NBA Prediction using Machine Learning Methods

Ritvik Chauhan[1], MS Information Systems, Northeastern University

Abstract. This Study attempts to predict the expected future National Basketball Association (NBA) results for the teams based on historical NBA statistics (Advanced and Basic). Specifically, this study looks at the team statistics throughout 2015-19 seasons. Various regression models are used on two different datasets to predict the future no of wins for the next season (2019-2020).

A variety of regression models were tested on both datasets: Linear Regression and Logistic Regression performing the best amongs: Support Vector Regression, Bayesian Ridge Regression, Neural Nets, Random Forest Regression and Xgboost Regression.

1 Introduction

Basketball is one of the most famous sports in the United States and has much fame abroad. It started as a basic gym exercise in the late 1800's and gradually moved from high schools and into universities. Finally, in the mid 1950's, the National Basketball Association (NBA) emerged as a major governing body of the professional version of the game. From that point onward, the NBA has managed a yearly competition between official professional teams from all over North America. The yearly competition is known as a season. Amid a season, each team goes up against others a few times and tracks their wins and losses. After a few games (in the hundreds), the 16 teams with the most wins compete with one another in an knockout tournament called Playoffs. Playoff matches are best four out of seven. The team that wins the Playoffs wins the entire season.

● Goals and objectives

My goal in this project is to attempt predicting the outcome of Basketball matches before they happen based on the characteristics of the two teams that are competing.

● Project Requirements

Project Requirements are collecting metrics, preprocessing the data, and using several regression methods.

- Linear Regression
- Logistic Regression
- Support Vector Regression
- Bayesian Ridge Regression
- Neural Nets
- Random Forest Regression
- Xgboost Regression

Multiple analytical tools were used for preprocessing, modeling the data and analyzing the results. To scrape the source data, Pandas was used. Matplotlib and Seaborn weres used for some of the data visualizations presented in this paper. The code was written in Python.

2 Background

NBA has 30 groups altogether, 15 in every conference (Eastern and Western), 5 in every division (Atlantic, Central, Southeast, Northwest, Pacific, and Southwest). NBA players are the most paid sportsmen. Fantasy Basketball is a virtual game where players can make teams dependent on real players and earn ranking dependent on the performance of their chosen basketball players. It might be beneficial for them to have the capacity to foresee how different teams will do and pick real players dependent on that. Fantasy sports generates millions per year.

Basketball is a perfect game for predictive modeling because it is a large and active market and the frequency of games is a lot higher than numerous different games, 82 games are played in a basketball season contrasted with 16 games played in a football season.

3 Initial Hypothesis

H0: Historical NBA statistical data cannot be used to predict future expected points scored per player

Linear Hypothesis

$$h_\theta(x) \neq \theta_0 + \theta_1 x$$

H1: Historical NBA statistical data is predictive of future expected points scored per player.

Linear Hypothesis

$$h_\theta(x) = \theta_0 + \theta_1 x$$

4 Data Selection, Collection and Preprocessing

The First Dataset used is NBA Advanced Stats (2018-19) and was collected from NBA-stuffer using pandas library in python. The data contains 30 rows and 24 columns
The columns in the dataset are :-

- RANK
- TEAM
- CONF
- DIVISION
- GP -> Games Played
- PTS/GM -> Points Per Game Average points per game
- aPTS/GM -> Points Allowed Per Game Average points allowed per game
- PTS DIFF -> Points Differential{(Total Points Scored) - (Total Points Allowed)}/ (Games Played)
- PACE -> Pace Estimate of Possessions Per 48 Minutes
- OEFF -> Offensive Efficiency Points scored per 100 possessions.
- DEFF -> Defensive Efficiency Points allowed per 100 possessions.
- EDIFF -> Efficiency Differential{(Total Offensive Efficiency) - (Total Defensive Efficiency)}/ (Games Played)
- SOS -> Strength of the Schedule Opponent efficiency differential average for all games played so far (venue of the games also taken into account) is used as an indicator of strength of the schedule. The higher the SoS rating, the tougher the schedule; where zero is average.
- rSOS -> Remaining Strength of the Schedule Opponent efficiency differential average for the remaining games (venue of the games also taken into account) is used as an indicator of strength of the schedule. The higher the rSOS rating, the tougher the remaining schedule; where zero is average.
- SAR -> Schedule Adjusted Rating An evaluation of teams based on efficiency differential and strength of schedule
- CONS -> Consistency Rating Consistency based on game-by-game efficiency differential variation. The higher the team has a consistency rating, the more unpredictable it is.
- A4F -> Adjusted Four Factors Calculated by applying weights to the differentials of offensive and defensive four factors. A4F explains the specified proportion of variability in wins.
- W -> Wins The most important goal in sports, unless your team is not tanking
- L -> Losses Total count of games lost
- WIN%
- eWIN% -> Correlated Gaussian Expected Winning Percentage Indicates the ideal winning percentage based on offensive and defensive performance.
- pWIN% -> Projected Winning Percentage Each point differential translates to 2.7 wins over the course of the season.
- ACH -> Achievement Level In Terms of Wins. The metric is based on differential between actual and expected winning percentage. Positive figures indicate overachievement while negative figures indicate the team should have won more games.
- STRK -> Current Streak Winning or losing streak for the season.

