NCAA Men's Basketball Games Analysis and Prediction

CSE 6242 Final Project Report

Team 41

Tzu-Wei Huang
Somayeh Hosseini Porgham
Yi-Chi Shao
Pao-Yang Tsai

Introduction and Motivation

NCAA men's basketball tournament is a popular annual sports events in U.S. colleges. Every year hundreds of teams and thousands of players do everything they can to win games and fulfill their dream - becoming national champion! However, predicting the outcomes of an entire tournament is a monumental challenge. Being enthusiastic basketball fans, we can't wait to participate in this "out-of-stadium" basketball competition!

Problem Definition

Our goal in this project is to predict outcome of each game with former statistical data. The first task is to gather and analyze the data in the past several seasons. Afterwards, we adopt distinct machine learning methods, trying to find out critical features, and gradually improve the prediction accuracy. Finally, we visualize the win rate of different battle combinations with the best predicting strategy.

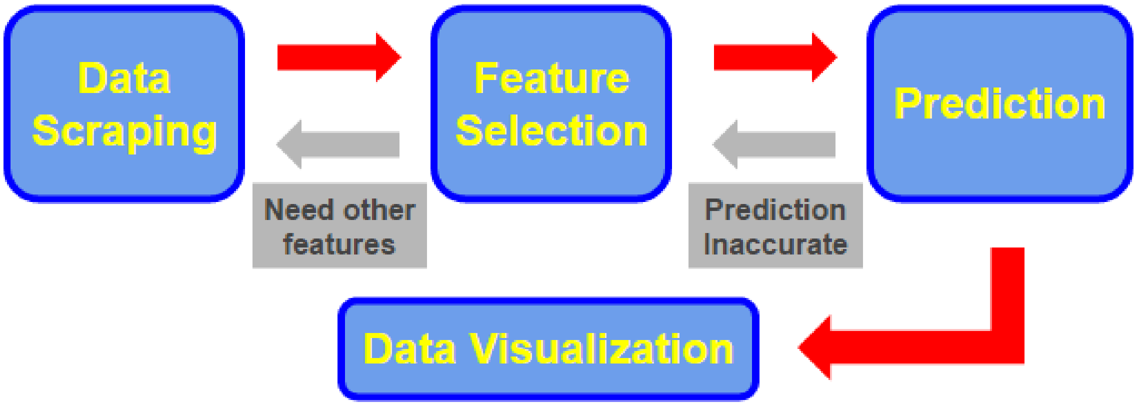


Figure 1. Process of our implementations

Survey

Previous research [1] introduces different types of data mining tasks and general life cycle of data mining. It also mentions popular data mining methods and applications of data mining in various field including sports. Research in [2] introduces modeling methods to predict sports outcome. Researchers have [3] which presents feature selection methods and classification techniques used in sport prediction systems. It also indicates advantages and disadvantages of each system. Implementation in [4] uses data mining to predict the outcomes of NBA matches. In this paper, it presents the attributes used to describe a basketball match. It uses Naive Bayes to get a 67% accuracy.

Logistic regression is a regression model [5] which measures the relationship between the categorical dependent variable and one or more independent variables by estimating probabilities using a logistic function, which is the cumulative logistic distribution. It is used widely in predicting: previous research [6] proposes the processes involved in developing a logistic regression model by student’s test scores, performance, and other factors to predict whether or not a student will eventually enroll. Research in [7] used robust logistic regression with three-fold cross validation to compare the classification and prediction of bankrupt firms.

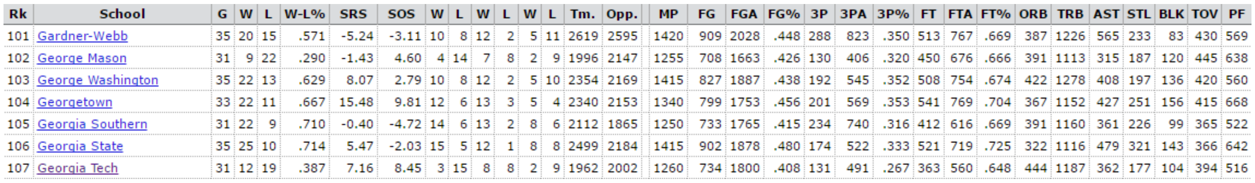
In [8], the authors evaluated several classification learners and evaluate the performance. And the paper indicates that attributes are more important than the model, simple model might perform better than complex model and the predictive accuracy has a glass ceiling around 75%. In [9], four neural networks method are used in predicting the success of basketball teams in NBA. Authors investigate the best features for prediction and they compared the result with NBA’s expert prediction. The best prediction is 74.33%. Paper [10] introduces several ML methods including Linear Regression, Maximum Likelihood classifier, Multilayer Perceptron, SVM. All the result is smaller than 75%. In paper [11], it introduces the combined logistic regression and Markov chain model for predicting the NCAA outcome. The estimation result is good since it does not only treat the outcome of games as binary event but give the estimation of the probability of winning team better than losing team.

