Predicting Optimal Game Day Fantasy Football Teams

Glenn Sugar and Travis Swenson
Aeronautical and Astronautical Engineering, Stanford University

Introduction

The popularity of fantasy sports has exploded in recent years. Websites such as Draft Kings and Fan Duel offer weekly competitions where people pay for the chance to build a fantasy team under a salary constraint for money. An article published in 2013 by Forbes estimates the market for fantasy football is $70 billion. With such a large market, even modest improvements in prediction capabilities can lead to large profits when leveraged over many games. The goal of this project is to use machine learning techniques to construct teams that consistently perform better than teams chosen by human intuition, and to obtain positive expected returns when playing these games.

In order to construct a winning team we need to maximize the number of predicted points scored by the players subject to a budget constraint. In order to do this we break our project into two parts. The first uses machine learning algorithms to predict the number of points any given player will score in a given game, while the second uses convex optimization to assemble teams with maximum expected return, and minimum risk.

Data Collection

In order to predict the performance of every player, we collected historical data on individual performance, opposition performance, and injuries. We created a webcrawler to collect this data from http://www.pro-football-reference.com and http://www.rotoworld.com /teams/injuries/nfl/all, resulting in over 40 statistics for every active players' performance in every game played since 2010.

Model and Feature Selection

We considered 3 basic feature types based off the statistics we collected.

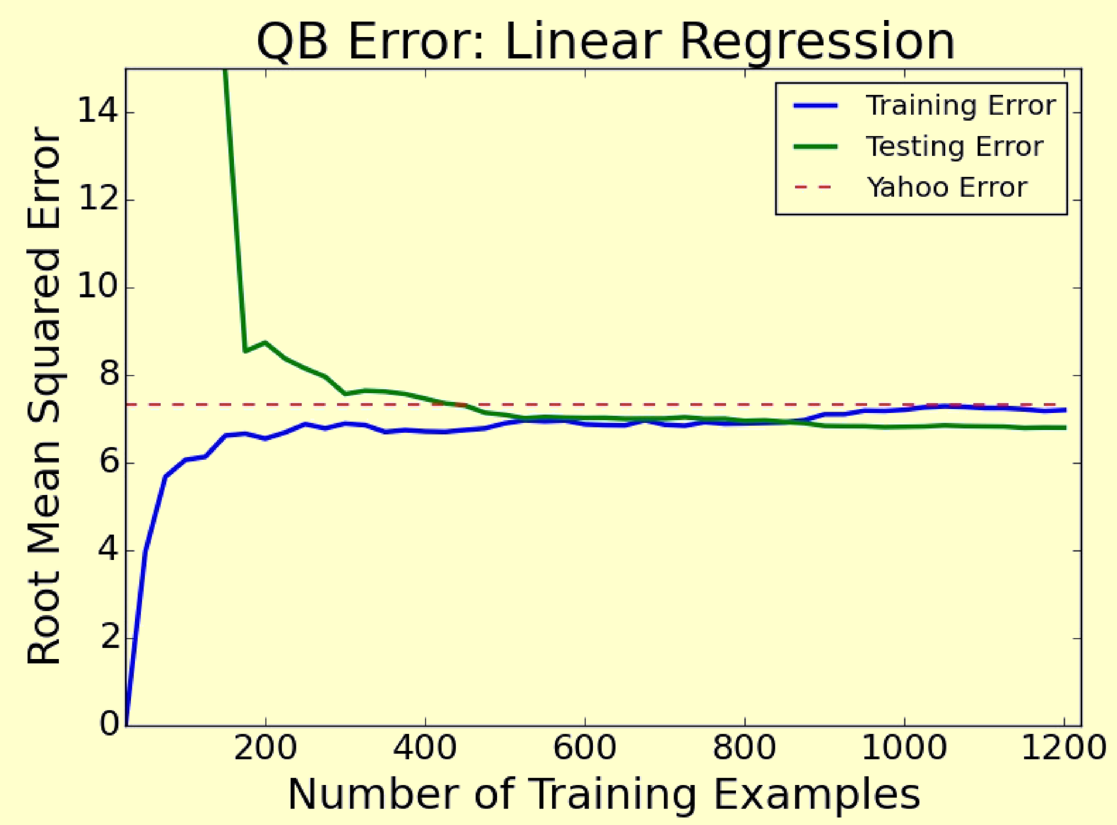
- **Career Features**: Running averages of performance metrics over a players entire career (e.g. average rush yards per game).
- **Current Features**: Information about the current game we are trying to predict (e.g. home vs away game).
- **Recent History**: Performance metrics of the player and their opposition in the *n* most recent games (e.g. rushing touch downs in each of the last 4 games) . This data could either be averaged, or included as *n* individual features.

