
Optimizing Preseason Fantasy Football Rankings

John O'Hollaren (jpo4)

Department of Electrical and Computer Engineering

Abstract

Fantasy football is a fast growing, multi-billion dollar industry centered around projecting which players will perform the best over the course of a season. Correctly predicting which players will play well greatly improves your team's year-long performance, whereas wasting a high pick on a low achiever can ruin your season. This paper focuses on an optimal strategy for predicting a list of the top 30 non-rookie NFL wide receivers for a season, using only data available prior to the start of the season.

1 Introduction & Related Work

According to the Fantasy Sports Trade Association, over 10 percent of the US population spent 12 billion dollars playing fantasy football in 2012. Given such a large market, a slight statistical advantage could provide significant profits in fantasy football leagues. Specifically, a correct pre-season ranking of wide receivers (WRs) will allow owners to pick the best players to join their teams during the pre season fantasy football draft, setting them up for year-long success. However, ranking players is difficult. Preseason rankings from teams of experts at ESPN and Yahoo are correct for only a small minority of players. This paper compares several machine learning techniques for ranking WRs, including linear regression, feature reduction, K-Means clustering, mixture models, and PCA. A PCA regression technique which outperforms ESPN and Yahoo experts is presented. Throughout the paper, standard scoring is assumed: Fantasy Points = $6 \cdot \text{TDs} + 0.1 \cdot \text{Yards}$.

While countless fantasy football websites exist, the major players are ESPN, Yahoo, the NFL, 4for4, and FantasyPros. None of these entities releases their methods publicly. Tim Mathews at the University of Southampton [1] explored using Bayesian reinforcement learning for week-to-week team management during the season. Matt Bookman at Stanford used linear regression to predict week-to-week performances during the season with limited success only in certain weeks [5]. The week-to-week prediction problem, however, is simplified through the use in-season data from each player, whereas in the pre-season, year-long ranking problem, predictions must be made for players who are 8 months removed from public games. Some attempts at machine learning approaches for the year-long ranking problem have been made [6], but did not outperform human experts. Other attempts use machine learning to outsmart other players in the same league [3], [8].

1.1 Data Summarization

My dataset includes NFL statistics for over 700 players for every year from 2007 to 2012, however only a subset are used for prediction. There are 13 features in total. 6 are standard stats: yards, touchdowns, targets, catches, ESPN rank, and Yahoo rank. I then add a field for previous year points, and 6 more 'trend' features to represent trends over the years. For example: $\Delta 1yrYds = (\text{Yards in year } Y-1) - (\text{Yards in year } Y-2)$, $\Delta 2yrYds = (\text{Yards in year } Y-1) - (\text{Yards in year } Y-3)$. A total of 6 of these features are used, for both yards and touchdowns for 1, 2, and 3 years into the past, bringing the total to 13.

1.2 Performance Criteria and Baseline

Discounted cumulative gain (DCG) will be used to assess WR rankings. It rewards ranking good players highly and penalizes ranking bad players too high. A higher DCG is better:

$$DCG = rel_1 + \sum_{i=2}^p \frac{rel_i}{\log_2(i)}, \text{ and the relevance weights are: } rel_i = \frac{1}{\text{actual end of year rank}}$$

Using DCG, the dataset can be analyzed by looking at the performance of ESPN, Yahoo, and simple regression across the years. Specifically note ESPN's 2012 DCG is 1.7991 and Yahoo's is 1.8100. These are the benchmarks for performance.

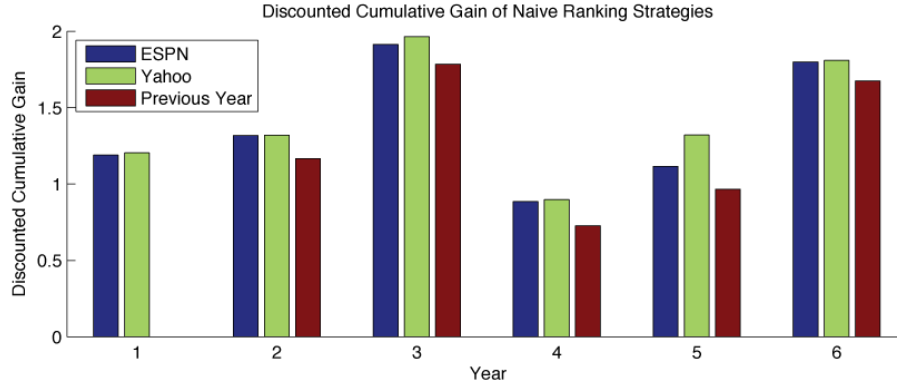


Figure 1: DCG of ESPN/Yahoo preseason rankings by year, along with a naive method using the previous years final rankings as the next years preseason rankings. The 2012 results (1.7991 for ESPN, 1.8100 for Yahoo) will be used as a benchmark.

