Mathematical model approach for draft picking in basketball

Lawrence Thanakumar Rajappa
IDA Linköping University
lawra776@student.liu.se

1 INTRODUCTION

Analytics is being used in all fields such as healthcare, manufacturing, banking and etc. for decision-making. Likewise, analytics is also playing a major role in Sports industry such as football, baseball, basketball and etc. to predict player's next move, injury analysis, position analysis and etc. Sports analytics is being spoken as a concept for many years which could be used to improve team performance, as a result, the revenue generation is very much improved for the team. For this project, we will mostly focus on basketball. In order to have a good prediction and analysis report, we need to have proper dataset and most importantly, the data must also have most important attributes that could provide insights for a given problem.

There are many ways that the data can be used by the team for various purposes. The kind of data that the team would use includes average stats of the players, per game stats, and etc. These data can be used to understand a player in terms of strengths and weaknesses, emotional stability and etc. These attributes could be used to assess the team's performance. Moreover, there are other attributes such as weather conditions, the condition of the field and even psychological factors such as the fans support should be included along with player's data to determine the team's performance. This document speaks about how the data are being used by a basketball team to select players using mathematical models.

2 AIM

In this project, the aim is to create a draft picking system for a basketball team by using 3 mathematical models namely, model 1 : model to predict whether a player will stay in the team for five years or not, model 2: model to determine the position of players based on their previous experiences and model 3: model to cluster or group players based on previous performances. These models would facilitate team managers and coaches to select players and make best out of them.

3 MOTIVATION

Before the advent of analytics, the selection of players or draft formation was done manually which was a time consuming and huge workload. The emergence of analytics and computing resources has paved a new way in recruiting best players based on their previous performances in a short period of time with minimal workload. However, Sports industry has restricted for the complete adoption of analytics into their respective teams because teams spend three-fourth of their revenue for paying salaries to the players and to cover other expenditures. Hence, the teams cannot afford to invest huge sum of money in technology, data and analytical tools [4]. This project would remove the above mentioned bottleneck and facilitate the teams to use analytical tools with much lower cost and at ease.

4 EARLIER SYSTEM

Earlier to 2005, the data was collected by a person watching the game using either a notepad and pen or black boards with chalks. This data was prone to human errors. As a result, the analysis carried out on this data and results were misleading. In 2005, two Isareli scientists, Gal Oz and Miky Tamir, created a system called *SportsVU* (see in figure 1) [13] [27]. This system captures the ball movement as well as athletes movement, all these data are combined together for statistical analysis using the statistical algorithms that the company has created [22]. Based on the statistial analysis inference, the players were chosen for a team, but this method was manual. Moreover, other data such as Rebounds, TurnOver and etc were calculated from this system as well as from manually gathered data.

Figure 1: SportsVU in basketball court.

5 BACKGROUND

In this part, some general terms that are relative to the topic are going to be discussed. It is important to understand them for further reading.

The terms that will be discussed are:

- Draft picking and its process.
- Machine learning.
- Sports Analytics.

5.1 Draft picking and its process

NBA draft is an annual event where basketball teams select players from american colleges and from international professional league for their rosters. Once a team selects a player, then the team has right to sign a NBA contract with the player.

In draft picking process, teams select eligible players in turns. There are two rounds in the draft where all 30 teams participate to select a player in turns, meaning every year 60 eligible players are drafted, but teams that did not reach the playoffs in the previous regular season or teams with worst performs selects a player by undergoing a process called *NBA Draft Lottery*. This process determines the selection order of the team or provides an opportunity for the team which wins the lottery to pick the first draft followed by other worst performing teams. The team with best records receives the 30th pick. During the second round in the draft, there is no lottery system, but teams pick the draft in the reverse order based on the previous regular season's standings. Moreover, the teams can exchange their draft picks with each other, for example, in 2019 the Minnesota Timberwolves traded the No. 11 pick and forward Dario Saric to the Phoenix Suns in exchange for the No. 6 pick. But, there are some restrictions based on *The Stepien Rule*, that is, this rule prevents the team from trading their first-round draft pick in consecutive years [7].

5.2 Machine learning

Machine learning is a general concept and broader area which consists of many definitions provided by recognized and reliable universities, institutions, professors and organizations and they are as follows,

- "Machine learning is based on algorithms that can learn from data without relying on rules-based programming." [19]
- "The field of Machine Learning seeks to answer the question “How can we build computer systems that automatically improve with experience, and what are the fundamental laws that govern all learning processes?" [14]
- and etc.

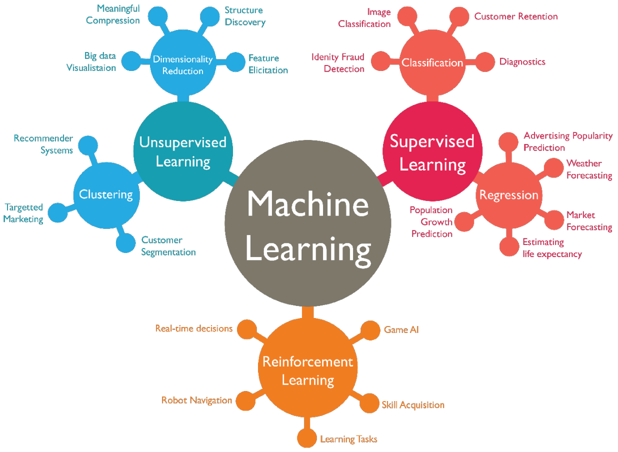


Figure 2: Machine Learning in an eagle view

Machine learning can be categorized in three types,

- Supervised Learning.
- Unsupervised Learning.
- Reinforcement Learning.

The definitions for the above terms are:

- "Supervised learning algorithms generate a function that maps inputs to desired outputs, based on a set of examples with known output (labeled examples)" [25].
- "Unsupervised learning algorithms find patterns and relationships over a given set of inputs (unlabelled examples)" [25].
- "Reinforcement learning, where an algorithm learns a policy of how to act given an observation of the world" [25].

In this project, we will mostly focus on Supervised and Unsupervised learning algorithms. The different types of algorithms in both supervised and unsupervised learning are given below,

Some algorithms of supervised learning:

- Nearest Neighbor
- Naive Bayes
- Support Vector Machine (SVM)
- Logistic Regression
- Linear Regression
- and etc.

Some algorithms of unsupervised learning:

- k-means clustering
- Association Rules [5]
- and etc.

5.3 Sports Analytics

Sports analytics is the application of above mentioned algorithms to sport in order to draw useful insights which could help an individual athlete's performance, or a team's performance for a season. It can also help teams to perform injury analysis and steps to mitigate them, salary of a player based on his previous performances and etc. Nowadays, many teams, coaches and even players are adopting sports analytics for decision making.

"The analytics split nicely between the front-office and back-office. Front-office analytics include topics like analyzing fan behavior, ranging from predictive models for season ticket renewals and regular ticket sales, to scoring tweets by fans regarding the team, athletes, coaches, and owners. This is very similar to traditional customer relationship management. Financial analysis is also a key area, especially for the pros where salary caps or scholarship limits are part of the equation. Back-office uses include analysis of both individual athletes as well as team play. For individual players, there is a focus on recruitment models and scouting analytics, analytics for strength and fitness as well as development, and predictive models for avoiding overtraining and injuries. Concussion research is a hot field. Team analytics include strategies and tactics, competitive assessments, and optimal roster choices under various onfield or on-court situations.” [24]

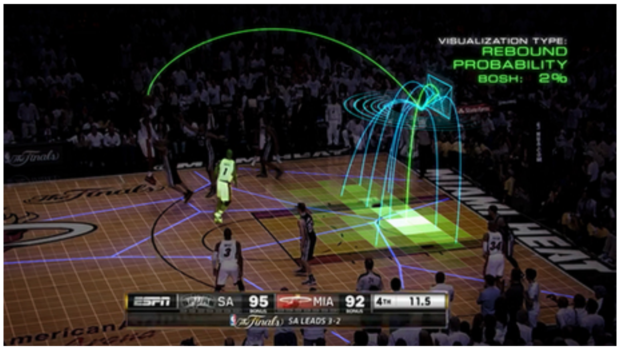


Figure 3: Sports Analytics

However, the analytical methods and data has to be kept safe and should be extremely careful because the data and methodology could lead to numerous problems such as issues with betting companies, non-ethical training of atheletes leading to injuries and etc. [15]

6 METHODOLOGY

This section describes the framework used for draft picking in basketball using Machine learning. The proposed framework consists of 3 mathematical models namely career longevity model, providing positions to selected players and grouping players based on their performance into two groups, either 1 or 0. Initially, the user or the coach or the team manager loads the players data to **model 1**, which provides list of players who will stay with the league for more than 5 years and list of players who don't stay for more than 5 years. This output is provided as an input to **model 2** which predicts the positions of the players based on previous performance. Finally, the output of 2nd model is provided as an input to the **model 3** where players are grouped as 1 and 0, that is, either good or bad pick. This proposed framework is time efficiency, environmental friendly (less paperwork) and user friendly.

