What It Takes To Win:
A Machine Learning Analysis of the College Football Box Score

John Hamann

Most advanced analysis of sports focus on predicting the results for the next game based on the results of previous games. For college football, the value of prediction extends beyond gambling due to the post-season format. College football has a large number of teams that play very few games, and this sparsity causes interesting problems for quantitative analysis. In addition college football is the only sport in the world in which six computer algorithms help determine who plays for the National Championship.[1]

Past Stanford projects have focused on using regular season games to predict the bowl games.[2] Predicting bowl games is particularly difficult, because bowl games occur four weeks after any other game, occur between relatively well-matched teams, and occur between teams in different conferences. Another project focused on betting against the spread.[3] Betting against the spread is difficult, because the spread is set by professionals who have a financial interest in understanding college football by any means including advanced statistical techniques. The computer rankings at the heart of the BCS also operate fairly efficiently as predictors with a 70% success rate in all games and a 54% success rate in games expected to be close.[4]

This project goes in a different direction and decides to take an a posteriori approach to college football. The analysis looks at the box score (statistical information) for an individual game without respect to any other games or outside factors (team rankings, past performance, home-field advantage) and predicts who won the game. How much underlying variance remains even if you know as much as possible about what happened in the game except the score? The difference between this approach and past approaches is the difference between estimating the current symbol from a noisy observation, $\hat{y}_n(x_n)$, and predicting the next symbol based on the past, $\hat{y}_n(y^{n-1},x^{n-1})$.

The Data Set

This project was inspired by the ungodly amount of college football data posted on a blog.[5] The data set is humongous with every game from 1998-2010. The 8000 games since 2000 have 50 box score stats apiece for the home team and visiting team. Many of these statistics can indirectly give you the final score (points per play, rushing touchdowns, safeties, field goals, and extra point attempts) and need to be removed. The second cut removes duplicates such as receiving yards which is the same as passing yards. The third step is to trim some of the more esoteric data such as punt return yardage, kickoff return yardage, fumble return yardage, and yards per punt. This step tosses some very good information because the stats are difficult to handle intelligently. For example, 200 punt return yards on 4 punts strongly implies that the punt returner either scored or got very close to the goal line which is valuable.

This led to the trimmed feature set. This trimmed feature set was still a little unwieldy for the machine learning algorithms. The final feature set removed features that could be derived from other features and features where the additional information was small. When evaluating the algorithms, some of these features were added back in, but they did not significantly improve performance. The final feature set had 10 features that can be found for any football game and capture the core of the game: rush yardage, pass yardage, rushing attempts, pass attempts, pass completions, first downs, punts, penalties, fumbles and interceptions.

Algorithms and Results

The problem as stated is relatively straight forward. It is a binary classification problem with 10 features per team. For this project I applied some of the simpler machine learning algorithms to this problem to see how well the box score predicts who won the game. Where necessary, the classifiers set aside 80% randomly for training and 20% for testing. Since the feature set is drawn from a single game rather than cumulative season totals, causality is not a problem, and the games can be drawn independently across all weeks and seasons. For this problem, the data set was large enough that over-fitting was not an issue. Table 1 shows the error rates for the various algorithms.

Turnover Margin Classifier

The Turnover Margin Classifier says that the team who had the fewest turnovers won the game. If both teams had the same number of turnovers, then the team with the most yardage won the game. This incredibly simple classifier correctly identified the winner 75% of the time suggesting that turnovers are a major factor in the outcome of a football game.

Algorithm	Error Rate
Turnover Margin Classifier	25.6%
Handpicked Weights	17.9%
Nearest Neighbor	16.4%
Naïve Bayes	14.3%
Logistic Regression	9.85%
Perceptron Algorithm	9.85%

Table 1: Algorithm Error Rates

Handpicked Weights Classifier

The simple weights classifier used weights guessed by hand to predict who would win the game. It applies the weights to the difference between the stats of team A and team B. The same weights are applied to both team A and team B, since the outcome should be symmetric. For this project there is no intercept term, since it is a zero sum game. It is guaranteed that half will win and half will lose. This simple weight with no additional tweaking led to an 18% error rate.

$$\hat{Y} = \begin{cases} 1 & if\ w^T(x_A - x_B) \geq 0 \\ 0 & if\ w^T(x_A - x_B) < 0 \end{cases}$$

Nearest Neighbor

I ran the standard nearest neighbor algorithm for a variety of distance measures and values of k. Euclidean distance between normalized features and majority vote of the 51 nearest neighbors gave 16% error.

