David Lin and Ben Wagle
Professor David Sontag
Machine Learning
31 October 2013

Predicting the Outcome of Sports Competitions

Abstract
People love to analyze the statistics of sports. Numbers are a great representation of how a player and even a whole team perform. What we hope to create are classification algorithms that can take as input a difference in statistics for a matchup between two NBA teams and output a prediction of who will be the victor of the game. Our goal is to see how much the barebones statistics can be used to predict the result of a sporting event. We are also curious to find if there is an upper limit to how correct a prediction algorithm using only statistical data can be.

Data
On ESPN's website, there are statistics posted for every game during a season. Using PHP to scrape these values, we will create our own dataset consisting of the average stats of each team based on a team's games played up until the matchup we are attempting to classify. We will also infer statistics from the existing ESPN data, such as current win streak, a home or away game, and the record against a specific team.

We plan to train and validate on three sequential seasons, and in the end test on the following fourth season. We want to avoid the 2011-2012 season because of the NBA lockout that resulted in a shortened season. We think the shortened season will have changed statistics across the league and will result in inconsistencies with the classifier we achieve.

Preprocessing
We plan to normalize all statistics to be real-valued numbers ranging from 0 to 1. This normalization should be simple, as most statistics are already percentage values. We will create a data point for each of the 1230 games per season. Each feature contained in a data point will consist of the difference in each stat between the two teams, i.e. (home team statistic – away team statistic).

Plan of action
We will initially be working with linear classifiers. By dotting the feature vector with a weight vector, each stat will be augmented accordingly by how much it pertains to the ultimate outcome. The result will represent the home team winning or losing, which ultimately classifies the game. Our final weight vector will account for which statistics are more pertinent for a team to win (for example average points scored versus average number of fouls committed).

Algorithms
We will use a linear classifier, first **perceptron** and then **PEGASOS**, and we think these will generalize well, but we expect that our data will not be linearly separable. We will then also try **kernel methods** and **decision trees** to classify our data points. We think a kernel will be able to produce nonlinear boundaries that will perhaps classify data more accurately than a linear

method. With a decision tree, the lower branches of the tree should allow us to observe what statistics made the difference in the close matchups (statistically evenly matched teams).

Evaluation

We plan to evaluate our classifiers using **K-fold cross-validation**. We expect to train on three consecutive seasons and in the end, test on the next (fourth) season. When an incorrect prediction is made, we will update the predictor accordingly to better-reflect how the different statistics should be weighted. Using data that spans four seasons should not be a problem. Possible changes in statistical trends, such as importance of offense, or need for long-range shooters, happen over a larger time span than four seasons. Thus, our classifier should be applicable to a dataset distribution that spans a small number of seasons.

Realistically, getting zero test error is not possible, due to real-life factors that cannot be quantifiable or predicted, like the effect of injuries or player suspensions. But if we can successfully predict the outcome of **60-70%** of games, we will be greatly satisfied with our results and our algorithm.

Questions

The major question we hope to answer is whether or not statistics can be used to accurately predict an outcome of a sporting event. Ultimately our resulting classifier will be a formula for how to predict the outcome of a game. The resulting predictors from our algorithms will illustrate a correspondence between each individual statistic and its effect on the outcome of a game. For example, it will be interesting to see how offensive or defensive statistics compare in their pertinence to the overall result of a game.

If we can successfully predict NBA games, we also plan to expand the algorithm to other sports. We start with basketball because it has the largest number of data points (games) and a good number of features that we believe matter to the outcome of a game. We will potentially gather data for football and soccer and see how the algorithm translates across different sports. We will get different classifiers that apply to each sport, but are curious to see if our algorithm can attain a good classifier for each without changing the methods used.

Another, very simple tweak we plan to add is to train and validate one algorithm on a single sport, such as NBA games, and then test on data points that are football or soccer matchups. From this, we will be able to see if there is a universal classifier for all sports. If a basketball-trained classifier does not lose accuracy when tested on soccer games, what will that say about the overall difference between the two sports? Are they basically the same activity with a different shaped ball, or do different sports have attributes and tactics that set them apart. We hope this portion of the project will yield some interesting results that tell us about the nature of sport in general.

Timeline

By the first week of November, we will have scraped all teams' statistics for four NBA seasons and have a database of all the necessary statistics. From there, we will work to organize the data into the form we plan to use: normalizing the data features, calculating the team averages game to game, and constructing the data points, which are composed of the differences in opponents'

statistics for a given matchup. Then we will work on writing the prediction and update code. The perceptron and Pegasus code should only take some slight modification of the programs we already have written. As for the kernel method, we may use the Gaussian, but also want to look into other methods that may better fit and generalize our data. All three of these methods we have some experience applying. The decision tree will be the most time consuming, especially calculating good attributes to branch on and the order in which to branch on them. We will vary the size of our training and validate sets, as well as the number of iterations through the data, searching for a classifier that can predict the largest number of games correctly with low validation error. By Thanksgiving we hope to have code written that can roughly execute our vision for the all four algorithms. From here, we will work out kinks in the prediction and updates, and start to play with different sports and cross sport classification.