Predicting NFL win percentage using fantasy football stats

Michael Brown, Zack Soden

UMBC DATA 606 Capstone, Spring 2022

# Background

For our Capstone project, we wanted to take a deeper look into fantasy football data and use linear regression techniques to see if we could make predictions regarding real-world NFL win percentages. We used fifty seasons’ worth of win-loss results and fantasy data to perform our analysis. Check out Github [here](https://github.com/mbrown-amc/DATA-606-Capstone-Project).

What exactly is fantasy football? Numerous sports media sites such as ESPN, Yahoo Sports, and CBS Sports host fantasy football leagues where a varying number of online players (commonly referred to as owners) create and manage their own teams composed of real life NFL players (Becker & Sun, 2016). As the NFL season progresses and real teams compete with each other, so do owners. A typical fantasy team roster may consist of a quarterback (QB), 2 running backs (RB), 2 wide receivers (WR), a tight end (TE) , a kicker (K), a team defense/special teams unit (DEF or DST), and a flex position (which can be filled with a RB, WR, or TE). Team owners accumulate points based on the actual game performance of said players. Scoring rules can be customized and may differ across sites and leagues, but a common format for offensive scoring is 6 points per touchdown thrown/caught/ran and single point for every 25 yards thrown/caught/ran. In Points Per Reception (PPR) leagues, a point is awarded for every catch. It is also possible to score negative points for fumbles and interceptions. Kickers and defenses can earn points as well, but our dataset did not include point totals for them.

There are several questions we hoped to answer as a result of our analysis:

* Do teams with players who accumulate a lot of fantasy points achieve more wins than teams that do not?
* Is there any correlation between fantasy points scored and team wins?
* Does a particular player position have a stronger correlation to wins?

We made several assumptions regarding the data:

* Teams with talented skill positions players (Quarterbacks, wide receivers, tight ends, and running backs) will have higher fantasy point totals because they typically score more touchdowns and accumulate more yards. In general, scoring more touchdowns should lead to more wins.
* The amount of fantasy football stats (and stats in general) a player accumulates will come down to the player’s usage. When we say usage, we are referring to how many rushes, catches, pass attempts, etc., a player can make. The flow of the game can impact a player’s usage (*Predicting Game Flow in the NFL*, 2020). Draft Kings took a sample of games from the 2012 and 2013 seasons and noted that quarterbacks on teams that were losing by 7 or less points in the 4th quarter of games had double the amount of pass attempts as quarterbacks on teams who were winning by 7 or less points. This is important to note because more pass attempts would generally lead to a player accumulating more yards and touchdowns (and possibly more interceptions).

While we did not find any existing research or projects where fantasy points were used to predict wins or team success, some existing projects we referenced that examine NFL Fantasy data include:

* A Kaggle user analyzed 2019 player fantasy points; made predictions on 2020 output. The goal was to determine which players would be underrated or overrated based on projected points and draft the players accordingly. It appears rather limited as only one year of fantasy data was used (Murphy, 2020).
* In a capstone project, a Louisiana Tech student used an ARIMA-based model to predict how many fantasy points a player would score in the future. Their model analyzed a player’s 2019 stats to produce 2020 performance. The results were pretty accurate according to a MAPE (Mean Absolute Percentage Error) of 4.65%. However, it seems this was treated more as a math problem than a data science project (Robinson, 2020).

