Finding Meaningful Biomarkers in Prostate Cancer

Hemni Sri Rajeswari Karlapalepu, Lokesh Gupta and Manan Patel

University of Windsor

***Abstract*— Prostate cancer starts in the cells of the prostates and this is the most common type of cancer that happen in Canadian men. We will see some introduction about cancer and problems that we are trying to solve in our project. The project is focused on classifying given datasets of Prostate cancer to find meaningful information like biomarkers. To perform different operations, we are provided with two datasets that contain clinical and gene data of prostate cancer. Datasets are merged and Feature selection is done using Tree-based feature selection method by which we extracted n-number of features from 60484 features. We tried to visualize the data by doing dimensionality reduction using PCA to 3 dimensions. We performed Undersampling/Oversampling/Combination of Under and Oversampling as the classes were imbalanced. Finally, to obtain useful information from this dataset, different classifications methods such as K-NN, Support Vector Machine (SVM) and Random Forest were used. After classification, each classifier's performance was evaluated based on the accuracy score.**

I. INTRODUCTION

A. Prostate Cancer

Prostate cancer is the development of cancer in the prostate, a gland in the male reproductive system. Most prostate cancers are slow growing; however, some grow relatively quickly. The cancer cells may spread from the prostate to other areas of the body, particularly the bones and lymph nodes. It may initially cause no symptoms.In later stages, it can lead to difficulty urinating, blood in the urine or pain in the pelvis, back, or when urinating.A disease known as benign prostatic hyperplasia may produce similar symptoms. Other late symptoms may include feeling tired due to low levels of red blood cells.[1]

Factors that increase the risk of prostate cancer include older age, a family history of the disease, and race. About 99% of cases occur in males over the age of 50. Having a first-degree relative with the disease increases the risk two to threefold. In the United States, it is more common in the African American population than the White American population. Other factors that may be involved include a diet high in processed meat, red meat or milk products or low in certain vegetables. An association with gonorrhea has been found, but a reason for this relationship has not been identified. An increased risk is associated with the BRCA mutations. Prostate cancer is diagnosed by biopsy. Medical imaging may then be done to determine if the cancer has spread to other parts of the body.[1]

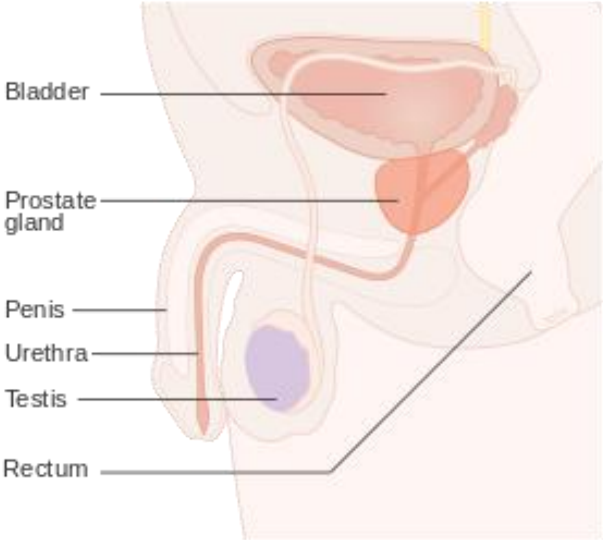

Fig. 1. Position of the prostate.

II. PROJECT OVERVIEW

There are two dataset files. One dataset contains the gene data of 60,484 genes of 499 patients. The other dataset provides the clinical data of each patient. We have performed machine learning techniques such as feature selection and classification in order to get meaningful biomarkers using Scikit (python 3.7).

The dataset is obtained from the Genomic Data Commons (formerly cBioPortal). Clinical data contains details about a particular patient like age of the patient, tumour level, MRI results, Vital status, tumour status, Primary site, Gleason score etc. Then there is genes data set which contains details about genes. Genes are divided according to the unique patient ids provided in the clinical dataset.

