PREDICTING LONGEVITY (AT LEAST 5 YEAR TENURE) OF NBA PLAYERS IN THE NBA LEAGUE USING MACHINE LEARNING MODELS

Sports could be considered as a contest (usually physical) played for results or perhaps the challenges that come with it.

Sports is widely known by most cultural backgrounds around the world. One could deem it as a purposeless activity due to the word "play" in its definition as the opposite of work. Nonetheless, one could argue that this "play" expends more effort, sacrifice and time than what is widely considered as "work".

Having to play a sport is voluntary and not for the sake of getting a paycheck. In lieu of that, in the real world, this "play" known as sports could have different motivations and motives are often diverse and quite impossible to determine.

Sports have proven over decades to be imperial to the success of a nation at large. Nowadays, countries are defined by their participation in various sporting events rather than by their terrain, politics, or economic status.

Having said all that, my main motivation for this project is my predilection for the game of basketball, which is a widely acclaimed sports coming next to the game of football(soccer). Having to deal with the statistics of players I idolized when I was a kid, gives me sense of being a part of their worlds.

It is often preached in high school and college, not to play so much attention to your statistics in a game but rather be more concerned about intangibles you possess as a player. Intangibles such as leadership skills, communication skills, relationship skills with your team members or organization's staff as a whole. It could be realized that most people who get drafted into the big leagues are not drafted high just because of their intangible capabilities. Their stats stand out in their selections to the big leagues and this prompted my research. This is an unanswered question on the minds of most college athletes trying to make it to the big leagues of any kind.

This paper will focus on the National Basketball Association (NBA), which is widely know as the best basketball in the world. It will take into account the stats of drafted players while they were in college and their tenure in the NBA, and use the average in making predictions as to whether a player will be survive at least five(5) years in the NBA.

One may ask the question, why 5 years? Well, the average longevity of a successful NBA player is 5 years. And players get into the league with an aspiration of lasting at least 5 years which is a great feat considering the level of competitiveness Thus, the stats of a desired player will be used in predicting this target(of enduring at least 5 years in the league)

In consequence, this brewed the motivation to predict how drafted players could last and giving them ideas to knowing how much work to put in.

PROBLEM

There is a lot of work and sacrifice that goes into doing this. College students literally compromise on their studies to train and put extra tolls on their bodies unlike their colleagues to make it that far as the NBA. It is depressing when you are not able to make it there and hence the need for potential players and even current players to have insight of stats could affect their survival in the said league.

The difficulty in knowing approximate advanced stats that will give a potential NBA player the chance to make it to the league and possibly last in there for 5 years is a lingering torn in their flesh.

The challenge of knowing exactly which aspects of your game to improve to be able to make the cut always stands out as a battle in the mind. Moreover, knowing which capabilities stand out to scouts and recruiters.

DATA

A link to the dataset could be found below:

https://github.com/kjaisingh/nba-rookie-predictor/blob/master/nba_logreg.csv

This dataset was retrieved from nbareference.com, a trusted source for NBA statistics, through GitHub.

The dataset of a '.csv' is then converted to a dataframe using the pandas library in python.

The original data frame had a shape of (1340 x 21), which explains that there are 1340 datapoints and 21 features to deal with. These datapoints contain stats of each NBA player and the features are basically the names given to these specific stats.

Features: Names, MIN, PTS, FGM, FGA, FG%, 3P Made, 3PA, 3P%, FTM, FTA, FT%, OREB, DREB, REB, AST, STL, BLK, TOV, TARGET_5Yrs

To clearly understand the features, we would have to spell out a glossary which could be found below:

a. Name,

b. GP (Games Played)

c. MIN (Minutes played per game)

d. PTS (Points Scored per game)

e. FGM (Field Goals Made)

f. FGA (Field Goals Attempted)

g. FG% (Field Goal Percentage)

h. 3P Made (3 pointers Made)

i. 3PA (3 pointers Attempted)

j. 3P% (3 point percentage)

k. FTM (Free Throws Made)

l. FTA (Free Throws Attempted)

m. FT% (Free Throw Percentage)

n. OREB (Offensive Rebounds)

o. DREB (Defensive Rebounds)

p. REB (Rebounds per game)

q. AST (Assists per game)

r. STL (Steals per game)

s. BLK (Blocks per game)

t. TOV (Turnovers per game)

u. TARGET_5Yrs

EDA

The dataset had just one object datatype out of the 21, which is the column “Name” and rest as numerical (int and float) datatypes.

Just a few of the features had means of zero and standard deviations of one which clearly did not prove enough standardization.

There were a few (11) missing values in the 3P% feature space.