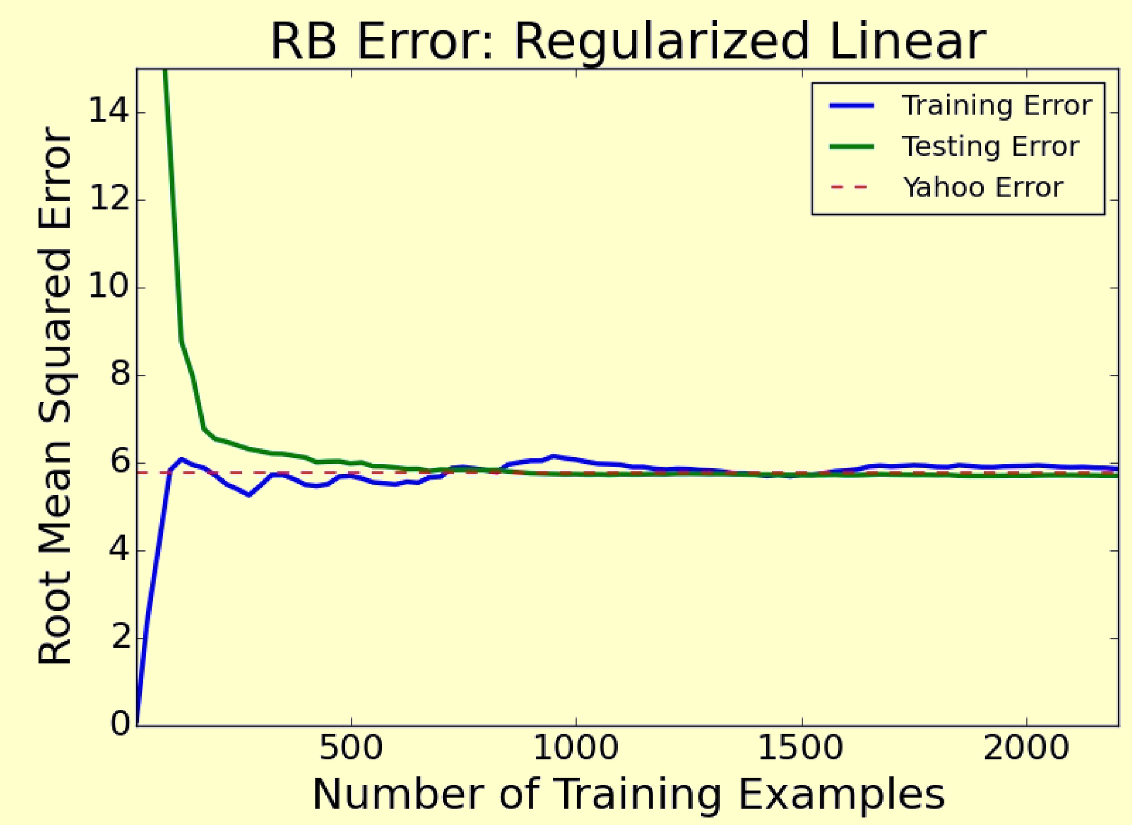
To select the optimal subset of features we used forward search selection, and experiment with then number of games *n.* We considered several machine learning algorithms, and picked a model for ever position by splitting our data into 70% training, and 30% test, and using cross validation.

