

---

# Upset Prediction in College Football

---

**A. Santiago Padrón**

Department of Aeronautics and Astronautics  
Stanford University  
Stanford, CA 94305  
padronas@stanford.edu

**Jeffrey D. Sinsay**

Department of Aeronautics and Astronautics  
Stanford University  
Stanford, CA 94305  
sinsay@stanford.edu

## 1 Introduction

The last few decades have seen the rise of more advanced statistical approaches to looking at sports. Many of these approaches build on the methods of baseball sabermetrics, famously advocated for by Bill James[1]. Similar ideas have been extended to American football by Carroll *et. al.*[2]. Interest in these methods has grown as they have shown themselves to be good predictors of team and player performance, useful information for franchise management, fantasy leagues and sports betting. In recent years, the use of machine learning to develop models for the prediction of football game outcome at both the college and professional level has been considered[3, 4]. In the project we seek to extend these ideas by applying machine learning algorithm for the prediction of upsets in college football. We feel that an approach which modifies the problem of predicting game outcomes to one focused on identifying cases where other prediction models have higher probability of incorrectly labeling outcomes presents several novel opportunities to exploit.

## 2 Data Collection

Data for NCAA Football Bowl Series Division teams from the 2013 season was used to train and test our prediction models, the features used in our model are summarized in table 1. The website cfbstats.com was used to gather in-game statistics for each team for weeks 1 through 14 of the season. Using traditional football statistics, a set of seven team-on-team performance features were created. These were based on taking a differential statistic between opposing teams (e.g. difference between underdog's average offensive rushing yardage per and favorite's average defensive rushing yardage allowed per game). The teams' statistical performance was recalculated for each week of the season using only games previously played at that point in the season. Betting spread and ranking information was also considered. These essentially represent input of expected outcome and team strength from other models, which we treat as black boxes. The betting spread data was scrapped from the betting website donbest.com. For the rankings of the college football teams, we considered the FoxSports rankings found in cfn.scout.com. The data was collected and processed into text feature and results files using a set of python scripts we developed for this project.

## 3 Methods

Our basic approach to identifying upsets was to classify them using a number of supervised learning algorithms. We looked at two basic types of upsets, one where the team favored by 1 or more points loses, and a second, "major upset", where the team favored by a touchdown (7 points) or more loses. Of the 723 games analyzed we found 136 upsets (19%), of which, 57 (8%) were major upsets. This indicates a significant skewing of the classification groups, particularly for the major upsets. Given the nature of our problem, we are therefore particularly concerned with making Type I errors in which we incorrectly classify a game as an upset. We examined the models precision and recall as

Feature	Source
Home Team Favored?	cfbstats.com
Average Points Score Differential	cfbstats.com
Underdog Offensive Rush Differential	cfbstats.com
Underdog Defensive Rush Differential	cfbstats.com
Underdog Offensive Pass Differential	cfbstats.com
Underdog Defensive Pass Differential	cfbstats.com
Underdog Turnover Margin	cfbstats.com
Underdog Time of Possession Margin	cfbstats.com
Game Spread	donbest.com
Underdog Rank	2013 CFN Rankings
Favorite Rank	2013 CFN Rankings

Table 1: Features used by models in predicting upsets.

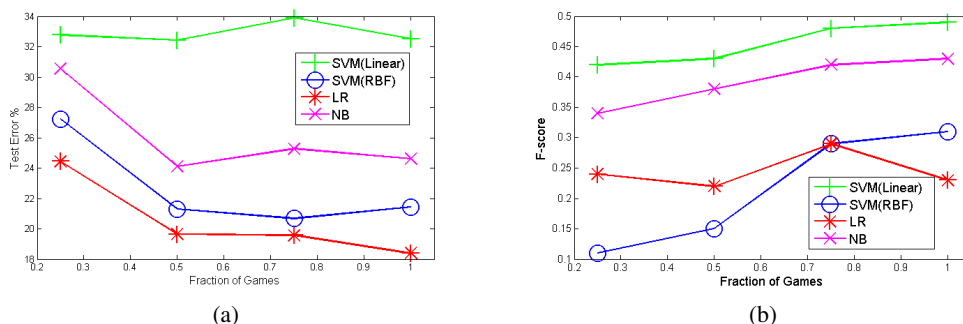


Figure 1: 10-fold validation averaged results: (a) testing error,(b) F-score.

captured by the F-score

$$F = 2 \frac{precision * recall}{precision + recall}, \quad (1)$$

where,

$$precision = \frac{TP}{TP + FP} \quad \text{and} \quad recall = \frac{TP}{TP + FN}. \quad (2)$$

The rate of improvement of the F-score started to level off as we increased the number of games considered to include all the games of the season (Fig 1b). Also, the testing error achieved reasonable level of convergence (Fig 1a) for all of the four learning algorithms we considered: Logistic Regression (LR), Naive Bayes (NB), SVM with Linear Kernel (SVM-Linear) and SVM with Gaussian Kernel (SVM-RBF). For all the learning algorithms we used the versions of them found in Matlab and it's toolboxes.

