**Capstone Project 2: Final Report**

**Problem Statement:**

The question this project will attempt to answer is: What NBA player statistics contribute the most towards determining what an NBA player’s points per game will be in the following season? This is a useful question to answer because points per game is one of if not the most important player statistic in the NBA & shows how much a player is contributing towards a team’s score & in turn a team’s chances of winning as well. Being able to predict a player’s points per game in the upcoming season will also provide insight into if a player is rising or declining offensively.

Potential clients who would be interested in this project include NBA players who want to see which aspects of their game they need to work on & improve in order to increase their points per game average. Increasing their points per game average will usually result in much higher salaries for the players when they sign new contracts. Another client would be NBA teams & executives who can use the insights from this project to make decisions about which players they should invest in order to win more games & possibly a championship.

**Description of Dataset:**

Training Data: <https://www.kaggle.com/drgilermo/nba-players-stats?select=Seasons_Stats.csv>

Testing Data: <https://www.basketball-reference.com/>

The training dataset used for this project was obtained from the above Kaggle link. It contains various player statistics from every NBA season spanning from 1950-2017. The final training dataset used has 8,163 rows of data with 24 variables each. The testing dataset being used was pulled together from the above website where NBA team & player statistics can be found for every NBA season. I extracted player statistics from the 2017-2020 seasons to serve as the testing set. The final testing dataset used has 918 rows of data with the same 24 variables each.

A list & description of the final variables used within the machine learning models that were created:

Age: Age of the player at the start of the season

G (Games): The number of games a player played in that season

GS (Games Started): The number of games a player started in that season

USG% (Usage %):

Estimate of the percentage of a team’s plays used by a player while playing

OBPM (Offensive Box Plus/Minus):

Estimate of the offensive points per 100 possessions a player contributed above a

league-average player

DBPM (Defensive Box Plus/Minus):

Estimate of the defensive points per 100 possessions a player contributed above a league-average player

Minutes\_Per\_Game: The average number of minutes per game a player played

FG\_Per\_Game: The average number of field goals a player made per game

FGA\_Per\_Game: The average number of field goals a player attempted per game

3P\_Per\_Game: The average number of three pointers a player made per game

3PA\_Per\_Game: The average number of three pointers a player attempted per game

2P\_Per\_Game: The average number of two pointers a player made per game

2PA\_Per\_Game: The average number of two pointers a player attempted per game

FT\_Per\_Game: The average number of free throws a player made per game

FTA\_Per\_Game: The average number of free throws a player attempted per game

ORB\_Per\_Game: The average number of offensive rebounds a player had per game

DRB\_Per\_Game: The average number of defensive rebounds a player had per game

AST\_Per\_Game: The average number of assists a player had per game

STL\_Per\_Game: The average number of steals a player had per game

BLK\_Per\_Game: The average number of blocks a player had per game

TOV\_Per\_Game: The average number of turnovers a player had per game

PF\_Per\_Game: The average number of personal fouls a player had per game

PTS\_Per\_Game: The average number of points a player had per game that season

PTS\_Per\_Game\_Next\_Season:

The average number of points a player had per game the following season

**Data Wrangling:**

After importing the original training data into a pandas dataframe, I first condensed the dataset to include only player statistics starting from the 1985 NBA season. I believed including seasons prior to that would decrease the predictive power of the models I would create as prior seasons are different from the modern NBA & many prior seasons do not have values for all the statistics I wanted to include. Next, I converted statistics that were recorded as totals over the entire season to per game averages by dividing the season total by the number of games a player played in that season. Converting to per game averages eliminates the skewness that would be caused by games played as players play in a different number of games from each other in every season. Since I wanted to predict the points per game average a player will have in the following season, I had to create a PTS\_Per\_Game\_Next\_Season variable & bring in a player’s PTS\_Per\_Game average from the following season as the value for this variable. Grouping data by player & year & then shifting PTS\_Per\_Game values in a separate dataframe had to be done to make this happen. A player’s last season would not have a value in this column, all rows with a missing PTS\_Per\_Game\_Next\_Season value were removed. This includes all rows from the 2017 season in the training data too. I also decided to remove all rows in which a player did not play in at least 40 games in a season or on average in 15 minutes per game. This was done to remove the effect players who did not play much at all during a season would have on models. Additional wrangling steps taken before implementing machine learning models included dropping non-statistical variables such as Tm (team) & Pos (position). These categorical variables change so many times for a player even within the same season that I decided to focus on only the statistics of a player to determine PTS\_Per\_Game\_Next\_Season. Statistics & more advanced measures that are calculated from values of other simpler measures already included within the dataset & being used were removed as well. Examples of these statistics & advanced measures include BPM, TRB\_Per\_Game, FG%, 3P%, 2P%, FT%, ORB%, DRB%, TRB%, AST%, STL%, BLK%, TOV%, TS%, 3PAr, FTr, eFG%, PER, VORP & all win share related statistics. The information provided by these advanced statistics is already being captured by specific combinations of other variables & it would be redundant to include them especially when they also are not directly related to points. Similar wrangling steps to the ones just listed were taken to structure the testing dataset to be in a form compatible with the training dataset.

**Initial Findings & EDA:**

Looking into the training dataset, the number of player records included per year ranges from 193 records from 1999 to 330 records from 2015. The PTS\_Per\_Game\_Next\_Season values range from 9.83 in 1998 to 12.22 in 1989. The values have minimal up & down variations between 1985 & 2016 which shows that consecutive years do not have a strong effect on PTS\_Per\_Game\_Next\_Season. This was the reasoning behind not performing a walk forward cross validation model based on year later on as it would not necessarily result in a stronger model. Summary statistics for each of the numeric attributes can be seen within the code.

A correlation matrix as well as scatterplots (please see code for plots) were created to be able to see the strength & direction of the correlation each of the predictor variables has with PTS\_Per\_Game\_Next\_Season. The correlation matrix findings are summarized below (correlation coefficient values provided in ()):

Negligible Negative Correlation (-0.39 - 0):

Age (-0.19)

DBPM (-.02)

Negligible Positive Correlation (0 - 0.39):

G (0.18)

3P\_Per\_Game (0.25)

3PA\_Per\_Game (0.26)

ORB\_Per\_Game (0.20)

DRB\_Per\_Game (0.38)

AST\_Per\_Game (0.39)

BLK\_Per\_Game (0.15)

PF\_Per\_Game (0.22)

Slight Positive Correlation (0.40 - 0.69)

GS (0.48)

OBPM (0.63)

Minutes\_Per\_Game (0.69)

STL\_Per\_Game (0.46)

TOV\_Per\_Game (0.66)

Strong Positive Correlation (0.70 - 1)

USG% (0.7)

FG\_Per\_Game (0.82)

FGA\_Per\_Game (0.81)

2P\_Per\_Game (0.76)

2PA\_Per\_Game (0.75)

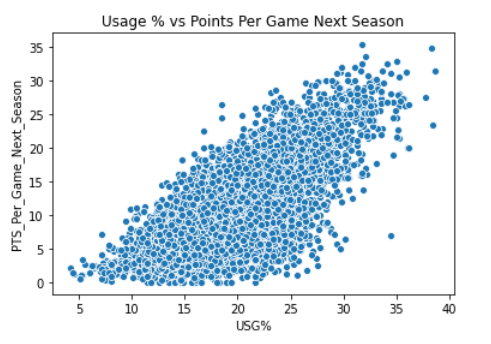
FT\_Per\_Game (0.75)

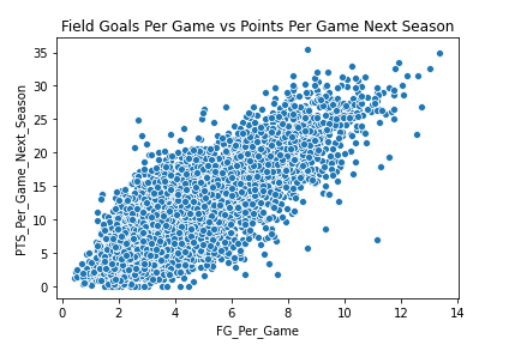
FTA\_Per\_Game (0.74)

