SYS 6018 Final Project Report

Luke Kang (sk5be), Runhao Zhao (rz6bg), Jing Sun (js6mj)

Identify the Problem

March Madness is the annual NCAA basketball competition. 68 teams compete in a bracket style tournament to win 6 games and subsequently the tournament. We are interested in applying machine learning techniques to predict the winning team for each of the game in the March Madness. Due to the large monetary amounts involved in betting, sports prediction becomes increasingly important. Club managers and owners are also able to understand and formulate strategies needed to win matches based on machine learning models. (Ogus, 2016)

The current situation of this problem is that there is no effective model which can successfully predict a perfect NCAA tournament bracket. According to statisticians, the odds of filling out a perfect bracket are one in around 9,223,372,036,854,775,808. While our ultimate goal of this project is to build a model to predict a perfect NCAA tournament bracket, the alternative situation is to predict the teams in the semifinals, and the probabilities of predicting perfect final four brackets in the past three years are 1.61%, 0.09% and 0.003%, respectively. Our primary objective of this project is to achieve a better predictive accuracy and decrease the classification errors in our predictions, and thus help club managers and owners make better decisions. Our objective would bring to related-stakeholders' attention. In specific, as it is one of the most famous annual sporting events in the United States, host cities and their local businesses are certainly influenced by the result / progress of March Madness in terms of economics impact. For example, it was estimated that in Philadelphia, the local economy would generate nearly $18 million when they hosted the East Regional in 2016. (Ogus, 2016)

Define Objectives and Metrics

As mentioned, the objectives are to produce a model that minimizes the classification errors and optimizes the predictive accuracy. Our evaluation metrics are stated as follows (Katz, 2015): the scoring function awards one point for each correct pick in Round 1 (Round of 64), two points in Round 2 (Round of 32), four points in Round 3 (Round of 16), eight points in Round 4 (Round of 8), 16 points in Round 5 (semi-final), 32 points for the ultimate winner.

Understand the State-of-the-Art

The historic NCAA tournaments (2003 - 2016) data have been investigated, and the method that received the best score is gradient boosting. Other methods, such as logistic regression and decision trees, were also applied to the data (March Machine Learning Mania 2016). Wright and Wiens (2015) found that using the scoring system discussed in the previous section, a mean of 85 points was scored per bracket. In specific, this was much higher than the mean score from 2016 which had only 69 points. This problem is particularly difficult to solve because it is often impossible to take into account the physical and mental states of the teams during the tournament when performing the analysis.

Define Hypotheses and Approach

The hypothesis of this project is that support vector machines will produce the best prediction results compared to logistic regression, decision tree, and random forest for the analysis of March Madness basketball data, since support vector machine is usually better when the data size is small and should be able to make better predictions. The assumption on svm is that the input should not consist of categorical variables. If exist, these variables would need to be transformed using variable encoding methods. The datasets we will be using consist of multiple csv files; 'RegularSeasonDetailedResult.csv' contains the game results and statistics for main season games. We will use each team's average main season yearly performance statistics as the predictors for which team will win the tournament games. 'TourneyDetailedResults.csv' has the outcome statistics for the March Madness games. 'TourneySlots.csv' and 'TourneySeeds.csv' show how the tournament games are conducted. 'Teams.csv' shows which teams correspond to which unique IDs. Moreover, while data cleaning process is required here, it is believed that our data is not subject to bias.

Also, the given basketball game data contains information from 2003 to 2016, we will use data from 2003 to 2013 as the training set, and data from 2014 to 2016 as the test set. We will perform cross validation on models.

