*Division of Computing Science and Mathematics*
*Faculty of Natural Sciences*
*University of Stirling*

**Project Title**

# Using Machine Learning Model to Predict which NBA team has the highest chances of winning the NBA championships

**Author Name**

**Degnan Kopa**

**Dissertation submitted in partial fulfilment for the degree of Master of Science in Artificial Intelligence**

**September 2022**

# Abstract

Commented [JF1]: Should be more than 1000 words

The National Basketball Association is a widely known sports franchise in which there is only one team that will make the championship. From the 32 teams that are within the NBA, only one among those team will win the championship. The players of those team train themselves on a yearly basis to lead their home team into victory. This research is about predicting which of the 16 teams within the NBA playoffs have the highest chance of winning the NBA championship using the CRISP-DM workflow of machine learning models. The data in which we got our NBA team statistics is from the NBA website, we used 4 different seasons, and from the 4 seasons, we used the 2021-2022 NBA team statistic to do a comparison of the previous three seasons (2018-2019, 2019-2020, 2020-2021). The best model within our research is the Random Forest Regression, which received a mean squared error score of 0.003, which indicated that it is an extremely accurate model.

## Attestation

**Commented [JF2]:** Ask if this is included in the words

I understand the nature of plagiarism, and I am aware of the University's policy on this.

I certify that this dissertation reports original work by me during my University project except for the following:

- The data frame was mostly taken from the NBA website.[1]
- The code on feature selection was from the courtesy of Bex T [2]
- The code concerning the Artificial Neural Network was largely taken from this site, however it was changed to be more interactive by me Degnan Kopa to be used by the user or data scientist.[3]
- The code concerning the Random Forest Regression was largely taken from this site, however it was changed to be more interactive by me Degnan Kopa to be used by the user or data scientist[4]

**Signature:** .. Degnan Kopa **Date: 21/09/2022**

## Acknowledgements

I give my thanks and gratitude to my two supervisors, that is Dr. Anthony O'Hare and Mrs. Anya Kirpichnikova for their constant help and support that they have shown me through this dissertation. For there guidance and input during this research.

I give my thanks to the National Basketball Association for providing the data that was used during this dissertation.

## Table of Contents

The table of contents below is automatically generated from the paragraphs of style *Heading N* and *Unnumbered N*. To update this after revisions, right-click in the table and choose *Update Field* for the entire table; you may also be prompted about updates when printing.

## List of Figures

# 1 Introduction

## 1.1 Background and Context

The National Basketball Association is a professional basketball team that consists of 30 different teams.

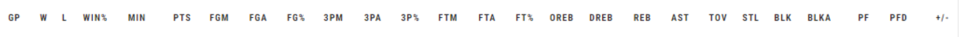
29 of these teams are located within the United States, and 1 team is in Canada, this team is known as the Toronto Raptors. Within the NBA, there are two types of seasons, these seasons are known as the regular season and the post season. In the regular season, each of the 30 teams will be playing 82 games. The 82 games are divided into two parts. Those two parts are the home and aways games.

However, for our dissertation, we will not be putting our focus on the regular season because predicting off of the regular seasons will give off the chances of making it to the playoffs.

That is because after the regular seasons, there will be 16 teams that will be going to the playoffs. These 16 teams are selected by their win-loss records.

Once these 16 teams make it to the playoffs, they will face 8 teams from the Eastern conference and 8 teams from the Western. If a team loses, they will be eliminated and thus not able to make it to the championship. Once the playoffs are finish that is after the 14 teams have been eliminated, 2 teams will be left, and these 2 teams will face each other to win the NBA championship.

There are some basketball terminologies to understand, these terminologies will be the independent variables within our data frame, and there will be one dependent variable which will be the winning percentage. Each of these variables contains values that have been marked down by data analyst during matches of the NBA to get the seasonal statistic of each individual team. The following shows the different basketball variables that will be used in our data frame and research.



| GP | W | L | WIN% | MIN | PTS | FGM | FGA | FG% | 3PM | 3PA | 3P% | FTM | FTA | FT% | OREB | DREB | REB | AST | TOV | STL | BLK | BLKA | PF | PFD | +/- |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|

*Figure 1: Different basketball actions and skills that will be used as our independent variable and dependent variables. Where the winning percentage is the dependent variable and the rest is the independent variable.*[1]

GP: Games Played – the number of games that have been played within the playoffs.

W: Wins – The number of wins.[5]

L: Losses – The number of losses.[5]

WIN%: Winning Percentage - This will be our dependent variable, or our target variable, this is the targeted basketball variable that we will try to predict using our machine learning models. Winning percentage is the fraction of games a team has won.[6]

MIN: Minutes – The number of minutes a team has played for.[6]

PTS: Points – The number of points a team has scored.[6]

FGM: Field Goal Made – The number of field goal shot that has been made. A field goal is any shot that has been made other than a free throw, depending on the distance it can be worth either two points or three points.[6]

FGA: Field Goal Attempts – The number of attempted field goals.

FG%: Field Goal Percentage – A percentage of all the shots taken during a game. It is used to get a measurement of how well a team shoots a ball during a game.

3PM: 3 Points Made – The number of three points that has made. This is the most difficult range to shoot from during a match, as the length between the basketball hoop and the three-point line ranges from 22 feet to 23 feet depending on the position a player is shooting from.

3PA: 3 Points Attempts – The number of three points that have been attempted.

3P%: 3 Points Percentage – A percentage of all the three points that have been taken during the game.

FTM: Free Throws Made – If a defender causes contact between himself and the ball-handler who is playing on offense this is called as a foul and allows the current ball-handler to have two free shots made behind the set line without it being hindered by any player. The average number of free throws made.

FTA: Free Throw Attempts – The average number of attempted free throws.

FT% - Free Throw Percentage – The average number of free throw attempts divided by the number of free throws made to get a measurement of how well a team shoots during a free throw.

OREB: Offensive Rebounds – The average number of rebounds that have been made by the offensive team, this is when the ball has been recovered by the offensive side and did not change possession.

DREB: Defensive Rebounds – The average number of rebounds that have been made by the defensive team; this is when the ball gets recovered by the defending team.

ASTS: Assists – The average number of assists that has been made. An assist happens when a player passes the ball to a team in such a way that it leads directly to a score by field goal.

TOV: Turnovers – A turnover comes to pass when a player loses possession of the ball to the opposing team before a shot has been attempted. TOV is the average number of turnovers that has been made.

STL: Steals – STL is the average number of steals. A steal occurs when a defensive player takes the ball from the offensive player.

BLK: Blocks – BLK is the average number of blocks made by the team. A block happens when a player on defence deflects a shot attempt from the offensive player.

BLKA: Block Attempts – BLKA is the average number of attempted blocks.

PF: Personal Fouls – PF is the average number of fouls that has been made by that team, a personal foul occurs when illegal contact has been made with an opponent.

PDF: Personal Fouls Drawn – This is the percentage of a team's personal foul that a player has drawn while on the court. It gives the average number of it.

+/-: POS/NEG: This is average number of point differential when a player or team is present on the floor.

Other than the winning percentage which will be our dependent variable, the rest of these will be our independent variable, which will be studied to see how much they effect and correlate with our dependent variable. This study will allow us to see which teams' performance is so well and linear to the point that they have a high chance of winning the NBA championship.

## 1.2 Scope and Objectives

In our dissertation, our objective is to use five different machine learning models to predict who has the highest chance of winning the NBA championship. The five different machine learning models we will be using is the Linear Regression, the Artificial Neural Network, the Random Forest Regression, the LASSO Regression, and the Ridge Regression technique. Each of these five models have a unique function, and two of them can be used in classification.

## 1.3 Achievements

During our dissertation we managed to achieve getting to rank of who would mostly win the championship, using a graphic design software, we created an ordered table showing us the ranking of the different teams that has the highest chances of winning next year and the lowest chances of winning this year.

We successfully averaged three different seasons, to remove outliers from just using one team, making our statistics more accurate and robust.

To understand the different basketball actions and how they affect the winning percentage, we used a method in machine learning known as feature selection, to see which of them causes an effect on the winning percentage.

## 1.4 Overview of Dissertation

The first one is called Linear Regression, we used linear regression because it a commonly used regression model. To find out how strong the relationship between two variables is linear regression is mostly used.

The second model we used in our research is the artificial neural network, the reasoning of why this was selected is because of its differences in dynamics in dealing with dependencies as compared to a regression model. Regression modelling is a method that is mainly used in dealing with dependencies that are linear however an artificial neural network can deal with nonlinearities. Artificial Neural Networks can learn and model non-linear and complex relationships.[7]

In data science, if there is a case where there are nonlinear dependencies, neural network is known to perform better than regressions.

The third machine learning model we used is the random forest regression, which uses multiple decision tree that each give an output, which we will be averaging according to the number of decision trees being used. [8]Random Forest Regression produces predictions that be understood easily and can handle a large dataset efficiently. It provides a higher level of accuracy when compared to a normal decision tree algorithm.

The fourth machine learn algorithm that is used in our research is known as the LASSO Regression, which uses a mathematical equation to perform shrinkage or regularize coefficients and to do variable selection for a linear regression model. The LASSO Regression can gather the subset of the independent variables that will decrease prediction error. It is used to avoid overfitting.

The Ridge Regression, which is our final machine learning model is used to analyse data that suffers from multicollinearity.[9]

Each of our five models has a special role that it will be playing in our model, and we will also be using them to study the dataset of the NBA model, and receive an output, that output will be known as the winning percentage.

# 2 History of Machine Learning

In a basic form, machine learning uses mathematical algorithm to analyze data. These mathematical algorithms are used to make predictions and to make decisions concerning the future. In the modern world, machine learning gives computer the ability to communicate with humans, drive cars, write and publish match reports on various sports, and predict natural disasters.

The concept of machine learning is dated back in 1950, when Alan Turing, a computer scientist and mathematician published an article that discussed whether machines could think. He formed a hypothesis that if a machine had successfully convinced humans that it is not a machine than it is possible that this machine had reached an intelligence that is artificial. This was called as the Turing Test.

In the year of 1957, Frank Rosenblatt, an American psychologist, who was well known in the domain of artificial intelligence designed the first neural network for computers.[10] Currently, this neural network is known as the perceptron model. Frank Rosenblatt was also known as the father of deep learning because of the perceptron algorithm which he designed to use and classify visual input then categorize subjects into one of two groups.

In the year of 1959, Bernard Widrow, a professor of electrical engineering at Stanford University, and Marcian Hoff, one of the inventors of the microprocessors, created two neural network models called as Adeline and Madeline. Adeline could detect binary patterns. Madeline was able to eliminate echoes on phone lines.[11]

In 1967, the Nearest Neighbor algorithm was written, this allowed computers to use very basic pattern recognition.

During the 1990s, machine learning begun to shift from a knowledge-driven approach to a data driven approach. Computer scientists begin to program computer that would be able to analyze large amounts of data and draw conclusions from the results. [11]

Adrien-Marie Legendre, a French mathematician was the first to develop the Least Squares Method which has an application to Linear Regression. During the year of 1805, Legendre used the Least Squares linear regression technique as a method to find a good rough linear fit to a set of points, it was also used by Carl Friedrich Gauss in the year of 1809 to get the prediction concerning planetary movement. [12]

The algorithm behind the Random Forests was introduced in 1995 using the random subspace technique, by Tin Kam Ho, she is a computer scientist that has made many contributions to the field of machine learning, and data mining. She has also done pioneering work in ensemble learning.

To improve prediction accuracy and to be able to interpret statistical models better, the LASSO Regression technique was introduced by Robert Tulshiram. He wrote a paper on Regression Shrinkage and Lasso Regression in the year of 1996.

In 1970, Hoerl and Kennard introduced the Ridge Regression method in the article called as Technometrics, it mentions that ridge regression is a biased estimation of nonorthogonal problems and deals with multicollinearity.

In the modern world machine learning has had a huge impact in many segments such as healthcare, education, transportation, food, and entertainment. To enhance different technological devices and increase the artificial intelligence of devices, technologies such as cloud computing and internet of things have become a growing implementation of machine learning. Big companies are now using machine learning to get a leverage in customer satisfaction, this means that the pattern we can find in data can be useful for companies.

Machine learning is the branch of artificial intelligence that deals with the designing and developing of algorithms that will learn from data and based off that data make predictions. The goal of machine learning is to automate the building of analytical models so that computers can learn from data without the use of being programmed in a detailed manner.[13]

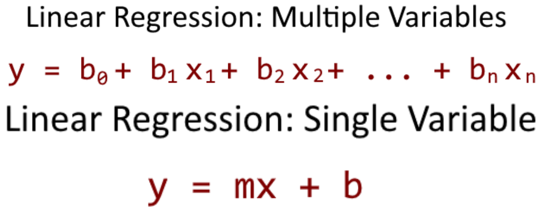
In conclusion, machine learning can be used as a great tool to make predictions. However, a machine learning model can only be good if the data that it inputted is good and well-detailed as well. To have a machine learning model be accurate, it must be given high-quality data that can well represent the real-world data that the machine learning algorithm will be used on.

# 3 Methodology

## 3.1 Linear Regression

The linear regression is a mathematical analysis technique that is mainly used to predict the value of a specific variable, this is usually known as ‘y’, and this will be impacted by one or more variables. The following formula shows the mathematical formula for a linear regression and a linear regression with multiple variables.

Where y is the dependent variable, the value aimed to be predicted, and the x represents the independent variable, and b representing the weight, value, or how much the specific independent variable effects the predicted variable.



Linear Regression: Multiple Variables

$$y = b_0 + b_1 x_1 + b_2 x_2 + \dots + b_n x_n$$

Linear Regression: Single Variable

$$y = mx + b$$

*Figure 2: Single linear regression formula using a single variable, and multiple linear regression using multiple variables.*

In linear regression, the variable that we are looking to predict is called as the dependent variable, and in this case, it is the WINP variable, which is short for winning percentage. As the name is implied, the dependent variable is dependent on other variables, which are known as the independent variables, and these independent variables are the other 25 variables which was mentioned earlier.

In our code, we will be using our 25 independent variables to predict the winning percentage.

There are a couple of goals that our regression model will seek to achieve as they are being used, the first is to see how well the independent variables will predict our dependent variable. The second goal is to see which of the twenty-five variables will be significant predictors of the outcome variable.

Some of the instruction used in our code has been used multiple time in our other regression models, this is because they are important in the processing of our data.

The main one that is constantly used is pandas. Pandas is a Python package that is mainly used in the analysis of data, in this case, we will be using it in machine learning and will denote this as pd when we begin to import our libraries.

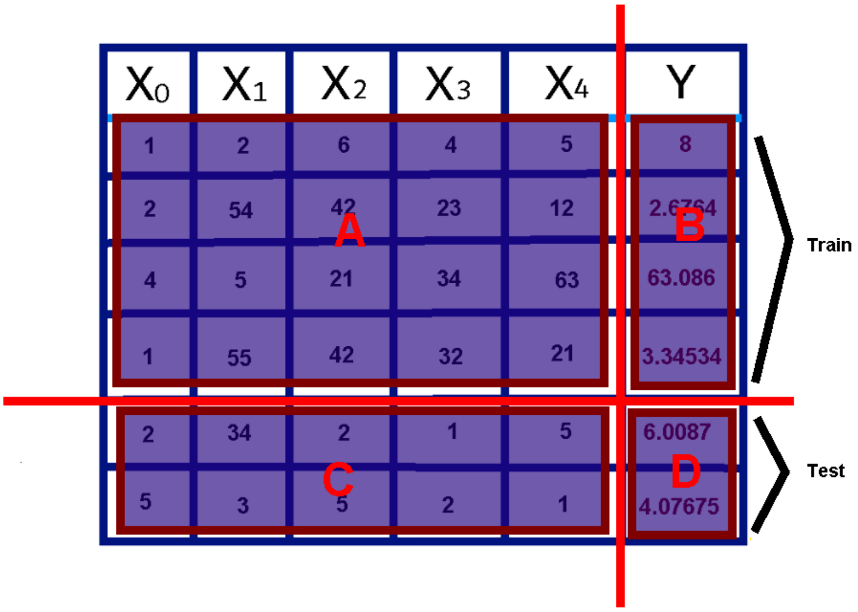
First, we need to call our data frame, after we call in our data frame, we will want to view our data frame, to check if there is anything that needs to be added or removed.

In our data frame, the column known as ‘Team’ is unnecessary in our regression model. The reason why it is unnecessary in our regression model is because the values will be consisting of strings, and these strings are the names of the 15 teams that made it to the playoffs. Since this is a regression model, where we are only dealing with numerical values, this column will be removed. After this process is done, it is possible that the data frame could contain NaN value, and if so, these NaN values will have to be removed.

This whole process is known as the data cleaning part.

Afterwards, we will define the independent and dependent variables, the ‘x’ variable will contain the values of the 25 independent variables, and the ‘y’ variable will contain the value of the dependent variable, which is the winning percentage.

Next is the one of the most important parts of our data process, from the libraries of sklearn of train_test_split.
To go in further to what the train_test_split does it will be splitting our arrays or our matrices into two subsets that are random, a train and training subset.
This can be done because, when we are initialized our independent and dependent variables, they were from into an array or matrices of these values.
The following figure shows a graphical representation or abstract representation of the split:



| $X_0$ | $X_1$ | $X_2$ | $X_3$ | $X_4$ | $Y$ |
|---|---|---|---|---|---|
| 1 | 2 | 6 | 4 | 5 | 8 |
| 2 | 54 | 42 | 23 | 12 | 2.6764 |
| 4 | 5 | 21 | 34 | 63 | 63.086 |
| 1 | 55 | 42 | 32 | 21 | 3.34534 |
| 2 | 34 | 2 | 1 | 5 | 6.0087 |
| 5 | 3 | 5 | 2 | 1 | 4.07675 |

*Figure 3: Diagram of the training and test split.*

(A) Represents the x_train.
(B) Represents the y_train.
(C) Represents the x_test.
(D) Represents the y_test.

During the train_test_split, we also setted the random state, the random state during this train and test split is set to zero, if the random state is set to 0, that means we will be getting the same train and test sets across different executions.
So, in a basic way to say this, it means that no matter how many times our code gets executed, the result would be the same, i.e same values in train and test dataset.
After splitting the training and test set, we will begin to start training the model on the training set.
We decided that the machine learning model that we would use is the Linear Regression, therefore from sklearn we will import that model.

### 3.1.1 Fitting the Model

Next, we will see how well the linear regression model generalizes to similar data to that on which it was trained. My x_train, and y_train are also recognized as data points, and what the fit function will do once, we put in those two parameters, it will see how well it fits into those data points.
The fit function will place a line between the data such that it touches as much data points as it can. It will give a function that represents a line that will best fit all the points.