In this project, we will be focusing mainly on finding least number of biomarkers responsible for predicting gleason score and laterality of the prostate cancer. Firstly, we tried to find the classification accuracy without doing feature selection. Then, we did Random Forest based feature selection to find the minimum number of genes with good accuracy. To visualize the data we used Principal Component Analysis (PCA) for dimensionality reduction to 3 dimensions. Since, the data is imbalanced based on both gleason score and laterality, we tried oversampling, undersampling and combination of oversampling and undersampling. For oversampling, we

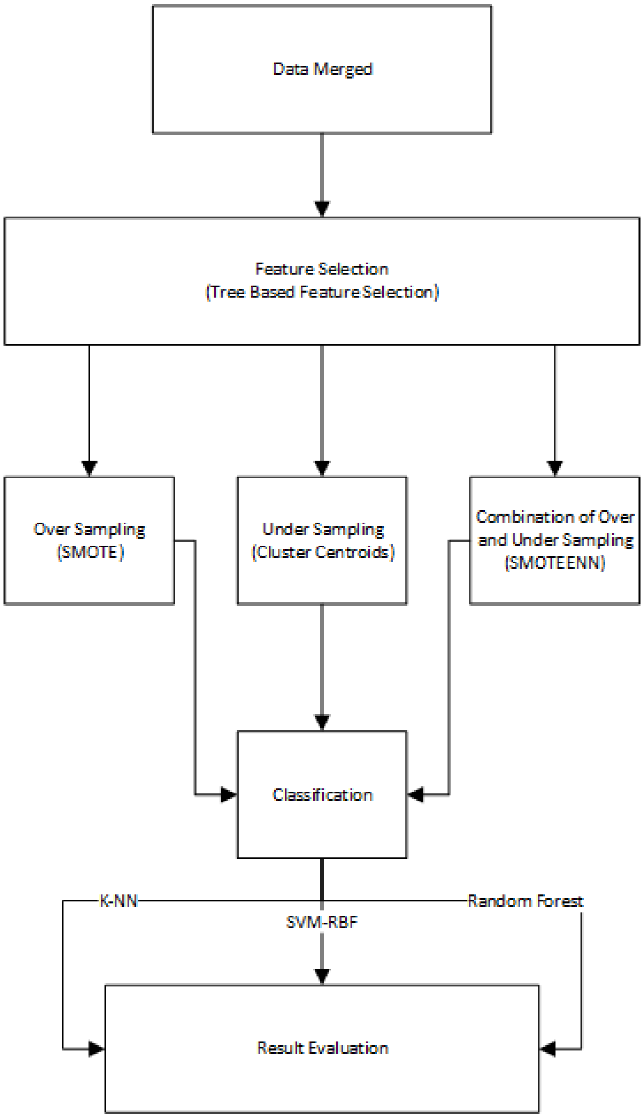

Fig. 2. Steps of Solution.

used Random Oversampling and SMOTE techniques. For undersampling, we used prototype generation technique. We also used combination of undersampling and oversampling technique SMOTEENN. Finally, we used K-NN, SVM-RBF and Random Forest for classification.

III. Proposed Solution

We used Scikit-learn libraries which are implemented in python for performing feature selection and classification. We used imblearn libraries to do undersampling and over-sampling. The detailed description of the project workflow is depicted in Figure 2.

A. Data Merging

Since the data was in two datasets, we had to merge the datasets in order to match the rows corresponding to the same patient using the attribute PATIENT_ID. After that we transposed the data so that the features(genes) are in columns. Finally we merged both the datasets using pd.merge() function.

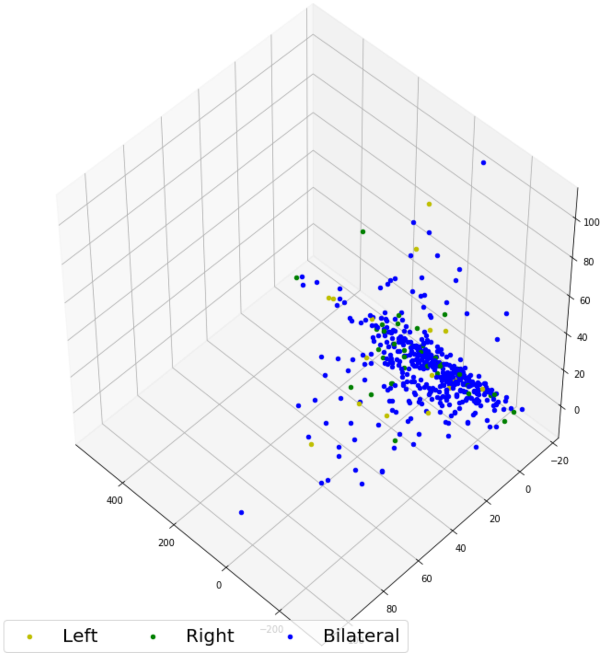

Fig. 3. 3-D Visualization of data after Feature Selection based on class label Laterality

