Megan Zhao's Final Project!

Predicting 2021 MLB Postseason MVPs

Megan Zhao

2025-04-23

(a) Data Selection

I took MLB pitching, batting, and fielding from the 2021 season from Baseball-Reference.com. I chose the 2021 postseason MLB data because 2021 World Series was so fun to watch— I remember I had time to watch every game because of COVID. Everyone was wearing their pearls for Joc Pederson, and I was supporting ATL (the champions) since I was born in Georgia and the Guardians were out, but Eddie Rosario had just transferred from CLE to ATL. I filtered the data to only include players who played in the 2021 world series.

(b) Question formulation

In this project, I will predict the world series MVP of the 2021 MLB playoffs based on batting and fielding statistics. I watch a good bit of baseball, but primarily for vibes. I thought I could learn more about the nuances of baseball and what makes one player stand out from the rest.

Baseball has many statistics, and one player's performance often depends on multiple parts including batting and running on offense and fielding of the ball on defense. A player can bat a grand slam in a deciding game, have a high RBI throughout the series, or turn key defensive plays, all of which are important to what makes a valuable player to their team's success in post-season. I am assuming that a player's overall performance in such areas are a good indicator of their value to a team, thus improving their chance of being named MVP.

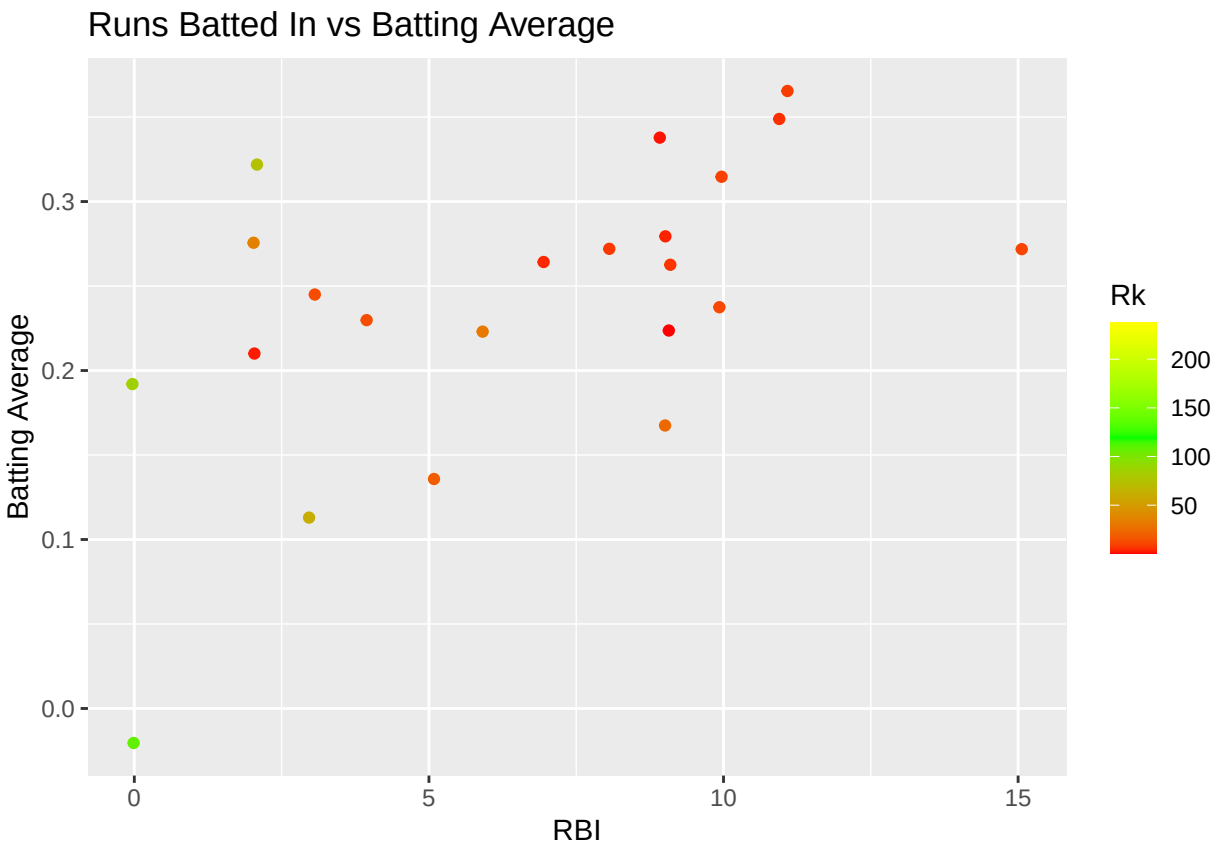
Overall, I want to discern which metrics are important to predicting which player will win a Championship Series MVP, and create/evaluate models to predict an MVP.

