A comparative analysis of machine learning algorithms in predicting 2021 NCAAM March Madness outcomes

Mehmet Ege OGUZMAN
StudentId: 20257961
Data Mining & Machine Learning 1, MSc in Data Analytics
National College of Ireland Dublin, IRELAND
Email: x20257961@student.ncirl.ie. URL: www.ncirl.ie

Abstract—**Predicting college basketball results is both challenging and an interesting problem to solve. Organized college leagues in the Unites States are receiving incredible attention. These leagues have also become a field for sports analysts, data scientists and even academics to work on. Teams, players and even coaches are compared according to season statistics and predictions are made about which teams is better for that season. In this study, the match statistics of the NCAA Men's Basketball League between 2003 and 2020 were collected and analyzed. The main purpose is to predict which of the two teams will win the match in 2021 March Madness which is the most important tournament of the NCAA Basketball league. This means that the binary classification problem must be solved. These collected data were trained with machine learning algorithms (namely xgboost, logistic regression, kNN, random forest, decision tree) and predictions were produced. Xgboost and logistic regression yield similar results, with an accuracy of over 92%.**

I. Introduction

The NCAA Basketball league is one the most watched leagues in the United States. This league, which is followed by millions of people, is also an important source of finance. It is a comprehensive league where many investments, advertisements and organizations are made. With millions of fans, college basketball teams are also source of prestige for universities.

The NCAA Basketball league is a very important league not only for viewers and fans, but also for players and coaches. Many star basketball players, such as Michael Jordan, first competed in this league in order to be selected by the NBA teams, and then they drafted by the NBA teams. In NCAA, which is followed closely by talent scouts and NBA team, the performance and statistics of young players are significant to be drafted by NBA teams.

The field of sports analytic has become quite common with the popularization of the concept of data science. When we look at investments and circulating capital in the field of sports,teams have also become businesses. A lot of commercial, player or team-based data is produced about the teams and this data plays an important role in determining the strategies of the teams. Analyzing and processing this data has become a major contributor to teams winning matches, becoming champions in their own leagues, and even determining their financial strategies.

In this report, college men's basketball teams will be analyzed for the NCAA (National Collegiate Athletic Association) league, which is very popular in the United States. By analyzing the match statistics and general conditions of the teams in the season and tournaments, and using the right machine learning algorithms, the champion of the March Madness tournament, the biggest tournament of the league, will be tried to be predicted.

March Madness is the iconic final tournament of the NCAA Division I Men's Basketball league. It is a single-elimination tournament where the top 68 colleges from each state of the United States compete for the national championship. Millions of dollars are spent on March Madness. Therefore, predicting the winners and even the champion of this tournament is both a fun and a difficult problem.

As in many sports, a lot of data is recorded in basketball matches. This data is used to predict match results with different statistics and models. Although it is a very challenging problem, models with high accuracy have been created in the past studies. For such sports analytic problems, feature engineering is the key point of the solution. Which features and which feature combinations of variables to use are of great importance for models to find the sweet spot.

For the study, datasets provided by Kenneth Massey will be used for the March Machine Learning Mania 2021 - NCAAM competition, which is published every year on the Kaggle platform in cooperation with GCP (Google Cloud Platform).On these datasets, feature engineering and some basketball statistics used in the literature will be used and binary classification models will be created. Feature engineering techniques will be applied on these datasets and after creating binary classification models using some basketball statistics used in the literature, match predictions will be tried to be made for the last 68 teams remaining in the March Madness tournament. In the literature, in order to predict the March Madness champion, a simulation is created and the matches are simulated thousands of times after each match combination of the 68 teams and the probability of the winning team in these match-ups are calculated. However, in this study, simulation will not be made and winners and losers will be determined according to the probability values found for each match and the champion will be tried to be predicted.

The purpose of the study is to analyze different types of data sets such as season and tournament matches played in the past, team statistics and to make a probability prediction about which teams will win in the tournament in the future. These winning probability predictions will allows to predict the final bracket of the tournament. In addition, since winning and losing between teams is a binary classification problem, different types of classification algorithms will be trained with historical data and the results will be examined.

II. Literature Review

Contributions to the literature for the outcome predictions of March Madness matches were collected in different categories. Different kinds of problems arising from these categories have been tried to be solved.

A. Seed Difference and Elo Ratings

March Madness is divided into four districts by state of the United States, and each team in the tournament is awarded a seed, which is an estimate of the standings of a team in their district of the group determined by the selection committee before the tournament. These seeds determine the ranking of the teams. 1 to 16 seeding predictions for each region are made for 16 teams, and the team with 1 seed is the strongest team in the region, while the team with 16 seeds is the weakest team in the region [1].

These seeds can be good predictive variables for finding the winner of the match. Sean McCrea examined whether the seed difference between teams was a good variable in determining the winner. When the matches between 1985-2005 are examined, it has been revealed that the seed gap is a good predictor for the first rounds [2].

The Elo ratings system provides a ranking according to the strength of the teams and is calculated with match-by-match statistics. Elo ratings can be calculated by different methods such as ordinal logistic regression (OLR), Sagarin's Computer method [3].

