PREDICTING THE NBA'S MOST IMPROVED PLAYER (MIP)

Summary
In this project, I create a logistic regression model to predict who will win the NBA MIP award. This model would benefit NBA General Managers, sports betters, fantasy basketball league managers, as well as basketball fans simply interested in who will the MIP award.

Sam Liang

STA 9797 Final Project

Sam Liang
Prof. Li
STA 9797 Final Project
due 12/23/23 @ Noon

Introduction

"The NBA's Most Improved Player Award (MIP) is an annual NBA award given to the player who has shown the most progress during the regular season compared to previous seasons" (Wikipedia, 2023). The award was inaugurated during the 1985-1986 season. My goal is to create a logistic regression model to determine which factors best predict who will win the award. My motive in doing so is to perform better in my fantasy basketball league. Drafting the player who wins the Most Improved Player (MIP) award (as well as strong candidates for the award) will greatly increase one's chance at winning a fantasy league. A crucial part of winning a fantasy basketball league is drafting well; the draft occurs before the actual NBA season begins. And a key component to drafting well is to find players who will perform better than their average draft pick would suggest. For example, in Yahoo! leagues, the average draft pick of Lauri Markannen, the winner of the MIP award last year (2022-2023), was 96. In 9-category formats (a popular format to play fantasy basketball), Markannen ended the season with a rank of 21 (Lloyd, 2023). Acquiring a player like Markannen for one's fantasy team greatly improves one's chances of winning. It is for this reason that I aim to better predict who will win or be solid candidates for the MIP award.

Regarding my dataset, I created the dataset myself with statistics provided by Wikipedia and basketball-reference.com. This dataset includes the winner of the MIP awards and the players who finished 2nd in voting each year since the inception of the award in 1985-1986. Additionally – and this is of utmost importance to mention –the statistics for the MIPs and runner-ups are for the year ***before*** they win or are nominated for the award. I do this because in fantasy basketball, for any given draft, a team manager only knows the statistics for the year prior to a player winning MIP. In other words, one cannot know the statistics of a breakout year before it happens. As such, a logistic model built from this data will help a fantasy team manager to predict who will win MIP before it happens, allowing the manager to draft better from this additional insight.

I originally included only the MIP and runner-ups in the dataset; however, as stated in my in-class presentation, the only variable that came out to be statistically significant was DraftPick. Per your suggestion, Professor Li, I decided to add more players to my dataset for two reasons: one, obtain a larger sample size and two, determine if variables that were shown to be statistically insignificant are actually significant. After all, the players I initially included in the data are similar in their statistics. The players who won MIP or were runner-ups in voting are primarily young players (in their younger 20's) who did not perform spectacularly in the season prior to their breakout season. As such, I decided to include the MVP as well as Rookie of the Year (ROY) for each year to get a more diverse and accurate sample of NBA players. This will also provide more contrast with the statistics of the MIP players; that is, the MVP of each year tends to be older and more experienced and will have a much better performing year than to-be MIPs. The ROYs will also be young players but they will tend to have higher performance statistics than the MIPs. With these additions, the data includes 154 total players. After updating the data in this way, I hope to obtain a better performing model than from my first attempt.

Lastly, I ought to mention two special cases in the data. First, there were three runner-ups in the MIP race during the 2010 season (Kevin Durant, Marc Gasol, and George Hill). As such, the prior year (2009) statistics for all three of these players are included. Second, the data for Isaac

Austin, the 1997 winner, is omitted. I decided to omit him because he did not play in the NBA during the 1995-1996 season, the season prior to his MIP season. (He played professionally in Turkey where he averaged 22.3 points and 13.9 rebounds.)

Column Dictionary

Year: this is the year of the NBA season. For instance, 1985, 1986, 1987, etc. Note that years in this dataset are recorded as the year the NBA season ended. For example, the 1984-1985 season is represented by 1985.
Name: this is the name of the player
Won: this column states whether the player won/was 2nd in voting the next year or neither of these. There are two levels: 1 for the player won or was a runner-up and 0 for the player did not win. Note that I decided to encode both winners and runner-ups as 1 because both players would provide good value for fantasy basketball.
Position: this column states the player's position: guard, guard/forward, forward, forward/center, center
Age: this is the player's age at the start of the season
DraftPick: this is the pick at which the player was drafted. Note: Darrell Armstrong was undrafted but I designate his draft pick at 55 because the last pick of his draft year was 54.
Ben Wallace was also undrafted but I designate his draft pick at 59 because the last pick of his draft year was 58.
TeamChange: This column states whether the player changed teams between the season in question and the end of the prior year (two levels: Yes or No). Note that if a player changed teams mid-season and he stayed on that team until the end of the next season, then this column's value will be No. For example, Kevin Duckworth, the Most Improved Player in 1988, changed teams (from the San Antonio Spurs to the Portland Trail Blazers) during the 1986-1987 season. He stayed with the Portland Trail Blazers during the 1987-1988 season and won the MIP award. His TeamChange column will have a value of No.
YearsNBA: this is the number of years of NBA experience the player has including the given season. Note that this does not include any years of experience the player had in a professional basketball league outside of the NBA.
GamesPlayed: the number of games the player played in the season
GamesStarted: the number of games the player started in the season
MPG: the number of minutes the player played per game on average
FGPerc: the player's average field goal percentage during the season
ThreePtPerc: the player's 3-point shooting percentage during the season. Note that there are some players in the dataset who did not attempt a three-point shot during a season. These players are James Donaldson, Kevin Duckworth, Alan Henderson, and Clint Capela. I designated these players' ThreePtPerc as 0.
FTPerc: the player's free throw shooting percentage during the season
RPG: the player's average rebounds per game during the season
APG: the player's average assists per game during the season
SPG: the player's average steals per game during the season
BPG: the player's average blocks per game during the season
TO: the player's average turnovers per game during the season
PPG: the player's average points per game during the season

FantasyPTS: the player's average fantasy points on Yahoo! Fantasy basketball. This is calculated with the following formula:

$$\text{FantasyPts} = 1(\text{Points}) + 1.2(\text{Rebounds}) + 1.5(\text{Assists}) + 3(\text{Steals}) + 3(\text{Blocks}) - 1(\text{Turnovers})$$

(Yahoo Fantasy Sports Basketball, 2023).

Description of Statistical Model

I used a multiple logistic regression model to predict who will win the MIP award. As such, the response variable is **Won**. As stated above, this column states whether the player won/was 2nd in voting the next year or neither of these. The predictor variables are the Position, Age, DraftPick, TeamChange, YearsNBA, GamesPlayed, GamesStarted, MPG, FGPerc, ThreePtPerc, FTPerc, and FantasyPTS columns. (The RPG, APG, SPG, BPG, TO, PPG columns are summarized by the FantasyPTS column.)

Referencing the class notes from Week 6: Multiple Logistic Regression, the model with k predictors is

$$\text{logit}\{\pi(x)\} = \log\frac{\pi(x)}{1-\pi(x)} = \alpha + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_k x_k$$

where β_i is the partial effect of x_i controlling for other variables in model $i = 1, \dots, k$ (Li, 2023).

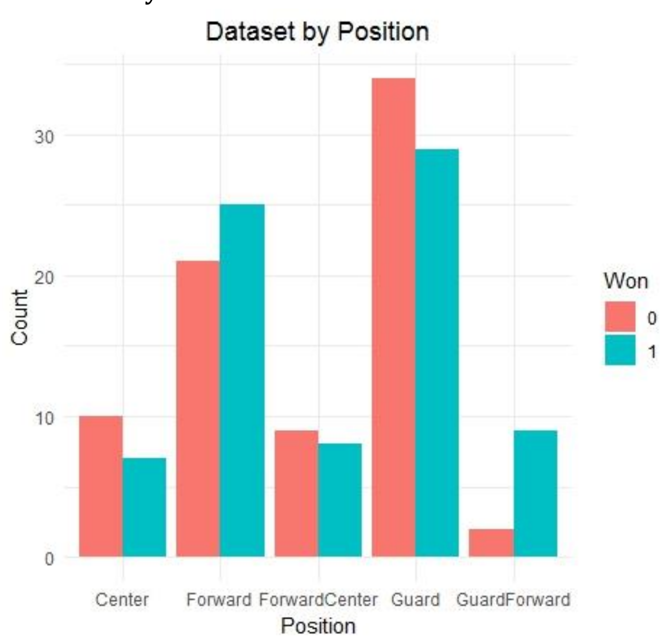
Exploratory Analysis

a. Position

I begin by analyzing the dataset by position. These are the counts:

Center	Forward	Forward/Center	Guard	Guard/Forward
17	46	17	63	11

When looking at the bar chart below, it is notable that guards and forwards are more prevalent in the MIP conversation than are centers. Of course, the positions most represented in those who did not win are also guards followed by forwards.



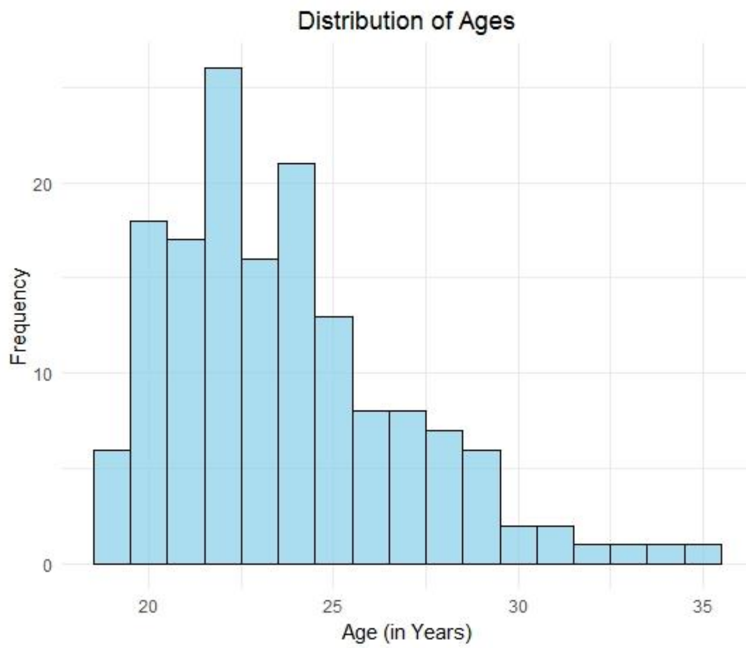
These are the exact counts by position for those who won (the green columns above):