The most successful algorithms included:

- **Linear Regression (LR)**
- **Ridge Regression (RR)**
- **Bayesian Ridge Regression (BR)**
- **Elastic Net (EN)**

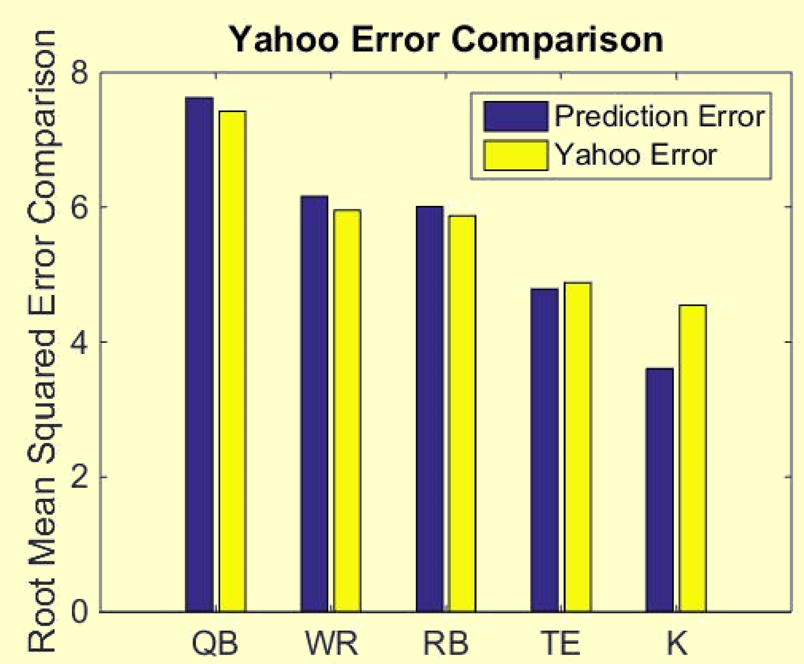
Model and Feature Selection



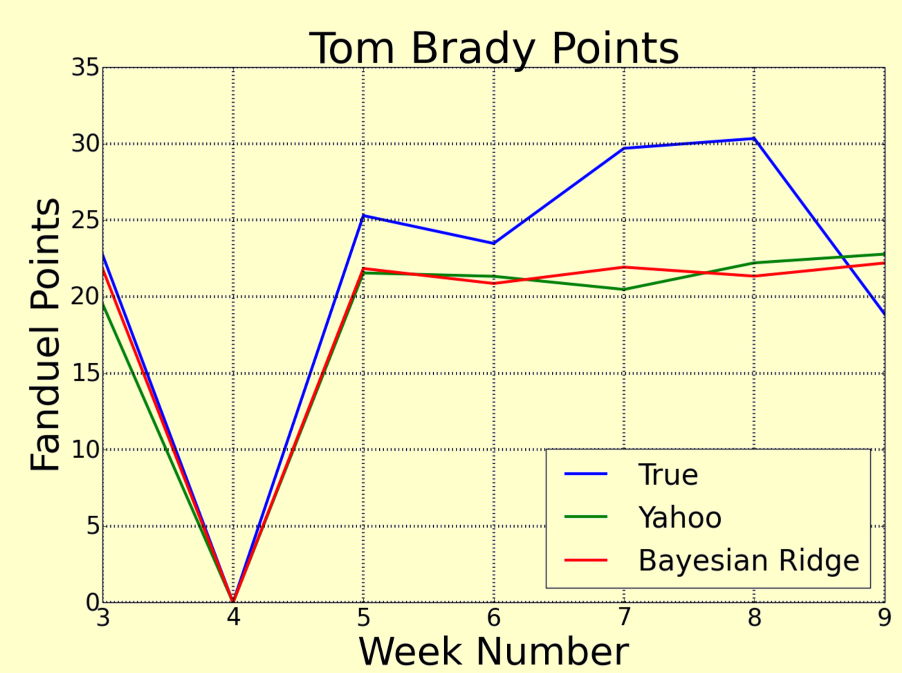


In order to diagnose the role of bias and variance in our generalization error, we ran our algorithms on training sets of increasing size and recorded the resulting training and testing error. As can be seen in the figures above, both errors converge to roughly the same value when using a large number of training examples, indicating variance is playing a low role in our generalization error. Instead, bias error is limiting the performance of our algorithms. This indicates we must further improve our model in order to beat Yahoo.