The Second Dataset used is NBA Basic Stats (2015-19) was collected from Basketball-reference.com using pandas library in python. The data contains 30 rows and 28 columns.
The columns in the dataset are :-
Rk -> Rank
From
To
Fr -> Franchise
Lg -> League
Yrs
G -> Games
W -> Wins
L -> Losses
W/L% -> Win-Loss Percentage
MP -> Minutes Played
FG -> Field Goals
FGA -> Field Goal Attempts
2P -> 2-Point Field Goals
2PA -> 2-Point Field Goal Attempts
3P -> 3-Point Field Goals
3PA -> 3-Point Field Goal Attempts
FT -> Free Throws
FTA -> Free Throw Attempts
ORB -> Offensive Rebounds
DRB -> Defensive Rebounds
TRB -> Total Rebounds
AST -> Assists
STL -> Steals
BLK -> Blocks
TOV -> Turnovers
PF -> Personal Fouls
PTS -> Points

After collecting the data it was cleaned all the irrelevant columns were dropped. Columns like Rank, From, To, Division, Conference etc were removed because they were irrelevant to the data. Columns like Win%, Losses, eWin%, W/L% etc were also removed because of their high correlation with the column Win.

Fig. 1. Columns Correlation 2015-2019 Basic Data

Fig. 2. Columns Correlation 2018-2019 Advanced Data

5 Overview of Regression Methods used and Results

Linear Regression

linear regression is a linear approach to modelling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression. This term is distinct from multivariate linear regression, where multiple correlated dependent variables are predicted, rather than a single scalar variable. (Linear Regression, n.d.)

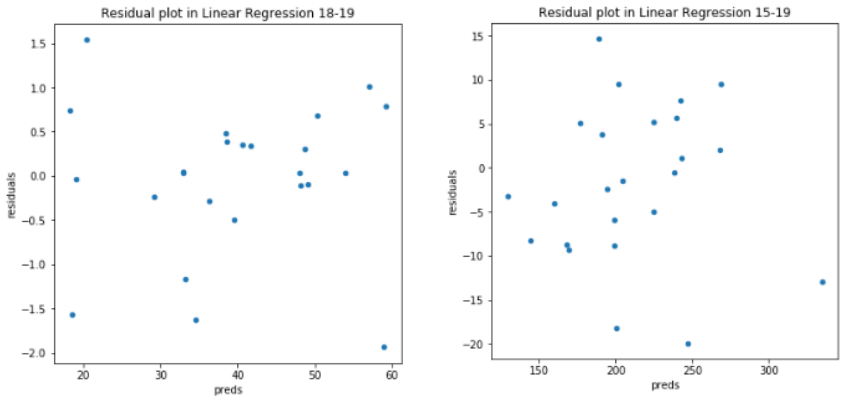

Fig. 3. Linear Regression Residual Plot

Logistic Regression

The logistic model (or logit model) is a widely used statistical model that in its basic form uses a logistic function to model a binary dependent variable, although many more complex extensions exist. In regression analysis, logistic regression (or logit regression) is estimating the parameters of a logistic model (a form of binary regression). (Logistic Regression, n.d.)

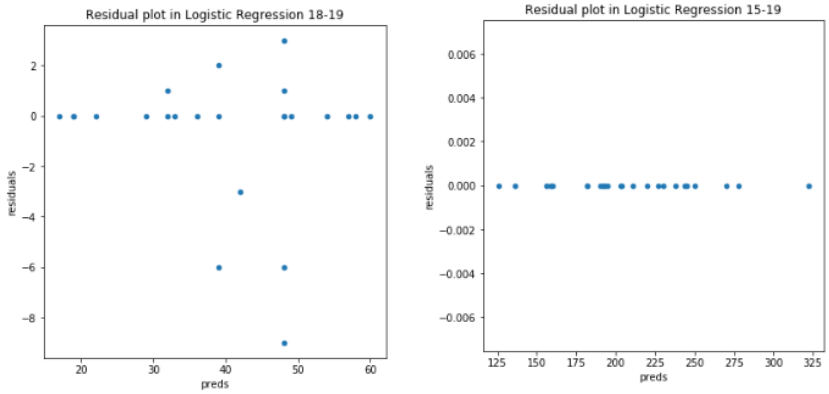

Fig. 4. Logistic Regression Residual Plot

Support Vector Machine

A support-vector machine constructs a hyperplane or set of hyperplanes in a high- or infinite-dimensional space, which can be used for classification, regression, or other tasks like outliers detection. Intuitively, a good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point of any class (so-called functional margin), since in general the larger the margin, the lower the generalization error of the classifier. (Support-vector Machine, n.d.)