Center	Forward	Forward/Center	Guard	Guard/Forward
7	25	8	29	9

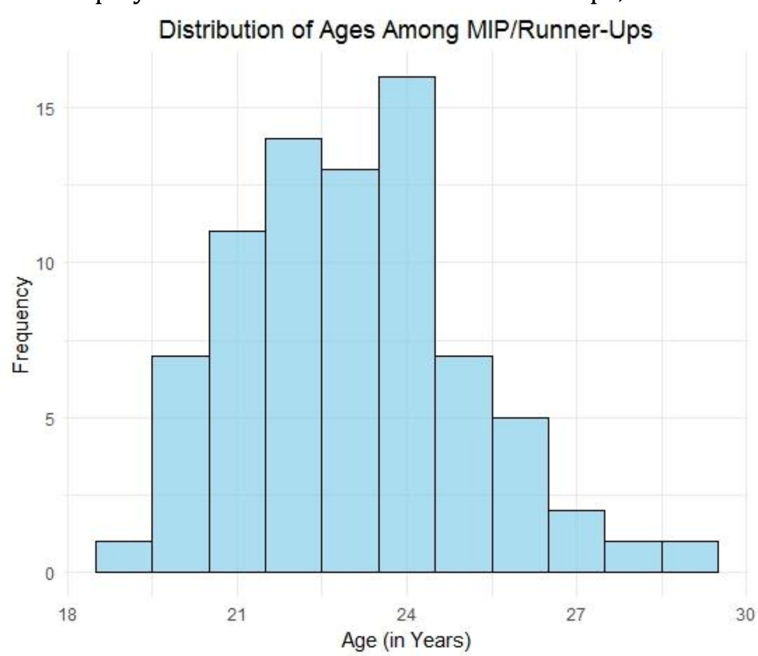
This denotes that when drafting for value in a fantasy league, one might aim to draft guards and forwards.

b. Age

When looking at the age distribution of all players, we have



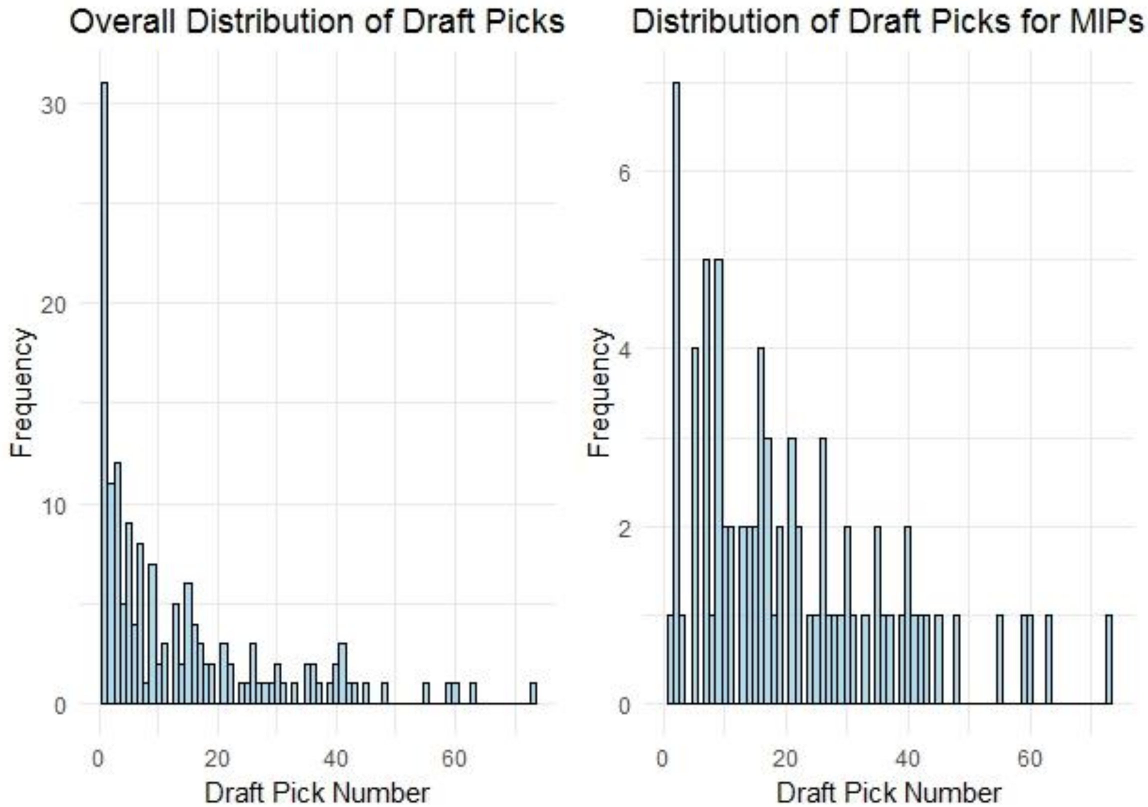
The distribution is skew right, meaning most players represented in the dataset are on the younger side (in their early 20's). The minimum age represented is 19 years and the maximum is 35 years. When filtering the data for players who won MIP or were runner-ups, we have



It is notable that no player in their 30's has ever won the MIP award. This means that when drafting players with the most potential for improvement, we ought to draft younger players who are in their 20's. The minimum age represented here is 19 years and the maximum is 29 years.

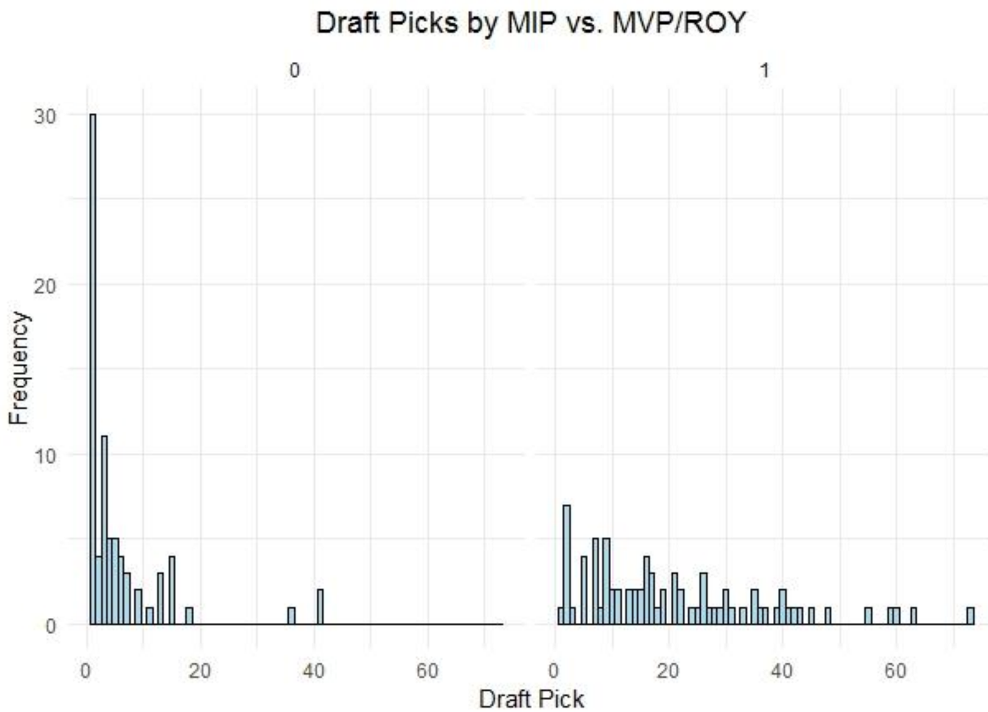
c. Draft Pick

Let's now examine the dataset through the lens of draft picks:



At first glance, it seems that both plots are similar. They have a similar shape and are both skew right, meaning players who win the MIP, MVP, or ROY awards tend to be drafted earlier. There are players of course who win these awards who were drafted late or were undrafted. As stated above in the column dictionary, Darrell Armstrong (MIP winner) and Ben Wallace (MIP runner-up were undrafted.

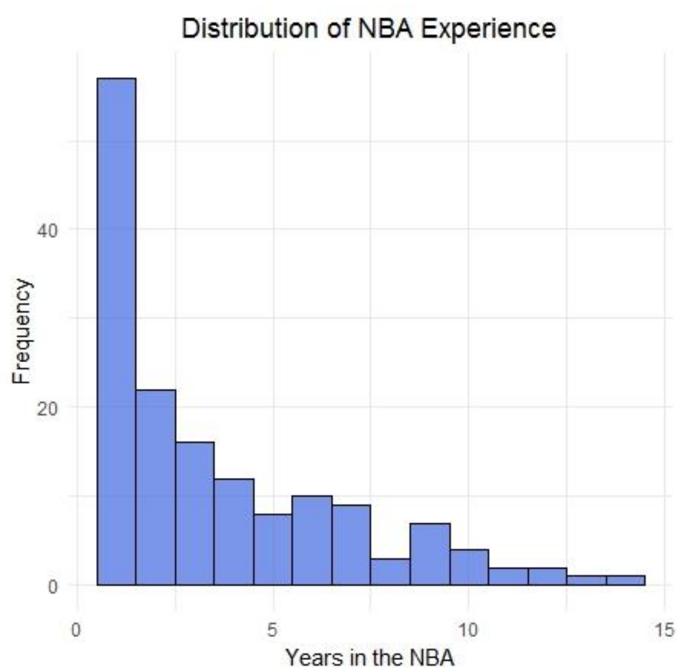
When we separate the two groups in the dataset, MIP winners/runner-ups against MVP/ROYs, the difference in distribution of draft pick becomes more apparent (see next page).



