CUNY SPS Capstone Project

Nicholas Capofari

5/1/2018

Introduction

The National Basketball Association uses a 4 round playoff system to determine which of its 30 teams will be crowned NBA Champion at the end of each season. Since the 1983-1984 season, the 8 best teams in the Eastern and Western Conference gain entry to the playoffs. The winners of each conference meet each other in the NBA Finals.

Teams in each conference are seeded 1 to 8 based upon their regular season winning percentage. The one caveat being that the winner of each division (there are 3 divisions in each conference) is guaranteed a top 4 seed, regardless of record. The higher seed in each round of the playoffs has home court advantage, meaning that the series will begin and end at the higher seed's arena and the majority of games will be played there.

The goal of this project is to create a model that predicts whether a home team will win an NBA playoff series. Only playoff series within each conference were inspected for this project, the NBA Finals were excluded. To create an accurate model, three types of binary classifiers were chosen to model the data set:

- Support Vector Classification (SVC)
- Random Forest (RFC)
- K-Nearest Neighbors (KNN)

Literature Review

NBA predictions are abundant. Game predictions and playoff series predictions can be found by the click of a button. It has been shown that using advanced statistics is more accurate predicting NBA team success compared to basic statistics. A set of advanced statistics are the offensive and defensive four factors. These statistics are highly correlated with NBA team performance (Oliver 2004).

Another useful prediction measure for NBA performance is a team's Pythagorean Winning Percentage (Winston 2012). Using points scored and points allowed, the Pythagorean Win Percentage is more accurate compared to traditional winning percentage at determining the success of a team's future games.

There are many types of models that can be used to predict the outcome of a binary event (Caruana and Niculescu-Mizil 2006). This project is only interested in models that perform well with small data sets. There have only been 476 NBA playoff series since the advent of the 16 team playoff system (NBA season 1983-1984). Models perform better when there are more observations. When creating the model, 20% of observations were set aside to be used as the data test set. This left only 380 observations for the training set.

For all models, and especially Support Vector Machines, hyperparameters need to be adjusted to get the best performance (Ben-Hur and Weston 2017). Without adjusting the hyperparameters, the resulting model will most likely not perform with the highest degree of accuracy.

The majority of the NBA playoff data set represents a team that won the playoff series. This is not an example of imbalanced data (He and Garcia 2009), but it is important to know how a data set will perform if the majority of the target values are the same. Close to 3 out of 4 home teams end up winning their NBA playoff series.

Past research suggests that big men, specifically centers, are the most valuable players on a basketball team (Lee and Berri 2008). Research conducted before the 2012-2013 may undervalue smaller players. Since then, the average number of 3 pointers attempted per team has increased from an average of 18 per game to 28 (NBA-reference.com 2018).

Data Collection

All raw statistics were collected from NBA-reference.com. A group of NBA scraper `R` scripts that can be found here. These scripts utilized `R` packages that are essential to any data scraping, data munging task.

Starting with the 1983-1984 playoffs and ending with the 2016-2017 season, all playoff results were collected and sanitized. Each observation represents a home team (which will always be the higher seeded team) regular season cumaltive statistics, their opponent's regular season cumaltive statistics, and the playoffs series result (Boolean: True if the team won the playoff series). All NBA Finals series were omitted from the data set.

Using each playoffs team's unique season-team link, regular season data for each team and each opponent was retrieved and added to the data set. For each team and opponent, all offensive and defensive stats were collected.

Of major importance for our model creation were Pythagorean Win Percentage and statistics that help us measure the NBA four factors (Oliver 2004):

- Offensive and Defensive Effective Field Goal Percentage
- Offensive and Defensive Turnover Percentage
- Offensive and Defensive Rebounding Percentage
- Offensive and Defensive Free Throw Rate

Pythagorean Win Percentage

$$Games \times (Team\ Points^{14} \div (Team\ Points^{14} + Opponent^{14}))$$

This formula is obtained by fitting a logistic regression model with $\log(Team\ Points \div Opponent\ Points)$ as the explanatory variable. Using this formula for all BAA, NBA, and ABA seasons, the root mean-square error (rmse) is 3.14 wins. Daryl Morey was the first person to apply Pythagorean Win Percentage to the NBA. This formula was originally created by the sabermatrician Bill James for Major League Baseball. Morey originally used 13.91 as the exponential value but using 14 has become the norm.

