

Fantasy Fumbball: Using Machine Learning to Predict NFL Player Rank

Wendy Yang

February 14, 2018

Data Science Part Time Course

Why do we care?

The Fantasy Sports industry is estimated at over \$7B per year

The ability to predict player rank accurately can lead to better drafting and ultimately winning the fantasy league

Personally took 11th place in my fantasy league this year 😞

The screenshot shows a mobile app interface for Fantasy Football. At the top, the status bar displays the time as 2:25. The app header reads "Season Story | Fantasy Football | Ya..." with a close button. Below the header, there's a circular profile picture of a cityscape and a large yellow circle containing a "C-" grade, labeled "Season Grade". To the left of the grade, the team name "The golden cloaks" is shown with a dropdown arrow. Below the team name, the following statistics are listed: "11TH PLACE", "6-8-0", "TOTAL POINTS 988.68", and "POINTS/WEEK 70.62". A horizontal blue line separates this section from the "Season Story" section below. Under "Season Story", there are four links: "Season Start: Draft Strategies", "In-Season Management", "Projected and Final Standings", and "Season Grade Summary". The "Season Start: Draft Strategies" link is highlighted with a green circle containing a "B-". Below this link, a paragraph explains that the chart below illustrates how positions were valued in the draft by each team, measuring the weighted distribution of draft picks. It notes that positions drafted earlier and more often are valued more highly, and that the height of the bar in the chart indicates the value of a position. It concludes by asking the user what position they valued most. At the bottom of the screen, a black navigation bar contains three icons: a back arrow, a home circle, and a recent apps square. Above this bar, the text "Draft Value by Position" is visible, indicating the start of a chart.

Season Story | Fantasy Football | Ya... X

The golden cloaks ▾

11TH PLACE 6-8-0 TOTAL POINTS 988.68 POINTS/WEEK 70.62

Season Grade

Season Story

[Season Start: Draft Strategies](#) [In-Season Management](#)

[Projected and Final Standings](#) [Season Grade Summary](#)

B- **Season Start: Draft Strategies**

Your season kicked off with the draft. The chart below illustrates how positions were valued in the draft by each team. It measures the weighted distribution of draft picks. Positions that were drafted earlier and more often are valued more highly. The taller the bar in the chart the more a position is valued. You can mouse over the bar to see what percent of draft resources was used on each position. What position did you value most?

Draft Value by Position

Fantasy Football Background

Fantasy Football teams are comprised of players from all teams in the NFL.

Players in the following positions may be drafted: Quarterback (QB), Running back (RB), Wide Receiver (WR), Tight End (TE), and Kicker (K).

Each player may only be drafted onto one fantasy team per league.

Each player in the league drafts one fantasy player for their team in rounds until the fantasy teams are filled.

Hypothesis

An NFL player's fantasy football rank can be predicted based on their fantasy football stats/points earned from a previous season

Why does this matter? Drafting players with the best rank should allow for the highest scoring team.

Data Collection

<https://github.com/BurntSushi/nflgame>

This pulls JSON files from NFL API

ESPN.com for 2016 top 300 fantasy player rank

Data exploration

This data set contained all players in the NFL, I was only interested in QB, RB, WR, TE, K

```
# Create new DataFrames using only QB, WR, RB, TE, K  
nfl116_QB = nfl116.loc[nfl116.pos == 'QB']  
nfl116_WR = nfl116.loc[nfl116.pos == 'WR']  
nfl116_RB = nfl116.loc[nfl116.pos == 'RB']  
nfl116_TE = nfl116.loc[nfl116.pos == 'TE']  
nfl116_K = nfl116.loc[nfl116.pos == 'K']
```

I also ignored defensive players since we draft entire team defenses rather than individual defensive players

