INDENG 142: Ranking NBA Players

Adit Roychowdhury, Augustin Kim, Ye Joon Han, Phuc Pham, Cosmin Deshmukh, Paul Loisel

Professor Stewart Liu

Motivation

We are trying to predict what qualifies an NBA player as an MVP (Most Valuable Player) because it is essential for front office members to make the proper decisions and trades for their team to win NBA games and ultimately championships. This information is not only important to franchises' but also to the players themselves. Every single NBA MVP there has been has made it to the hall of fame (must be retired for four years to qualify). Being in contention to win the award, but never actually winning it can be the difference between being a great player or being a hall of fame player. Since basketball is a team sport, discourse surrounding the legacies of players is heavily dictated by the amount of MVP awards a player has won. Philadelphia 76ers center Joel Embiid, for instance, has finished second in voting for the last 4 years, when asked about the prospect of winning his first MVP award he said " [If I don't win] I don't know what I have to do. I'll feel like they hate me (talking about MVP voters)." This project is perfect for a player like Joel Embiid. He thinks he should have won it, yet he can't put his finger on any particular reason why he hasn't. This project could shed light on potential holes in his game that may be perturbing voters from fully backing him.

Data

The data we used was collected as a result of web scraping where we used NBA statistics from the website: https://www.basketball-reference.com/ and turned the NBA player statistics into csv files. The data used were from NBA statistics dating from 1991 to 2021. The main datasets we used were **teams.csv, mvp.csv** and **players.csv.**

Players.csv is a dataset containing the names of all the players from the NBA 1991 to 2021 seasons along with their key metrics such as Age, Position, Team, and other statistics such as Points, Field Goal Averages, Blocks, Year they played, ECT.

Teams.csv contains data about each NBA Team from 1991 to 2021 and their statistics such as Win/Loss ratios and and their ratings.

Mvp.csv contains data about previous MVPs along with statistics about their rankings, points and the proportion of MVP votes they received ("Share").

EDA/Data Cleaning and Preprocessing

We then used Panda to convert these csv files into data frames. The data was then cleaned as follows:

players.csv: Unnamed columns and the 'Rk' column were deleted. Any player name containing an asterisk was replaced with the name without the asterisk. Rows where the team column equals 'TOT' were consolidated into a single row because the 'TOT' column refers to a player that belonged to Two Other Teams I.E. if a player was in the Lakers in 1992 and they were in the Golden State Warriors in 2005.

mvp.csv: The mvp.csv file did not contain much data to clean or process so decided to keep the relevant columns/features: Player, Year, Pts Won, Pts Max, Share, and Rank.

teams.csv: Unnamed columns were deleted, and any team name containing an asterisk was replaced with the name without the asterisk. Rows containing 'Division' in the 'W' column were removed.

After cleaning, the players.csv and mvp.csv datasets were merged to create a single dataset that includes each MVP winner and their corresponding player statistics for each year they won the award. A dictionary mapping team abbreviations to team names was created from the **ab.csv** file and used to replace abbreviations with team names in the dataset. Any missing values were replaced with 0. The dataset was then merged with the **teams.csv** dataset to add additional team statistics for each year. Any remaining missing values were removed.

Analytics models

Linear regression

We created our Linear Regression model by using the feature “Share” as the predictor because Share is the proportion of votes a player receives from officials to be determined as a NBA MVP, therefore the highest “Share” number determines the MVP.

After doing further analysis on the features, we observed that there are high correlations between the following features: 'Age', 'G', 'GS', 'MP', 'FG', 'FGA', 'FTper', 'ThreeP', 'ThreePA', 'ThreePper', 'TwoP', 'TwoPA', 'TwoPper', 'FT', 'FTA', 'FTper', 'ORB', 'DRB', 'TRB', 'AST', 'STL', 'BLK', 'TOV', 'PF', 'PTS', 'Year', 'W', 'L', 'GB', 'SRS' to our prediction feature (“Share”).