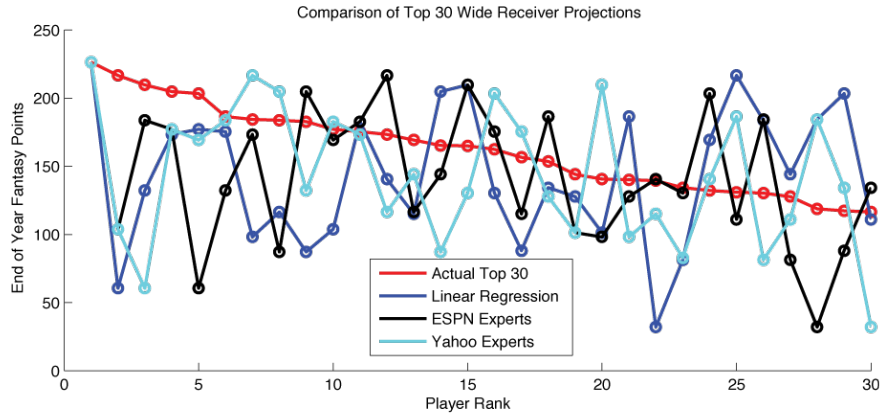


Figure 2: Preseason rankings versus actual points scored. Note how many of ESPN, Yahoo, and linear regression's preseason picks greatly underperform the actual best players in red. This graph shows how hard predicting rankings is.

2 Models and Methods

Linear Regression: β will be calculated using X_{train} as all data up until 2010, and Y_{train} as the 2011 end-of-season performances[4]. X_{test} will then be all data up until 2011, and the resulting $Y_{test} = \beta X_{test}$ will be the 2012 end-of-season performance which will be compared with the actual, ESPN, and Yahoo results using DCG. All features will be used for this 13-dimensional β .

$$\beta = (X^T \cdot X)^{-1} \cdot X^T \cdot Y$$

$$\beta = [w_{Points}, w_{Yds}, w_{TDs}, w_{Targets}, w_{Catches}, w_{\Delta 1yrYds}, w_{\Delta 1yrTDs}, w_{\Delta 2yrYds}, w_{\Delta 2yrTDs}, w_{\Delta 3yrYds}, w_{\Delta 3yrTDs}, w_{ESPN}, w_{Yahoo}]$$

Feature Reduction: First I will employ manual feature reduction (as opposed the PCA I use later)[2]. This helps reduce computational overhead, allows the results to be easier to interpret because they are lower dimensional, and also can help reduce overfitting. Different subsets, L , of features will be experimented with for linear regression using $\beta_L = [w_1 \dots w_L]$.

K-Means Mixture Model: 2 and 3 dimensional K-Means will be used to separate players in different groupings, and linear regression will be run on each of the K mixtures individually. Each mixture, $1 \dots K$, will have it's own learned weights, $\beta_1 \dots \beta_K$. Both the features used for K-means and the number of features used for regression will be varied [7], [9], [10].

PCA Regression: PCA will be used to estimate the regression coefficients. Different numbers of principle components will be experimented with. PCA on X gives a principle component score, $\Omega_{n \times p}$ = a representation of X in the principle component space, and also the principle component loadings in each column of $\Lambda_{p \times p}$. β is initially estimated as before, but by replacing X with Ω and mean centering Y to give: $\beta = (\Omega^T \cdot \Omega)^{-1} \cdot \Omega^T \cdot (Y - \mu_Y)$. Then to make the results easier to interpret, β is transformed to regression coefficients for uncentered variables by adding a mean term and adding a column of 1's to X during prediction: $\beta = [\mu_Y - \mu_X \cdot \Lambda \cdot \beta | \Lambda \cdot \beta]$, $Y = [1's | X] \cdot \beta$.