Researchers have [12] which suggests maximum score estimator to improve the number of correct predictions. Previous research [13] computes the probability of the seed combination in each round based on past tournaments. It also models the distribution of winning seeds as a truncated geometric random variable. In [14], the authors use statistical probit regressions for predicting the outcomes of both men’s and women’s basketball and tennis tournaments. It considers a relationship between the relative difference in rankings (seeding) and the relative frequency of winning teams. Implementation in [15] predicts the 2014 NCAA basketball tournaments. It aims to avoid data contamination (archival data for a season that includes the results of the final tournament from that year) and uses regularization to avoid overfitting and multicollinearity due to relatively smaller number of inputs and the large of number of covariates. It studies logistic regression, stochastic gradient boosting, neural networks, and ensembles and suggests that logistic regression, stochastic gradient boosting, and neural network algorithms provided the best prediction performance.

Proposed Methods

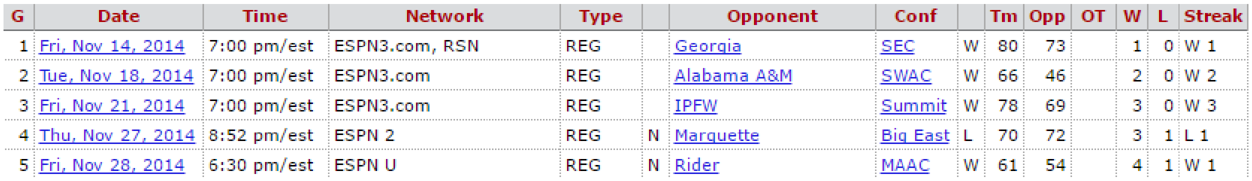
1) Data Scraping

The data we use in this project is mostly from *College Basketball @ Sports-Reference.com*. This website contains several aspects of statistical data of NCAA men’s basketball. Take figure 2 and 3 for instances, these two tables represent distinct information.



Rk	School	G	W	L	W-L%	SRS	SOS	W	L	W	L	W	L	Tm.	Opp.		MP	FG	FGA	FG%	3P	3PA	3P%	FT	FTA	FT%	ORB	TRB	AST	STL	BLK	TOV	PF
101	Gardner-Webb	35	20	15	.571	-5.24	-3.11	10	8	12	2	5	11	2619	2595		1420	909	2028	.448	288	823	.350	513	767	.669	387	1226	565	233	83	430	569
102	George Mason	31	9	22	.290	-1.43	4.60	4	14	7	8	2	9	1996	2147		1255	708	1663	.426	130	406	.320	450	676	.666	391	1113	315	187	120	445	638
103	George Washington	35	22	13	.629	8.07	2.79	10	8	12	2	5	10	2354	2169		1415	827	1887	.438	192	545	.352	508	754	.674	422	1278	408	197	136	420	560
104	Georgetown	33	22	11	.667	15.48	9.81	12	6	13	3	5	4	2340	2153		1340	799	1753	.456	201	569	.353	541	769	.704	367	1152	427	251	156	415	668
105	Georgia Southern	31	22	9	.710	-0.40	-4.72	14	6	13	2	8	6	2112	1865		1250	733	1765	.415	234	740	.316	412	616	.669	391	1160	361	226	99	365	522
106	Georgia State	35	25	10	.714	5.47	-2.03	15	5	12	1	8	8	2499	2184		1415	902	1878	.480	174	522	.333	521	719	.725	322	1116	479	321	143	366	642
107	Georgia Tech	31	12	19	.387	7.16	8.45	3	15	8	8	2	9	1962	2002		1260	734	1800	.408	131	491	.267	363	560	.648	444	1187	362	177	104	394	516