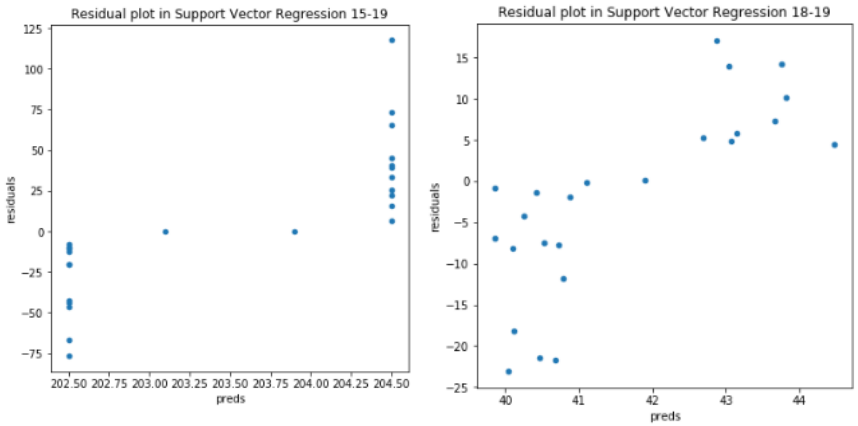

Fig. 5. Support Vector Regression Residual Plot

Bayesian Ridge Regression

Bayesian multivariate linear regression is a Bayesian approach to multivariate linear regression, i.e. linear regression where the predicted outcome is a vector of correlated random variables rather than a single scalar random variable. A more general treatment of this approach can be found in the article MMSE estimator. (Bayesian multivariate linear regression, n.d.)

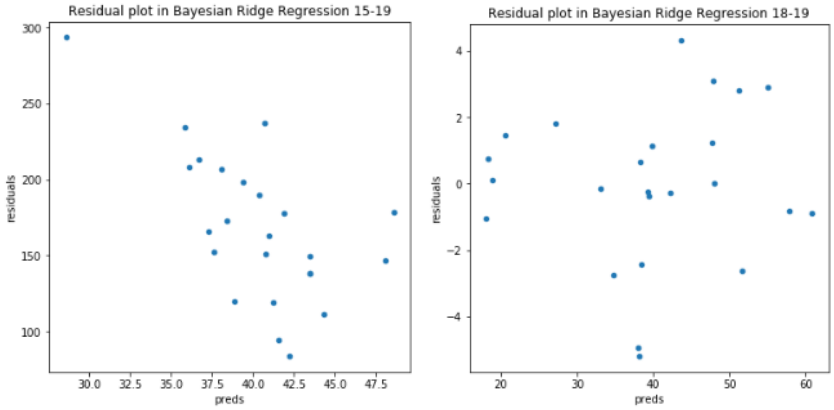

Fig. 6. Bayesian Ridge Regression Residual Plot

Neural Network

Artificial neural networks (ANN) or connectionist systems are computing systems vaguely inspired by the biological neural networks that constitute animal brains. The neural network itself is not an algorithm, but rather a framework for many different machine learning algorithms to work together and process complex data inputs. Such systems "learn" to perform tasks by considering examples, generally without being programmed with any task-specific rules. (Artificial neural network, n.d.)

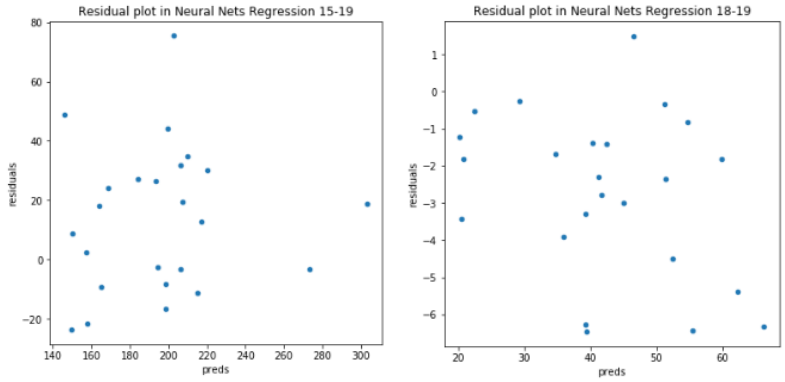

Fig. 7. Bayesian Ridge Regression Residual Plot

Random Forest Regression

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set. (Random forest, n.d.)

