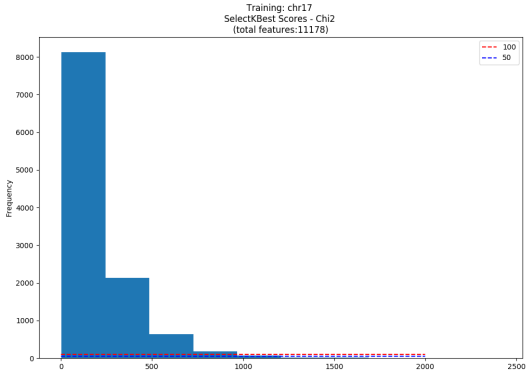


(e) Chromosome 15

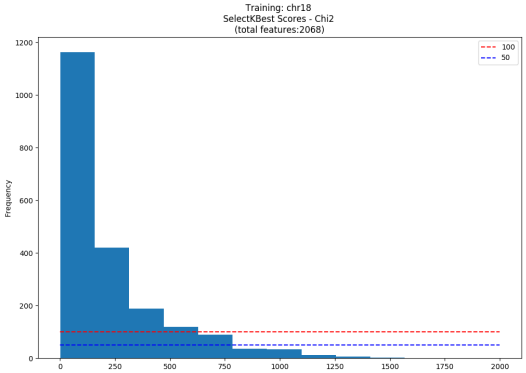


(f) Chromosome 16

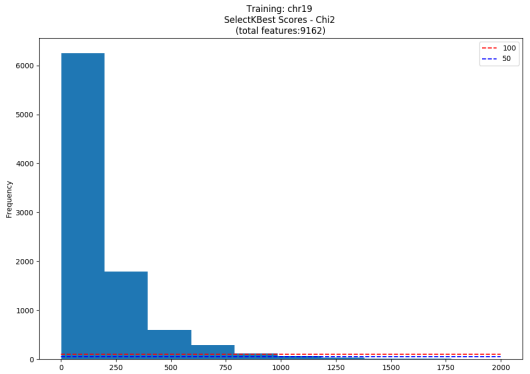
Figure 3.6: Chi-squared test scores of features in Chromosome 11-16.



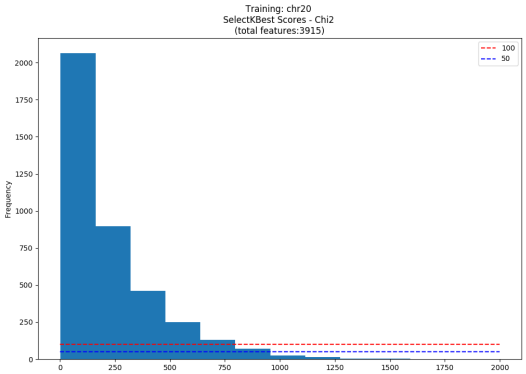
(a) Chromosome 17



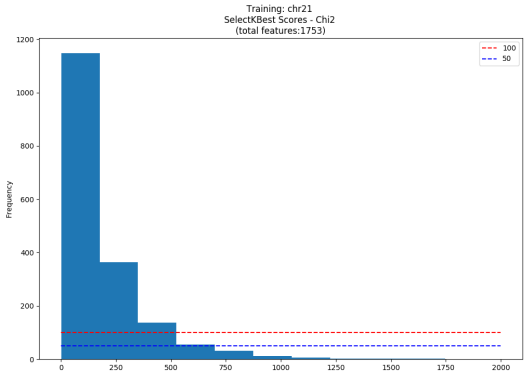
(b) Chromosome 18



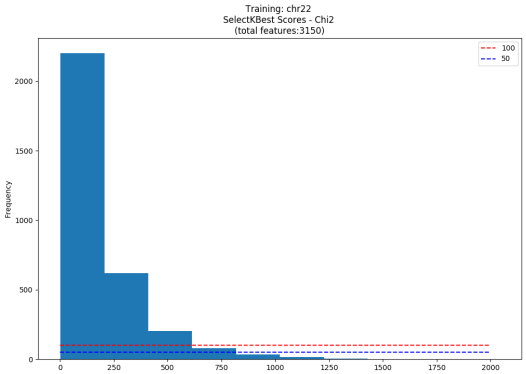
(c) Chromosome 19



(d) Chromosome 20



(e) Chromosome 21



(f) Chromosome 22

Figure 3.7: Chi-squared test scores of features in Chromosome 17-22.

3.4 Data Preprocessing

Data preprocessing is a technique used to convert raw data into a clean dataset. When data is gathered from different sources, raw data might not meet the requirements and cannot be used directly in the analysis. This step is necessary before applying any machine learning model.

3.4.1 Missing value imputation

While combining data from different sources there are a few missing values. Since TCGA is used for training, probes with any values missing are removed. Since we need the same features used in training to be present in test dataset (GEO DNA methylation dataset), required probes in GEO dataset were missing. These missing values were imputed based on two techniques as discussed in section 3.3.1.1 and section 3.3.1.2. The two different datasets obtained after applying the imputation methods were used separately.

3.4.1.1 Imputation with the mean of complete the dataset

In this method, mean of all the samples for each feature is calculated and missing values were imputed with this mean value. GEO dataset had missing values in the required features. Hence, mean of the samples were calculated and missing values were imputed with this value.

3.4.1.2 Imputation with the mean of k nearest neighbors

In this method, k nearest neighbors algorithm was applied on the training (TCGA) dataset using scikit-learn NearestNeighbors module. The model is fit using the training data. Then this model was used to find the nearest neighbors for test (GEO) dataset. In simple terms, for a sample A in test dataset we are finding k nearest neighbors (n1, n2, n3....nk) in the training dataset. For each feature f_i, mean of all the k neighbors is calculated i.e. mean(f_i)=((n1, n2, n3 ...nk)/k) and if feature f_i is missing in sample A, missing value is imputed with the computed mean i.e., mean(f_i). In this thesis, we used k as 5.

3.5 Data splitting

Training different machine learning models and hyper-parameter tuning of those models were performed using the TCGA dataset. TCGA data was split into 80 percent training, which is used to train the models and and 20 percent cross-validation, which is used for hyperparameter tuning of

the models. GEO dataset was used to test the performance of the best model selected after the hyperparameter tuning. While splitting the data into 80 percent training and 20 percent validation, we specified a random seed (any random number), which ensured the same data split every time the program was run.

3.6 Hyperparameter tuning using 10 fold cross validation

Machine learning models have parameters and hyperparameters. Model parameter is a configuration that is internal to the model and whose value can be estimated from the data. A hyperparameter is a configuration that is external to the model and whose value cannot be estimated from the data. These hyperparameters are specified by the practitioner and are often used to estimate model parameters. They have to be tuned to get the best performance out of the model. We used k-fold cross-validation technique to tune the hyperparameters with k as 10. In 10 fold cross-validation technique, dataset is divided into 10 sets and for each fold, the current set is used as the test set and the remaining 9 sets are used as training set. The model is trained on the training set and then evaluated on the test set. Figure 3.8 shows how 10-fold cross validation is performed. We used Grid Search module of scikit-learn library to implement cross validation to find the best hyperparameter values. Each machine learning model has different hyperparameters, each of which are discussed in the following sections.

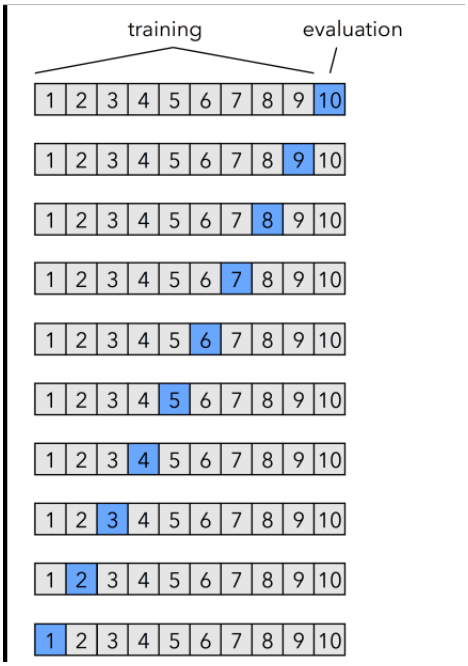


Figure 3.8: 10-fold cross validation

3.6.1 Hyperparameters in support vector machines

In support vector machines, the following hyperparameters were tuned using grid search with 10 fold cross-validation.

kernel specifies the kernel type to be used in the algorithm.