### 3.1.2 Line of Best Fit

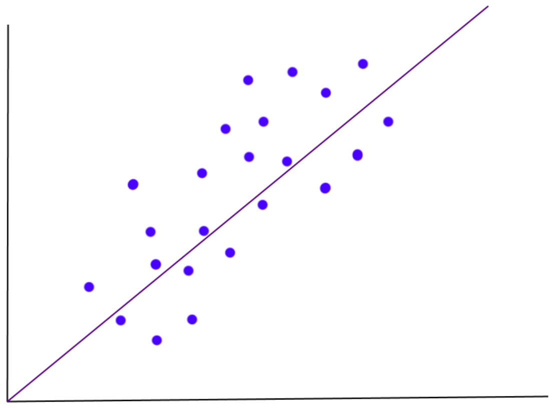


*Figure 4: A graphical representation of a line of best fit. Which is used by the Linear Regression to see the correlation of the data points.*

In all our regression, we will plot a line that will give us the line of best fit, which will be able to give us an expression of the relationship between those points.
This will cause it to reveal the trend of a data set by showing the correlation between variables. It is useful for making predictions.
The closer my data points get to the line of best fit, the closer my independent variables and dependent variables are closely related meaning it has high correlation. This means that the three-point percentage, the two-point percentage, offensive rebound, etc. all have a great amount of impact on the winning percentage.
After graphing, we created a table to have a visual of the difference

After the training of the model on the training set, we are ready to do our prediction, and predict the test set results.

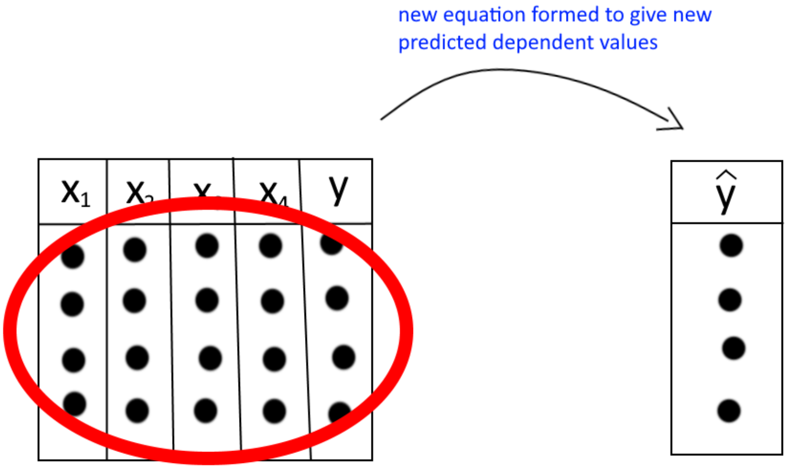


*Figure 5: A diagram that shows how the model fitting works in a basic explanation; this is how the predictions are gained.*

## 3.2 Artificial Neural Network

So, the general architecture of the artificial neural network looks like the following: [14]

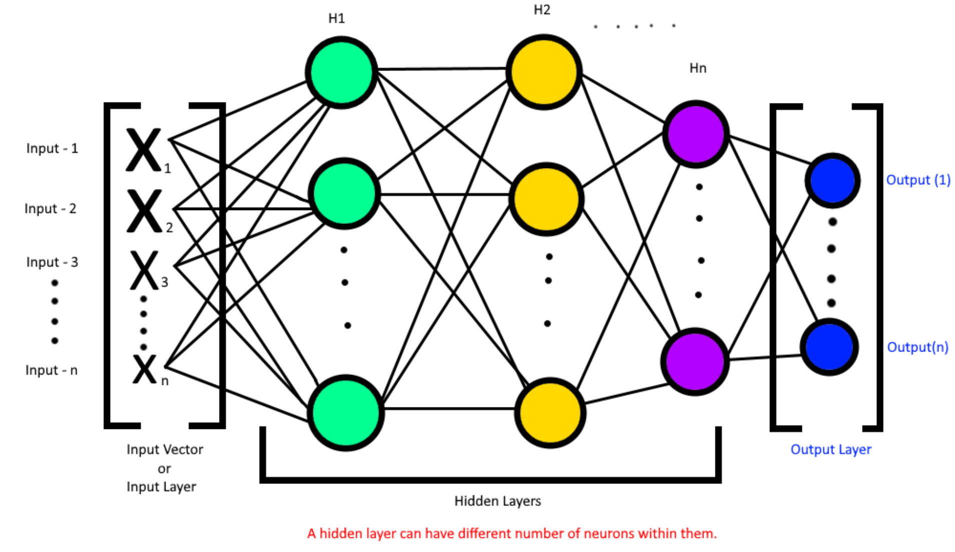


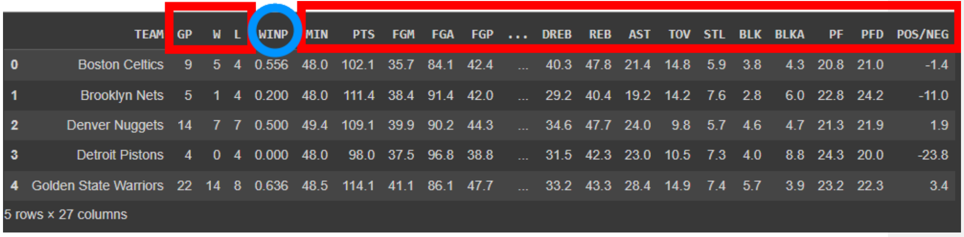
*Figure 6: A general architecture of the artificial neural network. Where the coloured circles are the neurons, and the lines represents the weights.*[15]

In this architecture, we can see that there is one input layer, one output layer, but there can also be multiple hidden layers, the hidden layers here are represented by H1, H2, and Hn. So, the question that is to be raised in this situation, is exactly how many hidden layers we should keep in our neural network.

So, this is a hyperparameter and depending on our data, we will decide how many layers need to be kept. The decision on the layer will be decided based on the accuracy. If the data tends to be complex, then that means more layers will be utilized to learn it. If the data is simple, then you will only need about one or two layers, and that much will be able to do the job for you.

The input layer and the output layer are decided by the number of values present within the data frame. The input layer is dependent on how many predictors, or independent variables we have.

According to our data frame,



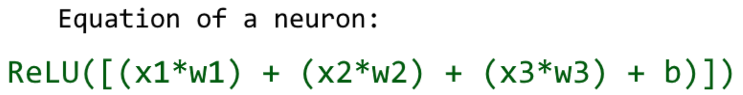
| | TEAM | GP | W | L | WINP | MIN | PTS | FGM | FGA | FGP | ... | DREB | REB | AST | TOV | STL | BLK | BLKA | PF | PFD | POS/NEG |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Boston Celtics | 9 | 5 | 4 | 0.556 | 48.0 | 102.1 | 35.7 | 84.1 | 42.4 | ... | 40.3 | 47.8 | 21.4 | 14.8 | 5.9 | 3.8 | 4.3 | 20.8 | 21.0 | -1.4 |
| 1 | Brooklyn Nets | 5 | 1 | 4 | 0.200 | 48.0 | 111.4 | 38.4 | 91.4 | 42.0 | ... | 29.2 | 40.4 | 19.2 | 14.2 | 7.6 | 2.8 | 6.0 | 22.8 | 24.2 | -11.0 |
| 2 | Denver Nuggets | 14 | 7 | 7 | 0.500 | 49.4 | 109.1 | 39.9 | 90.2 | 44.3 | ... | 34.6 | 47.7 | 24.0 | 9.8 | 5.7 | 4.6 | 4.7 | 21.3 | 21.9 | 1.9 |
| 3 | Detroit Pistons | 4 | 0 | 4 | 0.000 | 48.0 | 98.0 | 37.5 | 96.8 | 38.8 | ... | 31.5 | 42.3 | 23.0 | 10.5 | 7.3 | 4.0 | 8.8 | 24.3 | 20.0 | -23.8 |
| 4 | Golden State Warriors | 22 | 14 | 8 | 0.636 | 48.5 | 114.1 | 41.1 | 86.1 | 47.7 | ... | 33.2 | 43.3 | 28.4 | 14.9 | 7.4 | 5.7 | 3.9 | 23.2 | 22.3 | 3.4 |

5 rows × 27 columns

*Figure 7: The first five teams of our NBA dataframe, the rectangle line is selecting all of the independent variable we will be using, and the blue circle is selecting the dependent variable or target variable which is the winning percentage.*[1]

25 of our variables will be our predictors, and one will be our predicted variable. These 25 predictors will be the inputs, which means that will be having 25 inputs, and those 25 inputs will be sent to the layer.

The input layer does not contain neurons, but only the hidden layer and output layer. You only have placeholders in the input layer. The number of layers, and the number of neurons should be decided based on the accuracy.
Within a neuron, an equation will be formed, in the architecture above, each of those circles are just neurons. So, each of these circles will have one equation written, so while deciding how many hidden layers and neurons there should be, we must also understand that there will be a lot of computational power required.
The following shows the equation contained in each neuron:



Equation of a neuron:

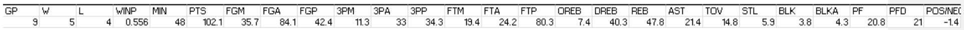
ReLU([(x1*w1) + (x2*w2) + (x3*w3) + b)])

*Figure 8: The formula of the each of the neurons, using the ReLU activation technique.*

This is because of the amount of equation that is to be formed, calculated, and optimized. The goal is to find out the minimum number of hidden layers, and the minimum number of neurons, which can give you the highest amount of accuracy. [12]We will continue this until a saturation point is reached, and the accuracy is no longer improving.
When we are using artificial neural networks in the case of regression modelling, we will be having only one output, because we are just predicting one number.

### 3.2.1 How does an artificial neural network learn the data?

So, as shown in the data frame above, the target variable in our case is the winning percentage or the WINP variable, and the rest of the variables such as the OREB, the DREB, the three-point percentages, etc., these are all predictors, and what these predictors are attempting to do is to help understand why the value of the winning percentage is currently what it is.



| GP | W | L | WINP | MIN | PTS | FGM | FGA | FGP | 3PM | 3PA | 3PP | FTM | FTA | FTP | OREB | DREB | REB | AST | TOV | STL | BLK | BLKA | PF | PFD | POS/NE( |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 9 | 5 | 4 | 0.556 | 48 | 102.1 | 35.7 | 84.1 | 42.4 | 11.3 | 33 | 34.3 | 19.4 | 24.2 | 80.3 | 7.4 | 40.3 | 47.8 | 21.4 | 14.8 | 5.9 | 3.8 | 4.3 | 20.8 | 21 | -1.4 |

*Figure 9: A slip of what is causing the effect to the winning percentage, the other variables other than the winning percentage.*[1]

All those values contained in our predictors is the reason why the winning percentage of the Boston Celtics is a 0.556.

The following process explains how the data frame will be passing through our neural network.

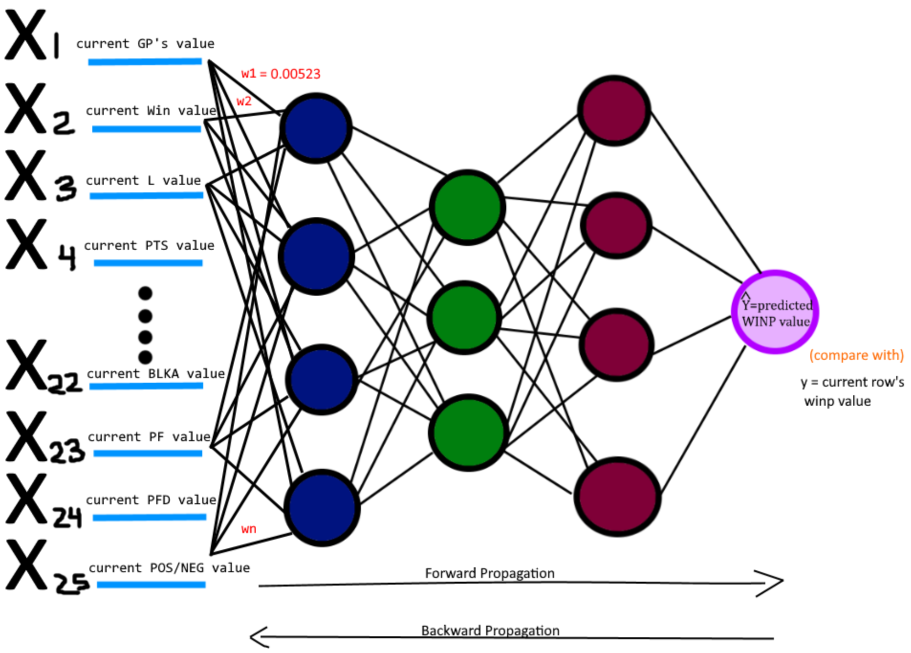


*Figure 10: A diagram showing what is happening to our data frame as it is going through the neural network.*

First, we have one output neuron which will be producing a number, and that number is the winning percentage, in our figure our output neuron is colored in lavender. Our data frame concerning the NBA playoff is based of predicting the winning percentage based on their overall skillset characteristic as a team. And when want to predict the next year or season winning percentage, you should be able to predict it based on its characteristic using the neural network that has been built. It can do that because it has learned all the value that is currently in our seasonal data frame.

So, the way this data will travel, is by first taking the first row, it will take the value that is to be predicted, in this case 0.556, this will be recognized as our y, and our neural network will try to predict a value close to 0.556, this will be dependent on the initialization of the weights within our neural network. In our figure, the weights are the lines connecting the neurons.

In place of the X1 to X25, it will start placing the values such as the MIN, PTS, FGA, FGM, etc. After the first row gets its learning process done, the second row's value will begin to populate the X1 to X25, after the second row, then the third-row value will begin to be populate the X1 to X25, and one by one, each of the row the neural network will be able to consume to be learned.

When a row is passed into the input what happens is that each of the values within our input layer will be passed into each of the neurons within our first hidden layer, as shown in the figure above.

If you remember the equation of a neuron, each of those values will be multiplied by a value of weight, then they will be added up. The lines within our figure represents weights, and these weights contain a value that will be used during the multiplication of our input within the equation of our neuron. Each of the weights (represented by the w1, w2, wn in our diagram) will be assigned a very small random value, and the reason this happens is because the data hasn’t properly been learned. But this value of the weights will begin to either increase or decrease based on the error produced as it gets closer and closer to the ‘y’ value. And as the value gets close to the value of the y, you won’t have to change the weights too much. The y is the current rows winning percentage. [16]

As you can see from the diagram, the outputs of neurons of the first layer will move on the second layer of neurons, then the output of the second layer of neurons will move on to the third layer of neurons. This process is known as forward propagation, which is the first step. Once you have done this, you will get a predicted winning percentage, and this value may or may not match with the value of the current row winning percentage. Our predicted winning percentage may not match the current winning percentage and it is possible that why may be far from each other. To calculate the distance, or how much they differentiate from each, we can calculate the sum of square error. This is the equation for that:

$$\text{Sum of Squared Error} = ( Y - \hat{Y} )^{2}$$

*Figure 11: Formula for the Sum of Squared Error that is used to calculate the difference between the actual value and the predicted value of our target variable*[17]

The further the SSE is away from 0, then more in error is your weightages. However, the ideal value of the SSE should be 0. If it is far away from the 0, then all the weightages within the equation of all the neurons within the network needs to be adjusted so that the SSE can be lessened. After your computer the cost, and get the sum of squared error, you will move on to the third step. The third step of the artificial network’s working is the backward propagation which is when you adjust the weightages starting from the last layer to the first layer. So, in total, there are three steps.

First is the forward propagation, then second is the computation of the cost, which means getting the SSE. The third is the backward propagation, which is when you begin the adjustment of the weightages.

For each of the row in our data, this process will be happening, the ‘y’ or the winning percentage will also change into the current row winning percentage. When this process happens for each of the row this is known as the Stochastic Gradient Descent, here the batch size has been set to 1. When the batch size is equal to the total number of rows, then this is known as full batch gradient descent. And if the batch size is set to either five, ten, or twenty then this is called as mini batch gradient descent. [18]

Mini batch gradient works in this manner, when the first row is passed, it will produce a sum of squared error, the same for all the rows, depending on the batch size we decided to set, we will add all the sum of squared error gathered for each row, and get the average of all the sum of squared error. This is how the mini batch gradient works. The following shows a visual representation of what has been explained: [18]

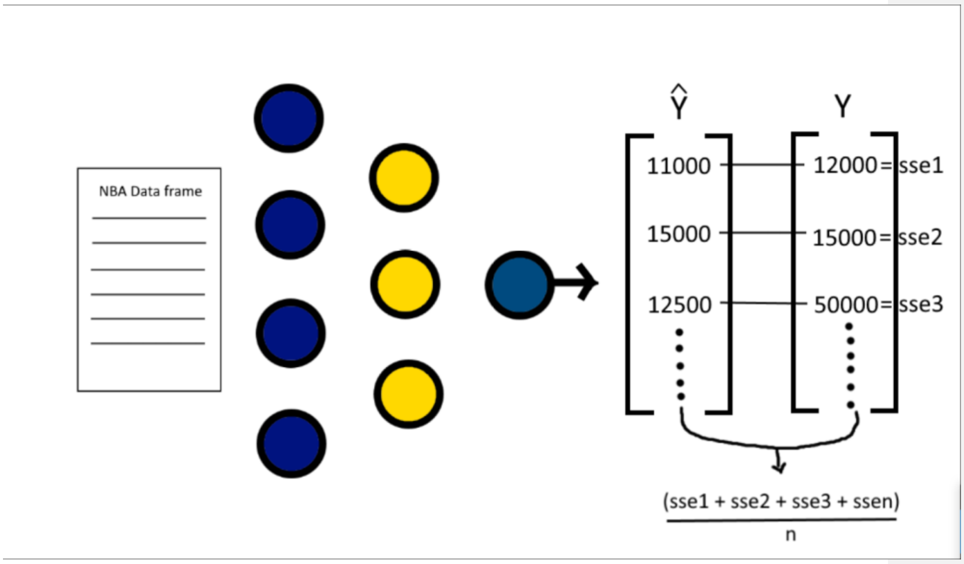


*Figure 12: A figure showing the workings of a mini batch gradient, and how it works.*

During the mini batch gradient, we will change the value of weights after the number of batch size it has been set to.
Epoch is when a whole data passes through a network. There can be multiple number of epochs. The difference between batch size and epoch is that for batch size, the weights change after a certain number of rows have been passed through the network. An epoch is when you pass the whole data, with the purpose of studying the whole data according to number of epochs. As the epoch increases, the number of errors should be decreasing as well.
This is the working of an artificial neural network.

In my code, the approach I took was more of a user-inputted approach, which makes it different than the average code that you would see on the internet.
First, I defined, the winning percentage, and the independent variables. But this time, I used a Scaler in this case, what a Scaler does is that it will cause us to optimize the variables. In machine learning algorithms, if the values of the features are closer to each other[16] then there is a chance that the algorithm will get trained well, and faster, instead of the dataset where the datapoints or features values have high differences with eachother where it will be taking more time to understand the data and the accuracy will be lower.[16]
If the data in any conditions have data points far from each other, scaling is a technique to make them closer to each other.[16] It is used to make data points generalized so that the distance between them will be lower.
The way we standardize features[9] is by removing the mean and scaling to the unit variance. The standard score of a sample x is calculated[9]:

$$z = (x-u)/s$$

*Figure 13: Formula for a standard score a sample*

Centering and scaling happen independently on each feature by computing the relevant statistics on the samples in the training set. The mean and standard deviation are then stored to be used on a later data using transform. After doing the standardization, we will perform the train and test split. Next, within our code, we allow the user or the data scientist to choose parameters of the artificial neural network. These parameters are the number of neurons, the activation style, he/she will have the ability to choose between the ReLU activation function, sigmoid/logistic activation, and the hyperbolic function. Once that is done, the input layer, the hidden layer, and the output layer comes in to play, and the number of neurons, and the activation type is set by the user. We will be having only one output, which will give out the winning percentage, and because this is a regression problem not a classification problem. The user will then enter the amount epoch, and will be receiving the loss amount, which is the mean squared error, the less loss there is the better. Loss represents the penalty for bad prediction. As the epoch increases, the loss decreases which indicates that the data frame is being well learned.