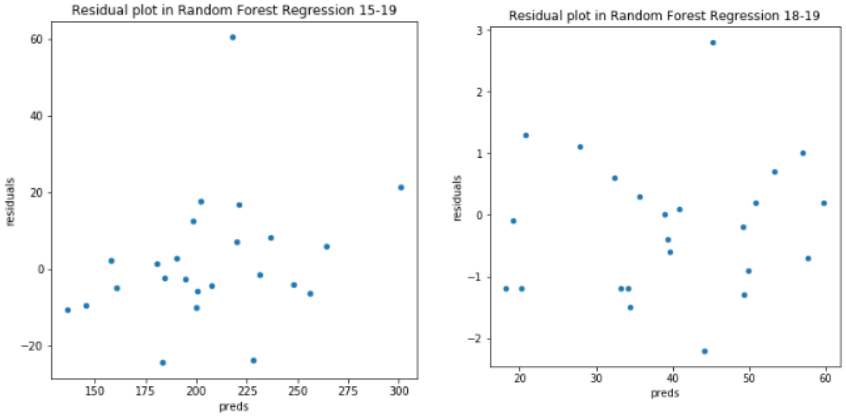

Fig. 8. Random Forest Regression Residual Plot

XGBoost Regression

XGBoost is well known to provide better solutions than other machine learning algorithms. In fact, since its inception, it has become the "state-of-the-art" machine learning algorithm to deal with structured data. XGBoost (Extreme Gradient Boosting) belongs to a family of boosting algorithms and uses the gradient boosting (GBM) framework at its core. It is an optimized distributed gradient boosting library. (Pathak, 2018)

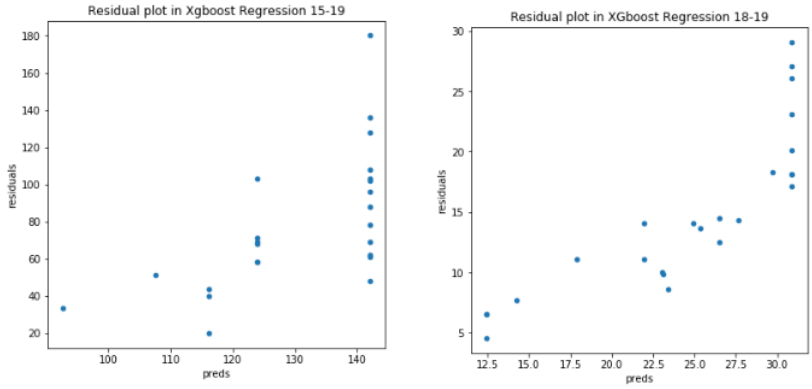

Fig. 9. XGBoost Regression Residual Plot

6 H2O and Variable Importance

H2O is open-source software for big-data analysis. It is produced by the company H2O.ai. H2O allows users to fit thousands of potential models as part of discovering patterns in data.

The H2O software runs can be called from the statistical package R, Python, and other environments. It is used for exploring and analyzing datasets held in cloud computing systems and in the Apache Hadoop Distributed File System as well as in the conventional operating-systems Linux, macOS, and Microsoft Windows. The H2O software is written in Java, Python, and R. Its graphical-user interface is compatible with four browsers: Chrome, Safari, Firefox, and Internet Explorer. (H2O, n.d.)

We used H2O to find out the most important features of our dataset or the features that H2O classifies as most important. These are the most important features that effect the Wins of a team.

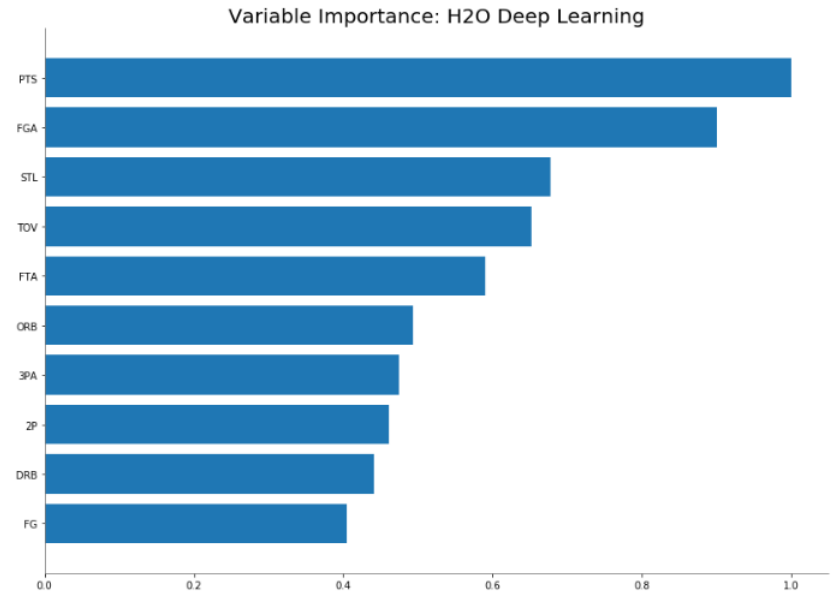

Fig. 10. Variable importance 2015-19 Basic NBA stats

According to H2O The most important variable effecting wins are:

- PTS – Points Scored :
  Makes sense as every team gets three points for a win
- FGA – Field Goal Attempts
  The more you attempt the more you score. The more you score the more games you will win.
- STL – Steals
  The more you Steal the less the opponent scores.