Subsection **6.1** describes the libraries or functions used in the proposed framework. Subsection **6.2** descibes about the source of data, structure of data, and pre-processing of data. Subsection **6.3** describe both supervised and unsupervised learning algorithms which are used to create 3 mathematical models for draft picking. The framework applies many algorithms and chooses one algorithm which performs better than the other algorithms for each model. This section concludes with Subsection **6.4**, which describes the evaluation metrics used.

6.1 Libraries

The following Python programming packages or libraries have been used by the proposed framework for data pre-processing and model building,

- NumPy for numerical computations.
- Pandas for data loading.
- MatplotLib and Seaborn for visualization.
- sklearn for evaluation metrics and model building (Supervised and Unsupervised learning).
- Pickle for loading and saving of data objects, mathematical model objects and etc.
- statsmodel for checking multicollinearity in data.

6.2 Dataset

6.2.1 Source of data.

The two main sources for dataset used by the proposed framework are,

- basketball-reference.com
- Data.world

The data provided by Basketball-reference.com contains the NBA players overall performance such as rebounds, turnovers, points, games played and etc. for a regular season which can be either web scrapped or downloaded in comma-separated format file (CSV). Data.world provides NBA players data such as players demographic data, players salary data and etc. which are less in number. By combining data from both sources, yields a large dataset required for model building.

6.2.2 Structure of data.

The dataset contains players performance with the following main attributes,

Attributes	Descriptions
GP	Games Played
POS	Position
MP	Minutes Played
PTS	Points
FG	Field Goals (both 2 point field goals and 3 point field goals)
FGA	Field Goal Attempts (includes both 2-point field goal attempts and 3-point field goal attempts)
FG%	Field Goal Percentage; the formula is FG / FGA.
3P	3-Point Field Goals (available since the 1979-80 season in the NBA)
3PA	3-Point Field Goal Attempts (available since the 1979-80 season in the NBA)
3P%	3-Point Field Goal Percentage (available since the 1979-80 season in the NBA); the formula is 3P / 3PA.
FT	Free Throws
FTA	Free Throw Attempts
FT%	Free Throw Percentage; the formula is FT / FTA
ORB	Offensive Rebounds
DRB	Defensive Rebounds
TRB	Total Rebounds
AST	Assist
AST%	Assist percentage the formula is 100 * AST / (((MP / (Tm MP / 5)) * Tm FG) - FG). Assist percentage is an estimate of the percentage of teammate field goals a player assisted while he was on the floor.
STL	Steal
BLK	Block
BLK%	Block Percentage the formula is 100 * (BLK * (Tm MP / 5)) / (MP * (Opp FGA - Opp 3PA)) Block percentage is an estimate of the percentage of opponent two-point field goal attempts blocked by the player while he was on the floor
TOV	Turnover
TOV%	Turnover percentage the formula is 100 * TOV / (FGA + 0.44 * FTA + TOV) Turnover percentage is an estimate of turnovers per 100 plays.
	and etc.

Table 1: Data attributes and its description

There are other attributes such as VORP, WS, WS/48 and etc. which can be known further from the following page **basketball-reference.com** glossary. These attributes contains both continuous and categorical data type.

6.2.3 Pre-processing.
The pre-processing process consists of following steps,

- Missing value imputation - Missing values in the data are imputed by taking median of a column.
- Converting target variable or variable to be predicted to category data type.
- Filtering data using a constraint for getting useful insights from data.
- Sampling, if the classes (response variable data) are imbalanced.
- Checking for linear relationship for best separaion in case of classification and best line fitting in case of regression.
- Scaling, that is, normalizing the data to have normal distribution.

6.3 Machine learning algorithms

This section describes about machine learning algorithms that were chosen for each model and reasons for chosing them.

6.3.1 Model 1 and Model 2.
The model 1 performs predicting whether a players will stay in the league for more than 5 years or not. The dataset used for this task is taken from *data.world* which contains 21 attributes where 20 attributes are independent variables and 1 attribute is response variable and has 1340 rows. After performing the above mentioned pre-processing steps, the separability between independent variables and dependent variable was found using scatter plot between games played and points earned coloured by response variable, see figure 4.

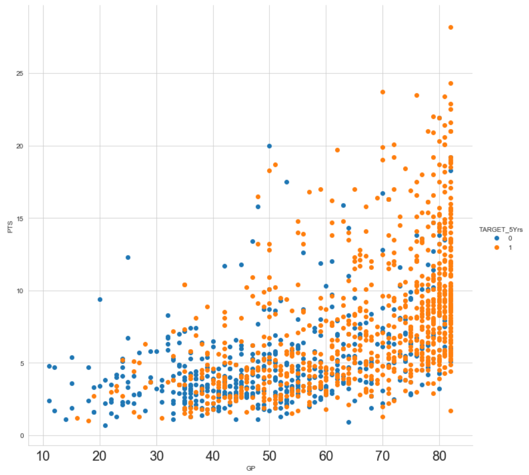


Figure 4: Model 1 scatter plot between independent and dependent variables

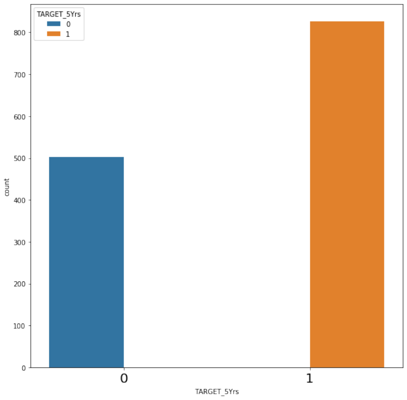


Figure 5: Model 1 count plot for classes in response variable

The model 2 performs predicting position in the basketball court based on the player's previous performance in the earlier regular seasons. The dataset used for this task is taken from *basketball-reference.com* which consists of 47 attributes where 46 attributes are independent variable and 1 variable is response variable and has over 10,000 rows of data. After performing the above mentioned pre-processing steps, the separability between independent variables and dependent variable was found using scatter plot between games played and points earned coloured by response variable, see figure 6.

