HarvardX Capstone: Predicting the Winner of NBA's 2020 Most Improved Player Award, and Finding Candidates for 2021

Sumitro Datta

May 25, 2020

Introduction

As the beginning of a new NBA season nears, predictions start to roll in from pundits. In terms of major player awards, the Most Improved Player is the most "predicted by gut feeling".

- there are usually a select group of players that have a chance at Most Valuable Player and Defensive Player of the Year, with minor turnover year-over-year
- Sixth Man of the Year (given to the best performing player that does not start the game on the court) has morphed into a type: the high scoring guard that ironically often places in the top 5 in minutes played on the team
  - a guard has won 13 of the past 14 years, and has placed in the top 5 in minutes played on the team 12 of the 14 years
- Rookie of the Year, by definition, has turnover every year, but predictions tend towards players taken near the beginning of the draft

The goal of this analysis is to use machine learning to make a two-fold prediction.

1. Predict the winner of this year's (2020) MIP award.
2. Predict candidates of next year's (2021) MIP award.

Steps that were performed:

1. loading the data and swapping NA's with 0's
2. visualizing how MIP winners have performed and how much vote share they received
3. create a version of the data that shows statistical jumps (current season minus previous season)
4. separating out evaluation sets (the 2020 season) and test sets (the 1987 season for predicting the 2020 winner, the 1986 and 2019 seasons for predicting 2021 candidates)
5. training five models: a linear baseline, a k-nearest-neighbors model, a decision tree model, and two separate random forest models

Note: to simplify the Season column, I refer to each season by its second half. So you might also hear the current season referred to as the 2019-2020 season.

Methods/Analysis

Loading Packages

Let's start by loading required packages.

```
if(!require(tidyverse))
  install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(caret))
  install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(readxl))
  install.packages("readxl", repos = "http://cran.us.r-project.org")
#Rborist and ranger are random forest algorithm wrappers
if(!require(Rborist))
  install.packages("Rborist", repos = "http://cran.us.r-project.org")
if(!require(ranger))
  install.packages("ranger", repos = "http://cran.us.r-project.org")
#matrix stats
if(!require(matrixStats))
  install.packages("matrixStats", repos = "http://cran.us.r-project.org")
#rpart.plot shows the decision tree of an rpart result
if(!require(rpart.plot))
  install.packages("rpart.plot", repos = "http://cran.us.r-project.org")
#kableExtra allows more customization of tables
if(!require(kableExtra))
  install.packages("kableExtra")
if(!require(RColorBrewer))
  install.packages("RColorBrewer", repos = "http://cran.us.r-project.org")
```

Downloading the Data

This is a dataset that I have compiled from Basketball-Reference. The "NBA Season Stats 1984 to 2020" Excel Workbook contains 155595 individual player seasons starting from 1984-1985 (one year before the inaugural awarding of the MIP award) to 2019-2020. Each player season contains some identifying factors:

- a unique Season ID
- the player's name
- the player's age as of February 1st of that season (per Basketball-Reference recordkeeping)
- the team that the player played for (if they played for multiple teams, it was recorded as TOT)
- the actual season
- the birth year
  - this is to break ties, as there are instances of 2 players having the same name and playing in the same season as well as father and son playing in different eras under the same name

The workbook contains five sheets:

- the first sheet contains per game stats
- the second sheet contains stats per 36 minutes
  - this just scales up bench player production to see what they might average with a starting player's minutes
- the third sheet contains stats per 100 possessions
  - this adjusts for pace, which is the number of possessions teams use in 48 minutes (the typical length of a game, excluding overtimes)
  - this is necessary because the late 1990s/early 2000s were notoriously slow, so raw statistics were depressed
- the fourth sheet contains advanced stats

  - for more information, refer to this *Basketball-Reference glossary*
- the fifth sheet combines all the above into one sheet

The MIP Vote Share Excel Workbook also contains the 15595 individual player seasons from 1984-1985 to 2019-2020. However the data in this workbook keeps track of the share of an MIP award that a player received. MIP share equals the amount of points received divided by the maximum number of points possible. Up to the 2002 season, voters were given one vote, and the player with the highest number of votes won the award. Starting in 2003, a point system was introduced, where:

- a first-place vote equals five points
- a second-place vote equals three points
- a third-place vote equals one point
- the player with the most ***points*** won the award

Theoretically, it is possible to have the most first-place votes and not win the award, but it has never happened in the history of the award.

MIP Vote Share is tracked for both the season in which the vote was received, as well as the season prior. The latter column will help identify candidates for next season. For example, Pascal Siakam of the Toronto Raptors won the MIP award in 2019 with an MIP vote share of 0.938. This would be recorded in Siakam's 2019 season under `MIP Share Current Season` and in Siakam's 2018 Season under `MIP Share Next Season`.

```
#divide urls into chunks to get pdf to wrap long urls
urlRemote<-"https://raw.github.com/sumitrodatta/most-improved-player/master/"
statsfile<-"NBA%20Season%20Stats%201984%20to%202020.xlsx"
mipsharefile<-"MIP%20Vote%20Share.xlsx"
tmp<-tempfile()
download.file(paste0(urlRemote,statsfile),tmp)
combined<-read_xlsx(tmp,sheet="Combined")
tmp2<-tempfile()
download.file(paste0(urlRemote,mipsharefile),tmp2)
mip_share<-read_xlsx(tmp2,sheet="Sheet1")
file.remove(tmp)
file.remove(tmp2)
rm(tmp,tmp2)
```

Let's see if there is any missing data.

```
sum(is.na(combined))
```

```
## [1] 15267
```

```
sum(is.na(mip_share))
```

```
## [1] 29738
```

Looks like all the players who never received an MIP vote have an NA, as well as all the players who don't need a birth year to separate them from another player with the same name. Let's convert those NAs to 0s, and then append the mip vote share data to the statistical data.

```
combined[is.na(combined)]<-0
mip_share[is.na(mip_share)]<-0
combined<-left_join(combined,mip_share)
rm(mip_share) #to clean up environment (all relevant info in combined)
```

Data Exploration and Visualizations

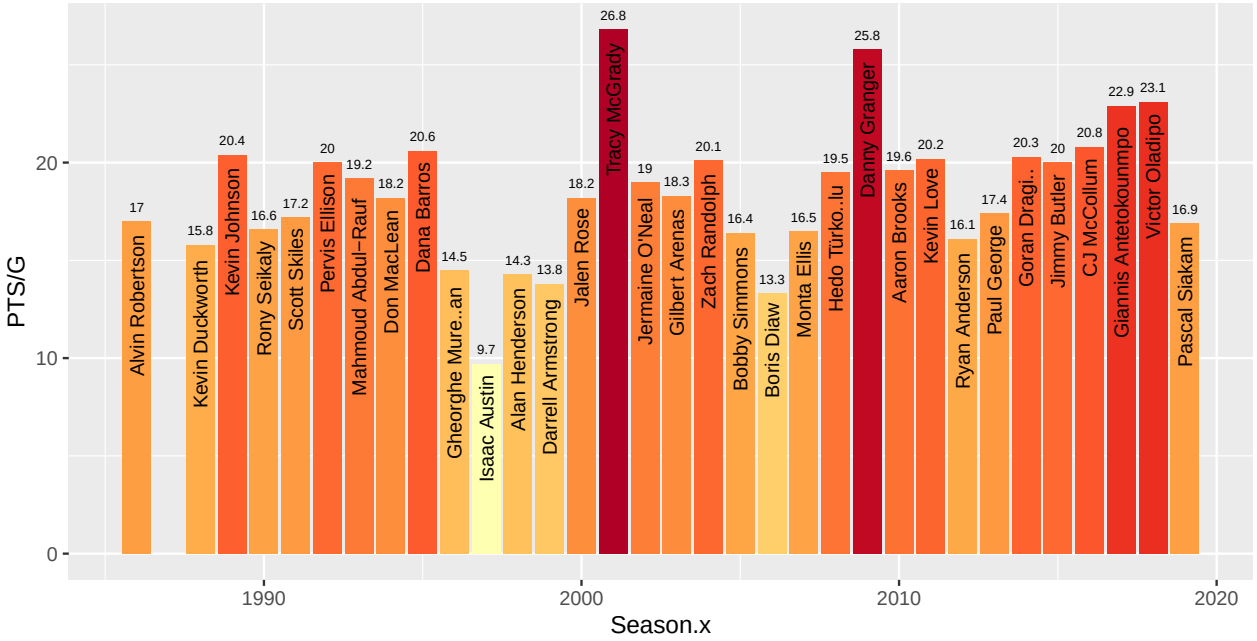
We'll start by seeing how much vote share MIP award winners have received.