Pitchers and position players are evaluated quite differently. Because the Cy Young award exists for pitchers, very few pitchers win MVP; therefore, I am choosing to only evaluate batting/fielding stats for position players to win MVP. The MLB awards 3 MVP awards in the postseason: ALCS MVP, NLCS MVP, and World Series MVP. I filtered the data to only include WS players even though the ALCS and NLCS have other games, since MVPs are usually chosen from the finals teams. This also allows for a more focused analysis and direct comparison between players who competed at the same stage of the tournament.

(c) Exploratory Analysis

```
# batting average vs rank
ggplot(data = PObatting, aes(x = RBI, y = BA, col = Rk)) +
  geom_jitter(width = 0.1, height = 0.05) +
  labs(title = "Runs Batted In vs Batting Average",
       x = "RBI",
```

```
      y = "Batting Average") +
  scale_color_gradientn(colors = c("red", "green", "yellow"))
```

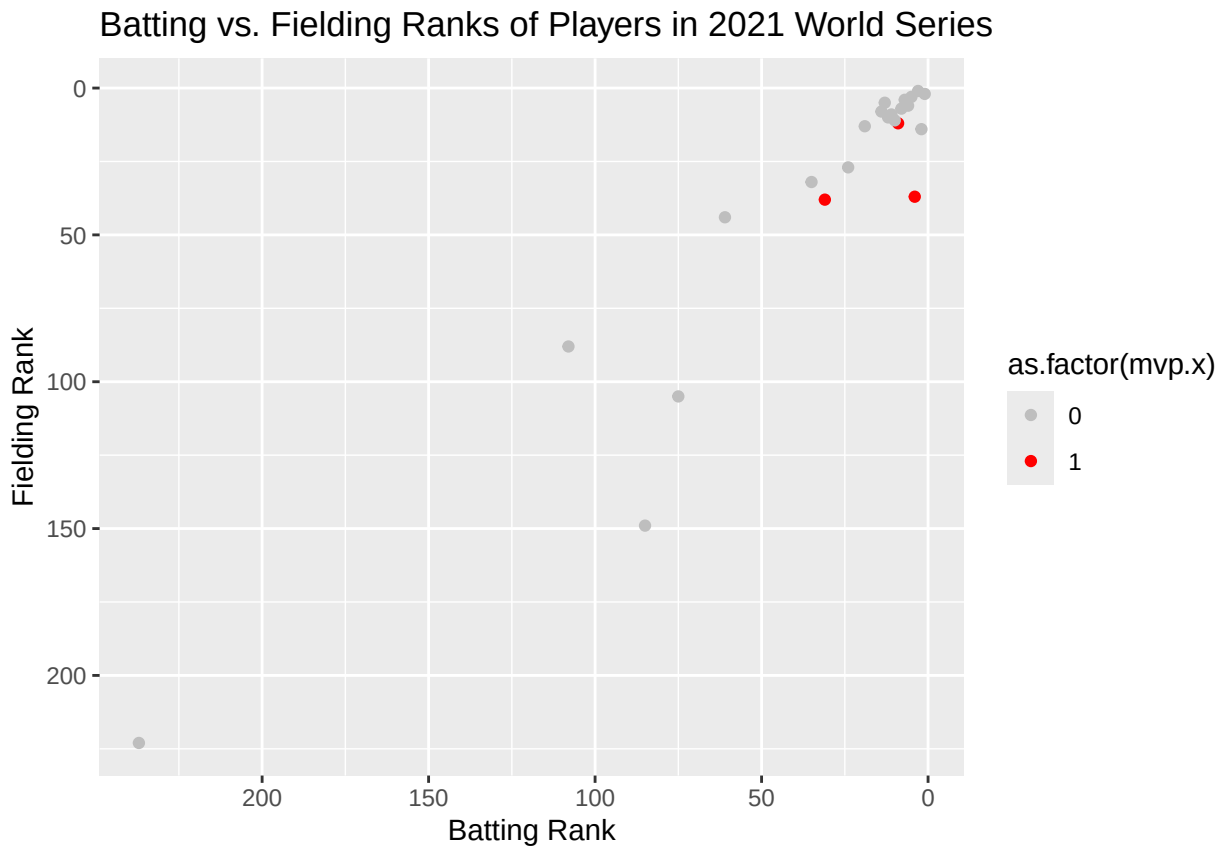



```
# offensive rank vs defensive rank
ranksubset <- left_join(PObatting %>% select(Player, Rk, mvp),
                        POfielding %>% select(Player, Rk, mvp),
                        by = "Player")
```

