Multivariate Final Project Proposal: NBA Champions and Runners-Up

Jack Kennedy

Becca Hittner

Dustin Casey

April 15, 2018

**Abstract**

The National Basketball Association (NBA) is an American professional basketball league comprised of thirty teams competing in two conferences and six divisions. The top team from each division as well as five teams with the next best record from each conference compete in the NBA Playoffs for the championship. Teams are seeded according to their record and play in best of seven games series. The NBA Championship is played after the winners of both conference championships have been decided. The result of this project will help fans, basketball teams, and betting agencies better predict the outcome of the NBA Championship based on past NBA Champions and their respective “runners-up”. In our project we will use SAS to create prediction models for the NBA season and playoffs. In our models we will mainly use regression to see correlation between various statistics and games won. Regression analysis is a tool that allows us to see how an independent variable is related to the dependant variable. This can lead to casual relationships between these variables. In certain cases it is possible for multiple regression to occur where two or more independent variables work together to form a relationship with the dependant variable. In our project we will test for multiple regression often as we have 25 variables to help us determine who the NBA champion and “runner-up” will be. Our data does not have any missing values so it will be easy for us to be able to paint a complete picture of all thirty teams that we will be analyzing. Our data is taken from the year 1980 through 2017 and it is our hope to be able to predict Game outcomes, Points Scored, and Champions.

**Table of Contents**

[**Introduction**](#_yk3jjcc9n44v) **3**

[**Data**](#_nj58omdc0glw) **4**

[**Predicting Game Outcomes**](#_olh1f9c6g7y1) **5**

[**Predicting Points Scored**](#_m6pr86am5jsa) **9**

[**Predicting Champions**](#_paq8k0a54nj2) **13**

[**Conclusions**](#_jh80mb1zi2qa) **15**

[**Appendix**](#_3v3u35zbpqqb) **16**

[**References**](#_ncos8sp45bs9) **24**

# 

# Introduction

In this project, we attempt to predict the game outcomes, points scored, and champions in the NBA Finals over the course of 37 years, from 1980-2017. Our data includes a variety of team statistics from the games in the Finals series, both from the champions and runners-up. The statistics included shooting data (field goal percentage, free throw percentage, three-point percentage, etc.), as well as non-shooting percentage (defensive rebounds, offensive rebounds, turnovers, etc.). Our hypotheses were that there would be relationships between the statistics and the game outcomes, points scored, and who came out on top as NBA Champions. First, the Data section, we describe the dataset we used in conducting this analysis. In the Predicting Game Outcomes section, we attempt to predict wins and losses in each game using a model with all statistics and as well as one with only non-shooting statistics. In the Predicting Points Scored, we followed the same steps, with first generating a model using all variables except field goals made, three pointers made, and free throws made. We explored the same prediction using only non-shooting stats. We also include a time series graph of offensive statistics among both champions and runners-up. In the Predicting Champions, we look to predict who comes in second and who would take home the trophy. In the last section, we discuss our findings and explain our conclusions.

# 

# 

# 

# Data

The dataset we are using consists of two Comma Separated Values (CSV) spreadsheets, both consisting of 216 records (rows) and 25 variables (columns). The total number of records (rows) for our project will be 432. Each record (row) represents each game’s stats for both the champions and the runners-up every year between 1980 and 2017. A brief description/explanation of each variable (column) in the data set can be found below. There are no missing values in the dataset.



# Predicting Game Outcomes

Our first objective of this project was to predict game outcomes (wins and losses) based on the team’s stats for the game. We used logistic regression to predict the probability of a team winning. From there, we used a manually set threshold of 0.5 to assign predictions to the records. If a team had a probability of 0.5 or higher, we assigned a PredictedWin value of 1 (win) to the record and vice versa of 0 for a loss. The two models we generated used a training set named “game\_training” to create the model and a testing set named “game\_testing” to test the fit of the model and see how accurate the model was at predicting game outcomes. The training and testing sets had a 75-25 split, with 75% of the available data assigned to the training set and 25% of the data withheld for testing.