Elo ratings, like the seeding system, have been included in many projects in the literature as a variable used when comparing two teams, as it provides information about the strength of the teams. FiveThirtyEight has been making elo rating calculations for many leagues such as the NBA, NFL and NCAA since 1950. While making these calculations, match statistics are used and they are used to measure the performances of the teams during the season [5].

Seeding and Elo ranking systems are among the most valuable variables that can be used to compare teams. It is frequently encountered in the literature to train logistic regression models with these variables [6]. In addition, many ranking systems and statistics such as Pomeroy, ESPN and Massey are also mentioned in the literature [8].

B. Feature Engineering

The most used approach for March Madness match predictions is the correct selection of features and data variables. There are many new statistics that can be calculated with match statistics.

Since the match to be predicted has not been played yet and there are no match statistics, the averages of the previous season statistics of the teams are used in most of the studies [7]. But which statistics to use is also of great importance. For example, there are more than 50 features for each match and these features need to be extracted or used to create new features.

Another data engineering approach seen in the literature is to create the data set from statistical differences [9]. Based on the past match statistics, a data set consisting of the differences of the statistics between the teams was prepared and a model training was tried. In this approach, the team with the larger expectation statistic, that is, the team with a positive statistical difference, won.

Another common approach seen in the literature is to analyze day-by-day match statistics [10]. When the in-match statistics are examined, four elements are very important. Effective field goal percentage, turnovers, offensive rebound and free throw rate. Also offensive and defensive efficiency are significant features. These features are calculated according to possessions. Then, a model was trained by taking the average of the teams.

C. Model Selection

Machine learning models will be trained to solve this binary classification problem. The most used models in the literature are logistic regression [13], xgboost and kNN [11].

In addition, the least squares pair wise comparison model in the literature has also added a different approach [12]. He created a two-stage model using both linear and logistic regression and a successful result was achieved.

In addition, it is mentioned in the literature that the random forest and decision tree machine learning algorithms also gives a successful result [14].

III. Datasets

Data sets have been published for the March Machine Learning Mania 2021 - NCAAM competition on Kaggle [15]. This Data set consists of many csv files. From these structured data sets, the data of the matches of the teams in the past season and the matches of the past March Madness tournaments will be used. Dataset descriptions can be seen in Table I.

Historical March Madness (Tournament) and season match statistics (Season) are the two main sets of dataset that will be used. Kaggle also provides the seed values of the teams for each March Madness tournament between 1985 and 2019 (Tournament Seeds). The Teams dataset gives the ids of the teams and the names of the teams corresponding to these ids. This data set will provide a more readable and understandable table for matching teams and creating brackets after the prediction phase.

Elo Ranks, MMasseyOrdinals and Kenpom datasets are provided from sources other than Kaggle and provide information such as Elo rating, team ranks and seed values of the teams.

TABLE I
DATASET DESCRIPTIONS

Dataset Name	Number of Sample	Number of Column
Season Detailed (2003-2020)	92832	34
Tournament Detailed (2003-2019)	1115	34
Tournament Seed (1985-2019)	2284	3
Season Compact (2021)	166881	8
Tournament Compact (2021)	2252	8
ELO Ranks (2021)	356	4
Teams (2021)	372	4
MMasseyOrdinals (2003-2020)	4120887	5
Kenpom (2002-2021)	1265	26

A. Tournament and Regular Season Statistics Features

The features of these two data sets that give match statistics are as follows:

- Season - Season in year.
- WTeamID - Winner team ID.
- WScore - Winner team score
- LTeamID - Loser team ID.
- LScore - Loser team score.
- WLoc - Categorical binary value. If the winner team ”Home” or ”Away”.
- NumOT - Binary integer value. Whether the match went to overtime or not.
- FGM - field goals made
- FGA - field goals attempted
- FGM3 - three pointers made
- FGA3 - three pointers attempted
- FTM - free throws made
- FTA - free throws attempted
- OR - offensive rebounds
- DR - defensive rebounds
- Ast - assists
- TO - turnovers committed
- Stl - steals
- Blk - blocks
- PF - personal fouls committed

These feature are provided for both winner team and loser team separately. Compact results contain only the first 7 columns of the detailed results to see the clear results without complex statistics.

B. Tournement Seeds

Kaggle provides the first ranking parameter, ”seed” values for each team and each season. These values are given for the 4 sides of United States (East, West, South, Midwest). Tournement Seeds dataset features are as follow:

- Season - Season in year.
- Seed - Seed value.
- TeamID - Team ID.

C. Elo Ranks

Elo and Insemination values were provided by two different sports statisticians. Ken Pomeroy [16] and Kenneth Massey (MMasseyOrdinals) are two famous United State sports statisticians. They estimate and publish these values every year with the elo rating and seeding estimation systems they have established. Another team ranking system that provides data to this study is created by Warren Nolan [4].

The Elo system will be used ready-made and the existing ranking values will be used. A new Elo calculation will not be made for this study.

IV. Methodology

In this research, a popular data mining methodology called KDD (Knowledge Discovery in Database) is followed, which is a seven-step process: (1) understanding the domain and developing the goals for the study; (2) data selection and integration; (3) data cleaning and preprocessing; (4) data transformation; (5) data mining; (6) pattern evaluation and interpretation; and (7) knowledge discovery and using predictions. This well-known methodology provides us with a systematic and structured approach to completing this data mining study, increasing the possibility of receiving accurate and dependable results.