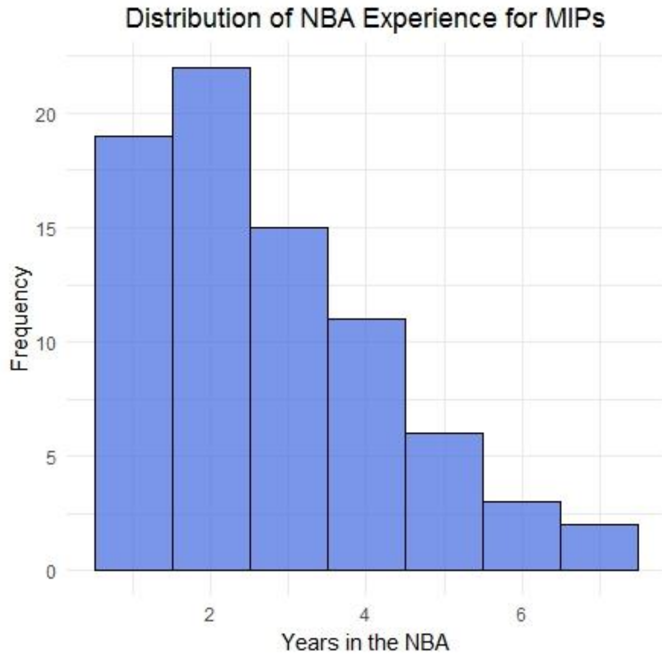
The histogram on the left shows the distribution of draft picks of the MVP and ROYs whereas the histogram on right shows the distribution for the MIP/runner-ups. We can more readily see from this plot that the MVP and ROYs awards favor high draft picks whereas the MIP award gives more credence to middle-of-the-pack and low draft picks. This indicates that a fantasy league manager shouldn't necessarily look for players who were drafted high in their draft classes.

d. Years of NBA Experience

In terms of NBA experience, we have



This distribution, just as it was with ages and draft picks, is also skew right. This indicates that MVP, ROY, and MIP awards tend to be won towards the early and prime years of a player's career rather than the latter stages. The minimum number is 1 year (rookies) and the maximum is 14 years. After filtering the data for MIP winners/runner-ups, we have



The distribution for MIPs is also skew right but it's less extreme. Additionally, the maximum number of years of NBA experience drops from 14 to 7 when filtering the data for MIPs. This indicates that MIP winners tend to be less experienced than MVP winners.
We can also see that the most common players to win or be nominated for MIP are third years, represented by those with two years of NBA experience, followed by sophomores, represented by those with one year, and fourth years, represented by those with three years. (Recall that this dataset tracks the years of players one year prior to when they win MIP.) For fantasy league managers, this means that one should target players who are in their first three years of playing in the NBA.

Analysis

Before building the logistic regression model, I split the data into a training and testing set for the purpose of cross-validation. The training set has 124 out of 154 of the players and the test set includes the other 30. After going through the process of backward stepwise selection, we eliminate Position, ThreePtPerc, FTPerc, TeamChange, GamesPlayed, FGPerc, MPG, GamesStarted, Age, and lastly YearsNBA. We are left with two variables, DraftPick, which has a p-value of 0.0441 and FantasyPTS, which has a p-value of 3.34e-06. With an estimated coefficient of 0.07624, an increase of one unit in DraftPick will increase the log odds that a player will win MIP by an average of 0.07624. Regarding FantasyPTS, with an estimated coefficient of -0.23879, an increase of one unit in FantasyPTS, decreases (since it's negative) the log odds that a player will win MIP by an average of 0.23879.

Coefficient	Estimate	p-value
Intercept	7.27627	5.28e-05
DraftPick	0.07624	0.0441
FantasyPTS	−0.23879	3.34e-06

Hence, our model can be summarized as

$$\log\frac{\pi(x)}{1-\pi(x)} = 7.27627 + 0.07624(\text{Draft Pick}) - 0.23879(\text{FantasyPTS})$$

Next, we check for model strength and predictive power. The following analyses are adapted from https://www.statology.org/logistic-regression-in-r/ as well as Week 6 of our class notes.

a. McFadden's R^2

First, we can compute McFadden's R^2 to assess the predictive power of our logistic regression model. Values over 0.40 indicate that a model fits the data very well (Bobbitt, 2020). We get a value of 0.6514361 which indicates that our model fits the data very well and has high predictive power.

b. Variable Importance

Next, I used the varImp function in R to check which variable is a more important predictor. DraftPick obtained a value of 2.013103 and FantasyPTS a value of 4.648834. These figures correspond with the p-values; that is, FantasyPTS had a lower p-value than DraftPick. This indicates that FantasyPTS is a more important predictor than DraftPick for MIP.

c. Variance Inflation Factor (VIF)

Third, I checked the VIF. Both DraftPick and FantasyPTS have a VIF of 1.000012. Since neither are over 5, we conclude that multicollinearity is not a problem in our model.

d. Model Diagnostics

After using our model to calculate the log odds for each individual in the test dataset, I create the confusion matrix:

		Predicted	
		Negative (0)	Positive (1)
Actual	Negative (0)	14	1
	Positive (1)	3	12

We obtain the following metrics:

$$\text{Sensitivity} = \frac{\text{True Positive}}{\text{True Positive + False Negative}} = \frac{12}{12+3} = 0.8$$

$$\text{Specificity} = \frac{\text{True Negative}}{\text{True Negative + False Positive}} = \frac{14}{14+1} = 0.9333$$

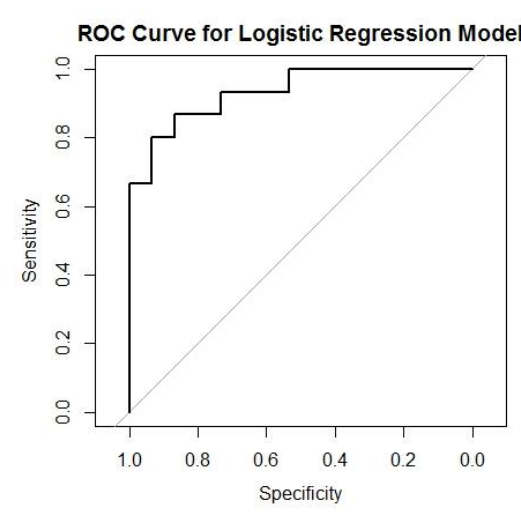
$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive + False Positive}} = \frac{12}{12+1} = 0.9231$$

$$\text{Model Accuracy} = \frac{\text{True Positive + True Negative}}{\text{Total}} = \frac{12+14}{30} = 0.8667$$

These numbers indicate that our model is sensitive, specific, precise, and accurate.

e. ROC Curve

According to p. 23/25 of our Week 6 notes, AUC is a scalar that represents the area under the ROC curve (Li, 2023). The ROC Curve for our model is shown right. An AUC value of 0.5 indicates a model that performs no better than a random guess whereas a value of 1 indicates a perfect model that correctly classifies all instances. This model obtains an AUC value of 0.9333, which indicates that the model does a good job of predicting whether a player will win MIP.



f. Predictions for Most Improved Player (2023-2024)

After building and analyzing the model, I put it to the test. To do so, I gathered the draft picks and average fantasy points per game for the 2022-2023 season of various players, including top contenders for the MIP award according to VegasInsider, including Tyrese Maxey, Coby White, Alperen Sengun, Scottie Barnes, Tyrese Haliburton, as well as unlikely candidates (as of December 23rd, 2023) such as Christian Braun, Moses Moody, Herbert Jones, and Devin Vassell (Staff, 2023). I also included players who aren't listed on VegasInsider as of December 23rd, 2023, such as Joel Embiid, Luka Doncic, Stephen Curry, and Terance Mann. Essentially, I gathered players who finished in the top 75 of fantasy leagues last year and rounded out the dataset with players who did not finish in the top 75 but are candidates for MIP. For reference, this dataset will be attached as MIP2024.

Our model states that among the players in the MIP2024 dataset, these are the players with the highest odds of winning Most Improved Player:

Name	DraftPick	FantasyPTS	Log Odds
Jose Alvarado	61 (undrafted)	18.86	0.9994034
Jae'Sean Tate	61 (undrafted)	18.91	0.9993962
Terance Mann	48	17.73	0.9987734
Christian Braun	21	10.38	0.9983391
Royce O'Neale	61 (undrafted)	23.47	0.9982083
Moses Moody	14	8.74	0.9980860
Cam Thomas	27	15.14	0.9967297
Jalen Johnson	20	14.6	0.9951059
Daniel Gafford	38	21.37	0.9937606
Louis King	61 (undrafted)	28.8	0.9936317

Conclusion, Implications, Future Questions & Ways to Analyze

Based on the model's predictions of the 2023-2024 Most Improved Player alone, it is evident that our model is too simple. Given the in-season statistics that we have as of today, December 23rd, 2023, the players with the highest odds of winning MIP are Tyrese Maxey, Coby White, Alperen Sengun, and Scottie Barnes. The model predicted none of these players.

Our model includes only two variables, DraftPick and FantasyPTS, which means that our model predicts that a player who was drafted later in their respective draft and averaged a meager amount of fantasy points during a season will win MIP. In other words, the models says that a player who went undrafted and did not play at all, earning zero fantasy points, has the highest odds of winning Most Improved Player of the year. This is obviously wrong. That being said, given the dataset I constructed, our model does an excellent job of predicting who will win MIP, as demonstrated by the metrics and model diagnostics (sensitivity, specificity, precision, model accuracy) shown above. Additionally, our model and exploratory analysis informs fantasy league managers to draft players with the following attributes:

- Players in their 20's
- Players in their first three years of the NBA
- Drafted late
- Put up sub-par numbers in the prior season

As George Box said, "All models are wrong, some are useful."

In the future, I ought to build a logistic regression model upon a more comprehensive dataset. The dataset I had found (on Kaggle) did have the yearly season stats of the thousands of

NBA players who have played in the league since 1950. However, because it did not have several variables that I wanted to assess for the model in draft pick, years of NBA experience, and whether the player switched teams, I decided against using this dataset. In other words, I forsook sample size for factors. Knowing what I know now after completing the analysis on my dataset, and with additional time, I aim to build a more accurate model and run the same analysis on the Kaggle dataset. I hope to find a more robust logistic regression model that will predict at least one front runner in this year's MIP race. Until then, I will fight to not be at the bottom of my fantasy league. But if being last is my lot, then I accept the milk mile, SAT, or community service as my fate.