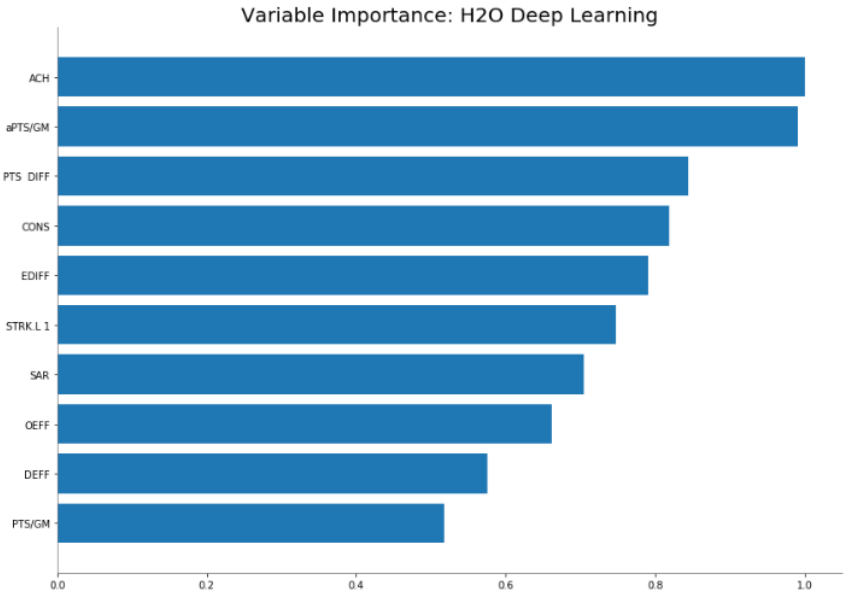

Fig. 11. Variable importance 2018-19 Advanced NBA stats

According to H2O the most important variable effectin Wins are:

- ACH - Achievement Level In Terms of Wins
  The metric is based on differential between actual and expected winning percentage. Positive figures indicate overachievement while negative figures indicate the team should have won more games.
- aPTS/GM - Points Allowed Per Game Average points allowed per game
- PTS DIFF - Points Differential
  {(Total Points Scored) - (Total Points Allowed)}/ (Games Played)

7 Comparison and Results

We are comparing our models on the basis of two parameters MSE and Error Percentage:

Mean Squared Error:
the mean squared error (MSE) or mean squared deviation (MSD) of an estimator (of a procedure for estimating an unobserved quantity) measures the average of the squares of the errors—that is, the average squared difference between the estimated values and what is estimated. MSE is a risk function, corresponding to the expected value of the squared error loss. The fact that MSE is almost always strictly positive (and not zero) is because of randomness or because the estimator does not account for information that could produce a more accurate estimate. (Mean squared error, n.d.)
If a vector of *n* predictions generated from a sample of *n* data points on all variables, and *Y* is the vector of observed values of the variable being predicted, then the within-sample MSE of the predictor is computed as:

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

Error Percentage:

The mean absolute percentage error (MAPE), also known as mean absolute percentage deviation (MAPD), is a measure of prediction accuracy of a forecasting method in statistics, for example in trend estimation, also used as a loss function for regression problems in machine learning. It usually expresses accuracy as a percentage, and is defined by the formula:

$$\mathrm{M} = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

where *At* is the actual value and *Ft* is the forecast value. The difference between *At* and *Ft* is divided by the actual value *At* again. The absolute value in this calculation is summed for every forecasted point in time and divided by the number of fitted points *n*. Multiplying by 100% makes it a percentage error.