We want to find at which parameters did the best in predicting our winning percentage for our NBA Dataframe, when we placed it into the model, the user will put in the amount of batch sizes, and the amount of epoch, the artificial neural network is then made, and for each batch sizes and epoch, it will check the accuracy. So, with a batch size of five, and an epoch of 20, the accuracy is 87.09, which is the highest accuracy among the other parameters.

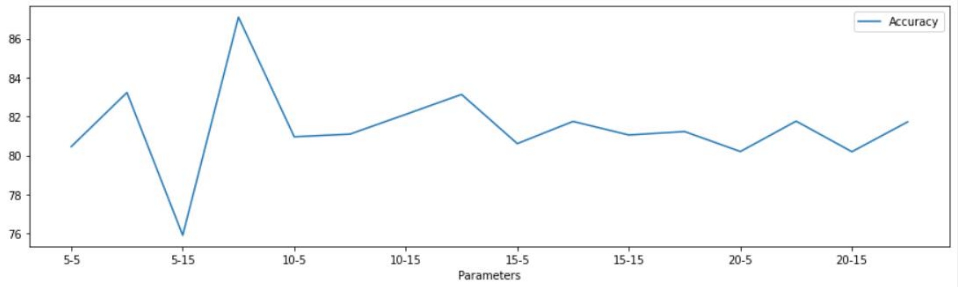
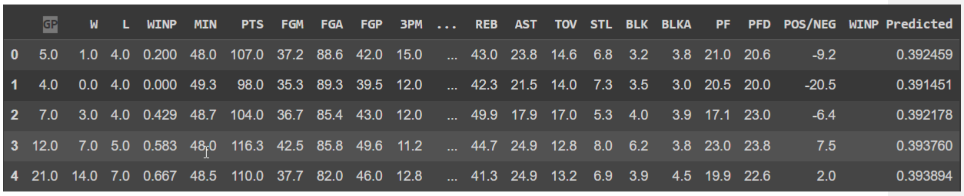


*Figure 14: A line graph showing at which point did our artificial neural had the best accuracy when using hyperparameter tuning.*



| | GP | W | L | WINP | MIN | PTS | FGM | FGA | FGP | 3PM | ... | REB | AST | TOV | STL | BLK | BLKA | PF | PFD | POS/NEG | WINP Predicted |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 5.0 | 1.0 | 4.0 | 0.200 | 48.0 | 107.0 | 37.2 | 88.6 | 42.0 | 15.0 | ... | 43.0 | 23.8 | 14.6 | 6.8 | 3.2 | 3.8 | 21.0 | 20.6 | -9.2 | 0.392459 |
| 1 | 4.0 | 0.0 | 4.0 | 0.000 | 49.3 | 98.0 | 35.3 | 89.3 | 39.5 | 12.0 | ... | 42.3 | 21.5 | 14.0 | 7.3 | 3.5 | 3.0 | 20.5 | 20.0 | -20.5 | 0.391451 |
| 2 | 7.0 | 3.0 | 4.0 | 0.429 | 48.7 | 104.0 | 36.7 | 85.4 | 43.0 | 12.0 | ... | 49.9 | 17.9 | 17.0 | 5.3 | 4.0 | 3.9 | 17.1 | 23.0 | -6.4 | 0.392178 |
| 3 | 12.0 | 7.0 | 5.0 | 0.583 | 48.0 | 116.3 | 42.5 | 85.8 | 49.6 | 11.2 | ... | 44.7 | 24.9 | 12.8 | 8.0 | 6.2 | 3.8 | 23.0 | 23.8 | 7.5 | 0.393760 |
| 4 | 21.0 | 14.0 | 7.0 | 0.667 | 48.5 | 110.0 | 37.7 | 82.0 | 46.0 | 12.8 | ... | 41.3 | 24.9 | 13.2 | 6.9 | 3.9 | 4.5 | 19.9 | 22.6 | 2.0 | 0.393894 |

*Figure 15: Winning predictions using Artificial Neural Network output*

## 3.3 Random Forest Regression

One of the most accurate of all the machine learning model is the random forest regression. Of all our machine learning model the random forest regression was[19] declared to have the highest accuracy. The way the random forest regression works is by using the machine learning algorithm known as the Decision Tree. To understand the random forest regression, must understand how a decision tree works.

A decision tree organizes a series of roots in a tree structure. A decision can predict the value, and this is done by the decision rule that is inferred from the training data. The following figure show the structure of a decision tree:

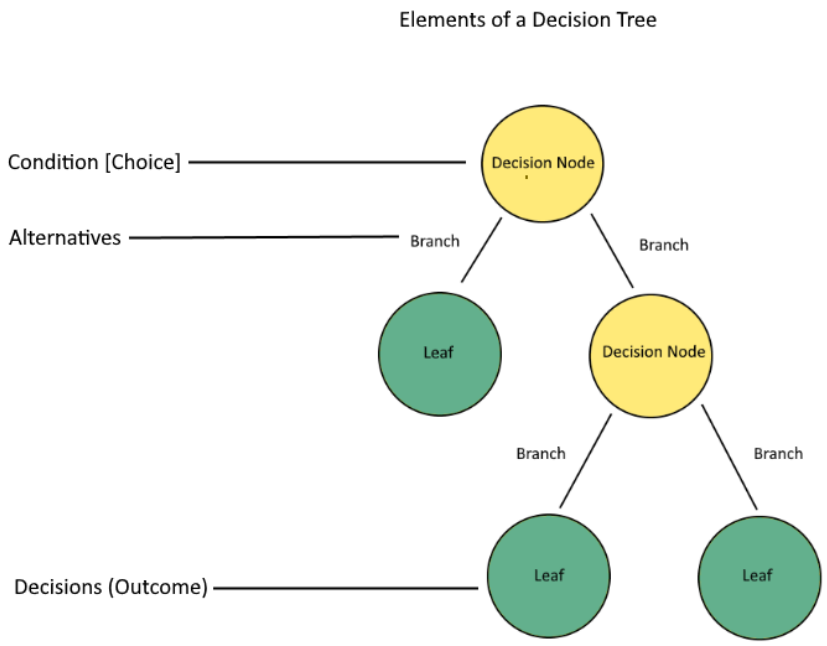


*Figure 16: A normal decision tree*

The decision tree starts from the root of the tree and will follow a branch based on the validation of the variable that is within the decision node, and this will continue until a leaf node has been reached and the result is given.

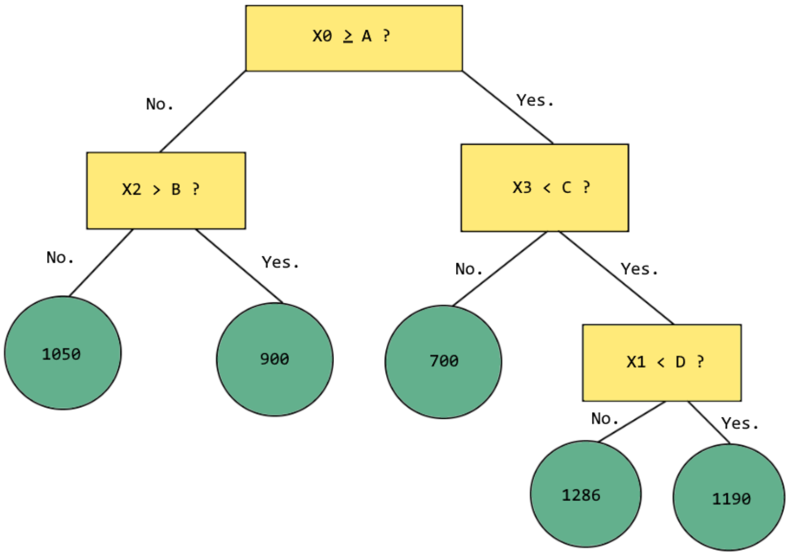


*Figure 17: Diagram explaining the algorithm of a decision tree.*

What I made here is a basic decision tree, which will be starting with the variable known as 'X0' and will be splitting based off specific criteria. If our answer is yes, the decision tree will follow through that specific branch. If our answer is no, the decision will go through the other path. This process will repeat until the decision tree reaches the leaf node and the resulting outcome is decided. The values of A, B, C, or D could be represented by either a numeric or a categorical value.

Ensemble learning is when we use multiple models, and it gets trained over the same data. We then average the result of each of those models to find a better result. What we require when using ensemble learning is that each of the model's error will be different from different tree. In the random forest regression, a concept known as bootstrapping is used, and this is the process of randomly sampling subsets of a given data frame, which will be done over several iterations, once the result is obtained then they are averaged together to obtain a better result.[20]

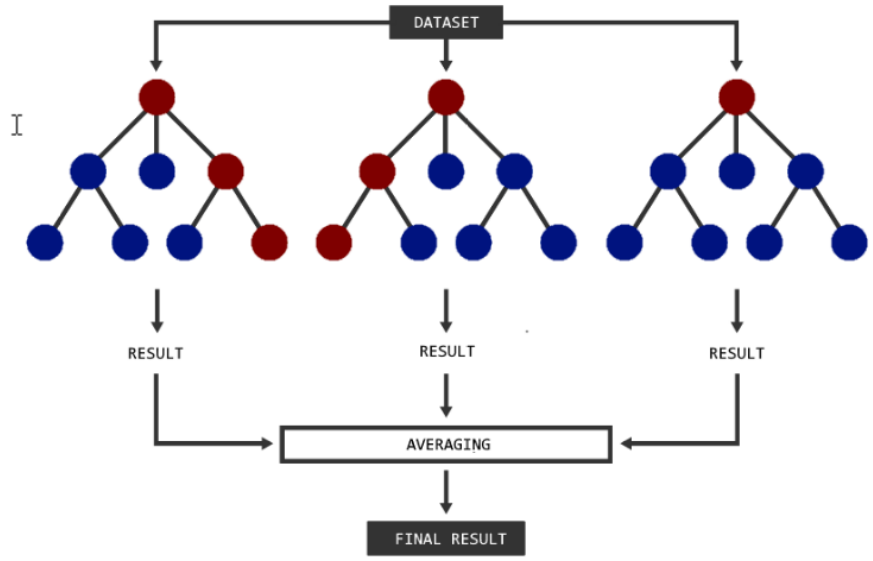


*Figure 18: Diagram showing how a Random Forest Regression works.*

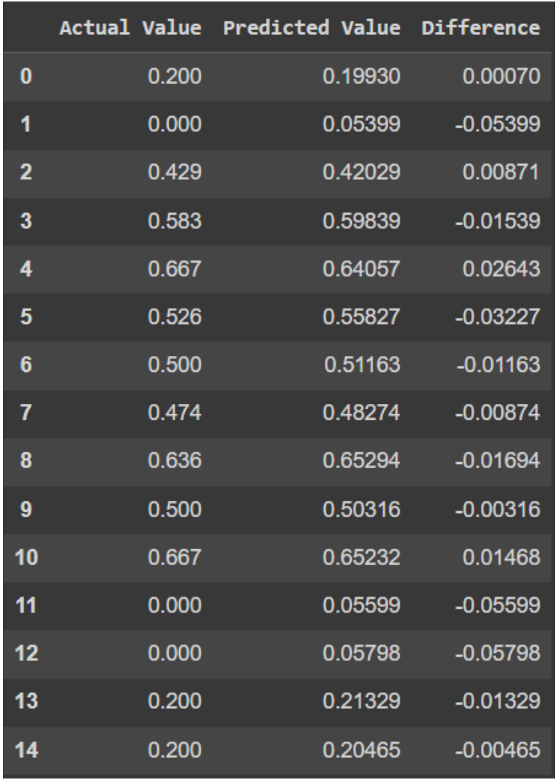
That is the working of a random forest regression.

As in the previous machine learning model, we did in our code, we upload the NBA data frame, after uploading the data frame, we need to clean our data, or check and see if our data frame needs any cleaning. It is possible that we may have some null values in our data frame, or a value that wasn't recorded.

Our data has gone through cleaning, and there is no null value, and any irrelevant column has been removed such as the Team columns, since this is a regression problem.

After we are doing the cleaning of the data, and the training and test split, we will begin initializing the model, to call it to our code, then we will fit the train and test split data frame, after that we will get what was predicted off the x_test and get what is now the predicted winning percentage.

This is what we got when we predicted the new winning percentage:



| | Actual Value | Predicted Value | Difference |
|---|---|---|---|
| 0 | 0.200 | 0.19930 | 0.00070 |
| 1 | 0.000 | 0.05399 | -0.05399 |
| 2 | 0.429 | 0.42029 | 0.00871 |
| 3 | 0.583 | 0.59839 | -0.01539 |
| 4 | 0.667 | 0.64057 | 0.02643 |
| 5 | 0.526 | 0.55827 | -0.03227 |
| 6 | 0.500 | 0.51163 | -0.01163 |
| 7 | 0.474 | 0.48274 | -0.00874 |
| 8 | 0.636 | 0.65294 | -0.01694 |
| 9 | 0.500 | 0.50316 | -0.00316 |
| 10 | 0.667 | 0.65232 | 0.01468 |
| 11 | 0.000 | 0.05599 | -0.05599 |
| 12 | 0.000 | 0.05798 | -0.05798 |
| 13 | 0.200 | 0.21329 | -0.01329 |
| 14 | 0.200 | 0.20465 | -0.00465 |

*Figure 19: This is the output of our Random Forest Regression program after modelling.*

The actual value represents the dependent variable of the 30% that was splitted when we did our train and test split. The y_pred represents, which in this case is the predicted value represents the new winning percentage.
To see the correlation of the actual winning percentage and the predicted winning percentage, the following graph was plotted, and this graph shows us there is a high correlation.

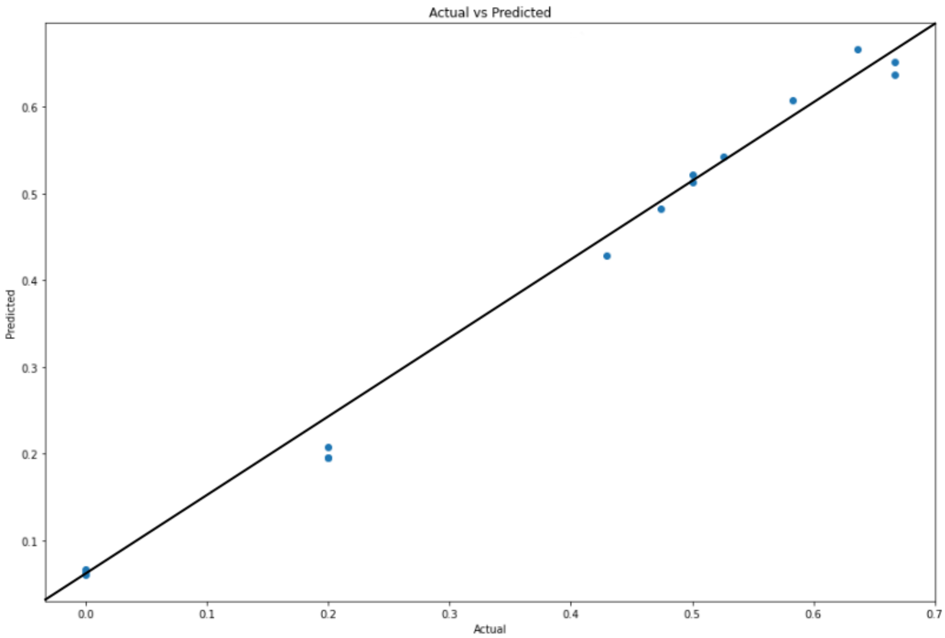
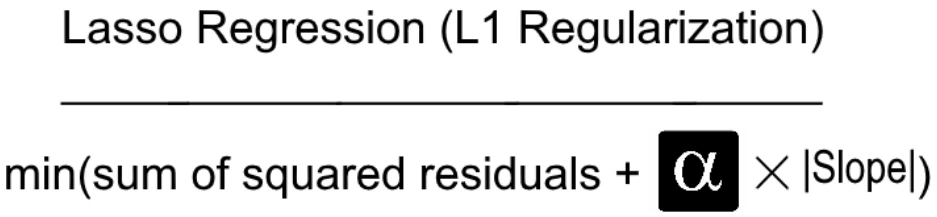


*Figure 20: Line of Best Fit when using the Random Forest Regression. Shows how accurate our model is, and how close the data points are to the line of best fit.*

## 3.4 LASSO Regression

The word 'LASSO' stands for Least Absolute Shrinkage and Selection Operator.[21]
The LASSO regression [21]technique is like the Ridge regression technique. The difference between the LASSO regression and the Ridge regression technique is that with LASSO what we will try to do is introduce a bias term rather than squaring the slope like we did in Ridge regression. In LASSO regression, we will make the slope an absolute value, and add it as a penalty term. The overall idea of this machine learning model is to avoid overfitting. In LASSO Regression, we will be minimizing the residuals of the sum of squared, while a penalizing term is also added to it. So, we get the follow equation:



Lasso Regression (L1 Regularization)

$$\min(\text{sum of squared residuals} + \alpha \times |\text{Slope}|)$$

*Figure 21: Lasso Regression's mathematical formula*

In Ridge regression, the slope instead of being an absolute value, it was squared. The penalty term in our equation is the alpha multiplied by absolute slope. The purpose of the penalty term is so that we can increase the generalization ability of regression model. In our equation, if we increase the alpha, the slope of the regression becomes more horizontal, and the model will become more tolerant of the different independent variables, as you can see here in our graph, as we increase the alpha, the lasso regression line becomes more horizontal.

The benefits of LASSO regression are that it can be useful for feature selection and if there are independent variables that are useless. LASSO regression also can reduce slope to be equal to zero, as you can see in our graph, the slope is almost equal to zero as we increase our alpha in Ridge Regression this cannot be done.