For our models, the difference between two teams' Pythagorean Win Percentage is used as a feature.

Four Factors

The four factors that determine if a team will win or lose a game boil down to:

- Shooting
- Turnovers
- Rebounding
- Free Throws

While these four factors are essential in determining the success of an NBA team, it is undecided how much weight should be given to each category. Below are the statistics that can be used to determine a team's four factor performance.

Effective Field Goal Percentage

$$(Field\ Goals + 0.5 \times Three\ Point\ Field\ Goals) \div Field\ Goals\ Attempted$$

Turnover Percentage

$$100 \times Turnovers \div (Field\ Goals\ Attempted + 0.44 \times Free\ Throws\ Attempted + Turnovers)$$

Rebounding Percentage

$$100 \times \frac{Defensive\ Rebounds \times (Team\ Minutes\ Played \div 5)}{Team\ Minutes\ Played \times (Team\ Defensive\ Rebounds + Opponent\ Offensive\ Rebounds)}$$

To find Offensive Rebounding Percentage, switch all occurrences of Defensive Rebounds to Offensive Rebounds and vice versa.

Free Throw Rate

$$Free\ Throws\ Made \div Field\ Goals\ Attempted$$

Non Factors

For each team, two other statistics were derived from the available data. First, playoff experience was calculated by taking the sum of all players on a team's historical playoff minutes. Also, team balance, which attempts to determine whether a team relies too heavily on a single player during the course of a season. Both of these statistics turned out to not be correlated with playoff series wins.

Model Creation

The basic model that will be used as baseline for model comparison is a model that simply chooses the home team to win each series. There were 476 playoff series in our data set, 14 series over 34 years.

By Matchup

Seed	Opponent Seed	n	Won Series	Win Percent
1	8	68	63	0.926
2	7	68	63	0.926
3	6	68	51	0.750
4	5	68	31	0.456
2	3	47	24	0.511
1	5	36	33	0.917
1	2	30	15	0.500
1	4	27	25	0.926
1	3	24	20	0.833
2	6	16	13	0.812
3	7	4	4	1.000

Seed	Opponent Seed	n	Won Series	Win Percent
4	8	4	3	0.750
1	6	3	2	0.667
2	4	3	1	0.333
2	5	3	3	1.000
3	4	2	2	1.000
1	7	1	1	1.000
2	8	1	0	0.000
3	5	1	1	1.000
5	8	1	1	1.000
6	7	1	0	0.000

The measurement we will be using to compare each model is *precision*.

$$precision = \frac{True\ Positives}{True\ Positives + False\ Positives}$$

Since our basic model will never produce any True Negatives or False Negatives, other metrics would be misleading. By relying solely on precision, only when models produce win predictions will the predictions and resulting accuracy be taken into account.

The basic model's $precision_{bm} = 0.7479$.

For each model created, scikit-learn's GridSearchCV was used to help determine the hyperparameters selected for each model. GridSearchCV exhaustively generates candidates from a grid of parameter values that are specified by the user.

To combat overfitting, cross-validation was utilized. Using cross-validation splits the training data into smaller data sets, thus reducing the chances that models will overfit the training data.

By Seed Difference

Seed Difference	n	Won Series	Win Percent
one	148	72	0.486
not one	328	284	0.866

As you can see above, when the `Seed Difference` is equal to one, flipping a coin will produce better results compared to our basic model (always choosing the home team). Because of the vast difference in the win percentages when `Seed Difference` is equal to one compared to when it is not equal to one, models were created for the following three data sets:

- All observations
- `Seed Difference == 1`
- `Seed Difference != 1`

Model Selection

Three algorithms were chosen to create the non-basic models. Each model is an example of a binary classifier.

- Support Vector Classification (SVC)

- Random Forest (RFC)
- K-Nearest Neighbors (KNN)

Support Vector Classification (SVC)

A Support Vector Machine is a very powerful model that can be used for classification purposes. These models are well suited for complex but small data sets (Geron 2017). SVMs, and in turn SVCs, use soft margin classification to find a good balance between margin violations and the size between decision boundaries.