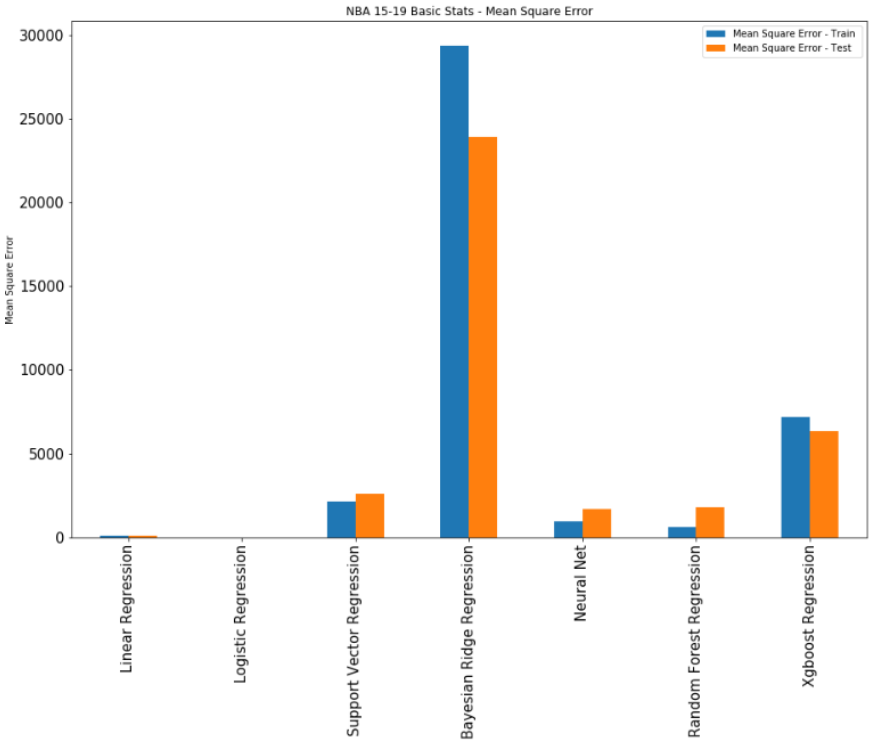

Fig. 12. Mean Square Error NBA 2015-19 Basic Stats

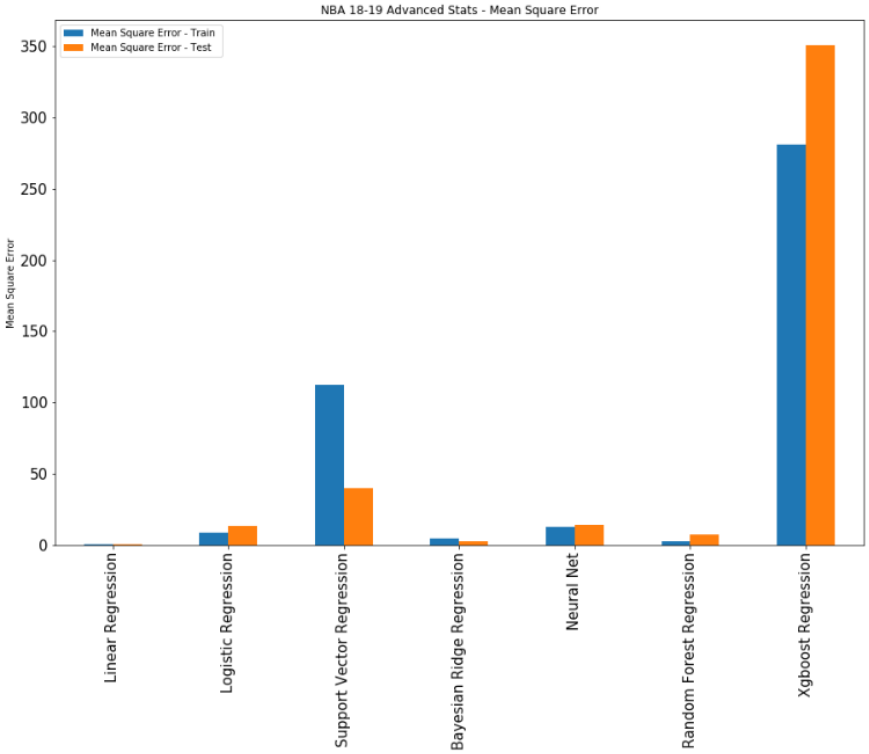

Fig. 13. Mean Square Error NBA 2018-19 Advanced Stats

We can see that Logistic Regression performes the best on the NBA 2015-2019 Basic set

and similarly Linear Regression performes the best on NBA 2018-19 Advanced set.