C is the penalty parameter or the error term

3.6.2 Hyperparameters in decision trees

In decision trees, the following hyperparameters were tuned using grid search with 10 fold cross-validation.

criterion is a function used to measure the quality of split.

min_samples_leaf is the minimum number of samples required to be at a leaf node.

max_depth is the maximum depth of the tree.

min_samples_split is the minimum number of samples required to split an internal node.

3.6.3 Hyperparameters in random forest classifier

In random forest classifier , the following hyperparameters were tuned using grid search with 10 fold cross-validation

criterion is a function used to measure the quality of split.

min_samples_leaf is the minimum number of samples required to be at a leaf node.

max_depth is the maximum depth of the tree.

min_samples_split is the minimum number of samples required to split an internal node.

3.6.4 Hyperparameters in k nearest neighbors

In k-nearest neighbors classifier, the following hyperparameters were tuned using grid search with 10 fold cross validation.

n_neighbors is the number of neighbours to use.

weights is the weight function used in prediction. There are two types of weight functions, uniform and distance. In uniform weights, all points in the neighborhood are weighted equally. In distance, points are weighted by the inverse of their distance closer neighbors of a query point will have a greater influence than neighbors which are further away.

3.7 Training and testing phase

After tuning the hyperparameters, the entire training data was used to fit the model with best values for hyperparameters. We used the test (GEO) dataset to measure the performance of the trained model.

3.8 Performance evaluation of selected models

The last step of the predictive analysis is to evaluate the performance of the model. In this thesis, we evaluated the performance of different models using accuracy, precision, recall and F1 score.

Chapter 4

Results

The pre-processed TCGA dataset (Section 3.2) were split into 80% training set and 20% validation set. Different machine learning models were trained on the training set and hyperparameters where tuned on the validation set. After getting the best hyperparameters, models were fit with the complete TCGA dataset. These models were then evaluated based on the GEO dataset. A combination of feature selection methods and missing value imputation methods are used. They are further discusses in Sections 4.1, 4.2 and 4.3

4.1 Feature selection using Chi-squared test and missing value imputation using KNN

In these experiments, features are selected from the TCGA dataset using scikit-learn Chi-Square method (Section 3.3.1.2). These selected features are extracted from the GEO dataset. For the missing values in the GEO dataset, values are imputed using KNN mean imputation method discussed in Section 3.4.1.2. The TCGA dataset is used for training the models and the GEO dataset is used to test the model performance. the sections below discuss the performance of each of the models and the best hyperparameters.

4.1.1 Support Vector Machines

Best hyperparameters C:1 , kernel: linear

Remarks

Support vector machines showed the best performance. Table 4.1 shows the overall performance of the SVM model. Fig. 4.1 shows the confusion matrix and Table 4.2 shows the classification

performance metrics.

Model Name	**SVM**
Best training accuracy	0.9476164826
Test set accuracy	0.9445341949
Test precision	0.9440388388
Test recall	0.9445341949
Test fscore	0.9439602647
Independent accuracy	0.8615288221
Independent precision	0.8908000751
Independent recall	0.8615288221
Independent fscore	0.8736936315

Table 4.1: Overall evaluation with SVM - Chi-squared test - KNN imputation

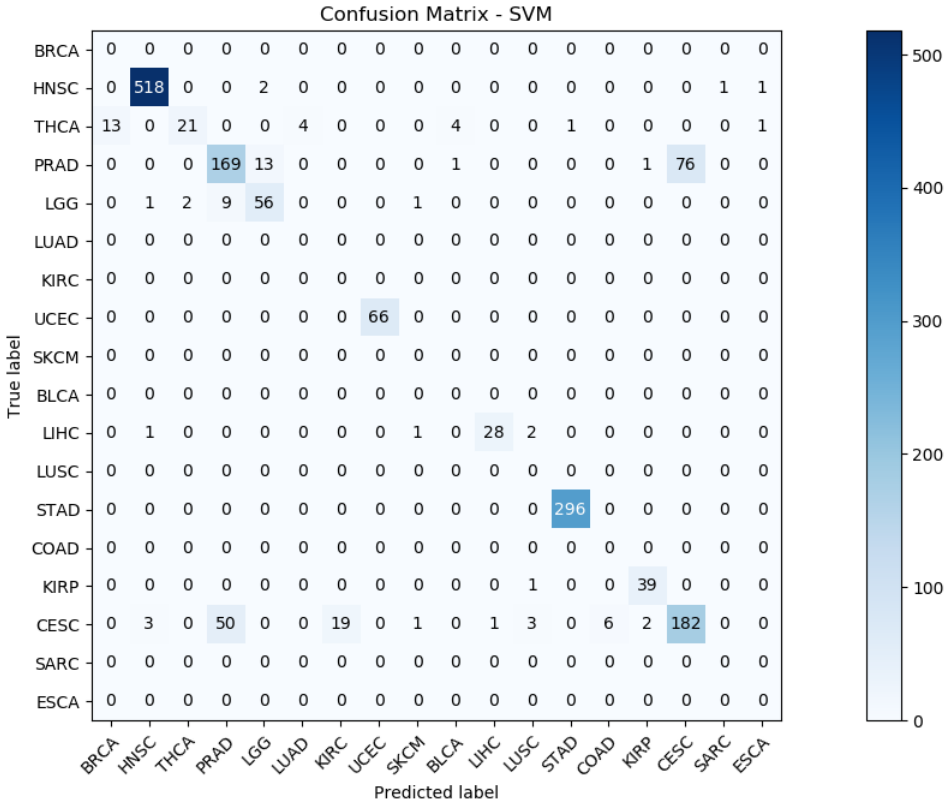


Figure 4.1: Confusion matrix with SVM - Chi-squared test - KNN imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.99	0.99	0.99	522
CESC	0.91	0.48	0.63	44
ESCA	0.74	0.65	0.69	260
HNSC	0.79	0.81	0.80	69
LAML	0.00	0.00	0.00	0
LGG	0.00	0.00	0.00	0
LIHC	1.00	1.00	1.00	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.97	0.88	0.92	32
PCPG	0.00	0.00	0.00	0
PRAD	1.00	1.00	1.00	296
SARC	0.00	0.00	0.00	0
SKCM	0.93	0.97	0.95	40
STAD	0.71	0.68	0.69	267
TGCT	0.00	0.00	0.00	0
UCEC	0.00	0.00	0.00	0
avg / total	0.89	0.86	0.87	1596

Table 4.2: Classification report with SVM - Chi-squared test - KNN imputation

4.1.2 Decision Trees

Best hyperparameters Criterion : entropy , max_depth : 9, min_samples_leaf : 6, min_samples_split : 2

Remarks

Table 4.3 shows the overall performance of the decision trees classifier. Fig. 4.2 shows the confusion matrix and Table 4.4 shows the classification report.