Figure 2. Season average of schools

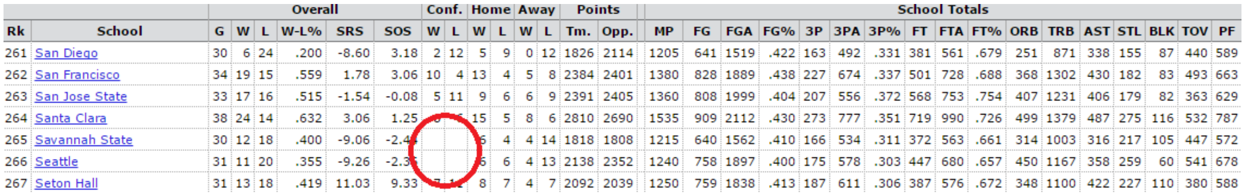


G	Date	Time	Network	Type		Opponent	Conf		Tm	Opp	OT	W	L	Streak
1	Fri, Nov 14, 2014	7:00 pm/est	ESPN3.com, RSN	REG		Georgia	SEC	W	80	73		1	0	W 1
2	Tue, Nov 18, 2014	7:00 pm/est	ESPN3.com	REG		Alabama A&M	SWAC	W	66	46		2	0	W 2
3	Fri, Nov 21, 2014	7:00 pm/est	ESPN3.com	REG		IPFW	Summit	W	78	69		3	0	W 3
4	Thu, Nov 27, 2014	8:52 pm/est	ESPN 2	REG	N	Marquette	Big East	L	70	72		3	1	L 1
5	Fri, Nov 28, 2014	6:30 pm/est	ESPN U	REG	N	Rider	MAAC	W	61	54		4	1	W 1

Figure 3. Outcomes of all games for one school in a season

Next, we use *Scrapy* which is an open source and collaborative framework for extracting the data from websites. There are some essential parts when we do the data scraping. The first one is "callback function": for spiders, we start by generating the initial requests to crawl the first URLs, and specify a callback function to be called with the response downloaded from those requests. This method is applied when we scrape outcomes of all the games. The initial requests are on URLs which contain a list of all games for each day, and then following requests call back those URLs which contain details of each game.

The second one is "row length checking": when scraping a table from HTML, each row should contain constant amount of elements. However, sometimes we might encounter element-missing problem as shown in figure 4. Those cells would be considered as NULLs and lead to errors if we are not aware of their existence. To avoid this, when scraping each row, the code should check the length of it. If the amount doesn't match what it should be, we can compensate some specific numbers into those NULL cells, or just discard the whole row.



		Overall						Conf.		Home		Away		Points		School Totals																
Rk	School	G	W	L	W-L%	SRS	SOS	W	L	W	L	W	L	Tm.	Opp.	MP	FG	FGA	FG%	3P	3PA	3P%	FT	FTA	FT%	ORB	TRB	AST	STL	BLK	TOV	PF
261	San Diego	30	6	24	.200	-8.60	3.18	2	12	5	9	0	12	1826	2114	1205	641	1519	.422	163	492	.331	381	561	.679	251	871	338	155	87	440	589
262	San Francisco	34	19	15	.559	1.78	3.06	10	4	13	4	5	8	2384	2401	1380	828	1889	.438	227	674	.337	501	728	.688	368	1302	430	182	83	493	663
263	San Jose State	33	17	16	.515	-1.54	-0.08	5	11	9	6	6	9	2391	2405	1360	808	1999	.404	207	556	.372	568	753	.754	407	1231	406	179	82	363	629
264	Santa Clara	38	24	14	.632	3.06	1.25	8	6	15	5	8	6	2810	2690	1535	909	2112	.430	273	777	.351	719	990	.726	499	1379	487	275	116	532	787
265	Savannah State	30	12	18	.400	-9.06	-2.44			6	4	4	14	1818	1808	1215	640	1562	.410	166	534	.311	372	563	.661	314	1003	316	217	105	447	572
266	Seattle	31	11	20	.355	-9.26	-2.35			6	6	4	13	2138	2352	1240	758	1897	.400	175	578	.303	447	680	.657	450	1167	358	259	60	541	678
267	Seton Hall	31	13	18	.419	11.03	9.33	7	12	8	7	4	7	2092	2039	1250	759	1838	.413	187	611	.306	387	576	.672	348	1100	422	227	110	380	588

Figure 4. Missing cells in a table

Afterwards, we scrape and store the tables in *Sqlite* database to in order to make the data more applicable. There are totally 32,756 games, 2,084 teams and 26,497 players in 6 years (2010-2011 ~ 2015-2016).