B. Feature Selection

Since we had around 60,000 features we had to do feature selection in order to remove irrelevant and redundant features in our dataset which inturn lead to improvement in the classifier performance. In Scikit learn library, various methods for feature selection are available but we selected Random Forest based classifier. A random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement if bootstrap=True (default)[2]. Figure 3 and 4 shows the 3-D visualization of the data after Feature Selection based on class labels :

1. Laterality (Figure 3)
2. Gleason Score (Figure 4)

C. Data Sampling

Before implementing the final classification we need to decide the classes on which we should perform the classification. For this we selected ”Gleason Score” and ”Laterality” from the clinical dataset. We observed that the both the selected classes are highly unbalanced, so we used samplers which are available in the imblearn module. This is shown in Table I and Table II. We have considered 3 types of sampling techniques which are explained below:

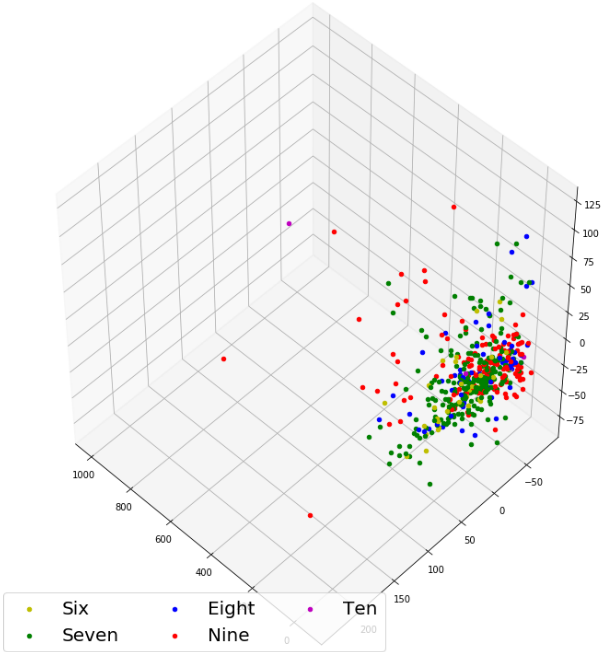

Fig. 4. 3-D Visualization of data after Feature Selection based on class label Gleason Score

TABLE I

Number of Data Samples under Class Label Laterality in original Data

Class Label	Number of Data Samples
Left	18
Right	38
Bilateral	430

1) Over Sampling: One way to fight the issue of classes being unbalanced is to over sample the data. For over sampling the data, we have used the SMOTE algorithm. Synthetic Minority Over-sampling Technique(SMOTE) is the most common technique for over sampling[3]. To illustrate the working of this algorithm we have some training data which has ’s’ samples and ’f’ features in the feature space of the data. The features are assumed to be continuous. To over sample a feature, consider its k nearest neighbors (in feature space). A vector is taken among those K-Nearest Neighbors and the current data point. This vector is also multiplied randomly by a number lying between 0 and 1. Finally, this is added to the current dataset. We got the plots as depicted in figure 5 & 6 after performing over sampling and reducing the dataset to 3 dimensions using PCA for both class labels i.e. Laterality and Gleason Score. As evident from the figures, the class labels are balanced after the application of over sampling.

