MARCH MADNESS PREDICTION USING MACHINE LEARNING

A PREPRINT

Soukarya Ghosh
Department of Computer Science
University of Virginia
Charlottesville, VA 22903
sg4fz@virginia.edu

Alex Wassel
Department of Computer Science
University of Virginia
Charlottesville, VA 22903
aw7re@virginia.edu

James W. Yun
Department of Computer Science
University of Virginia
Charlottesville, VA 22903
jy2gm@virginia.edu

April 30, 2019

ABSTRACT

Every year, the men's National Collegiate Athletic Association (NCAA) Division I basketball season concludes with a single elimination, 68-team tournament to determine the national championship, commonly refered to as "March Madness". Played mostly during March, it has become one of the most famous annual sporting events in the United States. In 2018, the Virginia Cavaliers basketball team, representing the University of Virginia (UVA), entered the tournament as the No. 1 seed overall but much to the dismay of its students and the country, suffered a historic upset in the first round to UMBC and became the first No. 1 seed to lose to a No. 16 seed in the era of the new tournament format established in 1985. It is proposed that the UVA's performance in the 2019 tournament can be predicted using a variety of machine learning regression algorithms on NCAA's historical basketball data. The findings of this project will answer whether UVA students should expect to win the "Big Dance" for the first time in program history.

1 Introduction

UVA was under the heat of the national spotlight after last year's loss to UMBC. This season, UVA (currently No. 2 at the time of writing) is on track to make a strong showing at this year's tournament, under three-time Henry Iba Award winner for national coach of the year, Tony Bennett. There have been millions of attempts to predict the outcomes in the tournament, but there has not been a single confirmed perfect bracket for all of recorded history. In fact, the odds of correctly predicting all of the 63 games in the NCAA tournament is one in 2^{63} or $9,223,372,036,854,775,808$ [1], an astronomically low probability. The strength of the model can be assessed by comparing the prediction model against the predictions of basketball analysts, as well as the outcome of the actual tournament. The results of this project can have implications in basketball analysis, sport news coverage, and the billion-dollar sport betting industry. The setting being considered throughout this project include individual team performance, player performance, tournament seeds, and analyst rankings.

March Madness is a prime target for machine learning enthusiasts, with countless tutorials for beginners appearing on a simple Google search. Many of the methodologies used in these prediction models use Random Forest Regressors, Decisions Trees, and Convolutional Neural Networks.

Examining previous related works, one particularly impressive project originates from a pair of students from Brigham Young University [2]. The team scraped their data from ESPN and cleaned for a total of ninety-four features. Conclusively, the results show that random ordering used in conjunction with k-nearest neighbor, Manhattan distance,

and random forest as the feature reduction algorithm provide the best classification accuracy of 0.7362. The group advised those who pursue this project in the future to consider individual player statistics and injury reports, win/loss record of the past ten games, and a few more features that could potentially affect the result of an individual match.

This project is inspired by UVA's current success in the regular season as well as its contributors' affinity for the team.

2 Method

Machine Learning techniques are great to use for this project because it is possible to gain insights from large amount of data. Since this is a complex problem for which there are no good traditional hand-tuned solutions, a machine learning approach would be optimal in converting a large dataset into meaningful estimations of a target feature.

Regression, a form of supervised learning, was used to make predictions was used for this project.

Throughout our experiments, a few distinct machine learning techniques were used, including some that have failed. At first, a custom regression model was created by using three dimensional analysis techniques and graphing tools, in order to minimize the log loss score and determine whether team 1 will beat team 2. We tried to do our own because we thought a unique and specific model to our data would be better than using the already-provided regression models. A satisfactory error rate was not achieved, therefore we introduced mathematically derived hyperparameters into our equation. Running this new model against a validation set resulted in even higher error rates.

We then decided to use a larger, more comprehensive dataset. In addition, we used play-by-play stats to determine a player rating for every NCAA men's basketball player. We then included the top five players per team as additional features for our dataset. Using this enhanced dataset, we were able to perform a variety of machine learning regressions, including linear SVM, linear regression, multivariate regression, and logistic regression on the dataset.

Another machine learning technique we performed is called attribute combination, for which we divided field goals made by field goals attempted, giving us the percent of shots that have been made into the basket.