2) Feature Selection

In order to predict the outcome of each game from scraped data (matrix.tsv), we need to select features from the data to train prediction models.

The season average status of a team is a good factor that shows the long-term performance of a team over a season. Another important factor is the average status of a team's opponents in the season. For instance, If two teams have similar average status in a season, the team with better average status of opponents has better performance than the other.

Besides the long-term performance, we also have to take short-term performance into consideration. The reason is that the performance of a team might change rapidly during a season, so it is better to consider both short-term and long-term performance together. To obtain short-term performance, we calculate the average status of previous N games of a team.

What's more, the composition of a team is also an important factor to consider. To represent the composition of a team, we use bag-of-words method to represent a team as a bag of players. At first, we use K means to cluster players in the dataset into K groups. Then, we create a histogram of K player types for a team. We can use this histogram as features that reflect the composition of a team.

3) Prediction

In this project, we use *scikit-learn* in *Python* to predict the outcomes of games.

To predict the outcome (i.e. win or lose), we concatenate the features of both teams together as the features of a game, and take the outcome as the label of a game. We use the features and labels from training dataset to train a model for prediction.

We choose logistic regression as the algorithm for prediction. The main reason is that, for our case, the accuracy of logistic regression is as good as other models such as SVM and random forest, but it has shorter training timer than others. What's more, in *scikit-learn*, the logistic regression provides not only predict() for classificaition but also predict_proba() to compute probability which is useful for data visualization.

Given a game between TeamA and TeamB with outcome O in training dataset, we not only train the model with [features of TeamA, features of TeamB] and label O but also [features of TeamB, features of TeamA] and label O. To predict the outcome of a game between TeamA and TeamB, we input [features of TeamA, features of TeamB] into the model to compute TeamA's win probability, Pa, in a game between TeamA and TeamB. We also input [features of TeamB, features of TeamA] into the model to compute TeamB's win probability, Pb, in a game between TeamB and TeamA. Eventually, we take (Pa + 1- Pb)/2 as TeamA's win probability. If the win probability is higher than 50%, we determine that TeamA will win the game and vice versa.

4) Data Visualization

We use Heatmap from *d3.js* library to implement interactive visualization of different battle combinations. In order to create the heatmap, we use a python script (sort.py) to manipulate the scraped data in matrix.tsv to sorted.tsv file.

For convenience, each team and its opponent in sorted.tsv is shown by a unique ID from 0 to 63. This file is also sorted so that the correlation between teams in the heatmap is shown in a more understandable fashion (figure 5).