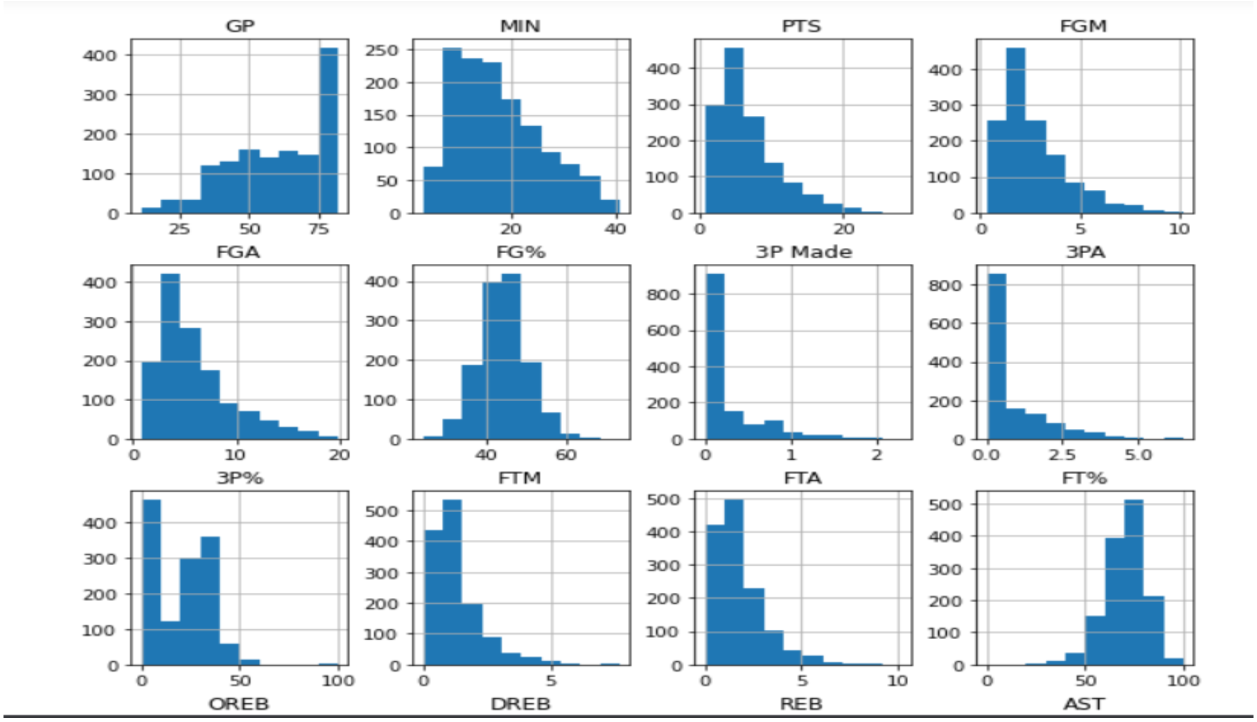
The target variable, ‘TARGET_5Yrs’ had unique values of 0 and 1.

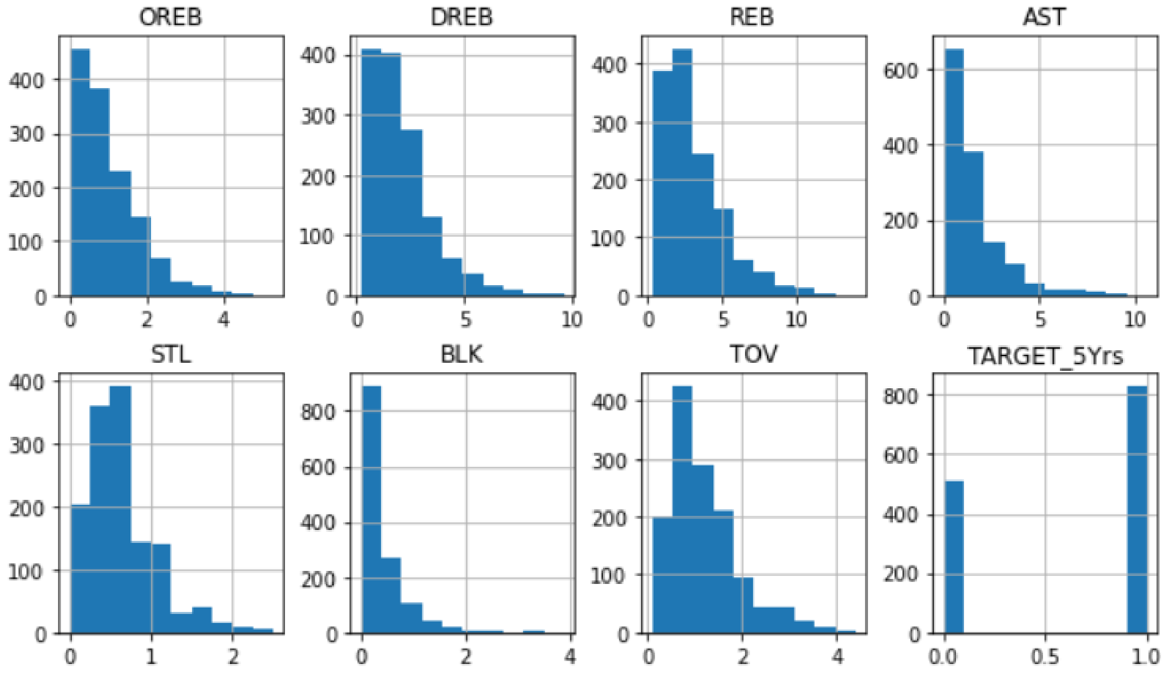
Zero(0) indicating that the player considered in that datapoint did not endure at least 5 years in the NBA.