Execute Approach and Report Results

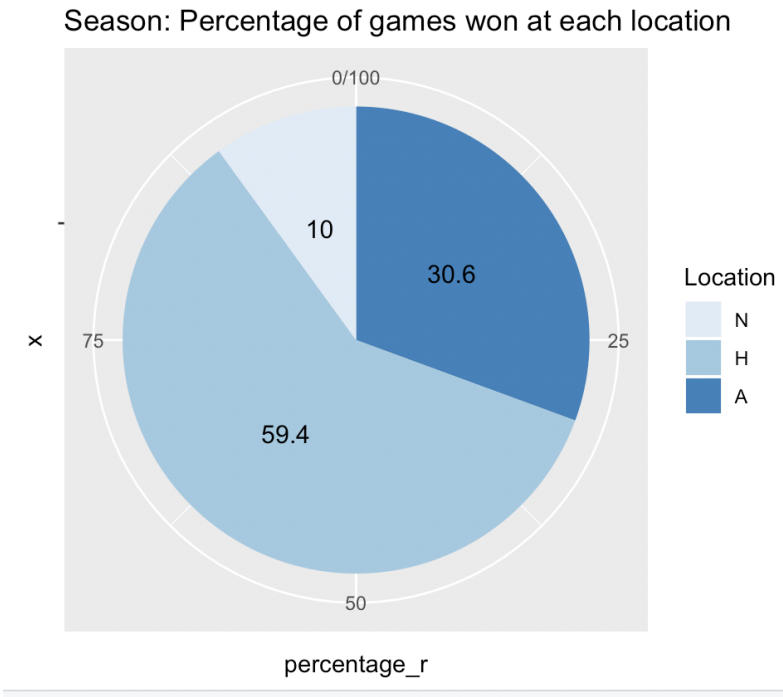
Data Processing

'RegularSeasonDetailedResults.csv' contains the game statistics of each game in each regular season. Each row in this dataset contains data for both the winning and the losing team, so we need to construct a new table to extract the game statistics for each team in each game in each regular season. Taking the mean of all the game statistics, we get the annual summary statistics of each team in each regular season.

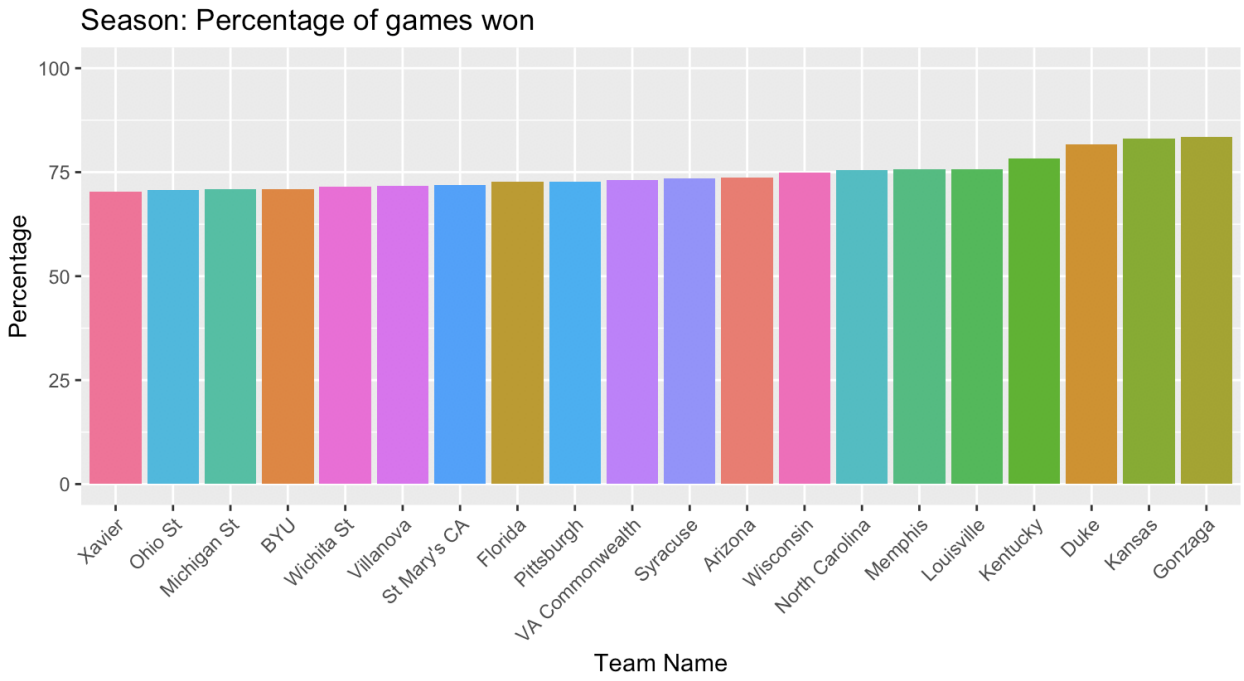
To construct our training set (data from 2003-2013), we divide the situation into two parts: the first part being having the winning team of each tournament game as Team 1, and losing team as Team 2, filled with corresponding team annual statistics; and the second part being having the losing team of each tournament game as Team 1, and winning team as Team 2, filled with corresponding team annual statistics. We then add in our response variable 'Win' with value 1 for every game where Team 1 wins, and value 0 for every game where Team 1 loses. We repeat the same process to construct our test set, but instead using the data from 2014-2016.