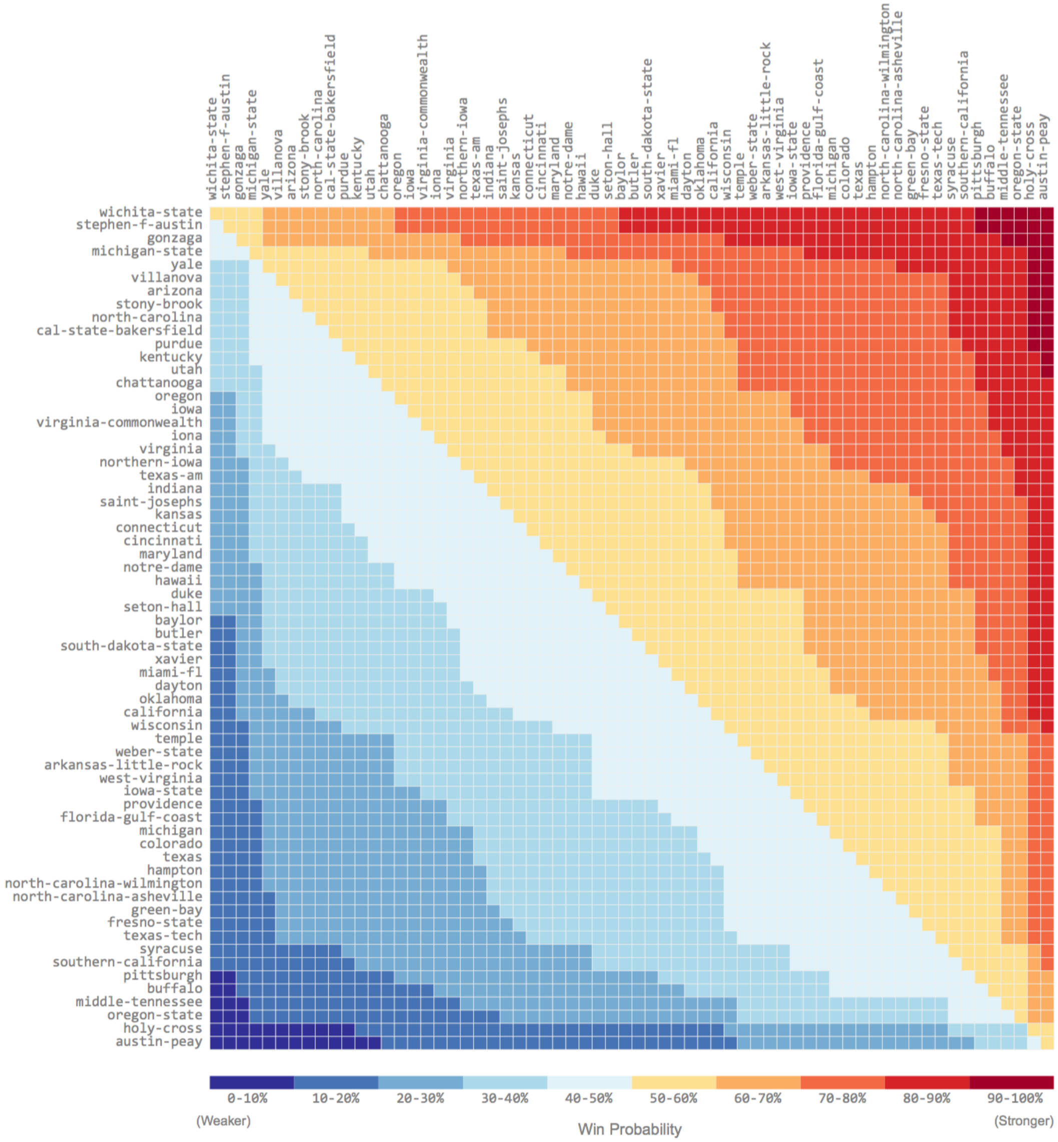


Figure 5. Heatmap - win rate of different battle combinations

As it can be observed from Figure 5, stronger teams are shown by hot range of colors like orange and red and weaker teams are shown by cold range of colors like blue.

Hovering over the cells of the map; shows the winning probability of the related team in that row against the team in the corresponding column while magnifies the name of the two competing teams among other teams.

Figure 6 shows the brackets of tournament with pie-shaped graph. The left one is the actual result, and the right one is generated based on our prediction. There are 64 teams in the first round which occupied the outermost rim. Afterwards, winning teams get into inner rims until the final champion is determined (the innermost circle). As we can observe, the accuracy of the first round is about 70% which is close to our performance. However, after the second round, the accuracy is gradually diluted so that the final outcomes are hardly predicted.

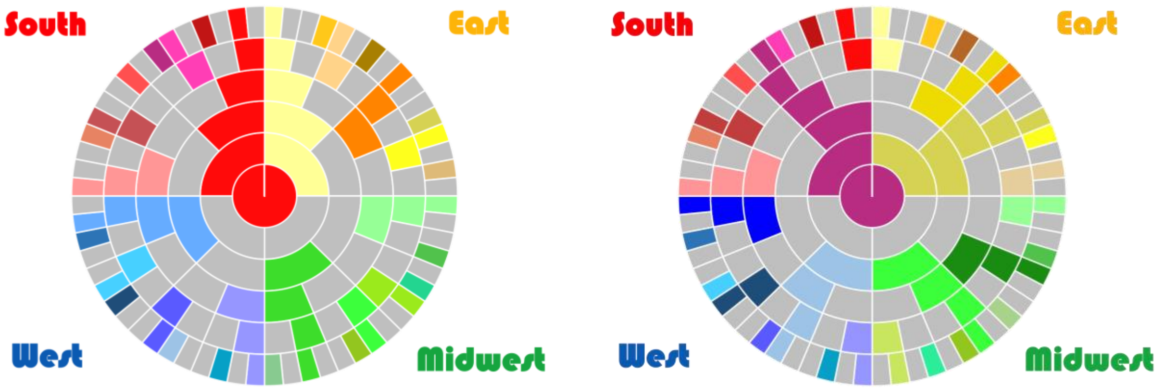


Figure 6. Brackets - NCAA tournament (L: actual result; R: predictive result)