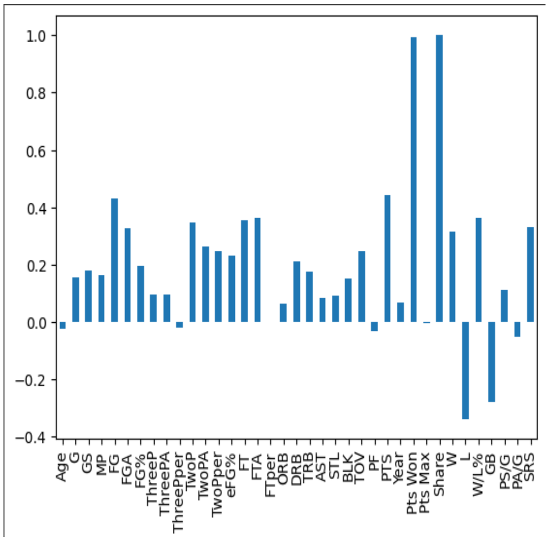

Figure 1: Bar chart displaying correlation of all features to Share

Using VIF for feature selection we observed that there were high amounts of multicollinearity between the following features: "FG", "FTper", "ThreeP", "ThreePper", ”TwoP”, “TwoPper”, “FT”, “FTper” , “ORB”, “DRB”, “TRB” , “W/L%”, “PS/G”, “PA/G”, and “eFG%”.

After dropping the features that had high amounts of multicollinearity we observe that our Model **Accuracy for Linear Regression is 0.0**. Our Model Accuracy suggests that the model is not able to capture any meaningful patterns in the data, or that the model is severely overfitting the training data. Our model is overfitting because it is too complex and fits the noise in the training data, rather than the underlying patterns, therefore, Linear Regression is not reliable when determining a NBA MVP.

Linear Regression with Multi-Layer Perceptron (MLP)

In order to fight this complexity, avoid overfitting and fitting the noise on the training data, we can try to use Linear regression but with neural network technique. We build a Multi-Layer Perceptron (MLP) which is feedforward neural network. In our case, we give it one hidden layer between the input and output layers. We chose to use 64 neurons in the hidden layer as a balance between model complexity and computational efficiency. Having more neurons can help the model capture more intricate patterns in the data, but too many neurons may lead to overfitting and increased training time. In our case, the choice of 64 neurons provided a good trade-off, resulting in a satisfactory model performance. The output layer works with a sigmoid activation function for binary classification (MVP or not) The sigmoid activation function is well-suited for binary classification problems because it transforms the input values into probabilities, ranging from 0 to 1. This property allows us to interpret the output as the likelihood of a player being an MVP. Moreover, the sigmoid function is smooth and differentiable, making it suitable for gradient-based optimization algorithms commonly used in training neural networks. In order to change the 'Share' into discrete variable instead of continuous we create a threshold saying that at 50% of the vote, the player is a MVP. The confusion matrix give the following result : 114 TP, 10TN, 7FP, 6FN. The **accuracy of the model is 0.9051** and the ROC curve is the following:

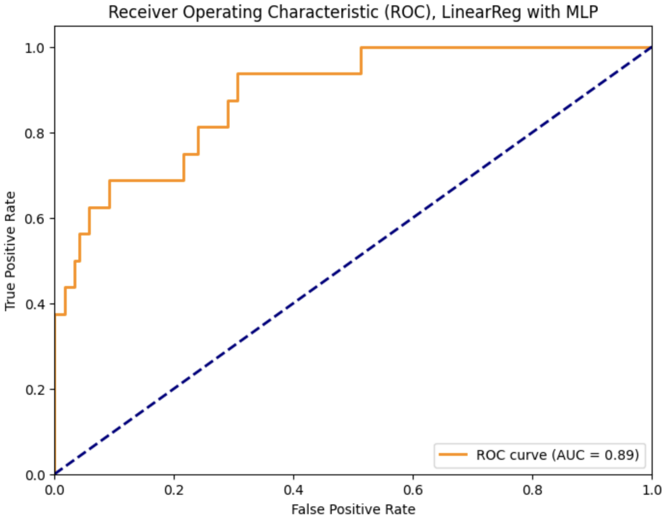

Figure 2:ROC Curve for LinReg with MLP

Logistic regression

Because we wanted to predict whether the player is MVP or not, which can be represented as a binary variable, we also utilized the logistic regression to predict the probability of the player being MVP. We first decided to predict the 'Share' variable with the model. Before building the model, we used VIF for feature selections. We found that the values of VIF are very huge for features including FG, FGA, FG%, 3P, 3PA, 2P, 2PA, 2P%, eFG%, FT, FTA, FT%, ORB, DRB, TRB, PTS, W, L, W/L%, PS/G, PA/G, SRS. Thus, after removing the features with large VIF, we decided to use Age, GS, MP, 3P%, eFG%, FTA, FT%, ORB, AST, STL, BLK, TOV, PF, Year, W, L, and GB for the model. After fitting the logistic model, we found that the p values for the coefficients of Age, GS, 3P%, eFG%, FT%, AST, STL, BLK, TOV, Year, W, and GB are not statistically significant. Thus, we decided to remove the features with large p value for the coefficient from the model, and as a result, we ended up using MP, eFG%, FTA, FT%, ORB, AST, PF, and L. After predicting the probabilities, we decided to use Bayes Optimal Classifier, so we used 0.5 as a cutoff, which means that the player is considered MVP if the predicted probability for the player is above 0.5. Moreover, we also observed that the values of 'Share' for MVP players are often above 0.8. Thus, we created binary variable called 'Is_MVP' using 0.8 as a cutoff so that we can use this feature to compare with our predictions. We also built the confusion matrix, and there were 126 True Negatives, 3 False Postives, 5 False Negatives, and 3 True Positives, and the overall **accuracy of the model was about 0.9416**.