**Model I:**

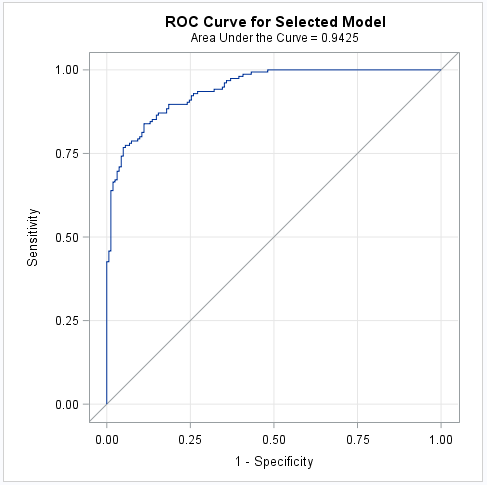
Our first model we generated used all of the columns to predict the game outcomes. We used a stepwise selection to remove and add the most significant variables. The final variables in the model were Field Goal Percentage (FGP), Defensive Rebounds (DRB), Turnovers (TOV), Personal Fouls (PF), Free Throws Made (FT), Offensive Rebounds (ORB), Three Point Percentage (TPP), Steals (STL), and Field Goals Attempted (FGA). Their p-values and point estimates are below:

# 

Based on this information, the model generated to predict for game outcome (Win) is as follows:

**Win = -0.18FGA + 41.85FGP + 4.03TPP + 0.073FT + 0.46ORB + 0.38DRB + 0.44STL – 0.39TOV – 0.16PF – 18.52**

The model has a Chi-Square value of 10.85, 7 degrees of freedom, and a p-value 0.1453. The Area Under the Curve for the model is 0.9425 or 94.25%.



This model was 86.79% accurate at predicting game outcomes on our test set of 106 records.

**Model II:**

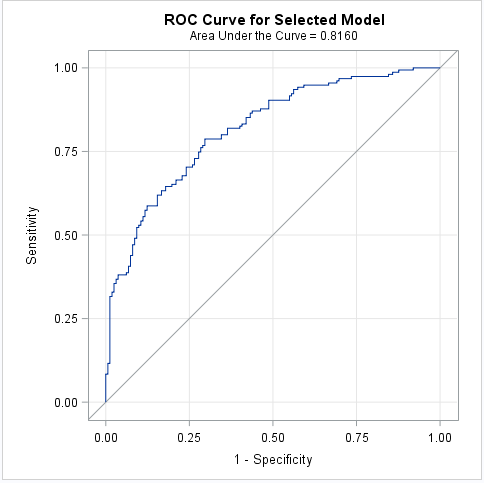
For our second model we generated, we wanted to see how non-shooting stats (excluding variables for shooting such as FG, TP, FT, etc.) affected our predictions and how accurately the model could predict game outcomes. We used a stepwise selection to remove and add the most significant variables. The final variables in the model were Defensive Rebounds (DRB), Personal Fouls (PF), Assists (AST), Turnovers (TOV), and Steals (STL). Their p-values and point estimates are below:

# 

Based on this information, the model generated to predict for game outcome (Win) is as follows:

**Win = 0.19DRB + 0.11AST + 0.16STL – 0.16TOV – 0.13PF – 3.94**

The model has a Chi-Square value of 1.0215, 2 degrees of freedom, and a p-value 0.6. The Area Under the Curve for the model is 0.8160 or 81.6%.



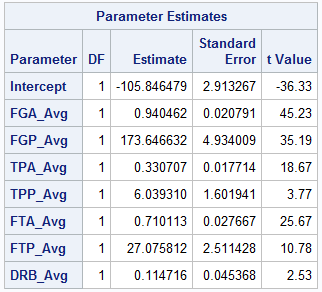
This model was 71.7% accurate at predicting the correct game outcome on our test set of 106 records.

# Predicting Points Scored

Our next objective of this project was to predict the average points scored (PTS\_Avg) based on the team’s average stats for the series. We used multiple regression to predict the average points scored. From there, we used the residual values (actual results minus predicted results) to test the accuracy of the model. The two models we generated used a training set named “training” to create the model and a testing set named “testing” to test the fit of the model and see how accurate the model was at predicting game outcomes. The training and testing sets had a 75-25 split, with 75% of the available data assigned to the training set and 25% of the data withheld for testing.

**Model I:**

Our first model we generated used all of the columns with the exception of Field Goals Made (FG\_Avg), Free Throws Made (FT\_Avg), and Three Point Shots Made (TP\_Avg) to predict the average points scored. The reason for this is if these variables are included in the model, the model will be 100% accurate as Points are directly derived from these three statistics. We used a stepwise selection to remove and add the most significant variables. The final variables in the model were Free Throw Attempts (FTA\_Avg), Field Goal Attempts (FGA\_Avg), Field Goal Percentage (FGP\_Avg), Three Point Attempts (TPA\_Avg), Free Throw Percentage (FTP\_Avg), Three Point Percentage (TPP\_Avg), and Defensive Rebounds (DRB\_Avg). Their t-values and point estimates are below:

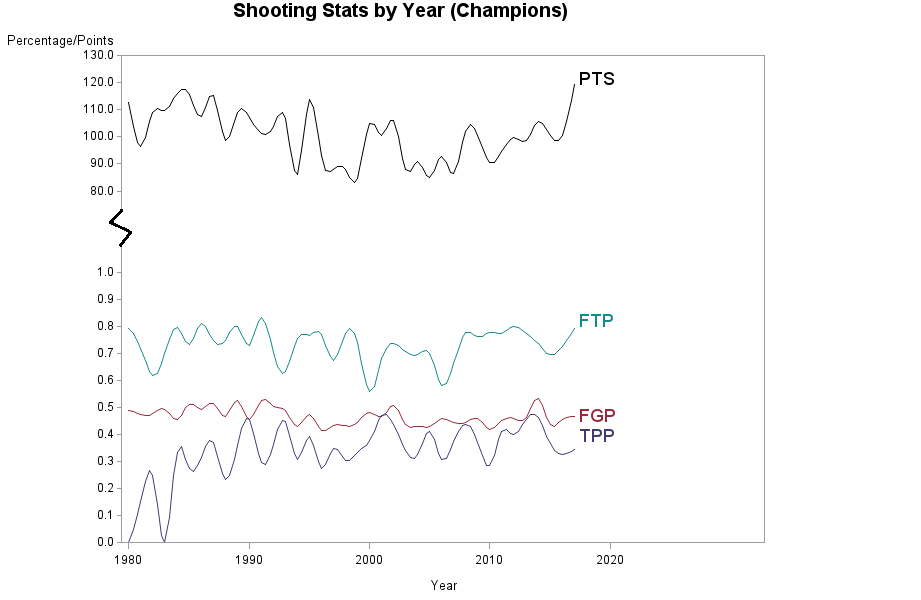


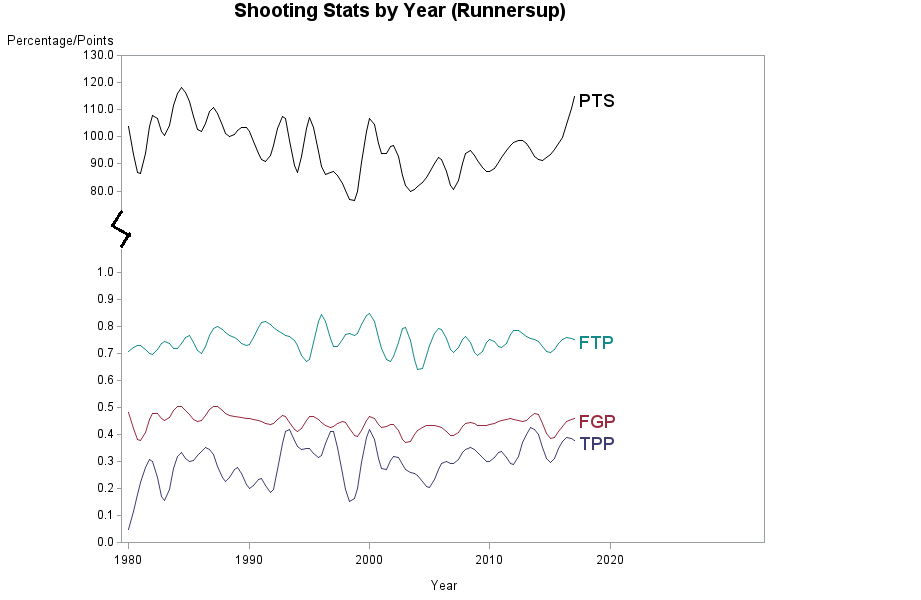
Based on this information, the model generated to predict for average points scored (PTS\_Avg) is as follows:

**PTS\_Avg = 0.94FGA\_Avg + 173.65FGP\_Avg + 0.33TPA\_Avg + 6.04TPP\_Avg + 0.71FTA\_Avg + 27.08FTP\_Avg + 0.11DRB\_Avg – 105.85**

The model has a R-Squared value of 0.9932 or 99.32% and an Adjusted R-Squared value of 0.9922 or 99.22%. The average residual value of the model is 0.1745.

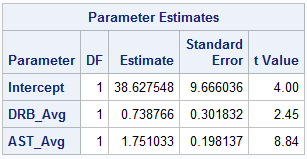
We also graphed a time series of champion and runnerups series averages to see the correlation of average points per game compared to shooting percentages for Field Goal Percentage (FGP), Free Throw Percentage (FTP), and Three Point Percentage (TPP).