2) Under Sampling: Undersampling is also one of the techniques used for handling class imbalance. In this technique, we under sample majority class to match the minority class. So in our example, we eliminated ”Bilateral” samples from the ”LATERALITY” class to match number of samples

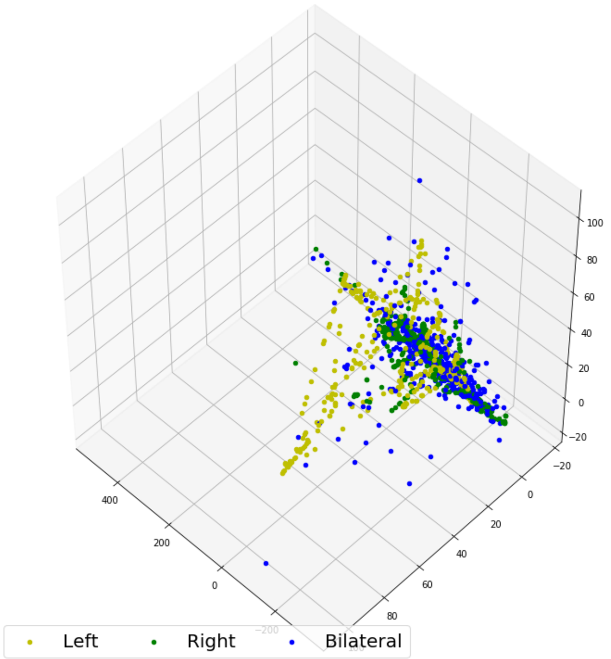

Fig. 5. Oversampling using SMOTE algorithm for class label Laterality.

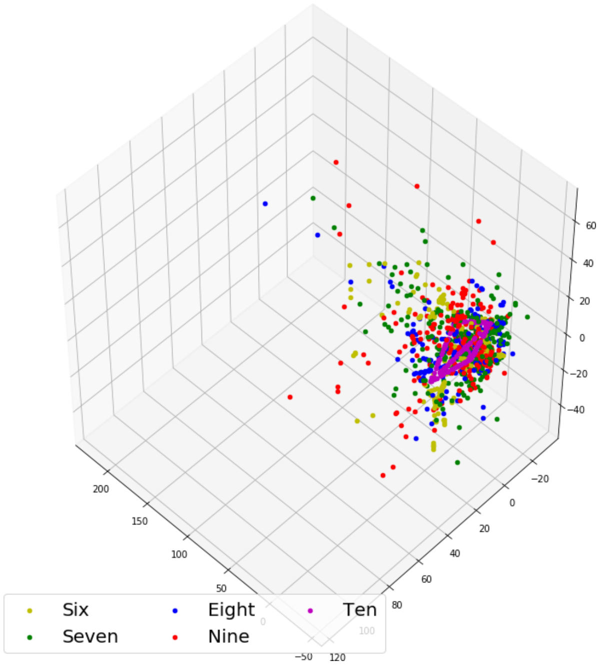

Fig. 6. Oversampling using SMOTE algorithm for class label Gleason Score.

TABLE II

Number of Data Samples under Class Label Gleason Score in original Data

Class Label	Number of Data Samples
6	44
7	246
8	63
9	137
10	4

from the other class. This makes sure that the training data has equal amount of samples from all the classes. For under sampling we have used Cluster Technique. This method under samples the majority class by replacing a cluster of majority samples by the cluster centroid of a KMeans algorithm. This algorithm keeps N majority samples by fitting the KMeans algorithm with N cluster to the majority class and using the coordinates of the N cluster centroids as the new majority samples.

3) Combination of over and under-sampling: The SMOTE algorithm as mentioned above can generate noisy samples by interpolating new points between marginal outliners and inliners. This can be solved by cleaning the space resulting from over-sampling. There two cleaning methods that are available: (i) SMOTETomek and (ii) SMOTEENN. In this project we have used SMOTEENN. This algorithm first uses SMOTE to perform over sampling and then clean the over sampled data using ENN. The ENN method[4], removes the instances of the majority class whose prediction made by KNN method is different from the majority class. So, if an instance has more neighbors of a different class, this instance will be removed. We got the plots as depicted in figure 7 & 8 after performing combination of over sampling and under-sampling and reducing the dataset to 3 dimensions using PCA for both class labels i.e. Laterality and Gleason Score. As evident from the figures, the class labels are balanced after the application.