In the first plot: Batting Average (Hits/At-bats) vs Runs Batted In (RBI = when the at-bat results in a run scored), there is a slight to moderate positive correlation between batting average and runs batted in. This suggests correlation in these and similar batting statistics, but the moderate variance of BA vs RBI tells us that the player's contribution to the game— in turns of helping to score— likely also depends on factors beyond hitting/batting average (such as their ability to hit in influential situations (ie runners on base), their spot in the batting lineup, batting strategy, etc). Also looking at how relatively high-ranked/similarly-ranked batters are spread with a decent amount of variability in terms of both RBI and BA, we can discern that our offensive predictors for high performers (candidates for MVP) may require more nuance than the detail that batting average can provide.

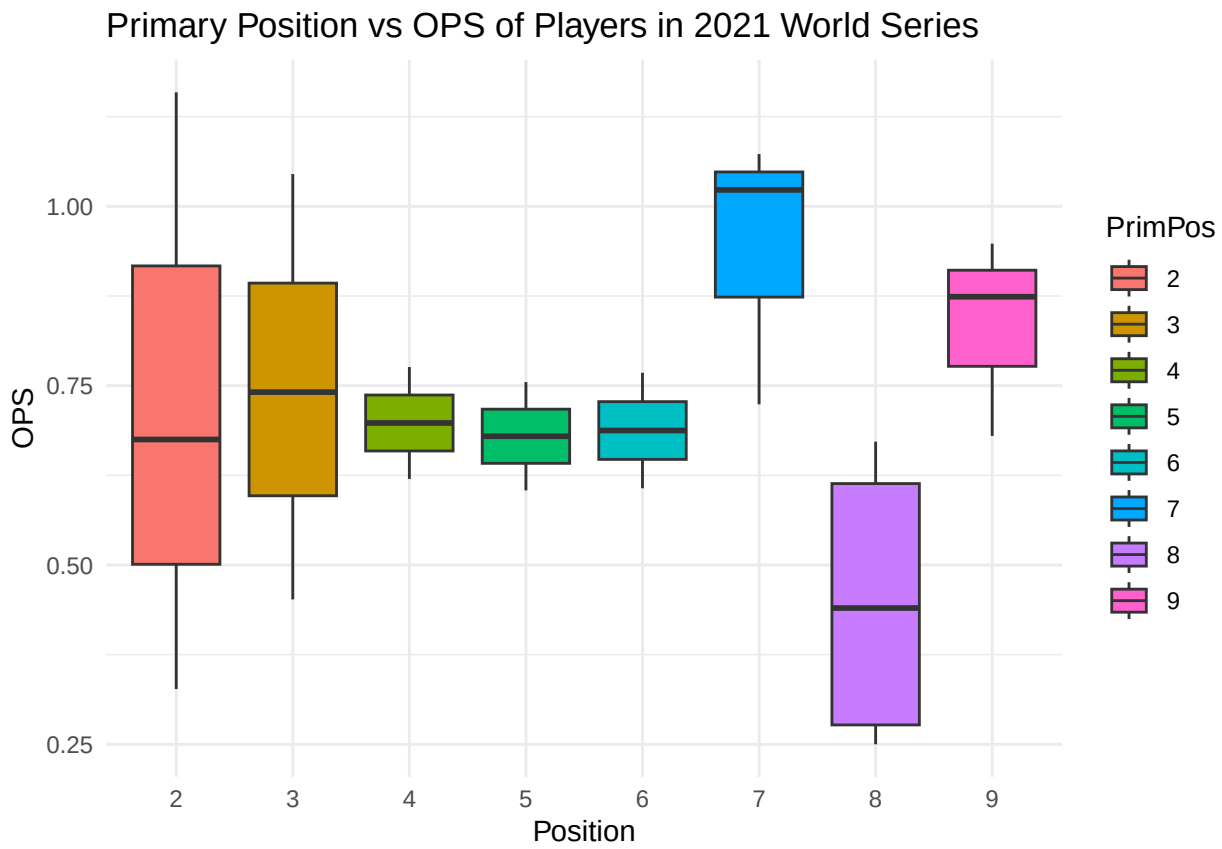
```
ggplot(ranksubset, aes(x = Rk.x, y = as.numeric(Rk.y), color = as.factor(mvp.x))) +
  geom_point() +
  labs(x = "Batting Rank",
       y = "Fielding Rank",
       title = "Batting vs. Fielding Ranks of Players in 2021 World Series") +
  scale_x_reverse() +
```

```
  scale_y_reverse() +
  scale_color_manual(values = c("1" = "red", "0" = "gray"))
```