Random Forest (RFC)

A Random Forest is an ensemble of Decision Trees. The Random Forest algorithm introduces more randomness when creating trees resulting in greater diversity in the produced trees. RFCs can produce better models compared to Decision Trees because they trade higher bias for lower variance, thus yielding an overall better model.

K-Nearest Neighbors (KNN)

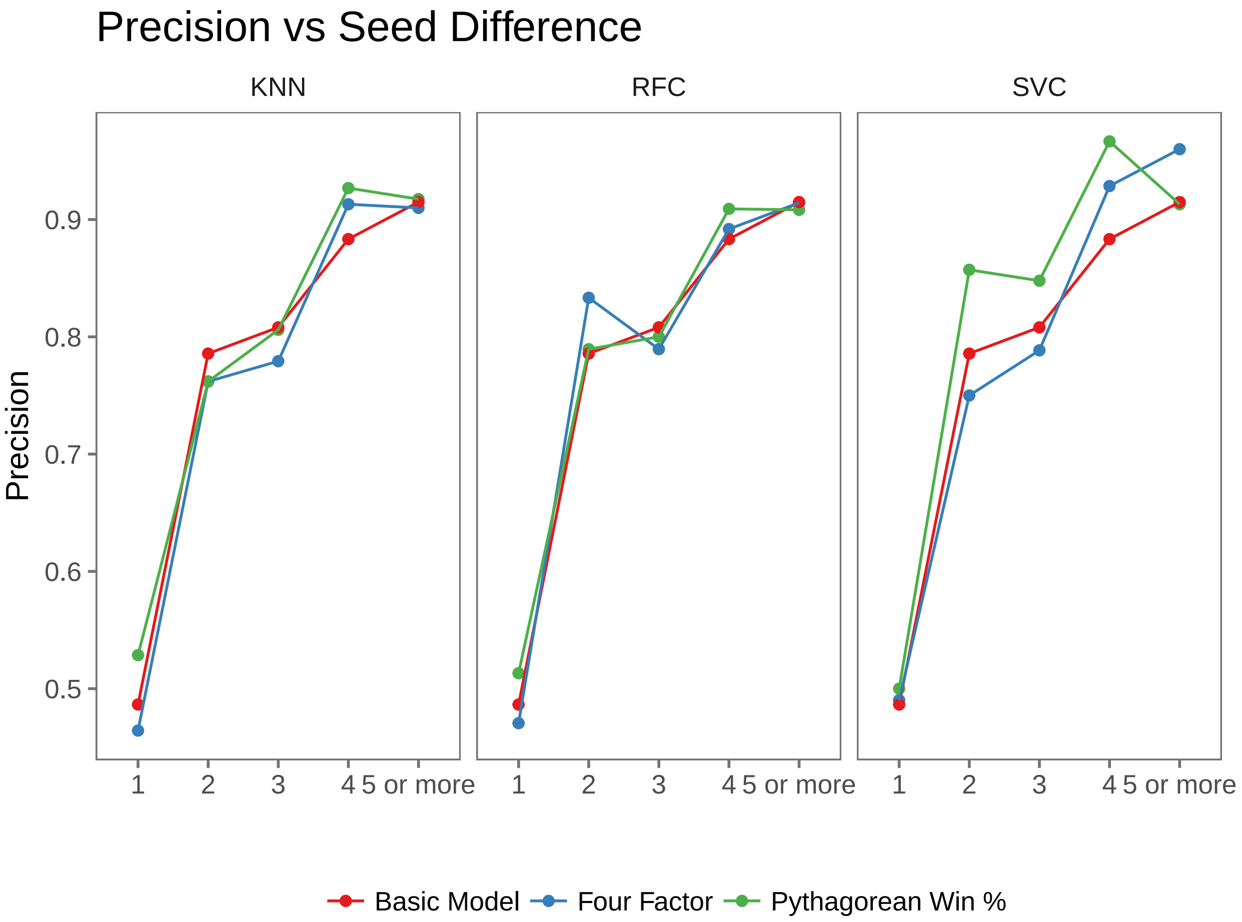
The K-Nearest Neighbor algorithm is a data classification algorithm that attempts to estimate how likely a data point is to be a member of a group. By using certain distance functions, group membership is determined.