Season	player	vote_share
1985	Kareem Abdul-Jabbar	0.0000000
1986	Alvin Robertson	0.3846154
1987	Kareem Abdul-Jabbar	0.0000000
1988	Kevin Duckworth	0.4125000
1989	Kevin Johnson	0.5647059
1990	Rony Seikaly	0.4021739
1991	Scott Skiles	0.2604167
1992	Pervis Ellison	0.4166667
1993	Mahmoud Abdul-Rauf	0.2551020
1994	Don MacLean	0.5445545
1995	Dana Barros	0.4761905
1996	Gheorghe Mureșan	0.4424779
1997	Isaac Austin	0.3565217
1998	Alan Henderson	0.2844828
1999	Darrell Armstrong	0.4576271
2000	Jalen Rose	0.2644628
2001	Tracy McGrady	0.5967742
2002	Jermaine O'Neal	0.4126984
2003	Gilbert Arenas	0.4881356
2004	Zach Randolph	0.6264463
2005	Bobby Simmons	0.6243902
2006	Boris Diaw	0.7761905
2007	Monta Ellis	0.5457364
2008	Hedo Türkoğlu	0.6080000
2009	Danny Granger	0.6016529
2010	Aaron Brooks	0.6552846
2011	Kevin Love	0.6896552
2012	Ryan Anderson	0.4297521
2013	Paul George	0.5183333
2014	Goran Dragić	0.6476190
2015	Jimmy Butler	0.8294574
2016	CJ McCollum	0.8600000
2017	Giannis Antetokounmpo	0.8560000
2018	Victor Oladipo	0.9881188
2019	Pascal Siakam	0.9380000
2020	Steven Adams	0.0000000

There are three seasons that have a vote share of zero: 1985 (which was the season before the first awarding of the MIP), 2020 (which is the season we are trying to predict for) and 1987.

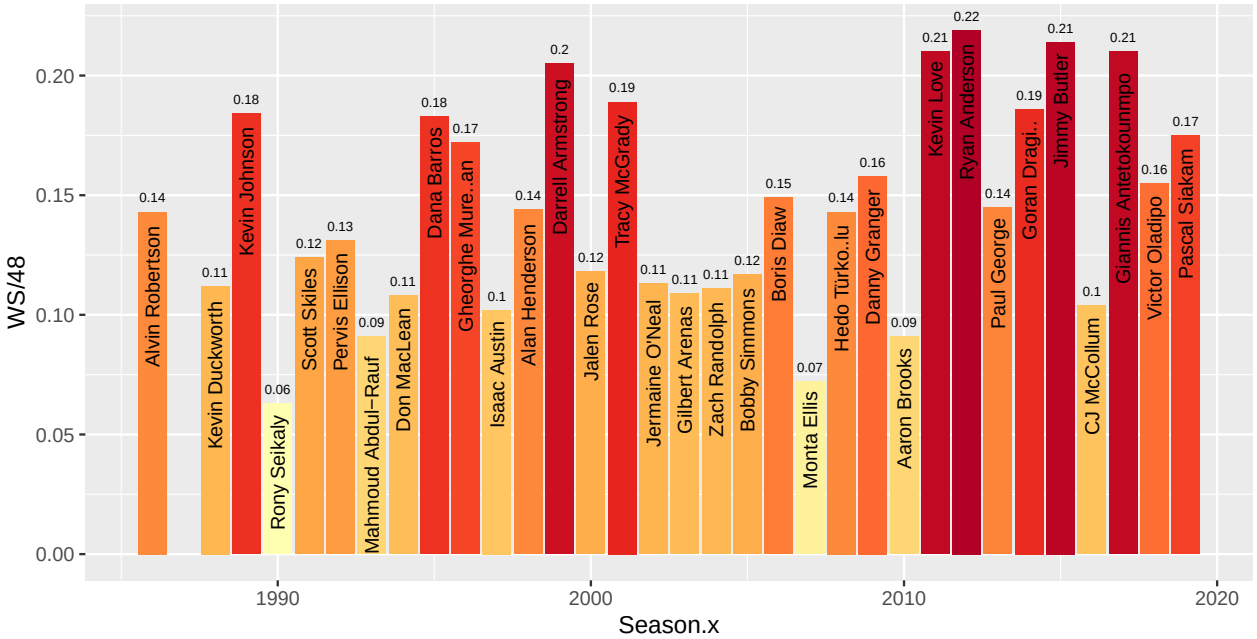
There was an MIP award given out in 1987 to Dale Ellis of the Seattle SuperSonics. However, when researching, I could not find any voting records for the season. I'll discuss what I plan to do with this season a little further in the report.

Secondly, we'll plot the scoring per game of the MIP winners. The three missing seasons (1985,1987,2020) have been removed.



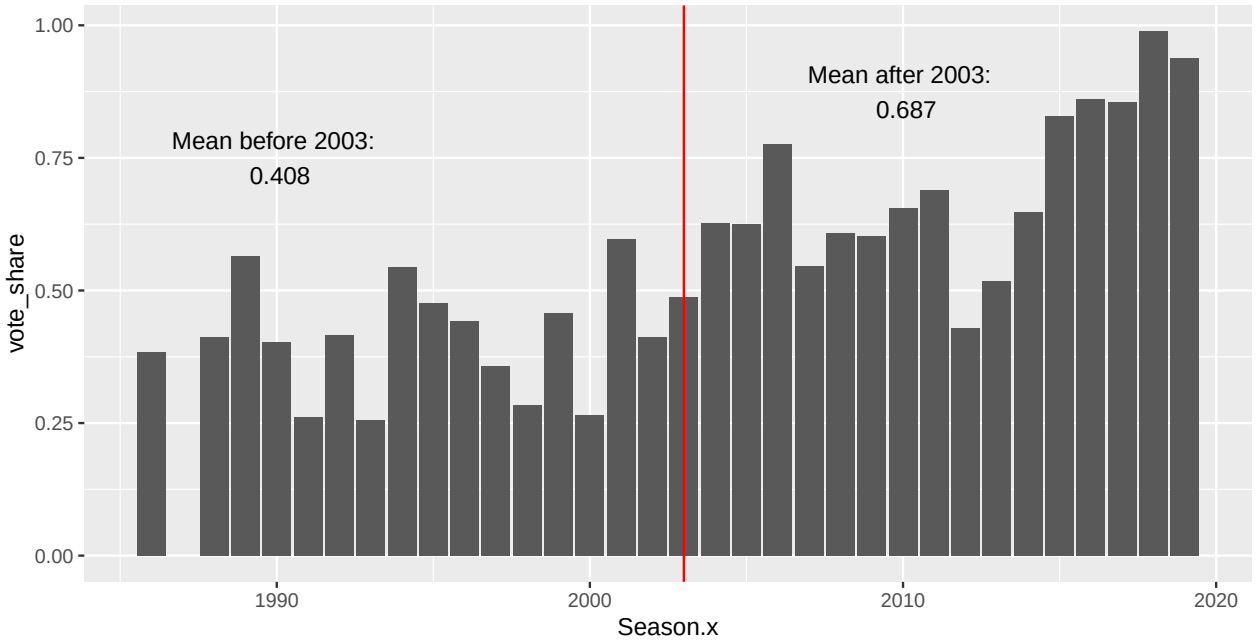
The lowest points per game average by an MIP is 9.7 (achieved by Isaac Austin in 1997), while the highest is 26.8 (achieved by Tracy McGrady in 2001).