Fantasy Football Scoring

Offensive Players

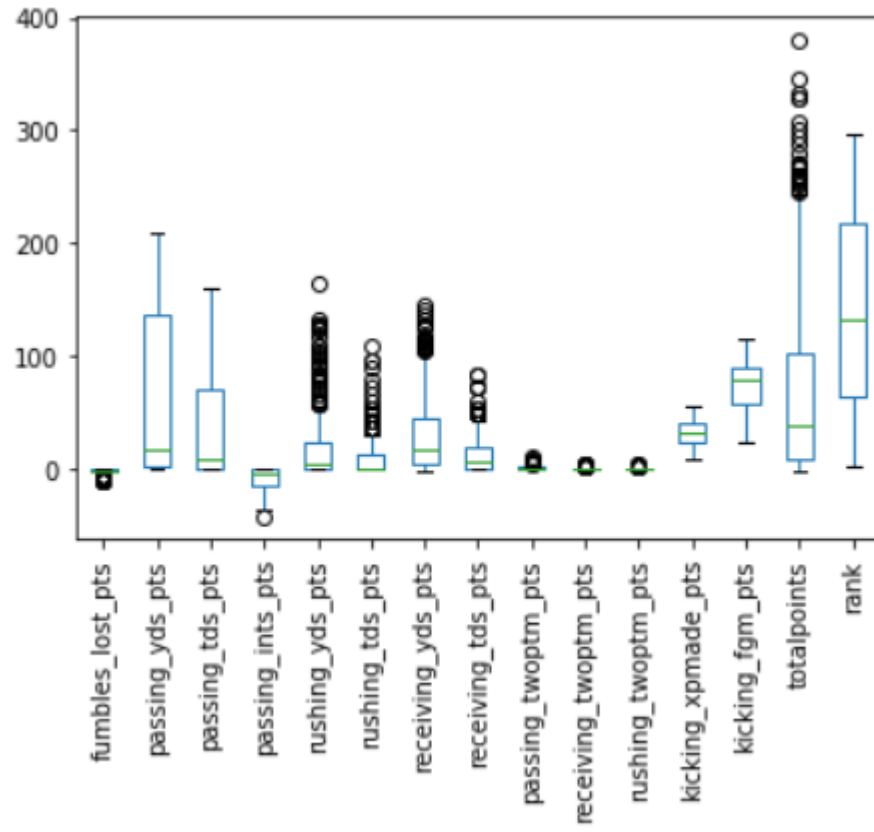
- Passing Yards: 1 point per 25 yards
- Passing Touchdowns: 4 points
- Passing Interceptions: -2 points
- Rushing Yards: 1 point per 10 yards
- Rushing Touchdowns: 6 points
- Receptions: 1 points (only if using PPR scoring)
- Receiving Yards: 1 point per 10 yards
- Receiving Touchdowns: 6 points
- 2-Point Conversions: 2 points
- Fumbles Lost: -2 points
- Fumble Recovered for a Touchdown: 6 points

Kicking

- Point After Touchdown (PAT) Made: 1 point
- Field Goal (FG) Made (0-49 yards): 3 points
- FG Made (50+ yards): 5 points

```
# Map player stats to points using list comprehensions
fumbles_lost_pts = [x * -2 for x in nfl16_off.fumbles_lost]
passing_yds_pts = [x / 25 for x in nfl16_off.passing_yds]
passing_tds_pts = [x * 4 for x in nfl16_off.passing_tds]
passing_ints_pts = [x * -2 for x in nfl16_off.passing_ints]
rushing_yds_pts = [x / 10 for x in nfl16_off.rushing_yds]
rushing_tds_pts = [x * 6 for x in nfl16_off.rushing_tds]
receiving_yds_pts = [x / 10 for x in nfl16_off.receiving_yds]
receiving_tds_pts = [x * 6 for x in nfl16_off.receiving_tds]
passing_twoptm_pts = [x * 2 for x in nfl16_off.passing_twoptm]
receiving_twoptm_pts = [x * 2 for x in nfl16_off.receiving_twoptm]
rushing_twoptm_pts = [x * 2 for x in nfl16_off.rushing_twoptm]
kicking_xpmade_pts = [x * 1 for x in nfl16_off.kicking_xpmade]
# For now, counting all field goals as 3 points for ease of data
# fgys doesn't work across entire season
kicking_fgm_pts = [x * 3 for x in nfl16_off.kicking_fgm]
```