Feature Selection

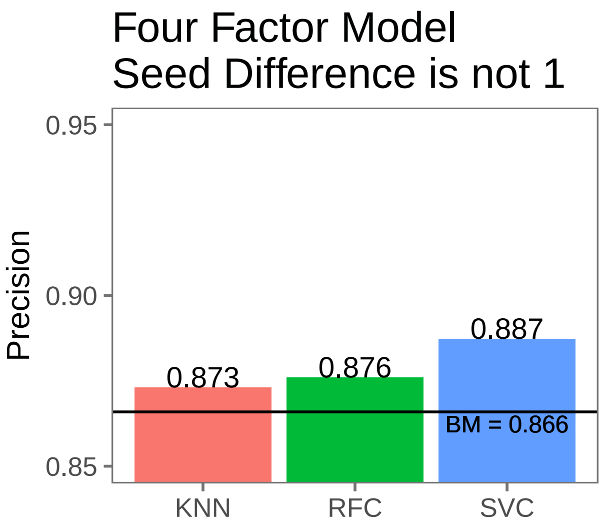
For each model type, two different sets of features were selected. The first set of features was just a single feature; the difference in Pythagorean Win Percentage of the two teams. The second feature set was the four offensive and defensive factors for each team (16 features in total). For the four factor models, the features were winnowed down in order to remove redundant or irrelevant features. Using ANOVA F-values, 5 features were chosen to remain in the model.

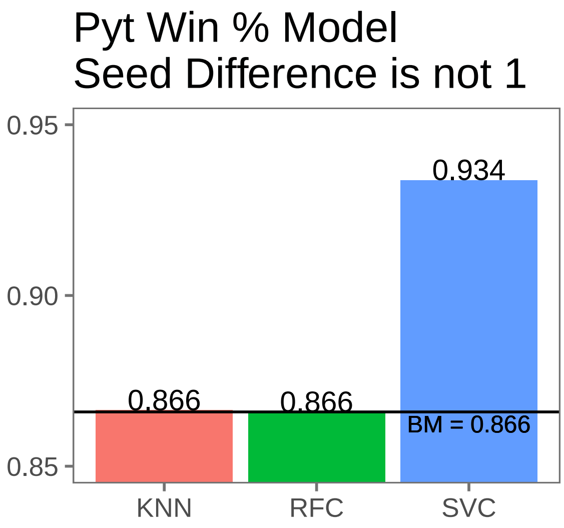
Model Performance

Since teams are seeded based upon their year end records, series encompassing teams that are seeded close to one another are difficult to predict. As seed differences increased the models were able to perform better.