Looking at Batting vs Fielding rank, we can see a pretty strong, clear positive correlation between a player's offensive and defensive performance at ranks above 50. Though this pattern may not always be the case, this makes sense as we may expect the strongest players in one aspect of the game to also be relatively strong overall/in other parts of their game. The 3 MVPs from the 2021 postseason are marked in red, detailing that MVPs tend to be high-ranking in both batting and fielding; therefore, ranking, or some adjusted ranking, may be a good surrogate for a player's chance of winning MVP. We see that there is more uncaptured nuance beyond the batting and fielding rankings, though, since the actual top ranked batters and fielders were not awarded an MVP award in 2021.

```
# OPS vs Position
ggplot(PObatting %>% filter(PrimPos>1), aes(x = PrimPos, y = OPS, fill = PrimPos)) +
  geom_boxplot() +
  labs(x = "Position",
       y = "OPS",
       title = "Primary Position vs OPS of Players in 2021 World Series") +
  theme_minimal()
```



OPS is On Base (OBP) + Slugging percentage (SLG), which adds how often a player reaches base and how many bases a player gets per at-bat. OPS is a statistic that can effectively represent a player's power and ability to get on base, which are arguably two of the most important offensive skills. We can see that the infield players (catcher (2), and positions 3-6) have relatively similar median OPS, while the outfield positions (7-9) differ quite significantly from the infield OPSs and from each other. From this we might think that players may systematically play different roles (offensively) dependent on their position or offer a different skill set that may aid a team's success on different criteria, which would be interesting to consider when modeling a player's contribution to team success to predict MVP.

(d) ANALYSIS

Linear/Mixed effects models

```
# Model 1: linear regression
m1 <- lm(ranksum ~ OPS + `Fld%`+ PrimPos + Inn + Age + Team, data = wsplayers)
summary(m1)
```

```
##
## Call:
## lm(formula = ranksum ~ OPS + `Fld%` + PrimPos + Inn + Age + Team,
##     data = wsplayers)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

```
## -38.475 -14.368  -2.854   8.854  46.844
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 203.2881   709.6276   0.286 0.781002
## OPS         -37.8554    39.2610  -0.964 0.360137
## `Fld%`      -39.8936   675.5751  -0.059 0.954202
## PrimPos3     15.0200    28.4695   0.528 0.610547
## PrimPos4    -10.6611    42.3752  -0.252 0.807010
## PrimPos5      0.1784    45.5810   0.004 0.996963
## PrimPos6      4.1202    41.3319   0.100 0.922778
## PrimPos7    -27.7848    32.7130  -0.849 0.417706
## PrimPos8    -18.7762    32.3666  -0.580 0.576074
## PrimPos9    -21.7511    33.6559  -0.646 0.534230
## Inn          -1.1888     0.1860  -6.393 0.000126 ***
## Age           1.7400     2.9957   0.581 0.575610
## TeamHOU     -10.7291    15.9178  -0.674 0.517228
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 34.77 on 9 degrees of freedom
## Multiple R-squared:  0.8824, Adjusted R-squared:  0.7257
## F-statistic:  5.63 on 12 and 9 DF,  p-value: 0.007155
```

```
#simplified
m2 <- lm(ranksum ~ OPS + `Fld%`+ Inn, data = wsplayers)
summary(m2)
```

```
##
## Call:
## lm(formula = ranksum ~ OPS + `Fld%` + Inn, data = wsplayers)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -65.034 -12.186   5.624  15.432  62.969
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 283.6748   482.1596   0.588    0.564
## OPS         -41.7498    27.6880  -1.508    0.149
## `Fld%`      -89.0758   480.9858  -0.185    0.855
## Inn          -1.1184     0.1482  -7.544 5.59e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 31.02 on 18 degrees of freedom
## Multiple R-squared:  0.8129, Adjusted R-squared:  0.7817
## F-statistic: 26.07 on 3 and 18 DF,  p-value: 9.035e-07
```