Model Name	Decision Trees
Best training accuracy	0.8266900081
Test set accuracy	0.8228325256
Test precision	0.8234860765
Test recall	0.8228325256
Test fscore	0.8202262989
Independent accuracy	0.4692982456
Independent precision	0.7866287912
Independent recall	0.4692982456
Independent fscore	0.5555049369

Table 4.3: Overall evaluation with decision trees - Chi-squared test - KNN imputation

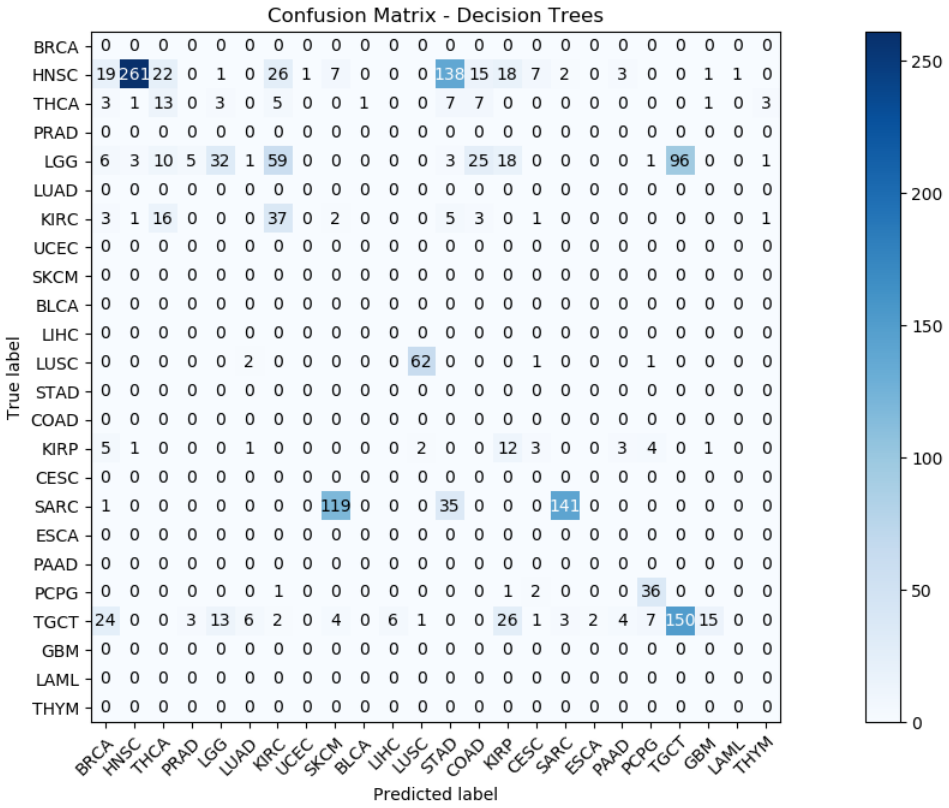


Figure 4.2: Confusion matrix with decision trees - Chi-squared test - KNN imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.98	0.50	0.66	522
CESC	0.21	0.30	0.25	44
COAD	0.00	0.00	0.00	0
ESCA	0.65	0.12	0.21	260
GBM	0.00	0.00	0.00	0
HNSC	0.28	0.54	0.37	69
KIRC	0.00	0.00	0.00	0
KIRP	0.00	0.00	0.00	0
LAML	0.00	0.00	0.00	0
LGG	0.00	0.00	0.00	0
LIHC	0.95	0.94	0.95	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.16	0.38	0.22	32
PCPG	0.00	0.00	0.00	0
PRAD	0.97	0.48	0.64	296
READ	0.00	0.00	0.00	0
SARC	0.00	0.00	0.00	0
SKCM	0.73	0.90	0.81	40
STAD	0.61	0.56	0.58	267
TGCT	0.00	0.00	0.00	0
THCA	0.00	0.00	0.00	0
UCEC	0.00	0.00	0.00	0
avg / total	0.79	0.47	0.55	1596

Table 4.4: Classification report with decision trees - Chi-squared test - KNN imputation

4.1.3 Random Forest Classifiers

Best hyperparameters Criterion : entropy , max_depth : 9, min_samples_leaf : 9, min_samples_split : 2

Remarks

Table 4.5 shows the overall performance of the random forest classifier.Fig. 4.3 shows the confusion matrix and Table 4.6 shows the classification report.

Model Name	**Random Forest**
Best training accuracy	0.8988688392
Test set accuracy score	0.8933764136
Test precision	0.8963491804
Test recall	0.8933764136
Test fscore	0.8814453177
Independent accuracy	0.6597744361
Independent precision	0.7449519328
Independent recall	0.6597744361
Independent fscore	0.6749811776

Table 4.5: Overall evaluation with random forest - Chi-squared test - KNN imputation

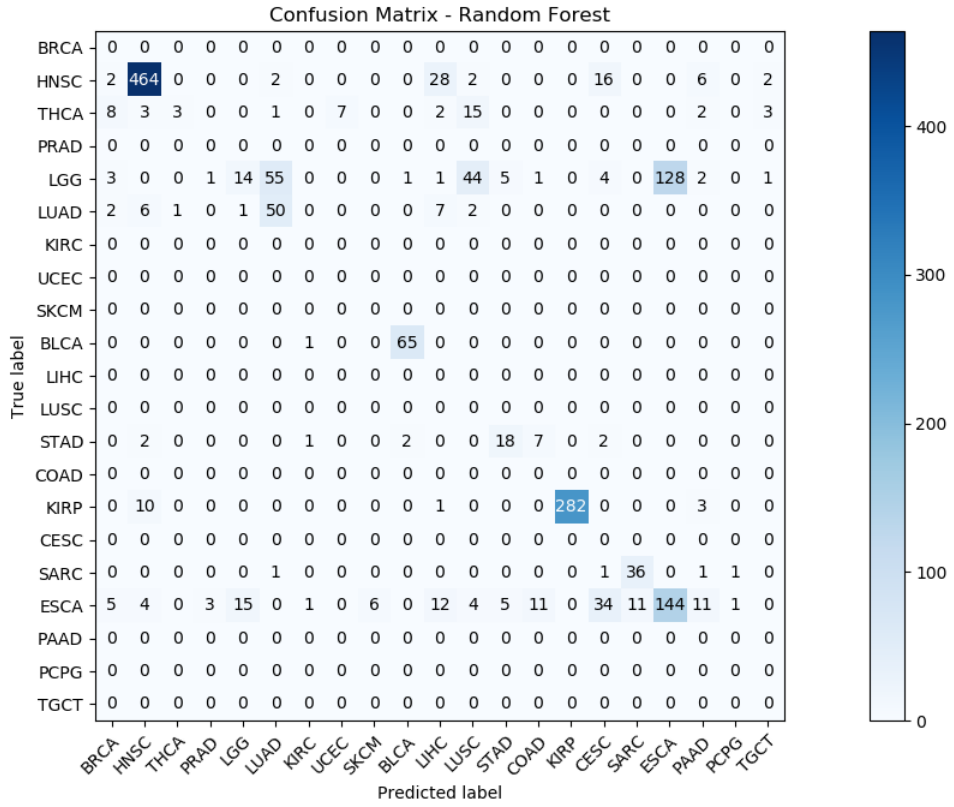


Figure 4.3: Confusion matrix with random forest - Chi-squared test - KNN imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.95	0.89	0.92	522
CESC	0.75	0.07	0.12	44
COAD	0.00	0.00	0.00	0
ESCA	0.47	0.05	0.10	260
HNSC	0.46	0.72	0.56	69
KIRC	0.00	0.00	0.00	0
LAML	0.00	0.00	0.00	0
LGG	0.00	0.00	0.00	0
LIHC	0.96	0.98	0.97	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.64	0.56	0.60	32
PCPG	0.00	0.00	0.00	0
PRAD	1.00	0.95	0.98	296
SARC	0.00	0.00	0.00	0
SKCM	0.77	0.90	0.83	40
STAD	0.53	0.54	0.53	267
TGCT	0.00	0.00	0.00	0
THYM	0.00	0.00	0.00	0
UCEC	0.00	0.00	0.00	0
avg / total	0.77	0.67	0.69	1596

Table 4.6: Classification report with random forest - Chi-squared test - KNN imputation

4.1.4 K-Nearest Neighbor Classifier

Best hyperparameters n_neighbors : 4 , weights : distance

Remarks

Table 4.7 shows the overall performance of the k-nearest neighbor classifier.Fig. 4.4 shows the confusion matrix and Table 4.8 shows the classification report.