Next, we'll plot the MIP winners by Win Shares per 48 minutes. Win Shares represent how much a player has contributed to his team's wins by comparing his output to a marginal player. Win Shares per 48 Minutes is a rate stat to compare how effective a player was during their time on the court, regardless of the actual minutes played.



The lowest win shares per 48 minutes by an MIP is 0.06 (achieved by Rony Seikaly in 1990), while the highest is 0.22 (achieved by Ryan Anderson in 2012).

Finally, we'll plot how the max voting share has evolved over the life of the award, placing a vertical line at the 2003 season when the voting system changed.



Looks like the average jumps up significantly after the voting system was changed. In recent years, there's been more consensus.

Preprocessing

Before we fit our model, we should do some pre-processing. We've already done some pre-processing by handling missing values.

For the 2020 award winner, we realize that players must have at least one prior season played in order to garner consideration. So we will subset out all players who have only one season played in the database. Removing one-season players has the unfortunate side effect on removing some players who played their last games in 1985, but had a career before then. This shouldn't be too worrisome, as no MIP has won in their final season in the league.

In addition, statistical jumps from the previous season play a much bigger part than the current season's statistics in determining the winner. A player going from 20 points per game to 21 points per game will not merit as much campaigning as a player going from 8 PPG to 18 PPG.

Finally, we will also remove identifying factors from the data because we don't want to predict based off of a player's name or team that they played on.

```
one_seasoners<-combined %>% group_by (Player,BirthYear) %>%
  mutate(n=n()) %>% filter(n==1) %>% select(SeasID,Player,BirthYear)
full_jumps_data<-combined %>%
  #remove the one-season players
  filter(!(SeasID %in% one_seasoners$SeasID)) %>%
  group_by(Player,BirthYear) %>%
  #arrange seasons from first to last within each player
  arrange(Season,.by_group=TRUE) %>%
  #subtract prior season from current season
  mutate_at(.vars=c(colnames(combined[8:88])),
            .funs=funs(`diff`=.-lag(.,default=first(.)))) %>%
```

```
  #do not keep the raw stats, only differences
  select(SeasID:BirthYear, G_diff:VORP_diff,
         `MIP Share Next Season`:`MIP Share Current Season`) %>%
  #remove first season from every group (no difference)
  slice(-1) %>% arrange(SeasID) %>% ungroup() %>%
  #do not keep identifying information
  select(SeasID,Age,Season,G_diff:`MIP Share Current Season`)
```

For predicting candidates in 2021, it's a much more diverse subset. So we'll just remove identifying factors from the data.

```
full_data<-combined
full_data<-full_data %>% select(-c(Player,Tm,Lg,BirthYear))
```

Now, let's check for variables that are near zero variance.

```
nzv<-nearZeroVar(full_jumps_data,saveMetrics = TRUE)
rownames(nzv)[which(nzv$nzv==TRUE)]
```

```
## [1] "MIP Share Next Season"    "MIP Share Current Season"
```

Looks like the only columns with near zero variance are our outcome columns. So there's no columns removed here.

Creating an Evaluation Set

Our evaluation set is straightforward: it is players who have played in the 2020 season. So we segment it off here.

```
eval_2020<-full_data %>% filter (Season==2020)
full_data<-full_data %>% filter (Season != 2020)
eval_jumps_2020<-full_jumps_data %>% filter (Season ==2020)
full_jumps_data<-full_jumps_data %>% filter (Season !=2020)
```

As discussed earlier, I could not find any voting records for the 1987 season. All I could find was the winner (Dale Ellis of the Seattle SuperSonics). Instead of removing the 1987 data, I realize that I could use it as a test set or additional performance metric before applying the algorithm to find the 2020 season winner. Ideally, the algorithm would project Ellis to be the winner, and also provide me with the other candidates from that season. In the same vein, I'll remove the 1986 season and 2019 season to predict the 2021 candidates.

```
eval_1986<-full_data %>% filter (Season==1986)
eval_2019<-full_data %>% filter (Season==2019)
full_data<-full_data %>% filter(Season !=1986 & Season !=2019)
eval_jumps_1987<-full_jumps_data %>% filter (Season ==1987)
full_jumps_data<-full_jumps_data %>% filter (Season !=1987)
```

Training Models

Models to Predict the 2020 Winner

We're not going to create a training set, because the built in train functions can perform cross validation (we'll be using 10 folds). We'll evaluate performance by checking RMSE, the predicted vote share of Dale Ellis in the 1987 test set and the top 3 candidates in 1987. Remember, Ellis won the 1987 MIP but there were no voting records found during research. Five models will be tested.

A quick note: we'll be excluding the games difference and games started difference (G_diff and GS_diff respectively) as predictors because the NBA has been unable to complete an 82-game slate for the 2020 season. By including these predictors, we can only lower our estimate.

First up is a linear model. It'll serve as a baseline.

```
linear<-train(`MIP Share Current Season`~.-Season-`MIP Share Next Season`-
                SeasID-G_diff-GS_diff, data=full_jumps_data,method="lm",
              trControl=trainControl(method="cv",number=10,p=0.9))
```

Let's get the top 5 most important variables of the linear model.

```
##                        Overall               Vars
## `\\`MP/G_diff\\``  100.00000  `\\`MP/G_diff\\``
## `\\`AST/G_diff\\``  65.54722 `\\`AST/G_diff\\``
## `\\`TOV/G_diff\\``  61.82795 `\\`TOV/G_diff\\``
## `\\`PF/G_diff\\``   60.86471  `\\`PF/G_diff\\``
## `\\`WS/48_diff\\``  51.62175 `\\`WS/48_diff\\``
```

While 3 of the top 5 variables make sense (minutes per game, assists per game and WS/48), the high importance of turnovers per game and personal fouls per game is curious. These are stats where a lower amount is considered better.

Let's see how the linear model performs in 1987.

Method	RMSE	Ellis Vote Share	Top 3 Candidates	Top 3 Vote Shares
Linear	0.0343344	0.0608291	Michael Jordan	0.0778186
Linear	0.0343344	0.0608291	Dale Ellis	0.0608291
Linear	0.0343344	0.0608291	Terry Porter	0.0389028

The vote shares are quite low. Ellis places second behind Michael Jordan. This is somewhat misleading, because Jordan broke his foot the previous year and missed 64 games. He came back with a vengeance, upping his points per game by 14.4. But Jordan was already a star, and stars are implicitly excluded from winning MIP.