K-fold Cross Validation with a Decision Tree Regressor

We first performed hyperparameter tuning using grid search cross-validation to find the best set of hyperparameters for a decision tree regression model. Then we created a DataFrame that shows the validation accuracy of the decision tree regression model for different values of the ccp_alpha hyperparameter. We then visualize these results by showing a scatter plot of the validation accuracy (y-axis) of the decision tree regression model for different values of the ccp_alpha hyperparameter (x-axis).

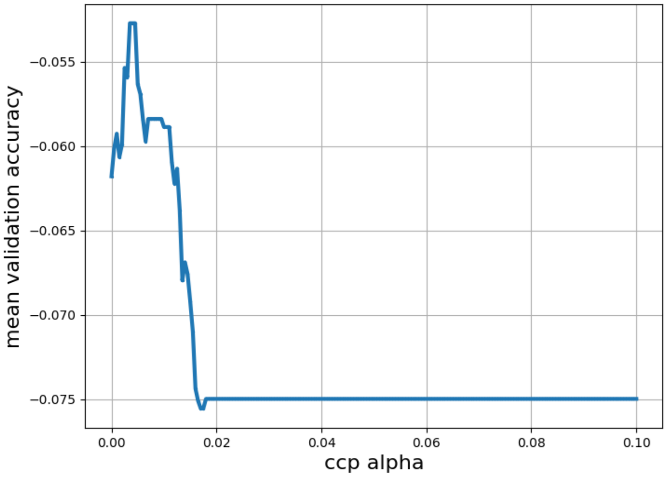

Figure 3: Validation Accuracy vs CCP Alpha

Lastly, we partake in the final steps of training the decision tree regression model with the best hyperparameters found during the hyperparameter search using GridSearchCV, and computed the accuracy on the test-set. Unfortunately, the accuracy was very low so we probably won't include this model when using the project in the real world.

Random Forest

Because we determined that Linear Regression could not be used to predict NBA MVP's, we decided to use Random Forest Algorithm because it is a supervised machine learning algorithm that can be used for Classification and can help solve more complex problems that could not be solved by Linear Regression.

In our model, we decided to train on the NBA statistics of all the years prior to 2021 while our test set would be the year 2021.

After fitting our model in the Random Forest Algorithm we can observed that there are slight differences between the rankings of NBA MVPs. Nikola Jokic was the actual NBA MVP for the year 2021 while our model predicted that Giannis Antetokounmpo would be NBA MVP. We can also observe that our model swapped Giannis's placing (4th) and Nikola's placing (1st). While our model did not correctly predict the

NBA MVP for 2021, our model **accuracy was 90% correct on average** on determining rankings for NBA MVP.

Impact

Our work has the potential to provide valuable insights to team managers and recruiters in the NBA industry. By predicting what qualifies a player as an MVP, our work can help teams make informed decisions and trades to improve their chances of winning games and championships. Additionally, our work can contribute to the broader understanding of what makes a successful NBA player and potentially inform training and development programs for aspiring players.

To improve the impact of our analysis, we might consider incorporating additional data sources and variables. For example, we could include data on player salaries, injuries, and off-court behavior to determine if these factors impact a player's MVP candidacy. Additionally, we might expand our analysis to include players who were not MVP winners but had successful seasons, to determine if there are other factors beyond MVP status that contribute to a player's success.

One possible negative consequence of our model is that it could perpetuate biases or reinforce stereotypes about what makes a successful NBA player. For example, if our model consistently identifies players from certain positions or with certain physical attributes as MVP candidates, this could reinforce existing biases and make it harder for players outside of those categories to be recognized for their contributions. Additionally, our model could potentially be misused by teams or recruiters to make decisions based solely on MVP status, without considering other important factors like team dynamics or a player's character.

Appendix

Our code, along with detailed results and graphs is attached to the bottom of this document, as well as submitted on bCourses