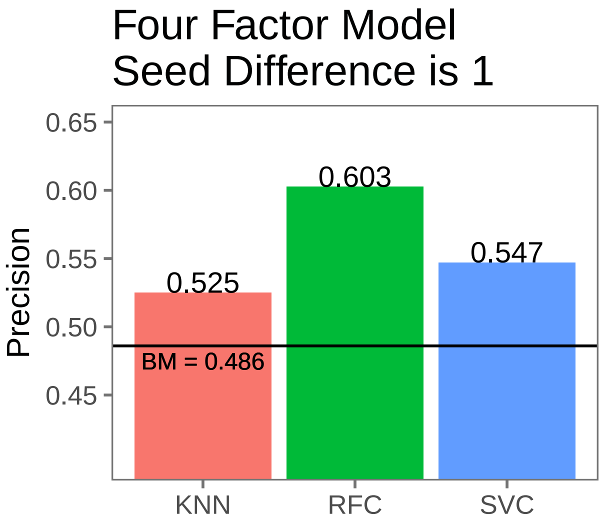


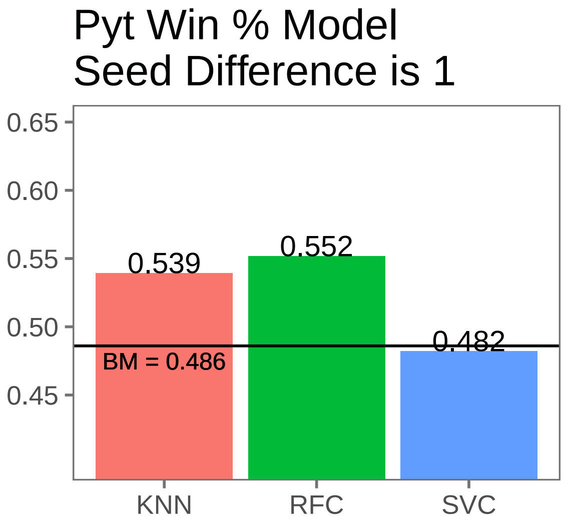
When the difference between two teams' seeds was not one, there was one model that performed better than all of the others. The SVC Pythagorean Win Percentage model's precision was 0.934.





When the seed difference was equal to one, the Random Forest Four Factor model performed the best.





Combine Model for ==1

In an attempt to increase precision for series where the seed difference is equal to one, a voting model was created that used each model's result as a vote for a series win or loss. This voting model was not able to top the RFC Four Factor model's precision so it was not used.

Conclusion

All model testing was performed using the same subset of our original data. One fifth of the data was withheld from the model creation stage. This data was used to test our final

The model chosen to select the winner of a NBA playoff series is a hybrid model. If the seed difference of the two teams is equal to one, the Random Forest Four Factor Model is used. If the seed difference is greater than one, the Support Vector Machine Classifier Pythagorean Win Percentage model is used. By choosing the appropriate model for the situation we are able to do a better job predicting the result of NBA playoff series. These results are much more accurate compared to simply choosing the team with the lower seed.

The hyperparameters for the Support Vector Machine Classifier that were altered were the **C** value and the γ value. The **C** value determines the width of the boundary. An increased width will lead to more margin violations. By altering the **C** a balance between keeping the margin as large as possible while producing the least amount of margin violations. The γ value acts as a regularization hyperparameter, by decreasing the value you can avoid overfitting.

The hyperparameters of the Random Forest Classifier that were changed from scikit-learn's default parameters were the number of trees in the forest. Since this model used the four offensive and defensive factors for each team, only the 5 most significant features were selected:

- Effective Field Goal Percentage
- Turnover Percentage
- Free Throw Rate
- Defensive Rebounding Percentage
- Opponent's Offensive Rebounding Percentage

The basic model's $precision_{bm} = 0.7479$, meaning that the higher seed wins about 75% of NBA playoff series (using the test data set produced the same precision). Using only the test data training set, the Support Vector Machine Classifier Pythagorean Win Percentage model for series where the seed difference was greater than one, $precision_{svc-pyt-g1} = 0.9772$. For series where the seed difference is greater than one, our SVC model was able to correctly predict the winner of the series 43 out of 44 times. Separately, for series where the seed difference is equal to one, the Random Forest Four Factor model $precision_{rfc-4fact-eq} = 0.3077$.

The Random Forest Four Factor model did not perform as well on the test data compared to the training data. This is where the strategy of relying on precision is not wise. Of the 476 playoff series, only 115 were series where the seed difference was 1. 20% of these observations were reserved for the test data set. The test data only predicted 13 of 33 series would result in a win (4 were correct).

The combined final model $precision_{combined} = 0.8246$. This final score of precision is greater than the basic model. Using this combined approach, and only focusing upon when the combined model produces a positive prediction (the team will win the series), it is possible to correctly predict the winner of an NBA playoff series 82% of the time.

Future Considerations

This project was able to produce an accurate model for predicting NBA playoff series in terms of the models' precision score. Since the basic model, just choosing the higher seed to win will never predict a team to lose, precision is the most appropriate scoring technique to consider. For future endeavors, other model scoring techniques should be implemented. The final model only produced win predictions on approximately 50% of the games. Future models should be compared with one another using their F1 scores. This way recall and precision will both play a part in determining how well the models performed.

