6.867: Final Project

1. Introduction

2. Data collection

In order to collect season data, we primarily used publicly available data from `http://basketballvalue.com/downloads.php`. Basic statistics, including points allowed and points scored, were available for seasons 2005-2011, and more sophisticated statistics, including rebounds, were avilable for seasons 2008-2011.

Many of the features generated for training and testing our classifier were derived from this game data. In order to create a general proxy for how strong a team was, we calculated running averages of points allowed, points scored, win rate, etc. and used these as input features to our classifier. These running averages were initialized to equal numbers for all teams at the beginning of the season and were subsequently updated as more data became available. Because these numbers were arbitrarily initialized to equal at the beginning of the season however, running averages near the beginning of the season serve as less reliable data points, as outliers can significantly affect the running average.

These statistics were therefore plotted over time throughout the season in order to approximate when the running averages for each of these statistics stabilized to reliable averages. An example graph is given in Figure 1, in which the running average win rate for a given team is plotted over course of the season.

In general, the running averages of the game data stabilized after approximately each team's twentieth game of the season. When generating our training, validation, and test sets therefore, we only used data after each team's twentieth game in order to ensure that the features extracted from the data were representative of true team performance.

In order to compare our algorithm and our predicted spreads against Vegas and other betting authorities, we created a scraper to scrape relevant websites with both current and historical spreads. We primarily scraped this information from `http://www.sportsbookreview.com/betting-odds/nba-basketball/`.

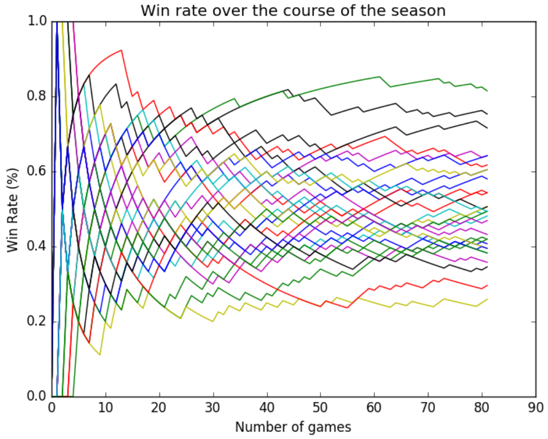

Figure 1. Win rate over time

3. Linear classifier

As a baseline, we started with a linear classifier using basic features including running averages of home/away team points scored, points allowed, Elo score, games won, games played, and win rate. Based on the 2006-2007 season, in predicting win/loss (as opposed to the spread of a win/loss), the linear classifier had an accuracy of 64.3% on training data and 63.3% on testing data.

After adding further features, such as home court advantage, were added to the model, the linear classifier has an accuracy of 72.3% on training data and 66.9% on testing data based on the 2008-2009 season.

We additionally explored the ability of the classifier to predict spreads on games, in which the margin of victory is taken into account instead of a simple win/loss. In this case, the accuracy metric was the absolute value of the difference between the predicted spread and the actual spread. Using basic features on the 2006-2007 season, the average absolute value spread error was 9.31 points for training data and 10.07 points for testing data.

After adding further features, the average absolute value spread error decreased to 8.53 points for training data but increased to 13.60 points for testing data.

	Win/Lose Accuracy	Spread Error
Basic features	63.3%	10.07
Extra features	66.9%	13.60

Table 1. Classifier performance on test dataset

This would imply that the classifier might be overfitting to the features that are provided as input to the algorithm, which would be expected for more complex architectures but was surprising in the context of the linear classifier explored here.

These results for the performance of the linear classifier are summarized in Table 1.

4. Logistic classifier

5. Breakdown of individual work

The breakdown of work between the two members of the group largely followed that given in the original project proposal. Andrea and Tyson jointly worked on preliminary review of the current techniques and the state of the art. Following this, Tyson was primarily involved in the collection of game data from past seasons, and Andrea created the scraper to collect data on historical spreads put out by various betting organizations on previous games. Andrea explored preliminary models such as linear regression, logistic regression, and regression with various basis functions and various degrees of regularization, and Tyson chiefly explored a neural network approach. As discussed during the project proposal meeting, we decided not to implement a convolutional neural network for this problem, as the architecture did not fit the structure of the problem as closely. We each tested the models we worked on against data test sets and historical spreads, and we split the writing up of the report equally.