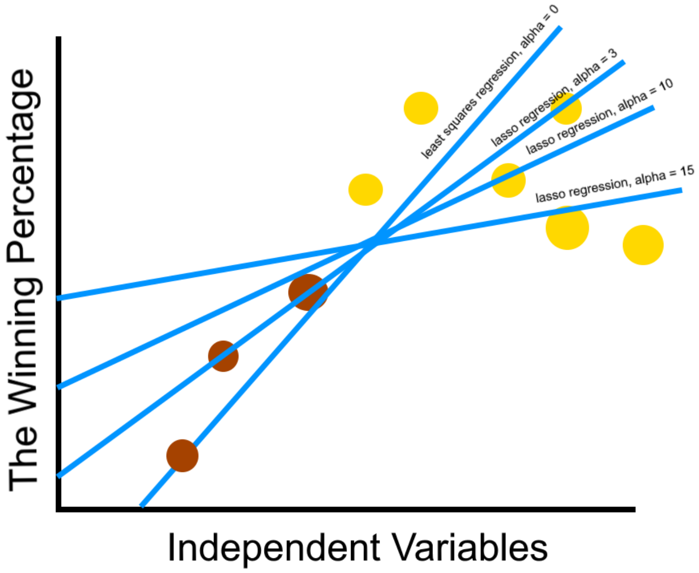
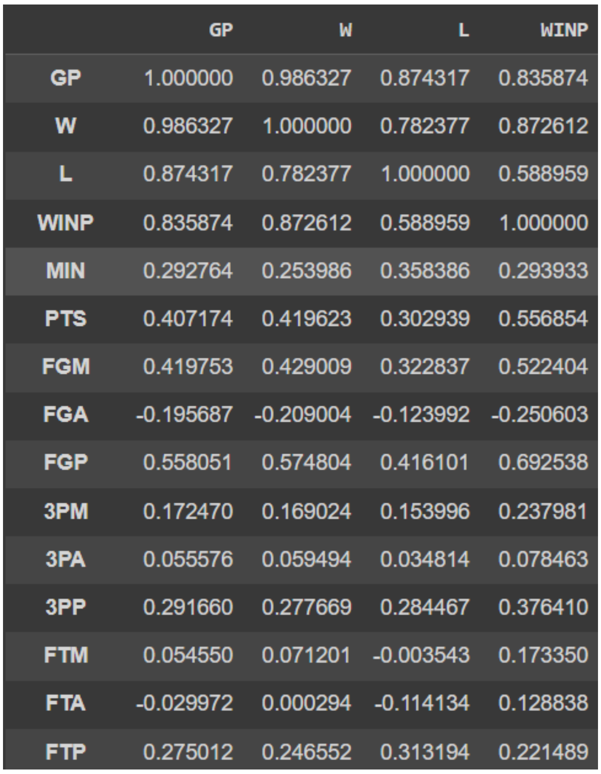


*Figure 22: This is what happens when the alpha begins to be increased when using the LASSO Regression. The line best fit becomes more horizontal.*

In our code, we were able to get the correlation between the variables, in order to see how much the variables relate to each other, in the other machine learning model, we chose to omit this technique. What the correlation of the variables will tell us is the relationship between two different variables. The following tables is the table of correlations of the different variables in our machine learning model.



| | GP | W | L | WINP |
|---|---|---|---|---|
| GP | 1.000000 | 0.986327 | 0.874317 | 0.835874 |
| W | 0.986327 | 1.000000 | 0.782377 | 0.872612 |
| L | 0.874317 | 0.782377 | 1.000000 | 0.588959 |
| WINP | 0.835874 | 0.872612 | 0.588959 | 1.000000 |
| MIN | 0.292764 | 0.253986 | 0.358386 | 0.293933 |
| PTS | 0.407174 | 0.419623 | 0.302939 | 0.556854 |
| FGM | 0.419753 | 0.429009 | 0.322837 | 0.522404 |
| FGA | -0.195687 | -0.209004 | -0.123992 | -0.250603 |
| FGP | 0.558051 | 0.574804 | 0.416101 | 0.692538 |
| 3PM | 0.172470 | 0.169024 | 0.153996 | 0.237981 |
| 3PA | 0.055576 | 0.059494 | 0.034814 | 0.078463 |
| 3PP | 0.291660 | 0.277669 | 0.284467 | 0.376410 |
| FTM | 0.054550 | 0.071201 | -0.003543 | 0.173350 |
| FTA | -0.029972 | 0.000294 | -0.114134 | 0.128838 |
| FTP | 0.275012 | 0.246552 | 0.313194 | 0.221489 |

*Figure 23: The output concerning the LASSO Regression of the predicting winning percentage when using the Python Programming language.*

A scatter plot representing these correlation is shown below, however not all the variables are shown because of the amount of independent variables we have. The user has the ability to choose which variables within the NBA are highly correlated with the winning percentage.

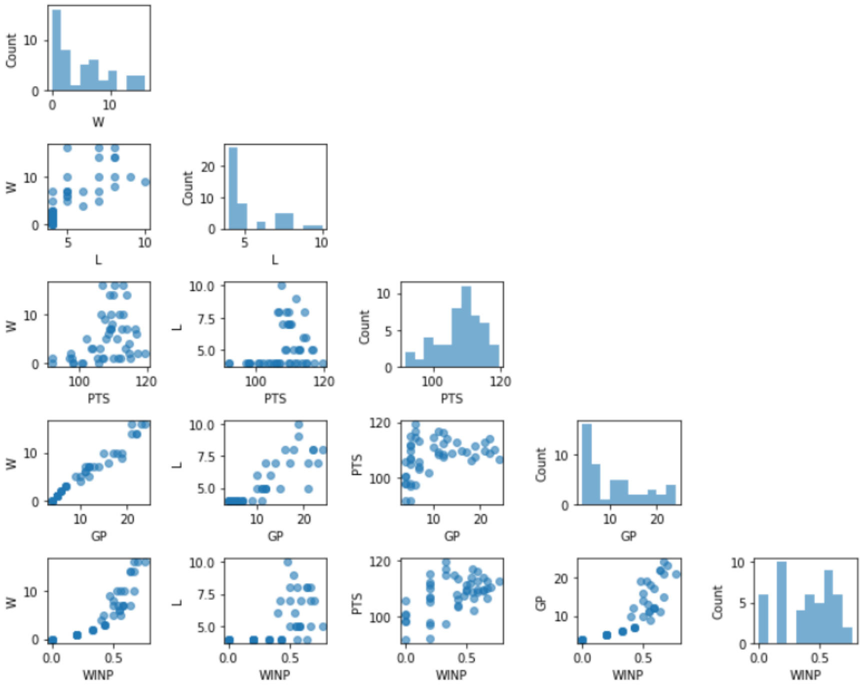


*Figure 24: An output showing the correlation of some of the variables between each other.*

Here you can see that, the wins and the winning percentage have a good correlation with each other.

Next, we will initialize the Lasso model and give it an alpha of 1.0.

With a coefficient of 0, our slope of 0.01 is present, when we are looking at the interception, we receive 0.41.

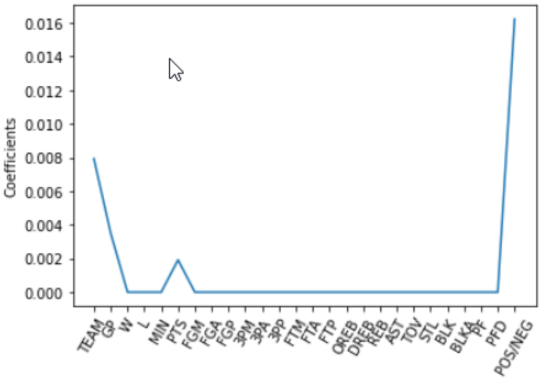
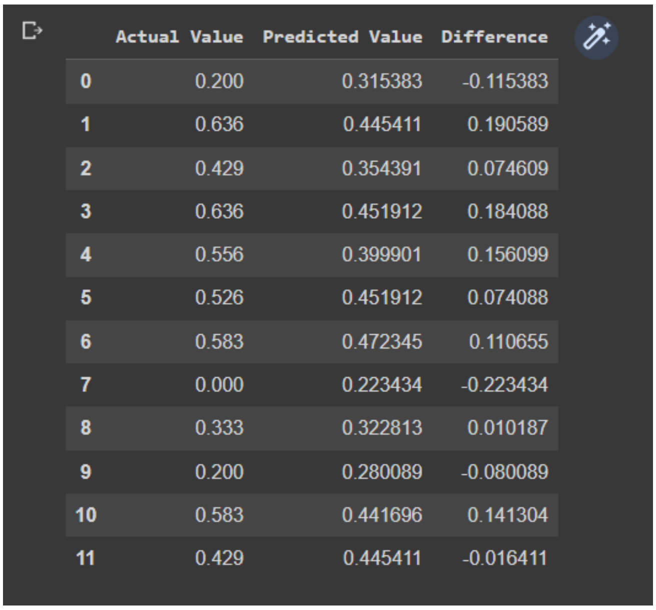


*Figure 25: This figure shows which variable is having the most effect on the winning percentage.*

Here this graph is telling us that the variables that had the most effect on our dependent variable was the games played, the pts, and the pos/neg.



| | Actual Value | Predicted Value | Difference |
|---|---|---|---|
| 0 | 0.200 | 0.315383 | -0.115383 |
| 1 | 0.636 | 0.445411 | 0.190589 |
| 2 | 0.429 | 0.354391 | 0.074609 |
| 3 | 0.636 | 0.451912 | 0.184088 |
| 4 | 0.556 | 0.399901 | 0.156099 |
| 5 | 0.526 | 0.451912 | 0.074088 |
| 6 | 0.583 | 0.472345 | 0.110655 |
| 7 | 0.000 | 0.223434 | -0.223434 |
| 8 | 0.333 | 0.322813 | 0.010187 |
| 9 | 0.200 | 0.280089 | -0.080089 |
| 10 | 0.583 | 0.441696 | 0.141304 |
| 11 | 0.429 | 0.445411 | -0.016411 |

*Figure 26: The predicted value of the winning percentage when using the LASSO Regression.*

These are the actual and predicted values.

## 3.5 Ridge Regression

When data frame suffers from the independent variables being highly correlated, and there is an issue of multicollinearity, we use a model tuning technique known as Ridge Regression. If multicollinearity were to come to pass within a given data, there is very high chance that the least-squares will be unbiased, and there will be a large variance. This means that the predicted values will a long distance from the real values.

Variance is how different a random variable is from an expected value, while bias is known as the amount of difference between the model's prediction and the target value. In another definition, variance is how much the target function estimate changes if another training data was given to it. High bias is when there are too many incorrect assumptions during the machine learning process. Overfitting is caused when a trained model does good on the training data but doesn't on the testing data.

Ridge Regression is machine learning algorithm that can be used to avoid overfitting. Overfitting happens when our model begins to learn the noises and the detail to the point that it has a negative impact on our model when it is given new data, it happens when our model models the training data too well.

In machine learning, what we seek is a model that can generalize patterns, this means it works best on both the training data and the test data.

Ridge Regression overcomes overfitting by applying a penalty term, which will reduce the weights and the biases.

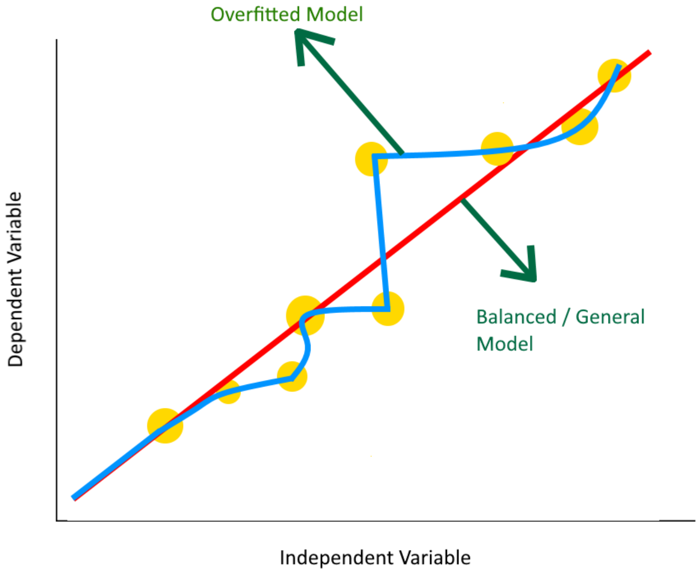


*Figure 27: The differences between an overfitted model and a balanced model. Where the overfitted model is represented by the blue line and the balanced model the red line.*

To improve the variance in a model, ridge regression increases the bias, this is done by changing the slope of the line. The model may do a little poor on the training dataset, but it will perform well on the testing dataset and the training as well. In the figure below, the least square regression managed to tackle all the training set, but it didn't tackle any of the training set. But when we used ridge regression it shifted our line of best fit in such a way that some of the testing set gets tackled with some of the training set.

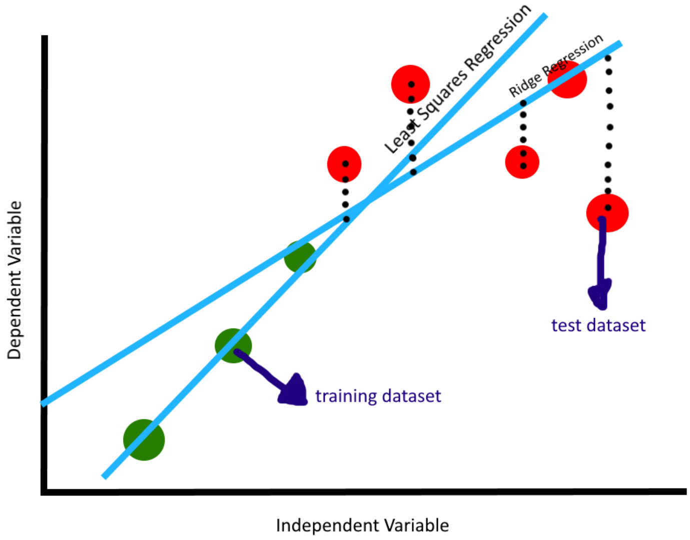


*Figure 28: When the Least Squares Regression is done, none of the rest dataset is recognized till we used the Ridge Regression causing a more balanced model.*

In our figure above the slope was able to be reduced because of the ridge regression penalty, and this causes it to more tolerant to the changes in the independent variable.

Ridge Regression:

Min(Sum of Squared Residuals + a * slope$^2$)

*Figure 29: The formula of the Ridge Regression*

In the graph below, what is been shown is as the alpha begin to increase, the slope of the ridge regression line becomes more horizonal, this means that the model is becoming less sensitive to the variation within the independent variables.

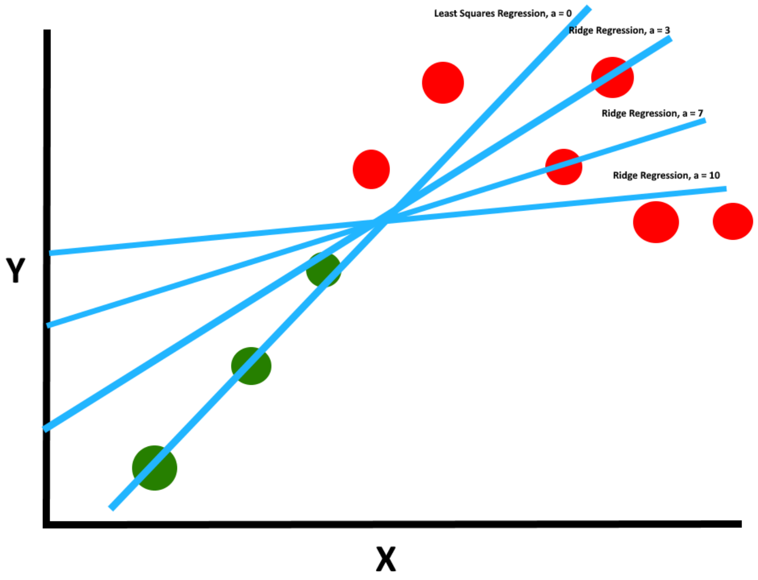


*Figure 30: As we increase the alpha of the ridge regression, our line of best fit becomes more horizontal.*

## 3.6 Feature Selection

Feature selection is a process in machine learning that aims to acquire the most consistent and most relevant features for usage in our machine learning model construction. The reasoning of why we use feature selection is the following:

1. To increase the predictive power of the algorithm by selecting the most important features and omitting the most irrelevant one.[22]
2. Shortening the times of training
3. Over-fitting is decreased, because of the unessential data which means there will be less chances of making decisions based off noise.

There are many types of Feature Selection methods, the first one is called as the Wrapper [22]methods. The Wrapper methods are used to train a model by using a subset of features. In Wrapper methods, we would look at the previous model, then after that decide if we should add or remove certain features. Forward selection, backward elimination, and recursive feature elimination are the different technique that are Wrapper methods.

Forward Selection:[22] In forward selection[22], we start with no features within the model, but as we iterate forward, after each repetition a new feature is added, and this is done until an addition of a new feature doesn't improve the performance of the model. [22]

Backward Selection: In backward selection, we begin with all the features, and after each iteration, the least important feature is removed, which improves the model.

Recursive Feature Elimination: By using optimization, this algorithm tries to find the best performing features. After each iteration[22], a new model is created and keeps aside the most significant feature and the less significant feature. A new model is then created using all the other features, the features are then ranked according to their elimination order.[22]

In our model, we decided to go with the Recursive Feature Elimination.

Another method type in Feature Selection is known as Filter methods, which is mainly used a step of preprocessing, and when they begin their selection, it won't be dependent on any machine learning algorithm. Along with correlation and the outcome variable, features are selected based off their scores by various statistical tests.

- Pearson's Correlation: Having an output that ranges from -1 to 1 and to quantify the linear dependence of two continuous variables, this method is used.

- LDA: To determine the linear combination of different features, we use a linear discriminant analysis that can differentiate between multiple categorical variables.

  Lastly, we have the Information Gain, which is the method, we decided to use for our feature selection.

  Information Gain is used to calculate the disorder within a system, which is known as entropy during the transformation of the dataset. In feature selection, we use this to evaluate the information gain of each of the variable.

For our models, we decided to use two feature selection technique to deduce which independent variable is good for our model, that is information gain, which is a filter method and recursive feature elimination, which is a wrapper method.

The reasoning of why the recursive feature elimination was selected is because of its popularity and because of its easy configuration process, utilization, it's effectiveness in selecting features within a training dataset. Information gain is good to check the reduction in entropy when the dataset is being transformed.

## 3.7 Prediction based off the Average

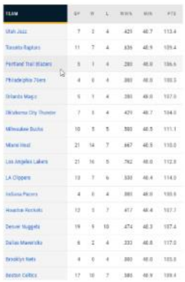
Originally in our results we took data of only one season, but the problem is that this does not give too much information. This is like making a conclusion towards an athlete based off one bad season that he may have and concluding that he isn't a good athlete, without considering his other seasons where he or she has done marvelous in his performance.

By taking three seasons or more, we are then increasing our sample size which leads to more accuracies and makes our results less bias. It makes it less bias because, we aren't concluding that a team will do well based off the one season he did in 2020-2021, but we are considering the other seasons as well. By doing this, we are also removing outliers as well, some of the teams in the NBA aren't necessary a playoff-making team, but because, it made it to the playoffs once by having only one season we conclude it is a playoff-making team, but really it is just an outlier if we were to take data of the past seasons.

So, the question here is what would happen if we were take the average of three seasons concerning the statistics of the NBA. This would allow us to have higher accuracy to what happened in the 2021-2022 NBA playoffs.