Data Exploration

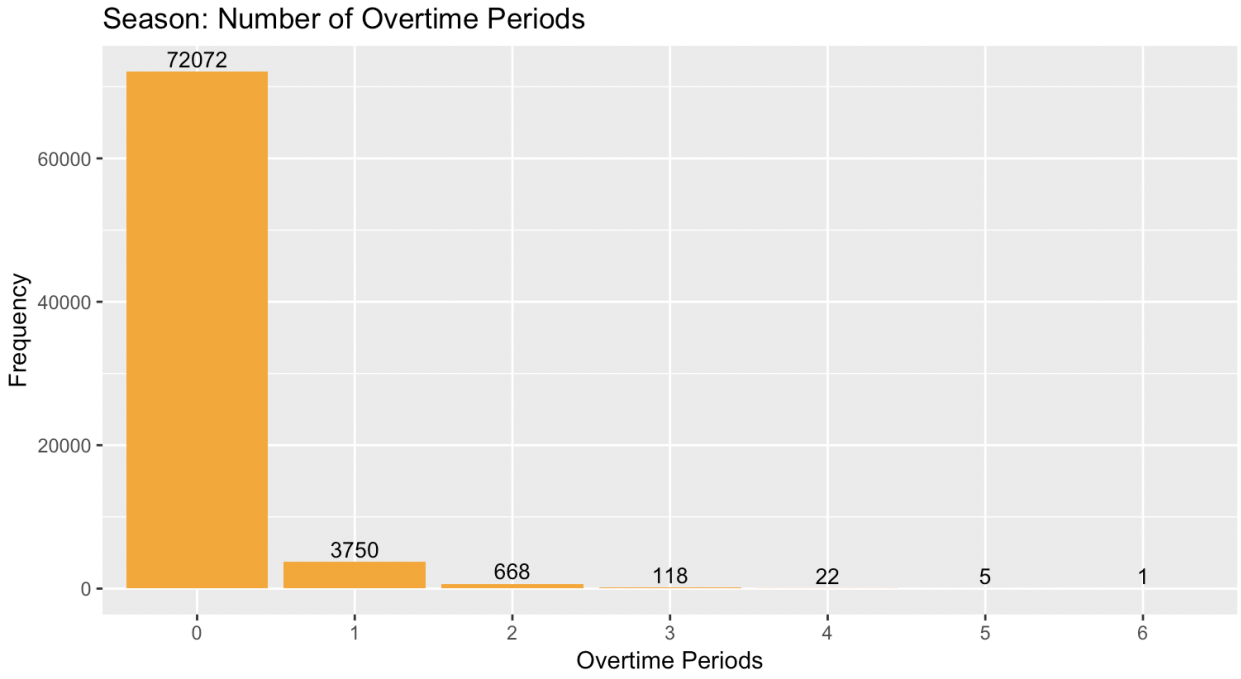
The pie chart below shows that 60 % of all wins occurred in the winning team's home stadium and 30.6% of winning teams was the visiting team.



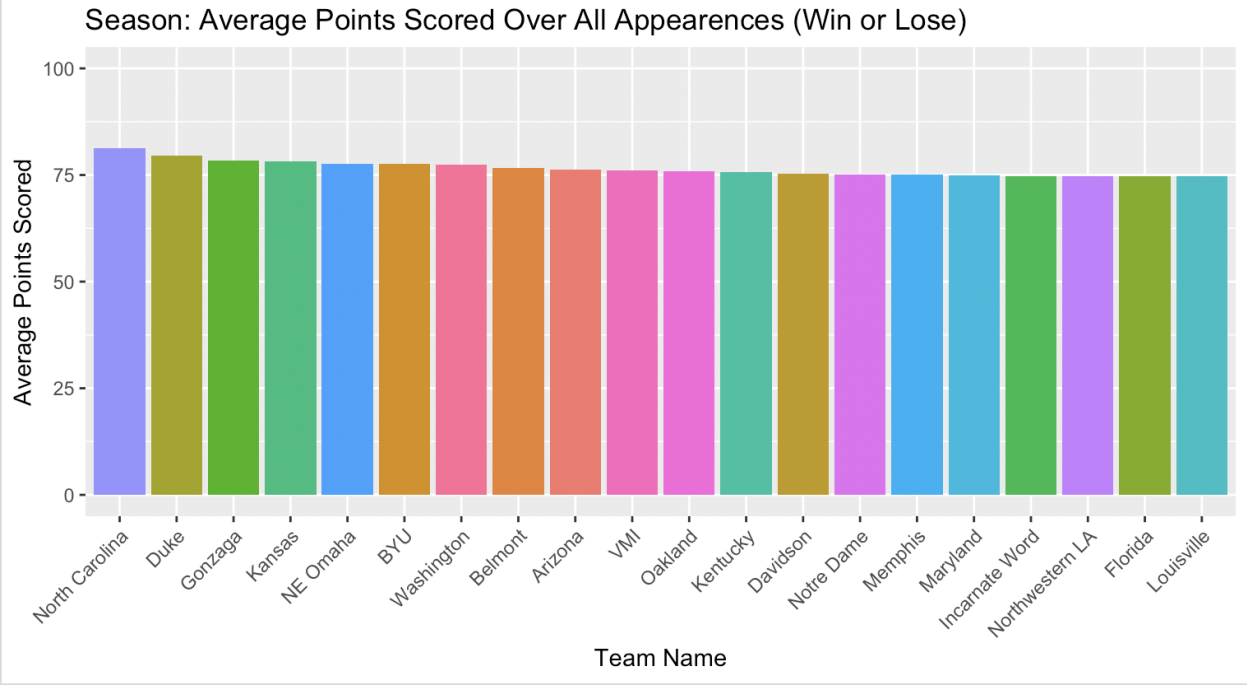
The plot below shows the top 20 teams with the highest percentage of games won. As we can see the Gonzaga University has the highest regular winning rate between 2003-2016.