References

1. (2023, 12 2). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/NBA_Most_Improved_Player_Award
2. (2023, 12 3). Retrieved from Yahoo Fantasy Sports Basketball: https://basketball.fantasysports.yahoo.com/nba/express_settings?type=head_point&guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAABchJtfIiqIOXdXFJnFeJMB9cMziJl09xHLiQmKKS_U9doUX2cPSqQ5NHsLRMyR7KRqZqRZrrC2gAaU0TsPqubB8jJZOxP
3. Bobbitt, Z. (2020, October 28). Retrieved from Statology: https://www.statology.org/logistic-regression-in-r
4. Li, Z. (2023, 10 13). Week 6: Multiple Logistic Regression.
5. Lloyd, J. (2023, 11 11). Retrieved from Youtube: https://www.youtube.com/watch?v=-lDrnrMY3mg&ab_channel=LockedOnFantasyBasketball

Appendix

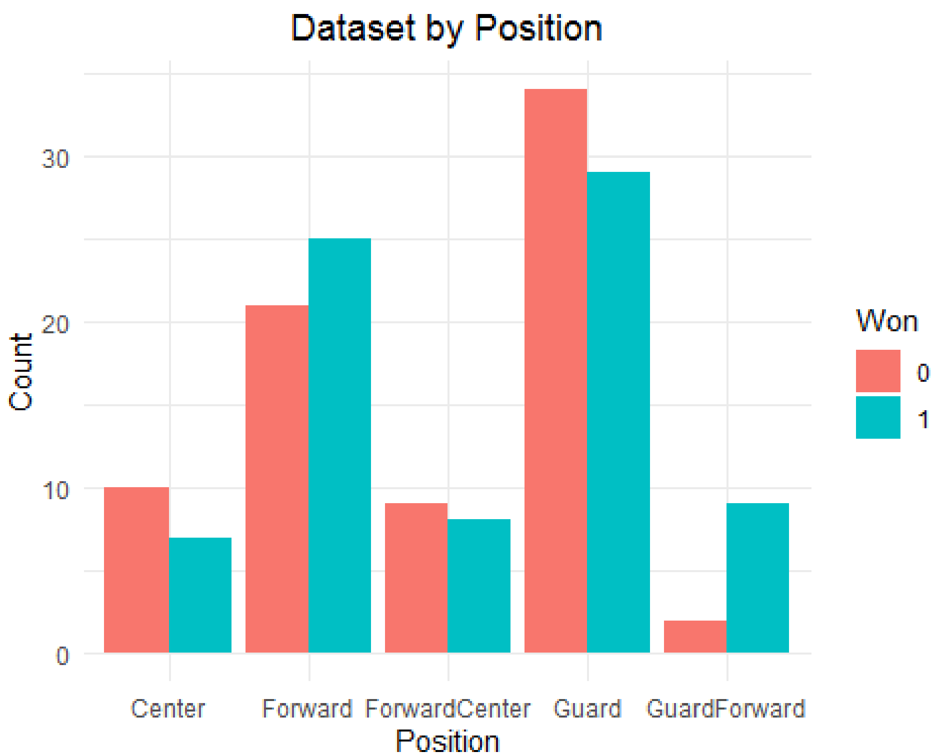
Note: The following output was created by R Markdown.

```
library(tidyverse)

## -- Attaching core tidyverse packages ------------------------ tidyverse 2.
0.0 --
## v dplyr     1.1.2     v readr     2.1.4
## v forcats   1.0.0     v stringr   1.5.0
## v ggplot2   3.4.2     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.0
## v purrr     1.0.1
## -- Conflicts ------------------------------------------ tidyverse_conflict
s() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all
 conflicts to become errors

library(ggplot2)
library(MASS)

##
## Attaching package: 'MASS'
##
## The following object is masked from 'package:dplyr':
```

```
##
##     select

library(caret)

## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift

library(RColorBrewer)
library(readxl)

MIP <- read_excel("C:/Users/Sam/Desktop/Baruch College/Advanced Data Analysis
 (Fall 2023)/Final Project/MIP.xlsx")
View(MIP)
#0. Data Prep
MIP$Won <- factor(MIP$Won) # convert it to factor data type
MIP$Position <- factor(MIP$Position) # convert it to factor data type
levels(MIP$Position)

## [1] "Center"        "Forward"       "ForwardCenter" "Guard"
## [5] "GuardForward"

MIP$TeamChange <- factor(MIP$TeamChange) # convert it to factor data type

############################################################
#I. Exploratory Data Analysis
############################################################
dim(MIP)

## [1] 154  23

#154 rows by 23 columns

#Summary
summary(MIP)

##       Year           Name              Won         MVP               ROY

##  Min.   :1985   Length:154         0:76   Min.   :0.0000   Min.   :0.0000

##  1st Qu.:1994   Class :character   1:78   1st Qu.:0.0000   1st Qu.:0.0000

##  Median :2004   Mode  :character          Median :0.0000   Median :0.0000

##  Mean   :2004                             Mean   :0.2468   Mean   :0.2532
```

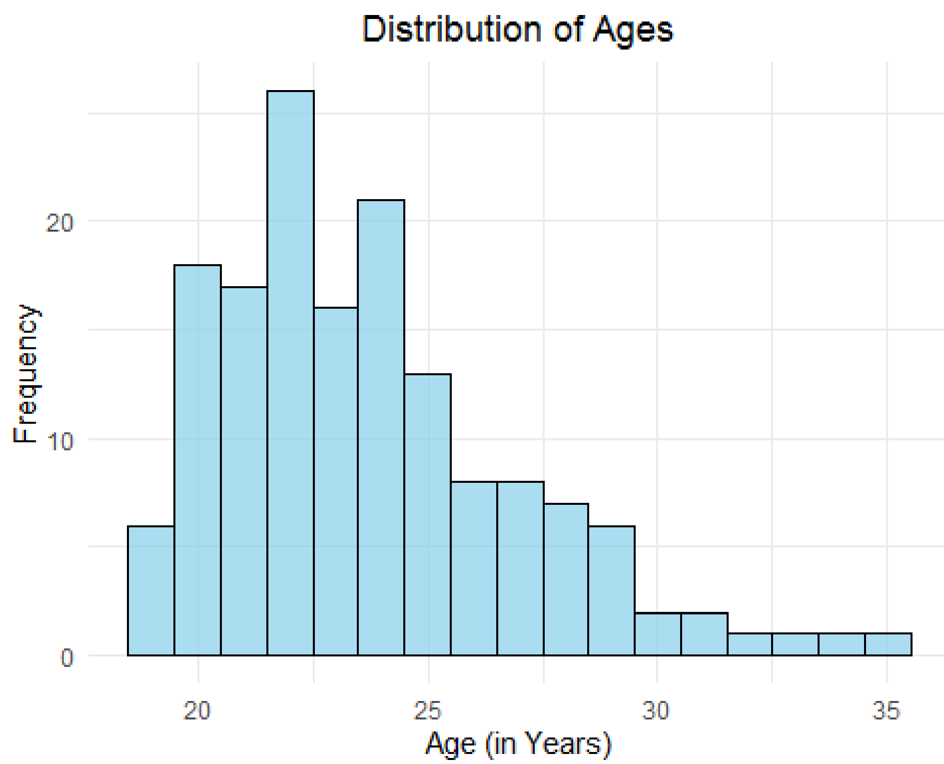
```
##  3rd Qu.:2013                            3rd Qu.:0.0000   3rd Qu.:0.7500

##  Max.   :2022                            Max.   :1.0000   Max.   :1.0000

##          Position        Age          DraftPick      TeamChange    YearsNBA

##  Center       :17   Min.   :19.0   Min.   : 1.00   No :142    Min.   : 1.0
00
##  Forward      :46   1st Qu.:21.0   1st Qu.: 2.00   Yes: 12    1st Qu.: 1.0
00
##  ForwardCenter:17   Median :23.0   Median : 7.00              Median : 2.0
00
##  Guard        :63   Mean   :23.7   Mean   :13.51              Mean   : 3.6
36
##  GuardForward :11   3rd Qu.:25.0   3rd Qu.:18.75              3rd Qu.: 5.7
50
##                     Max.   :35.0   Max.   :73.00              Max.   :14.0
00
##   GamesPlayed     GamesStarted        MPG            FGPerc
##  Min.   :27.00   Min.   : 0.00   Min.   : 8.30   Min.   :0.3720
##  1st Qu.:64.00   1st Qu.:31.25   1st Qu.:25.38   1st Qu.:0.4363
##  Median :76.00   Median :70.00   Median :33.50   Median :0.4680
##  Mean   :70.93   Mean   :55.92   Mean   :30.97   Mean   :0.4750
##  3rd Qu.:81.00   3rd Qu.:79.00   3rd Qu.:37.15   3rd Qu.:0.5075
##  Max.   :83.00   Max.   :82.00   Max.   :42.00   Max.   :0.6430
##   ThreePtPerc         FTPerc            RPG              APG
##  Min.   :0.0000   Min.   :0.3360   Min.   : 1.100   Min.   : 0.500
##  1st Qu.:0.2013   1st Qu.:0.7272   1st Qu.: 3.900   1st Qu.: 1.900
##  Median :0.3135   Median :0.7720   Median : 5.550   Median : 3.550
##  Mean   :0.2699   Mean   :0.7657   Mean   : 6.225   Mean   : 4.094
##  3rd Qu.:0.3670   3rd Qu.:0.8340   3rd Qu.: 8.075   3rd Qu.: 5.900
##  Max.   :0.5000   Max.   :0.9210   Max.   :13.900   Max.   :12.800
##       SPG              BPG               TO              PPG
##  Min.   :0.200   Min.   :0.0000   Min.   :0.500   Min.   : 3.90
##  1st Qu.:0.700   1st Qu.:0.2000   1st Qu.:1.525   1st Qu.:10.80
##  Median :1.100   Median :0.5000   Median :2.350   Median :16.15
##  Mean   :1.169   Mean   :0.7682   Mean   :2.356   Mean   :16.99
##  3rd Qu.:1.575   3rd Qu.:0.9000   3rd Qu.:3.100   3rd Qu.:23.38
##  Max.   :3.200   Max.   :3.9000   Max.   :5.400   Max.   :35.00
##    FantasyPTS
##  Min.   : 8.01
##  1st Qu.:22.40
##  Median :33.82
##  Mean   :34.06
##  3rd Qu.:43.44
##  Max.   :61.75

#Filter Data for Players Who Won MIP or were Runner-Ups
filteredData <- MIP[MIP$Won == 1, ]
View(filteredData)
```

```
#a. Position
summary(MIP$Position)

##        Center       Forward ForwardCenter         Guard  GuardForward
##            17            46            17            63            11

#Center: 17   Forward: 46     Forward/Center: 17     Guard: 63     Guard/Forw
ard: 11

ggplot(MIP, aes(x = Position, fill = Won)) +
  geom_bar(position = "dodge", stat = "count") +
  labs(title = "Dataset by Position", x = "Position", y = "Count") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```

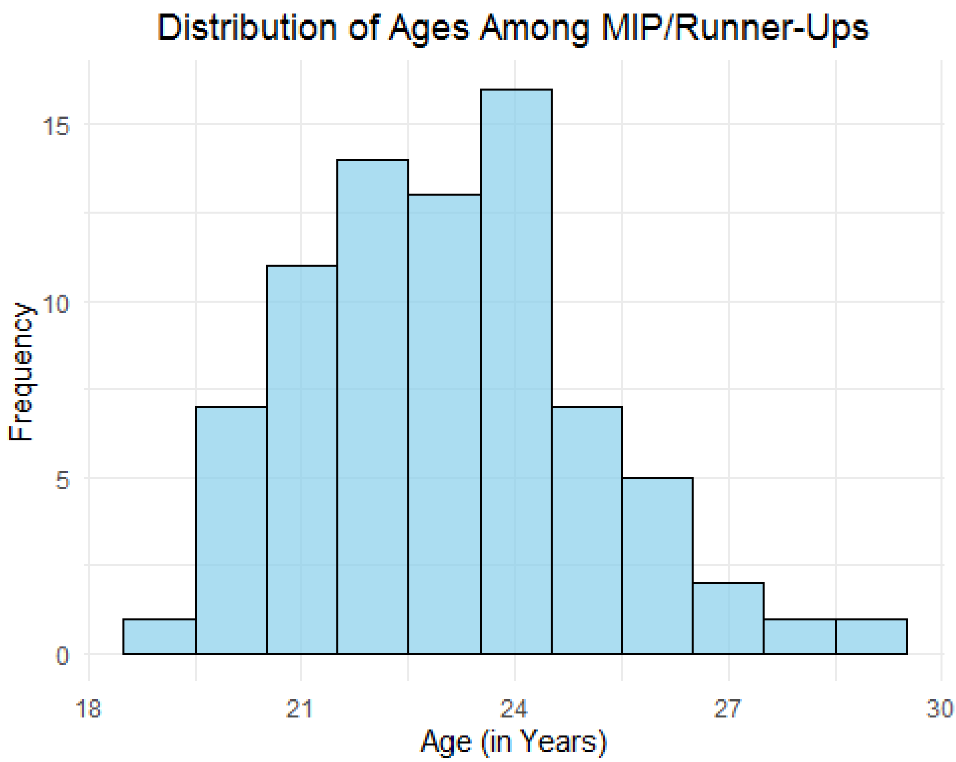