2018-2019

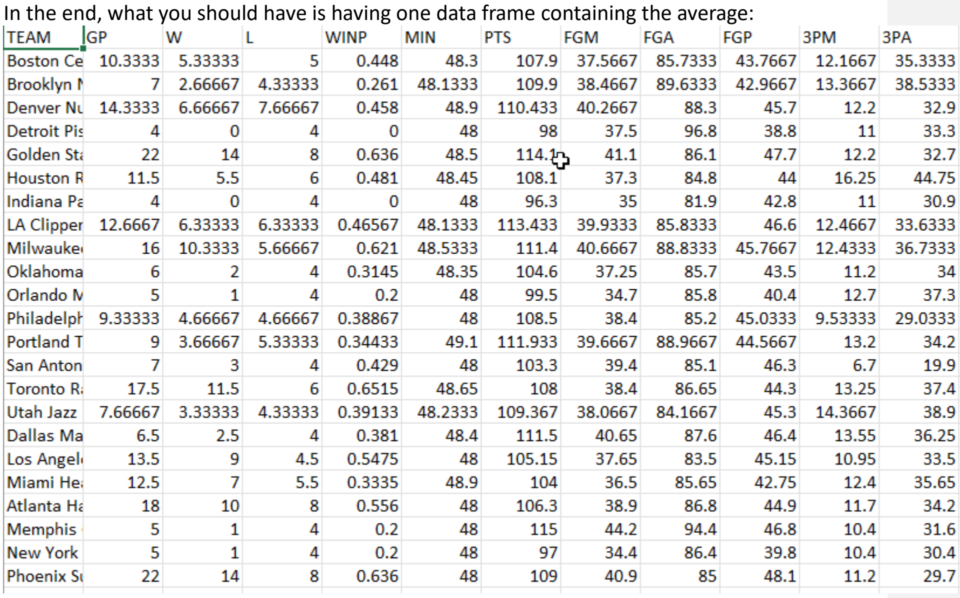
2019-2020



2020-2021

What we will do is we will take three different NBA seasons: 2018 to 2019, 2019 to 2020, and 2020-2021, we will then get the average turning it into a data frame of average of three NBA seasons, so we get the following:

We will be able to generate the average of these three seasons by using Python.
The first step is to gather the three seasons and add them together. This means taking all the values of each column of a season, and add them to the column of another season, after doing so, take the values of each of the column and divide them by three, thus this giving you the average.
In the end, what you should have is having one data frame containing the average:



| TEAM | GP | W | L | WINP | MIN | PTS | FGM | FGA | FGP | 3PM | 3PA |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Boston Ce | 10.3333 | 5.33333 | 5 | 0.448 | 48.3 | 107.9 | 37.5667 | 85.7333 | 43.7667 | 12.1667 | 35.3333 |
| Brooklyn N | 7 | 2.66667 | 4.33333 | 0.261 | 48.1333 | 109.9 | 38.4667 | 89.6333 | 42.9667 | 13.3667 | 38.5333 |
| Denver Nu | 14.3333 | 6.66667 | 7.66667 | 0.458 | 48.9 | 110.433 | 40.2667 | 88.3 | 45.7 | 12.2 | 32.9 |
| Detroit Pis | 4 | 0 | 4 | 0 | 48 | 98 | 37.5 | 96.8 | 38.8 | 11 | 33.3 |
| Golden Sta | 22 | 14 | 8 | 0.636 | 48.5 | 114.1 | 41.1 | 86.1 | 47.7 | 12.2 | 32.7 |
| Houston R | 11.5 | 5.5 | 6 | 0.481 | 48.45 | 108.1 | 37.3 | 84.8 | 44 | 16.25 | 44.75 |
| Indiana Pa | 4 | 0 | 4 | 0 | 48 | 96.3 | 35 | 81.9 | 42.8 | 11 | 30.9 |
| LA Clipper | 12.6667 | 6.33333 | 6.33333 | 0.46567 | 48.1333 | 113.433 | 39.9333 | 85.8333 | 46.6 | 12.4667 | 33.6333 |
| Milwauke | 16 | 10.3333 | 5.66667 | 0.621 | 48.5333 | 111.4 | 40.6667 | 88.8333 | 45.7667 | 12.4333 | 36.7333 |
| Oklahoma | 6 | 2 | 4 | 0.3145 | 48.35 | 104.6 | 37.25 | 85.7 | 43.5 | 11.2 | 34 |
| Orlando M | 5 | 1 | 4 | 0.2 | 48 | 99.5 | 34.7 | 85.8 | 40.4 | 12.7 | 37.3 |
| Philadelph | 9.33333 | 4.66667 | 4.66667 | 0.38867 | 48 | 108.5 | 38.4 | 85.2 | 45.0333 | 9.53333 | 29.0333 |
| Portland T | 9 | 3.66667 | 5.33333 | 0.34433 | 49.1 | 111.933 | 39.6667 | 88.9667 | 44.5667 | 13.2 | 34.2 |
| San Anton | 7 | 3 | 4 | 0.429 | 48 | 103.3 | 39.4 | 85.1 | 46.3 | 6.7 | 19.9 |
| Toronto R | 17.5 | 11.5 | 6 | 0.6515 | 48.65 | 108 | 38.4 | 86.65 | 44.3 | 13.25 | 37.4 |
| Utah Jazz | 7.66667 | 3.33333 | 4.33333 | 0.39133 | 48.2333 | 109.367 | 38.0667 | 84.1667 | 45.3 | 14.3667 | 38.9 |
| Dallas Ma | 6.5 | 2.5 | 4 | 0.381 | 48.4 | 111.5 | 40.65 | 87.6 | 46.4 | 13.55 | 36.25 |
| Los Angel | 13.5 | 9 | 4.5 | 0.5475 | 48 | 105.15 | 37.65 | 83.5 | 45.15 | 10.95 | 33.5 |
| Miami He | 12.5 | 7 | 5.5 | 0.3335 | 48.9 | 104 | 36.5 | 85.65 | 42.75 | 12.4 | 35.65 |
| Atlanta Ha | 18 | 10 | 8 | 0.556 | 48 | 106.3 | 38.9 | 86.8 | 44.9 | 11.7 | 34.2 |
| Memphis | 5 | 1 | 4 | 0.2 | 48 | 115 | 44.2 | 94.4 | 46.8 | 10.4 | 31.6 |
| New York | 5 | 1 | 4 | 0.2 | 48 | 97 | 34.4 | 86.4 | 39.8 | 10.4 | 30.4 |
| Phoenix Su | 22 | 14 | 8 | 0.636 | 48 | 109 | 40.9 | 85 | 48.1 | 11.2 | 29.7 |

# 4 Conclusion

## 4.1 Summary

## 4.2 Evaluation

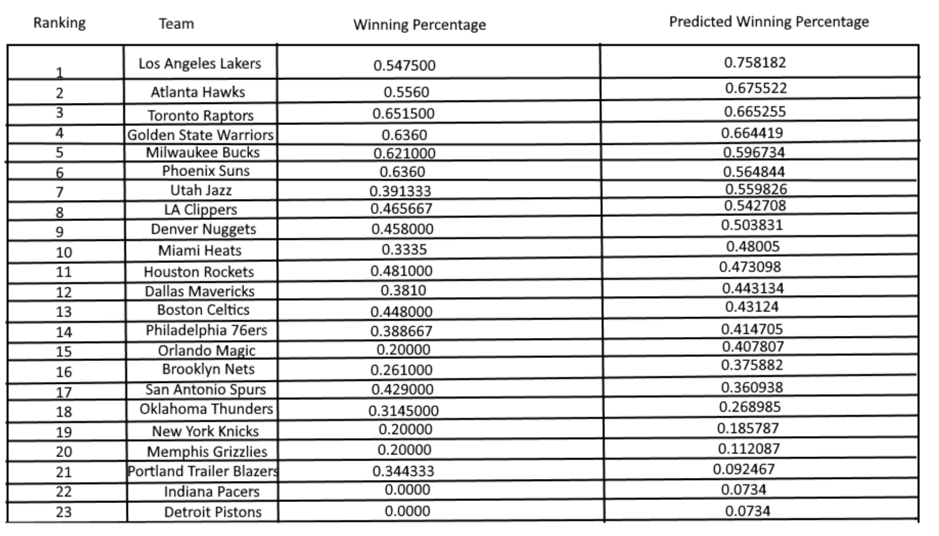
### 4.2.1 Results of the Linear Regression

4.2.1.1 Linear Regression Predicted Rankings

Ideally when it comes to linear regression within a perfect world, you want a line that is straight and can connect to all data point, if this was done, this means that our model is making perfect predictions. But in the case of the real world, you want a particular balance of where the model isn't too perfect, but it is still accurate. If our model however is too perfect than this is known as overfitting.

So, what we should be expecting within our results is a line that is connecting to almost all the data points, as mentioned in our methodology section this is known as the line of best fit.

We decided to test our data frame on the average of three seasons. Previously, we spoke on the methods and steps that was taken to achieve this, and the reasoning of why it was important to get these, but in this section, we will speak on what was the result.

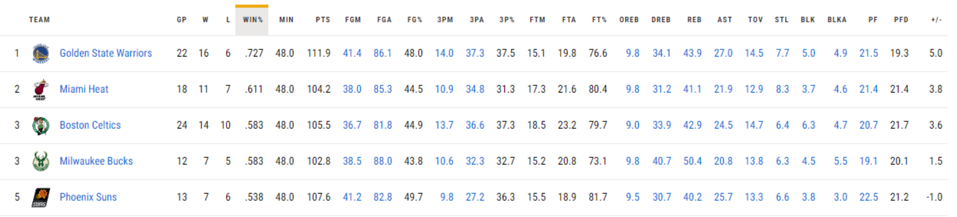


| Ranking | Team | Winning Percentage | Predicted Winning Percentage |
|---|---|---|---|
| 1 | Los Angeles Lakers | 0.547500 | 0.758182 |
| 2 | Atlanta Hawks | 0.5560 | 0.675522 |
| 3 | Toronto Raptors | 0.651500 | 0.665255 |
| 4 | Golden State Warriors | 0.6360 | 0.664419 |
| 5 | Milwaukee Bucks | 0.621000 | 0.596734 |
| 6 | Phoenix Suns | 0.6360 | 0.564844 |
| 7 | Utah Jazz | 0.391333 | 0.559826 |
| 8 | LA Clippers | 0.465667 | 0.542708 |
| 9 | Denver Nuggets | 0.458000 | 0.503831 |
| 10 | Miami Heats | 0.3335 | 0.48005 |
| 11 | Houston Rockets | 0.481000 | 0.473098 |
| 12 | Dallas Mavericks | 0.3810 | 0.443134 |
| 13 | Boston Celtics | 0.448000 | 0.43124 |
| 14 | Philadelphia 76ers | 0.388667 | 0.414705 |
| 15 | Orlando Magic | 0.20000 | 0.407807 |
| 16 | Brooklyn Nets | 0.261000 | 0.375882 |
| 17 | San Antonio Spurs | 0.429000 | 0.360938 |
| 18 | Oklahoma Thunders | 0.3145000 | 0.268985 |
| 19 | New York Knicks | 0.20000 | 0.185787 |
| 20 | Memphis Grizzlies | 0.20000 | 0.112087 |
| 21 | Portland Trailer Blazers | 0.344333 | 0.092467 |
| 22 | Indiana Pacers | 0.0000 | 0.0734 |
| 23 | Detroit Pistons | 0.0000 | 0.0734 |

The above table shows the results of what was predicted when using a Linear Regression model, according to our machine learning model, the Los Angeles Lakers had the highest chance of winning the NBA championship, while the Atlanta Hawks had the second highest chances of winning. Detroit Pistons and Indiana Pacers having the lowest percentage of winning were ranked the lowest.

#### 4.2.1.2 Comparing it to the 2021-2022 to judge its credibility

According to the 2021-2022 data frame, the top five highest percentage team were the Golden State Warriors, the Miami Heat, the Boston Celtics, and the Milwaukee Bucks, and Phoenix Suns. Compared to our model, the Golden State Warriors, and Milwaukee Bucks did make the top five team just not in the same order. The table below shows the data frame of the 2021-2022 Playoffs.



| | TEAM | GP | W | L | WIN% | MIN | PTS | FGM | FGA | FG% | 3PM | 3PA | 3P% | FTM | FTA | FT% | OREB | DREB | REB | AST | TOV | STL | BLK | BLKA | PF | PFD | +/- |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Golden State Warriors | 22 | 16 | 6 | .727 | 48.0 | 111.9 | 41.4 | 86.1 | 48.0 | 14.0 | 37.3 | 37.5 | 15.1 | 19.8 | 76.6 | 9.8 | 34.1 | 43.9 | 27.0 | 14.5 | 7.7 | 5.0 | 4.9 | 21.5 | 19.3 | 5.0 |
| 2 | Miami Heat | 18 | 11 | 7 | .611 | 48.0 | 104.2 | 38.0 | 85.3 | 44.5 | 10.9 | 34.8 | 31.3 | 17.3 | 21.6 | 80.4 | 9.8 | 31.2 | 41.1 | 21.9 | 12.9 | 8.3 | 3.7 | 4.6 | 21.4 | 21.4 | 3.8 |
| 3 | Boston Celtics | 24 | 14 | 10 | .583 | 48.0 | 105.5 | 36.7 | 81.8 | 44.9 | 13.7 | 36.6 | 37.3 | 18.5 | 23.2 | 79.7 | 9.0 | 33.9 | 42.9 | 24.5 | 14.7 | 6.4 | 6.3 | 4.7 | 20.7 | 21.7 | 3.6 |
| 3 | Milwaukee Bucks | 12 | 7 | 5 | .583 | 48.0 | 102.8 | 38.5 | 88.0 | 43.8 | 10.6 | 32.3 | 32.7 | 15.2 | 20.8 | 73.1 | 9.8 | 40.7 | 50.4 | 20.8 | 13.8 | 6.3 | 4.5 | 5.5 | 19.1 | 20.1 | 1.5 |
| 5 | Phoenix Suns | 13 | 7 | 6 | .538 | 48.0 | 107.6 | 41.2 | 82.8 | 49.7 | 9.8 | 27.2 | 36.3 | 15.5 | 18.9 | 81.7 | 9.5 | 30.7 | 40.2 | 25.7 | 13.3 | 6.6 | 3.8 | 3.0 | 22.5 | 21.2 | -1.0 |

Our linear regression model predicts that top five teams that has the lowest chances of winning the NBA were the New York Knicks, Memphis Grizzlies, the Portland Trail Blazers, the Indiana Pacers, and the Detroit Pistons.

According to the 2021-2022 data frame, the Brooklyn Nets, the Denver Nuggets, the Chicago Bulls, the Atlanta Hawks, and the Utah Jazz were the lowest teams. During the 2021-2022 playoffs, only the Memphis Grizzlies from our model made it to the playoffs, the rest didn't have the number of wins that was required to make it to the playoffs.

From what we can judge from our averaged values of three seasons, and the 2021-2022 data frame, our model isn't accurate in predicting its values. They are different then the 2021 to 2022.

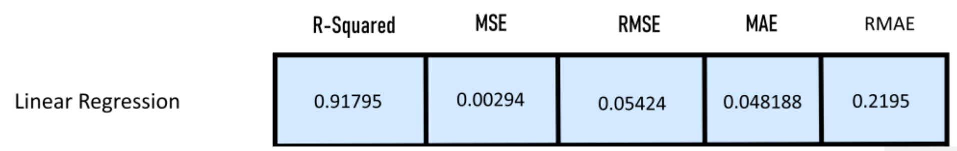
#### 4.2.1.3 The Performance of Our Model

To judge the performance of our model, we utilize a system called the metrics. Metrics are normally used to check the performance of our model. We explained previously what metrics are. So, as we check to see the performance of our model, we will be checking the R-Squared, the mean squared error, the mean absolute error, and the root mean squared error, the root means absolute error.

The score that we got concerning the R-Squared was 0.917, this means that the independent and the dependent variable of our model are highly correlated with eachother. In basketball, everything that you do counts, none of the variables in basketball would be insignificant to our dependent variable. By having a R-Square of 91%, this tells how the percentage of how much the winning percentage is being affected by selected actions during a basketball match.

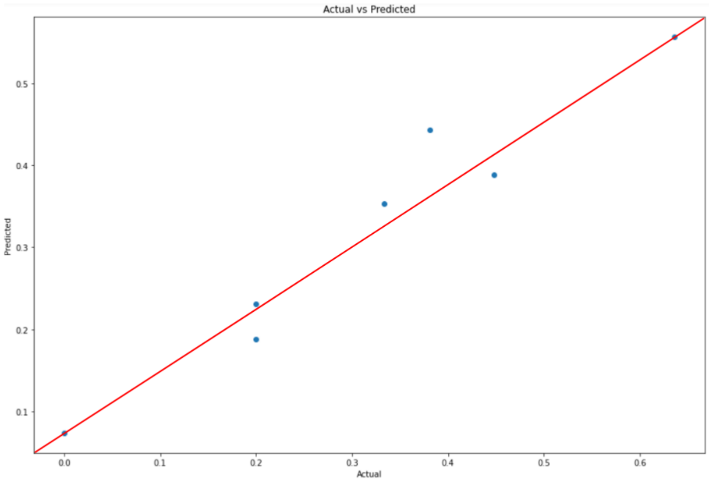
The score that we got for the mean squared error was 0.00294, this tells us that our data points are extremely close to our regression line. Because our mean squared error is as low as it is, we have a better forecast. Because our mean squared error is less, we have a small error, and this gives us a better estimator. The rooted mean squared error was 0.05.

Our score concerning the mean absolute error was 0.048188 which means that our linear regression model fits the basketball data well. It gives us the mean absolute difference between the predicted value and actual values within our data. The root mean absolute error was 0.21.



| | R-Squared | MSE | RMSE | MAE | RMAE |
|---|---|---|---|---|---|
| Linear Regression | 0.91795 | 0.00294 | 0.05424 | 0.048188 | 0.2195 |

#### 4.2.1.4 The Line of Best Fit



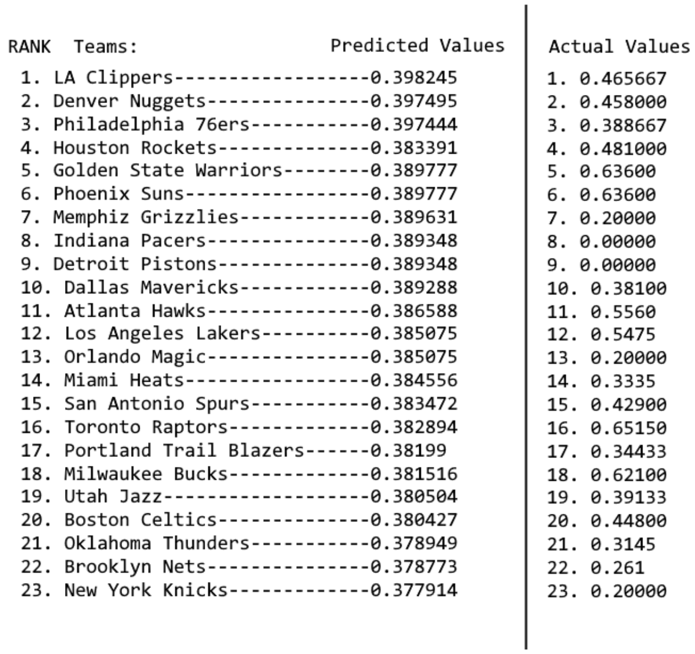
The data points in our graph are close to our regression line, this means that that the line is doing a good job at summarizing the pattern of the data points between the predicted values of dependent variables, and the actual value of dependent variable. Since our line is closer to our data points, this means that, our model is giving off good predictions. The predicted y-value from the line is close to the actual y-value for the data points.