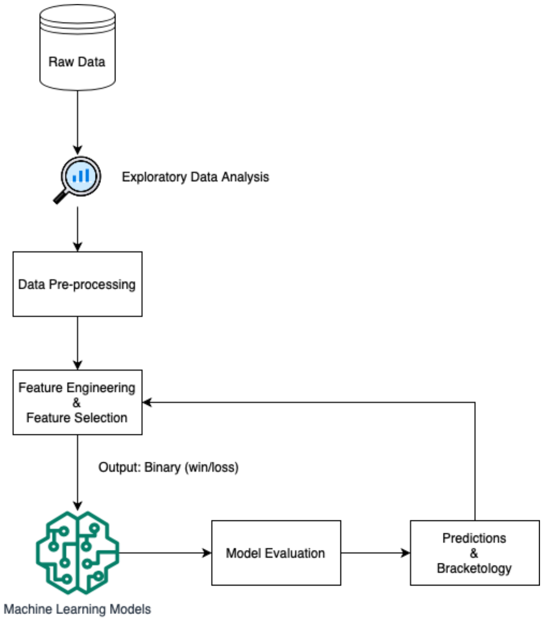
Raw Data
Exploratory Data Analysis
Data Pre-processing
Feature Engineering & Feature Selection
Output: Binary (win/loss)
Machine Learning Models
Model Evaluation
Predictions & Bracketology

Fig. 1. Project methodology.

A. Exploratory Data Analysis

Analyzes were made by extracting descriptive statistics from the each collected raw dataset. In particular, some key inferences were made on the regular season and tournament historical data.

Moreover, historical data of conferences and colleges are visualized and interpreted by analyzing compact regular season and tournament match statistics. Afterwards, college’s seeds and Elo ratings were examined and it was decided whether they could enter the training dataset.

1) Conference Comparison: According to the compact results, it has been examined which conference has won the most since 1985. The Atlantic Coast Conference have the most titles with 11, closely followed by the Big East Conference with 8 titles as it can be seen in Figure 2.

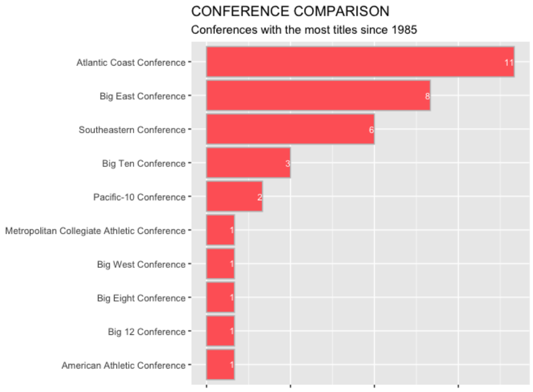

Fig. 2. Conference comparison graph.

2) College comparison: When it comes to championship games, the major programs undoubtedly fill out the top institutions, none of them are more than Duke. By far the most championship games have been played, with five crowns and four runners-up finishes. With four titles each, North Carolina and the Huskies round out the top three, with Michigan serving as the largest bridesmaid, as it can be seen in Figure 3.

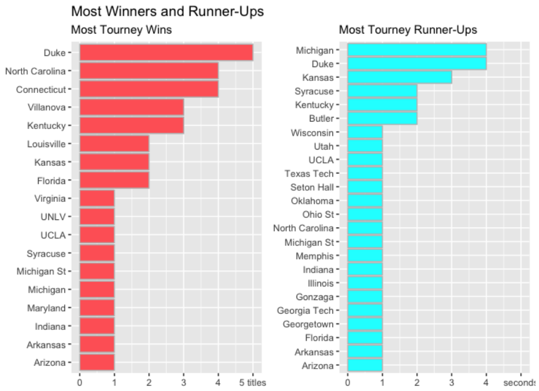

Fig. 3. College comparison graph.

3) Elo ranking significance: By examining the MMassey Ordinal data set, it will be decided whether the Elo ranks feature can be used as an estimator. Analyzes can be seen in Figure 4 and Figure 5.

Two system pivots were selected for review. The importance of this feature was measured according to Ken Pomeroy's and Jeff Sagarin's Elo ranking systems.

As can be shown, the higher Pomeroy rated team won 71.9 percent of the time in tournament games from 2003 to 2019. The 2008 season had the largest proportion of higher-ranked teams win, with 81 percent of games won, while 2014 was the poorest season for higher-ranked teams, with just 65 percent of games won. However, Sagarin's rankings were comparable, with an average performance of 71.7 percent, slightly behind Pomeroy's, as it can be seen in Figure 4.

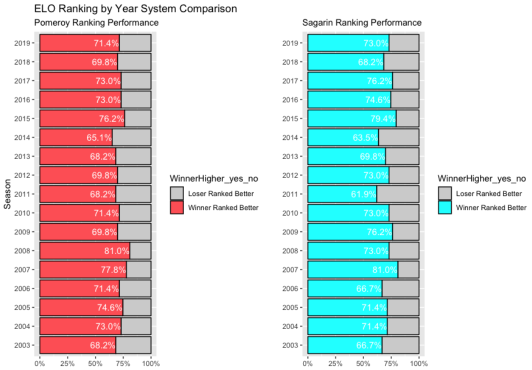

Fig. 4. Elo ranking comparison by year.