# Getting our data

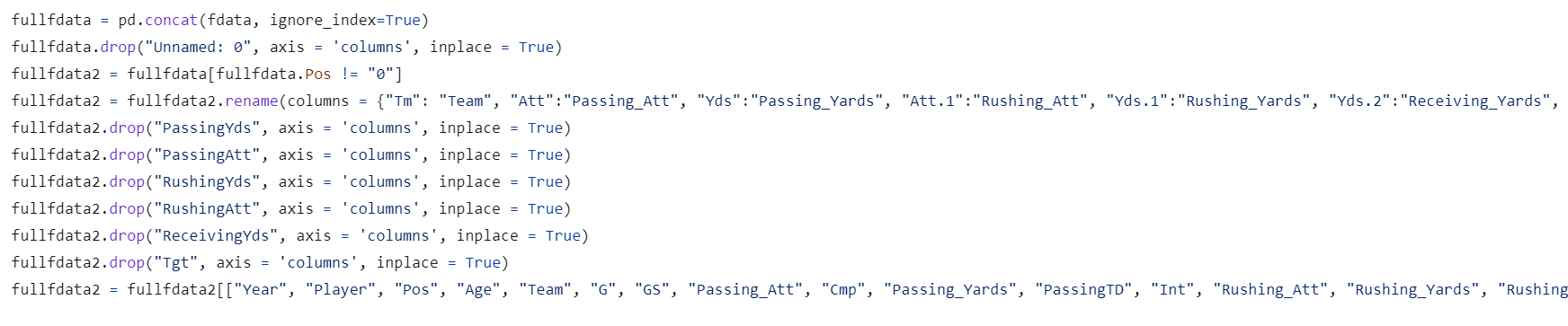
We obtained fantasy football data from [Fantasy Football Data Pros](https://www.fantasyfootballdatapros.com/csv_files). The site has NFL data from each NFL season from 1974 through the end of 2019. There was one .csv file for each season, which we merged into a single dataframe to start.

For NFL game standings and results, we used data from [Pro Football Reference](https://www.pro-football-reference.com/), where there was data available for NFL seasons 1970 through the end of 2019. There was an .xlsx file for each NFL conference (AFC and NFC), so we ended up with 100 files that were also merged into a single dataframe.

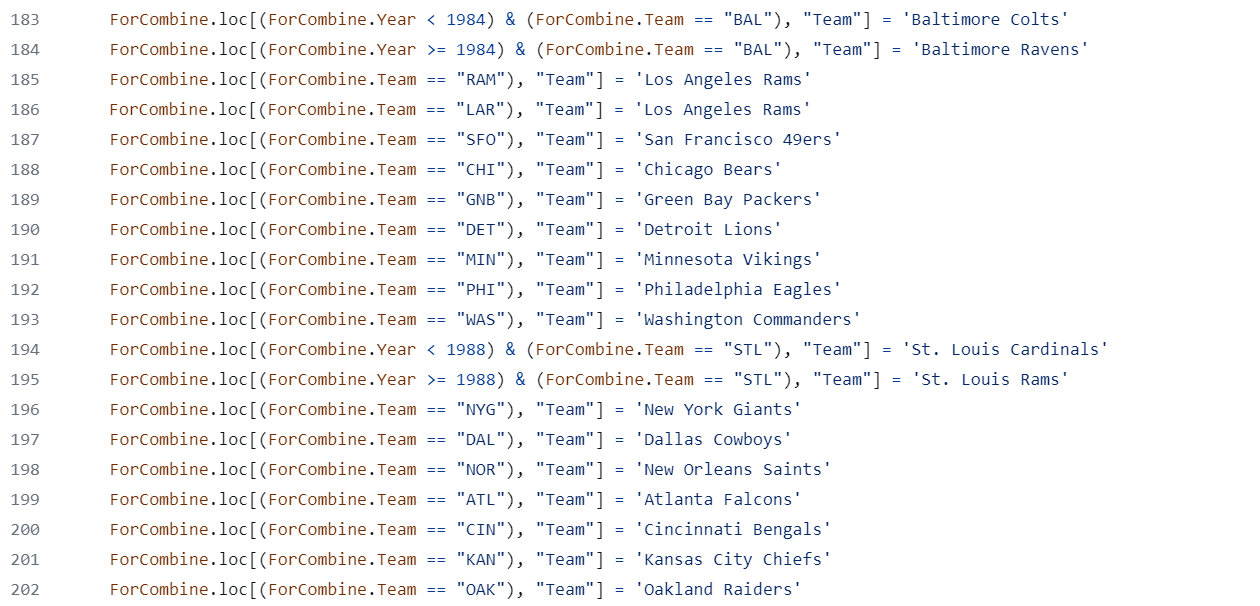
# Cleaning the data

With all the necessary files saved into our Github repository, we started by reading in and appending each CSV file before renaming some of the columns for clarity.