```
#simplified + team random effects
m3 <- lmer(ranksum ~ OPS + `Fld%` + Inn + (1 | PrimPos) + (1 | Team), data = wsplayers)
```

```
## boundary (singular) fit: see help('isSingular')
```

```r
summary(m3)
```

```
## Linear mixed model fit by REML ['lmerMod']
## Formula: ranksum ~ OPS + `Fld%` + Inn + (1 | PrimPos) + (1 | Team)
##    Data: wsplayers
##
## REML criterion at convergence: 183.5
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -2.0890 -0.3781  0.1785  0.4782  2.0055
##
## Random effects:
##  Groups   Name        Variance  Std.Dev.
##  PrimPos  (Intercept) 1.159e+01 3.405e+00
##  Team     (Intercept) 5.813e-07 7.624e-04
##  Residual             9.514e+02 3.084e+01
## Number of obs: 22, groups:  PrimPos, 8; Team, 2
##
## Fixed effects:
##             Estimate Std. Error t value
## (Intercept) 283.9294   480.9703   0.590
## OPS         -41.6669    27.7360  -1.502
## `Fld%`      -89.2972   479.9175  -0.186
## Inn          -1.1192     0.1476  -7.582
##
## Correlation of Fixed Effects:
##        (Intr) OPS    `Fld%`
## OPS     0.051
## `Fld%` -0.999 -0.089
## Inn    -0.476 -0.160  0.457
## fit warnings:
## Some predictor variables are on very different scales: consider rescaling
## optimizer (nloptwrap) convergence code: 0 (OK)
## boundary (singular) fit: see help('isSingular')
```

```r
# BICs
BIC(m1)
```

```
## [1] 242.1895
```

```r
BIC(m2)
```

```
## [1] 224.5942
```

```r
BIC(m3)
```

```
## [1] 205.15
```

```
cor_matrix <- wsplayers %>%
select(OPS, `Fld%`, Inn, Age) %>%
cor()
cor_matrix
```

```
##             OPS        Fld%        Inn         Age
## OPS  1.00000000  0.01427540  0.1387146  0.03203858
## Fld% 0.01427540  1.00000000 -0.4556981 -0.01059597
## Inn  0.13871457 -0.45569811  1.0000000  0.02110470
## Age  0.03203858 -0.01059597  0.0211047  1.00000000
```

```
#predictions
wsplayers.lp <- wsplayers
wsplayers.lp$predm3 <- predict(m3)
# Top 8 lowest predicted ranksum
top8possiblemvp <- wsplayers.lp %>%
  arrange(predm3) %>%
  select(predm3, Player, Team) %>%
  slice_head(n = 8)
top8possiblemvp
```