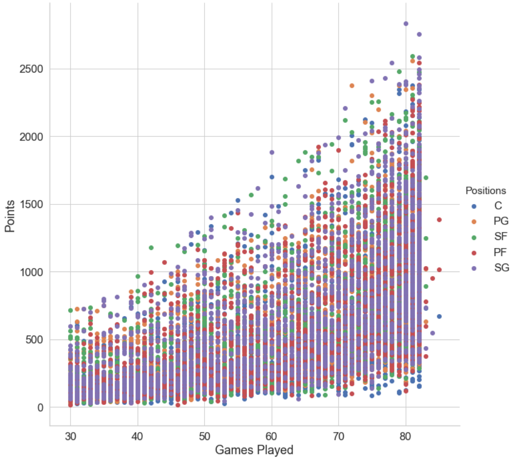


Figure 6: Model 2 scatter plot between independent and dependent variables

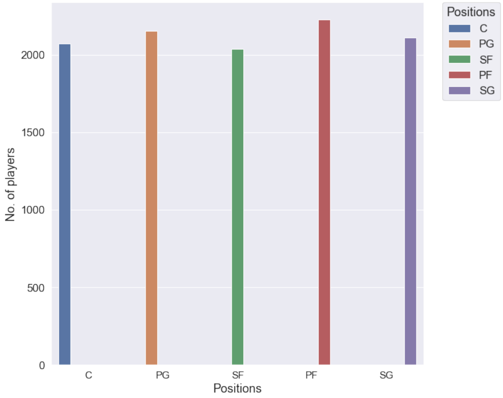


Figure 7: Model 2 count plot for classes in response variable

From the scatter plots, it can be observed that the data are not linearly separable. Moreover, it is not possible to create a classification boundary to separate data points that belong to differenct classes of the response variable. Hence, non-linear classification algorithms such as Support Vector Machine (SVM), K-Nearest Neighbor, and etc. has to be applied. Given the non-linear separability constraint and imbalanced dataset (model 1), see figure 5, it is best to choose Support Vector Machine (SVM) algorithm with kernel trick for both model 1 and model 2 because they train very well and yield good prediction results than other supervised learning algorithms [10]. Before getting into detailed execution and discussion of test results, let us first understand about Support Vector Machine (SVM) algorithm.

Support Vector Machine (SVM) is one of the supervised learning algorithm used for classification, regression and outlier detection. The objective of SVM is to find a hyperplane in N-dimensional space (N - number of attributes or features) which distinctly classifies the data points, see figure 8.

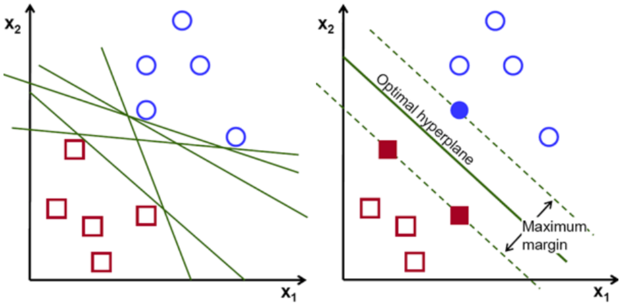


Figure 8: Hyperplane separating datapoints in N-dimesnional space

From the above figure, we could see that to separate two classes of data points, there are many hyperplanes that could be chosen. In order to choose the hyperplane that best separates the two classes of data points, we need to find *maximum margin* - the maximum distance between data points of both classes. Let us now look at hyperplanes and support vectors followed by maximum margin.

Hyperplanes are classification boundary that separated data points. Data points falling to either side of the hyperplane are attributed to different classes. The dimension of hyperplane depends on the number of input features, that is, if the number of input feature is 2, then the hyperplane is a line, see figure 9. If the number of input feature is 3, then the hyperplane becomes two dimensional plane, see figure 10. It might not be possible to imagine, if the number of features exceeds 3.

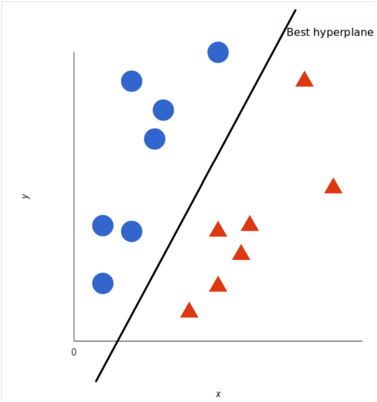


Figure 9: Hyperplane in 1-dimesnional space

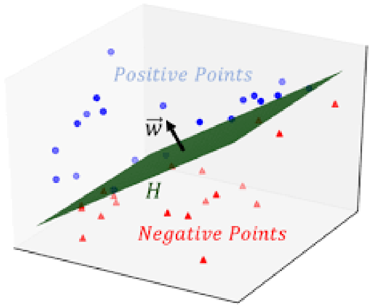
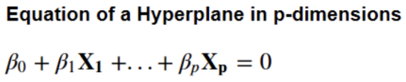


Figure 10: Hyperplane in 2-dimesnional space

Support vectors are data points which are closer to the hyperplane. The orientation and position of hyperplane are decided by these support vectors. The margin of the classifier is maximized using these support vectors. In SVM, we are trying to maximize the distance between the data points and hyperplane which is achieved by applying a loss function - *Hinge Loss*. The intuition behind maximizing the margin is to avoid *misclassification* of data points, that is, if we have made a mistake in placing the boundary, then the model will not generalize well on unforseen data. Moreover, maximizing the margin avoids local minima [11].

The equation for hyperplane in P-dimension is given below.

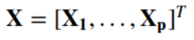


Equation of a Hyperplane in p-dimensions

$$\beta_0 + \beta_1 \mathbf{X_1} + \ldots + \beta_p \mathbf{X_p} = 0$$

Figure 11: Equation for P-dimesnional space

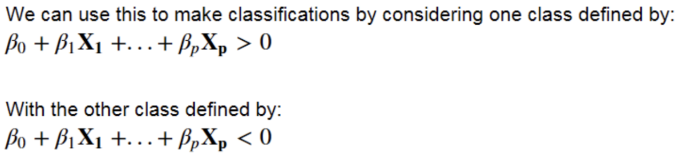
Where the datapoints are, see figure 12,



$$\mathbf{X} = [\mathbf{X_1}, \ldots, \mathbf{X_p}]^T$$

Figure 12: Equation for P-dimesnional space

and the datapoints are classified into different classes based on the below equation, see figure 13



We can use this to make classifications by considering one class defined by:

$$\beta_0 + \beta_1 \mathbf{X_1} + \ldots + \beta_p \mathbf{X_p} > 0$$

With the other class defined by:

$$\beta_0 + \beta_1 \mathbf{X_1} + \ldots + \beta_p \mathbf{X_p} < 0$$

Figure 13: Equation for classifying data points

Now, we understand that data which are not linearly separable are projected to higher dimensions and classified, but it may not be easy to perform this, if the number of features increase. It would be computationally expensive to project large dimesnional data to higher dimensions. This where *Kernel Trick* is applied in order to resolve this problem [26]. For model 1 and model 2, the number of features are high in number and the data is not linearly separable. Hence, SVM kernel trick has to be used to solve the task of classifying players and predicting positions of players. "The trick is that kernel methods represent the data only through a set of pairwise similarity comparisons between the original data observations x (with the original coordinates in the lower dimensional space), instead of explicitly applying the transformations $\phi(x)$ and representing the data by these transformed coordinates in the higher dimensional feature space" [26]. There are many kernels available in SVM namely *Linear*, *polynomial*, *Radial Base Function (RBF)*, and *sigmoid*. From the list of aforementioned kernels, only polynomial and rbf kernel methods can be applied to our data. Before choosing the kernel method that best suits our data, we need to understand some hyperparameters such as gamma, and C.

- Gamma - inverse of the standard deviation of the RBF kernel (Gaussian function), which is used as similarity measure between two points [17].
- C - Penalty factor for misclassification. [17].

The best kernel method is chosen by performing *GridSearch Cross Validation* with various values for aforementioned hyperparameters. This method provides the hyperparameters with values and kernel method which generalizes well on unforseen data. Based on the values provided by GridSearch method, the models are trained by using SVM function from sklearn library on a part of data and tested on remaining part of the data. The test results of the models are explained in the forthcoming sections.