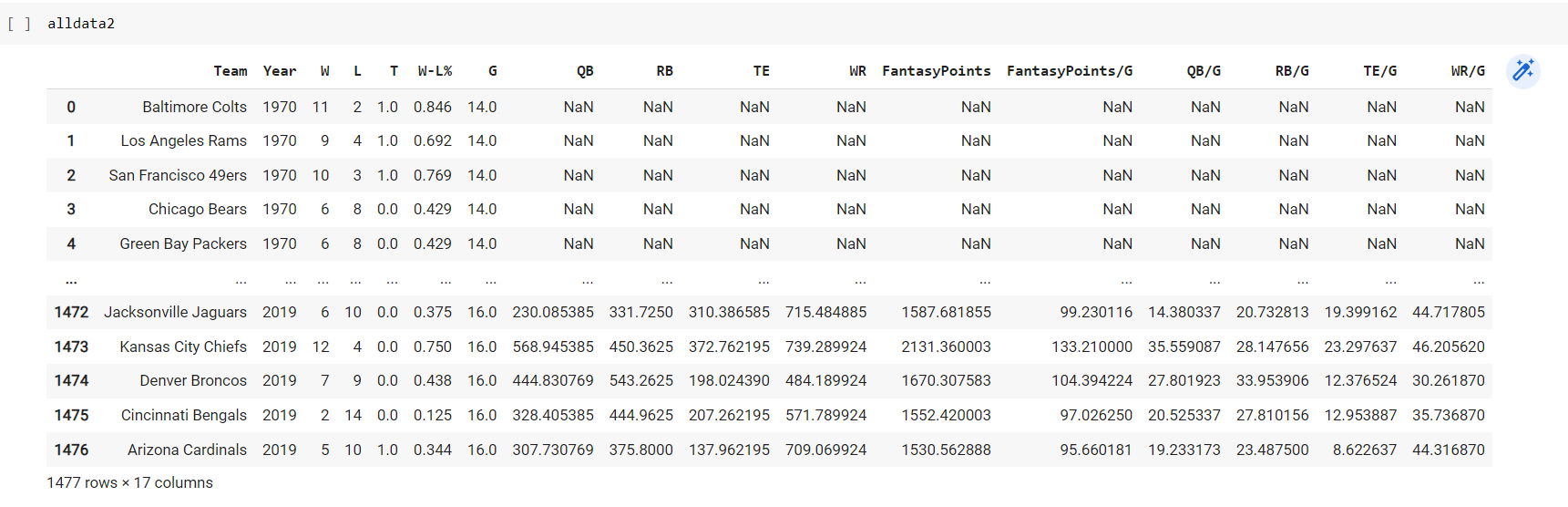
In the utils.py file, we built a function, get\_data, that essentially does all the heavy lifting. We pass an argument that includes the dataframe and also allows for selecting *x* amount of players from each position on a team. The function also allows us to choose how to fill in data for rookies: using the minimum positional value, using the max value, using an average, or removing them from the data set. In the graphic below, we show that the data frame had columns such as “Yds” and “Yds.1”**,** that were renamed to “Passing\_Yards” and “Rushing\_Yards”.

****

Similarly, we read in and appended all the .xlsx files to create a dataframe for the team standings data. We dropped several columns we would not need for analysis,such as strength of schedule, points for, and points against (“Sos”, “PF”, “PA”). For simplicity, we removed any player who played for multiple teams in a single season. To organize the teams, we needed to account for teams that had a city or name change. Baltimore’s team was the Colts up until they relocated to Indianapolis and became the Indianapolis Colts in the 1984 season. ‘BAL’’ here represents both the Baltimore Colts before 1984 and the Baltimore Ravens after 1984. Teams such as the Houston Texans/Oilers and the Tennessee Oilers/Titans also had to be cleaned up.