**Model II:**

For our second model we generated, we wanted to see how non-shooting stats (excluding variables for shooting such as FGA, FGP, TPA, TPP, FTA, FTP etc.) affected our predictions and how accurately the model could predict average points scored. We used a stepwise selection to remove and add the most significant variables. The final variables in the model were Assists (AST\_Avg) and Defensive Rebounds (DRB\_Avg). Their t-values and point estimates are below:

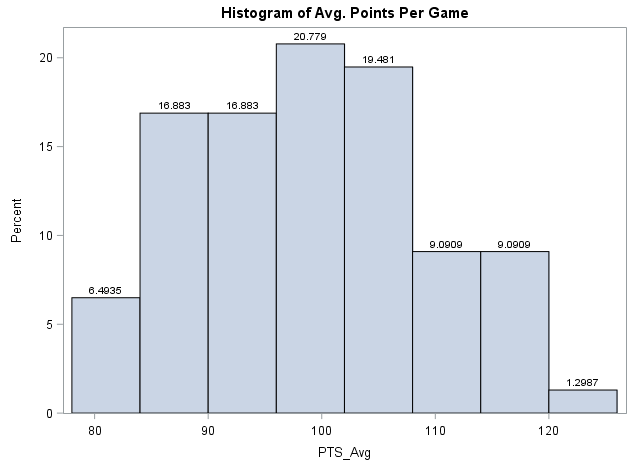


Based on this information, the model generated to predict for average points scored (PTS\_Avg) is as follows:

**PTS\_Avg = 0.739DRB\_Avg + 1.751 + 38.628**

The model has a R-Squared value of 0.6190 or 61.9% and an Adjusted R-Squared value of 0.6049 or 60.49%. The average residual value of the model is 0.5814.

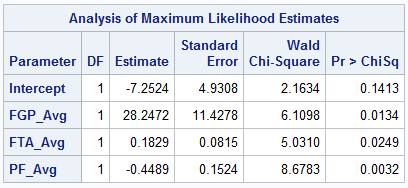
Along this the models above, we’ve created a histogram of average points scored to get an idea of how many teams fall into different average points groupings. It can be found below:

****

# Predicting Champions

Our last objective of this project was to predict the series champion (Champions) based on the team’s average stats for the series. We used logistic regression to predict the probability of a team being the series champions (ProbabilityOfChamp). From there, we used a manually set threshold of 0.36 to assign predictions to the records. If a team had a probability of 0.36 or higher, we assigned a PredictedChamp value of 1 (champion) to the record and vice versa of 0 for runnerup. The model we generated used a training set named “training” to create the model and a testing set named “testing” to test the fit of the model and see how accurate the model was at predicting the series champion. The training and testing sets had a 75-25 split, with 75% of the available data assigned to the training set and 25% of the data withheld for testing.

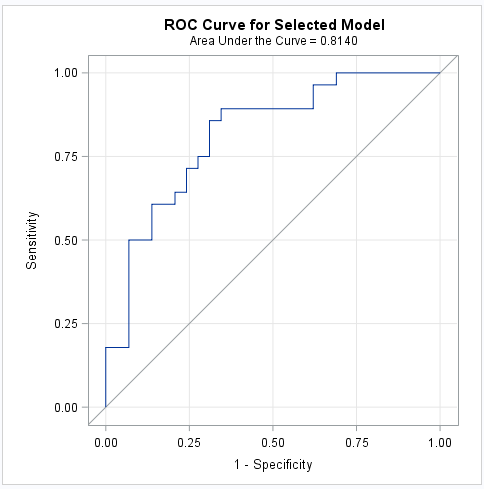
The model we generated used all of the columns to predict the game outcomes. We used a stepwise selection to remove and add the most significant variables. The final variables in the model were Field Goal Percentage (FGP\_Avg), Free Throws Attempts (FTA\_Avg), and Personal Fouls (PF\_Avg). Their p-values and point estimates are below:



Based on this information, the model generated to predict for the series champion (Champions) is as follows:

**Champions = 28.247FGP\_Avg + 0.183FTA\_Avg – 0.449PF\_Avg – 7.252**

The model has a Chi-Square value of 33.539, 13 degrees of freedom, and a p-value 0.0014. The Area Under the Curve for the model is 0.8140 or 81.4%.



This model was 65.0% accurate at predicting the series champions on our test set of 20 records.