Next up is the k-nearest neighbors model. The intuition behind using this model is that maybe Most Improved Player winners are most similar to other winners. We can tune the parameter k to change how many neighbors to compare to. We check values of k from 5 to 50.

```
knn<-train(`MIP Share Next Season`~.-Season-`MIP Share Current Season`-
             G_diff-GS_diff-SeasID, data=full_jumps_data,method="knn",
           tuneGrid=data.frame(k=seq(5,50)),
           trControl=trainControl(method="cv",number=10,p=0.9))
knn$finalModel
```

```
## 49-nearest neighbor regression model
```

Let's see how the knn model performs in 1987.

Method	RMSE	Ellis Vote Share	Top 3 Candidates	Top 3 Vote Shares
Linear	0.0343344	0.0608291	Michael Jordan	0.0778186
Linear	0.0343344	0.0608291	Dale Ellis	0.0608291
Linear	0.0343344	0.0608291	Terry Porter	0.0389028
KNN	0.0308246	0.0110901	Tim McCormick	0.0363522
KNN	0.0308246	0.0110901	Darnell Valentine	0.0354552
KNN	0.0308246	0.0110901	Tony Brown	0.0303645

While the RMSE has improved, the probabilities have significantly shrunk. The 3rd best probability in the linear model is greater than the best probability in the knn model. In addition, Dale Ellis drops out of the top 3. The stats of the KNN top 3 (Tim McCormick, Darnell Valentine and Tony Brown) don't jump off the page. In the cases of Brown and McCormick, their raw stats improved but their advanced stats actually suffered, implying that they simply received more opportunity but didn't do anything with it.

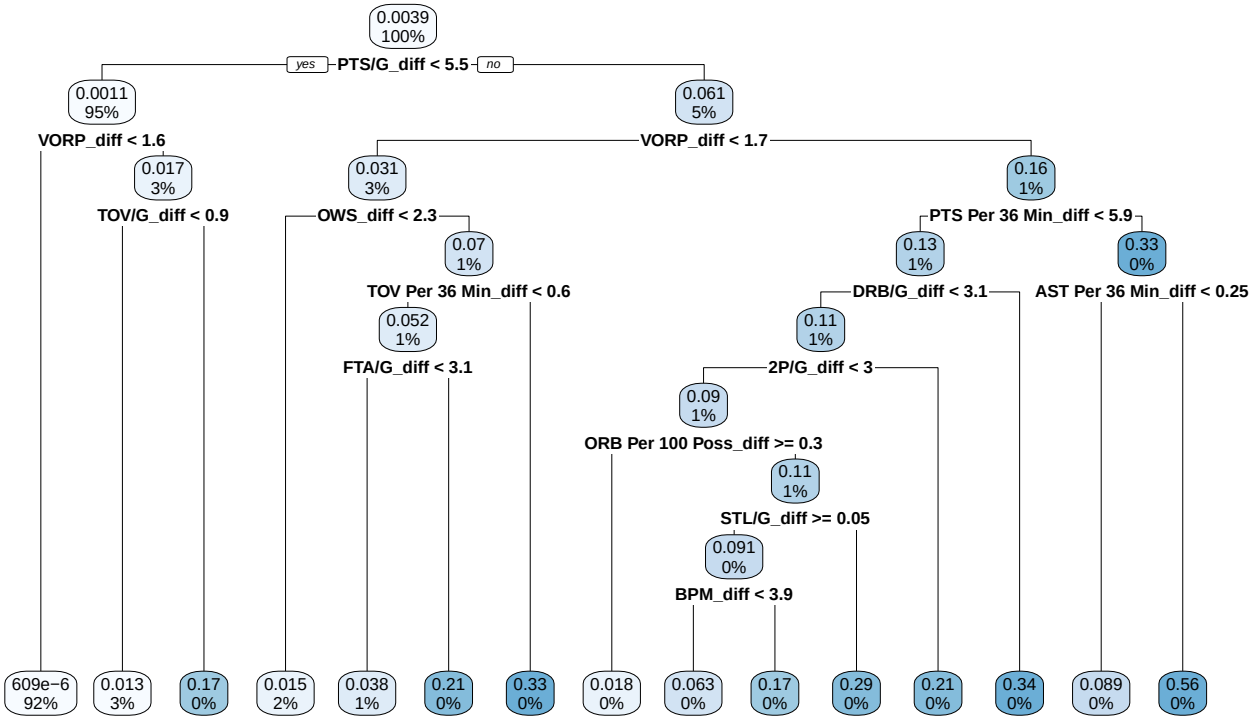
The next three models are based in decision trees. The first model returns a singular decision tree, while the latter two models are random forests which return the average of multiple decision trees. The intuition behind using models from the decision tree family is that maybe there are certain statistical thresholds that a potential MIP winner needs to meet in order to merit consideration. As a player passes more of these checkpoints, their vote share increases.

As mentioned, the first model is a single decision tree. We can tune the complexity parameter (cp) in this model. CP is the minimum for how much the residual sum of squares must improve for another partition to be added. A CP that is too high will have too few branches, while a CP that is too low will be difficult to follow since there are many branches. We lean towards the lower end of the spectrum. Since there exists an element of randomness in choosing samples to model a decision tree, we need to set a seed to keep the work reproducible.

```
set.seed(2,sample.kind = "Rounding")
rpart<-train(x=full_jumps_data[,!(names(full_jumps_data) %in%
                                    c("MIP Share Next Season",
                                      "MIP Share Current Season",
                                      "Season","SeasID"))],
             y=full_jumps_data$`MIP Share Current Season`,
             method="rpart", tuneGrid = data.frame(cp=seq(0,0.02,0.001)),
             trControl=trainControl(method="cv",number=10,p=0.9))
rpart$bestTune
```

```
##      cp
## 8 0.007
```

The model is optimized at cp = 0.007. Let's look at the decision tree.



The decision tree maximizes its prediction when a player does all of the following:

- increases their points per game by more than 5.5 points
- increases their VORP (Value Over Replacement Player) by more than 1.7
- increases their points per 36 minutes by more than 5.9 points
- increases their assists per 36 minutes by more than 0.25

The decision tree minimizes its prediction when a player does both of the following:

- increases their points per game by less than 5.5 points
- increases their VORP by less than 1.6

The fact that points per game difference is the first partition shows we're off on the right foot. It's the most straightforward way to tell whether a player has improved. The next partition is VORP, which is also a good sign. It's not enough that a player scores more points, they also have to be helping the team overall.

Let's see how the decision tree model performs in 1987.

Method	RMSE	Ellis Vote Share	Top 3 Candidates	Top 3 Vote Shares
Linear	0.0343344	0.0608291	Michael Jordan	0.0778186
Linear	0.0343344	0.0608291	Dale Ellis	0.0608291
Linear	0.0343344	0.0608291	Terry Porter	0.0389028
KNN	0.0308246	0.0110901	Tim McCormick	0.0363522
KNN	0.0308246	0.0110901	Darnell Valentine	0.0354552
KNN	0.0308246	0.0110901	Tony Brown	0.0303645
Decision Tree	0.0347075	0.5622985	Dale Ellis	0.5622985
Decision Tree	0.0347075	0.5622985	Michael Jordan	0.2144521
Decision Tree	0.0347075	0.5622985	Otis Thorpe	0.2144521

While the decision tree RMSE is larger than both the linear and knn RMSEs, the vote shares are significantly higher. In addition, this is the first model that predicts Dale Ellis as the winner. We've already discussed the shortcomings of modelling in regards to Jordan's 1987 season. Thorpe is a worthy candidate: he increased his scoring by nine points and WS/48 by 0.04 while decreasing his turnover percentage.