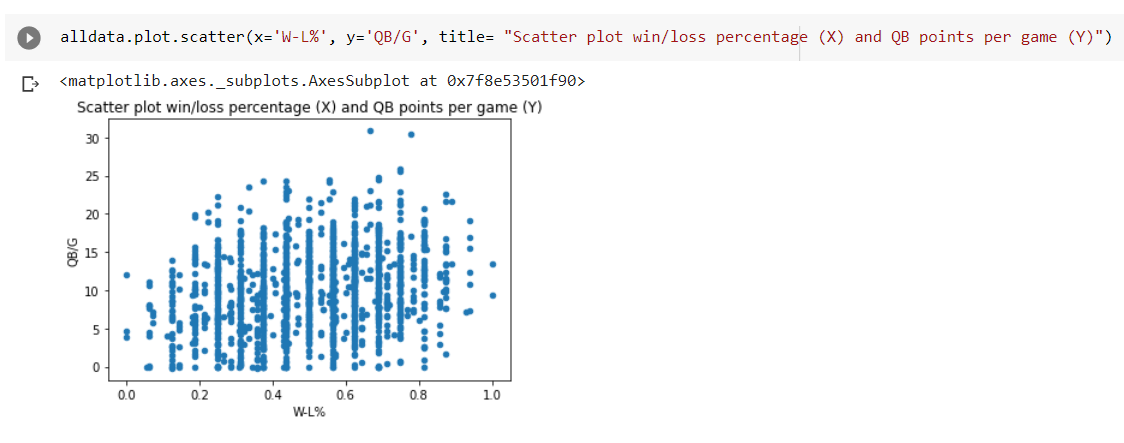


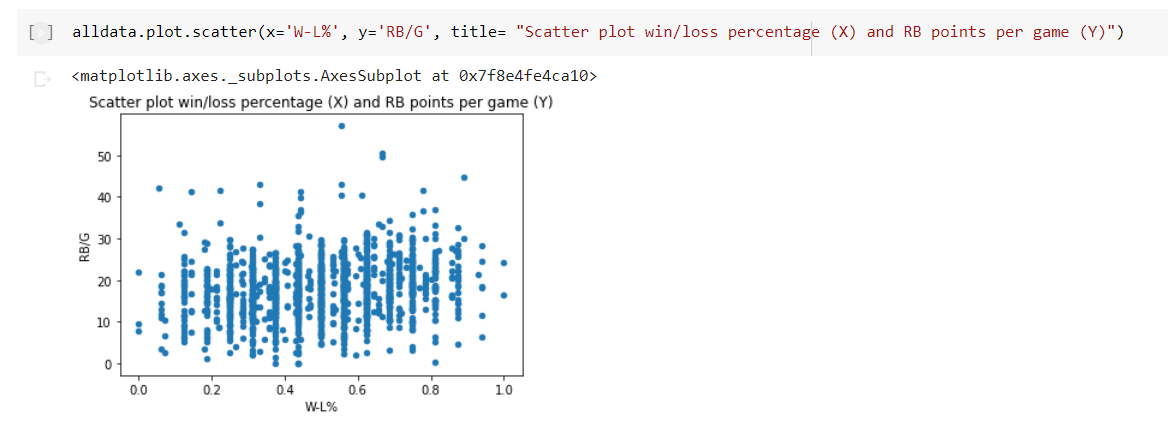
Our resulting dataframe **alldata,** shown below, has combined season by season win-loss results with the team’s total fantasy points for the year; broken down by position. Comprised of 1,477 rows and 17 columns, we included columns that average out fantasy points per game and fantasy points by position since not every NFL season had the same number of games (in the 1970’s, NFL teams played a 14 game schedule, but a 16 game schedule was implemented starting with the 1990 season).



# Exploratory analysis

Running alldata.corr() showed us some good news; that there was a correlation between win percentage and fantasy points. There was also a correlation between QB fantasy points and win percentage. We used scatter plots to visualize the data, as shown below:

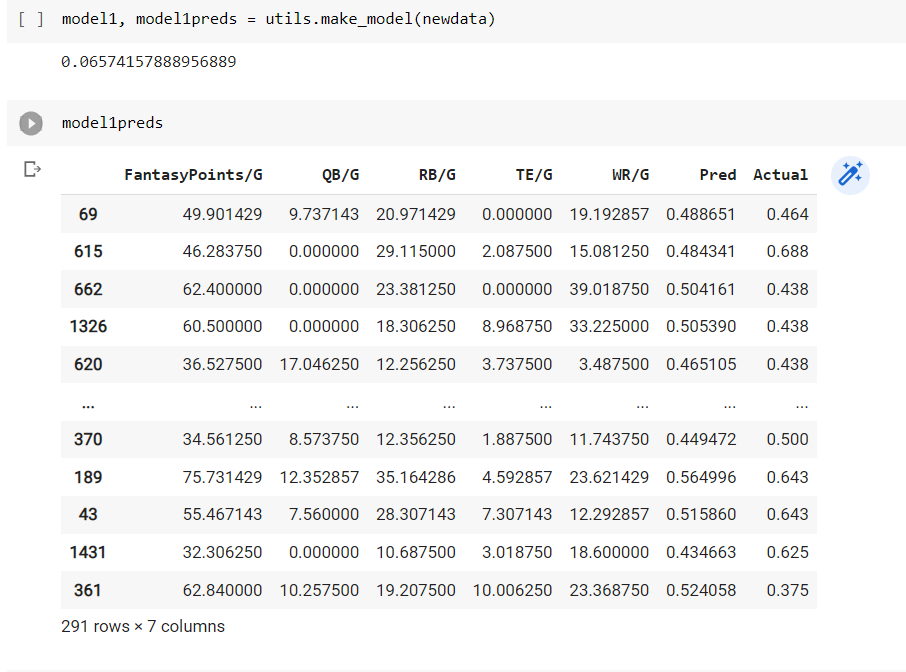




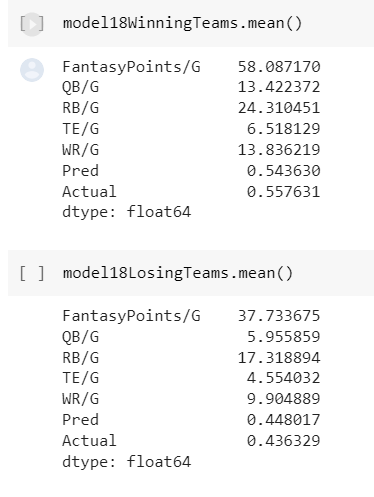
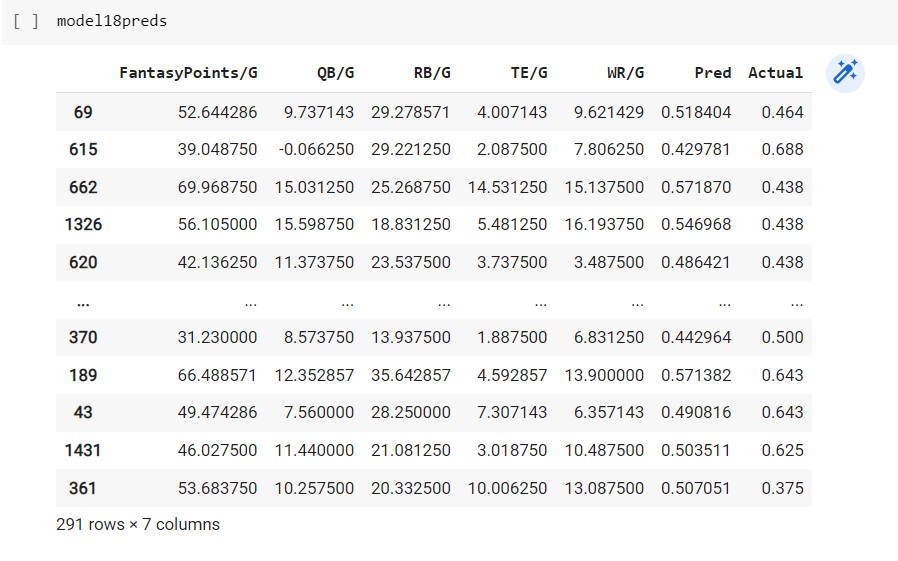
# Building the model/model performance