Finally, model 3 performs clustering or grouping of players into groups based on their previous performances. This is an unsupervised machine learning task where the dataset does not contain labels for prediction or grouping. The dataset used for this task is taken from *basketball-reference.com* which consists of 47 attributes where 46 attributes are independent variable and 1 variable is response variable and has over 28,000 rows of data. Data was grouped by players and fileterd based on the number of games played by players is greater than 30 after performing the above mentioned pre-processing steps because data with games played less than 30 will not provide meaningful insight. There were 3 features missing from the data namely FT% - Free Throw percentage, 2P% - 2 Point percentage, and 3P% - 3 Point percentage which were created by applying feature engineering, that is, creating new features with available data points. The simplest form of clustering is the partitional clustering which aims at partitioning the given data into set of disjoint subsets (clusters). The most commonly used clustering criterion is *clustering error* which measures the quality of clustering. A popular clustering method that minimizes the clustering error is the k-means algorithm [12]. Hence, K-means clustering algorithm would be a good technique to cluster the players. Moreover, in K-means we can specify the number of clusters we require which is an added advantage. Elbow method, see figure 14, Silhouette score, see figure 15 were used to find the optimal number of clusters. From the plot in figure 14, it will be harder to find the optimal number of clusters. Hence, there is another way to find the optimal number of clusters, **Silhouette Score** method. Silhouette score ranges between -1 to +1 where values close to -1 implies the data points are assigned to wrong cluster and values closer to +1 means data points are closer to the cluster and belong to the right cluster. From the plot in figure 15, it could be seen that optimal number clusters is **K = 2**. Before getting into detailed execution and discussion of test results, let us first understand about K-means algorithm.

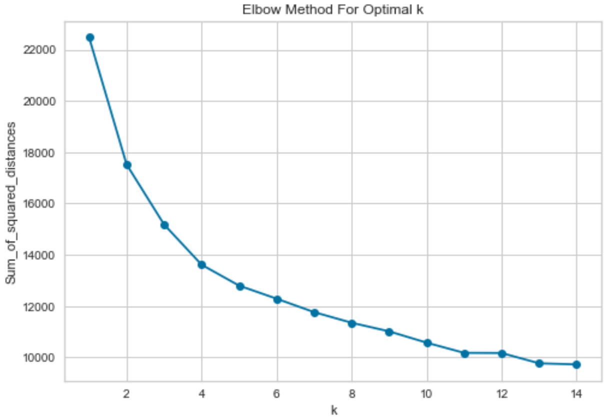


Figure 14: Elbow method

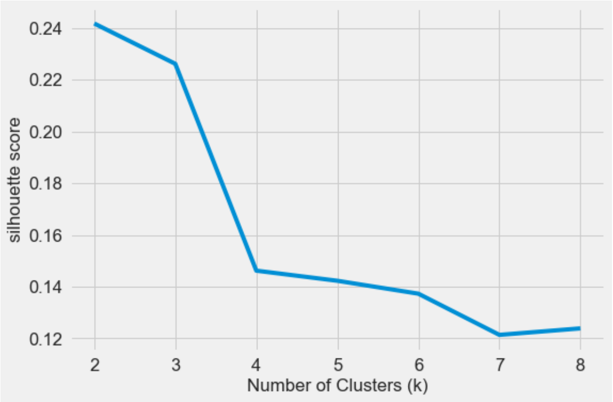


Figure 15: Silhouette score

K-means algorithm is one of the unsupervised algorithm mainly used for clustering of unlabelled data. A good clustering solution is one that finds clusters such that the observations within each cluster are more similar than the clusters themselves, see figure 16.

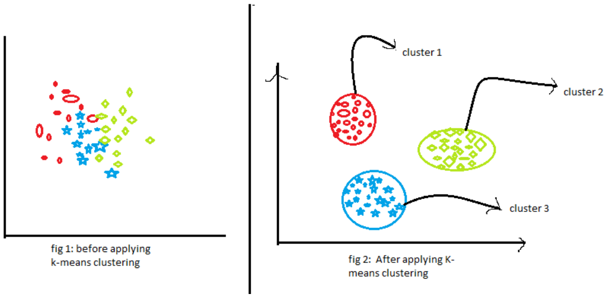


fig 1: before applying k-means clustering
cluster 1
cluster 2
cluster 3
fig 2: After applying K-means clustering

Figure 16: K-means clustering

The process behind K-means algorithm is very simple. Firstly, we need to choose an appropriate value for "K". Secondly, we need to randomly choose an initial centroid (centre coordinates) for each cluster, then we need apply two step process which is given below,

- Assignment step - Assign each data point to it's nearest centre.
- Update step - Update the centroids as being the centre of of their observation.

We repeat these steps over and over until there is no further change in the clusters. At this point the algorithm converges and we retrieve the final clusterings. Every data point is assigned to each of the clusters thus reducing the incluster sum of squares. In other words, K-means identifies K centroids and allocates the data points to each K centroids while keeping the centroid as small as possible [6].

The pictorial representation of working of K-means algorithm, see figure 17,

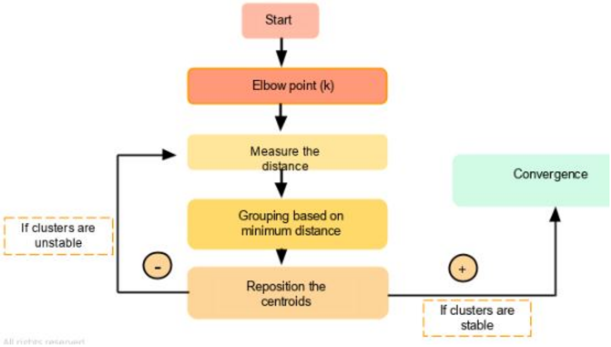


Figure 17: Working of K-means clustering

The value of "K" is determined by two methods namely *Elbow method* and *Silhouette score*. However, elbow method is the most commonly used to determine K value. The elbow method runs the K-means algorithm on the dataset for various K values, then for each K value computes the average score for all clusters and stores. When these values are plotted to visually determine the best value for K. If the line chart looks like an arm, then the “elbow” (the point of inflection on the curve) is the best value of k. The “arm” can be either up or down, but if there is a strong inflection point, it is a good indication that the underlying model fits best at that point [20]. The pictorial representation of choosing K using elbow method, see figure 18. The test results of this model is explained in the forthcoming sections.

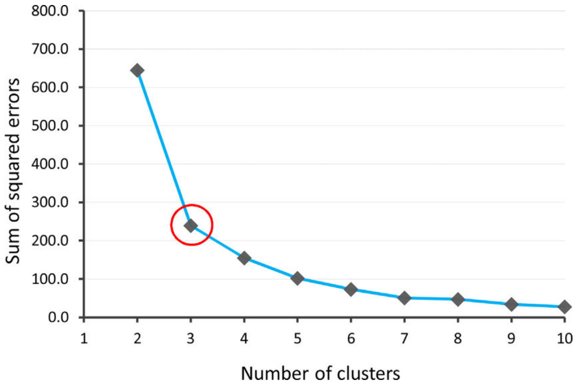
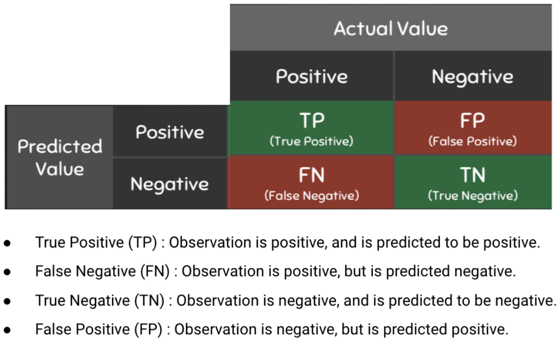


Figure 18: Elbow Method

6.4 Evaluation Metrics

In the field of machine learning there are several measures to know the quality and characteristics of a model. The most commonly used evaluation metrics for classification task are *Accuracy*, *Recall*, *Precision*, *F1 score* and *Receiver Operating Characteristic (ROC) curve*. The evaluation metrics used for regression task are *Adjusted* R^2, *Mean Squared Error(MSE)*, *Root-Mean-Squared-Error(RMSE)*, and etc. The evaluation metrics used for clustering tasks are *Silhouette Coefficient* and *Dunn’s index* [18]. In this project, we are going to use the evaluation metric for classification and clustering alone. Let us first understand about each evaluation metric before reading further.

Evaluation of the performance of a classification model is based on the counts of test records correctly and incorrectly predicted by the model. The confusion matrix, see figure 19 which provides a more insightful picture which is not only the performance of a predictive model, but also which classes are being predicted correctly and incorrectly, and what type of errors are being made [3].