Model Name	**KNN Classifier**
Best training accuracy	0.9060059251
Test set accuracy score	0.901453958
Test precision	0.9006212563
Test recall	0.901453958
Test fscore	0.8978247206
Independent accuracy	0.7531328321
Independent precision	0.7862158433
Independent recall	0.7531328321
Independent fscore	0.7613929591

Table 4.7: Overall evaluation with KNN classifier - Chi-squared test - KNN imputation

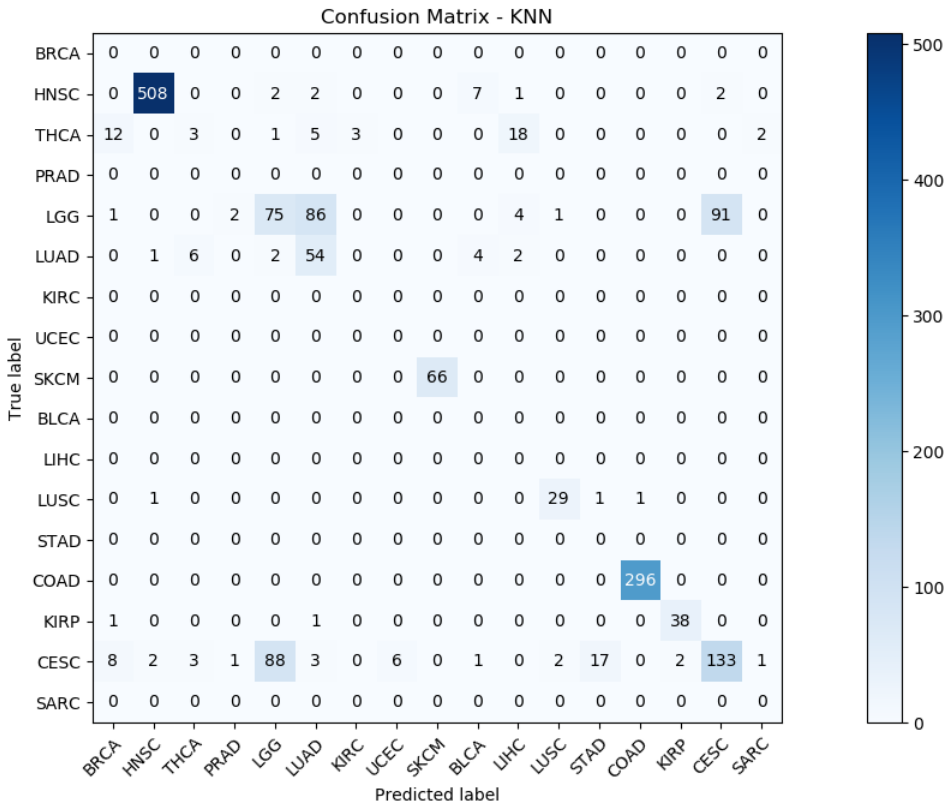


Figure 4.4: Confusion matrix with KNN classifier - Chi-squared test - KNN imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.99	0.97	0.98	522
CESC	0.25	0.07	0.11	44
COAD	0.00	0.00	0.00	0
ESCA	0.45	0.29	0.35	260
HNSC	0.36	0.78	0.49	69
LAML	0.00	0.00	0.00	0
LGG	0.00	0.00	0.00	0
LIHC	1.00	1.00	1.00	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.91	0.91	0.91	32
PCPG	0.00	0.00	0.00	0
PRAD	1.00	1.00	1.00	296
SKCM	0.95	0.95	0.95	40
STAD	0.59	0.50	0.54	267
UCEC	0.00	0.00	0.00	0
avg / total	0.79	0.75	0.76	1596

Table 4.8: Classification report with KNN classifier - Chi-squared test - KNN imputation

4.2 Feature selection using PCA and missing value imputation using mean

In these experiments, PCA is performed as discussed in Section 3.3.1.1 to get PCs of TCGA and GEO dataset. Missing values of GEO dataset are imputed by mean imputation as discussed in Section 3.4.1.1. PCA dataset of TCGA is split into 80% training set and 20% cross validation set. The training set is used to train different models and cross validation set is used to tune hyperparameters. GEO dataset is used to test the model performance. Results of different models are discussed below.

4.2.1 Support Vector Machines

Best hyperparameters C:1 , kernel: linear

Remarks

Support vector machines showed the best performance.Table 4.9 shows the overall performance of the SVM model. Fig. 4.5 shows the confusion matrix and Table 4.10 shows the classification report.

Model Name	**SVM**
Best training accuracy	1
Test set accuracy	0.9493807216
Test precision	0.9502773244
Test recall	0.9493807216
Test fscore	0.9494486524
Independent accuracy	0.7675438596
Independent precision	0.8858267479
Independent recall	0.7675438596
Independent fscore	0.8090693417

Table 4.9: Overall evaluation with SVM - PCA - mean imputation

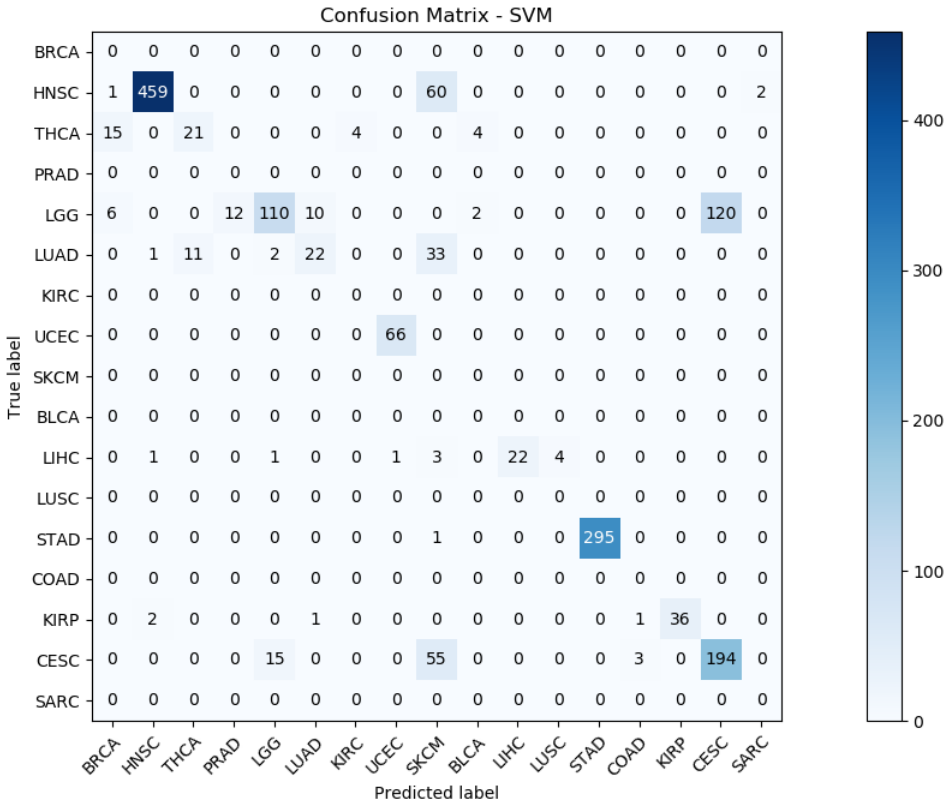


Figure 4.5: Confusion matrix with SVM - PCA - mean imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.99	0.88	0.93	522
CESC	0.66	0.48	0.55	44
COAD	0.00	0.00	0.00	0
ESCA	0.86	0.42	0.57	260
HNSC	0.67	0.32	0.43	69
LAML	0.00	0.00	0.00	0
LIHC	0.99	1.00	0.99	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	1.00	0.69	0.81	32
PCPG	0.00	0.00	0.00	0
PRAD	1.00	1.00	1.00	296
SARC	0.00	0.00	0.00	0
SKCM	1.00	0.90	0.95	40
STAD	0.62	0.73	0.67	267
UCEC	0.00	0.00	0.00	0
avg / total	0.89	0.77	0.81	1596

Table 4.10: Classification report with SVM-PCA - mean imputation

4.2.2 Random Forest Classifiers

Best hyperparameters Criterion : entropy , max_depth : 10, min_samples_leaf : 2, min_samples_split : 8

Remarks

Table 4.11 shows the overall performance of the random forest classifier. Fig. 4.6 shows the confusion matrix and Table 4.12 shows the classification report.