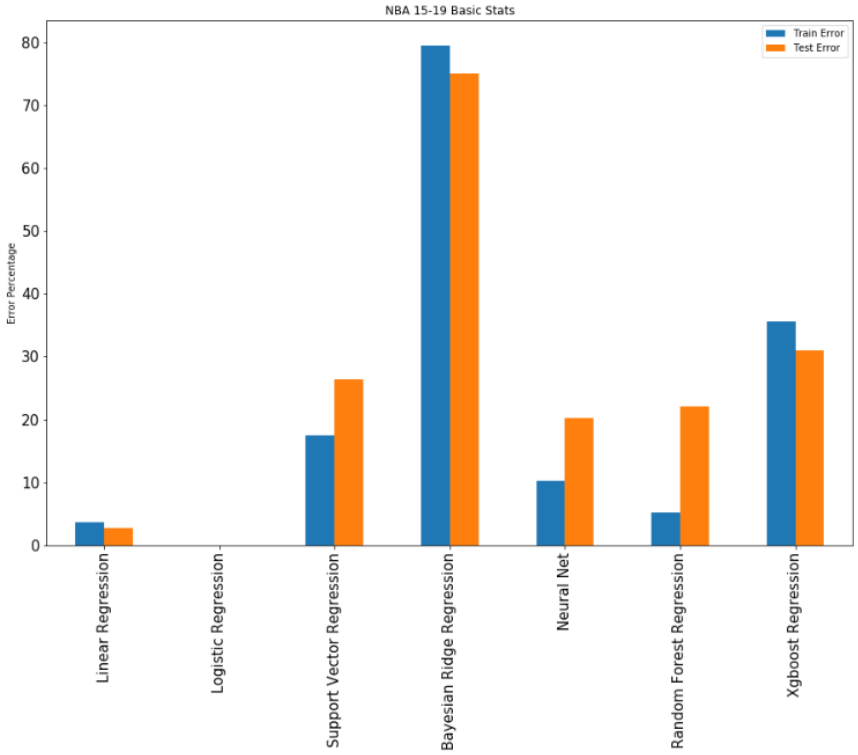

Fig. 14. Error percentage NBA 2015-19 Basic Stats

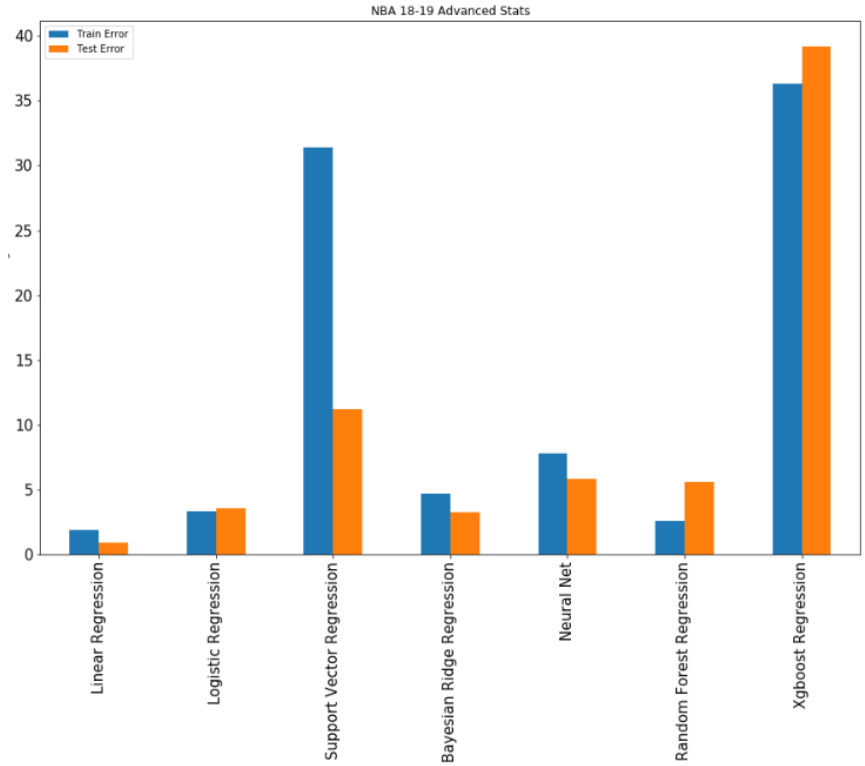

Fig. 15. Error percentage NBA 2018-19 Advanced Stats

Similarly, for Error percentage we can see that Logistic Regression performes the best on the NBA 2015-2019 Basic set and similarly Linear Regression performes the best on NBA 2018-19 Advanced set.

Therefore, we can say that Linear Regression is the best model for our 2018-19 dataset and Logistic Regression is the best model for our 2015-19 dataset.

Predictions

	Teams	Wins
1	Golden State Warriors	80.0
2	San Antonio Spurs	70.0
3	Toronto Raptors	68.0
4	Houston Rockets	68.0
5	LA Clippers	62.0
6	Oklahoma City Thunders	61.0
7	Boston Celtics	61.0
8	Portland Trail Blazers	60.0
9	Cleveland Cavaliers	58.0
10	Utah Jazz	57.0
11	Milwaukee Bucks	55.0
12	Indiana Pacers	55.0
13	Washington Wizards	53.0
14	Miami Heat	52.0
15	Denver Nuggets	51.0
16	Atlanta Hawks	51.0
17	Memphis Grizzlies	49.0
18	New Orleans Pelicans	48.0
19	Detroit Pistons	48.0
20	Charlotte Hornets	48.0
21	Dallas Mavericks	46.0
22	Chicago Bulls	46.0
23	Sacramento Kings	40.0
24	Philadelphia 76ers	40.0
25	Minnesota Timberwolves	40.0
26	Orlando Magic	39.0
27	Brooklyn Nets	37.0
28	Los Angeles Lakers	34.0
29	Phoenix Suns	32.0
30	New York Knicks	32.0