```
#As we can see, guards and then forwards have won the NBA MIP
#the most often.

#Let's examine the exact counts of those who won by position:
summary(filteredData$Position)

##        Center       Forward ForwardCenter         Guard  GuardForward
##             7            25             8            29             9

#Center: 7     Forward: 25     Forward/Center: 8     Guard: 29  Guard/Forward:
 9
```

```
#b. Age
summary(MIP$Age) #Min: 19 Max: 35

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    19.0    21.0    23.0    23.7    25.0    35.0

ggplot(MIP, aes(x = Age)) +
  geom_histogram(binwidth = 1, color = "black", fill = "skyblue", alpha = 0.7
) +
  labs(title = "Distribution of Ages", x = "Age (in Years)", y = "Frequency")
 +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```

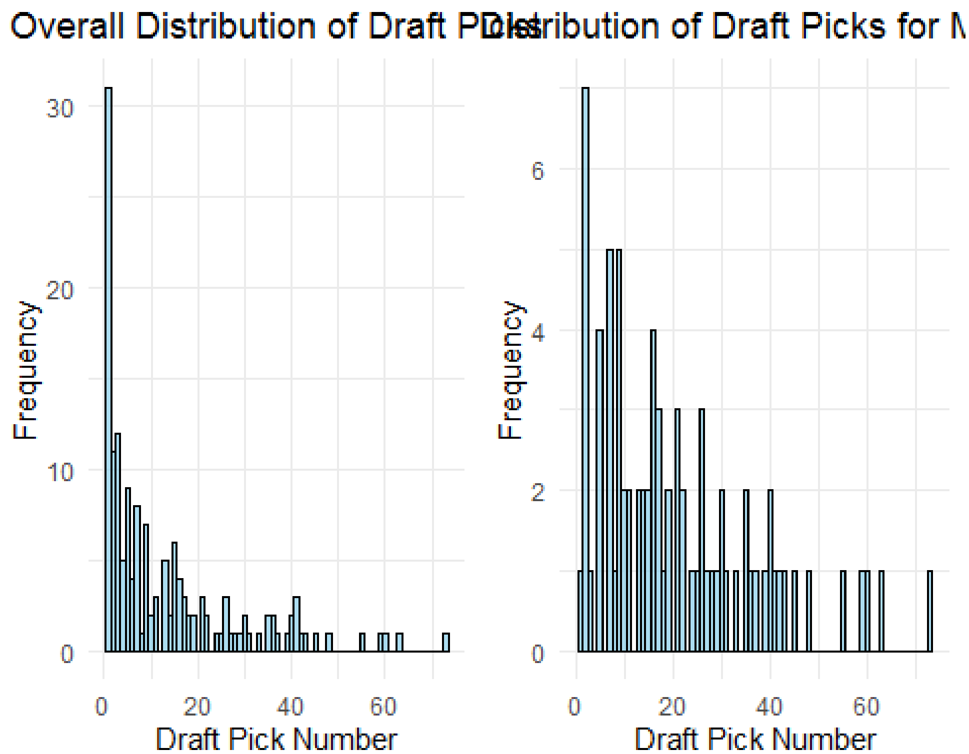



```
#Let's check the filtered dataset for the age distribution:
summary(filteredData$Age)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   19.00   22.00   23.00   23.04   24.00   29.00

ggplot(filteredData, aes(x = Age)) +
  geom_histogram(binwidth = 1, color = "black", fill = "skyblue", alpha = 0.7
) +
  labs(title = "Distribution of Ages Among MIP/Runner-Ups", x = "Age (in Year
s)", y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```

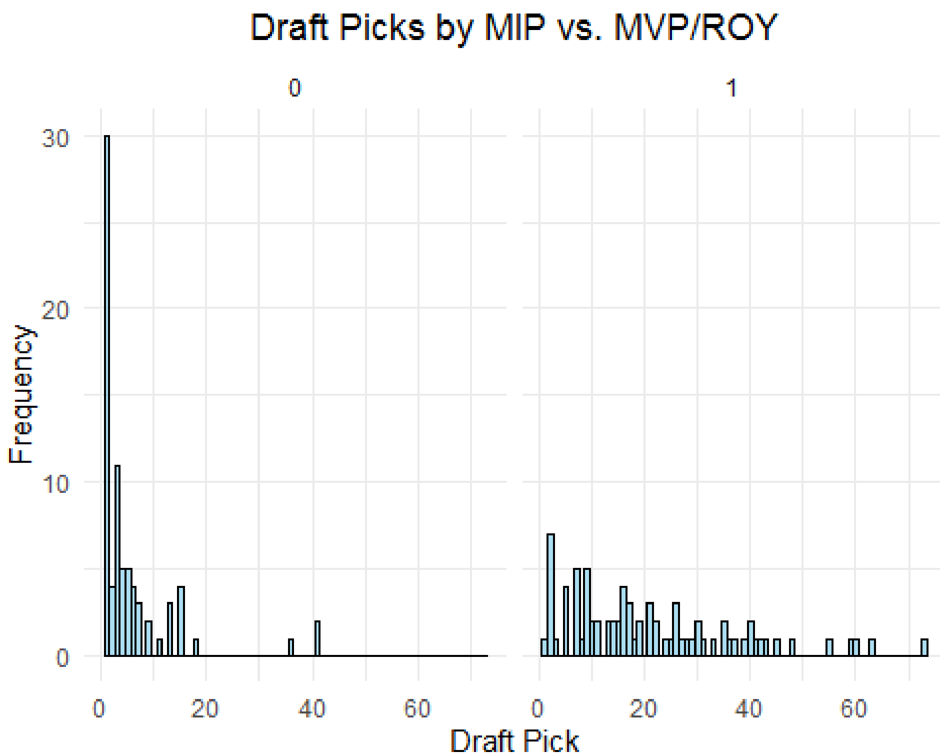



```
#We see that no player in their 30's has ever won the award.

#c. DraftPick
summary(MIP$DraftPick) #Min: 1 Max: 73 (undrafted)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    2.00    7.00   13.51   18.75   73.00

library(gridExtra)

##
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##     combine

ggplot1 <- ggplot(MIP, aes(x = DraftPick)) +
  geom_histogram(binwidth = 1, color = "black", fill = "skyblue", alpha = 0.7
) +
  labs(title = "Overall Distribution of Draft Picks", x = "Draft Pick Number"
, y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))

#Let's examine the distribution of draft picks after filtering for MIP winner
s
```

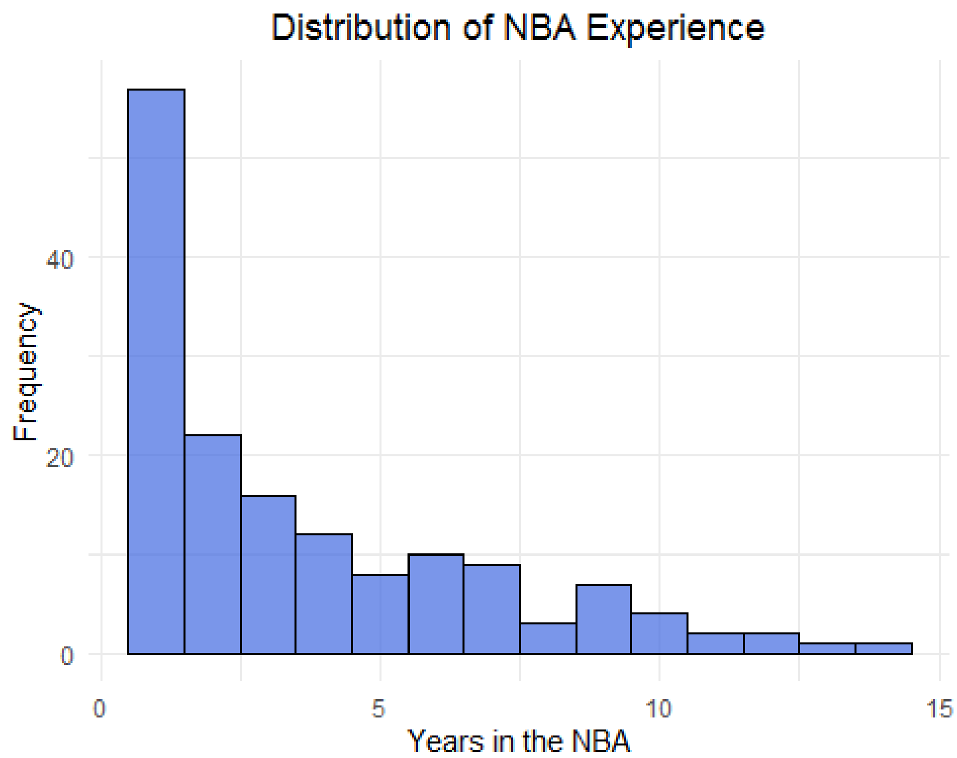
```
ggplot2 <- ggplot(filteredData, aes(x = DraftPick)) +
  geom_histogram(binwidth = 1, color = "black", fill = "skyblue", alpha = 0.7
) +
  labs(title = "Distribution of Draft Picks for MIPs", x = "Draft Pick Number
", y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))