```
## # A tibble: 8 x 3
##   predm3 Player          Team
##    <dbl> <chr>           <chr>
## 1 -4.56  Freddie Freeman ATL
## 2  0.556 Eddie Rosario   ATL
## 3  1.10  Kyle Tucker     HOU
## 4  6.08  Carlos Correa   HOU
## 5  6.47  Austin Riley    ATL
## 6  9.08  Jose Altuve     HOU
## 7 11.4   Travis d'Arnaud ATL
## 8 12.7   Adam Duvall     ATL
```

First, we will use a linear regression model to predict (batting rank + fielding rank) as a proxy variable for overall player contribution (and the highest rank(s) (lowest nominal value) contribution would be MVP). We are adding the rankings together to capture both the offensive and defensive contribution of players, while also maintaining the importance of "extreme" performances (if a player is extremely strong offensively much lower ranked in fielding, their overall summed rank may still reflect a more noticeable contribution).

We will start with the predictors OPS (described previously), fielding percentage, primary position, number of innings played, age, and team as a random effect (could capture variability in systematic team effects like coaching, culture, history in tournament/matchups, etc). Variables were chosen to avoid multicolinearity (see correlation matrix, all correlation < 0.7) and capture various parts of players' performance.

In model 1, the simple linear regression, we see that the only significant (alpha $= 0.05$) predictor is the number of innings; for every additional inning played, the model predicts a decrease of 1.1888 in a player's aggregated batting/fielding ranking holding all else constant. Intuitively, a player who plays more innings is likely to be better at their role/less replaceable, which adds to their value on the team. Because no other variable were significantly impacting players' sum ranking, I chose to simplify the model in model 2. Innings were still the only significant predictor, but the BIC of the model (224.59) decreased from that of model 1 (BIC $= 242.19$). Therefore, model 2 is a better fit to the data after penalizing for complexity.

In model 3, I added in random effects of team and position to account for the fact that players from different teams or positions might have systematic differences in their performance that cannot be explained solely

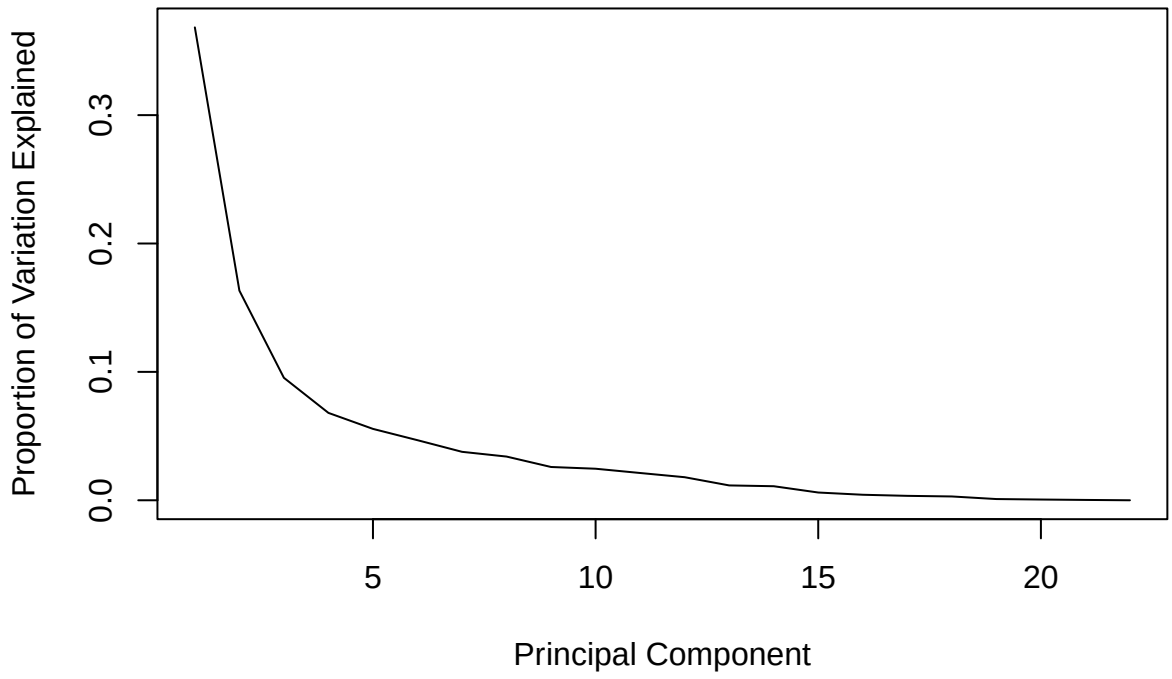
by the fixed effects. Different positions have different expectations/baselines for performance in different measurements, while roster, team dynamics, and team legacy could lead to higher rank sum values. We see a random effects variance of 11.59 for position, meaning there is a decent amount of variability in ranksum at the position level, but the variance for team is extremely small (possibly because there are only 2 teams in the world series). Though the BIC for model 3 (205.15) is the smallest here suggesting the best balance between fit and complexity, the residual variance of 9.514e+02 tells us that there is significant unexplained variability in predicting the summed rank of players after considering predictors and our random effects.

Using model 3, our model predicted lowest ranksums for Freddie Freeman (ATL) (though a negative sum rank does not make interpretable sense), Eddie Rosario (ATL), Kyle Tucker (HOU), and Carlos Correa (HOU). Based on our model, I would predict that the NLCS/ALCS MVPs are one of those players based on their post-season statistics.

Unsupervised Learning (PCA)

```
prsub <- wsp %>% select(where(is.numeric)) %>% select(-c(Rk.x, Rk.y, ranksum, PB, SB.y, CS.y, `CS%`))
rownames(prsub) <- wsp$Player
pr_out <- prcomp(prsub, center = TRUE, scale. = TRUE)