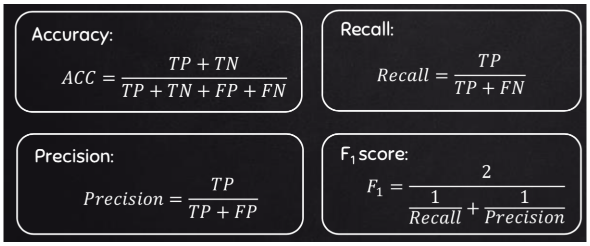


		Actual Value	
		Positive	Negative
Predicted Value	Positive	TP (True Positive)	FP (False Positive)
	Negative	FN (False Negative)	TN (True Negative)

- True Positive (TP) : Observation is positive, and is predicted to be positive.
- False Negative (FN) : Observation is positive, but is predicted negative.
- True Negative (TN) : Observation is negative, and is predicted to be negative.
- False Positive (FP) : Observation is negative, but is predicted positive.

Figure 19: Confusion Matrix

Accuracy: In general, the accuracy metric measures the ratio of correct predictions over the total number of instances evaluated [9], see figure 20 for formula.
Recall: Recall is used to measure the fraction of positive patterns that are correctly classified [9], see figure 20 for formula.
Precision: Precision is used to measure the positive patterns that are correctly predicted from the total predicted patterns in a positive class [9], see figure 20 for formula.
F1 Score:This metric represents the harmonic mean between recall and precision values [9], see figure 20 for formula.



Accuracy:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}$$

Recall:

$$Recall = \frac{TP}{TP + FN}$$

Precision:

$$Precision = \frac{TP}{TP + FP}$$

F_1 score:

$$F_1 = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}}$$

Figure 20: Accuracy, Precision, Recall and F1 Score

ROC Curve:ROC is a major visualization technique for presenting the performance of a classification model. It summarizes the trade-off between the *true positive rate (tpr)* and *false positive rate (fpr)*, see figure **??** for a predictive model using different probability thresholds [3].

$$true\ positive\ rate = \frac{true\ positives}{true\ positives + false\ negatives} \qquad false\ positive\ rate = \frac{false\ positives}{false\ positives + true\ negatives}$$

Figure 21: Equation of tpr and fpr

A ROC curve, see figure 22, plots the true positive rate (tpr) versus the false positive rate (fpr) as a function of the model's threshold for classifying a positive. Finally, we can assess the performance of the model by the area under the ROC curve (AUC). As a rule of thumb, 0.9−1 = excellent; 0.8-.09 = good; 0.7−0.8 = fair; 0.6−0.7 = poor; 0.50−0.6 = fail [3].

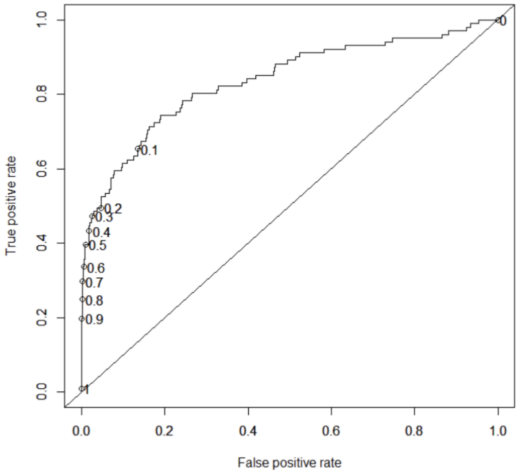


Figure 22: ROC Curve

Clusters are evaluated based on some similarity or dissimilarity measure such as the distance between cluster points. If the clustering algorithm puts the data points of similar distance together and disimilar points away from the cluster, then it has performed well. The aforementioned two metrics for clustering algorithm is given below,

Silhouette Coefficient

$$s = \frac{b - a}{max(a, b)}$$

Figure 23: Silhouette Coefficient

The silhouette coefficient consists of two scores in the above formula where,

- a - The mean distance between a sample and all other points in the same cluster.
- b - The mean distance between a sample and all other points in the next nearest cluster.

For each sample, mean silhouette coefficient is calculated and if the value is bound between -1, then it is incorrect clustering. If the value is bound between 1, then it is correct clustering. If the value is around zero, then it is overlapping clusters [1].

Dunn's Index

Dunn's Index is equal to the minimum inter-cluster distance divided by the maximum cluster size. Large inter-cluster distance = better separation, smaller inter-cluster distance = more compact cluster leading to high Dunn's index value. A higher Dunn's index score represents better clustering [1]. In the next section, we shall read about the experimental results of 3 models along with their evaluation scores and plots.

7 EXPERIMENTAL RESULTS & DISCUSSIONS

In this section, we will be seeing the results of 3 machine learning models applied on aforementioned datasets. The output is evaluated based on the metrics mentioned in *section 6.4*.

7.1 Model 1

In model 1, after performing necessary pre-processing techniques, the dataset was split into three sets of data.

- 1st set of data contains all features or all attributes.
- 2nd set of data contains only selected features based on correlation plot, see figure 24.
- 3rd set of data contains only selected features based on sklearn's selectKBest method.

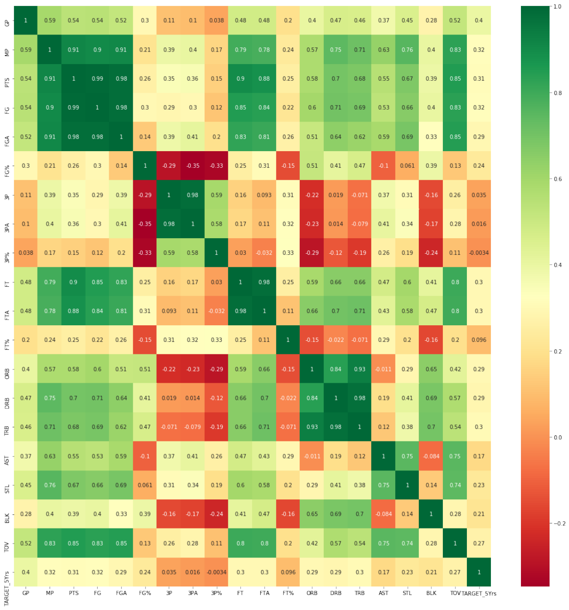
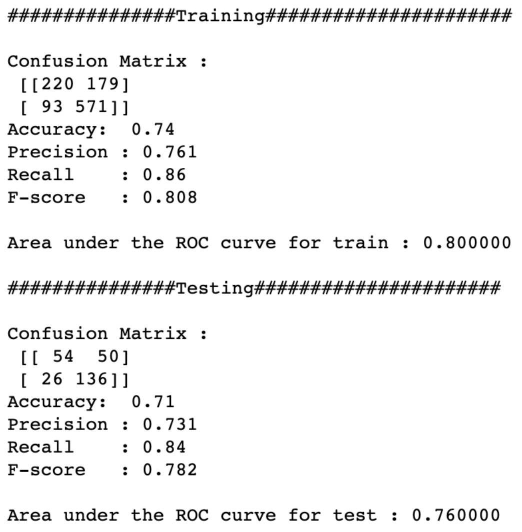


Figure 24: correlation plot for model 1

From the correlation plot, the attributes 3P, 3PA, and 3P% were dropped from the data leading to 2nd set of data. Using the sklean's selectKBest method, the attributes namely GP, MP, PTS, FG, FGA, FT, FTA, ORB, DRB, and TRB were selected leading to 3rd set of data. The 3 sets of data are split into two parts each - train data and test data where train data contains 80% of data and test data contains 20% of the data. This split up was done using sklearn's *train-test-split* method. SVM classification method with hyperparameter **C = 1** was applied on each train data and tested on each test data. The results of 3 SVM classification models are given in the forthcoming sections.