# Conclusions

In this model we were able to build three separate prediction models: Game outcomes, points scored, and NBA Champion. Our data was from the years 1980-2017 and consisted of both shooting data (field goal percentage, free throw percentage, three-point percentage, etc.), as well as non-shooting percentage (defensive rebounds, offensive rebounds, turnovers, etc.). For the points scored prediction models we tested to see how shooting stats affected our prediction model versus how non shooting statistics affected our models. The model using shooting stats to predict points will be 100% accurate as Points are directly derived from these three statistics. Our model that used non shooting stats was able to predict points with 60% accuracy. For the game outcome prediction model we used logistic regression to help us determine with 86.79% accuracy game outcomes using the significant variable of Field Goal Percentage (FGP), Defensive Rebounds (DRB), Turnovers (TOV), Personal Fouls (PF), Free Throws Made (FT), Offensive Rebounds (ORB), Three Point Percentage (TPP), Steals (STL), and Field Goals Attempted (FGA). We then build a second model that was 71.7% accurate at predicting the correct game outcome on our test set of 106 records using non-shooting stats with significant variables of Defensive Rebounds (DRB), Personal Fouls (PF), Assists (AST), Turnovers (TOV), and Steals (STL). The third prediction model was built to predict champions. The most significant variables selected were Field Goal Percentage (FGP\_Avg), Free Throws Attempts (FTA\_Avg), and Personal Fouls (PF\_Avg). Our model was able to predict with 65% accuracy who the champion would be. It is not surprising to us that this model is predicted with the lowest accuracy because there are many other factors that occur throughout a season that leads to a team becoming the champions.

# 

# 

# Appendix

/\* Team Members: Jack Kennedy, Becca Hittner, Dustin Casey

Final Project Code

Date: 4/28/18

Program Description: Predicting Points, Game Outcomes, and NBA Champions using historical NBA Championship Series data.

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Data Cleansing \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\* Reading in Champion series data and the variables. Variable descriptions and types are included in Project Report.;

data champions;

infile "Z:\Students\100133984\STAT40\Final Project\championsdata.csv" dsd firstobs=1; \*Skipping the first line of data (data headers;

input X Year Team $ Game Win Home MP FG FGA FGP TP TPA TPP FT FTA FTP ORB DRB TRB AST STL BLK TOV PF PTS;

Champions = 1; \*Creating a new variable to represent if the team won the championship for the given year/series. 1 in the case represents champions;

run;

\* Reading in Runnerup series data and the variables. Variable descriptions and types are included in Project Report.;

data runnerups;

infile "Z:\Students\100133984\STAT40\Final Project\runnerupsdata.csv" dsd firstobs=1; \*Skipping the first line of data (data headers;

input X Year Team $ Game Win Home MP FG FGA FGP TP TPA TPP FT FTA FTP ORB DRB TRB AST STL BLK TOV PF PTS;

Champions = 0; \*Creating a new variable to represent if the team won the championship for the given year/series. 0 in the case represents runnerups;

run;

\* Creating a data table of the champions for each series/year for use in exploratory analysis;

data champs\_team\_year;

set champions;

where Game = 1;

run;

\* Creating a data table of the runnerups for each series/year for use in exploratory analysis;

data runnerups\_team\_year;

set runnerups;

where Game = 1;

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Data Aggregation \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\* Creating a data table of combined champion and runnerups series data;

data combined\_data;

set champions runnerups;

by X; \*X represents the game/year unique id;

where TPP <> .;

run;

\*Using Proc SQL, we are aggregating the series data for champions and saving it in a new table called "champ\_series\_avgs";

proc sql;

create table champ\_series\_avgs as

select Champions, Year, Team, avg(FG) as FG\_Avg, avg(FGA) as FGA\_Avg, avg(FGP) as FGP\_Avg,

avg(TP) as TP\_Avg, avg(TPA) as TPA\_Avg, avg(TPP) as TPP\_Avg, avg(FT) as FT\_Avg,

avg(FTA) as FTA\_Avg, avg(FTP) as FTP\_Avg, avg(ORB) as ORB\_Avg,

avg(DRB) as DRB\_Avg, avg(TRB) as TRB\_Avg, avg(DRB) as DRB\_Avg,

avg(AST) as AST\_Avg, avg(STL) as STL\_Avg, avg(BLK) as BLK\_Avg,

avg(TOV) as TOV\_Avg, avg(PF) as PF\_Avg, avg(PTS) as PTS\_Avg

from combined\_data

where Champions = 1

group by Champions, Year, Team;

run;

\*Using Proc SQL, we are aggregating the series data for runnerups and saving it in a new table called "runnerups\_series\_avgs";

proc sql;

create table runnerups\_series\_avgs as

select Champions, Year, Team, avg(FG) as FG\_Avg, avg(FGA) as FGA\_Avg, avg(FGP) as FGP\_Avg,

avg(TP) as TP\_Avg, avg(TPA) as TPA\_Avg, avg(TPP) as TPP\_Avg, avg(FT) as FT\_Avg,

avg(FTA) as FTA\_Avg, avg(FTP) as FTP\_Avg, avg(ORB) as ORB\_Avg,

avg(DRB) as DRB\_Avg, avg(TRB) as TRB\_Avg, avg(DRB) as DRB\_Avg,

avg(AST) as AST\_Avg, avg(STL) as STL\_Avg, avg(BLK) as BLK\_Avg,

avg(TOV) as TOV\_Avg, avg(PF) as PF\_Avg, avg(PTS) as PTS\_Avg

from combined\_data

where Champions = 0

group by Champions, Year, Team;

run;