# elbow
plot(1:length(pr_out$sdev), summary(pr_out)$importance[2,], xlab="Principal Component",
ylab="Proportion of Variation Explained", type="l")
```

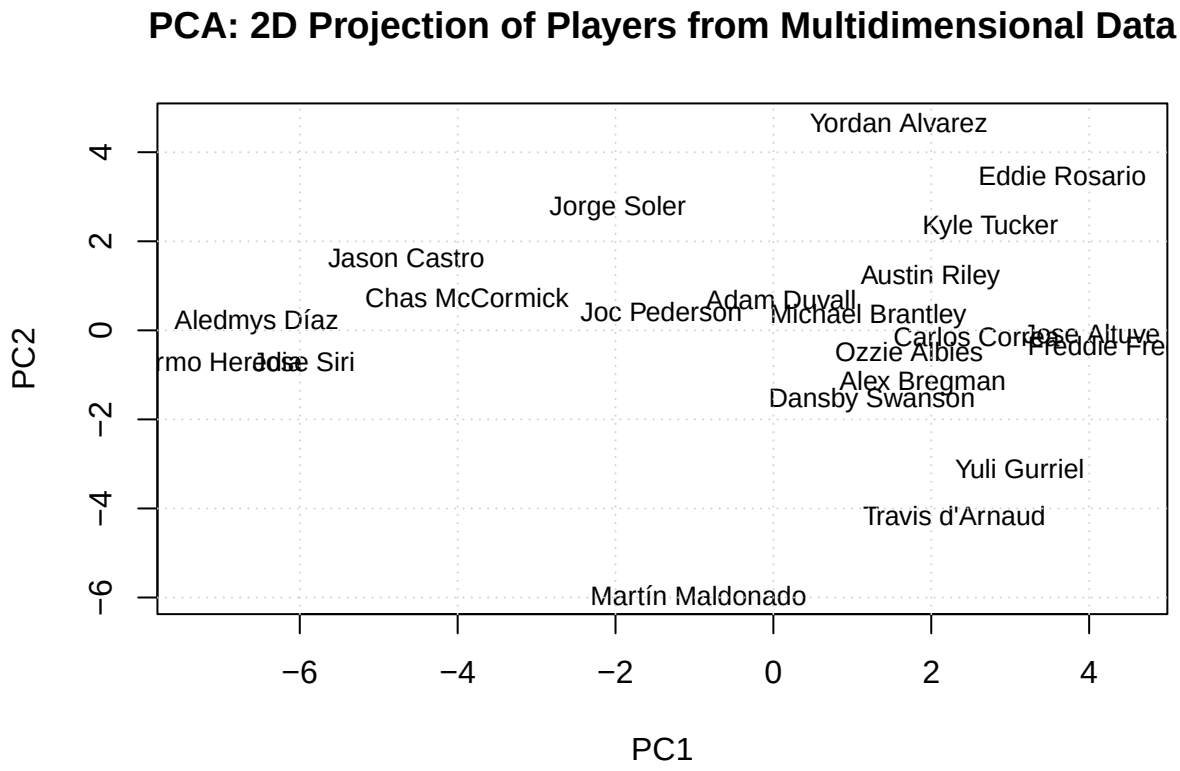


```r
summary(pr_out)$importance[3,]
```

```
##     PC1     PC2     PC3     PC4     PC5     PC6     PC7     PC8     PC9    PC10
## 0.36835 0.53164 0.62703 0.69510 0.75071 0.79751 0.83524 0.86927 0.89519 0.91975
##    PC11    PC12    PC13    PC14    PC15    PC16    PC17    PC18    PC19    PC20
## 0.94100 0.95897 0.97053 0.98147 0.98751 0.99180 0.99524 0.99820 0.99916 0.99974
##    PC21    PC22
## 1.00000 1.00000
```

```r
# pr_out$rotation

# PCA Scatterplot
plot(pr_out$x[,1], pr_out$x[,2],
xlab = "PC1", ylab = "PC2",
main = "PCA: 2D Projection of Players from Multidimensional Data",
col = "white", pch = 16)
text(pr_out$x[,1], pr_out$x[,2],
labels = rownames(prsub), col = "black", cex = 0.8)
grid()
```



Principal components analysis can help us cluster players based on numerical performance metrics from both batting and fielding data. The elbow plot show that around 5 PCs capture most of the variance within the dataset (could be interpreted from around PC4-PC7 too). In the 2D PCA plot, we see a few pairs and small clusters. We can see Yordan Alvarez and Eddie Rosario near each other in the top right representing the AL/NL MVPs. There are quite a few batting (including plate appearances, at bats, total bases) predictors

contributing to the first 2 principal components which comprise of most of the variability, but not any one variable to a largely significant degree.

Supervised Learning (PCR)