7.1.1 SVM Model 1.

This model used dataset with all features and produced the following results.



```
###############Training######################

Confusion Matrix :
 [[220 179]
 [ 93 571]]
Accuracy:  0.74
Precision : 0.761
Recall    : 0.86
F-score   : 0.808

Area under the ROC curve for train : 0.800000

###############Testing#####################

Confusion Matrix :
 [[ 54  50]
 [ 26 136]]
Accuracy:  0.71
Precision : 0.731
Recall    : 0.84
F-score   : 0.782

Area under the ROC curve for test : 0.760000
```

Figure 25: Classification Report for SVM model with all features

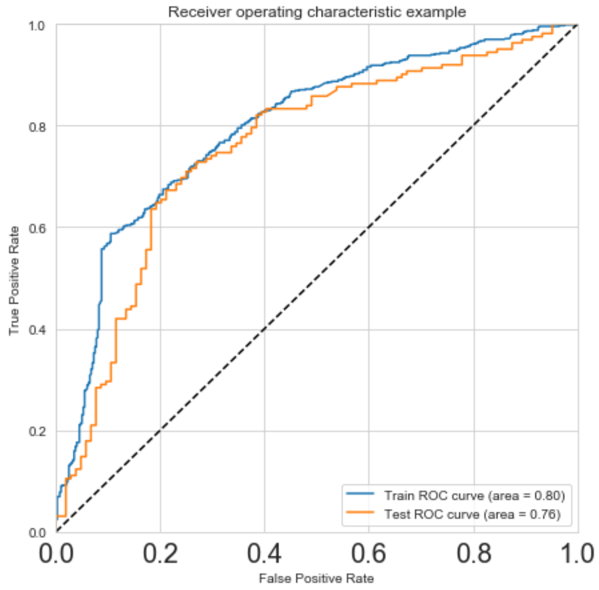
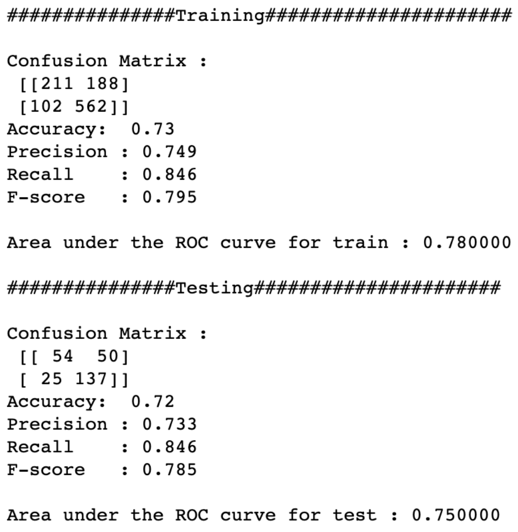


Figure 26: AUC-ROC curve for SVM model with all features

7.1.2 SVM Model 2.

This model used dataset with selected features from correlation plot and produced the following results.



```
###############Training#####################

Confusion Matrix :
 [[211 188]
 [102 562]]
Accuracy:  0.73
Precision : 0.749
Recall    : 0.846
F-score   : 0.795

Area under the ROC curve for train : 0.780000

###############Testing#####################

Confusion Matrix :
 [[ 54  50]
 [ 25 137]]
Accuracy:  0.72
Precision : 0.733
Recall    : 0.846
F-score   : 0.785

Area under the ROC curve for test : 0.750000
```

Figure 27: Classification Report for SVM model with selected features from correlation plot

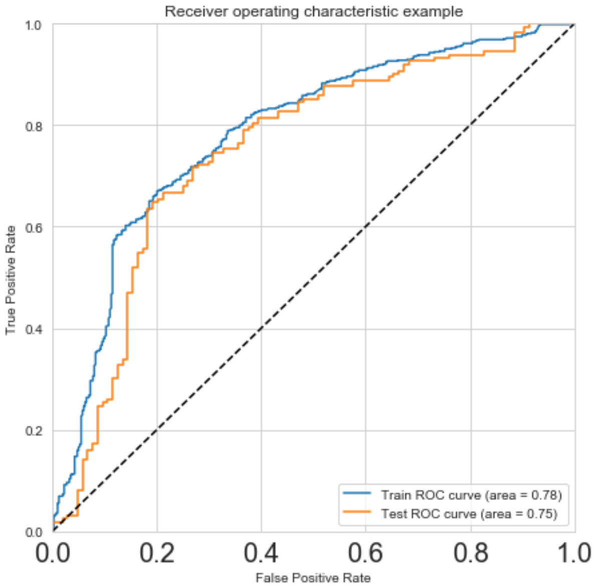
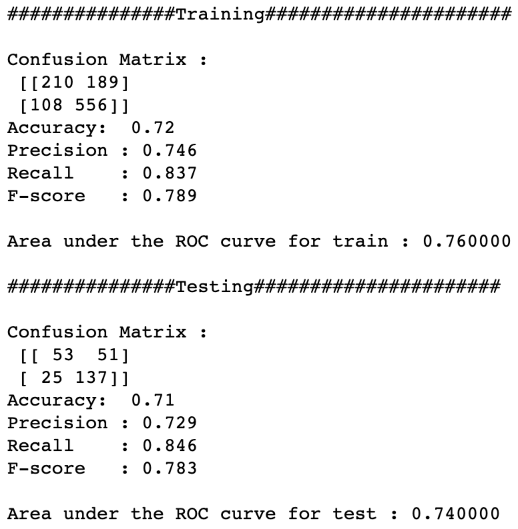


Figure 28: AUC-ROC curve for SVM model with selected features from correlation plot

7.1.3 SVM Model 3.

This model used dataset with selected features from feature selection and produced the following results.



```
###############Training#####################

Confusion Matrix :
 [[210 189]
 [108 556]]
Accuracy:  0.72
Precision : 0.746
Recall    : 0.837
F-score   : 0.789

Area under the ROC curve for train : 0.760000

###############Testing#####################

Confusion Matrix :
 [[ 53  51]
 [ 25 137]]
Accuracy:  0.71
Precision : 0.729
Recall    : 0.846
F-score   : 0.783

Area under the ROC curve for test : 0.740000
```

Figure 29: Classification Report for SVM model with selected features from feature selection

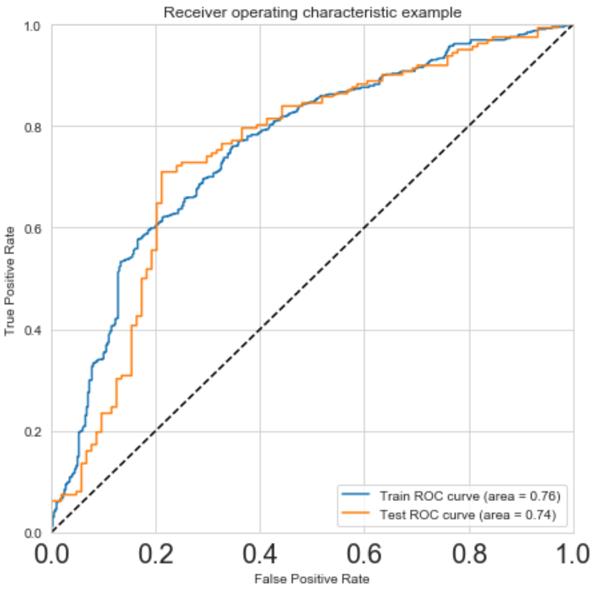


Figure 30: AUC-ROC curve for SVM model with selected features from feature selection

From the above AUC ROC curve plots (with all features, With selected features based on domain knowledge and with features selected using feature selection techniques), it can be seen that model performs better with all features than with selected features. Also comparing our AUC ROC curve plot(with all features) with the above theoritical AUC ROC curve, see figure 31 [2], it can be seen that our model is a skillful classifier . Since the dataset is imbalanced, we need to use different metrics to evaluate the model, hence it is best to choose AUC ROC curve over accuracy. Based

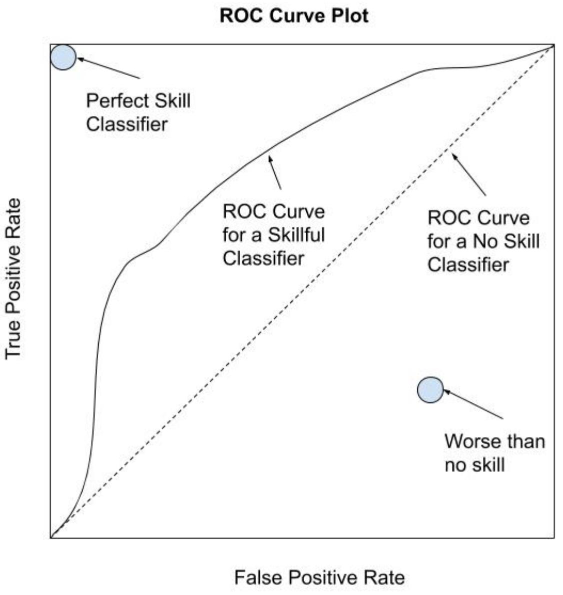


Figure 31: Theoritical ROC curve

on the observation, it can be seen that AUC ROC value with all features is higher (76% for test data and 80% for training data) than the ROC value with selected features. Moreover, we could see that the model's performance is average, this could be because of one of the following reasons:

- Imbalanced Dataset.
- Small data size.