3 Results

Using linear regression, β is calculated and then used to predict point totals for each player at the end of the next year. The top 30 players are then ranked by their point totals, and the ranking is scored using DCG. Note that as we use more training data, the RMSE of the point predictions reduces, which in turn leads to the DCG increasing as shown in Figure 3.

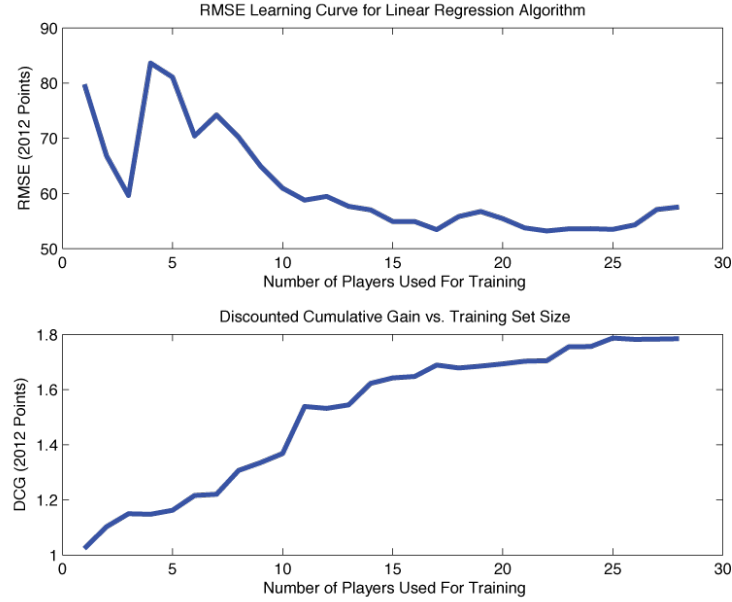


Figure 3: Graph of RMSE and DCG as the training set size increases. Results improve with larger training sets.

The linear regression results can be improved by picking out specific features to perform regression on. Multiple combinations of features were experimented with. The best results were shown with less than the full 13 features, which led to less overfitting on the training set and better results in the test. Specifically, 2, 3, and 7 feature regression worked well as shown in Table 1.

# Features	Features	DCG
13	Points, Yds, TDs, Targets, Catches, $\Delta 1yds$, $\Delta 1yds$, $\Delta 2yds$, $\Delta 2tds$, $\Delta 3yds$, $\Delta 3tds$, ESPN, Yahoo	1.7179
7	Points, Yds, TDs, Targets, Catches, $\Delta 1yds$, $\Delta 1yds$, ESPN, Yahoo	1.7722
6	Points, Yds, TDs, Targets, Catches, $\Delta 1yds$, $\Delta 1yds$, ESPN	1.7621
3	Points, Yds, TDs	1.7806
2	Points, TDs	1.7896
2	Points, Yds	1.7863

Table 1: Results of feature reduction showing improved results in lower dimensional spaces.

The DCG is much improved. Results approaching the accuracy of experts at ESPN and Yahoo are approached, but it is still short. The best DCG obtained was 1.7896, which is 1.2 percent short of the Yahoo rankings.

To further improve the results, a k-means based linear mixture model will be employed. This method takes advantage of the fact that there are many types of receivers in the NFL, including those who play in the slot and catch lots of short passes, those who play out wide and catch very few longer passes, and all-stars who do it all. This insight prompts clustering receivers by type period to prediction, which is done using K-Means as shown in Figure 4.