When the systems are analyzed on a round basis, Sagarin's ranking system looks better for the first 3 rounds, while Pomeroy's system looks better for the last 3 rounds, as it can be seen in Figure 5.

As the competition continues through the stages, it appears that better rated teams are winning fewer games. Interestingly, in the Elite 8 round, the higher-ranked Pomeroy's system only wins slightly more than half of the time, while higher-ranked Sagarin's system only win half of the time.

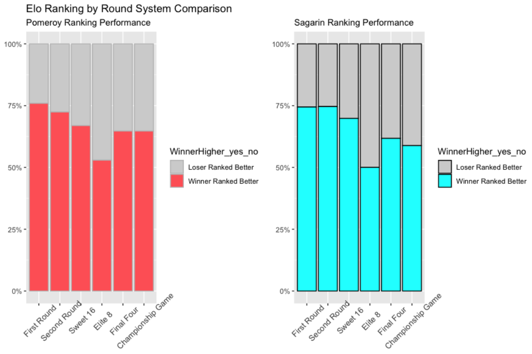

Fig. 5. Elo ranking comparison by rounds.

According to the analysis, Ken Pomeroy statistics will be used in the study since Pomeroy's ranking system guides better.

4) Seeds significance: Since 1985, the one seed has won 22 championships, making it by far the most common seed to win. The second seed has five victories, while the third seed has four victories. Surprisingly, the number 5 seed did not win a tournament throughout the time period studied, as it can bee seen in Figure 6.

Teams' seeding would have a great predictive effect on game outcomes, but can it be demonstrated how correct this notion has been in the past?

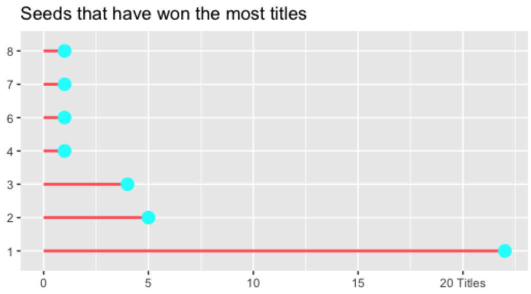

Fig. 6. Seed difference analysis.

When the Figure 7 is analyzed, the higher seed will win around 75% of the time in the first round, but just 70% of the time in the second round and 70% of the time in the Sweet 16. The higher seed's winning rate then drops significantly during the Elite 8 to less than 50%.

The frequency of games involving the same seeds is greater in the Final Four and Championship game. Even yet, the lower-ranked college will only win approximately 20% of the time in the last four, and only about 5-10 percent of championship games.

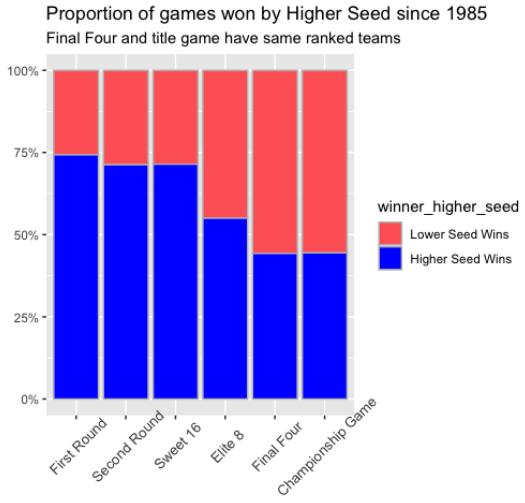

Fig. 7. Number of titles according to the seeds.

B. Feature Engineering

In this section, new features will be tried to be created by using the statistics in the data sets. The basketball statistics in the literature were used to give the model better high accuracy.

First, detailed match statistics datasets for the tournament and regular season were combined. Then, in order to add both the winner and loser label feature to the model, the statistics of the winning team and the losing team were changed in some rows in the combined dataset. In this way, only the training data set for the beaten team became both the defeated and the defeated (0,1) training dataset, and now the binary classification approach has become applicable.

Later, new features were added to the training dataset by applying transformations with the features of both the winning team and the losing team. The new statistics added are as follows:

1) Field Goal Percentage: It is the ratio of field goals made to field goals attempted.

$$FGP = FGM/FGA * 100$$

2) Possession: When a team is on offense, they have possession of the ball.

$$Possession = FGA + (FTA * 0.475) + TO - OR$$

0.475 value is declared according to the Ken Pomeroy's blog [17]. It is the constant value for NCAA while calculating the possession.