As mentioned, the next two models use a random forest. Random forests are better than decision trees in that they reduce instability by averaging multiple trees. Unfortunately, the cost is interpretability. There is no tree diagram that is representative of the decisions made unlike the previous model. We will be using `ranger` and `Rborist` rather than `randomForest` to decrease computational time. Since ranger and Rborist are *random* forest algorithms, we need to set a seed to keep the work reproducible.

Ranger creates 500 trees. Random forest algorithms require an explicit call for variable importance, so we'll ask for permutation importance. A simplified explanation for permutation importance is shuffling a predictor's values and seeing how much the error increases. As a predictor's importance increases, it is difficult for the rest of the model to compute accurate predictions without it. In addition, ranger has 3 tuning parameters:

- mtry is the number of variables to split at each decision node
  - the default is the rounded square root of the number of variables, which in this case would be `round(sqrt(80))`
- splitrule is the rule used to split a node
  - the default is to minimize variance
- min.node.size is the minimum number of observations needed to split a node
  - the default is 5 for a regression problem like this

Unfortunately, due to limited computing capacity, we'll be using the defaults.

```
set.seed(1,sample.kind = "Rounding")
ranger<-train(x=full_jumps_data[,!(names(full_jumps_data) %in%
                                     c("MIP Share Next Season",
                                       "MIP Share Current Season","G_diff",
                                       "GS_diff","Season","SeasID"))],
              y=full_jumps_data$`MIP Share Current Season`,
              method="ranger", importance="permutation",
              trControl=trainControl(method="cv",number=10,p=0.9))
```

Let's get the top 5 most important variables of the ranger model.

```
##                       Overall              Vars
## PF/G_diff          100.00000          PF/G_diff
## STL/G_diff          94.81200         STL/G_diff
## MP/G_diff           83.39124          MP/G_diff
## TOV%_diff           71.93909          TOV%_diff
## 3P Per 36 Min_diff  69.68516 3P Per 36 Min_diff
```

Personal fouls per game seems like an odd choice for most important variable, but it's much likely correlated with minutes played per game. Turnover percentage is also in the top 5, implying that improved players are better at keeping the ball in control.

Let's see how the ranger model performs in 1987.

Method	RMSE	Ellis Vote Share	Top 3 Candidates	Top 3 Vote Shares
Linear	0.0343344	0.0608291	Michael Jordan	0.0778186
Linear	0.0343344	0.0608291	Dale Ellis	0.0608291
Linear	0.0343344	0.0608291	Terry Porter	0.0389028
KNN	0.0308246	0.0110901	Tim McCormick	0.0363522
KNN	0.0308246	0.0110901	Darnell Valentine	0.0354552
KNN	0.0308246	0.0110901	Tony Brown	0.0303645
Decision Tree	0.0347075	0.5622985	Dale Ellis	0.5622985
Decision Tree	0.0347075	0.5622985	Michael Jordan	0.2144521
Decision Tree	0.0347075	0.5622985	Otis Thorpe	0.2144521
Ranger	0.0312297	0.1608000	Dale Ellis	0.1608000
Ranger	0.0312297	0.1608000	Michael Jordan	0.1394003
Ranger	0.0312297	0.1608000	Michael Cage	0.1064854

The ranger model has the lowest RMSE, but has the second-highest predicted vote shares after the decision tree. Ellis is also predicted to win here, with Jordan coming in second. Cage increased his points per game by nine and upped his shooting percentage by 4% while cutting down his fouling rate.

Rborist is another R package for random forest machine learning. It has two tuning parameters: predFixed is equivalent to mtry in ranger, and minNode is equivalent to min.node.size in ranger. As above, we'll be using the defaults due to limited computing capacity.

```
set.seed(25,sample.kind="Rounding")
rborist<-train(x=full_jumps_data[,!(names(full_jumps_data) %in%
                                     c("MIP Share Next Season",
                                       "MIP Share Current Season","G_diff",
                                       "GS_diff","Season","SeasID"))],
              y=full_jumps_data$`MIP Share Current Season`,
              method="Rborist", importance="permutation",
              trControl=trainControl(method="cv",number=10,p=0.9))
```

Let's get the top 5 most important variables of the Rborist model.

```
##              Overall       Vars
## WS_diff    100.00000    WS_diff
## PTS/G_diff  92.28782 PTS/G_diff
## 2P/G_diff   85.99061  2P/G_diff
## OWS_diff    75.48548   OWS_diff
## FGA/G_diff  73.92730 FGA/G_diff
```

The 5 most important variables match up with expectations. Win Shares and Offensive Win Shares are advanced holistic statistics. Differences in 2 point field goals made per game and points per game measure offensive improvement (which is easier to recognize than defensive improvement). A difference in field goal attempts per game represents opportunity.

Let's see how the Rborist model performs in 1987.

Method	RMSE	Ellis Vote Share	Top 3 Candidates	Top 3 Vote Shares
Linear	0.0343344	0.0608291	Michael Jordan	0.0778186
Linear	0.0343344	0.0608291	Dale Ellis	0.0608291
Linear	0.0343344	0.0608291	Terry Porter	0.0389028
KNN	0.0308246	0.0110901	Tim McCormick	0.0363522
KNN	0.0308246	0.0110901	Darnell Valentine	0.0354552
KNN	0.0308246	0.0110901	Tony Brown	0.0303645
Decision Tree	0.0347075	0.5622985	Dale Ellis	0.5622985
Decision Tree	0.0347075	0.5622985	Michael Jordan	0.2144521
Decision Tree	0.0347075	0.5622985	Otis Thorpe	0.2144521
Ranger	0.0312297	0.1608000	Dale Ellis	0.1608000
Ranger	0.0312297	0.1608000	Michael Jordan	0.1394003
Ranger	0.0312297	0.1608000	Michael Cage	0.1064854
Rborist	0.0313513	0.2336699	Dale Ellis	0.2336699
Rborist	0.0313513	0.2336699	Michael Jordan	0.1428699
Rborist	0.0313513	0.2336699	Bill Cartwright	0.1378326

The Rborist model has similar RMSE to the ranger model, but higher predicted vote shares. As with the previous two tree-based models, Ellis and Jordan place first and second. Cartwright's case is similar to Jordan as a player who didn't play often in the previous season due to injury.

Models to Predict 2021 MIP Candidates

We will apply the same models we used for predicting the 2020 winner. We'll continue to evaluate performance using RMSE but now we have 2 test sets: 1986 and 2019.

```
linear_2021<-train(`MIP Share Next Season`~.-Season-`MIP Share Current Season`
                   -SeasID,
                data=full_data,method="lm",
                trControl=trainControl(method="cv",number=10,p=0.9))
```

Let's get the top 5 most important variables according to the linear model.

```
##                           Overall                    Vars
## Age                     100.00000                     Age
## `\\`FT Per 36 Min\\``    14.31358  `\\`FT Per 36 Min\\``
## `\\`3PA/G\\``            14.06343          `\\`3PA/G\\``
## `\\`USG%\\``             13.55688           `\\`USG%\\``
## `\\`PTS Per 36 Min\\``   13.52540 `\\`PTS Per 36 Min\\``
```

Age is by far the most important variable, which makes sense. Only two players over the age of 28 have won MIP. The next four variable importances are a drastic drop.