Visualizing Data



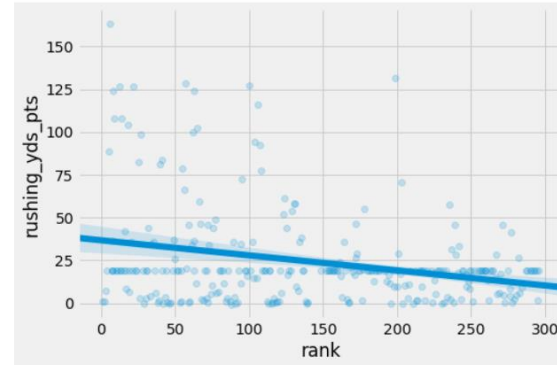
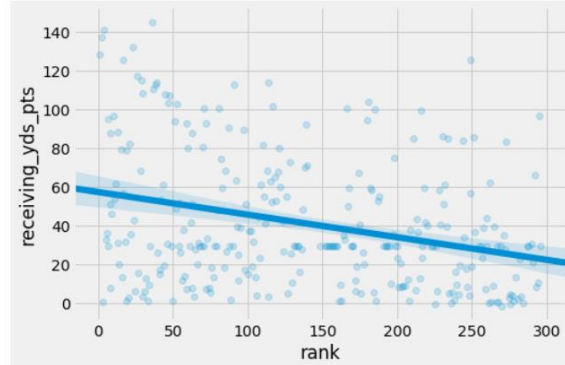
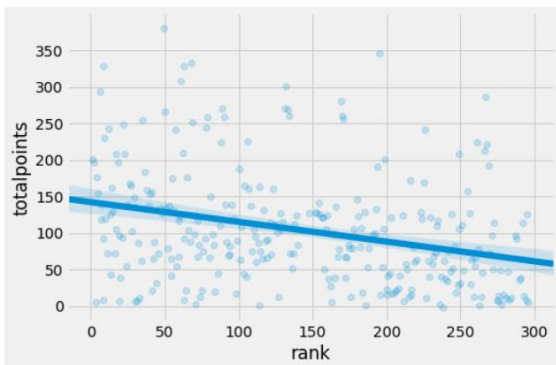
Plotted all of the points as box plots to see if anything was an obvious predictor.

Seems like Total Points has the most outliers along with rushing/receiving yards and touchdowns

Random Forest

Random Forest is a quick preliminary model that can determine important features without having to do feature scaling.

Totalpoints, receiving yards, and rushing yards are the most important features in this set. All appear to be negatively correlated with rank.



	feature	importance
13	totalpoints	0.307617
6	receiving_yds_pts	0.257463
4	rushing_yds_pts	0.169146
7	receiving_tds_pts	0.086722
0	fumbles_lost_pts	0.039199
5	rushing_tds_pts	0.032189
2	passing_tds_pts	0.021820
12	kicking_fgm_pts	0.018654
11	kicking_xpmade_pts	0.018611
9	receiving_twoptm_pts	0.014018
1	passing_yds_pts	0.011532
3	passing_ints_pts	0.009027
10	rushing_twoptm_pts	0.007756
8	passing_twoptm_pts	0.006246

Linear Regression

```
print linreg.intercept_  
print linreg.coef_  
zip(imp_feature_cols, linreg.coef_)
```

```
211.808173066  
[-0.12615973 -0.94943358 -0.81969633]  
  
[('totalpoints', -0.12615973386161361),  
 ('rushing_yds_pts', -0.94943358138773681),  
 ('receiving_yds_pts', -0.81969632808597337)]
```

RMSE = 74.76

Null RMSE = 80.80

My model does better than null!

Conclusions

Random forest was able to determine the most important features

Linear regression was able to predict how player's performance may change player rank by interpreting the coefficients.

A player would have to gain ~ 1 rushing or receiving yard point to improve their rank by 1. Based on the point mapping, that means they would have to gain at least 10 more yards to improve their rank.

A players would have to gain ~ 8 total points in order to improve their rank by 1.

Future Work

See if I can predict improvement in players across years.

Split apart field goals into 3 and 5 points. For now, just scored all field goals made as 3 points since I did not want to loop through all kickers of each game across the entire season to determine which kicks were longer than others.

Aggregate individual players into team defenses and pull a full team score across the season to further improve my rank predictions.

Try to draft position rank besides positions (e.g. WR1, WR2)

Turn this into a product with a user interface to select a player and determine their rank against other players. (Unfortunately, while a good lesson, "who should I pick" already exists as a product)