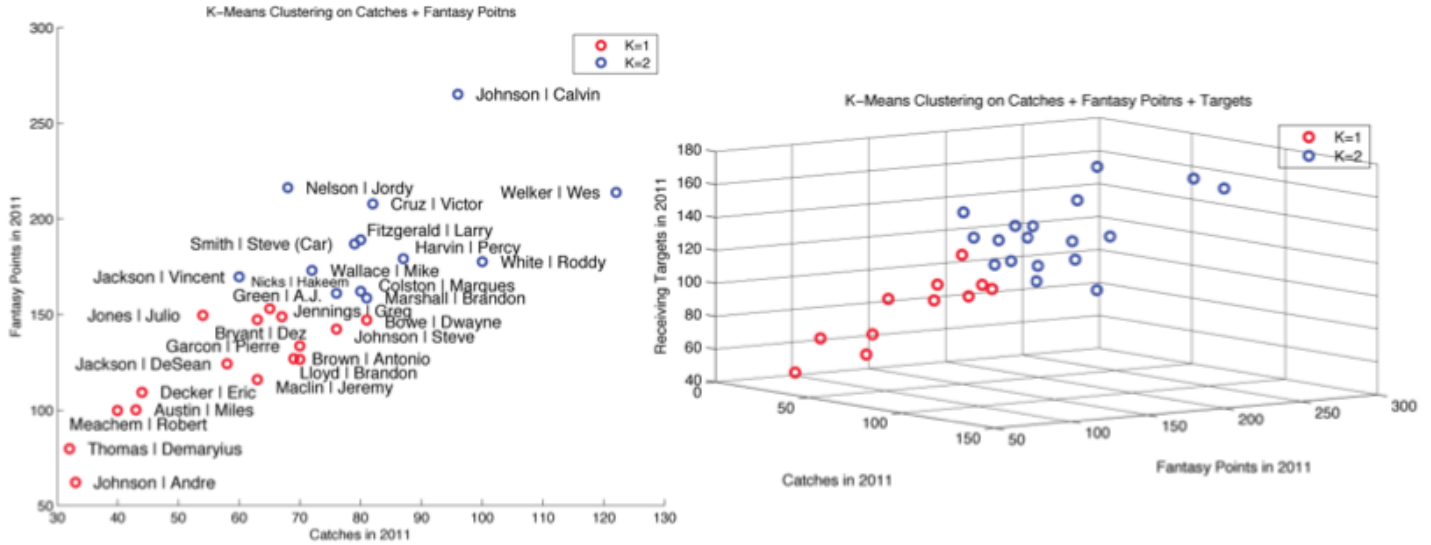


Figure 4: Results of k-means clustering in 2 and 3 dimensions. K-means was applied across multiple features, these are 2 example runs to illustrate the groupings. Note how in the left plot, k-means tended to separate high production, elite WRs in blue from role players and secondary receivers in red.

After grouping players using K-Means, each group is predicted individually. Based on the above regression results, both 2 and 6 feature regression will be tested. The results of each mixture's predictions are then combined to produce a single top 30 WR list for 2012, which is then scored using DCG as shown in Table 2.

# Mixtures	K-Means Features	DCG (6 Features)	DCG (2 Features)
2	Catches, Fantasy Points	1.7601	1.8063
2	Fantasy Points	1.7601	1.7925
3	Fantasy Points	1.3621	1.7880
2	Targets	1.2053	1.7877
2	Catches, Fantasy Points, Targets	1.1887	1.7863
2	Catches	1.1811	1.7814
2	Touchdowns	1.1107	1.7747
2	Yards	1.0857	1.4428

Table 2: K-Means mixture model results through several types of runs.

The best result obtained is a DCG of 1.8063 using 2 features (catches and fantasy points in previous year) for k-means and then 2 features (touchdowns and fantasy points in previous year) for regression. This beats ESPN's experts, but not Yahoo's.

To further improve the results, a new method must be employed. We turn to PCA.

The results of PCA and PCA Regression in Figure 5 show that the majority of the variance in our data is explained by the first 4 principle components. Using PCA regression with 10 principle components, following the method outlined in the 'Methods' section, we are able to out-perform the experts at both Yahoo and ESPN by producing a DCG of 1.8132.