One(1) indicating that the player considered in the particular datapoint endured at least 5 years in the NBA.

The target value had value counts of: 831 for 1.0 (i.e. Approximately 62% of the data) and 509 for 0.0.(i.e. Approximately 48% of the data) Considering this fact, it could be that this target variable is experiencing some data imbalance.

A visualization of the features using a histogram could be seen with the figures below.





Most of the features are not unimodal; that is, they do not peak once as typical of a normal distribution curve, thereby making most of them asymmetric. Some have wide spreads like 3Pt% and FT% with a few outliers. Most of them heavily skewed either to the right or left which indicates a high density in one region of the feature than the other.

Data Preprocessing

To start with, 1340 datapoints is considered to be a fairly small, yet relatively large and therefore the data processing approach adopted for this paper does not seek to find ways to reduce the number of rows unless in cases where they are deemed to be highly redundant to the purpose of this research.

As said earlier, the feature '3P%' had 11 missing values and I decided to fill these missing values with the median value which was 22.4 using the 'fillna' method in the pandas library.

Since 'Name' feature was an object and obviously proved redundant to the purpose of our study, I completely dropped that feature from the dataset.

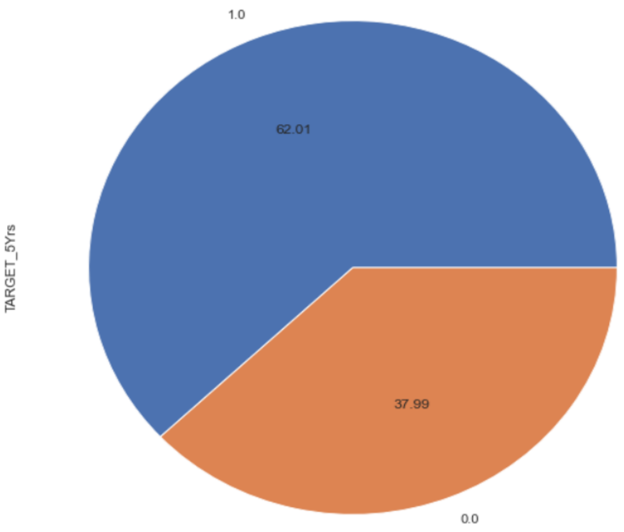
I used a box plot to visualize outliers and there was a total of 271 datapoints which were outliers. The outliers were dropped and saved to a new dataframe to assess the performance.

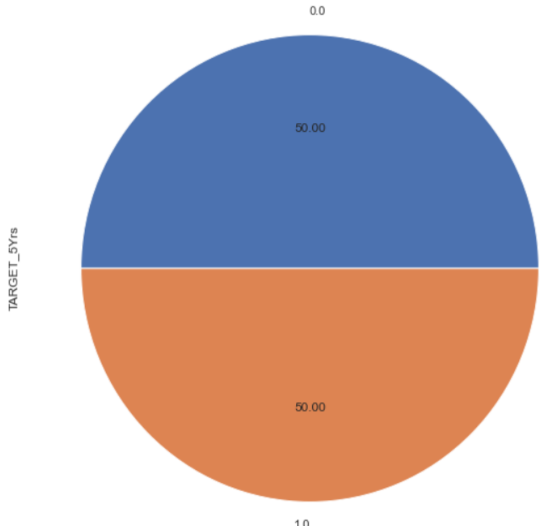
I checked the correlation coefficients for each feature to the target variable using the heatmap and compared the original dataset(df) to the dataset of dropped outliers(dfwoOutliers) and realized the correlation coefficients dropped significantly. The highest correlation coefficient to the target variable was GP which dropped from 0.4 to 0.32 and this happened to most the features and therefore decided to proceed with the original data frame with outliers.