Naïve Bayes

Naïve Bayes had an error rate of about 14%. For a few of the features (such as yardage) where the number of possible values was large, I discretized to a smaller number of bins.

Logistic Regression

I ran standard logistic regression with adaptive gradient descent to solve for better weights than those I picked by hand. The value of α was decremented by an order of magnitude every 30000 iterations starting at $\alpha = .1$ and ending at $\alpha = 10^{-15}$. This converged to a set of weights with 10% error.

Perceptron Algorithm

The perceptron algorithm ran under the same problem set up as logistic regression with a hard decision rather than the soft decision allowed by the sigmoid function. The algorithm normalized the weights after each iteration for slightly better performance. Once again the value of α was decremented by an order of magnitude every 30000 iterations starting at $\alpha = .1$ and ending at $\alpha = 10^{-15}$. This led to an error of 10%.

Evaluation of Classifier Results

The logistic regression and perceptron algorithm converged to similar values for the weights. Table 2 shows the weights that the perceptron algorithm derived and applies them to the box score of the 2006 Rose Bowl National Championship Game between Texas and USC. Texas outperformed the Trojans in all but net passing yardage with the lone USC interception being the primary difference in the game.

The final numbers give an error of about 10%. There are certainly more advanced machine learning algorithms that may give better results. However, the window for improvement is not particularly large, and I expect that advanced algorithms will not offer a major advantage. This section evaluates what the resulting classifiers imply about what it takes to win football games.

Feature	Weights	Texas	USC	Texas	USC
Net Rushing Yards	0.002955	289	209	0.85385	0.61749
Net Passing Yards	0.003392	267	365	0.90569	1.23812
Rushing Attempts	-0.036382	36	41	-1.30975	-1.49166
Pass Attempts	-0.110320	40	41	-4.41280	-4.52312
Pass Completions	0.074790	30	29	2.24370	2.16891
First Downs	0.081161	30	30	2.43483	2.43483
Number of Punts	-0.291680	2	2	-0.58336	-0.58336
Number of Penalties	-0.009914	4	5	-0.03965	-0.04957
Fumbles Lost	-0.710750	1	1	-0.71075	-0.71075
Interceptions Thrown	-0.619650	0	1	0.00000	-0.61965
		Classifier:	Texas	-0.61824	-1.51876
		Result:	Texas	41	38

Table 2: Sample Box Score and Classification

Since the error rate was 10%, this means that what happens in-between the goal lines provides a fairly accurate observation of the final result. If the statistics collected did not lead to a good prediction of the final result, then that would suggest the game has significant latent randomness and that more descriptive statistics are needed.

The weights conform very nicely to conventional football wisdom, and both methods converged to very similar values. Table 3 gives the weights for both algorithms and the weights for when first downs were removed from the list of features. The most important fact is that turnovers are incredibly costly; an interception is worth approximately 182 passing yards. Punting is about half as costly as a turnover. Penalties are a surprisingly negligible component of the statistics.

The value of a yard is the same whether it is passing or rushing; the difference is that a pass has a higher risk/reward component since passes frequently fall incomplete for zero yards. The weights can also verify whether a single play or sequence of plays was successful. For example a 1 yard rush on third down leads to a first down and increases the net margin over the other team by 0.047. An unsuccessful rush on third down and subsequent punt decreases the net margin over the other team by 0.328.

	Perceptron Algorithm	Logistic Regression	Perceptron Without First Downs
Net Rushing Yards	0.002955	0.003394	0.005118
Net Passing Yards	0.003392	0.003355	0.005257
Rushing Attempts	-0.036382	-0.045232	-0.018601
Pass Attempts	-0.110320	-0.109950	-0.095430
Pass Completions	0.074790	0.068244	0.086468
First Downs	0.081161	0.096268	N/A
Number of Punts	-0.291680	-0.285300	-0.317000
Number of Penalties	-0.009914	-0.002678	-0.032076
Fumbles Lost	-0.710750	-0.708460	-0.699040
Interceptions Thrown	-0.619650	-0.623380	-0.626770

Table 3: Weights for Various Algorithms and Feature Sets

When first downs are removed from a feature set, it is easy to calculate the number of yards needed for an average rusher or passer to increase the margin over his opponents if you ignore the costs of turnovers and penalties. If first downs are included, then each run has a potential to be a first down depending on the situation, so the calculus becomes more difficult. The 3.63 yards per carry average agrees with conventional wisdom that a running back needs to average 4 yards per carry to be successful. The 7.13 yards per passing attempt also agrees with conventional wisdom of what a quarterback needs to achieve to be considered successful. These numbers allow one to evaluate the performance of running backs and quarterbacks.