We then compared our models accuracy on their respective validation set by comparing the log loss error of our matchup predictions. We identified that the linear SVM model produced the smallest log loss error of 0.55629. Since this model had the smallest log loss error, we selected it for our final model.

3 Experiments

The datasets being used for this project may be found at the following website:

`https://www.kaggle.com/c/mens-machine-learning-competition-2019/data`

This collection is a compilation of 6 distinct, but related datasets:

1. **Dataset 1** contains team ID's, team names, tournament seeds, final scores of all regular season conference tournament games, NCAA tournament games since the 1984-85 season, and season-level details.
2. **Dataset 2** holds game-by-game stats at a team level (free throws attempted, defensive rebounds, turnovers, etc.) since the 2002-03 season.
3. **Dataset 3** provides city locations of all regular season, conference tournament, and NCAA tournament games since the 2009-10 season.
4. **Dataset 4** contains weekly team rankings for dozens of top rating systems - Pomeroy, Sagarin, RPI, ESPN, etc., since the 2002-2003 season.
5. **Dataset 5** contains play-by-play event logs for 99% of regular season, conference tournament, and NCAA tournament games since the 2009-10 season - including plays by individual players.
6. **Dataset 6** contains additional supporting information, including coaches, conference affiliations, alternative team name spellings, bracket structure, game results for NIT and other postseason tournaments.

We carried out our project in the steps listed below:

1. Discovery - First, in order to gather a preliminary understanding of the data, a assessment of feature correlations as well as scatterplot visualizations of the dataset will be conducted.
2. Data preparation - As some of our features contain categorical data, categorical inputs will need to be encoded into numerical values. These features will also be scaled.

3. Model selection - A few machine learning algorithms we may use are multivariate regression, logistic regression, gradient decent, and support vector machine (SVM), random forest regressor, and decision tree regressor.

4. Fine tuning - Various hyperparameters associated with the selected model will be fine-tuned to achieve regularization and avoid overfitting. We will conduct a grid search to find the optimal hyperparameters

5. Launch and monitoring - The machine learning algorithm will be evaluated on the test set by following the college basketball games happening before the NCAA tournament. The model's correctness during this valuation set will be used to further to the model ahead of the actual tournament. Performance measures such as RMSE (L2 Norm) and log loss will also be used to measure the distance between the predicted values and targeted values.

First, a custom regression model experiment was conducted by calculating and visualizing a unique and specific regression model that would fit the data and minimize the log loss score. This equation was: Pr(Primary Win) = $\frac{1}{\eta}(y^{\alpha} - x^{\beta} + 16)$, where $\eta \geq 32$ and $\beta \leq 1$, where x and y are the ranks of the teams and η, β, α are the hyperparameters. This was done by running many trials of changing variables and numbers until reaching the least log loss score. However, this method did not turn out to work as well as expected, because it assumed the distribution was linear. Thus, it was decided to turn to other built-in models such as the SVM model.

Individual weights were assigned that describe the significance of each feature in a basketball game such as steals, offensive rebounds, defensive rebounds, dead rebounds, assists, blocks, personal fouls, technical fouls, turnovers, free throws, dunks, layups, tips, and jump throws. These features were calculated using a Python program to get individual player ratings out of 100, outputting the result into a separate CSV file.

The individual team stats and player stats were aggregated for training against the test set. This was done by training on regular season games and testing on that same year's NCAA tournament.

Using this enhanced dataset, we ran many different models and examined each one's validation set error rates. We first ran a linear regression model, which resulted in a 2.7594 log loss error. We then tried a random forest regressor which improved to a 1.6424 log loss error. Finally, we found our best model, which was a linear support vector machine (SVM) with a log loss error of 0.55629. Since this machine learning model resulted in the smallest error, we decided that linear SVM was the best model for this project.

4 Results

We acquired the predictions under the guidelines of the Kaggle Data Challenge, with a percentage prediction for every possible game in the tournament. Since there are sixty-four teams in the tournament, the total matchup predictions we made were 2,278. This accounted for all possible permutations of the tournament. The interpretation of this format lends itself to the log-loss format of error evaluation. Whereas, if the prediction percentage is below 0.5, then the $away$ team is predicted to be winning by a margin of $0.5 - \hat{y}$. Likewise, the $home$ team is predicted to win if the percentage is above 0.5, then the $home$ team is predicted to win by a margin of $\hat{y} - 0.5$.