D. Classification Methods

The features which were selected from the Randon Forest Classifier, we have used three classifiers namely K-Nearest Neighbors(KNN), Support Vector Machine (RBF Kernel) and Random Forest. A brief information about each of the classifier is given below:

1) K-Nearest Neighbors(K-NN): The k-nearest neighbors algorithm (k-NN) is a non-parametric method used for classification and regression. In both cases, the input consists of the k closest training examples in the feature space. In k-NN classification, the output is a class membership. An object is classified by a majority vote of its neighbors, with the object being assigned to the class most common among its k nearest neighbors (k is a positive integer, typically small). If k = 1, then the object is simply assigned to the class of that single nearest neighbor. The algorithm is explained in figure 9[5]. A commonly used distance metric for continuous variables is Euclidean distance. So according to the figure, the GREEN

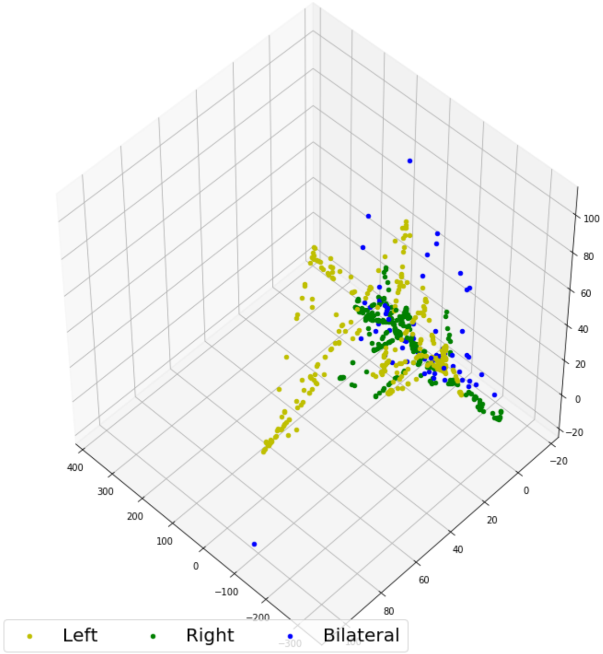

Fig. 7. Combination of oversampling and under-sampling using SMOTEENN algorithm for class label Laterality.

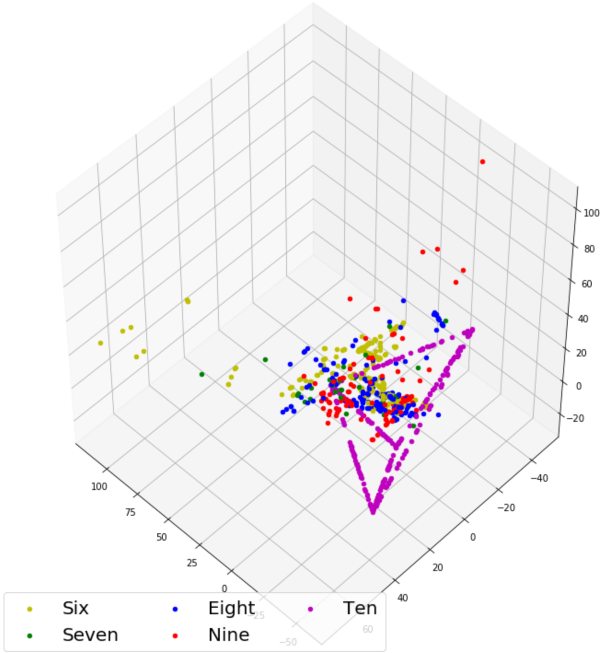

Fig. 8. Combination of oversampling and under-sampling using SMOTEENN algorithm for class label Gleason Score.