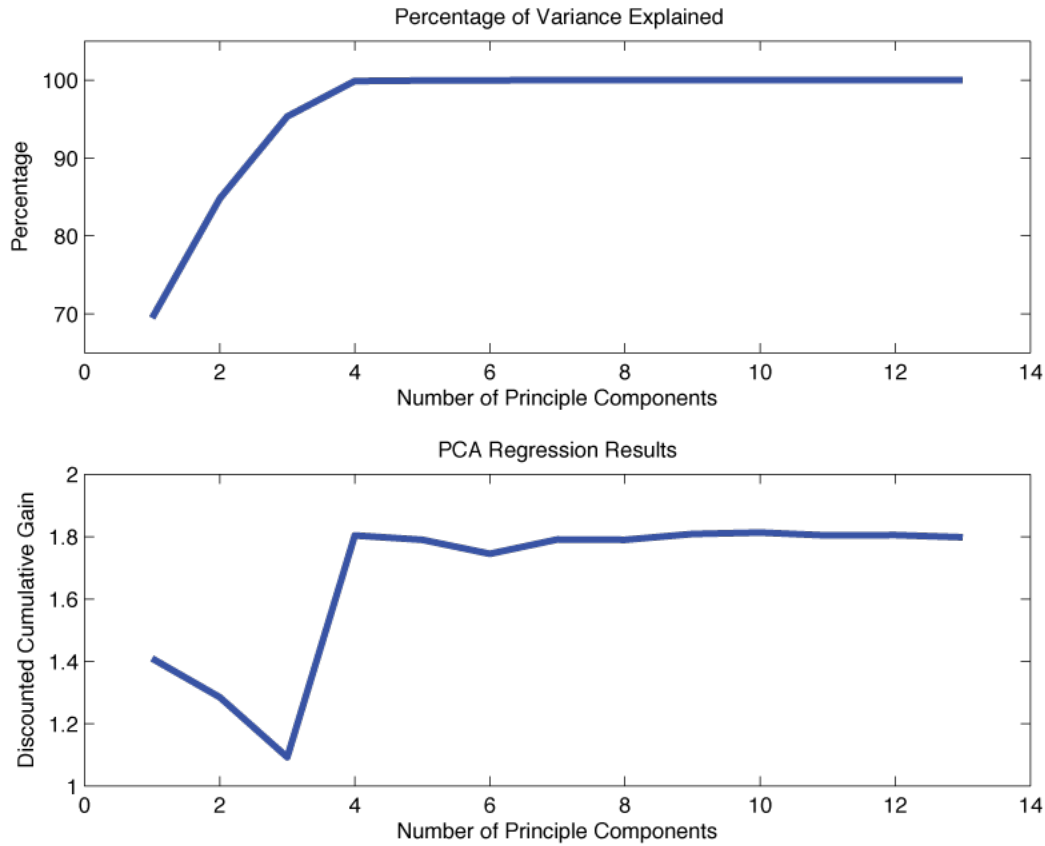


Figure 5: PCA and PCA regression run with varying numbers of principle components. Note how the DCG goes up as enough principle components are chosen to explain the majority of the variance.

4 Conclusions

Several machine learning approaches to ranking NFL receivers are shown. The best results come from PCA regression with 10 principle components. This method outperforms the experts at ESPN and Yahoo for the 2012 season. Future work could include examining expanding this to predict which players will have the best careers, predicting 2 or 3 years in advance, and refining the model by incorporating more data from previous years.

5 Bibliography

References

- [1] Mathews, Tim, et. al. Competing with Humans at Fantasy Football: Team Formation in Large Partially-Observable Domain. Association for the Advancement of Artificial Intelligence, 2012.
- [2] Fodor, Imola K. "A survey of dimension reduction techniques." Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Technical Report, June 2002.
- [3] Farquhar, Lee, et. al. Types of Fantasy Sports Users and Their Motivations. Journal of Computer-Mediated Communication, 2007.
- [4] Torres-Reyna, Oscar. Linear Regression, version 6.0 Data and Statistical Services, Princeton University.
- [5] Bookman, Matt. "Predicting Fantasy Football - Truth in Data." Stanford CS 229 Final Project. December 14, 2012.
- [6] Kapania, Nitin. "Predicting Fantasy Football Performance with Machine Learning Techniques." Stanford CS 229 Final Project. December 14, 2012.
- [7] Rasmussen, Carl Edward. The Infinite Gaussian Mixture Model. Advances in Neural Information Processing Systems, 2012.
- [8] Becker, Adrian, et. al. "An Analytical Approach for Fantasy Football Draft and Lineup Management." MIT ORC Technical Report. November 2010.
- [9] Steinbach, Michael, et. al. A Comparison of Document Clustering Techniques. Technical Report, University of Minnesota. May 2000.
- [10] McLachlan, G. J., et. al. A mixture model-based approach to the clustering of microarray expression data. Bioinformatics, 2001.