Model Name	Random Forest
Best training accuracy	0.9644492324
Test set accuracy	0.901453958
Test precision	0.8970338916
Test recall	0.901453958
Test fscore	0.8923179612
Independent accuracy	0.7261904762
Independent precision	0.8371031691
Independent recall	0.7261904762
Independent fscore	0.7223502708

Table 4.11: Overall evaluation with random forest with PCA - mean imputation

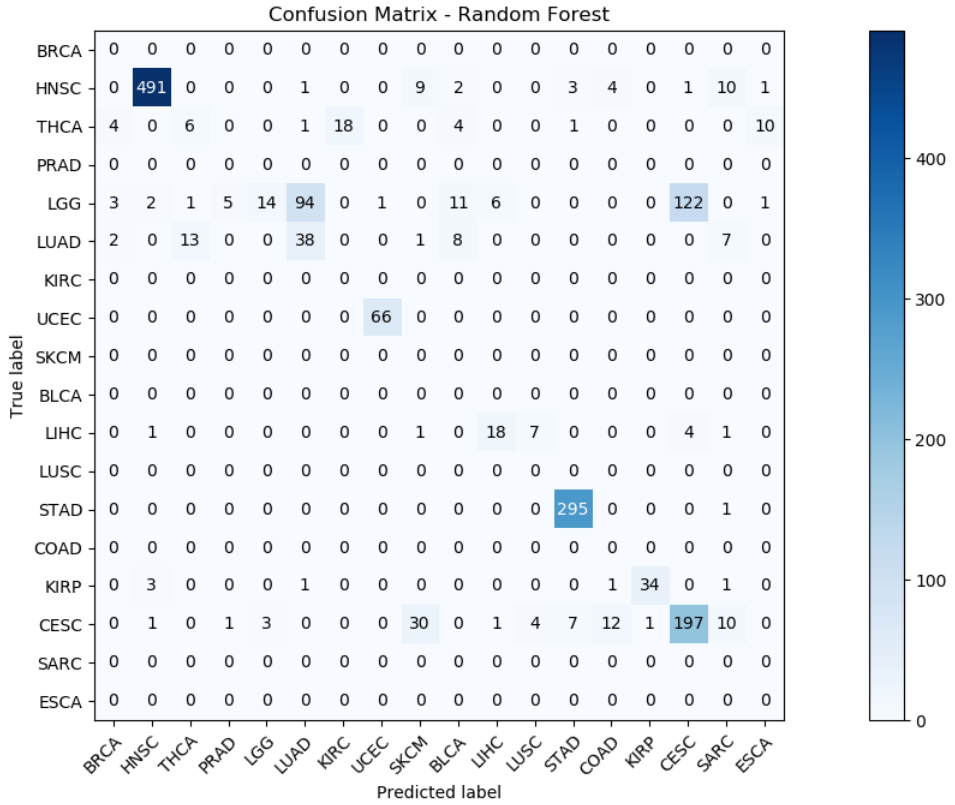


Figure 4.6: Confusion matrix with random forest - PCA - mean imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.99	0.94	0.96	522
CESC	0.30	0.14	0.19	44
COAD	0.00	0.00	0.00	0
ESCA	0.82	0.05	0.10	260
HNSC	0.28	0.55	0.37	69
LAML	0.00	0.00	0.00	0
LIHC	0.99	1.00	0.99	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.72	0.56	0.63	32
PCPG	0.00	0.00	0.00	0
PRAD	0.96	1.00	0.98	296
SARC	0.00	0.00	0.00	0
SKCM	0.97	0.85	0.91	40
STAD	0.61	0.74	0.67	267
TGCT	0.00	0.00	0.00	0
UCEC	0.00	0.00	0.00	0
avg / total	0.84	0.73	0.72	1596

Table 4.12: Classification report with random forest - PCA - mean imputation

4.2.3 Decision Trees Classifiers

Best hyperparameters Criterion : entropy , max_depth : 10, min_samples_leaf : 6, min_samples_split : 2

Remarks

Table 4.13 shows the overall performance of the decision trees model. Fig. 4.7 shows the confusion matrix and Table 4.14 shows the classification report.

Model Name	Decision Trees
Best training accuracy	0.9367088608
Test set accuracy	0.8465266559
Test precision	0.842896825
Test recall	0.8465266559
Test fscore	0.8429361117
Independent accuracy	0.6967418546
Independent precision	0.8469197546
Independent recall	0.6967418546
Independent fscore	0.7241530148

Table 4.13: Overall evaluation with decision trees - PCA - mean imputation

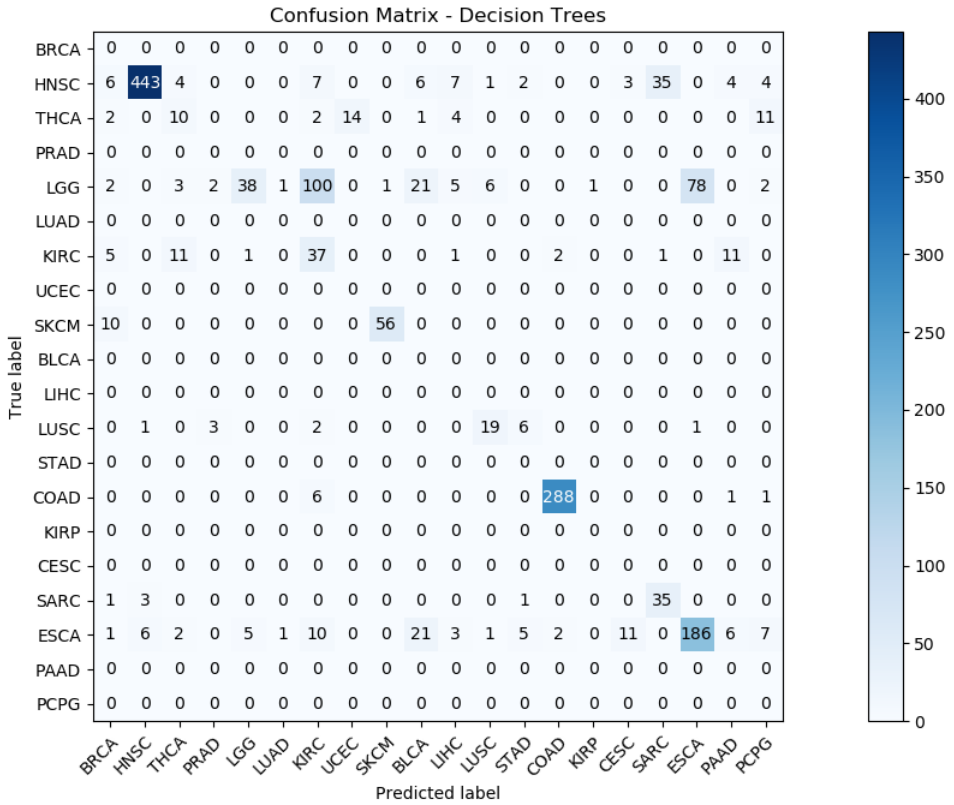


Figure 4.7: Confusion Matrix with Decision trees - PCA - mean imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.98	0.85	0.91	522
CESC	0.33	0.23	0.27	44
COAD	0.00	0.00	0.00	0
ESCA	0.86	0.15	0.25	260
GBM	0.00	0.00	0.00	0
HNSC	0.23	0.54	0.32	69
LAML	0.00	0.00	0.00	0
LIHC	0.98	0.85	0.91	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.70	0.59	0.64	32
PCPG	0.00	0.00	0.00	0
PRAD	0.99	0.97	0.98	296
READ	0.00	0.00	0.00	0
SARC	0.00	0.00	0.00	0
SKCM	0.49	0.88	0.63	40
STAD	0.70	0.70	0.70	267
TGCT	0.00	0.00	0.00	0
UCEC	0.00	0.00	0.00	0
avg / total	0.85	0.70	0.72	1596

Table 4.14: Classification report with decision trees -PCA - mean imputation

4.3 Feature selection using PCA and missing value imputation using KNN

In these experiments, PCA is performed as discussed in Section 3.3.1.1 to get PCs of TCGA and GEO datasets. Missing values of GEO dataset are imputed by knn imputation as discussed in Section 3.4.1.2. PCA dataset of TCGA is split into 80% training set and 20% cross validation set. The training set is used to train different models and cross validation set is used to tune hyperparameters. GEO dataset is used to test the model performance. Results of different models are discussed below.