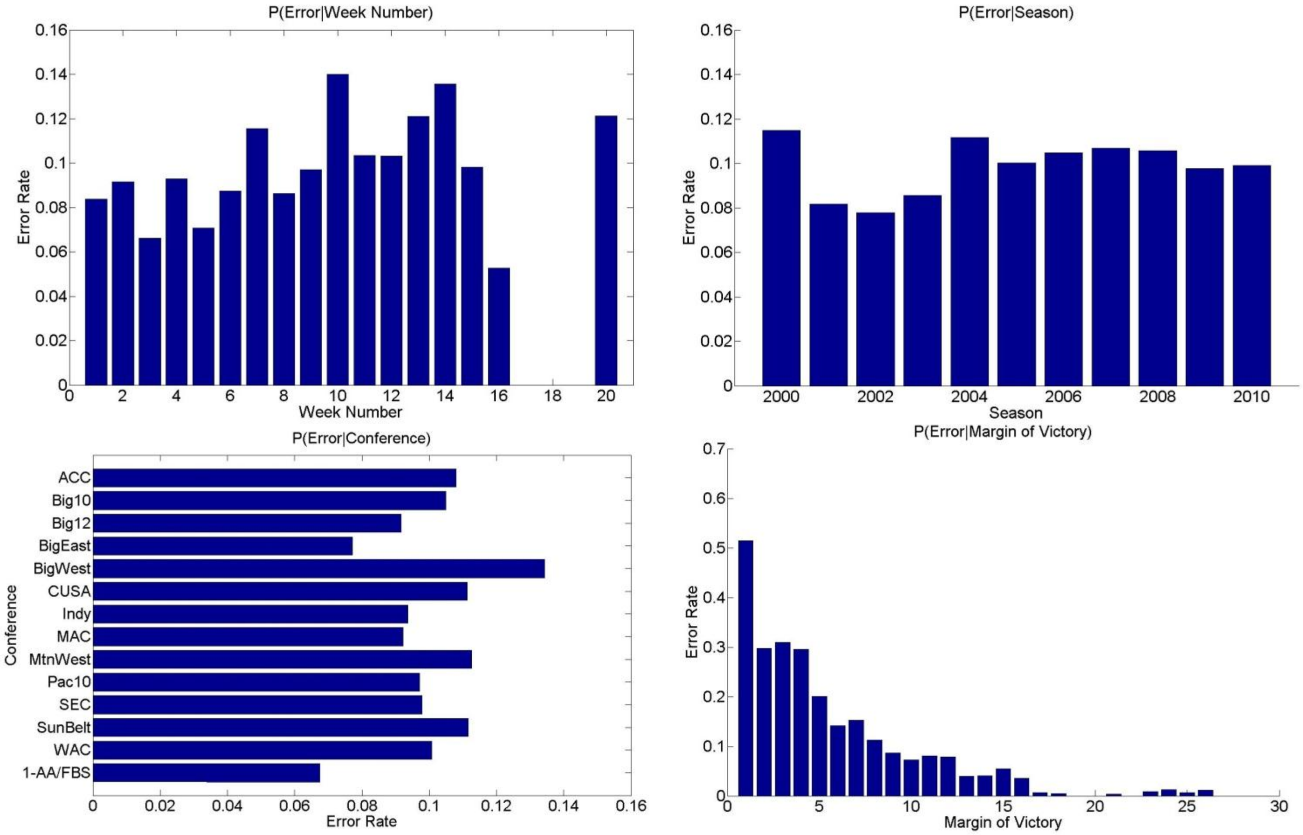
$$\frac{Yards}{Rush} = \frac{\left(\frac{Pts}{Att}\right) + prob(Fumble) * \left(\frac{Pts}{Fumble}\right) + prob(Pen) * \left(\frac{Pts}{Pen}\right)}{-(\frac{Pts}{Yard})} = \frac{-.018 + 0 + 0}{-.0051} = 3.63$$

$$\frac{Yards}{Pass} = \frac{\left(\frac{Pts}{Att}\right) + prob(Comp) * \left(\frac{Pts}{Comp}\right) + prob(Int) * \left(\frac{Pts}{Int}\right) + prob(Pen) * \left(\frac{Pts}{Pen}\right)}{-(\frac{Pts}{Yard})} = \frac{-.095 + \frac{2}{3} * .086 + 0 + 0}{-.0052} = 7.13$$

Evaluation of Errors

There are a couple of possible explanations for where the 10% error rate comes from. Part of the 10% may be due to luck in realms that we are not allowed to see. For example, the difference in a game between two perfectly matched teams may be a missed 50 yard field goal. Another part of the error may be due to extrinsic data about the game and the teams involved such as home field advantage and talent level. The classifier does provide a method to say which team statistically played better and compare that to who actually won the game.