1998-1999 Season

The 1998-1999 NBA season was a strike shortened season. Each team played only 50 games instead of the usual 82. This season is the only season in our data set (NBA seasons from 1983 to 2017) where a number 8 seeded team made it to the NBA Finals. Only 2 teams seeded 5 or lower have made it all the way to the NBA Finals in this data set. Models should be created without the 1998-1999 season to see if model performance increases. The 1998-1999 season should be considered an outlier because of the shortened schedule and surprising playoff results.

2005-2006 Season

Another outlier that could be excluded from the data set are some of the series from the Western Conference playoffs in the 2005-2006 season. In the 2004-2005 season, each NBA conference expanded from 2 divisions to 3 divisions. Before the change occurred, division winners were guaranteed the top 2 seeds in each conference. If the best 2 teams in a conference were in the same division, these teams would be seeded number 1 and 3 and would be guaranteed not to meet until the conference finals.

When the switch to 3 divisions occurred the rule that division winners were guaranteed the top seeds was not changed. In the 2005-2006 season, the Denver Nuggets had the 7th best record in the Western Conference but were the 3 seed because they were the best team in their division. The Dallas Mavericks, who had the 2nd best record in the conference, were relegated to the 4th seed because they did not have the best record in their division. This lead to the two best teams in the conference meeting in the conference semifinals instead of a possible confrontation in the conference finals.

2017-2018 NBA Playoffs

The current NBA season was not part of the data set. The 2017-2018 NBA playoffs began in April. Currently, the second round is underway. This year, more so when compared to years past, the playoffs are expected to produce unexpected results. Some of the highest seeded teams are missing star player because of injuries. Lebron James, whose team has appeared in the last 7 NBA Finals, is on a number 4 seeded team.

References

scikit-learn

R Packages

Hadley Wickham, Romain Francois, Lionel Henry and Kirill Müller (2017). dplyr: A Grammar of Data Manipulation. R package version 0.7.4. https://CRAN.R-project.org/package=dplyr

H. Wickham. ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York, 2009.

Jeffrey B. Arnold (2017). ggthemes: Extra Themes, Scales and Geoms for 'ggplot2'. R package version 3.4.0. https://CRAN.R-project.org/package=ggthemes

Hadley Wickham (2016). rvest: Easily Harvest (Scrape) Web Pages. R package version 0.3.2. https://CRAN.R-project.org/package=rvest

Hadley Wickham (2017). scales: Scale Functions for Visualization. R package version 0.5.0. https://CRAN.R-project.org/package=scales

Hadley Wickham (2018). stringr: Simple, Consistent Wrappers for Common String Operations. R package version 1.3.0. https://CRAN.R-project.org/package=stringr

Papers and Books

Ben-Hur, Asa, and Jason Weston. 2017. “A User’s Guide to Support Vector Machines.” http://pages.cs.wisc.edu/~yliang/cs760_fall17/howtoSVM.pdf.

Caruana, Rich, and Alexandru Niculescu-Mizil. 2006. “An Empirical Comparison of Supervised Learning Algorithms.” Proceedings of the 23rd International Conference on Machine Learning (ICML). http://www.cs.cornell.edu/~caruana/ctp/ct.papers/caruana.icml06.pdf.

Geron, Aurelien. 2017. “Hands-on Machine Learning with Scikit-Learn & Tensorflow.” O’Reilly.

He, Haibo, and Edwardo A. Garcia. 2009. “Learning from Imbalanced Data.” Transactions on Knowledge; Data Engineering. http://www.ele.uri.edu/faculty/he/PDFfiles/ImbalancedLearning.pdf.

Lee, Young Hoon, and David Berri. 2008. “A Re-Examinatin of Production Functions and Efficiency Estimates for the National Basketball Association.” Scottish Journal of Political Economy. http://daveberri.weebly.com/uploads/6/1/3/8/61387427/2008hoonleeberrisjpe.pdf.

NBA-reference.com. 2018. “NBA-Reference.com.” https://www.basketball-reference.com/leagues/NBA_stats.html.

Oliver, Dean. 2004. *Basketball on Paper.* Potomac Books.

Winston, Wayne. 2012. *Mathletics.* Princeton University Press.