The target variable had an imbalance where there was 62.0% (831) of 1(i.e.. Players lasting at least 5 years in the league) and 37.99%(509) of 0(Players not lasting at least 5 years in the league).

Therefore, I used the RandomOverSampler object from the imblearn library and set a ratio of 1 as the parameter. In this light, it will add 322 randomly to the minority class by picking with replacement from the majority class.

Eventually the target variable was spread 50/50 and I saved the results to a whole new data frame for future training.





Lastly, of the processed dataset, I assigned X = all columns without the Name and Target_5Yrs and set y = the column of the target variable and performed feature scaling using the StandardScaler, where mean is zero and standard deviation is 1.

Training data was set to 0.8 and test/validataion data was set to 0.2 throughout the project.

METHOD

All code was written with the python language using jupyter notebook.

The idea is to explore five(5) different machine learning models on the train and test splits.

Below are the models employed:

Logistic Regression

Random Forest

XGBoost

Neural Networks (TensorFlow)

Gradient Descent Boost

I varied the training to the models in various ways. I trained using the scaled dataset for the most part.

I employed the use of PCA(Principal Component Analysis) to further reduce the dimensions (n_components) of the original data to 5. I tried various numbers but 5 gave the best results.

In theory, Random Forest(RF) is in most cases supposed to perform better than the Logistic regression, yet for this scaled dataset, when I used the RFC model, it failed to

perform better than the Logistic regression model so I used the 'GridSearchCV' module to implement a Hyperparameter Tuning. I considered the 3 parameters which are the max_depth, min_sample_split, n_estimators).

To expand on this, it is known that the RF is an ensembled learning model which is a collection of decision trees and these trees are split. Thus, factoring in this definition, I used the parameters that dealt with the number of trees, the minimum number of sample splits and the depth of the trees.

After using this hyperparameter tuning approach on the RFC model, it performed slightly better than the previous results without the tuning.

It must be noted that the Hyperparameter tuning really used a lot of time in running because of the brute force approach in iteration.

Lastly, I trained various models with feature selections I did where I considered features with correlation coefficients better that the threshold I set for the to the target variable to be 0.3.

Logistic Regression to my surprise a basic traditional approach to classification problems performed relatively better than most of the models I used in the research.

The gradient boost was the least performing model for most of the variations I did with the datasets and hyperparameters.

XGBoost was the penultimate best performing model with the different variations in datasets and hyperparameters I employed.

The fully connected Neural Networks proved to be the best performing model, which is perhaps not a surprise. However, there was an interesting realization that the more the epochs were increasing I experienced the accuracy score peaking and eventually reducing. This prompted that at a point, the model was overfitting and therefor I had to implement an early stop of 0.75 approach based on the accuracy scores I obtained using the 'callback' method in TensorFlow to stop the epochs when it finds an accuracy score of .75.

Models (with different factors considered)	Evaluation (Accuracy Score)
Logistic regression (scaled data)	0.738
Random Forest (scaled data)	0.701
Random Forest (Hyper Parameter Tuning)	0.7164
Random Forest (Over Sampling)	0.7201
XGBoost (scaled data)	0.738
XGBoost (PCA)	0.735
XGBoost (Feature Selection)	(0.742
Gradient Boost (scaled data)	0.705
Neural Networks (Over Sampling)	0.746
Neural Networks (scaled data)	0.75

CHALLENGES AND FUTURE APPROACH

Finding appropriate datasets for the desired problem analysis was a huge problem. The imbalance in the target variable took a huge toll on the performance of the models no matter how robust the models were theoretically.

Lack of experience in knowing exactly which parameter to tweak or drop to achieve better results.
Longer minutes of execution because I couldn't figure out the best approach to hyperparameter tuning.

CONCLUSION

In as much as there is more attention given to intangibles in college, athletes should also pay close attention to their stats.

It was demonstrated that some stats prove imperative for good longevity in the NBA league and athletes should assign relatively more effort on certain stats in lieu of others.

References

https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.RandomOverSampler.html

https://builtin.com/data-science/step-step-explanation-principal-component-analysis

https://towardsdatascience.com/hyperparameter-tuning-for-machine-learning-models-1b80d783b946

https://www.youtube.com/watch?v=OJedgzdipC0&t=314s