### 4.2.2 Results of the Artificial Neural Network

#### 4.2.2.1 Expectation due to using a nonlinear regression

In our artificial neural network, we will be judging whether this is a good model for our NBA data frame based off the results and metrics, we receive. The difference between an artificial neural network and a regression method is that with an artificial neural network, it can perform better with non-linear dependencies. Regression models deal with linear dependencies better. The expectation for the artificial neural network is that it won't really perform well as compared to the other model because within our linear regression model, our dependent and independent variable were highly correlated, and our predictions were good. With that information it is safe to assume that the NBA data frame is that which is linear. Since the artificial neural network deals mainly with nonlinear dependencies, we can expect many errors or questionable results to occur.

4.2.2.2 Artificial Neural Network Predicted Ranking

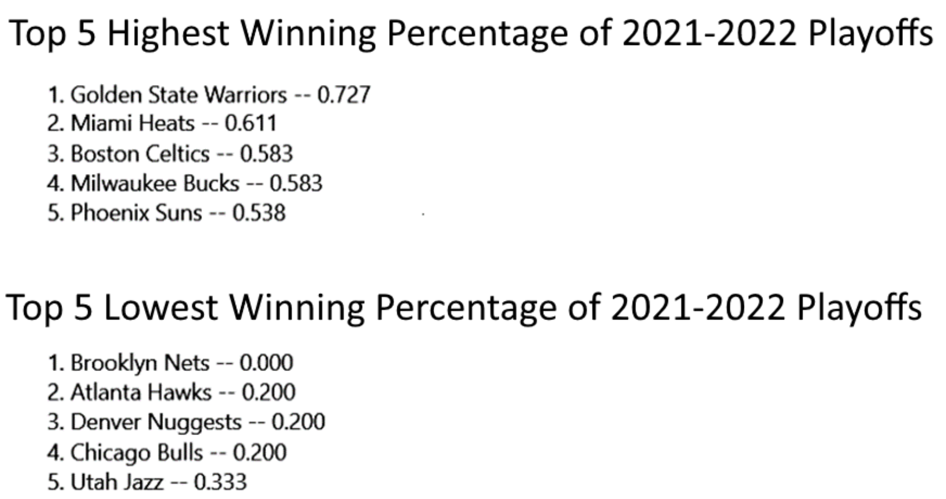


```
RANK  Teams:                Predicted Values  |  Actual Values
 1. LA Clippers------------------0.398245      |  1. 0.465667
 2. Denver Nuggets---------------0.397495      |  2. 0.458000
 3. Philadelphia 76ers-----------0.397444      |  3. 0.388667
 4. Houston Rockets--------------0.383391      |  4. 0.481000
 5. Golden State Warriors--------0.389777      |  5. 0.63600
 6. Phoenix Suns-----------------0.389777      |  6. 0.63600
 7. Memphiz Grizzlies------------0.389631      |  7. 0.20000
 8. Indiana Pacers---------------0.389348      |  8. 0.00000
 9. Detroit Pistons--------------0.389348      |  9. 0.00000
 10. Dallas Mavericks------------0.389288      |  10. 0.38100
 11. Atlanta Hawks---------------0.386588      |  11. 0.5560
 12. Los Angeles Lakers----------0.385075      |  12. 0.5475
 13. Orlando Magic---------------0.385075      |  13. 0.20000
 14. Miami Heats-----------------0.384556      |  14. 0.3335
 15. San Antonio Spurs-----------0.383472      |  15. 0.42900
 16. Toronto Raptors-------------0.382894      |  16. 0.65150
 17. Portland Trail Blazers------0.38199       |  17. 0.34433
 18. Milwaukee Bucks-------------0.381516      |  18. 0.62100
 19. Utah Jazz-------------------0.380504      |  19. 0.39133
 20. Boston Celtics--------------0.380427      |  20. 0.44800
 21. Oklahoma Thunders-----------0.378949      |  21. 0.3145
 22. Brooklyn Nets---------------0.378773      |  22. 0.261
 23. New York Knicks-------------0.377914      |  23. 0.20000
```

As suspected, we can see that there are some errors within our prediction, and we can see that our predictions are from off from the actual values. As we can see from our figure, the top five team were the LA Clippers, the Denver Nuggets, the Houston Rockets, and the Golden State Warriors, if we add up all the values together and try to get the average, we receive an average of 0.3932704, which tells us the central tendency of the values. The central tendency tells us that these values are almost repetitive, as we receive three team with a 0.39. This model is telling us that all the teams almost all have an even chance.

A person who watches a basketball will be able to suspect the failure within this model by just looking at teams that has no potential according to their recent statistic and be able to deduce that this model isn't good for NBA data, Detroit Pistons isn't a skilled team yet in our model IT predicted that it has a higher winning percentage then the Lakers, Brooklyn Nets, and Boston Celtics, which are highly regarded teams within the NBA.

4.2.2.3 Comparing it to the 2021-2022 NBA Data



Top 5 Highest Winning Percentage of 2021-2022 Playoffs

1. Golden State Warriors -- 0.727
2. Miami Heats -- 0.611
3. Boston Celtics -- 0.583
4. Milwaukee Bucks -- 0.583
5. Phoenix Suns -- 0.538

Top 5 Lowest Winning Percentage of 2021-2022 Playoffs

1. Brooklyn Nets -- 0.000
2. Atlanta Hawks -- 0.200
3. Denver Nuggests -- 0.200
4. Chicago Bulls -- 0.200
5. Utah Jazz -- 0.333

When we compare our predicted values of the top five highest rankings, we can see that the Golden State Warriors were the only team that was the top five within both our model's predicted value and the 2021-2022. We can say that, even within a failed model, the Golden State Warriors is a basketball team that still stands strong within being the top best teams of the NBA.  2021-2022 was a rough year for the Brooklyn Nets, a very skilled basketball team that has very skilled players such as James Harden, Kyrie Irving, and Kevin Durant, but even after these players, the playoffs were rough. During the 2021-2022, they had a winning percentage of 0.000, meaning they won no games, yet lost four games, the moment they entered the playoffs. Our model did predict the Brooklyn Nets as being the last five team to have the lowest chance of making it to the Playoffs. So, although, in terms of winning percentage, our model didn't do a good job at predicting that. But when it came to being the last five teams of making it to the playoffs. It did do that. So, our model isn't terrible.

4.2.2.4 Why is it possible that the Artificial Neural didn't predict so well?

As mentioned earlier, the artificial neural network is a nonlinear statistical model is supposed to demonstrate complex relationship between the input and the output. One article mentions, that one of the disadvantages of the artificial neural network is its black box nature, this means that we are unable to know how or why a neural network would come up with a certain output. In Neural Network, it is possible to feed in images, an example of that would be a dog, but it is possible that a neural network may predict it to be a bird, and there would be no clue on to how our artificial neural network arrived at this point.

Another reason as to why our neural network didn't predict so well is because of the lack of data within our model, it is possible that if we had put my independent variables then the number we already have then we would possibly would have had a more accurate model, that is because neural networks need more data than the traditional machine learning algorithms, usually machine learning problem can be solved using less amount of data, this is why our Linear Regression, Random Forest Regression, etc did better than the artificial neural network. This is just a few of the things of why our model didn't do so well. There are other drawbacks of the artificial network such as the duration of development and it being computationally expensive.

#### 4.2.2.5 The Performance of Our Model

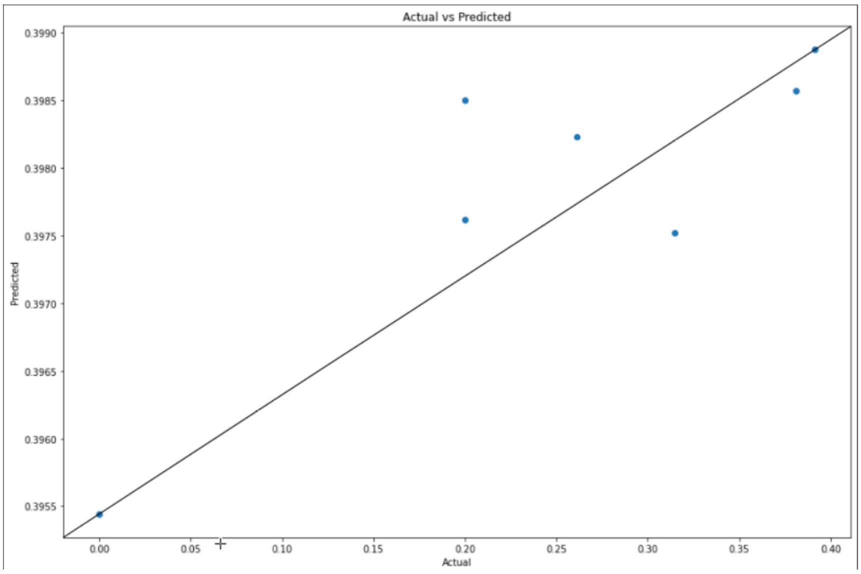
The Mean Squared Error of our model is 0.0372, although the Artificial Neural Network didn’t perform well in predicting variables that is similar or close to the ones in 2021-2022. Our program states that the forecasting of our model is good, and that it doesn’t have much error. By having this amount of score concerning the mean square error, this indicates that the number of errors within our model is about 3%. The root mean squared error of our model is 0.19, which means that our model can fit a dataset well.

The R-Square of our model is -1.395, and if we look at the figure below this is reasonable because our model most likely does not believe that the independent variable is causing a big effect on the variation of the winning percentage.

The artificial neural network is a model that seeks to have more data, and the less amount of data that was given is most likely the cause of the low amount of r-squared value. A way to possibly fix this is by adding more non-correlated independent variables to our model. A small R-Square doesn’t mean that our model is bad.

For the mean absolute error, the score ended up being 0.1481, such low mean absolute error indicates that our prediction accuracy is good. But compared to the 2021-2022, the prediction accuracy didn’t do too well in predicting the future games.

#### 4.2.2.6 The Line of Best Fit



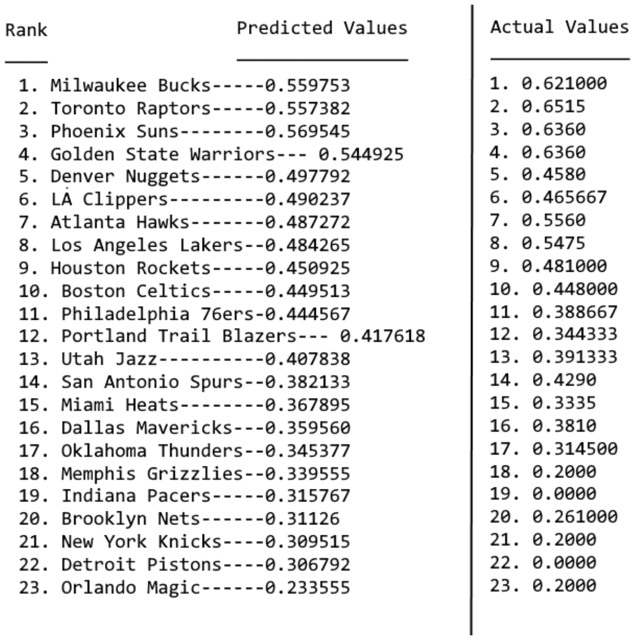
The data points of our model aren’t too close to our line of best fit as compared to our other machine learning models except for one data point but they aren’t extremely far away either. Although, our model didn’t do well in predicting the actual future value of the 2021-2022 playoffs, the model isn’t extremely bad. Artificial Neural Network shouldn’t be used if there isn’t a lot of independent variables, it is also possible that our model didn’t go through enough learning process, which caused it to not get the output that we want it.

The Artificial Neural Network is a machine learning model that constantly learns, and the amount of learning it does is dependent on the amount of batch and epoch it is given.

### **4.2.3 Results of Random Forest Random**

4.2.3.1 Random Forest Regression Predicted Rankings

In domain of machine learning engineering, the random forest algorithm is one of the best-known algorithms and is also known as to be one of the best machine learning models to use. It uses multiple decision trees, gets an output from each of those decision trees, and then gets the average of them. It can be used for both regression and classification. Our expectation of the results then should be a model that will perform well on its prediction and has an almost accurate rankings of the NBA teams when we compare it to the 2021-2022 data that was taken from the NBA website.

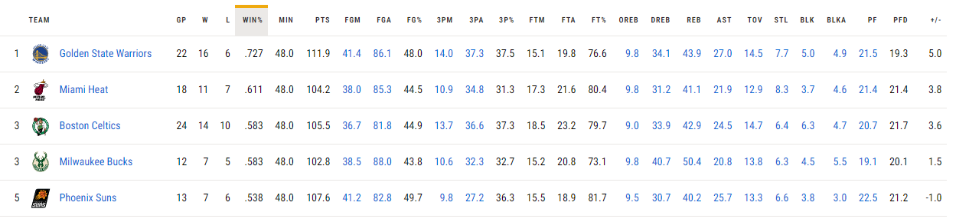


```
Rank                 Predicted Values              | Actual Values
____                 ________________              | _____________

 1. Milwaukee Bucks-----0.559753                   | 1. 0.621000
 2. Toronto Raptors-----0.557382                   | 2. 0.6515
 3. Phoenix Suns--------0.569545                   | 3. 0.6360
 4. Golden State Warriors--- 0.544925              | 4. 0.6360
 5. Denver Nuggets------0.497792                   | 5. 0.4580
 6. LA Clippers---------0.490237                   | 6. 0.465667
 7. Atlanta Hawks-------0.487272                   | 7. 0.5560
 8. Los Angeles Lakers--0.484265                   | 8. 0.5475
 9. Houston Rockets-----0.450925                   | 9. 0.481000
 10. Boston Celtics-----0.449513                   | 10. 0.448000
 11. Philadelphia 76ers-0.444567                   | 11. 0.388667
 12. Portland Trail Blazers--- 0.417618            | 12. 0.344333
 13. Utah Jazz---------0.407838                    | 13. 0.391333
 14. San Antonio Spurs--0.382133                   | 14. 0.4290
 15. Miami Heats--------0.367895                   | 15. 0.3335
 16. Dallas Mavericks---0.359560                   | 16. 0.3810
 17. Oklahoma Thunders--0.345377                   | 17. 0.314500
 18. Memphis Grizzlies--0.339555                   | 18. 0.2000
 19. Indiana Pacers-----0.315767                   | 19. 0.0000
 20. Brooklyn Nets------0.31126                    | 20. 0.261000
 21. New York Knicks----0.309515                   | 21. 0.2000
 22. Detroit Pistons----0.306792                   | 22. 0.0000
 23. Orlando Magic------0.233555                   | 23. 0.2000
```

Comparing this the artificial neural network, it can be said, that the Random Forest Regression gave off a much better prediction, putting the very skilled NBA team at their expected positioning and the not-so-skilled NBA team at their positioning. When we look at the difference between some of the value, we can see that they aren't far from each other, so we can assume that our forecasting accuracy will be great concerning the Random Forest Regression. The teams that have the highest chance of winning the NBA championship are the Milwaukee Bucks, who won the 2020-2021 NBA championship. The Toronto Raptors who won the 2018-2019 championship. The Phoenix Suns who although never won a championship but were close to winning a championship in the year of 2021. However, our team that has had a dynasty in previous years, and was the winner of the recent 2021-2022, Golden State Warriors came out to be fourth place in our rankings. Lastly is the Denver Nuggets, who's closest they came to winning a championship was 2019-2020.

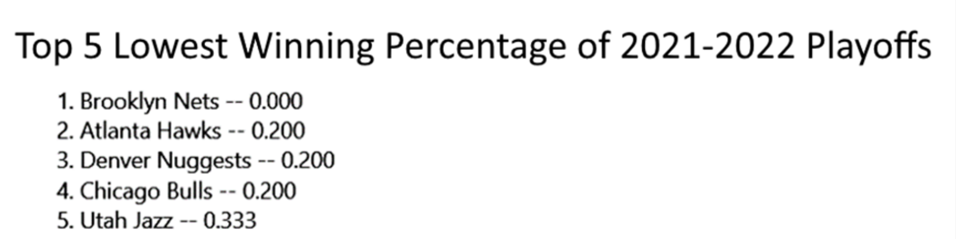
#### 4.2.3.2 Comparing our predictions to the 2021-2022 Recent NBA Playoffs



| | TEAM | GP | W | L | WIN% | MIN | PTS | FGM | FGA | FG% | 3PM | 3PA | 3P% | FTM | FTA | FT% | OREB | DREB | REB | AST | TOV | STL | BLK | BLKA | PF | PFD | +/- |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Golden State Warriors | 22 | 16 | 6 | .727 | 48.0 | 111.9 | 41.4 | 86.1 | 48.0 | 14.0 | 37.3 | 37.5 | 15.1 | 19.8 | 76.6 | 9.8 | 34.1 | 43.9 | 27.0 | 14.5 | 7.7 | 5.0 | 4.9 | 21.5 | 19.3 | 5.0 |
| 2 | Miami Heat | 18 | 11 | 7 | .611 | 48.0 | 104.2 | 38.0 | 85.3 | 44.5 | 10.9 | 34.8 | 31.3 | 17.3 | 21.6 | 80.4 | 9.8 | 31.2 | 41.1 | 21.9 | 12.9 | 8.3 | 3.7 | 4.6 | 21.4 | 21.4 | 3.8 |
| 3 | Boston Celtics | 24 | 14 | 10 | .583 | 48.0 | 105.5 | 36.7 | 81.8 | 44.9 | 13.7 | 36.6 | 37.3 | 18.5 | 23.2 | 79.7 | 9.0 | 33.9 | 42.9 | 24.5 | 14.7 | 6.4 | 6.3 | 4.7 | 20.7 | 21.7 | 3.6 |
| 3 | Milwaukee Bucks | 12 | 7 | 5 | .583 | 48.0 | 102.8 | 38.5 | 88.0 | 43.8 | 10.6 | 32.3 | 32.7 | 15.2 | 20.8 | 73.1 | 9.8 | 40.7 | 50.4 | 20.8 | 13.8 | 6.3 | 4.5 | 5.5 | 19.1 | 20.1 | 1.5 |
| 5 | Phoenix Suns | 13 | 7 | 6 | .538 | 48.0 | 107.6 | 41.2 | 82.8 | 49.7 | 9.8 | 27.2 | 36.3 | 15.5 | 18.9 | 81.7 | 9.5 | 30.7 | 40.2 | 25.7 | 13.3 | 6.6 | 3.8 | 3.0 | 22.5 | 21.2 | -1.0 |

Our algorithm of random forest regression was excellent in get close to predicting a couple of things, so far, other than the Linear Regression, and the Artificial Neural Network, we can say that the Random Forest Regression is the most reliable machine learning model. When it came to predicting that the teams of Golden State Warriors, Milwaukee Bucks, and the Phoenix Suns would make the top five of making to the NBA playoffs and having a good chance of winning the playoffs it did well.