The misclassification errors are scattered fairly uniformly across games. Figures 1-3 illustrate that the error rate is relatively consistent across weeks, years and conferences. Figure 4 shows that when the game was competitive (margin of victory was small) the classifier made a lot of mistakes. As the margin of victory increases, the games become less competitive statistically, and the classifier makes fewer mistakes. Figure 1 shows that errors increase slightly as time passes perhaps due to the fact that conference play is more evenly-matched and competitive.



Figures 1-4: Probability of Error Given Season, Week, Conference and Margin of Victory

If one team is more skilled than another, then it seems reasonable that they would be more likely to win close games. Table 4 shows the probability that Team A beat Team B given that Team A outplayed Team B statistically. Even if an FCS team has a better statistical outing and outplays a BCS team, they still only win 51% of the time. This compares poorly to the expected 90% record. This gap could be due to the differing levels in talent, coaching or home field advantage. Table 5 shows the value of home field. When the home team plays better than the visiting team, they win 93% of the time. When the visiting team plays better than the home team, they only win 86% of the time. As might be expected, playing a close game at home is a significant advantage.

		Team B		
		BCS	**Mid-major**	**1-AA/FCS**
Team A	**BCS**	89.48%	96.65%	98.91%
	Mid-major	79.08%	88.55%	96.22%
	1-AA/FCS	51.72%	68.49%	N/A

Table 4: Talent Disparity

Better Team	**Accuracy**
Home	93.13%
Neutral	88.28%
Away	85.56%

Table 5: Home Field Advantage

Conclusion

The non-scoring statistics do a remarkable jump of capturing who won a football game. With 10 basic features for both sides, a football fan can predict who won the game with 90% accuracy. Most of the errors occur in the very close games where it is difficult to state with certainty which team had the better game. The accuracy is remarkably consistent over weeks, seasons, and conference. Home field advantage allows home teams a better chance to win the game despite being outplayed. Teams in BCS conferences have better talent and coaching which may explain why they perform better at closing out games against lesser opponents and winning games in which they were outplayed.

The weights in the linear classifiers offer some insight into what wins football games. Offensive turnovers can only be overcome by forcing defensive turnovers or through complete domination in other facets of the game. Yardage counted the same no matter where it came from; the difference between rushing and passing is due to the mechanisms of attempts and completions. Testing with additional statistics provided very little gain in performance, but there were a few statistics that were not included. Sacks, tackles for loss, quarterback hurries, and passes broken up are a few defensive statistics that might be useful.

The weights may have some value in predicting future games. They can provide a quick way to incorporate a variety of features into a single number. I tested a quick predictor which applied the weights to the cumulative season statistics and had 65% accuracy on the back half of games for the 2005 season.

While prediction has value for fans and bettors, post-game classification has value for coaches and players. The weights allow coaches to examine the performance of the passing offense separately from the performance of the running game. Defenses can be compared across teams with a more efficient measure than yards allowed. This project has shown that the non-scoring statistics in football have a high descriptive capability of who won the game.

References

[1] “The BCS Formula.” <http://www.bcsknowhow.com/bcs-formula>
[2] “A Better BCS.” Rahul Agrawal, Sonia Bhaskar and Mark Stefanski. (CS229, Fall 2010)
[3] “Beating the NCAA Football Point Spread.” Brian Liu and Patrick Lai. (CS229, Fall 2010)
[4] “The BCS System: Rate Not, Lest Ye Be Rated.” Tom Brennan. (Posted 12/14/2011).
< http://barkingcarnival.fantake.com/2011/12/14/the-bcs-system-rate-not-lest-ye-be-rated/>
[5] “An Ungodly Amount of College Football Data.” (Updated 7/11/2011).
<http://thenationalchampionshipissue.blogspot.com/2005/08/ungodly-amount-of-football-data.html>.