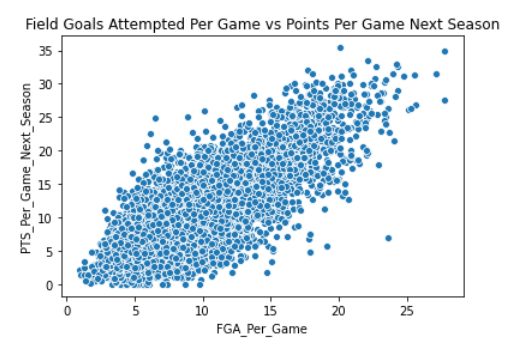
PTS\_Per\_Game (0.84)

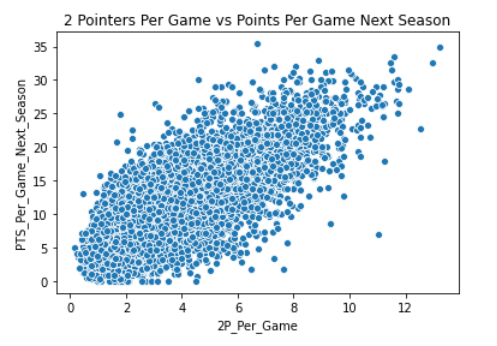
Histograms were also created to view the distributions & frequencies of the numeric variables (please see code for histograms).

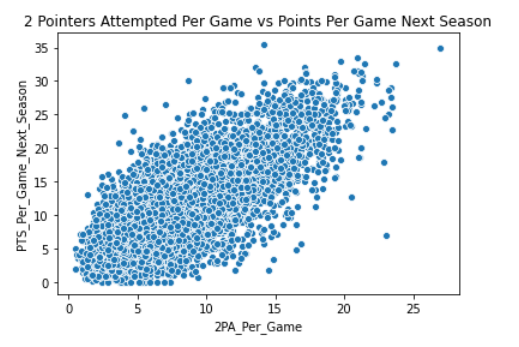
Scatterplots of variables strongly correlated with PTS\_Per\_Game\_Next\_Season:

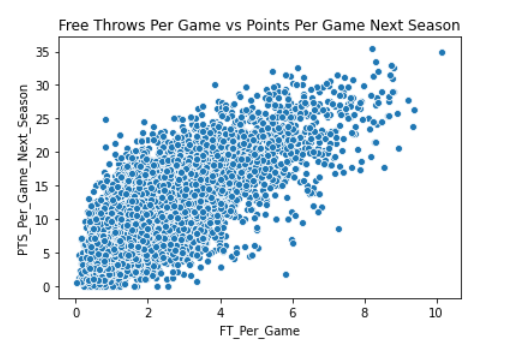


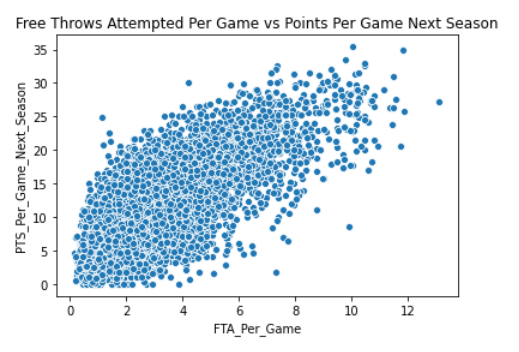


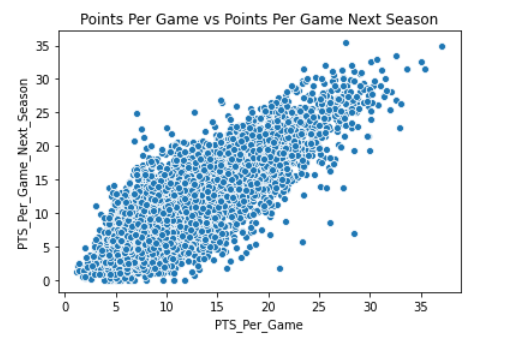












**In-Depth Analysis Using Machine Learning:**

I started my machine learning model building to predict PTS\_Per\_Game\_Next\_Season with linear regression models. My first model was a standard linear regression model which resulted in a R^2 score of 74.94% on the testing set & an RMSE of 3.05 points. The variables with the highest magnitude coefficient values in the standard linear regression model were FT\_Per\_Game (0.90), 2P\_Per\_Game (0.55), & FG\_Per\_Game (0.54).