	Teams	Wins
1	Milwaukee Bucks	59.0
2	Golden State Warriors	59.0
3	Toronto Raptors	57.0
4	Denver Nuggets	54.0
5	Portland Trail Blazers	53.0
6	Houston Rockets	53.0
7	Utah Jazz	51.0
8	Philadelphia 76ers	50.0
9	Oklahoma City Thunder	49.0
10	Boston Celtics	49.0
11	San Antonio Spurs	48.0
12	LA Clippers	48.0
13	Indiana Pacers	48.0
14	Orlando Magic	42.0
15	Brooklyn Nets	42.0
16	Detroit Pistons	41.0
17	Sacramento Kings	39.0
18	Miami Heat	39.0
19	Charlotte Hornets	39.0
20	Los Angeles Lakers	37.0
21	Minnesota Timberwolves	36.0
22	Memphis Grizzlies	35.0
23	Washington Wizards	33.0
24	New Orleans Peicans	33.0
25	Dallas Mavericks	33.0
26	Atlanta Hawks	29.0
27	Chicago Bulls	20.0
28	Phoenix Suns	19.0
29	New York Knicks	19.0
30	Cleveland Cavaliers	18.0

Table. 1a & 1b. Prediction of Linear Regression and Logistic Regression respectively

Combining both of the tables and averaging out the prediction we get the following final predictions.

	Teams	Wins
1	Golden State Warriors	70.0
2	Toronto Raptors	62.0
3	Houston Rockets	60.0
4	San Antonio Spurs	59.0
5	Milwaukee Bucks	57.0
6	Portland Trail Blazers	56.0
7	Oklahoma City Thunder	55.0
8	LA Clippers	55.0
9	Boston Celtics	55.0
10	Utah Jazz	54.0
11	Indiana Pacers	52.0
12	Denver Nuggets	52.0
13	Miami Heat	46.0
14	Philadelphia 76ers	45.0
15	Detroit Pistons	44.0
16	Charlotte Hornets	44.0
17	Washington Wizards	43.0
18	Memphis Grizzlies	42.0
19	Sacramento Kings	40.0
20	Orlando Magic	40.0
21	New Orleans Pelicans	40.0
22	Dallas Mavericks	40.0
23	Brooklyn Nets	40.0
24	Atlanta Hawks	40.0
25	Minnesota Timberwolves	38.0
26	Cleveland Cavaliers	38.0
27	Los Angeles Lakers	36.0
28	Chicago Bulls	33.0
29	Phoenix Suns	26.0
30	New York Knicks	26.0

Table. 2. Prediction of Linear Regression and Logistic Regression combined

References

Fantasy basketball models/research papers:

http://www.sloansportsconference.com/wp-content/uploads/2014/06/DraftKings.pdf

https://playbook.draftkings.com/uncategorized/estimating-value-for-nba-on-draftkings

https://www.teamrankings.com/blog/nfl/fanduel-strategy-scoring-by-position

https://github.com/BenBrostoff/draft-kings-fun

http://datashoptalk.com/double-yo-money/

http://cs229.stanford.edu/proj2015/104_report.pdf

http://dionny.github.io/NBAPredictions/website/

https://github.com/nikbearbrown/CSYE_7245/blob/master/Projects/Research_Topic_Samples/Topic_Example_Predicting_NBA_games_using_machine_learning.pdf

http://homepages.cae.wisc.edu/~ece539/fall13/project/AmorimTorres_rpt.pdf

https://www.mbeckler.org/coursework/2008-2009/10701_report.pdf

http://www.units.miamioh.edu/sumsri/sumj/2003/NBAstats.pdf

http://cs229.stanford.edu/proj2017/final-reports/5231214.pdf

Data sources:

http://stats.nba.com/stats/

https://www.basketball-reference.com/

http://www.espn.com/nba/

Artificial neural network. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/Artificial_neural_network adresinden alındı

Bayesian multivariate linear regression. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/Bayesian_multivariate_linear_regression adresinden alındı

H2O. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/H2O_(software) adresinden alındı

Linear Regression. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/Linear_regression adresinden alındı

Logistic Regression. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/Logistic_regression adresinden alındı

Mean squared error. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/Mean_squared_error adresinden alındı

Pathak, M. (2018, July 10). *Using XGBoost in Python*. DataCap: https://www.datacamp.com/community/tutorials/xgboost-in-python adresinden alındı

Random forest. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/Random_forest adresinden alındı

Support-vector Machine. (tarih yok). Wikipedia: https://en.wikipedia.org/wiki/Support-vector_machine#Motivation adresinden alındı