Experiments/ Evaluation

1) Description of Testbed

In the experiment, we store the data in the SQLite database and use it for data preprocessing. The conference and NCAA tournament games data from 2010-2011 to 2014-2015 seasons and the conference games data of 2015-2016 is used to train the machine. After training stage, we used the data of 2015-2016 NCAA tournament games to do testing. To generate the input data format, SQL commands are adopted to join related tables together.

For example, there are 2 tables: “AllGamesScore”, which contains date, names and scores of both team, type, and home/away of all games; “AllTeamsConfAvg”, which contains season, team name, conference games average numbers like point, field goal percentage, rebounds, assists, blocks, and turnovers of all teams. To train the data based on only conference average, we join these two tables with both of the team names, and thus the new table would look like <date, teamA name, teamB name, teamA score, teamB score, type, home/away, conference games average numbers of teamA, conference games average numbers of teamB>. Afterwards, we filter out some unnecessary features and games to clean the table and thus a fundamental testbed is created.

In addition, we still have some tables such as “AllPlayersConfAvg”, which contains conference average numbers of all players. With all of these data including 32,756 games, 2,084 teams and 26,497 players, our goal is to test NCAA tournament games by training conference games and achieve prediction accuracy as higher as possible.

2) Details of Experiments

The goal of our experiments is to figure out how different features affect the accuracy. We use instinctive combinations of features: TeamSeasonStatus, TeamSeasonStatus + BagOfPlayers and TeamSeasonStatus + TeamPreviousStatus to train the model and calculate the accuracies.

TeamSeasonStatus

This experiments use average status in conference as the feature of a team. We use the games data of all the games in 2010-2011, 2011-2012, 2012-2013, 2013-2014 and 2014-2015 seasons to train the model, and predict the outcomes of 2015-2016 NCAA tournament games. What's more, instead of just testing the accuracy for 2015-2016 NCAA tournament, we also use 6-folds cross validation to calculate the average accuracy. The average accuracy is 60.44% and the accuracy of 2015-2016 tournament is 68.66%. We will take these results as the baseline of other experiments.

BagOfPlayers

In this experiment, we use K-means to cluster all the players in the dataset based on their average status in conference. Then, for each team, we classify each player to one of the K types and build a histogram of the K types to represent the composition of the team. To train the model, we concatenate the histogram to TeamSeasonStatus to build a new features for each team.

We try different K values from 1 to 10. The result shows that the average accuracy is highest when K is 4. With K=4, the average accuracy is 61.70% and the accuracy of 2015-2016 tournament is 70.15%.