grid.arrange(ggplot1, ggplot2, ncol = 2)
```



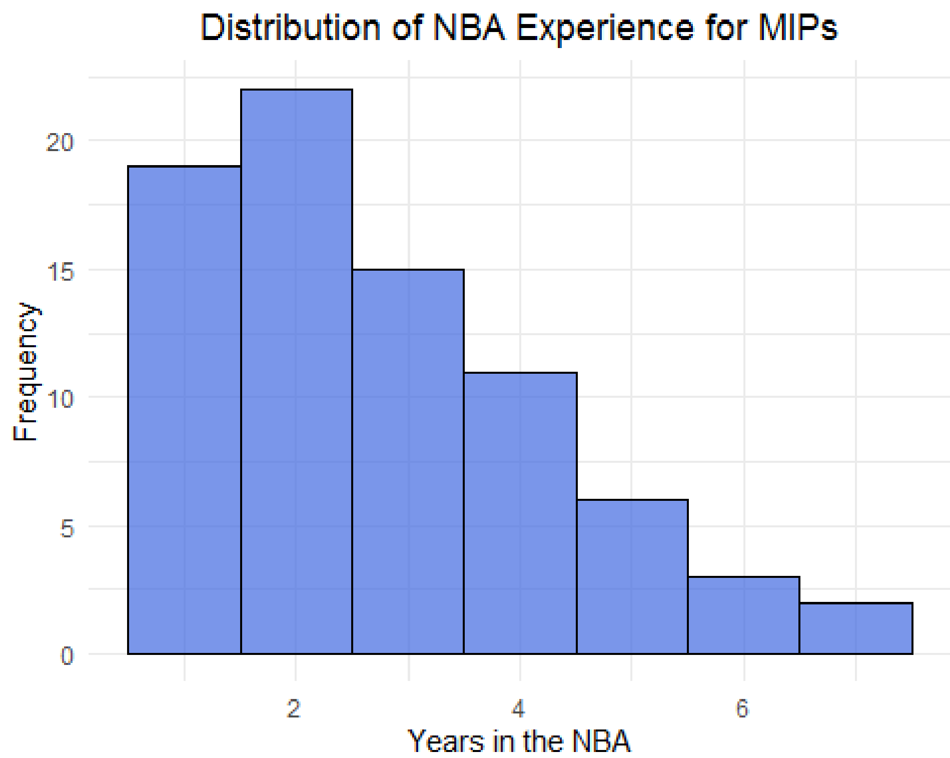

```
#Note that the distribution is skew right, meaning most players who win or
#were runner-ups for MIP were drafted earlier in their respective drafts
#(lower than 30)

#Draft Pick vs. Won?
ggplot(MIP, aes(x = DraftPick)) +
  geom_histogram(binwidth = 1, color = "black", fill = "skyblue", alpha = 0.7
) +
  labs(title = "Draft Picks by MIP vs. MVP/ROY", x = "Draft Pick", y = "Frequ
ency") +
  facet_wrap(~ Won) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```




```
#d. YearsNBA
summary(MIP$YearsNBA) #Min: 1 Max: 14

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   2.000   3.636   5.750  14.000

#Median: 2 Mean: 2.766
ggplot(MIP, aes(x = YearsNBA)) +
  geom_histogram(binwidth = 1, color = "black", fill = "royalblue", alpha = 0
.7) +
  labs(title = "Distribution of NBA Experience", x = "Years in the NBA", y =
"Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```




```
#Distribution of NBA experience
counts2 = table(MIP$YearsNBA)
counts2

##
##  1  2  3  4  5  6  7  8  9 10 11 12 13 14
## 57 22 16 12  8 10  9  3  7  4  2  2  1  1

#Let's examine MIP winners/runner-ups:
summary(filteredData$YearsNBA) #Min: 1 Max: 7

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   2.000   2.000   2.744   4.000   7.000

ggplot(filteredData, aes(x = YearsNBA)) +
  geom_histogram(binwidth = 1, color = "black", fill = "royalblue", alpha = 0
.7) +
  labs(title = "Distribution of NBA Experience for MIPs", x = "Years in the N
BA", y = "Frequency") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5))
```




```
#Hence, the most common players to win or be nominated for MIP are juniors
#(third years) followed by sophomores (second years) and seniors (fourth year
s).
#It is notable that the distribution is skew right, meaning most players who
#get nominated are relatively new to the NBA.

############################################################
#II. Logistic Regression (based on Week 6 Code)
############################################################
#a. Cross-Validation: Let's create a training set and a testing set.
#According to p. 24/25 of Week 6 Notes: in practice, we split the data manual
ly,
#which leads to training and testing sets. This strategy is called cross-vali
dation.

#I obtained this code from ChatGPT:
# Split the data into training and testing sets (80% for training, 20% for te
sting)
set.seed(123)  # Setting seed for reproducibility
train_index <- createDataPartition(MIP$Won, p = 0.8, list = FALSE)
training_set <- MIP[train_index, ]  # Training set
testing_set <- MIP[-train_index, ]  # Testing set

#Check the training and testing sets
View(training_set)
View(testing_set)
dim(training_set)  # Dimensions of training set
```

```
## [1] 124  23

dim(testing_set)   # Dimensions of testing set

## [1] 30 23

#b. Build the Logistic Regression Model
logisticModel = glm(Won~Position+Age+DraftPick+TeamChange+YearsNBA
                    +GamesPlayed+GamesStarted+MPG+FGPerc+ThreePtPerc+FTPerc+F
antasyPTS
                    ,data=training_set,family=binomial)
summary(logisticModel)

##
## Call:
## glm(formula = Won ~ Position + Age + DraftPick + TeamChange +
##     YearsNBA + GamesPlayed + GamesStarted + MPG + FGPerc + ThreePtPerc +
##     FTPerc + FantasyPTS, family = binomial, data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.84065  -0.17322   0.00718   0.19717   1.58124
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)             12.73495   10.32656   1.233  0.21749
## PositionForward          1.97302    3.19144   0.618  0.53643
## PositionForwardCenter    2.33733    3.16937   0.737  0.46083
## PositionGuard            0.34228    2.99821   0.114  0.90911
## PositionGuardForward     0.05628    3.72335   0.015  0.98794
## Age                     -0.59590    0.36930  -1.614  0.10662
## DraftPick                0.14070    0.05707   2.466  0.01368 *
## TeamChangeYes            1.86746    1.67768   1.113  0.26566
## YearsNBA                 0.60631    0.43685   1.388  0.16517
## GamesPlayed             -0.06135    0.04399  -1.394  0.16317
## GamesStarted            -0.04485    0.04648  -0.965  0.33448
## MPG                      0.21476    0.23637   0.909  0.36358
## FGPerc                  12.45840   17.11964   0.728  0.46678
## ThreePtPerc             -0.14113    5.35180  -0.026  0.97896
## FTPerc                   1.33638    9.45674   0.141  0.88762
## FantasyPTS              -0.31606    0.11304  -2.796  0.00517 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  46.535  on 108  degrees of freedom
## AIC: 78.535
##
## Number of Fisher Scoring iterations: 8
```

```
#There are two columns that are significant: DraftPick (p-value: 0.01368)
#and FantasyPTS (p-value: 0.00517)
#Position (GuardForward) has the largest p-value: 0.98794

#c. Model Selection (Backward Stepwise Selection)
#Let's remove Position (the variable with the largest p-value)
logisticModel1=update(logisticModel,~.-Position)
summary(logisticModel1)

##
## Call:
## glm(formula = Won ~ Age + DraftPick + TeamChange + YearsNBA +
##     GamesPlayed + GamesStarted + MPG + FGPerc + ThreePtPerc +
##     FTPerc + FantasyPTS, family = binomial, data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.63047  -0.18042   0.00949   0.19809   1.97992
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   16.11101    8.99803   1.791  0.07337 .
## Age           -0.65521    0.36147  -1.813  0.06989 .
## DraftPick      0.11241    0.04767   2.358  0.01837 *
## TeamChangeYes  0.90123    1.43731   0.627  0.53064
## YearsNBA       0.78508    0.40752   1.926  0.05404 .
## GamesPlayed   -0.04148    0.03969  -1.045  0.29607
## GamesStarted  -0.04151    0.03939  -1.054  0.29189
## MPG            0.23936    0.20457   1.170  0.24197
## FGPerc        13.66142   15.57237   0.877  0.38033
## ThreePtPerc   -0.48740    5.02947  -0.097  0.92280
## FTPerc        -2.54694    8.42196  -0.302  0.76233
## FantasyPTS    -0.34499    0.11092  -3.110  0.00187 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  49.658  on 112  degrees of freedom
## AIC: 73.658
##
## Number of Fisher Scoring iterations: 7

#Let's now remove ThreePtPerc (p-value: 0.92280)
logisticModel2=update(logisticModel1,~.-ThreePtPerc)
summary(logisticModel2)