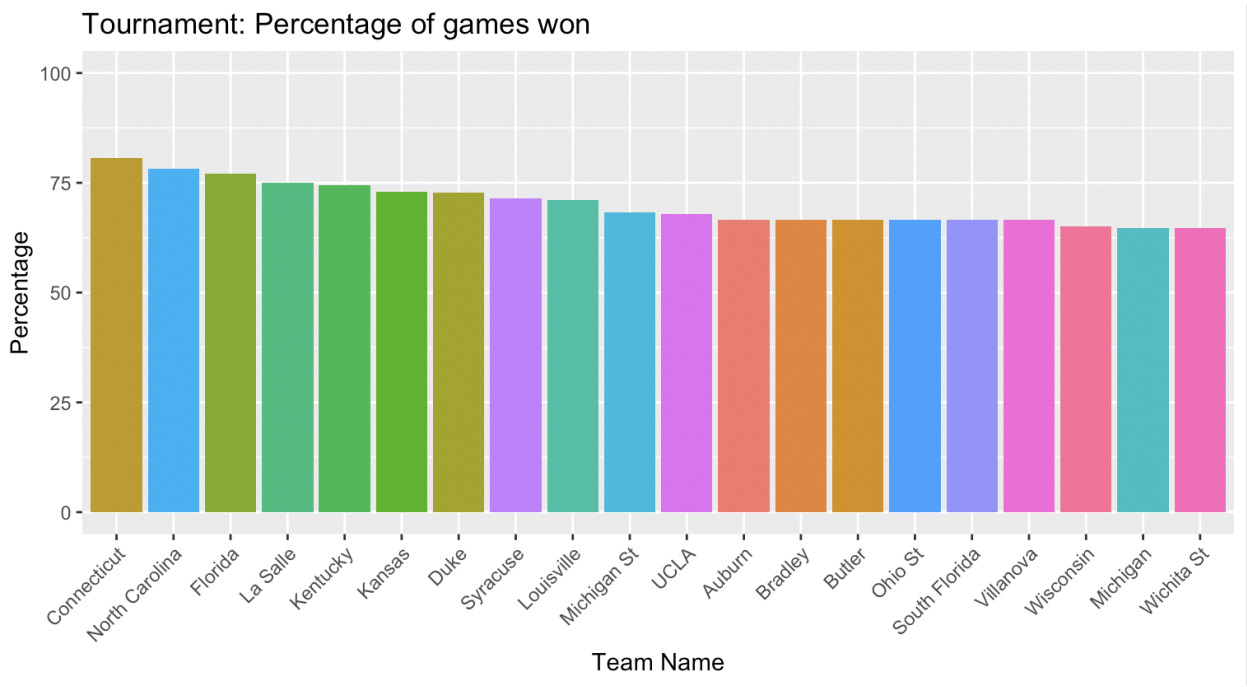
The plot below of the number of overtime periods clearly indicates that most games did not go overtime and were completed in 4 quarters.



The plot below shows top 20 teams with the highest average points scored per game. As compared to the plot of the percentage of game won, we find that the teams with high winning rates usually score more points per game.



We then explore the tournament data set, and the plot below shows the top 20 teams with the highest winning rates in the tournament. Most teams are similar to the teams in the regular season plot.



Data Analysis

To find the best parameters for each algorithm, we use 10-fold cross validation, repeated three times. We choose our final model for each method based on the accuracy of predictions on the test set.