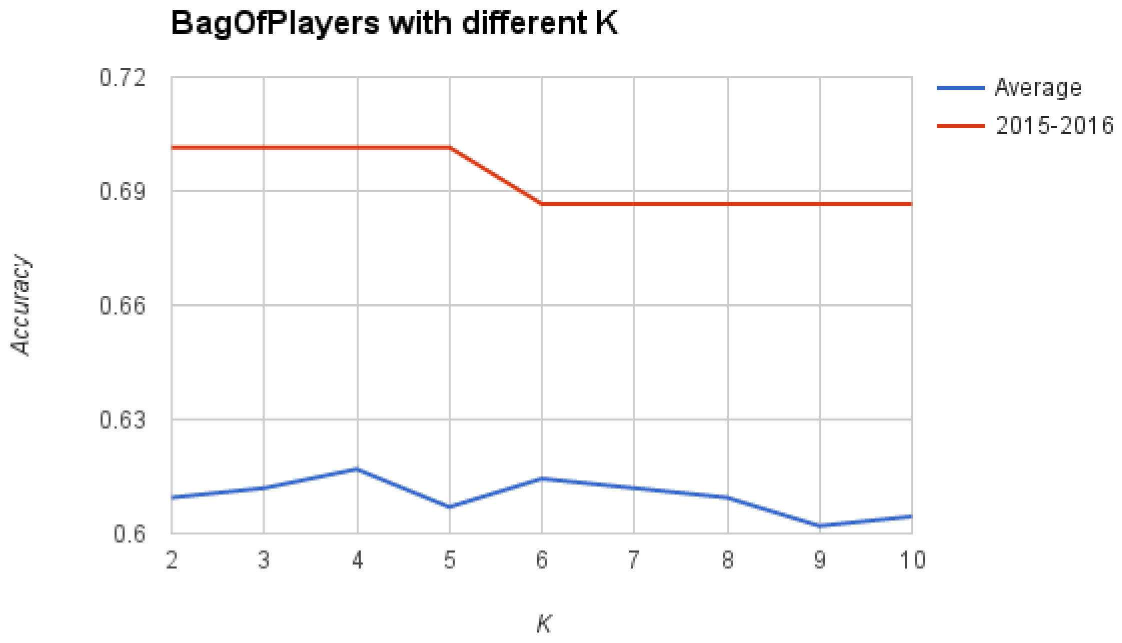


Figure 7. Bag of players with different K

With BagOfPlayers, the average accuracy is improved by 1.26% and the accuracy of 2015-2016 tournament is improved by 1.49%. This result shows the composition of a team is an important factor to consider for prediction.

TeamPreviousNGamesStatus

In this experiment, we not only use average status in conference but also use average status of previous N games as the features of a team. We calculate the average status of the most recently N games before of both teams of a game and append it to TeamSeasonStatus to train the model and predict the outcome.

We try different N values (3, 5, 7, 9) to figure how the status of previous N games affect the accuracy. The result shows that TeamPreviousNGamesStatus cannot improve the accuracy. Even worse, it decreases the accuracy.

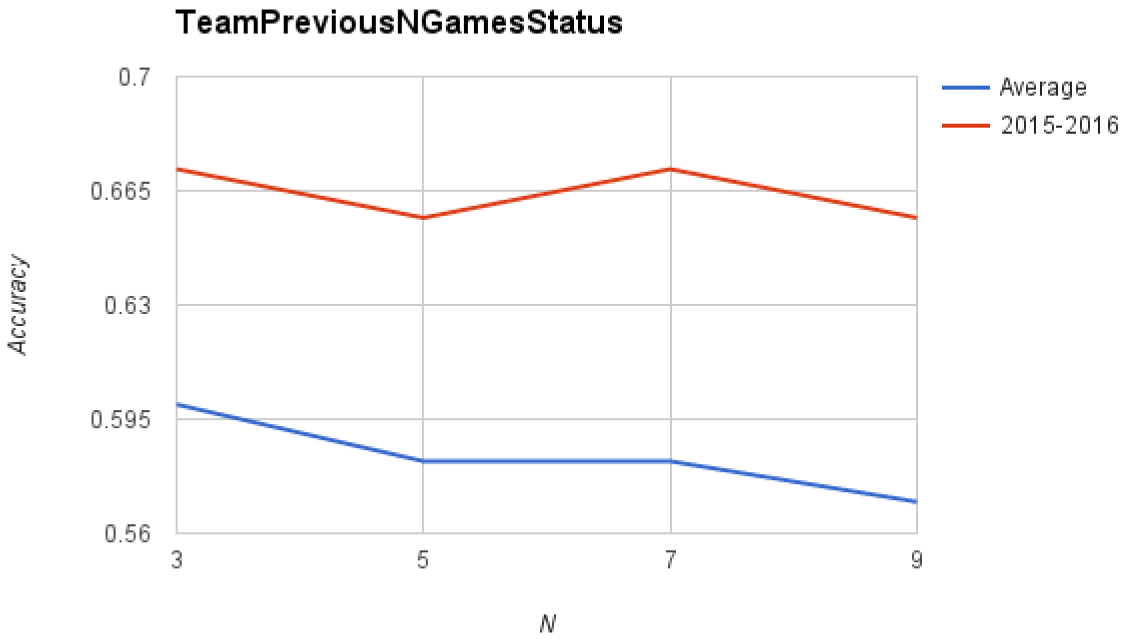


Figure 8. Team previous N games status

A possible reason of this phenomenon could be that the status of a team can changes rapidly in a short time period. Therefore, short term factor is not representative to indicate the strength of a team.