3) Effective Field Goal Percentage: It is a measurement of how successful your team is from the field.

$$Efgper = ((FGM - FGM3) + 1.5 * FGM3)/FGA$$

4) 3 Point Percentage: It is a measurement for average 3 points per game.

$$3Pper = FGM3/FGA3 * 100$$

5) Offensive Ratio: It is a measurement of offensive proficiency rating per game.

$$OffRatTeam1 = (T1Score/(T1Poss + T2Poss)) * 100$$

$$OffRatTeam2 = (T2Score/(T1Poss + T2Poss)) * 100$$

6) Defensive Ratio: It is a measurement of defensive proficiency rating per game.

$$DefRatTeam1 = (T2Score/(T1Poss + T2Poss)) * 100$$

$$DefRatTeam2 = (T1Score/(T1Poss + T2Poss)) * 100$$

7) Strength of Schedule: It represents a team's average schedule difficulty.

$$SOSTeam1 = T1OffRat/T2OffRat$$

$$SOSTeam2 = T2OffRat/T1OffRat$$

8) Turnover by Possession: It is an turnover ratio per possession.

$$TOPoss = TO/Poss$$

9) Offensive Rebound by Possession: It is an offensive rebound ratio per possession.

$$ORperTeam1 = T1OR/(T1OR + T2DR)$$

$$ORperTeam2 = T2OR/(T2OR + T1DR)$$

10) Free Throw Rate: It refers to a team's ability to make free throw attempts.

$$FTR = FTM/FTA * 100$$

Then, using Pomeroy's Elo ranking system, both teams' Elo rankings for that season were added to the dataset for each match. Moreover, the seeding estimates of the teams were drawn according to the same ranking system, and the seed differences between the two teams were added to the training set as a feature.

The final version of the dataset has 93947 samples and 59 features and is ready for the training phase.

C. Feature Selection

Feature selection is important for this problem. It is vital for the values with high correlations to come out of the 59 features and for the model to find the sweet spot.

Not all of them were trained because the newly produced traits that underwent Transformation were highly correlated with other traits. Also, when all 59 features are in training dataset, all models tend to overfit.

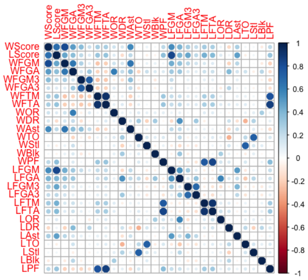

Fig. 8. Correlation matrix for final training dataset.

The correlation matrix for the last remaining features in the training dataset is shown in Figure 8. Although there is high correlation between some features, the correlation values of these features have been ignored.

D. Model Evaluation

After the data set is prepared, machine learning will be applied in this section. For the study, the five most used machine learning models in the literature (xgboost, logistic regression, kNN, decision tree and random forest) will be trained and the binary classification problem will be tried to be solved by outputting the probability values of the match results.

Test-train split approach has been applied on the main data that will come to the models. Moreover, k-fold cross validation technique is used for some models such as xgboost.

In addition to the data set prepared with basketball statistics, training will be conducted with only the Elo difference and seed difference features. The data sets that will enter the training are as follows.

Main Dataset:

- T1FGP & T2FGP
- T13Pper & T23Pper
- T1FTR & T2FTR
- T1OR & T2OR
- T1DR & T2DR
- T1TO & T2TO
- T1Stl & T2Stl
- T1Blk & T2Blk
- T1PF & T2PF
- T1Ast & T2Ast
- T1ELO & T2ELO
- T1WON

Seed Difference Only:

- SEEDDIFF (T1SEED - T2SEED)
- T1WON

Elo Rank Only:

- T1ELO & T2ELO
- T1WON

The models were trained and evaluated one by one with these data sets.

1) XGBoost: XGBoost, which stands for Extreme Gradient Boosting, is a regularized version of the Gradient Boosted Tree Algorithm. Boosting is an ensemble strategy that works by developing new models that repair faults generated by older models. All models are then compounded until no more improvements can be made. Gradient Boosting takes the strategy of developing new models that forecast the residuals of previous models and then combining them to generate a prediction. It minimizes its loss function using a gradient descent process, thus the name Gradient Boosting. XGBoost is a framework that extends gradient-boosted algorithms with system, model, and algorithm advancements. Hyperparameters that have been fine-tuned include:

- nfold: n fold for cross-validation.
- eta: Learning rate.
- max.depth: Max depth.
- min_child_weight: Minimum number of samples in node to split.
- gamma: Minimum loss reduction for split.
- subsample: Proportion of training data to use in tree.
- colsample_bytree: Number of variables to use in each tree.
- nrounds: Number of rounds.

5 models are trained with different hyperparameters. The learning rate parameter was the parameter that affected the model the most. Training was performed with 5 different learning rate parameters such as 0.05, 0.5, 0.1, 0.01 and 0.3. The results are shown in Figure 9.

Finally, it was decided that the learning rate value of 0.05 was the most optimal result. Model is trained 120 epoch with the parameters max_depth 7, gamma 0 and the early stopping round 20. Results of this model are shown in Figure 10.

The latest xgboost model scored well with an accuracy of 0.94. In addition, the recall and precision values of the model

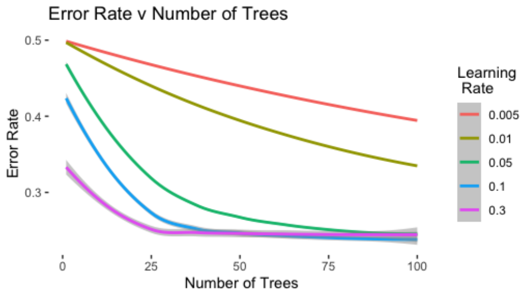

Fig. 9. Five xgboost model comparison by learning rate.