To test the algorithms and for all of the following results we used  $k$ -fold cross validation ( $k = 10$ ) on our entire data set.

## 4 Results

Figure 2 shows the distribution of upsets by spread. Upsets are significantly more common when the spread is low, as would be expected, since closer spreads indicate more evenly matched teams. The more games an algorithm correctly predicts as an upset, the more it also mislabels upsets. This trend is even true, when we considered the heuristic of always picking the highest rank team to upset. We were surprised to find out that pretty much all of our models, except SVM-Linear, don't make predictions for the higher upsets. We explored lowering the confidence needed for the algorithm to predict to encourage the algorithm to pick upsets. For our logistic regression model we changed the decision boundary by changing the probability above which an upset is predicted to explore its effect on precision and recall (Fig. 3). We found that increasing the number of correct upsets we predict

(recall), comes at the expense of misclassifying games as upsets when there are not, thus sacrificing precision. This behavior is unacceptable since taking actions based on the misclassification of a game as an upset, will have significant more cost than failing to identify an upset and the resulting lack action.

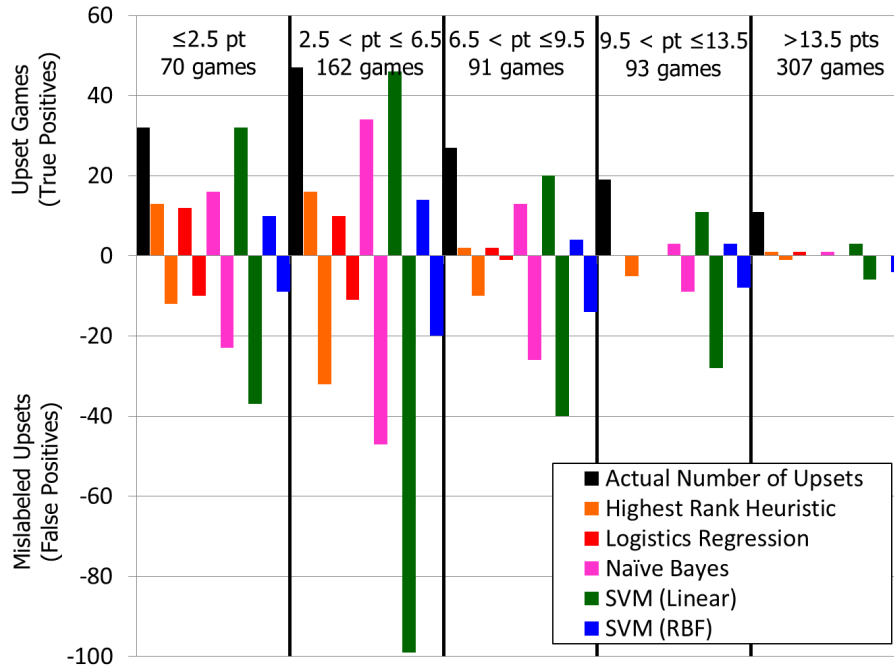


Figure 2: Frequency of upset and predictive accuracy of algorithms examined versus spread. Difficulty in predicting major upset and unfavorable labeling of false positives by some algorithms is evident.

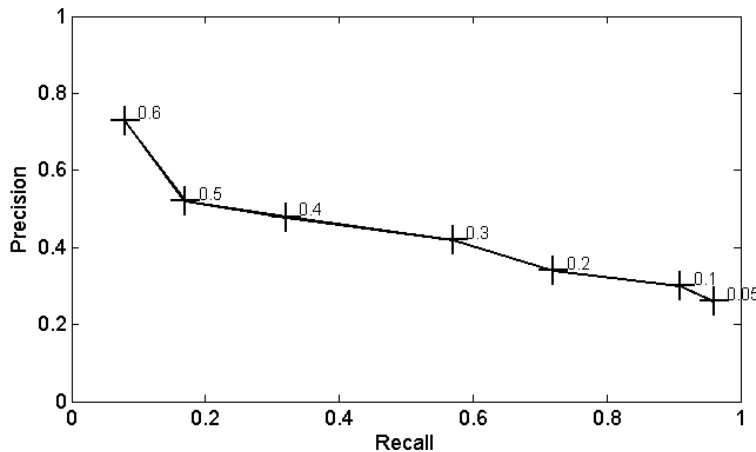


Figure 3: Trade-off in precision and recall for the logistic regression algorithm.

We further explored the tendency of the models to not predict major upsets by performing a principal component analysis (PCA) of the season data. From Figure 4 we see that it would be difficult to separate upsets and especially major upsets. The PCA also corroborates the trend shown in the recall precision diagram (Fig. 3) and in the histogram (Fig. 2) that if you want to predict more upsets correctly you will inevitably incorrectly predict a large number of upsets.