Let's see how the linear model performs in 1986 and 2019.

Method	RMSE	1987 Top Candidates	1987 Top Vote Shares	2020 Top Candidates	2020 Top Vote Shares
Linear	0.0319365	Charles Barkley	0.0151579	Trevon Duval	0.0164597
Linear	0.0319365	Alvin Robertson	0.0150723	Trae Young	0.0141080
Linear	0.0319365	John Stockton	0.0129943	Mitchell Robinson	0.0131543

The predicted vote shares are extremely low. Robertson and Barkley finished 1st and 2nd in 1986's MIP voting and an MIP has never won the award twice. Stockton's underlying stats primed him for a breakout, but it wasn't until 1988 that he started the majority of games for the Utah Jazz and received MIP votes. This will be a recurring caveat, since we can't predict opportunity/playing time.

Duval only played 3 games in 2019, and hasn't played a single game in 2020. Young was named a All-Star Game starter this year. Robinson has been effective when he has played, but unfortunately his fouling issues continue to plague him and limit his minutes.

```
knn_2021<-train(`MIP Share Next Season`~.-Season-
                  `MIP Share Current Season`-SeasID,
            data=full_data,method="knn", tuneGrid=data.frame(k=seq(5,50)),
            trControl=trainControl(method="cv",number=10,p=0.9))
knn_2021$finalModel
```

```
## 50-nearest neighbor regression model
```

Let's see how the knn model performs in 1986 and 2019.

Method	RMSE	1987 Top Candidates	1987 Top Vote Shares	2020 Top Candidates	2020 Top Vote Shares
Linear	0.0319365	Charles Barkley	0.0151579	Trevon Duval	0.0164597
Linear	0.0319365	Alvin Robertson	0.0150723	Trae Young	0.0141080
Linear	0.0319365	John Stockton	0.0129943	Mitchell Robinson	0.0131543
KNN	0.0325144	Darrell Walker	0.0265280	Aaron Gordon	0.0426862
KNN	0.0325144	Larry Drew	0.0248165	Thomas Bryant	0.0354024
KNN	0.0325144	LaSalle Thompson	0.0234361	Josh Richardson	0.0322350

The RMSE is higher compared to the linear, but the predicted vote shares are also higher. Thompson, Drew and Walker didn't end up receiving extra minutes (and in Thompson's case, had his minutes decreased). Walker became more efficient, while the other two became less efficient.

The top 3 predicted players for 2020 have all received votes (albeit a small amount) in previous seasons: Richardson and Gordon in 2017, Richardson and Bryant in 2018. Gordon has been projected to make the leap for what feels like the last 3 seasons. Bryant had his minutes marginally increased, and subsequently made a marginal improvement. Richardson had the opportunity to be the focal point for the Miami Heat in 2020, but was traded to the Philadelphia 76ers before the season. The 76ers had an established offensive hierarchy, so Richardson wasn't able to put up increased per game stats. The KNN model wouldn't know these things.

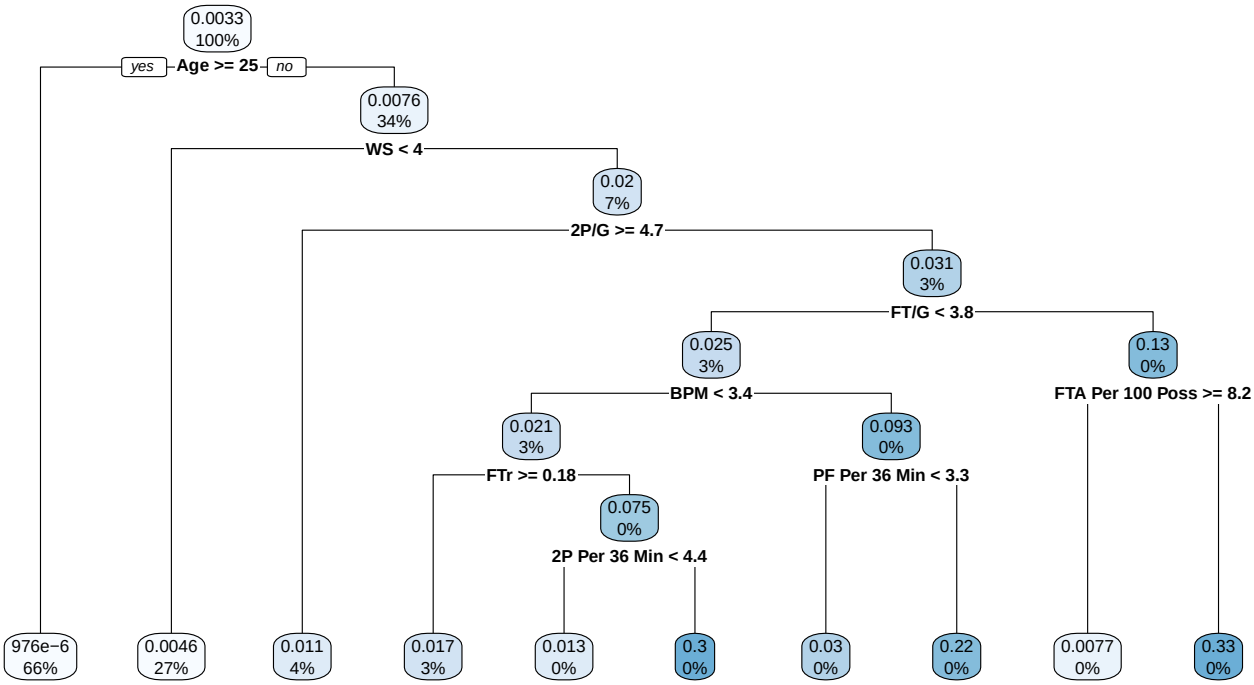
With the decision tree model, we will lower the upper limit on the complexity parameter to 0.01. Since there's higher variability, a higher complexity parameter might prune the entire tree.

```
set.seed(50,sample.kind = "Rounding")
rpart_2021<-train(x=full_data[,!(names(full_data) %in%
                                   c("MIP Share Next Season",
                                     "MIP Share Current Season","Season",
                                     "SeasID"))],
             y=full_data$`MIP Share Next Season`,method="rpart",
```

```
            tuneGrid = data.frame(cp=seq(0,0.01,0.001)),
            trControl=trainControl(method="cv",number=10,p=0.9))
rpart_2021$bestTune
```

```
##       cp
## 10 0.009
```

The model is optimized at cp = 0.009. Let's look at the decision tree.



The decision tree maximizes its prediction when a player does all of the following:

- they are less than 25 years of age
- they have accumulated more than 4 Win Shares
- they have made at most 4.7 2-point field goals per game
- they have made at least 3.8 free throws per game
- they have attempted less than 8.2 free throws per 100 possessions

The decision tree minimizes its prediction when a player is older than 25 years of age.

Age is a good starting partition, because most votes tend to skew towards younger players. The subsequent decisions paint a picture of a young player who is able to help the team in ways other than scoring, like playmaking and defense. However, he is still able to get to the free throw line when necessary.