\* Creating a combined series averages data set;

data series\_avgs;

set champ\_series\_avgs runnerups\_series\_avgs;

by Year;

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Data Summarization \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

\* Creating a new data table of series averages and assigning a random number ID to each record;

\* We do this so we can create a training and testing data set for predictive analysis on the series\_avgs data set;

data series\_avgs\_temp;

set series\_avgs;

n = ranuni(77); \* Assigning a random number ID to each record from 1 to 77;

proc sort data = series\_avgs\_temp; \*Sorting the new data set by the random number ID "n";

by n;

data training testing; \* Creating training and testing data sets with 75% of the records assigned to training and 25% of records assigned to testing;

set series\_avgs\_temp nobs = nobs;

if \_n\_ <= .75\*nobs then output training;

else output testing;

run;

\* Replicating the same steps as above but on the raw series/game data. These training and testing data sets will be used for predicting game outcomes and points scored;

data series\_temp;

set combined\_data;

n = ranuni(432);

proc sort data = series\_temp;

by n;

data game\_training game\_testing;

set series\_temp nobs = nobs;

if \_n\_ <= .75\*nobs then output game\_training;

else output game\_testing;

run;

\* Here, we begin doing regression analysis on our training sets;

ods graphics on;

\* The first model we create is a logistic regression modeling for Champions;

proc logistic data = training desc;

model Champions = FG\_Avg FGA\_Avg FGP\_Avg TP\_Avg TPA\_Avg TPP\_Avg

FT\_Avg FTA\_Avg FTP\_Avg ORB\_Avg DRB\_Avg TRB\_Avg AST\_Avg STL\_Avg

BLK\_Avg TOV\_Avg PF\_Avg PTS\_Avg / selection=stepwise ctable outroc=work.ROC1Data; \*We use a stepwise selection to find the important variables;

store LogisticModel1; \*Storing the Logistic Model as "LogisticModel1" for use on the testing set;

run;

\* The next model is a logistic regression predicting individual game outcomes (Win);

\* This model uses all of the variables in prediction;

proc logistic data = game\_training desc;

model Win = FG FGA FGP TP TPA TPP FT FTA FTP ORB DRB TRB AST

STL BLK TOV PF PTS / selection=stepwise ctable outroc=work.ROC2Data; \* Using stepwise selection again;

store LogisticModel2; \* Storing the Logistic Model as "LogisticModel2" for use on the game\_testing set;

run;

\* The next model is a logistic regression predicting individual game outcomes (Win);

\* This model uses excludes shooting statistics (FG, FGA, FGP, TP, TPA, TPP, FT, FTA, FTP) variables in prediction;

\* We wanted to see how accurately the model could predict game outcomes based on non-shooting stats and how much these variables influence game outcomes;

proc logistic data = game\_training desc;

model Win = ORB DRB TRB AST STL BLK TOV PF / selection=stepwise ctable outroc=work.ROC3Data;

store LogisticModel3; \* Storing the Logistic Model as "LogisticModel3" for use on the game\_testing set;

run;

\* The next model is a multiple regression predicting Average Points Scored in for the series (PTS\_Avg);

\* This model uses all of the variables in prediction except FG\_Avg, FT\_Avg, and TP\_Avg. We did this because we realized after testing the model ;

\* that if FG\_Avg, TP\_Avg, and FT\_Avg were included, the model will have 100% accuracy. This is because, if you know how many Field Goals,;

\* Free Throws, and Three Point shots were made, you would be able to add those all up and have the Points Scored;

proc glmselect data= training plots=all;

model PTS\_Avg = FGA\_Avg FGP\_Avg TPA\_Avg TPP\_Avg

FTA\_Avg FTP\_Avg ORB\_Avg DRB\_Avg TRB\_Avg AST\_Avg STL\_Avg

BLK\_Avg TOV\_Avg PF\_Avg / selection=stepwise;

store RegressionModel1; \* Storing the Regression Model as "RegressionModel1" for use on the testing set;

run;

\* The next model is a multiple regression predicting Average Points Scored in for the series (PTS\_Avg);

\* This model uses excludes shooting statistics (FG, FGA, FGP, TP, TPA, TPP, FT, FTA, FTP) variables in prediction;