Under these conditions, we filled out the 2019 NCAA March Madness Tournament bracket in order as suggested by our model. For the round of 64, we predicted 25 out of the 32 games correctly predicted, including some of the upsets that experts did not see coming. One of these is the $13th$ seeded UC Irvine beating the $4th$ seeded Kansas State. For the round of 32, we predicted 11 out of the 16 correctly. Although none were direct upsets, we did have a huge player in Auburn moving onto the Sweet Sixteen. The following round, we predicted 5 out of 8 games correctly, with an unforeseen Auburn beating a favorite UNC. This was perhaps our strongest correctly predicted round, with most of the Elite Eight games correctly formed.

As for the Elite Eight, we only got 2 out of 4 game correct, with out ultimate champion, Duke, getting knocked out by Michigan State. This was the biggest blow to our bracket, since Duke was the highest rated team in the predictions made by the model. Going into the Final Four, we had UVA beating Auburn, which mirrored real life. However, the other side of the bracket completely fell apart for us at this point. For the championship game, we had predicted that Duke and UVA would face off with Duke ultimately being crowned champion. However, in reality this did not happen and this is where the biases of the model started to fall apart.

Overall we predicted 43 out of 63 games correctly. This is above random chance, thus the biases of the model were ultimately on the right path. However, more cleaning and tuning needs to be done to our model in order to get it to a higher level.

5 Conclusion

As UVA students, this project was relevant to the well being of Virginia, as it was regarding an event that brought pride and joy to its residents, especially to the school, increasing team spirit. Although our model ended up ranking our bracket in the 90th percentile, it did not predict the winner of the tournament accordingly. However, our bracket predicted two out of four of the teams that reached the Final Four (UVA and Auburn). Sadly, the odds of 2^{63} to predict the outcomes of the tournament were not in our favor; we were not the first people in history to have had a perfect bracket, and thus just part of the millions of attempts to do so.

In the future, we are planning on conducting this project annually for the March Madness tournament, hoping to win the bracket challenge. As we now know how to go about these methods and have experience, we will conduct more and deeper experiments with other models, looking into neural networks as well. We will also look into way to update the model after each round in order to predict the next round. Additionally, we will try to do some data scrapping to also get data from before 2010 as well. We also want to have a system where we weigh each round more, as more significant games occur, by minimizing the error more in the championship game at the expense of less significant games such as in the round of 64.

The billion dollar betting industry could also benefit from using our model. By being able to accurately predict the outcomes of games, bettors and bookkeepers can use this information to create better odds or to game the current odds. This model can not only be applied to NCAA Basketball but also other sport like football, soccer, baseball, poker tournaments, etc. both at the college and professional levels. Our model's results will be able to catch a few upsets that even professional analysts overlook. Hopefully, losses such as the one to UMBC in 2018 will come less unexpectedly.

Overall, we found that predicting the outcomes of the NCAA Basketball tournament is nearly impossible and very hard to achieve a perfect bracket. However, with the power of Machine Learning, we can come very close to that and do better than random guesses as well as analysts.

6 Member Contribution

We started with a large collection of CSV files containing a range of features and datapoints. From these, we split up the data according to games, teams, and seasons and started cleaning. James and Soukarya worked on aggregating this data to form on dataframe, while Alex worked on framing the data and getting rid of fields which were not needed, as well as combining similar ones.

In addition we brain-stormed how to combine a few elements and account for players by team. We came up with a model that allows us to rate players based on season performance where Alex helped adjust feature weights, Soukarya created the algorithm for the ratings and James worked on integrating the player data with the team data. We spoke with Professor Nguyen for assistance and we all contributed equally towards the checkpoint report.

References

[1] Daniel Wilco. The absurd odds of a perfect NCAA tournament bracket put into perspective, `https://www.ncaa.com/news/basketball-men/article/2019-02-13/absurd-odds-perfect-ncaa-tournament-bracket-put-perspective`

[2] Jared Forsyth and Andrew Wilde. A Machine Learning Approach to March Madness. In 2014, IEEE, 2014. `http://axon.cs.byu.edu/ martinez/classes/478/stuff/Sample_Group_Project3.pdf`