##
## Call:
```

```
## glm(formula = Won ~ Age + DraftPick + TeamChange + YearsNBA +
##     GamesPlayed + GamesStarted + MPG + FGPerc + FTPerc + FantasyPTS,
##     family = binomial, data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.64230  -0.17532   0.00964   0.19742   1.97674
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  16.10390    8.98107   1.793  0.07296 .
## Age          -0.65136    0.35926  -1.813  0.06982 .
## DraftPick     0.11269    0.04751   2.372  0.01769 *
## TeamChangeYes  0.86535   1.38803   0.623  0.53300
## YearsNBA      0.78596    0.40755   1.928  0.05380 .
## GamesPlayed  -0.04173    0.03950  -1.057  0.29074
## GamesStarted  -0.04272   0.03742  -1.142  0.25363
## MPG           0.24453    0.19785   1.236  0.21648
## FGPerc       14.17088   14.70096   0.964  0.33507
## FTPerc       -3.17045    5.43757  -0.583  0.55985
## FantasyPTS   -0.34669    0.10990  -3.155  0.00161 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  49.667  on 113  degrees of freedom
## AIC: 71.667
##
## Number of Fisher Scoring iterations: 7

#Let's now remove FTPerc (p-value: 0.55985)
logisticModel3=update(logisticModel2,~.-FTPerc)
summary(logisticModel3)

##
## Call:
## glm(formula = Won ~ Age + DraftPick + TeamChange + YearsNBA +
##     GamesPlayed + GamesStarted + MPG + FGPerc + FantasyPTS, family = binom
ial,
##     data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.71590  -0.15627   0.01056   0.19618   1.83258
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  13.76939    7.83554   1.757   0.0789 .
```

```
## Age            -0.67435    0.35376  -1.906   0.0566 .
## DraftPick       0.10889    0.04558   2.389   0.0169 *
## TeamChangeYes   0.78797    1.33718   0.589   0.5557
## YearsNBA        0.79842    0.40449   1.974   0.0484 *
## GamesPlayed    -0.03783    0.03935  -0.961   0.3364
## GamesStarted   -0.04201    0.03715  -1.131   0.2582
## MPG             0.23529    0.19588   1.201   0.2297
## FGPerc         15.09387   14.51911   1.040   0.2985
## FantasyPTS     -0.34754    0.10945  -3.175   0.0015 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.87  on 123  degrees of freedom
## Residual deviance:  50.01  on 114  degrees of freedom
## AIC: 70.01
##
## Number of Fisher Scoring iterations: 7
```

```
#Next, let's remove TeamChange (p-value: 0.5557)
logisticModel4=update(logisticModel3,~.-TeamChange)
summary(logisticModel4)
```

```
##
## Call:
## glm(formula = Won ~ Age + DraftPick + YearsNBA + GamesPlayed +
##     GamesStarted + MPG + FGPerc + FantasyPTS, family = binomial,
##     data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.73054  -0.15787   0.00974   0.23831   1.82595
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  14.06863    7.88422   1.784 0.074358 .
## Age          -0.67657    0.34253  -1.975 0.048244 *
## DraftPick     0.10651    0.04560   2.336 0.019500 *
## YearsNBA      0.82881    0.39273   2.110 0.034828 *
## GamesPlayed  -0.03735    0.03845  -0.971 0.331334
## GamesStarted -0.04083    0.03568  -1.144 0.252517
## MPG           0.23480    0.19237   1.221 0.222261
## FGPerc       15.21685   14.50711   1.049 0.294213
## FantasyPTS   -0.35931    0.10774  -3.335 0.000853 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
```

```
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  50.387  on 115  degrees of freedom
## AIC: 68.387
##
## Number of Fisher Scoring iterations: 7

#Next, let's remove GamesPlayed (p-value: 0.331334)
logisticModel5=update(logisticModel4,~.-GamesPlayed)
summary(logisticModel5)

##
## Call:
## glm(formula = Won ~ Age + DraftPick + YearsNBA + GamesStarted +
##     MPG + FGPerc + FantasyPTS, family = binomial, data = training_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9064  -0.1955   0.0107   0.1986   1.7567
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  12.30617    7.47902   1.645 0.099882 .
## Age          -0.65422    0.33446  -1.956 0.050462 .
## DraftPick     0.10345    0.04522   2.288 0.022160 *
## YearsNBA      0.80930    0.37864   2.137 0.032565 *
## GamesStarted -0.05973    0.03222  -1.854 0.063706 .
## MPG           0.25859    0.18948   1.365 0.172344
## FGPerc       12.07355   13.08651   0.923 0.356218
## FantasyPTS   -0.34148    0.10042  -3.401 0.000672 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  51.331  on 116  degrees of freedom
## AIC: 67.331
##
## Number of Fisher Scoring iterations: 7

#Let's remove FGPerc (p-value: 0.356218)
logisticModel6=update(logisticModel5,~.-FGPerc)
summary(logisticModel6)

##
## Call:
## glm(formula = Won ~ Age + DraftPick + YearsNBA + GamesStarted +
##     MPG + FantasyPTS, family = binomial, data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
```

```
## -2.93493  -0.20091   0.01026   0.18934   1.86838
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  16.34480    6.25691   2.612  0.00899 **
## Age         -0.55245    0.29891  -1.848  0.06458 .
## DraftPick    0.10104    0.04562   2.215  0.02678 *
## YearsNBA     0.70832    0.34069   2.079  0.03761 *
## GamesStarted -0.04988    0.02982  -1.673  0.09437 .
## MPG          0.18464    0.17268   1.069  0.28494
## FantasyPTS  -0.29723    0.08684  -3.423  0.00062 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  52.281  on 117  degrees of freedom
## AIC: 66.281
##
## Number of Fisher Scoring iterations: 7
```

```
#Let's remove MPG (p-value: 0.28494)
logisticModel7=update(logisticModel6,~.-MPG)
summary(logisticModel7)
```

```
##
## Call:
## glm(formula = Won ~ Age + DraftPick + YearsNBA + GamesStarted +
##     FantasyPTS, family = binomial, data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.80671  -0.25855   0.01119   0.15841   2.08044
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  17.50603    6.14680   2.848   0.0044 **
## Age         -0.45401    0.27825  -1.632   0.1028
## DraftPick    0.08065    0.03777   2.136   0.0327 *
## YearsNBA     0.60886    0.32176   1.892   0.0585 .
## GamesStarted -0.02646    0.01950  -1.357   0.1747
## FantasyPTS  -0.24538    0.06251  -3.925 8.67e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  53.513  on 118  degrees of freedom
```

```
## AIC: 65.513
##
## Number of Fisher Scoring iterations: 7
```

```
#Let's remove GamesStarted (p-value: 0.1747)
logisticModel8=update(logisticModel7,~.-GamesStarted)
summary(logisticModel8)
```

```
##
## Call:
## glm(formula = Won ~ Age + DraftPick + YearsNBA + FantasyPTS,
##     family = binomial, data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.57349  -0.23155   0.01818   0.21007   2.01467
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) 17.23628    6.00525   2.870   0.0041 **
## Age         -0.46688    0.27111  -1.722   0.0850 .
## DraftPick    0.07780    0.03933   1.978   0.0479 *
## YearsNBA     0.62638    0.31320   2.000   0.0455 *
## FantasyPTS  -0.27781    0.06019  -4.616 3.91e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  55.474  on 119  degrees of freedom
## AIC: 65.474
##
## Number of Fisher Scoring iterations: 7
```

```
#Let's remove Age (p-value: 0.0850)
logisticModel9=update(logisticModel8,~.-Age)
summary(logisticModel9)
```

```
##
## Call:
## glm(formula = Won ~ DraftPick + YearsNBA + FantasyPTS, family = binomial,
##     data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -2.93610  -0.29044   0.02564   0.20736   2.22717
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.79552    1.95876   3.980 6.90e-05 ***
```

```
## DraftPick    0.06142    0.03956   1.552    0.121
## YearsNBA     0.13719    0.12199   1.125    0.261
## FantasyPTS  -0.26267    0.05881  -4.466 7.96e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  58.704  on 120  degrees of freedom
## AIC: 66.704
##
## Number of Fisher Scoring iterations: 7

#Let's remove YearsNBA (p-value: 0.261)
logisticModel10=update(logisticModel9,~.-YearsNBA)
summary(logisticModel10)

##
## Call:
## glm(formula = Won ~ DraftPick + FantasyPTS, family = binomial,
##     data = training_set)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.07070  -0.29158   0.02363   0.21245   2.14483
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  7.27627    1.79977   4.043 5.28e-05 ***
## DraftPick    0.07624    0.03787   2.013   0.0441 *
## FantasyPTS  -0.23879    0.05137  -4.649 3.34e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 171.868  on 123  degrees of freedom
## Residual deviance:  59.907  on 121  degrees of freedom
## AIC: 65.907
##
## Number of Fisher Scoring iterations: 7

#Our model says that of the original variables, DraftPick (p-value: 0.0441)
#and FantasyPTS (p-value: 3.34e-06) are the two statistically significant var
iables
#of whether a player will win MIP. With an estimated coefficient of 0.07624,
an
#increase of one unit in DraftPick will increase the log odds that a player w
ill
```

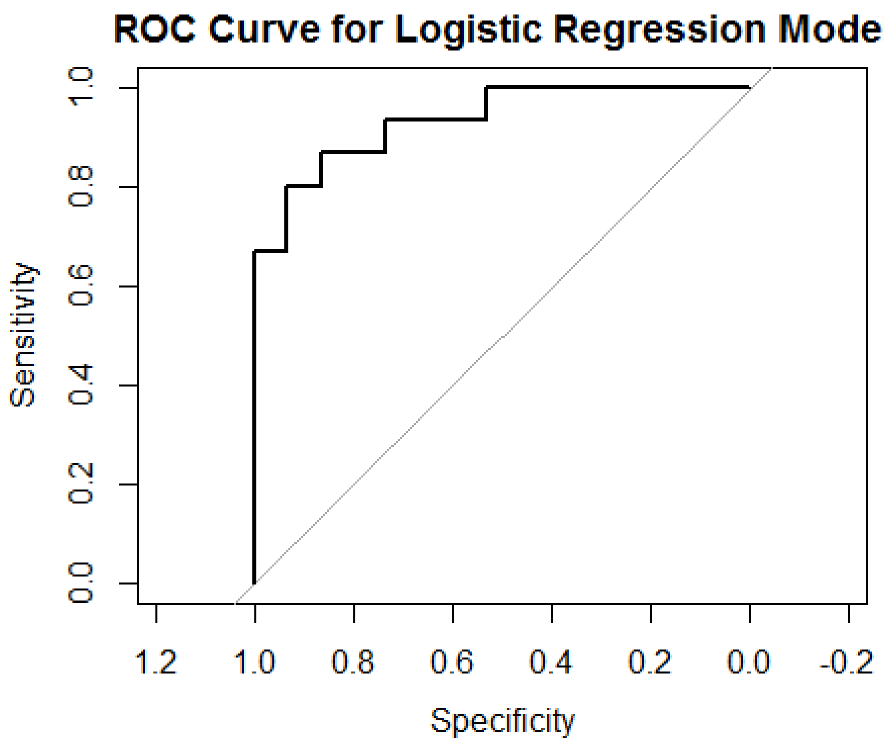
```
#win MIP by an average of 0.07624. For FantasyPTS, with an estimated coeffici
ent of
#-0.23879, an increase of one #unit in FantasyPTS, decreases (since it's nega
tive)
#the log odds that a player will win MIP by an average of 0.23879.