The lasso regression model (best alpha value found to be an alpha of .0001) had a very similar R^2 score (74.96%) & RMSE value (3.05 points) as the standard linear regression model but different coefficient values for the predictor variables. FG\_Per\_Game (1.15) & FT\_Per\_Game (0.71) have even larger coefficients in this model & many variables had their coefficient value shrunk to 0. These variables included BLK\_Per\_Game, AST\_Per\_Game, FTA\_Per\_Game, 2PA\_Per\_Game, 2P\_Per\_Game, Minutes\_Per\_Game, & 3P\_Per\_Game.

The ridge regression model (best alpha value found to be an alpha of .01) also had a similar R^2 score (74.67%) & RMSE value (3.07 points). The coefficient values for the variables in the ridge regression model were similar to their coefficient values in the standard linear regression model. FT\_Per\_Game (0.64), FG\_Per\_Game (0.45), & 2P\_Per\_Game (0.44) were the variables with the highest magnitude coefficient values in the ridge regression model.

After the linear regression models, random forest & gradient boosting models were created to see how they would perform in predicting PTS\_Per\_Game\_Next\_Season. The random forest model had a R^2 score of 73.88% & an RMSE value of 3.11 points on the testing set which shows this model does not perform as well as the linear regression models. PTS\_Per\_Game was the most important feature within the random forest model.

Out of all the models that were implemented, the gradient boosting model was the best performing model. The gradient boosting model had an R^2 score of 75.32% & an RMSE value of 3.03 points when predicting PTS\_Per\_Game\_Next season with the testing set. PTS\_Per\_Game was by far the most important feature within the gradient boosting model as well (feature importance of 0.90).

The final models created to predict PTS\_Per\_Game\_Next\_Season were support vector regression & knn models. The support vector regression model (with a linear kernel) performed similar to the gradient boosting model. It had an R^2 score of 75.28% & an RMSE value of 3.03 points. The knn model was not as accurate in determining PTS\_Per\_Game\_Next Season. The knn model had a lower R^2 score (70.38%) & a higher RMSE value (3.32 points) than all of the other models.

A summary of how these models performed can be seen in the table below. Please look at code to see coefficient values or feature importance values for all the variables included.

Testing the best performing model, the gradient boosting model, on the most recent NBA season (using 2019 statistics to predict 2020 points per game), resulted in an R^2 score of 77.45% & an RMSE of 3.01 points. Overall, previous season statistics explain a high % of the variation in the points per game average a player will have in the following season & the models created can predict that points per game average fairly accurately.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Regression Type** | **Training R^2** | **Training RMSE** | **Testing R^2** | **Testing RMSE** |
| Linear Regression | 73.51% | 3.11 | 74.94% | 3.05 |
| Lasso Regression | 73.48% | 3.11 | 74.96% | 3.05 |
| Ridge Regression | 73.49% | 3.11 | 74.67% | 3.07 |
| Random Forest | 73.62% | 3.10 | 73.88% | 3.11 |
| Gradient Boosting | 74.87% | 3.03 | 75.32% | 3.03 |
| Support Vector Regression | 73.39% | 3.12 | 75.28% | 3.03 |
| KNN | 72.33% | 3.18 | 70.38% | 3.32 |

**Future Work:**

Possible future work &/or extensions of this project could include seeing the effect specific team’s & coaches have on a player’s points per game. Another idea could be to see what factors correlate to the number of wins/win shares a player is responsible for in a season.