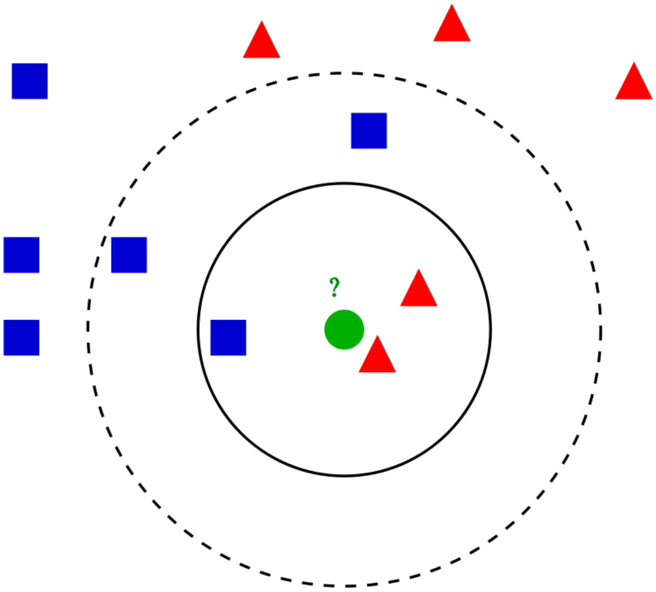

Fig. 9. Working of K-NN

circle will be classified in to RED class represented by triangle according to the majority voting.

2) Support Vector Machine(SVM): "Support Vector Machine" (SVM) is a supervised machine learning algorithm which is used for classification problems. Support Vectors are simply the co-ordinates of individual observation. A support vectors are the point in the n-dimensional space which are closest to the seperating hyper plane. The algorithm tries to find a hyper plane with the maximum distance from all the support vector from different classes. For instance in the Figure 10, we can see that among all the hyperplane which linearly separates the two classes A and B, the black hyper plane is at the maximum distance. Thus the SVM algorithm selects this as the separting hyper plane and performs the classification accordingly. Support Vector Machines are very powerful classification algorithm. When used in conjunction with random forest and other machine learning tools, they give a very different dimension to ensemble models. Hence, they become very crucial for cases where very high predictive power is required. Such algorithms are slightly harder to visualize because of the complexity in formulation[6].

3) Random Forest Classifier: Random Forest is a flexible, easy to use machine learning algorithm that produces, even without hyper-parameter tuning, a great result most of the time. It is also one of the most used algorithms, because of it's simplicity and the fact that it can be used for both classification and regression tasks. Random forest algorithm tries to build multiple decision trees and merges them together to get a more accurate and stable prediction. Random Forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model. Random Forest also helps in performing feature selection. For this Sklearn provides a great tool that measures a features importance by considering how much impurity is reduced across all the trees in the forest which are using that feature[7]. The figure 11 shows a basic functioning of a Random Forest Classifier.

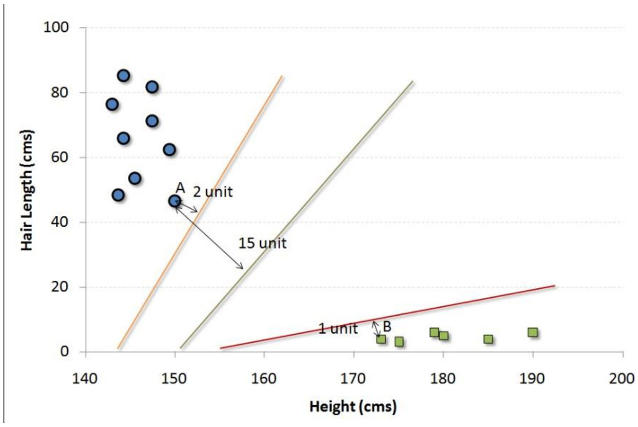

Fig. 10. Support Vector Machine

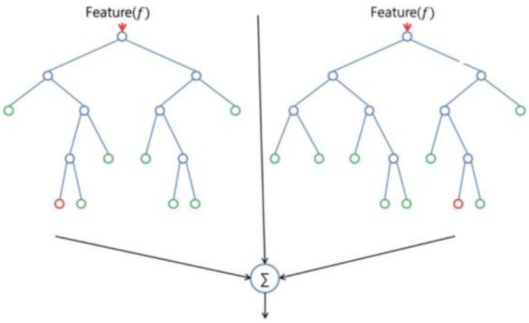

Fig. 11. Random Forest Classifier

IV. Results and observations

After finishing with the process of classification using various classifiers such as K-NN, SVM(RBF) etc., we have come up with certain results. Before we present our final results, we have firstly computed the results without performing feature selection on the two class labels "Laterality" and "Gleason Score". These results are shown in Table III and Table IV.