#d. Check for Model Strength/Predictive Power
#According to https://www.statology.org/logistic-regression-in-r/, we can
#compute McFadden's R^2 to assess our model's predictive power. Values over 0
.40
#indicate that a model fits the data very well.
#install.packages('pscl')
library(pscl)

## Classes and Methods for R developed in the
## Political Science Computational Laboratory
## Department of Political Science
## Stanford University
## Simon Jackman
## hurdle and zeroinfl functions by Achim Zeileis

pscl::pR2(logisticModel10)["McFadden"]

## fitting null model for pseudo-r2

##  McFadden
## 0.6514361

#We get a value of 0.6514361 which indicates that our model fits the data ver
y
#well and has high predictive power.

#e. VarImp (Variable Importance)
varImp(logisticModel10)

##              Overall
## DraftPick  2.013103
## FantasyPTS 4.648834

#Overall
#DraftPick  2.013103
#FantasyPTS 4.648834
#This matches up with the p-values from earlier.
#FantasyPts is the more important predictor and then DraftPick.

#f. VIF
car::vif(logisticModel10)

##  DraftPick FantasyPTS
##   1.000012   1.000012
```

```r
#DraftPick: 1.000012
#FantasyPTS: 1.000012
#Since neither column has a VIF over 5, we conclude that multicollinearity
#is not a problem in our model.

#g. Predictions
#Define NBA player (player who was not drafted high with low fantasy points i
n
#a season). This player is Jose Alvarado of the New Orleans Pelicans.
new <- data.frame(DraftPick = 61, FantasyPTS = 18.86)

#Predict probability of winning MIP
predict(logisticModel10, new, type="response")

##         1
## 0.9994034

#Our model predicts that Jose Alvarado has a 0.9998335 probability of winning
#the 2024 MIP award.

#h. Test Dataset
#Calculate probability of Won for each individual in test dataset
predicted <- predict(logisticModel10, testing_set, type="response")
predicted

##           1           2           3           4           5           6
## 0.925913048 0.547482348 0.009364797 0.008414500 0.099829572 0.919417899
##           7           8           9          10          11          12
## 0.884723116 0.906236258 0.107792447 0.180228697 0.239001776 0.976013314
##          13          14          15          16          17          18
## 0.002664880 0.994989900 0.095091135 0.623199598 0.673268802 0.448242860
##          19          20          21          22          23          24
## 0.979899053 0.477156474 0.007391458 0.282352586 0.994753133 0.505669128
##          25          26          27          28          29          30
## 0.063020520 0.001007751 0.993891446 0.278322072 0.107433979 0.417105233

#1            2            3            4            5            6            7
#0.925913048 0.547482348 0.009364797 0.008414500 0.099829572 0.919417899 0.88
4723116
#8            9            10           11           12           13           14
#0.906236258 0.107792447 0.180228697 0.239001776 0.976013314 0.002664880 0.99
4989900
#15           16           17           18           19           20           21
#0.095091135 0.623199598 0.673268802 0.448242860 0.979899053 0.477156474 0.00
7391458
#22           23           24           25           26           27           28
#0.282352586 0.994753133 0.505669128 0.063020520 0.001007751 0.993891446 0.27
8322072
#29          30
#0.107433979 0.417105233
```

```
predictedBinary <- ifelse(predicted >= 0.5, 1, 0)
predictedBinary

##  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25
 26
##  1  1  0  0  0  1  1  1  0  0  0  1  0  1  0  1  1  0  1  0  0  0  1  1  0
  0
## 27 28 29 30
##  1  0  0  0

typeof(predictedBinary)

## [1] "double"

############################################################
#III. Model Diagnostics
############################################################
#a. Confusion Matrix
#Based on https://www.statology.org/logistic-regression-in-r/
#Any player in our test dataset with a probability of Won
#greater than 0.5 will be predicted to be MIP/runner-up.
testing_set$Won

##  [1] 1 1 0 0 0 1 1 1 0 0 0 1 0 1 0 0 1 1 1 0 0 0 1 1 0 0 1 1 1 0
## Levels: 0 1

testing_set$Won <- factor(testing_set$Won)
predictedBinary <- factor(predictedBinary)
confusionMatrix(testing_set$Won, predictedBinary)

## Confusion Matrix and Statistics
##
##             Reference
## Prediction  0  1
##          0 14  1
##          1  3 12
##
##                Accuracy : 0.8667
##                  95% CI : (0.6928, 0.9624)
##     No Information Rate : 0.5667
##     P-Value [Acc > NIR] : 0.0004563
##
##                   Kappa : 0.7333
##
##  Mcnemar's Test P-Value : 0.6170751
##
##             Sensitivity : 0.8235
##             Specificity : 0.9231
##          Pos Pred Value : 0.9333
##          Neg Pred Value : 0.8000
##              Prevalence : 0.5667
```

```
##          Detection Rate : 0.4667
##    Detection Prevalence : 0.5000
##       Balanced Accuracy : 0.8733
##
##        'Positive' Class : 0
##

#
#             Reference
#Prediction  0  1
#         0 14  1
#         1  3 12

#p. 21/25 of Week 6
#Sensitivity: the conditional probability that the test is positive
#given the player won MIP.
#Sensitivity = True Positive/(True Positive + False Negative)
#Sensitivity = 12/(12+3) = 0.8

#Specificity = True Negative/(True Negative + False Positive)
#Specificity = 14/(14+1) = 0.9333

#Precision Rate = True Positive/(True Positive + False Positive) = 12/(12+1)
#Precision Rate = 0.9231

#Model Accuracy = (True Positive + True Negative)/Total = (12+14)/30
#Accuracy rate (overall fraction of correct predictions) = 0.8667

#b. ROC Curve
library(pROC)

## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var

roc_data <- roc(testing_set$Won, predicted)

## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

roc_data

##
## Call:
## roc.default(response = testing_set$Won, predictor = predicted)
```

```
##
## Data: predicted in 15 controls (testing_set$Won 0) < 15 cases (testing_set
$Won 1).
## Area under the curve: 0.9333

plot(roc_data, main = "ROC Curve for Logistic Regression Model")
```




```
auc(roc_data) #Area under the curve: 0.9333

## Area under the curve: 0.9333

#According to p. 23/25 of Week 6, AUC is a scalar that represents
#the area under the ROC curve. A value of 0.5 indicates a model that
#performs no better than a random guess whereas a value of 1 indicates
#a perfect model that correctly classifies all instances. Given our
#AUC value of 0.9333, our model does a good job of predicting whether
#a player will win MIP.

############################################################
#IV. Predictions
############################################################
library(readxl)
MIP2024 <- read_excel("C:/Users/Sam/Desktop/Baruch College/Advanced Data Anal
ysis (Fall 2023)/Final Project/MIP2024.xlsx")
View(MIP2024)
predictions <- predict(logisticModel10, newdata = MIP2024, type = "response")
predictions
```

```
##           1           2           3           4           5           6 
## 0.002603035 0.002321794 0.017685648 0.019407790 0.009524642 0.011556712 
##           7           8           9          10          11          12 
## 0.032858238 0.018942255 0.009257763 0.115193135 0.128199167 0.031990936 
##          13          14          15          16          17          18 
## 0.078729352 0.047762329 0.026325444 0.051393033 0.006427447 0.135417785 
##          19          20          21          22          23          24 
## 0.076291082 0.108450831 0.345371926 0.106893319 0.067900802 0.157425103 
##          25          26          27          28          29          30 
## 0.051531044 0.233931758 0.634401499 0.137044365 0.185739201 0.039617921 
##          31          32          33          34          35          36 
## 0.072383071 0.176478098 0.357235925 0.418348637 0.834686625 0.620756183 
##          37          38          39          40          41          42 
## 0.213037124 0.924809771 0.719759198 0.692628544 0.470783841 0.846939302 
##          43          44          45          46          47          48 
## 0.029423695 0.335299749 0.964749274 0.133589355 0.085262828 0.531055786 
##          49          50          51          52          53          54 
## 0.923375467 0.891534763 0.241247934 0.833551102 0.670265526 0.466278137 
##          55          56          57          58          59          60 
## 0.567764306 0.271218244 0.339148384 0.993631694 0.176526967 0.618822257 
##          61          62          63          64          65          66 
## 0.910767518 0.353906546 0.724073551 0.056038595 0.658817497 0.164306800 
##          67          68          69          70          71          72 
## 0.639673639 0.833784211 0.362092136 0.932765718 0.972777749 0.578739553 
##          73          74          75          76          77          78 
## 0.989379690 0.925645010 0.921860261 0.996729665 0.970532630 0.986194442 
##          79          80          81          82          83          84 
## 0.965292989 0.979651793 0.911997775 0.991168562 0.988955581 0.999396206 
##          85          86          87          88          89          90 
## 0.993760616 0.992643638 0.999403368 0.956483782 0.993180897 0.998773377 
##          91          92          93          94 
## 0.998208283 0.995105904 0.998086037 0.998339147

typeof(predictions) #double

## [1] "double"

#from ChatGPT:
topTen <- head(sort(predictions, decreasing = TRUE), 10)
print(topTen)

##        87        84        90        94        91        93        76 
   92 
## 0.9994034 0.9993962 0.9987734 0.9983391 0.9982083 0.9980860 0.9967297 0.99
51059 
##        85        58 
## 0.9937606 0.9936317

#According to our model, Jose Alvarado, Jae'Sean Tate, Terance Mann, Christia
n Braun, Royce O'Neale, Moses Moody, Cam Thomas, Jalen Johnson, Daniel Gaffor
d, Louis King have the highest odds of winning MIP 2024.
```