Let's see how the decision tree model performs in 1986 and 2019.

Method	RMSE	1987 Top Candidates	1987 Top Vote Shares	2020 Top Candidates	2020 Top Vote Shares
Linear	0.0319365	Charles Barkley	0.0151579	Trevon Duval	0.0164597
Linear	0.0319365	Alvin Robertson	0.0150723	Trae Young	0.0141080
Linear	0.0319365	John Stockton	0.0129943	Mitchell Robinson	0.0131543
KNN	0.0325144	Darrell Walker	0.0265280	Aaron Gordon	0.0426862
KNN	0.0325144	Larry Drew	0.0248165	Thomas Bryant	0.0354024
KNN	0.0325144	LaSalle Thompson	0.0234361	Josh Richardson	0.0322350
Decision Tree	0.0347424	Doc Rivers	0.2176770	Monte Morris	0.2994455
Decision Tree	0.0347424	Benoit Benjamin	0.0168186	Bam Adebayo	0.0168186
Decision Tree	0.0347424	Jay Humphries	0.0168186	Jarrett Allen	0.0168186

As with the 2020 winners model, the decision tree RMSE is greater than the linear and knn RMSEs. The vote shares for the top player are much higher than any other probabilities. Rivers did improve significantly in advanced stats, but his points per game moved up marginally. He helped the team through playmaking and defense. Morris is a 2019 lite version of Doc Rivers, in that he's not known for his scoring. However, he was never going to start considering the significant investment his team has made in their starting backcourt. Therefore, he wouldn't be able to generate the per game stats. The 2nd and 3rd candidates are tied with 4 others in 1987 and 14 others in 2019.

```
set.seed(75,sample.kind = "Rounding")
ranger_2021<-train(x=full_data[,!(names(full_data) %in%
                                    c("MIP Share Next Season",
                                      "MIP Share Current Season","Season",
                                      "SeasID"))],
             y=full_data$`MIP Share Next Season`,
             method="ranger", importance="permutation",
             trControl=trainControl(method="cv",number=10,p=0.9))
```

Let's get the top 5 most important variables according to the ranger model.

```
##                   Overall           Vars
## FTA/G           100.00000          FTA/G
## FG/G             56.75510           FG/G
## MP/G             51.48888           MP/G
## FGA/G            43.54215          FGA/G
## PTS Per 36 Min   40.27553 PTS Per 36 Min
```

3 of the top 5 most important variables in the ranger model relate to scoring (free throws attempted per game, field goals made per game and field goals attempted per game).

Let's see how the ranger model performs in 1986 and 2019.

Method	RMSE	1987 Top Candidates	1987 Top Vote Shares	2020 Top Candidates	2020 Top Vote Shares
Linear	0.0319365	Charles Barkley	0.0151579	Trevon Duval	0.0164597
Linear	0.0319365	Alvin Robertson	0.0150723	Trae Young	0.0141080
Linear	0.0319365	John Stockton	0.0129943	Mitchell Robinson	0.0131543
KNN	0.0325144	Darrell Walker	0.0265280	Aaron Gordon	0.0426862
KNN	0.0325144	Larry Drew	0.0248165	Thomas Bryant	0.0354024
KNN	0.0325144	LaSalle Thompson	0.0234361	Josh Richardson	0.0322350
Decision Tree	0.0347424	Doc Rivers	0.2176770	Monte Morris	0.2994455
Decision Tree	0.0347424	Benoit Benjamin	0.0168186	Bam Adebayo	0.0168186
Decision Tree	0.0347424	Jay Humphries	0.0168186	Jarrett Allen	0.0168186
Ranger	0.0321523	Alton Lister	0.0266745	JaVale McGee	0.0238565
Ranger	0.0321523	John Stockton	0.0188303	Willy Hernangómez	0.0196823
Ranger	0.0321523	Kevin Willis	0.0167161	Marc Gasol	0.0194817

The ranger model has the best RMSE so far, and has higher predicted vote shares than all but the top candidates from the decision tree model.

Lister is the first candidate we've seen that received extra opportunity. While he increased his per game statistics due to being given 8 more minutes of playing time per game, his rate statistics dropped. Willis improved across the board in 1987. Although his minutes per game only increased by 4, he attempted more shots and converted them at a higher rate (despite conventional wisdom stating that as shot volume goes up, shooting percentages go down). Willis had sound defensive impact, but he improved more on the offensive end.

McGee is 32 and Gasol is 35; the oldest player to have won MIP was Darrell Armstrong (who was 30). Hernangomez has spent enough time sitting on the bench in the past few seasons that it's safe to say there won't be any improvement: he is who he is.

One curious component about the Ranger model is that the predicted big men (Lister, McGee, Gasol and Hernangomez) had a significantly older average age than the guards (30 vs 23).

```
set.seed(100,sample.kind = "Rounding")
rborist_2021<-train(x=full_data[,!(names(full_data) %in%
                                     c("MIP Share Next Season",
                                       "MIP Share Current Season","Season",
                                       "SeasID"))],
             y=full_data$`MIP Share Next Season`,
             method="Rborist", importance="permutation",
             trControl=trainControl(method="cv",number=10,p=0.9))
```

Let's get the top 5 most important variables according to the rborist model.

```
##                   Overall           Vars
## Age             100.00000            Age
## ORB%             85.20456           ORB%
## 3P%              80.19982            3P%
## FGA Per 36 Min  61.41245 FGA Per 36 Min
## PTS Per 36 Min  60.75217 PTS Per 36 Min
```

Age is the most important factor in the rborist model, and 4 rate stats follow. This makes more sense than the ranger model, in that Most Improved Player candidates wouldn't have per game statistics that jump off the page, but rather rate statistics that would be showing ability.

Let's see how the Rborist model performs in 1986 and 2019.

Method	RMSE	1987 Top Candidates	1987 Top Vote Shares	2020 Top Candidates	2020 Top Vote Shares
Linear	0.0319365	Charles Barkley	0.0151579	Trevon Duval	0.0164597
Linear	0.0319365	Alvin Robertson	0.0150723	Trae Young	0.0141080
Linear	0.0319365	John Stockton	0.0129943	Mitchell Robinson	0.0131543
KNN	0.0325144	Darrell Walker	0.0265280	Aaron Gordon	0.0426862
KNN	0.0325144	Larry Drew	0.0248165	Thomas Bryant	0.0354024
KNN	0.0325144	LaSalle Thompson	0.0234361	Josh Richardson	0.0322350
Decision Tree	0.0347424	Doc Rivers	0.2176770	Monte Morris	0.2994455
Decision Tree	0.0347424	Benoit Benjamin	0.0168186	Bam Adebayo	0.0168186
Decision Tree	0.0347424	Jay Humphries	0.0168186	Jarrett Allen	0.0168186
Ranger	0.0321523	Alton Lister	0.0266745	JaVale McGee	0.0238565
Ranger	0.0321523	John Stockton	0.0188303	Willy Hernangómez	0.0196823
Ranger	0.0321523	Kevin Willis	0.0167161	Marc Gasol	0.0194817
Rborist	0.0318266	Alton Lister	0.0299793	Kevin Love	0.0291414
Rborist	0.0318266	John Stockton	0.0269551	Luka Dončić	0.0276821
Rborist	0.0318266	Kevin Willis	0.0262253	JaVale McGee	0.0276594

The Rborist RMSE is similar to the linear RMSE. The 1987 Rborist top 3 is the same as the 1987 Ranger top 3, but with higher predicted vote shares.