Logistic Regression

The full model using all predictor variables returns an AIC of 1715.7 and accuracy of 0.6965 on the test set. Removing any variables results in lower accuracy on the test set. Thus, we will use the full model for our final predictions

Decision Tree

We perform decision tree method with two different stopping criteria. First, we use information gain criteria. By using repeated 10-fold cross validation, we find the entropy that returns the best accuracy is 0.0049, which gives an accuracy 0.6021 on the training set, and 0.6368 on the test set. The most important feature is T1WinPct, followed by T2WinPct, T2PF, T2Fgm, etc.

We then use gini criteria, and again with repeated 10-fold cross validation. The gini impurity that returns the highest accuracy is 0.0042, which gives an accuracy value of 0.6047 on the training set, and 0.6194 on the test set. The most important feature is T1WinPct, followed by T2WinPct, T2PF and T2Fgm, which agrees with the result obtained from information gain criteria. Since information gain criteria returns a higher accuracy for the predictions on the test set, we will use this model for our final decision tree predictions.

Random Forest

We iterate through a number of values for the number of variables used to construct one tree (mtry), again using repeated 10-fold cross validation, to find the range of the best mtry that returns the highest prediction accuracy. We then manually test on the values surrounding the best mtry value returned after the iterations to find that mtry = 5 returns the highest accuracy of

0.6318 on the test set. The most important features based on random forest are T1WinPct and T2WinPct, which again agrees with the previous decision tree results.

Support Vector Machines (SVMs)

We performed SVM with three differents kernels (linear, polynomial and radial). We iterate through cost values of 0.25, 0.5, 1, 5, 10, 15 and 20 to find the best cost value that returns the highest accuracy using repeated 10-fold cross validation. The result shows that cost of 0.5 gives the highest accuracy of value 0.6627 on the training set, and 0.6841 on the test set.
For SVM with polynomial kernel, we iterate through a number of values of cost, scale and degree to find the combination that returns the highest accuracy using repeated 10-fold cross validation. The output shows that cost of 0.25, scale of 0.1 and degree of 1 returns the highest accuracy of 0.6523 on the training set, and 0.6667 on the test set.

We then iterate through cost values of 0.25, 0.5, 1, 5, 10 and 15 and sigma of 0.005 to find the best cost value for SVM with radial kernel using repeated 10-fold cross validation. The output suggest that cost of 1 and gamma of 0.005 returns the highest accuracy of 0.6580 on the training set, and 0.6567 on the test set.

Results

The table below shows the scores of predictions by each method calculated using the scoring system specified in the metric section: one point for Round 1, two points for Round 2, four points for Round 3, and double the points for each round afterwards. The results suggest that SVM with polynomial kernel gives the best result in 2014, random forest gives the best result in 2015, followed by logistic regression, and SVM with linear kernel and logistic regression are equally good in 2016.

Overall, we can conclude that logistic regression is the best model with the highest average score of 82 on the predictions of the test set. This result rejects our null hypothesis that

support vector machine would return the best result compared to logistic regression, decision tree and random forest for the analysis of March Madness basketball data. In the alternative situation which is to predict the teams in the semifinals, our models are likely to have better performances and improved accuracy due to an decrease in the number of games predicted.

Year	**logisReg**	**decTree**	**RF**	**svmLinear**	**svmPoly**	**svmRadial**
2014	44	38	50	44	52	47
2015	107	62	126	104	94	86
2016	95	45	67	95	89	80
Average	82	48.3	81	81	78.3	71

Reference

Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani. (2013). An introduction to statistical learning : with applications in R. New York :Springer.

Katz, Josh. (2015). Here's How Our N.C.A.A. Bracket Works. [online] The New York Times. Available at: https://www.nytimes.com/2015/03/16/upshot/heres-how-our-ncaa-bracket-works.html [Accessed 10 Dec. 2018]

March Machine Learning Mania 2016. [online] Kaggle. Available at: https://www.kaggle.com/c/march-machine-learning-mania-2016#evaluation [Accessed 19 Nov. 2018].

Ogus, S. (2016). *The Economic Impact Of March Madness From First Four To Final Four*. [online] Forbes. Available at: https://www.forbes.com/sites/simonogus/2016/03/17/the-economic-impact-of-the-ncaa- basketball-tournament-from-first-four-to-final-four/#5669a0e61b56 [Accessed 19 Nov. 2018].

Wright, Mason, and Jenna Wiens. (2015) "Method to Their March Madness: Insights from Mining a Novel Large-Scale Dataset of Pool Brackets."