In our algorithm, the Random Forest Regression predicted that the Milwaukee Bucks would have a winning percentage of 0.559, and in the 2021-2022 Recent NBA Playoffs they ended up getting a 0.583 winning percentage, if we subtract the differences, we get 0.024 as the values, indicating that there is only 2% difference between the two-winning percentage. So, in terms of almost accurately predicting Milwaukee Bucks winning percentage for the recent NBA playoffs it did well. The same also for also predicting for Phoenix Suns, where the difference was only a 3% difference because the Phoenix Suns according to our model had a 0.569 winning percentage, and in the 2021-2022 recent NBA playoffs it had 0.538 winning percentages, an excellent prediction of winning percentage between the two teams of Phoenix Suns and Milwaukee Bucks'.



Top 5 Lowest Winning Percentage of 2021-2022 Playoffs

1. Brooklyn Nets -- 0.000
2. Atlanta Hawks -- 0.200
3. Denver Nuggests -- 0.200
4. Chicago Bulls -- 0.200
5. Utah Jazz -- 0.333

One team make according to our model did make top five team of losing the NBA championship, and that is the Brooklyn Nets who according to our model had a winning percentage of 0.3611.

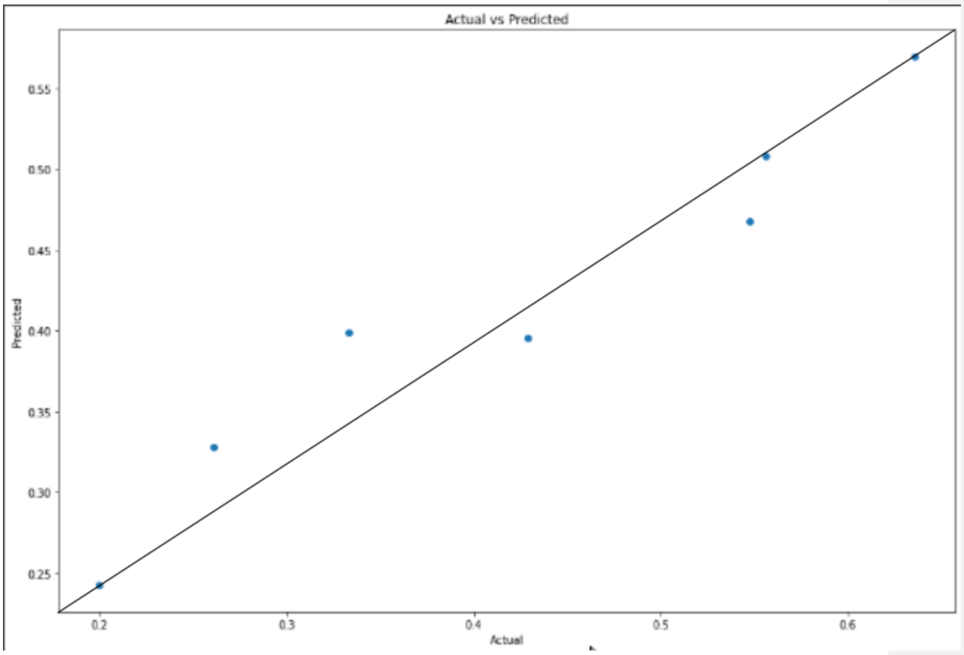
#### 4.2.3.3 The Performance of Our Model

As always, to judge the performance of our models, we use metrics. Our R-Square had a score of 0.85, showing that our independent variables and dependent variables are highly correlated with each other's. Because of this, we can expect that the line of best fit will be passing through many of the data points. Our mean squared error had score of 0.003 showing that our model is good and does not have to many errors within its predictions, indicating we have a good forecasting, and it shows that we should be expecting a regression line close to our data points.

The Mean Absolute Error was 0.05 showing that on average, the forecast distance from actual value is 0.05. Showing that our model did well in trying to predict the actual values. It also tells us the amount of error in our measurement. We can see that we had error of only 5% according to the mean absolute error.

The Mean Absolute Percentage Error was 0.11, Because the score of the MAPE is close to zero, the predictions were good, and our model is accurate. If we get the square root of all the scores, we get the following: 0.05 as the root mean squared error. 0.22 as the Root mean absolute error, and 0.344 as the root mean absolute percentage error.

4.2.3.4 Line of Best Fit

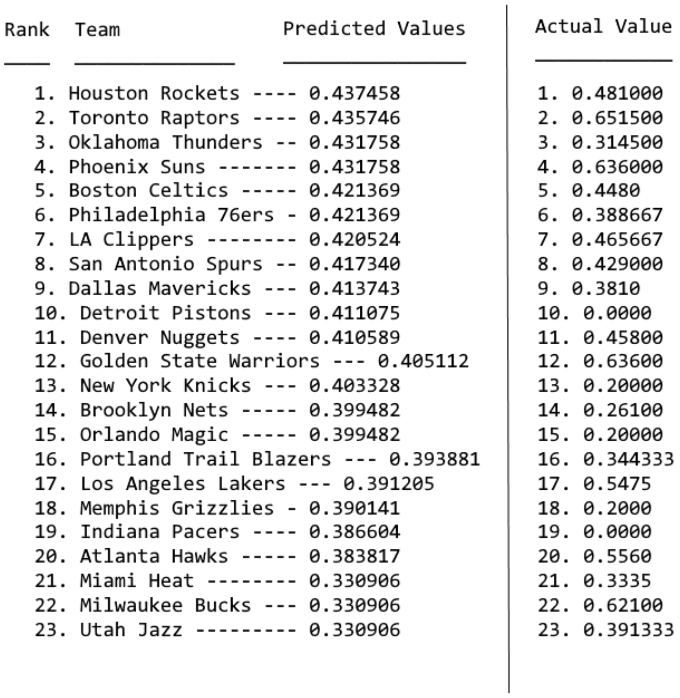


As suspected, our data point is close to our regression line, showing how well our model is in its prediction. So far, the Random Forest Regression is the best machine learning model. The R-Square told us how close our data point was to our regression.

### 4.2.4 Results of LASSO Regression

The objective of our LASSO regression model is to gather any predictors that minimized the prediction error for a quantitative response variable by putting some constraints on the model, the LASSO regression coefficient for some of the variable will be reduce to closely zero. Our expectation for the LASSO regression is a better prediction accuracy because the regularization of the lasso regression model helps to increase the model interpretation. We used the LASSO regression to see how our data frame will react when it is input into this model, and to deal with any multicollinearity within our variables.

#### 4.2.4.1 LASSO Regression Predicted Rankings



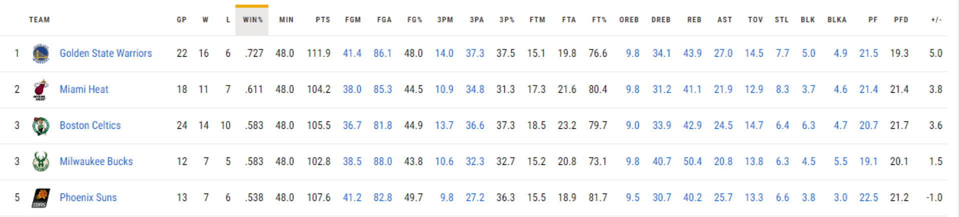
```
Rank   Team                  Predicted Values   |   Actual Value
____   ______________        ________________   |   ____________

   1. Houston Rockets ---- 0.437458             |   1. 0.481000
   2. Toronto Raptors ---- 0.435746             |   2. 0.651500
   3. Oklahoma Thunders -- 0.431758             |   3. 0.314500
   4. Phoenix Suns ------- 0.431758             |   4. 0.636000
   5. Boston Celtics ----- 0.421369             |   5. 0.4480
   6. Philadelphia 76ers - 0.421369             |   6. 0.388667
   7. LA Clippers -------- 0.420524             |   7. 0.465667
   8. San Antonio Spurs -- 0.417340             |   8. 0.429000
   9. Dallas Mavericks --- 0.413743             |   9. 0.3810
   10. Detroit Pistons --- 0.411075             |   10. 0.0000
   11. Denver Nuggets ---- 0.410589             |   11. 0.45800
   12. Golden State Warriors --- 0.405112       |   12. 0.63600
   13. New York Knicks --- 0.403328             |   13. 0.20000
   14. Brooklyn Nets ----- 0.399482             |   14. 0.26100
   15. Orlando Magic ----- 0.399482             |   15. 0.20000
   16. Portland Trail Blazers --- 0.393881      |   16. 0.344333
   17. Los Angeles Lakers --- 0.391205          |   17. 0.5475
   18. Memphis Grizzlies - 0.390141             |   18. 0.2000
   19. Indiana Pacers ---- 0.386604             |   19. 0.0000
   20. Atlanta Hawks ----- 0.383817             |   20. 0.5560
   21. Miami Heat -------- 0.330906             |   21. 0.3335
   22. Milwaukee Bucks --- 0.330906             |   22. 0.62100
   23. Utah Jazz --------- 0.330906             |   23. 0.391333
```

What we can observe in our database is that our predicted values are between 0.330906 to 0.43758. From these predicted values, it can be said that this model isn't too bad, but it certainly isn't a reliable model when it comes to predicting values. Golden State Warriors being lower than majority isn't likely, unless you remove star players such as Stephen Curry, Klay Thompson, and Draymond Green, and replace them with unskilled player from other teams such as the Indiana Pacers, the chances of Golden State Warriors being ranked number 12 in the chances of winning the NBA championship is not at all likely.

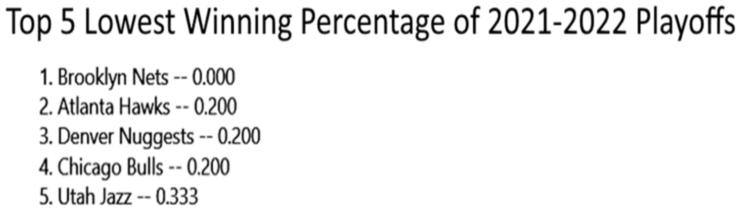
Another issue with our prediction is that a lot of the teams that have high winning percentage were ranked low, and their winning percentage were reduced to an unreasonable amount indicating that our LASSO regression isn't reliable.

#### 4.2.4.2 Comparing our predictions to the 2021-2022 Playoffs



| | TEAM | GP | W | L | WIN% | MIN | PTS | FGM | FGA | FG% | 3PM | 3PA | 3P% | FTM | FTA | FT% | OREB | DREB | REB | AST | TOV | STL | BLK | BLKA | PF | PFD | +/- |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Golden State Warriors | 22 | 16 | 6 | .727 | 48.0 | 111.9 | 41.4 | 86.1 | 48.0 | 14.0 | 37.3 | 37.5 | 15.1 | 19.8 | 76.6 | 9.8 | 34.1 | 43.9 | 27.0 | 14.5 | 7.7 | 5.0 | 4.9 | 21.5 | 19.3 | 5.0 |
| 2 | Miami Heat | 18 | 11 | 7 | .611 | 48.0 | 104.2 | 38.0 | 85.3 | 44.5 | 10.9 | 34.8 | 31.3 | 17.3 | 21.6 | 80.4 | 9.8 | 31.2 | 41.1 | 21.9 | 12.9 | 8.3 | 3.7 | 4.6 | 21.4 | 21.4 | 3.8 |
| 3 | Boston Celtics | 24 | 14 | 10 | .583 | 48.0 | 105.5 | 36.7 | 81.8 | 44.9 | 13.7 | 36.6 | 37.3 | 18.5 | 23.2 | 79.7 | 9.0 | 33.9 | 42.9 | 24.5 | 14.7 | 6.4 | 6.3 | 4.7 | 20.7 | 21.7 | 3.6 |
| 3 | Milwaukee Bucks | 12 | 7 | 5 | .583 | 48.0 | 102.8 | 38.5 | 88.0 | 43.8 | 10.6 | 32.3 | 32.7 | 15.2 | 20.8 | 73.1 | 9.8 | 40.7 | 50.4 | 20.8 | 13.8 | 6.3 | 4.5 | 5.5 | 19.1 | 20.1 | 1.5 |
| 5 | Phoenix Suns | 13 | 7 | 6 | .538 | 48.0 | 107.6 | 41.2 | 82.8 | 49.7 | 9.8 | 27.2 | 36.3 | 15.5 | 18.9 | 81.7 | 9.5 | 30.7 | 40.2 | 25.7 | 13.3 | 6.6 | 3.8 | 3.0 | 22.5 | 21.2 | -1.0 |

In our model the team that won the NBA championship was the Houston Rockets have a winning percentage of 0.437458. According to the data of the 2021-2022 playoffs, the Houston Rockets didn't make it to the playoffs. However, the Boston Celtics and the Phoenix Sun as predicted in our LASSO regression model did make the top five team with highest winning percentage, and in the 2021-2022, they made it in the top five as well.



Top 5 Lowest Winning Percentage of 2021-2022 Playoffs

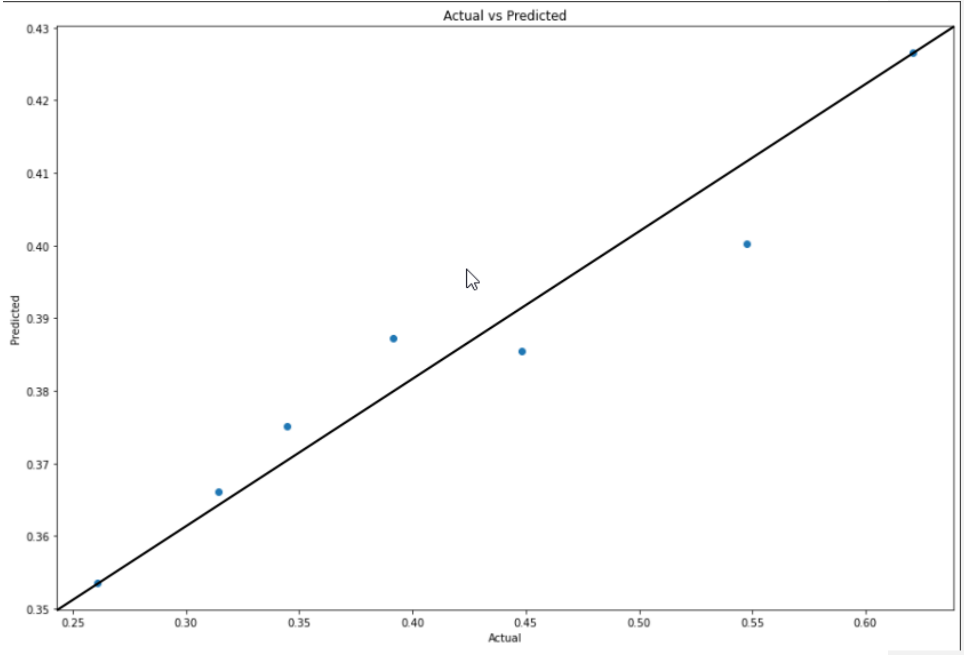
1. Brooklyn Nets -- 0.000
2. Atlanta Hawks -- 0.200
3. Denver Nuggests -- 0.200
4. Chicago Bulls -- 0.200
5. Utah Jazz -- 0.333

Utah Jazz was the lowest team of winning the NBA championship, and according to the 2021-2022 playoffs, they were also the lowest team.

In conclusion to this study, we can conclude that our LASSO regression model wasn't a reliable model in predicting who win the 2021-2022 playoffs, but it was reliable in selection Boston Celtics and the Phoenix Suns as the teams to make the top five of having the highest chance of winning the championship.

### 4.2.4.3 The Performance

According to our metrics, our model had an R-Square of 0.24, which means that the independent variables and dependent variable doesn't have a good correlation. The mean squared error is 0.010 which mean that there isn't that much error within our model, and our prediction accuracy is good. The mean absolute error is 0.083 which mean that actual value and the predicted value aren't far away from eachother. The mean absolute percentage error measures the accuracy of our model's forecast, the score for it was 0.19, which means we don't have a lot of error which is comes to prediction accuracy. The lower our Mean Absolute Percentage Error means that our data mining method's performance is good.

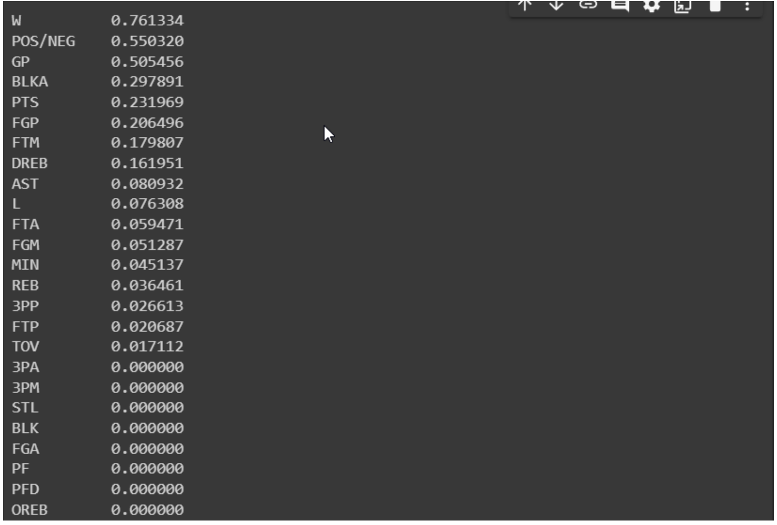


Our data points aren't too far away from our regression line, showing that we didn't have a lot of errors within our predictions.

### 4.2.5 Results of Feature Selection

In the code, we are seeking to see which of our independent variables are our best features based off the winning percentage, which is our continuous target variable, our goal concerning the target variable is to estimate the mutual information. Mutual information between two random variable is a non-negative value, that measures the dependency between variables. It is equal to zero, if two or more random variables are independent and but if there is a higher value than there is a higher dependency.

Since the values of our independent variables are float and this is a regression problem, we need to remove any categorical columns such as the Teams. We are to find the mutual information between h our features (independent variables) along with the winning percentages while winning percentage is the output feature. In order to get the mutual information, we perform a train and test split between the independent variables and dependent variables, afterwards we do a mutual information regression on the x-train and the y-train, which gives us an array of values which contains value more than zero but less than one, each of these values are corresponding to an independent feature which is telling us how important it is with respect to the dependent feature.



```
W          0.761334
POS/NEG    0.550320
GP         0.505456
BLKA       0.297891
PTS        0.231969
FGP        0.206496
FTM        0.179807
DREB       0.161951
AST        0.080932
L          0.076308
FTA        0.059471
FGM        0.051287
MIN        0.045137
REB        0.036461
3PP        0.026613
FTP        0.020687
TOV        0.017112
3PA        0.000000
3PM        0.000000
STL        0.000000
BLK        0.000000
FGA        0.000000
PF         0.000000
PFD        0.000000
OREB       0.000000
```

#### 4.2.5.1 Information Gain

The following figure shows the mutual information concerning the independent features and their numerical effect on the dependent feature. According to what, we can see on our output, we can see that the feature, 'W' is has a high effect on the winning percentage. This is making a lot of sense because making it to the NBA championship is dependent on the amount of wins a team has made, the more wins a team has the more chances he increases of making it to the playoffs. Wins is the most important factor when making it to the championships and winning it as well, that is the key variable in determining whether you will win the championship. Golden State Warriors in the seasons of 2021 to 2022 had the highest winning

percentage, they had a winning percentage of 0.727, out of all the other 15 teams, and they had the most wins, the number of games they won were 16 with only 6 loses.
The second most impactful feature is the POS/NEG, which is also known as the Roland Rating. The POS/NEG measures the plus-minus statistics for a given player when the player is present in the game, compared to when the player is not present in the game.
For example, if the Brooklyn Nets were to score 100 points (per 100 possessions) while a given player is on the court and only score 80 points while he is off the court, his net offensive plus-minus is +20 (100 – 80 = 20).
The third most impactful feature was the GP, which stands for Games Played, when you play more games, the chances of you increasing your wins heightens, and so does the other variables such as the PTS, and POS/NEG. The Detroit Pistons according to our Average statistic datasheet, only played an average of 4 games, and had a low win percentage of zero, along with the team known as the Indiana Pacers who also had a low win percentage of zero. Logically, if you played the most games, it means you were persistent in making breakthrough such as going through the seasonal games, then the playoffs, and eventually, you made it to the Championship, Golden State Warriors and the Phoenix Suns were the two-basketball team that made it to the last seasons championship games.