Love has already won MIP back in 2011, and at 31 years of age could be considered on the back nine of his career. Doncic is the truly interesting case, as he vaulted himself from good rookie to being in the conversation for Most Valuable Player.

Results

We've trained our models, and now it's time to test them on the 2020 data.

2020 Winner

We won't use the knn model to generate predictions, since it didn't include Dale Ellis as a top 3 candidate and wasn't consistent with our expectations. In addition, we will use the sum of arithmetic average and median of the four models to make our final predictions, as well as take the top 5 candidates instead of top 3. The reason for adding mean and median together is that mean favors players who have a few models projecting large probabilities, while median favors players who are consistently in the top 5.

Model	Top 5 Candidates	Top 5 Predicted Vote Shares
Linear	Bam Adebayo	0.0455398
Linear	Trae Young	0.0453228
Linear	Devonte' Graham	0.0452669
Linear	Luka Dončić	0.0426084
Linear	Shai Gilgeous-Alexander	0.0372333
Decision Tree	Brandon Ingram	0.5622985
Decision Tree	Jayson Tatum	0.5622985
Decision Tree	Luka Dončić	0.3345672
Decision Tree	Devonte' Graham	0.3345672
Decision Tree	Trae Young	0.0887957
Ranger	Devonte' Graham	0.1330465
Ranger	Trae Young	0.1033661
Ranger	Luka Dončić	0.0949651
Ranger	Bam Adebayo	0.0755821
Ranger	Shai Gilgeous-Alexander	0.0729338
Rborist	Devonte' Graham	0.1514178
Rborist	Trae Young	0.1233206
Rborist	Luka Dončić	0.1127907
Rborist	Shai Gilgeous-Alexander	0.0780316
Rborist	Jayson Tatum	0.0634210

Model	Top 5 Candidates	Top 5 Predicted Vote Shares
Median+Mean	Devonte' Graham	0.3083068
Median+Mean	Luka Dončić	0.2501107
Median+Mean	Jayson Tatum	0.2387586
Median+Mean	Brandon Ingram	0.2017616
Median+Mean	Trae Young	0.1862822

Based on the composite model, Devonte' Graham should be the MIP. This tracks, since Graham was the third-string guard on his team last year and developed into a fringe All-Star candidate this year. The other four players in the top 5 all took a leap, and made the All-Star team this year. In Doncic's case, he jumped all the way into conversation for Most Valuable Player. Bam Adebayo and Shai Gilgeous-Alexander made appearances in single-model top 5s, but weren't able to crack the composite top 5. Adebayo has anchored the Miami Heat defense, while Gilgeous-Alexander has cemented himself as a cornerstone on the Oklahoma City Thunder.

2021 Candidates

We'll include the KNN model this time around.

Model	Top 5 Candidates	Top 5 Predicted Vote Shares
Linear	Luka Dončić	0.0153845
Linear	Trae Young	0.0142266
Linear	Jaren Jackson	0.0137777
Linear	James Harden	0.0133347
Linear	Coby White	0.0123414
KNN	Malik Monk	0.0300542
KNN	OG Anunoby	0.0298461
KNN	Harrison Barnes	0.0269492
KNN	Christian Wood	0.0248537
KNN	De'Andre Hunter	0.0227423
Decision Tree	Jarrett Allen	0.0168186
Decision Tree	Mikal Bridges	0.0168186
Decision Tree	Kristaps Porziņģis	0.0168186
Decision Tree	Mitchell Robinson	0.0168186
Decision Tree	Christian Wood	0.0168186
Ranger	Kelly Oubre	0.0294037
Ranger	Donte DiVincenzo	0.0209979
Ranger	Stephen Curry	0.0181069
Ranger	Jarrett Allen	0.0162904
Ranger	Jaren Jackson	0.0160376
Rborist	Kelly Oubre	0.0412818
Rborist	Stephen Curry	0.0401681
Rborist	Jaren Jackson	0.0337806
Rborist	Mitchell Robinson	0.0324955
Rborist	Kristaps Porziņģis	0.0311625

Model	Top 5 Candidates	Top 5 Predicted Vote Shares
Median+Mean	Jarrett Allen	0.0337680
Median+Mean	Christian Wood	0.0310652
Median+Mean	Ivica Zubac	0.0307065
Median+Mean	OG Anunoby	0.0293226
Median+Mean	Jaren Jackson	0.0275697

There's a more eclectic mix of candidates, and the vote shares are much lower than the 2020 winner predictions. This was expected, since there is far more variance in player seasons before receiving MIP votes. Two players might have very similar statistics in one season, but one of the players received more playing time in the subsequent season and received MIP votes. The algorithm would be confused and split down the middle.

Allen, Zubac and Anunoby all exemplify the young player asked to fill a certain role on championship contending teams. They are not asked to shoulder the scoring load, but are expected to focus on defense and contribute when the ball swings to them on offense. Jackson is a highly touted franchise cornerstone for the Memphis Grizzlies.

Wood is an intriguing candidate. He was talked up during the offseason and first half of the season, but played limited minutes off the bench. At the trade deadline, Wood's team traded the starter at his position and increased Wood's minutes. He blossomed, flashing star-like potential. Perhaps a full season as a starter will show the league what Wood is really capable of.

Four players placed in the top 5 of multiple models, but didn't crack the top 5 in the composite.

- Kelly Oubre (2 out of 5): third option on the rising Phoenix Suns
- Mitchell Robinson (2 out of 5): needs to cut down on fouling so he can stay on the court for the New York Knicks
- Stephen Curry (2 out of 5): NBA superstar for the Golden State Warriors, but only played 5 games due to a broken hand (Ranger and Rborist got confused by seeing such a great player play so few games)
- Kristaps Porzingis (2 out of 5): already an All-Star I don't foresee a huge statistical jump with Doncic being the unquestioned face of the Dallas Mavericks

Conclusion

Using a database of 15595 individual player seasons from 1984 to 2020, we set out to predict both the 2020 Most Improved Player award winner, as well as MIP candidates for 2021. Using the sum of mean and median for four models (linear, decision tree and 2 random forests) that were trained on statistical jumps data (current season statistics minus previous season statistics), **Devonte' Graham** is projected to be the 2020 winner. Rounding out the top 5 are Luka Doncic, Jayson Tatum, Brandon Ingram and Trae Young. Using the sum of mean and median for five models (the previously mentioned four as well as knn) trained on regular season data, the top 5 candidates for 2021 are Jarrett Allen, Jaren Jackson, Ivica Zubac, Christian Wood and OG Anunoby.

Both models suffer from the inability to quantify intangibles such as narrative and popularity. Brandon Ingram was traded from the very popular Los Angeles Lakers in the offseason. A large amount of fans have monitored his progress due to this previous association. The 2020 winner model is limited by not knowing why a player played so few games in the prior season. Was it due to injury or genuinely bad play? The 2021 candidates model is limited by the fact that we can't predict opportunity.

By nature, the per game & per 36 minutes & per 100 possessions statistics are very correlated. I didn't remove highly correlated variables by design, as I wanted the models themselves to perform feature selection and determine what the most important variables were.

Further work might involve looking at the problem through a classification lens. Instead of predicting vote share, we can instead classify a player as having won the MIP or not. Other future work could involve removing highly correlated variables or using the given subsets instead of the combined set. These machine learning algorithms could also be applied to the NBA's other player awards.