\* We wanted to see how accurately the model could predict Avg. Points Scored based on non-shooting stats and how much these variables influence Avg. Points Scored;

proc glmselect data= training plots=all;

model PTS\_Avg = ORB\_Avg DRB\_Avg TRB\_Avg AST\_Avg STL\_Avg

BLK\_Avg TOV\_Avg PF\_Avg / selection=stepwise;

store RegressionModel2; \* Storing the Regression Model as "RegressionModel2" for use on the testing set;

run;

ods graphics off;

\* Testing LogisticModel1 on the Testing set to predict the ProbabilityOfChamp (the probability that a team would be champions);

\* After getting the Probabilities, we will create a probability threshold to predict 1 (the team won the championship);

\* or 0 (the team lost the series and is a runnerup). From there, we can get the accuracy of the model (did the team actually win or lose the series);

proc plm source=LogisticModel1;

score data=Testing out=work.TestL1Scored predicted = ProbabilityOfChamp / ilink; \* Outputing the results to table "TestL1Scored";

run;

\* Testing LogisticModel2 on the Game\_Testing set to predict the ProbabilityOfWin (the probability that a team would win the game);

\* After getting the Probabilities, we will create a probability threshold to predict 1 (the team won the game);

\* or 0 (the team lost the game). From there, we can get the accuracy of the model (did the team actually win or lose the game);

proc plm source=LogisticModel2;

score data=Game\_Testing out=work.TestL2Scored predicted = ProbabilityOfWin / ilink; \* Outputing the results to table "TestL2Scored";

run;

\* Testing LogisticModel3 on the Game\_Testing set to predict the ProbabilityOfWin (the probability that a team would win the game);

\* After getting the Probabilities, we will create a probability threshold to predict 1 (the team won the game);

\* or 0 (the team lost the game). From there, we can get the accuracy of the model (did the team actually win or lose the game);

proc plm source=LogisticModel3;

score data=Game\_Testing out=work.TestL3Scored predicted = ProbabilityOfWin / ilink; \* Outputing the results to table "TestL3Scored";

run;

\* Testing RegressionModel1 on the Testing set to predict the PredictedAvgPoints(the predicted Avg. Points Scored for the Series);

\* After getting the Predictions, we can get the accuracy of the model (how close were the predictions to the actually outcomes);

proc plm source=RegressionModel1;

score data=Testing out=work.TestR1Scored predicted = PredictedAvgPoints; \* Outputing the results to table "TestR1Scored";

run;

\* Testing RegressionModel2 on the Testing set to predict the PredictedAvgPoints(the predicted Avg. Points Scored for the Series);

\* After getting the Predictions, we can get the accuracy of the model (how close were the predictions to the actually outcomes);

proc plm source=RegressionModel2;

score data=Testing out=work.TestR2Scored predicted = PredictedAvgPoints; \* Outputing the results to table "TestR2Scored";

run;

\* Outputing scored test sets and looking at logistic regression accuracy;

data work.TestL1ScoredOutput;

set work.TestL1Scored (keep= Year Team Champions ProbabilityOfChamp);

label ProbabilityOfChamp = "Probability of Champion";

if ProbabilityOfChamp > .36 then PredictedChamp = 1;

else PredictedChamp = 0;

label PredictedChamp = "Predicted Champion";

if PredictedChamp = Champions then CorrectPrediction = 1;

else CorrectPrediction = 0;

run;

proc export data = TestL1ScoredOutput outfile='Z:\Students\100133984\STAT40\Final Project\TestL1ScoredOutput.csv'

dbms = csv replace;

run;

data work.TestL2ScoredOutput;

set work.TestL2Scored (keep= Year Team Game Win ProbabilityOfWin);

label ProbabilityOfWin = "Probability of Win";

if ProbabilityOfWin > .5 then PredictedWin = 1;

else PredictedWin = 0;

label PredictedWin = "Predicted Outcome";

if PredictedWin = Win then CorrectPrediction = 1;

else CorrectPrediction = 0;

run;

proc export data = TestL2ScoredOutput outfile='Z:\Students\100133984\STAT40\Final Project\TestL2ScoredOutput.csv'

dbms = csv replace;

run;

data work.TestL3ScoredOutput;

set work.TestL3Scored (keep= Year Team Game Win ProbabilityOfWin);

label ProbabilityOfWin = "Probability of Win";

if ProbabilityOfWin > .5 then PredictedWin = 1;

else PredictedWin = 0;

label PredictedWin = "Predicted Outcome";

if PredictedWin = Win then CorrectPrediction = 1;

else CorrectPrediction = 0;

run;

proc export data = TestL3ScoredOutput outfile='Z:\Students\100133984\STAT40\Final Project\TestL3ScoredOutput.csv'

dbms = csv replace;

run;