Results



Comparison of the RMSE for Yahoo and our predictor. Note that not all positions had the same optimal predictor.



Tom Brady's Fanduel points for weeks 3-9. Note that week 4 was a bye week so he did not play.

Our Bayesian Ridge predictor performed about as well as yahoo's predictions. We had slightly more error for QB, WR, and RBs, but slightly less error for TE and significantly less error for Ks. We used two sets of features when generating predictions: a complete set that included all the data, and a limited set that was obtained using feed forward feature selection.

	QB	WR	RB	TE	K
Regressor	BR	BR	EN	BR	EN
Features	All	Limited	All	Limited	All

Team Selector

The team selector can be framed as a binary linear program:

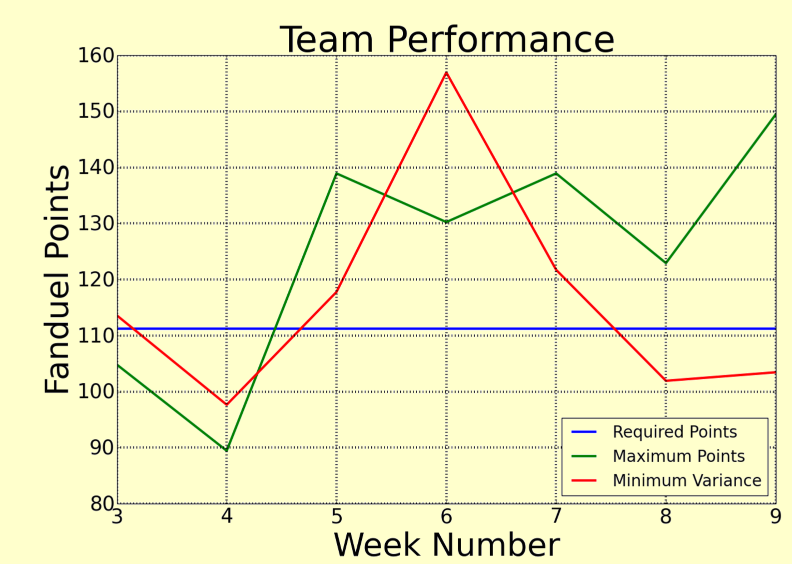
$$\min_x f^T x$$

$$\text{Subject to: } \begin{cases} x_i \in \{0,1\} \\ Ax = b \\ Gx \le h \end{cases}$$

When selecting a team, we can choose to either maximize points, or minimize variance. The average winning score of a Fanduel 50/50 league is 111.21 points. Therefore, choosing a team that has a projected total of 120 with a small variance might be better than choosing a team with a projected total of 125 with a high variance. The formulation of the binary linear program is given below for both cases:

Maximum Point Team:
x is the chosen team
b holds the position requirements
A holds players' position information
f holds players' predicted points
h is the maximum salary of the team
G holds players' salary information

Minimum Variance Team:
f holds players' variance
h is the maximum salary of the team and minimum points
G holds players' salary information and predicted points



Team performance for weeks 3-9 of the current season. The blue line is the average minimum amount of points needed to win a 50/50 Fanduel league in 2014. The maximum point team won 71.4%, while the minimum variance team won 57.1%.

Conclusion and Future Work

We have developed a fantasy football predictor that is on par with the Yahoo predictor. After generating the predictions, we solved a binary linear program to pick the optimal team to use for Fanduel based on the salary and position restrictions.

Future work would involve making improvements to the prediction model. Currently, we have a single predictor for all QBs. However, there are different types of QBs: some prefer to stay in the pocket, while others are more mobile and tend to have more rushing yards. By clustering QBs and identifying what kind of QB a player is, we could have better models to predict that player's performance. We would extend this idea to all positions, with the expectation of reducing the RMSE for every position.