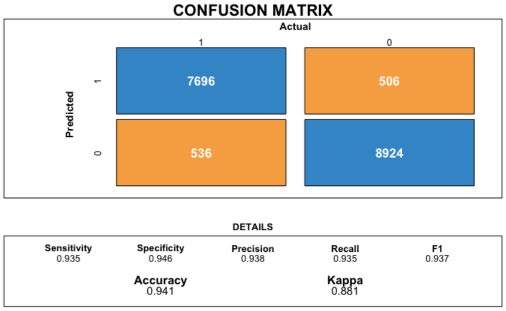

Fig. 10. Confusion matrix of final xgboost model.

are also high with a score of 0.93. The ROC curve of the model is shown in Figure 11.

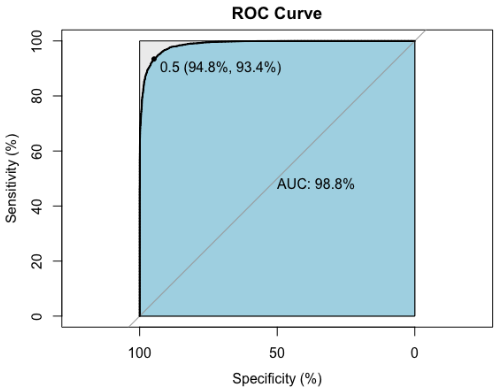

Fig. 11. ROC curve of final xgboost model.

2) Logistic Regression: The linear regression model is extended with a binary classification technique. The logit function is used to calculate the likelihood of a particular class.

The binomial regression model was applied for the main data set and only for the Elo rank and seed data set and the results were obtained.

The results for the Logistic regression master dataset are as in Figure 12. A threshold of 0.5 has been applied to probability values.

The logistic regression model for the main data set achieved an accuracy of 0.95. The ROC curve for Logistic regression is shown in Figure 13.

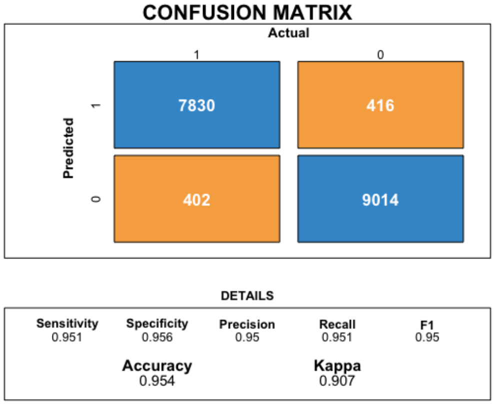

Fig. 12. Confusion matrix of logistic regression model.

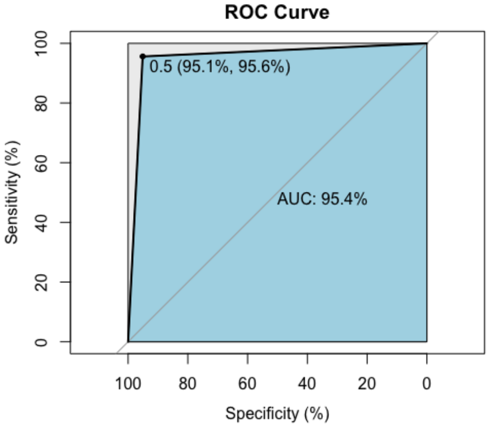

Fig. 13. ROC curve of logistic regression model.

It has been shown in the literature that teams are trained on logistic regression models with only seed and Elo differences.

The results for the bo models are not very encouraging. The accuracy rates of the models are shown in Table II.

TABLE II
LOGISTIC REGRESSION WITH SEED DIFFERENCE AND ELO RANKS

Model	Features	Accuracy	Precision	Recall
Logistic regression	Seed Difference	0.7	0.714	0.792
Logistic regression	Elo Ranks	0.689	0.676	0.634

3) K-Nearest Neighbour: The kNN model is a non-parametric prediction approach that works well with real-world data. Fundamentally, the kNN model uses the information in a particular observation to anticipate the result of the observation by computing which past observations most closely resemble the observation of key interest. The kNN model determines the ”euclidean distance” between the current observation and all prior observations. The model then averages the results of the k nearest neighbors to create a forecast for the observed value.

Two different k values were used for the study. According to the literature, the square root of the total number of samples or the total number of features is the appropriate k value. The same method was applied in this study. Results of kNN model that is trained with main dataset can be shown in Table III.

TABLE III
KNN RESULTS WITH TWO DIFFERENT K VALUES

Model	k	Accuracy	Precision	Recall
kNN	5	0.86	0.844	0.857
kNN	265	0.838	0.821	0.834

4) Decision Tree and Random Forest: This ”divide and conquer” strategy produces tree topologies in which leaves represent labels and branches represent feature combinations. At each stage, the algorithm selects the appropriate variable to partition the dataset based on the target’s values based on its discriminative ability [18]. The objective is to create a model that predicts the value of a target variable based on basic decision rules derived from data. Decision tree in Caret library has one parameter ”cp”. It specifies the complexity of the model. Iteratively more than one decision tree are trained and the impact of the ”cp” parameter on the model can be shown in Figure 14.