In our utils.py file, we have a function called make\_model where we introduce

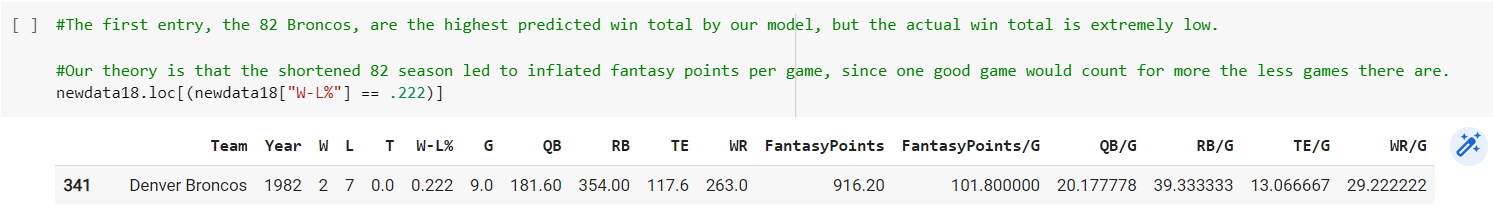
The Scikit-learn library to build our linear regression model. We split the incoming dataframe into train and test sets, using the fantasy score related columns as the X data and the win percentage as the Y data. We used the StandardScaler function to standardize our values for total fantasy points per game, and the positional points per game. The function then returns the model and the dataframe with the actual and predicted win percentage values.

1. Our first model (**model1**) used the full dataset and had an accuracy score of 6.5% It uses the full dataset but drops all rookies. 

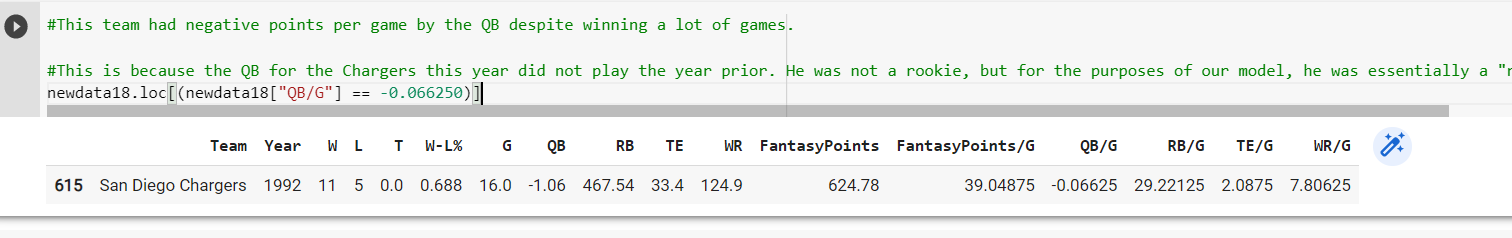
1. From here, we ran more than 20 different models with various combinations of *x* amount of players and minimum/maximum/average/no rookie stats.
2. Our most accurate model (**model18**) had a score of 10.5% which used 1 QB, 1 WR, 1 TE, and 4 RBs, and the minimum values for rookies. We dug into this model a little more and noticed a few things. We divided the data into two sets; teams predicted to win more than 50% of their games and teams that were predicted to win less than 50% of their games. When averaged out, our actual vs.predicted scores were pretty close: Winning teams had a predicted win percentage of 54.3% against an actual 55.7%. Losing teams had a predicted 44.8% against an actual 43.6%.



We also looked at a few outliers in that data and noticed the 1982 Broncos were predicted to win a lot more games than they actually did. We believe this is due to playing only nine games that season (there was a player strike), which increased the per game average for fantasy points.



The 1992 Chargers also had a QB who our model treated as a rookie with the minimum fantasy values. He was assigned negative stats, but the team still ended up winning a lot of games.