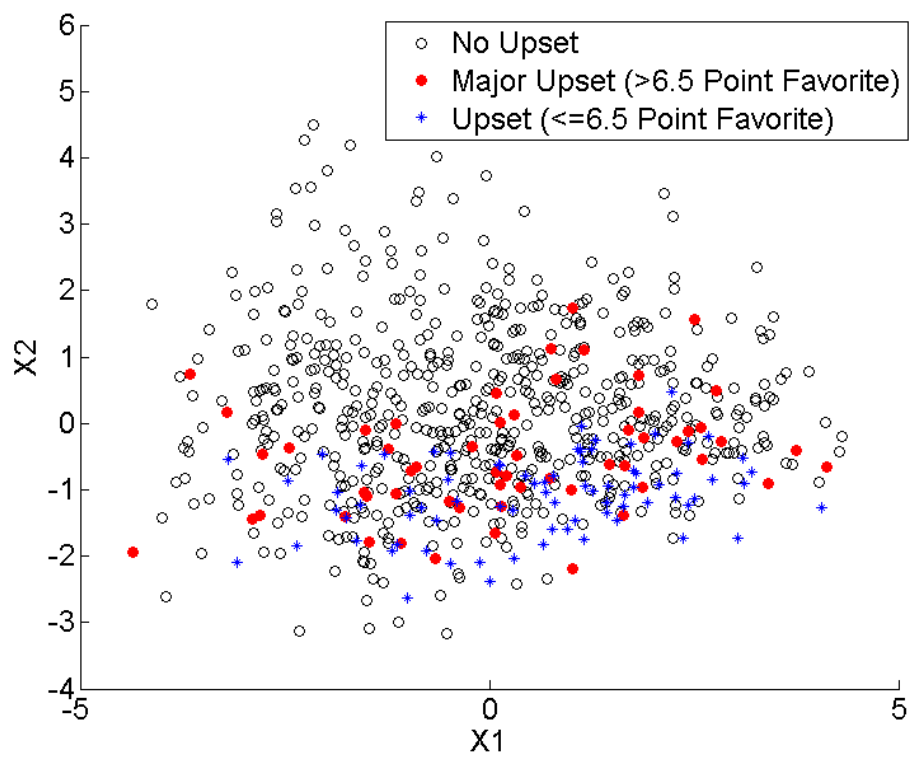


Figure 4: Principle component analysis of our 13 features showing the difficulty in separating upsets, particularly major upsets, out from other games.

## 5 Conclusions

Error analysis of the methods we attempted showed that our learned models had great difficulty in predicting upsets when one team is favored by a touchdown or more. Adjustment of the decision threshold, while increasing the number of true positive cases identified, came at the significant expense of additional false positives. Principal component analysis indicates that this is likely due to the fact that we have yet to identify a set of features which clearly separates out upset games, particularly when a team is favored by a touch down or more. Aside from the relatively poor performance of the SVM with linear kernel, the algorithms investigate appear to give similar performance.

Several different approaches for improving the features used exist and deserve further investigation. By constructing the problem as an upset prediction labeling it is possible to take as feature input a more rich set of prediction information from other models. Looking at the variance of team ranking and spread accross more external computer models than the two we used as data sources may provide a good feature for identifying games with higher uncertainty on the outcome. Additionally, while not presented here, we did investigate briefly the use of expert opinion, in the form of sports writers game picks for a subset of games. These expert opinions potentially provide a way to capture unique facts related to injuries, momentum, weather that might be cause of a team to perform dramatically different in a game. An experiment including expert opinion on a subset of Pac-12 games did show promise. Unfortunately gathering and standardizing this data is significant task, and we ultimately abandoned it as infeasible for the current project. Work in modern football game statistics also indicates that looking at statistics in a more complex fashion, in particular the interrelationship between various raw statistics could be powerful.

Defining an algorithm to systematically explore combinations of the above features in more complex ways than afforded by an SVM or similar learning algorithm could provide significant pay-off. Neural networks[5] and genetic algorithms with variable length chromosomes[6, 7] have shown promise for feature selection in other applications, and may be appropriate here.

## References

- [1] Sabermetrics. In *Encyclopedia Britannica*. 2013.
- [2] B. Carroll, P. Palmer, and J. Thorn. *The Hidden Game of Football*. Warner Book, New York, NY, 1988.
- [3] B. Liu and P. Lai. Beating the ncaa football point spread. December 2010.
- [4] M. Bookman. Predicting fantasy football. December 2012.
- [5] R. Setiono and H. Liu. Neural-network feature selector. *Neural Networks, IEEE Transactions on*, 8(3):654–662, 1997.
- [6] D. E. Goldberg, B. Korb, and K. Deb. Messy genetic algorithms: Motivation, analysis, and first results. *Complex Systems*, 3(5):493–530, 1989.
- [7] R. Srikanth, R. George, N. Warsi, D. Prabhu, F.E. Petry, and B.P. Buckles. A variable-length genetic algorithm for clustering and classification. *Pattern Recognition Letters*, 16(8):789 – 800, 1995.