Conclusions and discussion

The prediction to NCAA men's basketball tournament result is always an exciting and challenging thing. Our goal in this project is to predict outcome of each game with former statistical data. The data we used are scraped from *College Basketball @ Sports-Reference.com* which includes totally 32,756 games, 2,084 teams and 26,497 players in 6 seasons (2010-2011 ~ 2015-2016).

The machine learning model we choose to predict the win rate is logistic regression. Feature selection is a very important phase during the whole model training process. We consider both the long term features and short term features. The long term features could show the average status of the team for whole season and the short term feature could show the recent status of the team before the game starts. The result of the experiments shows that short-term features does not improve the prediction. The new features we introduced are the bag of players method which improves the performance a little. The reason is that from bag of players, we know the composition of a team which could be a helpful information to predict.

After we trained the model, we plot the heatmap and the doughnut-shaped bracket to do visualization from which we could know different battle combination with their related win rate and the comparison of our prediction and the real result.

Distribution of team member effort

(1) Pao-Yang Tsai, (2) Somayeh Hosseini Porgham, (3) Tzu-Wei Huang, (4) Yi-Chi Shao

- Data collection (1), (3)
- Data cleaning and integration (1)
- Feature selection (1), (4)
- Try different data-mining algorithms (3)
- Finish progress report (1), (2), (3), (4)
- Build and test the system (3), (4)
- Data visualization (2)
- Build user interface for demo (2)
- Finish final report and presentation (1), (2), (3), (4)

*All team members contribute similar amount of effort.

References

[1] Padhy, N., Mishra, D., & Panigrahi, R. (2012). The survey of data mining applications and feature scope. arXiv preprint arXiv:1211.5723.
[2] Schumaker, R. P., Solieman, O. K., & Chen, H. (2010). Predictive modeling for sports and gaming (pp. 55-63). Springer US.
[3] Haghighat, M., Rastegari, H., & Nourafza, N. (2013). A review of data mining techniques for result prediction in sports. Advances in Computer Science: an International Journal, 2(5), 7-12.
[4] Miljkovic, D., Gajić, L., Kovacevic, A., & Konjovic, Z. (2010, September). The use of data mining for basketball matches outcomes prediction. In Intelligent Systems and Informatics (SISY), 2010 8th International Symposium on (pp. 309-312). IEEE.
[5] Peng, C. Y. J., Lee, K. L., & Ingersoll, G. M. (2002). An introduction to logistic regression analysis and reporting. The Journal of Educational Research, 96(1), 3-14.
[6] Sampath, V., Flagel, A., & Figueroa, C. (2009). A logistic regression model to predict freshmen enrollments.
[7] Hauser, R. P., & Booth, D. (2011). Predicting bankruptcy with robust logistic regression. Journal of Data Science, 9(4), 565-584.
[8] Shi, Z., Moorthy, S., & Zimmermann, A. (2013, October). Predicting NCAAB match outcomes using ML techniques–some results and lessons learned. In Machine Learning and Data Mining for Sports Analytics ECML/PKDD 2013 workshop.
[9] Loeffelholz, B., Bednar, E., & Bauer, K. W. (2009). Predicting NBA games using neural networks. Journal of Quantitative Analysis in Sports, 5(1).
[10] Y. H. Hu. (2013) Prediction of NBA games based on Machine Learning Methods
[11] Kvam, P., & Sokol, J. S. (2006). A logistic regression/Markov chain model for NCAA basketball. Naval research Logistics (NrL), 53(8), 788-803.
[12] Caudill, S. B. (2003). Predicting discrete outcomes with the maximum score estimator: The case of the NCAA men's basketball tournament. International Journal of Forecasting, 19(2), 313-317.
[13] Jacobson, S. H., Nikolaev, A. G., King, D. M., & Lee, A. J. (2011). Seed distributions for the NCAA men's basketball tournament. Omega, 39(6), 719-724.
[14] Boulier, B. L., & Stekler, H. O. (1999). Are sports seedings good predictors?: an evaluation. International Journal of Forecasting, 15(1), 83-91.
[15] Yuan, L. H., Liu, A., Yeh, A., Kaufman, A., Reece, A., Bull, P., ... & Bornn, L. (2015). A mixture-of-modelers approach to forecasting NCAA tournament outcomes. *Journal of Quantitative Analysis in Sports*, *11*(1), 13-27.
[16] http://bl.ocks.org/ianyfchang/8119685