4.3.1 Support Vector Machines

Best hyperparameters C:1 , kernel: linear

Remarks

Support vector machines showed the best performance.Table 4.15 shows the overall performance of the SVM model. Fig. 4.8 shows the confusion matrix and Table 4.16 shows the classification report.

Model Name	**SVM**
Best training accuracy	1
Test set accuracy	0.9531502423
Test precision	0.9536514752
Test recall	0.9531502423
Test fscore	0.9529836548
Independent accuracy	0.7694235589
Independent precision	0.8827951733
Independent recall	0.7694235589
Independent fscore	0.8102281632

Table 4.15: Overall evaluation with SVM - PCA - KNN imputation

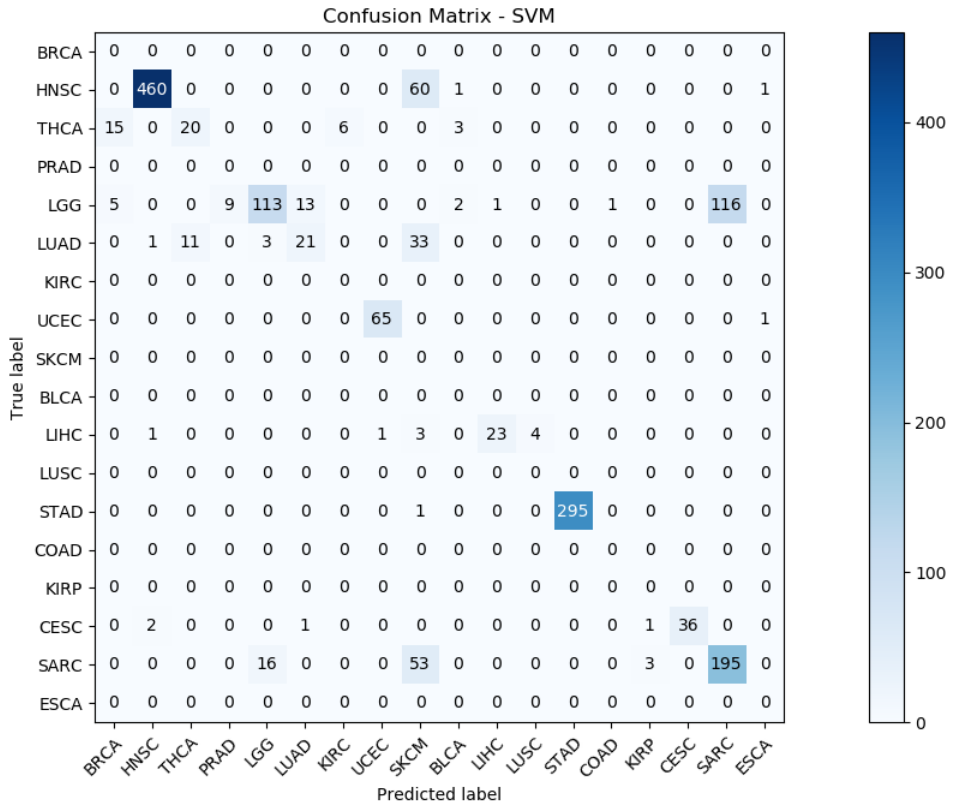


Figure 4.8: Confusion matrix with SVM - PCA - KNN imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.99	0.88	0.93	522
CESC	0.65	0.45	0.53	44
COAD	0.00	0.00	0.00	0
ESCA	0.86	0.43	0.58	260
HNSC	0.60	0.30	0.40	69
LAML	0.00	0.00	0.00	0
LIHC	0.98	0.98	0.98	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.96	0.72	0.82	32
PCPG	0.00	0.00	0.00	0
PRAD	1.00	1.00	1.00	296
READ	0.00	0.00	0.00	0
SARC	0.00	0.00	0.00	0
SKCM	1.00	0.90	0.95	40
STAD	0.63	0.73	0.67	267
UCEC	0.00	0.00	0.00	0
avg / total	0.88	0.77	0.81	1596

Table 4.16: Classification report with SVM- PCA - KNN imputation

4.3.2 Random Forest Classifiers

Best hyperparameters Criterion : entropy , max_depth : 10, min_samples_leaf : 2, min_samples_split : 8

Remarks

Table 4.17 shows the overall performance of the random forest classifier. Fig. 4.9 shows the confusion matrix and Table 4.18 shows the classification report.

Model Name	Random Forest
Best training accuracy	0.9660651764
Test set accuracy	0.8998384491
Test precision	0.8918644445
Test recall	0.8998384491
Test fscore	0.8894310016
Independent accuracy	0.6766917293
Independent precision	0.7541433239
Independent recall	0.6766917293
Independent fscore	0.6902700727

Table 4.17: Overall evaluation with random forest- PCA - KNN imputation

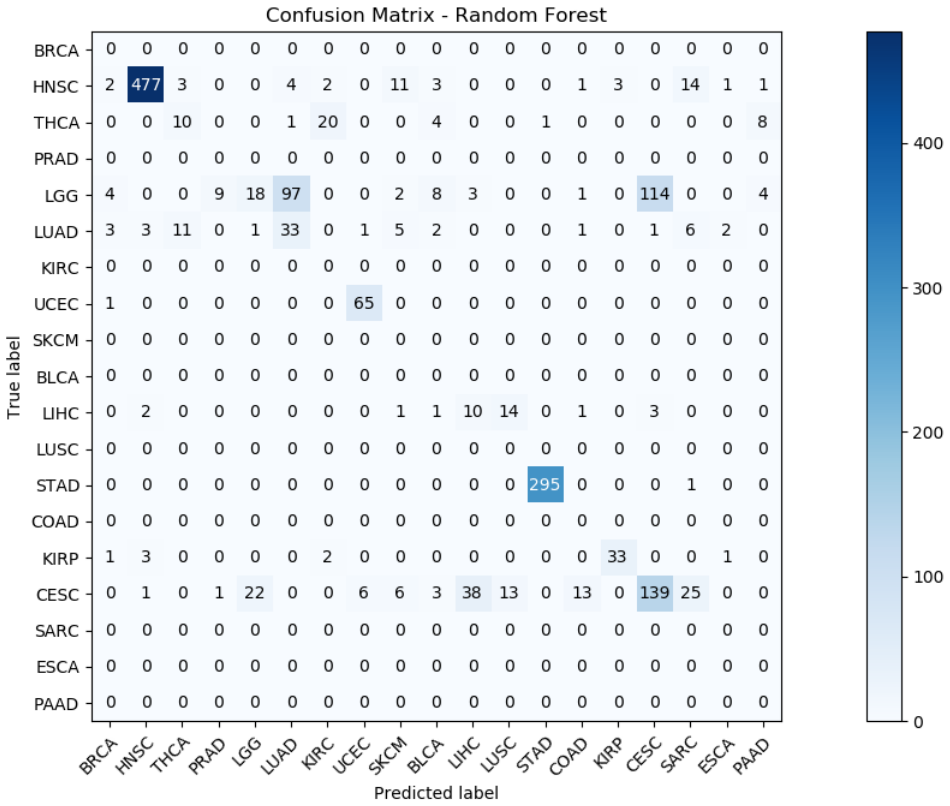


Figure 4.9: Confusion matrix with random forest- PCA - KNN imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.98	0.91	0.95	522
CESC	0.42	0.23	0.29	44
COAD	0.00	0.00	0.00	0
ESCA	0.44	0.07	0.12	260
HNSC	0.24	0.48	0.32	69
LAML	0.00	0.00	0.00	0
LIHC	0.90	0.98	0.94	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.20	0.31	0.24	32
PCPG	0.00	0.00	0.00	0
PRAD	1.00	1.00	1.00	296
SARC	0.00	0.00	0.00	0
SKCM	0.92	0.82	0.87	40
STAD	0.54	0.52	0.53	267
TGCT	0.00	0.00	0.00	0
THYM	0.00	0.00	0.00	0
UCEC	0.00	0.00	0.00	0
avg / total	0.75	0.68	0.69	1596

Table 4.18: Classification report with random forest-PCA - KNN imputation

4.3.3 Decision Trees Classifiers

Best hyperparameters Criterion : entropy , max_depth : 10, min_samples_leaf : 6, min_samples_split : 2

Remarks

Table 4.19 shows the overall performance of the decision trees classifier. Fig. 4.10 shows the confusion matrix and Table 4.20 shows the classification report.