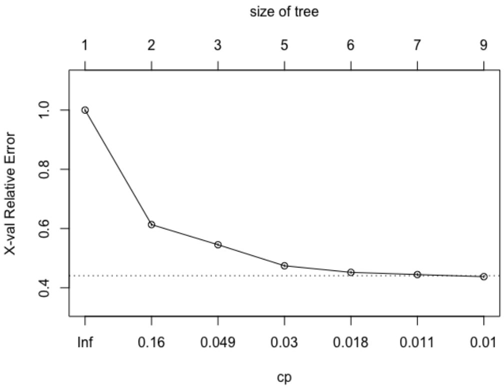

Fig. 14. Decision tree complexity comparison.

Random forest is a meta estimator that improves predicted accuracy by fitting alternative decision tree and average outputs [18]. Aside from accounting for unusually complicated decision boundaries, it is a fast-to-train strategy that reduces generalization error, has been shown not to overfit, and is computationally efficient. 2 parameter (ntree, mtry) passed to the random forest model where ”mtry” specifies the number of variables randomly sampled as candidates at each split and ”ntree” specifies the number of trees to grow. For this study, ”ntree” is 128 and ”mtry” is 5.

The impact of number of trees on model performance can be seen in Figure 15. The parameters are selected according to the sweet spot.

Decision tree and random forest results can be shown in Table IV. Random forest algorithm gave a much better accuracy score and eliminated the danger of overfit created by the decision tree.

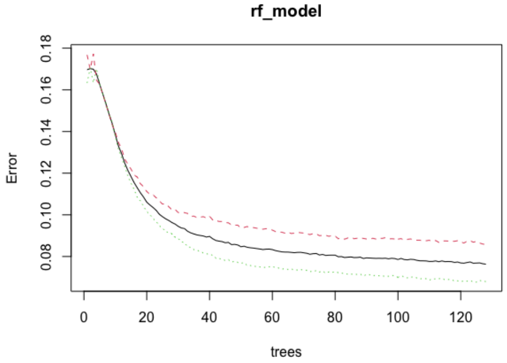

Fig. 15. Random forest number of trees comparison.

TABLE IV
KNN RESULTS WITH TWO DIFFERENT K VALUES

Model	Accuracy	Precision	Recall
Decision Tree	0.809	0.828	0.747
Random Forest	0.928	0.924	0.922

V. RESULTS

As a result, when the results of each trained model were examined, Table V was obtained.

TABLE V
RESULTS OF MODELS.

Model	Dataset	Accuracy	Precision	Recall
XGBoost regression	Main Dataset	0.941	0.938	0.935
Logistic regression	**Main Dataset**	**0.954**	**0.95**	**0.951**
Logistic regression	Seed Difference	0.7	0.714	0.792
Logistic regression	Elo Ranks	0.689	0.676	0.634
KNN (k=5)	Main Dataset	0.86	0.844	0.857
KNN (k=265)	Main Dataset	0.838	0.821	0.827
Decision Tree	Main Dataset	0.809	0.828	0.746
Random Forest	Main Dataset	0.928	0.924	0.922

When Table V is examined, it is seen that logistic regression gives the highest performance with an accuracy score of 0.95. In addition, xgboost and random forest algorithms followed logistic regression with their performances above 0.90, while other algorithms were far below the expected performance.

When the difference of the data sets is examined, it can be determined that the performance of the models using only the Elo differences and seed differences of the teams is low. However, when seed and elo differences are used together with other match statistics, very effective results have been achieved. As a result, the logistic regression model trained with the main data set (match statistics) was chosen as the estimator.

In order to make predictions for the 2021 March Madness tournament, the first 68 teams qualified to participate in the tournament in 2021 are taken as a basis. Which of these teams will be is specified in the seeding dataset. Then, every possible match within these 68 teams was determined and a total of 2278 team matches came out from the binary combination of 68. For each match, a previously saved logistic regression model was loaded into the system to get the probability values that determine which team will win, and predictions were obtained.

Instead of the matches made with the IDs of the teams, the names of the colleges were placed with the help of the data set in which the names of the teams were kept, and more legible results were obtained and recorded.

Finally, a bracket was created for the 2021 March Madness tournament according to the match result probabilities given by the model. The original results for 2021 appear in Figure 16 and the resulting bracket from the model appears in Figure 17.

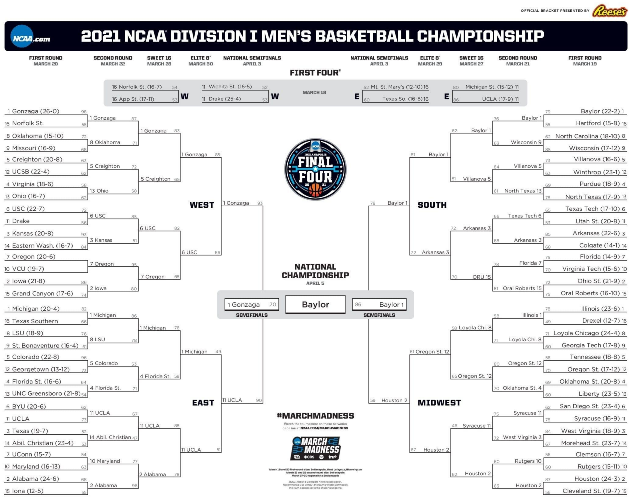

Fig. 16. 2021 NCAAM Basketball Division I. March Madness original bracket.