TABLE III

Accuracy Score without Feature Selection and Data Sampling for Class Label Laterality

Classifier	Accuracy
K-NN	87.09%
SVM-RBF	87.09%
Random Forest	86.69%

TABLE IV

ACCURACY SCORE WITHOUT FEATURE SELECTION AND DATA SAMPLING FOR CLASS LABEL GLEASON SCORE

Classifier	Accuracy
K-NN	52.67%
SVM-RBF	49.84%
Random Forest	61.89%

We can see from the Table III that all the three selected classifiers gives approximately 87% accuracy for the class label ”Laterality”. The parameters used for classifying the class label ”Laterality” is given as follows:

- K-NN: n_neighbors = 6
- SVM-RBF: c = 1.0, gamma = ’auto’
- Random Forest: n_estimators = 1000

Similarly, from the Table IV that all the three selected classifiers gives approximately 55% accuracy for the class label ”Gleason Score”. The highest was given by the Random Forest classifiers. The parameters used for classifying the class label ”Gleason Score” is given as follows:

- K-NN: n_neighbors = 11
- SVM-RBF: c = 1.0, gamma = ’auto’
- Random Forest: n_estimators = 1000

TABLE V

NUMBER OF FEATURES FOR CLASS LABELS LATERALITY AND GLEASON SCORE AFTER RANDOM FOREST FEATURE SELECTION

Class Label	Number of Features
Laterality	36
Gleason Score	68

After performing feature selection using the Random Forest Classifier with n_estimators = 1, we got 36 and 68 features for ”Laterality” and ”Gleason Score” respectively. This is depicted in Table V.

After that we did Data Oversampling using SMOTE Data OverSampling algorithm for Class Label Laterality which gave us the following data shape:

- Bilateral: 430
- Right : 430
- Left : 430

TABLE VI

ACCURACY SCORE WITH FEATURE SELECTION AND DATA OVERSAMPLING FOR CLASS LABEL LATERALITY

Classifier	Accuracy
K-NN	80.29%
SVM-RBF	84.18%
Random Forest	99.07%

Data Oversampling using SMOTE algorithm gave us very good results with only 36 features(genes) for Class label Laterality. The results are shown in Table VI. Random Forest classifier even gave us the near perfect accuracy pertaining to the fact that the 36 biomarkers that we found after feature selection are correct and are responsible for determining the Laterality of the prostate cancer. The parameters used for the different classifiers are given below:

- K-NN: n_neighbors = 1
- SVM-RBF: c = 1.0, gamma = ’auto’
- Random Forest: n_estimators = 1000

Resampled dataset shape after running SMOTE Data OverSampling algorithm for Class Label Gleason Score is given below:

- 6: 246
- 7 : 246
- 8 : 246
- 9 : 246
- 10 : 246

Feature Selection using Random Forest for class label Gleason Score gave us 68 features. We ran SMOTE algorithm for Data Oversampling and then ran the classifiers to check the accuracy score. From Table VII, it is evident that the all the three classifiers performed very well for classifying ”Gleason Score” which tells us that the features (genes) selected are responsible for determining the Gleason score of the prostate cancer. Moreover, Random Forest classifier gave us the highest accuracy(90%) among the three. The parameters used for the different classifiers are given below:

- K-NN: n_neighbors = 1
- SVM-RBF: c = 1.0, gamma = ’auto’
- Random Forest: n_estimators = 1000

After Data Oversampling we also ran Combination of Data Oversampling and Undersampling just to be sure of Data overfitting which is very common issue with just doing Data Oversampling and Undersampling. We ran SMOTEENN algorithm to do combination of Data Oversampling and Undersampling. Resampled dataset shape after running SMOTEENN algorithm for Class Label Laterality is given below
:

- Bilateral: 62
- Right : 331
- Left : 347

From Table VIII, we can clearly see that the accuracy scores for all the classifiers has increased substantially from the initial score. The greatest improvement we can observe is in K-NN classifier whose accuracy score increased from 87% to 97%. If we compare the accuracy scores with the Over Sampling technique (SMOTE), we can see SMOTEENN performs better at sampling the unbalanced classes. The parameters used for the different classifiers are given below:

- K-NN: n_neighbors = 1
- SVM-RBF: c = 1.0, gamma = ’auto’
- Random Forest: n_estimators = 1000

Resampled dataset shape after running SMOTEENN, Combination of Data OverSampling and UnderSampling algorithm for Class Label Gleason Score is given below:

- 6: 222
- 7 : 27
- 8 : 199

- 9 : 103
- 10 : 246

Table IX shows the accuracy scores for class label ”Gleason Score” with using SMOTEENN as the data sampling algorithm. Clearly we can see that K-NN outperformed among the three classifiers used with accuracy score of 99%. The accuracy score of SVM-RBF remained almost the same. Moreover, we can also see that accuracy scores obtained from the SMOTEENN data sampling technique are better than the those obtained from the SMOTE technique.

TABLE VII

ACCURACY SCORE WITH FEATURE SELECTION AND DATA OVERSAMPLING FOR CLASS LABEL GLEASON SCORE

Classifier	Accuracy
K-NN	84.18%
SVM-RBF	74.47%
Random Forest	90.04%

TABLE VIII

ACCURACY SCORE WITH FEATURE SELECTION AND COMBINATION OF DATA OVER SAMPLING AND UNDER SAMPLING FOR CLASS LABEL LATERALITY

Classifier	Accuracy
K-NN	97.38%
SVM-RBF	94.19%
Random Forest	95.35%

TABLE IX

ACCURACY SCORE WITH FEATURE SELECTION AND COMBINATION OF DATA OVER SAMPLING AND UNDER SAMPLING FOR CLASS LABEL GLEASON SCORE

Classifier	Accuracy
K-NN	99.09%
SVM-RBF	86.99%
Random Forest	94.60%

V. ROLES AND RESPONSIBILITIES

- **Hemni Sri Rajeswari:** Data visualisation using PCA, Over- sampling using SMOTE, Random Forest Classification, Documentation.
- **Lokesh Gupta:** Combination of over and under sampling using SMOTEENN, SVM-RBF classification, Data visualisation using PCA, Documentation.
- **Manan Patel:** Dataset Merging, Feature Selection, K-NN clasification, Documentation.

VI. CONCLUSION

We have found that the best 36 Gene IDs for classifying ”Laterality” class label and 68 Gene IDs for classifying ”Gleason Score” class label out of 60400 Gene IDs using the Random Tree Classifier based feature selection technique are responsible for determining the Laterality and Gleason score of the Prostate cancer respectively. We have performed Over sampling and combination of over and under sampling technique using SMOTE and SMOTEENN algorithm respectively since the selected classes were highly imbalanced. We have concluded that after feature selection the accuracy scores of all the three classifiers increased. Moreover, among the three classifiers which we selected for classification, we found that Random forest classifier outperformed after we applied SMOTE technique with an accuracy of more than 90% for both the class labels while K-NN gave the best results if we performed SMOTEENN algorithm with an accuracy score of more than 97% for both the class labels. Getting good accuracy with all three classifiers used justify the fact that the features(genes) returned by the feature selection are the primary genes responsible for the Laterality and Gleason Score of the prostate cancer and if required the biologists can look further into these features(genes). This project report is also accompanied with two excel sheets containing the list of Gene IDs with their importance scores as found using the feature selection technique used. The names of these files are ”Feature Selection Gleason Score.xlsx” and ”Feature Selection Laterality.xlsx”.

REFERENCES

[1] ”What is prostate cancer?” https://en.wikipedia.org/wiki/Prostate_cancer

[2] ”Random Forest Classifier” https://scikit-learn.org

[3] Chawla, N., Bowyer, K., Hall, L. and Kegelmeyer, W.P. (2002) SMOTE: Synthetic Minority Over-sampling Tech-nique. Journal of Artificial Intelligence Research, 16, 321-357.

[4] Wilson, D.L. (1972) Asymptotic Properties of Nearest Neighbor Rules Using Edited Data. IEEE Transactions on Sys-tems, Man, and Communications, 2, 408-421.

[5] ”K-Nearest Neighbors” https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

[6] https://www.analyticsvidhya.com/blog/2014/10/support-vector-machine-simplified/

[7] https://towardsdatascience.com/the-random-forest-algorithm-d457d499ffcd