The finalized model is saved using *pickle* package in Python for future use.

7.2 Model 2

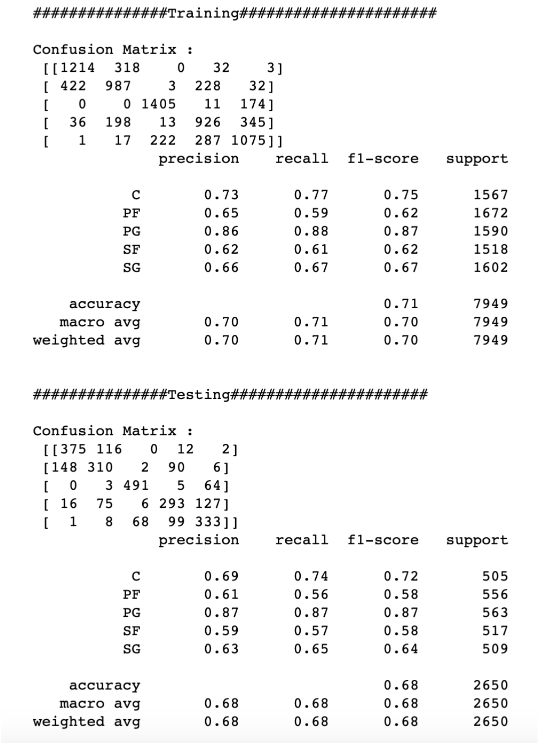
In model 2, after performing necessary pre-processing techniques, the dataset was split into two sets of data.

- 1st set of data contains all features or all attributes.
- 2nd set of data contains only selected features based on sklearn's selectKBest method.

It may not be possible to select features from the correlation plot, as the number of attributes for this data was very large. Hence, sklearn's *selectKBest* method would be an ideal choice to select the features from the dataset. Using the sklean's selectKBest method, the attributes namely **Position, 3PAr, ORB%, DRB%, TRB%, AST%, BLK%, DBPM, ORB, AST,** and **BLK** were selected leading to 2nd set of data. The 2 sets of data are split into two parts each - train data and test data where train data contains 80% of data and test data contains 20% of the data. This split up was done using sklearn's *train-test-split* method. SVM classification method hyperparameters were calculated by using sklearn's *GridSearchCV* method which produced the following values for the hyperparameters, **C: 10, gamma: 0.01, and kernel: 'rbf'**. SVM classification method with the calculated hyperparameters was applied on each train data and tested on each test data. The results of 2 SVM classification models are given in the forthcoming sections.

7.2.1 *SVM Model 1.*

This model used dataset with selected features and produced the following results.

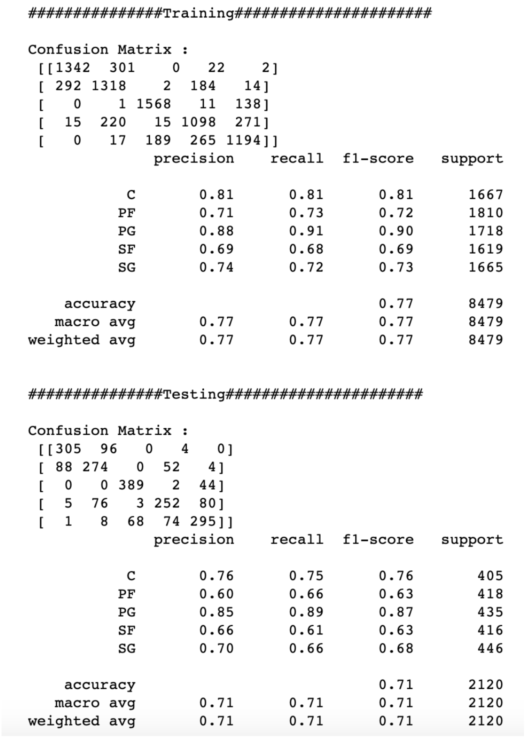


```
##############Training#####################

Confusion Matrix :
 [[1214  318    0   32    3]
 [ 422  987    3  228   32]
 [   0    0 1405   11  174]
 [  36  198   13  926  345]
 [   1   17  222  287 1075]]
              precision    recall  f1-score   support

           C       0.73      0.77      0.75      1567
          PF       0.65      0.59      0.62      1672
          PG       0.86      0.88      0.87      1590
          SF       0.62      0.61      0.62      1518
          SG       0.66      0.67      0.67      1602

    accuracy                           0.71      7949
   macro avg       0.70      0.71      0.70      7949
weighted avg       0.70      0.71      0.70      7949


##############Testing#####################

Confusion Matrix :
 [[375 116   0  12   2]
 [148 310   2  90   6]
 [  0   3 491   5  64]
 [ 16  75   6 293 127]
 [  1   8  68  99 333]]
              precision    recall  f1-score   support

           C       0.69      0.74      0.72       505
          PF       0.61      0.56      0.58       556
          PG       0.87      0.87      0.87       563
          SF       0.59      0.57      0.58       517
          SG       0.63      0.65      0.64       509

    accuracy                           0.68      2650
   macro avg       0.68      0.68      0.68      2650
weighted avg       0.68      0.68      0.68      2650
```

Figure 32: Classification Report for SVM model with selected features

7.2.2 SVM Model 2.

This model used dataset with all features and produced the following results.



```
##############Training#####################

Confusion Matrix :
 [[1342  301    0   22    2]
 [ 292 1318    2  184   14]
 [   0    1 1568   11  138]
 [  15  220   15 1098  271]
 [   0   17  189  265 1194]]
              precision    recall  f1-score   support

           C       0.81      0.81      0.81      1667
          PF       0.71      0.73      0.72      1810
          PG       0.88      0.91      0.90      1718
          SF       0.69      0.68      0.69      1619
          SG       0.74      0.72      0.73      1665

    accuracy                           0.77      8479
   macro avg       0.77      0.77      0.77      8479
weighted avg       0.77      0.77      0.77      8479


##############Testing#####################

Confusion Matrix :
 [[305  96   0   4   0]
 [ 88 274   0  52   4]
 [  0   0 389   2  44]
 [  5  76   3 252  80]
 [  1   8  68  74 295]]
              precision    recall  f1-score   support

           C       0.76      0.75      0.76       405
          PF       0.60      0.66      0.63       418
          PG       0.85      0.89      0.87       435
          SF       0.66      0.61      0.63       416
          SG       0.70      0.66      0.68       446

    accuracy                           0.71      2120
   macro avg       0.71      0.71      0.71      2120
weighted avg       0.71      0.71      0.71      2120
```

Figure 33: Classification Report for SVM model with all features

From the above classification reports (with all features, with features selected using feature selection techniques), it can be seen that model performs better with all features than with selected features. Based on the observation, it can be seen that accuracy value with all features is higher (71% for test data and 77% for training data) than the accuracy vlaue with selected features. The finalized model is saved using *pickle* package in Python for future use.