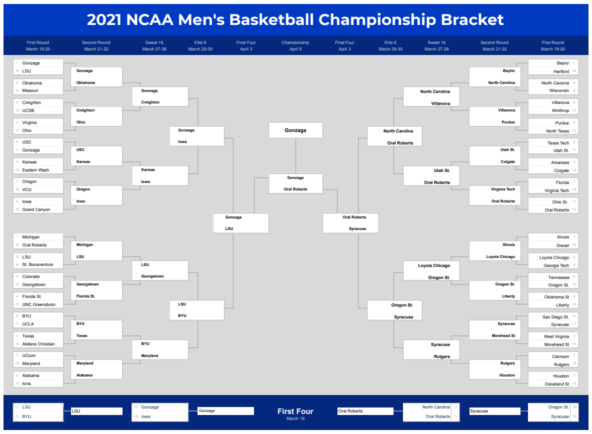

Fig. 17. 2021 NCAAM Basketball Division I. March Madness predicted bracket.

VI. Conclusion and Future Work

With the spread of sports analytics, it is explained how data scientists and sports industry can work together. Considering that the teams are also companies, it has been understood how much benefit can be provided by processing the data they bring there. It has been observed how the processing and conversion of these raw data into interpretable information can affect the performances of both athletes and teams.

In this study, it was tried to create the final bracket for the 2021 NCAAM Division I. Basketball March Madness tournament and machine learning models were compared. Although the problem is interesting, it is a difficult one.

The logistic regression model fed by basketball and match statistics gave the highest performance with an accuracy of 0.95 in the problem that was tried to be solved by using machine learning algorithms such as xgboost, kNN, decision tree and random forest. However, as a result of the study, it was concluded that no matter how effective the model selection is, the main key operation is feature engineering and feature selection.

It has been proven that more successful results can be achieved with the help of more detailed statistics and mathematics for further studies, examining each match and correct feature engineer techniques.

References

[1] Shouvik Dutta, Sheldon H. Jacobson and Jason J. Sauppe. Identifying NCAA tournament upsets using Balance Optimization Subset Selection. April 21, 2017.

[2] Jordan Gumm, Andrew Barrett, and Gongzhu Hu. A Machine Learning Strategy for Predicting March Madness Winners. Department of Computer Science, Central Michigan University. 2015.

[3] Brady T. West. A Simple and Flexible Rating Method for Predicting Success in the NCAA Basketball Tournament.The Berkeley Electronic Press. 2006.

[4] NCAA Men's Basketball 2021 ELO and Ranks. Accessed: 10 November, 2021. [Online]. Available: https://www.warrennolan.com/basketball/2021/elochess

[5] Jay Boice and Nate Silver. How Our March Madness Predictions Work, March 11, 2018. Accessed on: December 10, 2021.[Online]. Available: https://fivethirtyeight.com/features/how-our-march-madness-predictions-work/#anchor

[6] Jared Forsyth, Andrew Wilde. A Machine Learning Approach to March Madness. Brigham Young University. 2014.

[7] Andrew Levandoski and Jonathan Lobo. Predicting the NCAA Men's Basketball Tournament with Machine Learning. 25 April, 2017.

[8] Lo-Hua Yuan, Anthony Liu, Alec Yeh, Aaron Kaufman, Andrew Reece, Peter Bull, Alex Franks, Sherrie Wang, Dmitri Illushin and Luke Bornn. A mixture-of-modelers approach to forecasting NCAA tournament outcomes. 2015.

[9] Nicholas Bennett. Comparing various machine learning statistical methods using variable differentials to predict college basketball. Spring 2018.

[10] Zifan Shi, Sruthi Moorthy, Albrecht Zimmermann. Predicting NCAAB match outcomes using MLtechniques – some results and lessons learned. KU Leuven, Belgium. 14 Oct 2013.

[11] Adarsh Kannan, Brian Kolovich, Brandon Lawrence, Sohail Rafiqi. Predicting National Basketball Association Success: A Machine Learning Approach. 2018.

[12] Ayala Neudorfer and Saharon Rosset. Predicting the NCAA basketball tournament using isotonic least squares pairwise comparison model. 2018.

[13] Bryce Brown. Predictive Analytics for College Basketball: Using Logistic Regression for Determining the Outcome of a Game. Peter. T Paul College of Business and Economics University of New Hampshire. May 2019.

[14] Cody Kocher and Tim Hoblin. Predictive Modelfor the NCAA Men's Basketball Tournament. Ball State University Muncie, IN. April 2017.

[15] March Machine Learning Mania 2021 - NCAAM. Accessed: 10 November, 2021. [Online]. Available: https://www.kaggle.com/c/ncaam-march-mania-2021

[16] Ken Pomeroy statistics. Accessed: 10 November, 2021. [Online]. Available: https://kenpom.com

[17] Ken Pomeroy statistics. Accessed: 10 November, 2021.Accessed: 10 November, 2021. [Online]. Available: https://kenpom.com/blog/

[18] João Fonseca. March Madness prediction using machine learning techniques. November 2017.