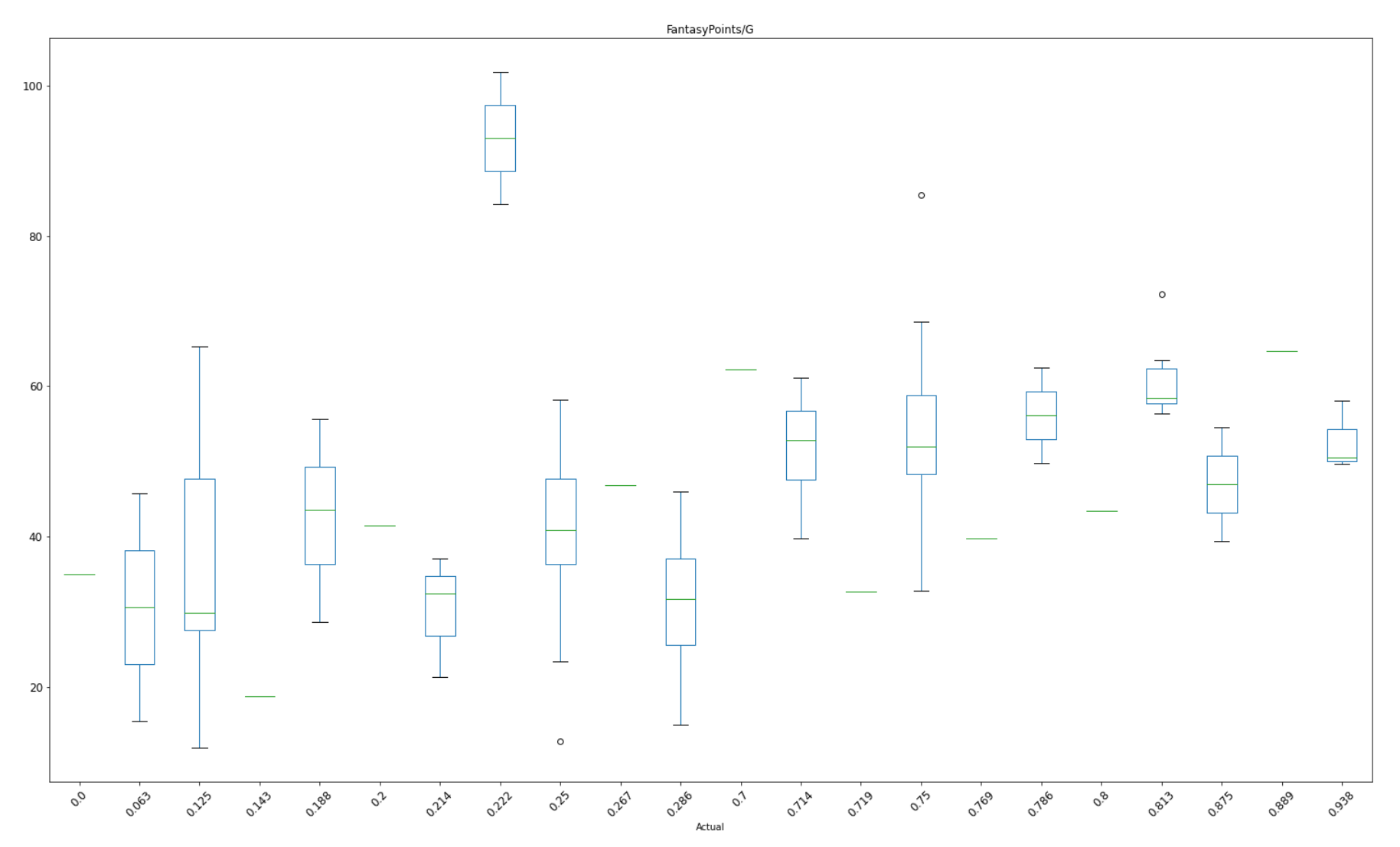


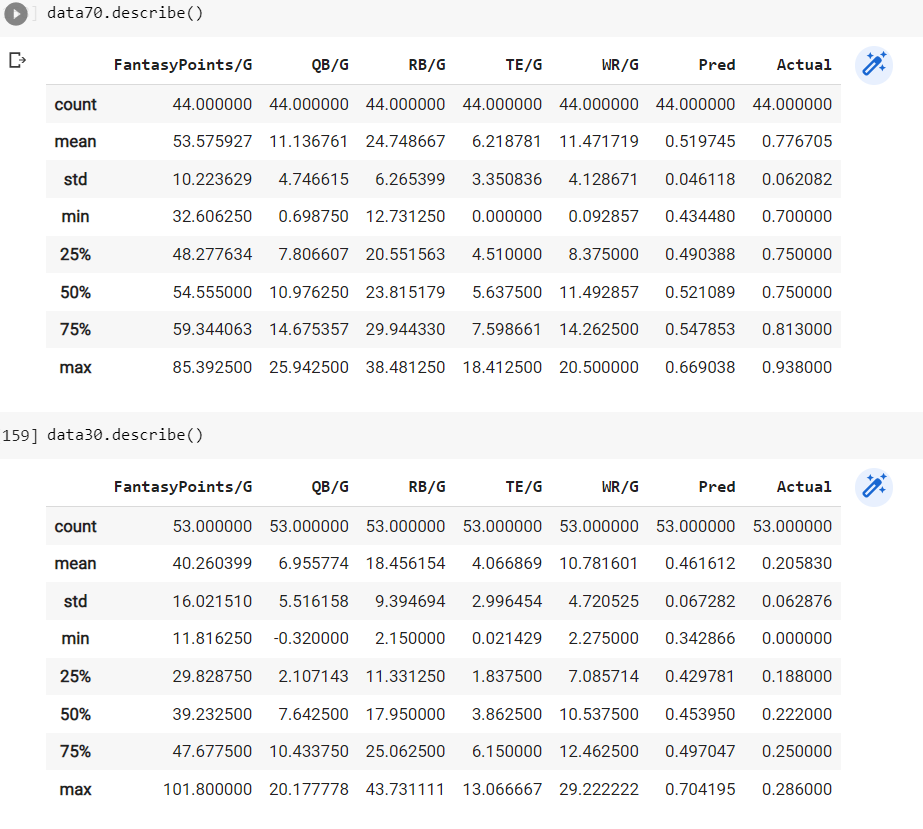
# Conclusion

While we were able to get accurate, quality data for both sets of data, we wonder how our models would have been impacted if our fantasy data included fantasy points for defense and kickers. Our models only take into account offensive player performance so perhaps having kicker points and especially defensive points would have resulted in our model being trained differently. Defensive teams have an innate ability to defend against the scoring of offensive players ((Becker & Sun, 2016), so it is reasonable to assume a high scoring defense can contribute to a team win. On the other hand, fantasy football players do not value top defenses and top kickers as high as the top offensive players; defensive units and kickers are often drafted in later rounds or sometimes not drafted at all. Points wise, they do not contribute much to a fantasy team’s overall score.

Let’s revisit our questions:

1. Do teams with players who accumulate a lot of fantasy points achieve more wins than teams that do not? **Yes**. Using the dataframe from our most accurate model, we divided our data into two sets; one set with teams that had won 70% or more of their games, and the other with teams that lost 70% or more of their games. The teams that won 70% or more of their games scored about 53 fantasy points per game, while those that lost 70% only averaged about 40 points; a 32.5% difference. We created a box plot visualization to show the distribution of fantasy points between the successful and unsuccessful teams. The successful teams have a higher floor and ceiling when compared to the less successful teams.