7.3 Model 3

In model 3, after performing necessary pre-processing and feature-engineering techniques, the dataset size was reduced by filtering based on the number of games played by players is greater than 30. K-means algorithm is applied on the filtered and scaled data with optimal number of clusters (K=2) found using Elbow method and Silhouette method. Since, we don't have labelled data to evaluate the performance of the model, we cannot calculate accuracy, F1 score, and etc. So, it would be a better choice to test the model with real world data prediction and comparing the predicted train data with real world data.

The predicted train data was split into two parts - cluster 0 data and cluster 1 data where cluster 0 holds good performing players data while cluster 1 holds worst performing players data. With the help of nba.com and another webpage, found previous regular season (2019-2020) best player was *LeBron James* [16] and one of worst players was *Gorgui Dieng* [8]. LeBron James's performance data was present in cluster 0 data, but not in cluster 1 data. Similarly, Gorgui Dieng's performance data was present in cluster 1 data, but not in cluster 0 data. This finding lead me to understand that cluster 0 is for good performing players data while cluster 1 is for players with bad performance records. In order to test this model's performance furthermore, two players (LeBron James and Miles Plumlee) performance data were taken from the season 2018-2019 in basketball-reference.com. After performing same pre-processing technique on the unforseen data, it is passed into the trained model. The model predicted or clustered LeBron James [21] to cluster 0 and Miles Plumlee [23] to cluster 1, by this result, it could be seen that this model does pretty well.

Earlier, companies used to input these data into their own statistical algorithms manually. As a result, it was time consuming and requires lot of human resources. But, this system is time efficient and less error-prone. In forthcoming section, we shall read about how these models can be improved and put into action, how this system can be used in otherways and etc.

8 CONCLUSION

The trained models were saved to disk using *Pickle* library in Python to put them into later use. The evaluation scores of these models were as expected and generalizes well on unforseen data. These models could be either incorporated into the software that will be used by team coaches, team managers, and etc. or made available as a Web API. Moreover, this system could also be used for team formation, that is, team coaches could select players whose performance data matches with the opponent team's players performance data. As a result, it would be easy to form a strategy to win the forthcoming matches. This would increase the team's popularity and revenue generation.

In conclusion, combining these 3 mathematical models would be an innovative approach for the teams in picking players every year. However, one must understand that data is trivial for these models, if there are more noises or errors in data, then the models would provide incorrect prediction. As a result, the team might underperform during a match and eventually, lose the game which is very cost expensive.

8.1 Further improvements

This approach could be further improved by performing any one of the following items;

- The models performance and prediction power can be improved by training these models on real time players data which is of larger size, that is, big data.
- Making an app using ***Streamlit*** and incorporating these models which would make the system an opensource.
- Further, many models could be linked with these models such as model to predict salary, model for injury analysis and etc.

REFERENCES

[1] AnalyticsVidhya. 2020. *Quick Guide to Evaluation Metrics for Supervised and Unsupervised Machine Learning.* https://www.analyticsvidhya.com/blog/2020/10/quick-guide-to-evaluation-metrics-for-supervised-and-unsupervised-machine-learning/ Accessed: 2020-12-27.

[2] Jason Brownlee. 2020. *Tour of Evaluation Metrics for Imbalanced Classification.* https://machinelearningmastery.com/tour-of-evaluation-metrics-for-imbalanced-classification/ Accessed: 2020-12-27.

[3] Fintech industry Clare Liu. 2020. *More Performance Evaluation Metrics for Classification Problems You Should Know.* https://www.kdnuggets.com/2020/04/performance-evaluation-metrics-classification.html Accessed: 2020-12-26.

[4] Thomas H Davenport. 2014. Analytics in sports: The new science of winning. *International Institute for Analytics* 2 (2014), 1–28.

[5] David Fumo. [n.d.]. *Types of Machine Learning Algorithms You Should Know.* https://towardsdatascience.com/types-of-machine-learning-algorithms-you-should-know-953a08248861 Accessed: 2020-12-24.

[6] Dr. Michael J. Garbade. 2018. *Understanding K-means Clustering in Machine Learning.* https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1 Accessed: 2020-12-25.

[7] Jack Green. [n.d.]. *How does the NBA draft work?* https://blog.betway.com/basketball/how-does-the-nba-draft-work-nba-draft-explained/ Accessed: 2020-12-24.

[8] JOSH HERWITT. 2020. *The 10 Worst Players in the NBA Right Now.* https://www.complex.com/sports/2019/02/10-worst-players-in-the-nba-right-now/gorgui-deng Accessed: 2020-12-27.

[9] Mohammad Hossin and MN Sulaiman. 2015. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process* 5, 2 (2015), 1.

[10] Tasadduq Imam, Kai Ming Ting, and Joarder Kamruzzaman. 2006. z-SVM: An SVM for improved classification of imbalanced data. In *Australasian Joint Conference on Artificial Intelligence.* Springer, 264–273.

[11] Vikramaditya Jakkula. 2006. Tutorial on support vector machine (svm). *School of EECS, Washington State University* 37 (2006).

[12] Aristidis Likas, Nikos Vlassis, and Jakob J Verbeek. 2003. The global k-means clustering algorithm. *Pattern recognition* 36, 2 (2003), 451–461.

[13] Zach McCann. 2012. Player tracking transforming NBA analytics. *Tech-ESPN Playbook, May* 19 (2012).

[14] Tom Michael Mitchell. 2006. *The discipline of machine learning.* Vol. 9. Carnegie Mellon University, School of Computer Science, Machine Learning ….

[15] Munni. [n.d.]. *Sports analytics have changed the way sports are played.* https://tanukamandal.com/2017/12/12/sports-analytics-changed-play/ Accessed: 2020-12-25.

[16] NBA. 2020. *Who are the best players in the NBA? Ranking the top 30 players for the 2019-20 season.* https://ca.nba.com/news/who-are-the-best-players-in-the-nba-ranking-the-top-30-players-for-the-2019-20-season/898voekw0nvd1hoeb5gvkyiu7 Accessed: 2020-12-27.

[17] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* (2011).

[18] David MW Powers. 2020. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061* (2020).

[19] Dorian Pyle and Cristina San José. [n.d.]. *An executive's guide to machine learning.* https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/an-executives-guide-to-machine-learning Accessed: 2020-12-24.

[20] Scikit. 2019. *Elbow Method.* https://www.scikit-yb.org/en/latest/api/cluster/elbow.html Accessed: 2020-12-25.

[21] Jonathan Sherman. 2019. *Bleacher Report Ranks LeBron James as 2nd Best Player of All Time.* https://lakersdaily.com/bleacher-report-ranks-lebron-james-2nd-best-player-all-time Accessed: 2020-12-27.

[22] Rusty Simmons. [n.d.]. *Competitive fire helps Kirk Lacob make his own name with Warriors.* https://www.sfgate.com/warriors/article/Competitive-fire-helps-Kirk-Lacob-make-his-own-6339796.php Accessed: 2020-12-23.

[23] B/R Studios. 2012. *Miles Plumlee to Indiana Pacers: Best and Worst-Case NBA Player Comparison.* https://bleacherreport.com/articles/1235975-miles-plumlee-best-and-worst-case-nba-player-comparison Accessed: 2020-12-27.

[24] Walter Tichy. 2016. Changing the Game: Dr. Dave Schrader on sports analytics. *Ubiquity* 2016, May (2016), 1–10.

[25] George Tzanis, Christos Berberidis, and Ioannis Vlahavas. 2009. Machine learning and data mining in bioinformatics. In *Handbook of research on innovations in database technologies and applications: Current and future trends.* IGI Global, 612–621.

[26] Drew Wilimitis. 2018. *The Kernel Trick in Support Vector Classification.* https://towardsdatascience.com/the-kernel-trick-c98cdbcaeb3f Accessed: 2020-12-25.

[27] wsbc.uoregon.edu. 2015. Industry Insights: The Evolution of Basketball through Technology | Warsaw Sports Business Club.