\* Outputing scored test sets and looking at multiple regression accuracy using Residual values;

data work.TestR1ScoredOutput;

set work.TestR1Scored (keep= Year Team PTS\_Avg PredictedAvgPoints);

label PredictedAvgPoints = "Predicted Avg. Points";

Residual = PTS\_Avg - PredictedAvgPoints;

run;

proc export data = TestR1ScoredOutput outfile='Z:\Students\100133984\STAT40\Final Project\TestR1ScoredOutput.csv'

dbms = csv replace;

run;

data work.TestR2ScoredOutput;

set work.TestR2Scored (keep= Year Team PTS\_Avg PredictedAvgPoints);

label PredictedAvgPoints = "Predicted Avg. Points";

Residual = PTS\_Avg - PredictedAvgPoints;

run;

proc export data = TestR2ScoredOutput outfile='Z:\Students\100133984\STAT40\Final Project\TestR2ScoredOutput.csv'

dbms = csv replace;

run;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Data Visualization \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

proc freq data = champs\_team\_year order = freq;

tables Team;

run;

proc gchart data = champs\_team\_year;

vbar Team / maxis = count;

run;

proc freq data = runnerups\_team\_year order = freq;

tables Team;

run;

proc gchart data = runnerups\_team\_year;

vbar Team / maxis = count;

run;

data linetext;

length function color $ 8 text $ 20 style $ 25;

retain function 'label' xsys ysys '2' hsys '3' position '6'

size 3;

set champ\_series\_avgs end=last;

if last then do;

x=Year;

y=FGP\_Avg; text=' FGP'; color='depk'; output;

y=FTP\_Avg; text=' FTP'; color='vibg'; output;

y=TPP\_Avg; text=' TPP'; color='mob'; output;

y=PTS\_Avg; text=' PTS'; color='black'; output;

end;

/\* Add a title to the graph \*/

title1 'Shooting Stats by Year (Champions)';

footnote1 height=2 angle=90 ' ';

/\* Create axis definitions \*/

axis1 minor=none offset=(1,22)

label=('Year');

axis2 order=(0 to 1 by .1, 50,60, 80 to 130 by 10) minor=none offset=(0,0)

label=('Percentage/Points' justify=right);

/\* Choose the interpolation method and assign symbol characteristics \*/

symbol1 interpol=spline width=1 color=depk;

symbol2 interpol=spline width=1 color=vibg;

symbol3 interpol=spline width=1 color=mob;

symbol4 interpol=spline width=1 color=black;

/\* Produce the plot \*/

proc gplot data=champ\_series\_avgs;

plot (FGP\_Avg FTP\_Avg TPP\_Avg PTS\_Avg)\*Year / overlay frame

haxis=axis1 vaxis=axis2

annotate=linetext;

run;

quit;

\* Replicating same shooting stats graphic for runnersup\_series\_avg;

data linetext;

length function color $ 8 text $ 20 style $ 25;

retain function 'label' xsys ysys '2' hsys '3' position '6'

size 3;

set runnerups\_series\_avgs end=last;

if last then do;

x=Year;

y=FGP\_Avg; text=' FGP'; color='depk'; output;

y=FTP\_Avg; text=' FTP'; color='vibg'; output;

y=TPP\_Avg; text=' TPP'; color='mob'; output;

y=PTS\_Avg; text=' PTS'; color='black'; output;

end;

/\* Add a title to the graph \*/

title1 'Shooting Stats by Year (Runnersup)';

footnote1 height=2 angle=90 ' ';

/\* Create axis definitions \*/

axis1 minor=none offset=(1,22)

label=('Year');

axis2 order=(0 to 1 by .1, 50,60, 80 to 130 by 10) minor=none offset=(0,0)

label=('Percentage/Points' justify=right);

/\* Choose the interpolation method and assign symbol characteristics \*/

symbol1 interpol=spline width=1 color=depk;

symbol2 interpol=spline width=1 color=vibg;

symbol3 interpol=spline width=1 color=mob;

symbol4 interpol=spline width=1 color=black;

/\* Produce the plot \*/

proc gplot data=runnerups\_series\_avgs;

plot (FGP\_Avg FTP\_Avg TPP\_Avg PTS\_Avg)\*Year / overlay frame

haxis=axis1 vaxis=axis2

annotate=linetext;

run;

quit;

proc sgplot data= series\_avgs;

histogram PTS\_Avg / datalabel = percent;

title 'Histogram of Avg. Points Per Game';

run;

# References

Dataset: <https://www.kaggle.com/daverosenman/nba-finals-team-stats>