Model Name	Decision Trees
Best training accuracy	0.9660651764
Test set accuracy	0.8998384491
Test precision	0.8918644445
Test recall	0.8998384491
Test fscore	0.8894310016
Independent accuracy	0.6766917293
Independent precision	0.7541433239
Independent recall	0.6766917293
Independent fscore	0.6902700727

Table 4.19: Overall evaluation with decision trees- PCA - KNN imputation

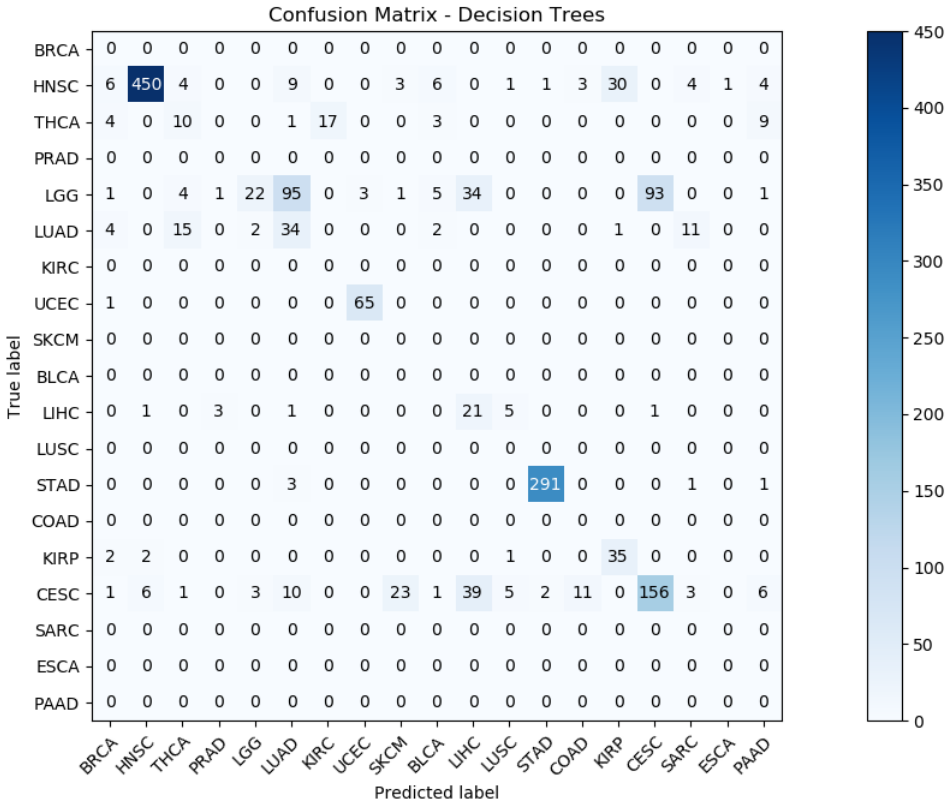


Figure 4.10: Confusion matrix with decision trees- PCA - KNN imputation

Label	precision	recall	f1-score	support
BLCA	0.00	0.00	0.00	0
BRCA	0.98	0.86	0.92	522
CESC	0.29	0.23	0.26	44
COAD	0.00	0.00	0.00	0
ESCA	0.81	0.08	0.15	260
HNSC	0.22	0.49	0.31	69
LAML	0.00	0.00	0.00	0
LIHC	0.96	0.98	0.97	66
LUAD	0.00	0.00	0.00	0
LUSC	0.00	0.00	0.00	0
PAAD	0.22	0.66	0.33	32
PCPG	0.00	0.00	0.00	0
PRAD	0.99	0.98	0.99	296
SARC	0.00	0.00	0.00	0
SKCM	0.53	0.88	0.66	40
STAD	0.62	0.58	0.60	267
TGCT	0.00	0.00	0.00	0
THCA	0.00	0.00	0.00	0
UCEC	0.00	0.00	0.00	0
avg / total	0.82	0.68	0.69	1596

Table 4.20: Classification report with decision trees-PCA - KNN imputation

4.4 Results Summary

Tables 4.21, 4.22 and 4.23 shows the results of all the models. The best accuracy of 86% on the independent dataset is achieved by using SVM on the dataset obtained by using Chi-Squared test feature selection method and KNN imputation method. PCA dataset was over fitting the models giving 100% training accuracy, on the SVM. Decision trees, random forest classifier and k-nearest neighbor classifiers where not performing well neither on PCA feature extraction method nor the Chi-squared test feature extraction method.

Model Name	SVM	Decision Trees	Random Forest	KNN
Best training accuracy	0.9476	0.8267	0.8989	0.9060
Test set accuracy	0.9445	0.8228	0.8934	0.9015
Test precision	0.9440	0.8235	0.8964	0.9006
Test recall	0.9445	0.8228	0.8934	0.9015
Test fscore	0.9440	0.8202	0.8814	0.8978
Independent accuracy	**0.8615**	0.4693	0.6598	0.7531
Independent precision	0.8908	0.7866	0.7449	0.7862
Independent recall	0.8615	0.4693	0.6598	0.7531
Independent fscore	0.8737	0.5555	0.6750	0.7614

Table 4.21: Evaluation metrics - Chi-squared test - KNN Imputation

Model Name	SVM	Decision Trees	Random Forest
Best training accuracy	1	0.9367088608	0.9644492324
Test set accuracy score	0.9493807216	0.8465266559	0.901453958
Test precision	0.9502773244	0.842896825	0.8970338916
Test recall	0.9493807216	0.8465266559	0.901453958
Test fscore	0.9494486524	0.8429361117	0.8923179612
Independent accuracy	0.7675438596	0.6967418546	0.7261904762
Independent precision	0.8858267479	0.8469197546	0.8371031691
Independent recall	0.7675438596	0.6967418546	0.7261904762
Independent fscore	0.8090693417	0.7241530148	0.7223502708

Table 4.22: Evaluation metrics - PCA-Mean imputation

Model Name	SVM	Decision Trees	Random Forest
Best training accuracy	1	0.9660651764	0.9660651764
Test set accuracy	0.9531502423	0.8998384491	0.8998384491
Test precision	0.9536514752	0.8918644445	0.8918644445
Test recall	0.9531502423	0.8998384491	0.8998384491
Test fscore	0.9529836548	0.8894310016	0.8894310016
Independent accuracy	0.7694235589	0.6766917293	0.6766917293
Independent precision	0.8827951733	0.7541433239	0.7541433239
Independent recall	0.7694235589	0.6766917293	0.6766917293
Independent fscore	0.8102281632	0.6902700727	0.6902700727

Table 4.23: Evaluation metrics - PCA - KNN imputation

Chapter 5

Conclusion and Future Work

In this thesis, we developed a model to predict the tissue origin of cancer by applying machine learning models on DNA methylation expression profiles using Chi-squared test and PCA as feature selection methods. The training data was collected from TCGA and the independent test dataset was collected from GEO. DNA methylation expression profiles in both of the labs were generated using 450k methylation arrays. We performed pre-processing to clean the data and applied various feature section methods to reduce the dimensionality of the genome data. Different models where trained on the training data and performance was evaluated on independent data.

The pre-processed data was used to train different machine learning models like support vector machines, random forest classifiers, decision trees classifier and k-nearest neighbor classifiers. Support vector machines applied on features selected using Chi-squared tests gave best training accuracy of 94.7% and test accuracy of 94.45%. It also gave an accuracy of 86% on the independent dataset. In order to increase the classification accuracy from 86% we can in future try creating a deep neural networks which can be trained to find patterns in the huge genome data, which can classify the sample even better.