The fourth feature that was impactful was the ‘BLKA’ feature which stands for ‘Blocks Against’, this is when the attempted field goal gets blocked by a defender. Defending is a very important part of basketball, depending on how good a team’s defense is, you can decrease the chances of another team making a point, and because of this you are also decreasing the chances of the other team winning, yet increasing your chances of winning provided that the other team’s defense is bad. As mentioned in the introduction earlier, a field goal is any shot other than a free throw, which is worth two or three points depending on the distant between the basketball hoop and where the shot was attempted.

The fifth feature that had an impact on our machine learning model is the PTS, which stands for points, to win a basketball match, you need to have the most amount points between your opposing team. The two teams that had the most amount of points on average of the three seasons prior to the 2021-2022 season was the Golden State Warriors and the Memphis Grizzlies, understanding why points is an important is feature is quite simple, you need points in order to win your game, if you aren’t making any points, and the other team is, the chances of them making it to playoffs increases. Basketball games is a battle of making the most points to advance.

When we used information gain as our feature selection type, we ended up getting field goal percentage, which is abbreviated as ‘FGP’ as our sixth impactful feature selection, field goal percentage doesn’t necessarily mean that a point has been made, but it is the number of made shots made by the total number of shot attempts. Having a low field goal percentage can tell whether a team is playing poor offensively or if the team is attempting to make to many difficult shots. It doesn’t completely tell the skill of a team.

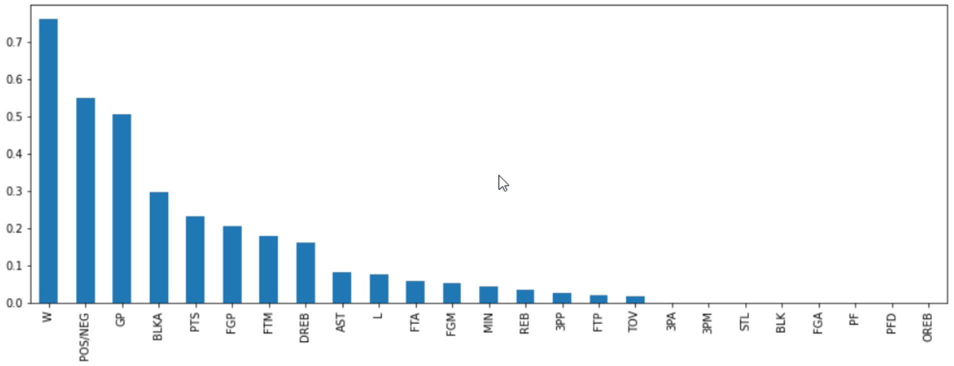
‘FTM’, which stands for Free Throw Makes was the seventh most impactful variable, when using information gain to do feature selection to our model. Free Throws doesn’t really have a high impact on the actual game of basketball because, if a team is really below the team it is losing against, the two free throws, you possibly made will have an impact on the PTS, but not a huge one, so it is very important to have a high amount of points or an equal amount of point, so that the free throws that are made are significant.
Free throws provide the basketball team with a chance to score points not within the game time limit during a match.

Lastly, we have the ‘DREB’ feature which stands for Defensive Rebound, as mentioned previous, having a good defense in some aspect does increase your chances of winning because, if the team is really good offensively, and your defensive stance is really bad, then the chances of the other team winning increases and this will lower your teams winning percentage, in addition to that this is defensive rebound, which mean the team caught the ball while the opposing team was trying. The Chicago Bulls’ during that Michael Jordan era had very powerful defender known as Dennis Rodman, this player was one of the key components that led to their team being a dynasty, he was known to be very great at play defensive and catching rebounds because of him, their team was able to make it to the championship. In addition to that, Dennis Rodman had no offensive skills. Defensive Rebound had only a 16% impact on our dependent variable of winning percentage.

When it comes to basketball, a balance of the defense and the offensive is important but this isn’t always absolute, if you have great defense but is bad offensively, eventually the other team will end up winning because of the lack of points you are making, if you have good offensive skills, yet bad defense, and the team has a both great offensive, and great defensive, the other team will have an edge. However, offense is the best defense.

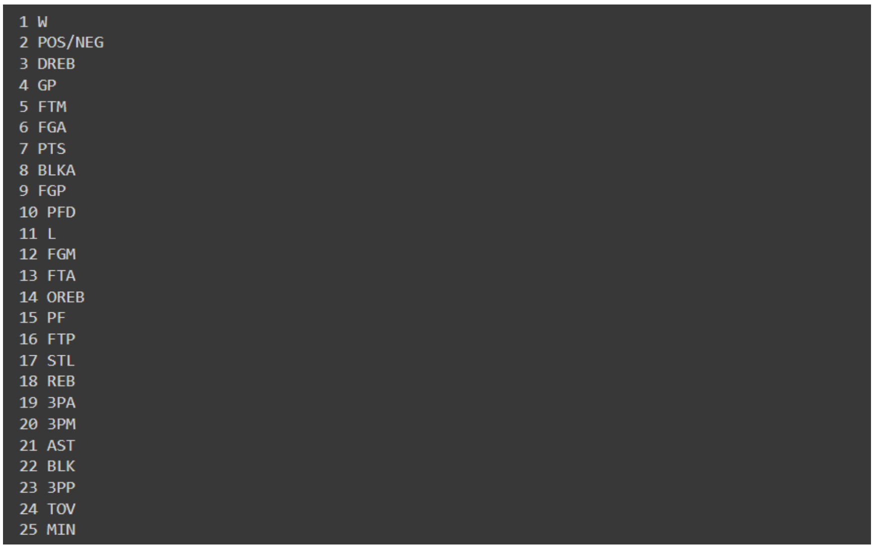
The rest of the feature influenced our dependent variable to a percentage that is lower than 10%, such as assist, lose, free throw attempts, field goal makes, minutes played, rebounds, three-point percentage, free throw percentage, and the turnovers. One of these features, ‘L’, which stands for Losses should have had a zero-percentage effect because when reaching the playoffs and championship it is important to have the least number of losses, if possible, then to not have any losses at all. This shows our feature selection is not in perfection in showing which feature causes an effect to our dependent feature. Losses decreases your winning percentage, not increase them.

The following figure shows our feature’s significance number in the form of a graph:



4.2.5.2 Recursive Feature Elimination

When we did Recursive Feature Elimination, we got the following as our output:



```
1 W
2 POS/NEG
3 DREB
4 GP
5 FTM
6 FGA
7 PTS
8 BLKA
9 FGP
10 PFD
11 L
12 FGM
13 FTA
14 OREB
15 PF
16 FTP
17 STL
18 REB
19 3PA
20 3PM
21 AST
22 BLK
23 3PP
24 TOV
25 MIN
```

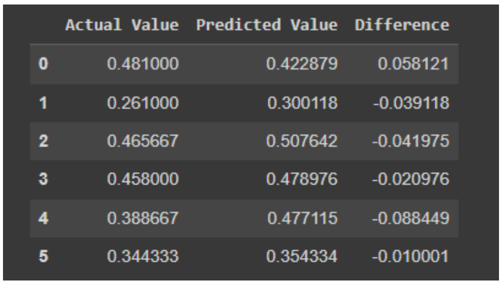
This isn’t too different from the Information Gain Feature Selection technique, but here in the of the Games Played being third place, we instead got the Defensive Rebound as the third place, and many other chances.

Between the two, I would say feature selection are correct in their selection, but in basketball everything you do matters. One of the drawbacks on the Feature Selection is the lack of not seeing the importance of three pointers, as making a lot of three pointers can become a game changer. One of the reasons why Golden State Warriors is successful in making it to the championship is because of a consistent three-point shooter known as Stephen Curry. A game could be 198-200, but the team the team that is 198 could have a three-point shooter, and saved the game, and they advance to the next game. So, the lack of seeing the significance of three-point shots is one of the drawbacks on the two feature selections, we have chosen.

## 4.3 The Result of the Ridge Regression

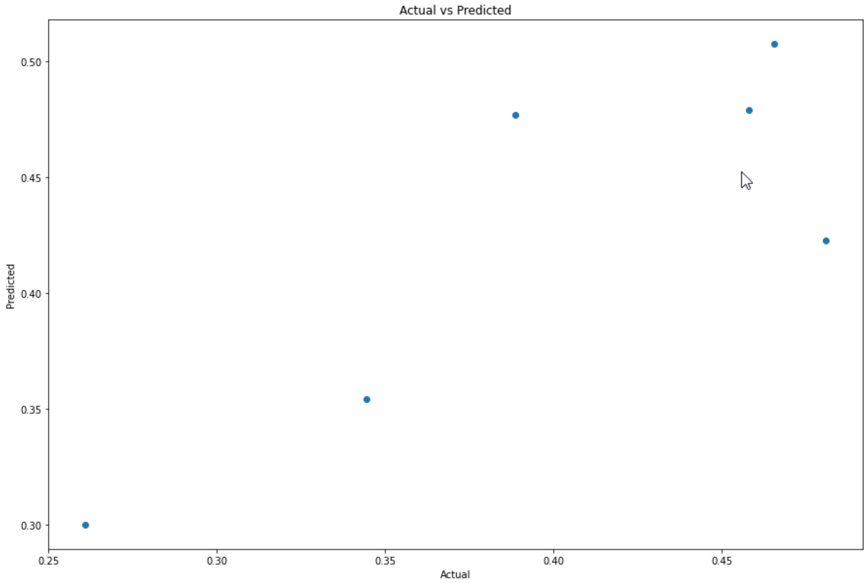
Ridge Regression is a machine learning model that is used to solve the problem of overfitting, which means that almost 100% of our data points are being touched by the regression line. If our model is overfitting, the ridge regression machine learning model will add a small amount of bias, so that our model will be able to the model according to the true values of the data. If our data suffers from multicollinearity, our ridge regression will handle that issue. The NBA Data doesn’t suffer from multicollinearity as our independent variables is linear when it comes to affecting the dependent variable, since everything thing that you do in the NBA counts, that is the rebound, the 3-pointers, the blocks, and steals.

### 4.3.1 The Predicted Rankings of Ridge Regression



| | Actual Value | Predicted Value | Difference |
|---|---|---|---|
| 0 | 0.481000 | 0.422879 | 0.058121 |
| 1 | 0.261000 | 0.300118 | -0.039118 |
| 2 | 0.465667 | 0.507642 | -0.041975 |
| 3 | 0.458000 | 0.478976 | -0.020976 |
| 4 | 0.388667 | 0.477115 | -0.088449 |
| 5 | 0.344333 | 0.354334 | -0.010001 |

### 4.3.2 The Line of Best Fit for Ridge Regression



## 4.4 Future Work

Some of the limitations within our results is the inaccuracy of predicting the 2021-2022 NBA Data frame to an agreeable degree. This is because of the lack of variables caused this, when it comes to sports, there are many variables that need to be considered to get a very great model, and the ones that we used wasn't nearly enough to predict a very accurate model that could predict the 2021-2022 NBA Championship winner. To fix these limitations, other variables such as team chemistry, hours of team practice, emotional state of the average player within the team, and other variables need to be considered as well. Another drawback is that in statistical modelling, we are only studying the pattern of how the line within the graph is flowing and making assumptions of how it may continue further. One way to overcome this is by using mathematical modelling to make better predictions.

To further develop our work, we will need to add more variables to our machine learning models, and an addition to that, we will have to also use mathematical modelling for a prediction.

## References

[1] L. (“NBAMV”), N. T. L. (“NBATV”) and N. P. Inc. (“NBAP”) (NBAMV, N. and N. collectively referred to as the “NBA E. and T. D. B. S. Inc. NBA Media Ventures, “NBA Dataframe from the NBA Website,” *NBA Media Ventures, LLC (“NBAMV”), NBA TV, LLC (“NBATV”) and NBA Properties, Inc. (“NBAP”) (NBAMV, NBATV and NBAP collectively referred to as the “NBA Entities”) and Turner Digital Basketball Services, Inc.* https://www.nba.com/stats/teams/traditional?sort=TEAM_NAME&dir=1&Season=2018-19&SeasonType=Playoffs (accessed Sep. 23, 2022).

[2] Bex T, “Feature Selection code.” https://towardsdatascience.com/powerful-feature-selection-with-recursive-feature-elimination-rfe-of-sklearn-23efb2cdb54e (accessed Sep. 23, 2022).

[3] Farukh Hashmi, “Using Artificial Neural Networks for Regression in Python.” https://thinkingneuron.com/using-artificial-neural-networks-for-regression-in-python/ (accessed Sep. 23, 2022).

[4] Chaya Bakshi, “Random Forest Regression code,” Jun. 08, 2020.

[5] Neil Weinberg, “Complete List (Pitching),” *FANGRAPHS*, Dec. 18, 2014. https://library.fangraphs.com/pitching/complete-list-pitching/ (accessed Sep. 23, 2022).

[6] Ondřej Hubáček, Gustav Šourek, and Filip Železný, “Exploiting sports-betting market using machine learning,” *Czech Technical University*, Feb. 19, 2019. https://www.researchgate.net/publication/331218530_Exploiting_sports-betting_market_using_machine_learning (accessed Sep. 23, 2022).

[7] Sergio Ledesma, Mario-Alberto Ibarra-Manzano, Dora-Luz Almanza-Ojeda, Pascal Fallavollita, and Jason Steffener, “Artificial Intelligence to Analyze the Cortical Thickness Through Age,” *Frontiers Media S.A.*, Oct. 2021. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC8548778/ (accessed Sep. 23, 2022).

[8] Peter Rubbens, Ruben Props, Frederiek-Maarten Kerckhof, Nico Boon, and Willem Waegeman, “PhenoGMM: Gaussian mixture modelling of microbial cytometry data enables efficient predictions of biodiversity,” Jun. 19, 2020. https://www.biorxiv.org/content/10.1101/641464v2 (accessed Sep. 23, 2022).

[9] ΣΕΗΒΑΚΖ⍰ ΓΖΜΖΣΡΗΟ⍰, “Price prediction and forecasting for items in the online game Path of Exile using Machine and Deep Learning methods,” *ΔΘΝΙΚΟ ΜΔΣ⍰ΟΒΙΟ ΠΟΛΤΣΔΥΝΔΙΟ*, 2019. http://artemis.cslab.ece.ntua.gr:8080/jspui/bitstream/123456789/17245/1/Diploma%20Thesis%20Tzivakis%20Dimitrios.pdf (accessed Sep. 23, 2022).

[10] Abdel Karim KASSEM, “Système intelligent basé sur des techniques de l’apprentissage automatique pour l’évaluation de la sécurité et la détection des cyber-intrusions,” Jul. 2021, Accessed: Sep. 23, 2022. [Online]. Available: https://tel.archives-ouvertes.fr/tel-03522384/document

[11] Verzo, “What is machine learning, History of Machine Learning.” https://www.verzeo.in/blog-what-is-machine-learning (accessed Sep. 23, 2022).

[12] William Mounter, Chris Ogwumike, Huda Dawood, and Nashwan Dawood, “Machine Learning and Data Segmentation for Building Energy Use Prediction—A Comparative Study,” *School of Computing, Engineering and Digital Technology (SCEDT), Teesside University*, Sep. 18, 2021. https://research.tees.ac.uk/ws/files/34005621/Machine_Learning_and_Data_Segmentation_for_Building_Energy_Use_Prediction_A_Comparative_Study.pdf (accessed Sep. 23, 2022).

[13] Kaarthik Elangovan, “How To Create A Fraud Detection System In A Financial Market,” Sep. 2013, Accessed: Sep. 23, 2022. [Online]. Available: http://docplayer.net/3895061-Real-time-fraud-detection-in-financial-markets.html

[14] Kristof Kipp, Aaron J Cunanan, and John Warmenhoven, “Relative importance of lower extremity net joint moments in relation to bar velocity and acceleration in weightlifting,” Jun. 15, 2020. https://www.tandfonline.com/doi/full/10.1080/14763141.2020.1718196 (accessed Sep. 23, 2022).

[15] Rosa María García-Gimeno, César Hervás-Martínez, and Maria Isabelde Silónizc, “Improving artificial neural networks with a pruning methodology and genetic algorithms for their application in microbial growth prediction in food”, Accessed: Sep. 23, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0168160501006080?via%3Dihub

[16] YUGESH VERMA, “Why Data Scaling is important in Machine Learning & How to effectively do it,” *Developers Corner*, Aug. 29, 2021. https://analyticsindiamag.com/why-data-scaling-is-important-in-machine-learning-how-to-effectively-do-it/ (accessed Sep. 23, 2022).

[17] Fuzhi Su and Meihong Chen, “Basketball players’ score prediction using artificial intelligence technology via the Internet of Things,” *J Supercomput (2022)*, Jun. 17, 2022. https://link.springer.com/article/10.1007/s11227-022-04573-6 (accessed Sep. 23, 2022).

[18] Daniel Mapar, “Introduction To Machine Learning With TensorFlow,” Mar. 27, 2021. https://github.com/danielmapar/IntroductionToMachineLearningWithTensorFlow (accessed Sep. 23, 2022).

[19] Prasad Kulkarni, Tushar Patil, Aditya Pandey, Vishwesh Vyawahare, Dhiraj Magare, and Gajanan Birajdar, “Performance Assessment of Hetero-Junction Intrinsic Thin Film HIT Photovoltaic Module Using Machine Learning Methods,” *ITM Web Conf.*, May 05, 2022. https://www.itm-conferences.org/articles/itmconf/abs/2022/04/itmconf_icacc2022_01009/itmconf_icacc2022_01009.html (accessed Sep. 23, 2022).

[20] Nima Beheshti, “Random Forest Regression,” *Towards Data Science*. https://towardsdatascience.com/random-forest-regression-5f605132d19d (accessed Sep. 21, 2022).

[21] Javatpoint, “Regression in Data Mining.” https://www.javatpoint.com/regression-in-data-mining (accessed Sep. 23, 2022).

[22] H2O AI, “Feature Selection,” *https://h2o.ai/wiki/feature-selection/*. https://h2o.ai/wiki/feature-selection/ (accessed Sep. 23, 2022).