1. Is there any correlation between fantasy points scored and team wins? **Yes**, but we found the correlation to be pretty weak.
2. Does a particular player position have a stronger correlation to wins? **Yes,** running back fantasy points seem to have the greatest impact on being able to predict win percentage. This is backed up by our most accurate model using 4 running backs and just one player at each of the other positions. External research shows this may be because running backs have the least amount of variation game to game and can have consistent fantasy output. This is in contrast to WRs and TEs, who depend on solid QB play; their fantasy output is at the mercy of having a talented QB who throws accurate passes (Dunnington, 2015). A good NFL defense can scheme ways to take away the best receivers, causing the QB to throw the ball elsewhere.

References

Becker, A., & Sun, X. A. (2016). An analytical approach for fantasy football draft and lineup management. *Journal of Quantitative Analysis in Sports*, *12*(1). https://doi.org/10.1515/jqas-2013-0009

Murphy, C. (2020). *ESPN 2019 Stats and2020 NFL Fantasy Projections*. Kaggle.com; Kaggle. https://www.kaggle.com/mur418/espn-2019-stats-and-2020-nfl-fantasy-projections

*Predicting Game Flow in the NFL*. (2020, June 26). DraftKings Nation; Vox Media, LLC. https://dknation.draftkings.com/nfl/2020/6/26/21309173/predicting-game-flow-in-the-nfl

Robinson, C. (2020). The Prediction of Fantasy Football. *Louisiana Tech - Mathematics Senior Capstone Papers*, *12*(1). https://digitalcommons.latech.edu/mathematics-senior-capstone-papers/20