Different patterns were identified in this research, which could help to classify the tumor samples according to tissue site. Cancer tumor samples could be collected and the DNA methylation profiles could be fed into the model to identify to which tissue it might belong. This can help doctors with easy and quick diagnosis of the tissue origin of cancer, which can help in better and specific treatment as required. For future work, we could also apply more advanced machine learning algorithms and test the models on different independent datasets.

Bibliography

[BDH+97] Leonardo Bottaci, Philip J Drew, John E Hartley, Matthew B Hadfield, Ridzuan Faro uk, Peter WR Lee, Iain MC Macintyre, Graeme S Duthie, and John RT Monson. Artificial neural networks applied to outcome prediction for colorectal cancer patients in separate institutions. *Cancer Informatics*, 350:469–472, 1997.

[CD04] Hong Chai and Carlotta Domeniconi. An Evaluation of Gene Selection Methods for Multi-class Microarray Data Classification. *Proceedings of the Second European Workshop on Data Mining and Text Mining in Bioinformatics*, 2004.

[CW06] Joseph A. Cruz and David S. Wishart. Applications of Machine Learning in Cancer Prediction and Prognosis. *Cancer Informatics*, 2, 2006.

[Dab18] Caner Dabakoglu. https://medium.com/@cdabakoglu/what-is-support-vector-machine-svm-fd0e9e39514f. 2018.

[DHR09] Vincent Theodore DeVita, Samuel Hellman, and Steven A Rosenberg. DeVita, Hellman, and Rosenberg's Cancer Principles Practice of Oncology Review. *2nd Edition*, 2009.

[FCD+00] Terrence S. Furey, Nello Cristianini, Nigel Duffy, David W. Bednarski, Michl Schummer, and David Haussler. Support vector machine classification and validation of cancer tissue samples using microarray expression data. *Bioinformatics*, 16(10):906–914, 2000.

[Gra17] Vincent Granville. https://www.datasciencecentral.com/profiles/blogs/types-of-machine-learning-algorithms-in-one-picture. 2017.

[GS12] Narinder Kaur Gosall and Gurpal Singh. Doctor's Guide to Critical Appraisal (3. ed.). Knutsford: PasTest. ISBN 9781905635818. page 129130, 2012.

[HTW+18] Zilong Hu, Jinshan Tang, Ziming Wang, Kai Zhang, Ling Zhang, and Qingling Sun. Deep learning for image-based cancer detection and diagnosisA survey. *Pattern Recognition*, 83:134–149, 2018.

[Ins15] National Cancer Institute. https://www.cancer.gov/about-cancer/understanding/what-is-cancer. 2015.

[JLR11] Bilian Jin, Yajun Li, and Keith D. Robertson. DNA Methylation Superior or Subordinate in the Epigenetic Hierarchy? *Genes Cancer*, 2(6):607617, 2011.

[KEE+15] Konstantina Kourou, Themis P. Exarchos, Konstantinos P. Exarchos, Michalis V. Karamouzis, and Dimitrios I. Fotiadis. Machine learning applications in cancer prognosis and prediction. *Computational and Structural Biotechnology Journal*, 13:8–17, 2015.

[KLC+17] Shuli Kang, Qingjiao Li, Quan Chen, Yonggang Zhou, Stacy Park, Gina Lee, Brandon Grimes, Kostyantyn Krysan, Min Yu, Wei Wang, Frank Alber, Fengzhu Sun, Steven M. Dubinett, Wenyuan Li, and Xianghong Jasmine Zhou. CancerLocator: non-invasive cancer diagnosis and tissue-of-origin prediction using methylation profiles of cell-free DNA. *Genome Biology*, 18:53, 2017.

[Koe17] Will Koehrsen. https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d. 2017.

[Le18a] James Le. https://jameskle.com/writes/tour-10-machine-learning-algorithms. 2018.

[Le18b] James Le. https://www.datacamp.com/community/tutorials/decision-trees-R. 2018.

[LLK+18] Wenyuan Li, Qingjiao Li, Shuli Kang, Mary Same, Yonggang Zhou, Carol Sun, Chun-Chi Liu, Lea Matsuoka, Linda Sher, Wing Hung Wong, Frank Alber, and Xianghong-Jasmine Zhou. CancerDetector: ultrasensitive and non-invasive cancer detection at the resolution of individual reads using cell-free DNA methylation sequencing data. *Nucleic Acids Research*, 46:e89, 2018.

[LSB10] Daniel Lieber, Kerry Samerotte, and Brian Beliveau. http://sitn.hms.harvard.edu/wp-content/uploads/2010/09/Epigenetics-Part-1.pdf? 2010.

[mect18] The American Cancer Society medical and editorial content team. https://www.cancer.org/cancer/cancer-unknown-primary/about/cancer-of-unknown-primary.html. 2018.

[Mur12a] Kevin P. Murphy. Machine Learning : A Probabilistic Perspective, MIT Press, 2012. . pages 544–563, 2012.

[Mur12b] Kevin P. Murphy. Machine Learning : A Probabilistic Perspective, MIT Press, 2012. . pages 596–597, 2012.

[Mur12c] Kevin P. Murphy. Machine Learning : A Probabilistic Perspective, MIT Press, 2012. . pages 388–392, 2012.

[(NC] National Cancer Institute (NCI). https://gdc.cancer.gov/resources-tcga-users/tcga-code-tables/tcga-study-abbreviations.

[PdMV+16] Martyn Plummer, Catherine de Martel, Jerome Vignat, Jacques Ferlay, Freddie Bray, and Silvia Franceschi. Global burden of cancers attributable to infections in 2012: a synthetic analysis. *The Lancet Global Health*, 13:P607–615, 2016.

[RTR+02] Sridhar Ramaswamy, Pablo Tamayo, Ryan Rifkin, Sayan Mukherjee, Chen-Hsiang Yeang, Michael Angelo, Christine Ladd, Michael Reich, Eva Latulippe, Jill P. Mesirov, Tomaso Poggio, William Gerald, Massimo Loda, Eric S. Lander, and Todd R. Golub. Multiclass cancer diagnosis using tumor gene expression signatures. *Proceedings of the National Academy of Sciences*, 98(26):15149–15154, 2002.

[Sam00] Arthur L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 44:206–226, 2000.

[Tuc18] Ian Tucker. https://www.theguardian.com/technology/2018/jun/10/artificial-intelligence-cancer-detectors-the-five. 2018.

[TWY+18] Wei Tang, Shixiang Wan, Zhen Yang, Andrew E. Teschendorff, and Quan Zou. Tumor origin detection with tissue-specific miRNA and DNA methylation markers. *Bioinformatics*, 34:398406, 2018.

[WLD01] Shahjehan A. Wajed, Peter W. Laird, and Tom R. DeMeester. DNA methylation: an alternative pathway to cancer. *Annals of surgery*, 234:10–20, 2001.

[ZZH+17] Lianghong Zheng, Jie Zhu, Rui Hou, William Shi, Jiayi Hou, Danni Lin, Gen Li, Rui-Hua Xu, Xiaoke Hao, Heng Zhang, Edward Zhang, Shaohua Yi, Jian-Kang Zhu, Michal Krawczyk, Christopher Chung, Xin Fu, Charlotte Zhang, Bennett A. Caughey, Wenqiu Wang, Juan Wang, Kang Zhang, Ken Flagg, Debanjan Dhar, Michael Karin, Liang Zhao, Maryam Jafari, Huiyan Luo, and Wei Wei. DNA methylation markers for diagnosis and prognosis of common cancers. *Proceedings of the National Academy of Sciences*, 28:7414–7419, 2017.

Curriculum Vitae

Graduate College
University of Nevada, Las Vegas

Sravani Gannavarapu Surya Naga
sravani.ganna@gmail.com

Degrees:

Bachelor of Technology,Computer Science 2014
University of Nevada Las Vegas

Thesis Title: Machine learning classification of primary tissue origin of cancer from DNA methylation markers

Thesis Examination Committee:

Chairperson, Dr. Fatma Nasoz, Ph.D.
Committee Member, Dr. Kazem Taghva, Ph.D.
Committee Member, Dr. Ajoy Datta, Ph.D.
Graduate Faculty Representative, Dr. Mira Han, Ph.D.