```
#ranksum as dependent variable
pcrsub1 <- wsp %>% select(where(is.numeric)) %>% select(-c(Rk.x, Rk.y, PB, SB.y, CS.y, `CS%`))
rownames(pcrsub1) <- wsp$Player
m4 <- pcr(ranksum~., center=TRUE, scale=TRUE, data=pcrsub1)
m4$coefficients[, , 1:6]
```

```
##            1 comps    2 comps    3 comps    4 comps    5 comps     6 comps
## Age     0.08485401 -0.4873785  1.8676931  1.6229464  1.8121536  1.39566367
## G      -4.03413755 -4.2228437 -5.0246439 -6.3886180 -6.3760501 -6.54103240
## PA     -4.37393469 -4.3836631 -5.0277697 -5.9135322 -5.9389850 -6.01492295
## AB     -4.29171140 -4.3458633 -5.1459526 -5.9164056 -6.0665749 -6.21288969
## R      -3.57518066 -3.2653761 -4.1553062 -3.2640113 -2.9723860 -2.72937489
## H      -4.13947569 -3.8979335 -3.9587013 -4.2388310 -4.5434700 -4.84134272
## `2B`   -3.05745312 -2.5007784 -3.2077214 -3.5283592 -4.0829129 -3.82896139
## `3B`   -0.83625139 -0.1683552  0.3501276 -1.6480029 -0.9719988 -1.19213001
## HR     -2.65684359 -2.2540386 -1.7945550 -1.7490894 -1.4556813 -1.19998123
## RBI    -3.38069784 -2.9989555 -2.7625557 -3.8930818 -4.0043797 -4.25062620
## SB.x   -1.80357059 -1.7060089 -2.3459578 -0.4786268 -1.1104562 -1.18432698
## CS.x   -1.03735922 -0.6327750  0.0589354  0.7723080  0.9458993  0.11453589
## BB     -3.33724114 -2.9553714 -2.1999135 -2.4136051 -1.8776615 -1.51486536
## SO     -3.40392179 -3.5835458 -4.0886395 -5.8902193 -5.9723201 -5.63949252
## BA     -3.01980033 -2.4770155 -0.9980750 -0.4495564 -0.7125954 -0.86103948
## OBP    -1.97038511 -1.2768163  0.8082870  1.5958117  1.7220417  1.78640768
## SLG    -2.38722149 -1.6588925  0.1812692  0.7147858  0.7264553  0.92011742
## OPS    -2.37057825 -1.6182413  0.3927232  1.0306843  1.0787234  1.24091805
## TB     -4.12301841 -3.6662330 -3.5693972 -3.9381286 -4.0199416 -4.06882298
## GIDP   -2.24232722 -2.7422243 -2.1190385 -3.3352933 -3.5597916 -3.40530776
## HBP    -0.05662878 -0.3217406 -1.3932758 -3.9735771 -3.3266584 -3.24172287
## SH     -0.05265329 -0.5270154 -0.1572205 -3.0801821 -3.6143067 -3.96522723
## SF     -0.92505965 -1.0015174 -1.0649383 -3.2982680 -2.2849280 -1.98472403
## IBB    -1.69203802 -2.0067586 -1.5022489  0.1046787  0.6004795  0.04723114
## GS     -3.94195007 -4.3636966 -5.1882302 -5.5738729 -5.7476851 -5.92616545
## CG     -4.01963202 -4.3725036 -5.2217531 -4.3400668 -4.5508918 -4.56681220
## Inn    -3.95278786 -4.3541524 -5.1793288 -5.3517659 -5.5347288 -5.71160057
## Ch     -2.36218357 -3.2832367 -1.9887510 -1.9899223 -2.0364638 -1.86503644
## PO     -1.69114320 -2.5512388 -0.5055996 -1.0767060 -1.1501060 -1.11349945
## A      -2.12873454 -2.3446675 -4.5573100 -2.8966577 -2.8474413 -2.40274547
## E      -2.09549584 -2.7669877 -3.2317255 -1.3187619 -0.7636022 -1.02999599
## DP     -2.41913418 -2.7740032 -2.8758250 -0.6537751 -0.4786964 -0.13631791
## `Fld%`  1.93561574  2.2124528  3.5965644  1.6830552  0.9246168  1.26582326
## `RF/9` -0.40127834 -1.1587043  1.3870467  2.2427405  2.2231371  2.52898078
## `RF/G` -1.95234976 -2.8661311 -1.2411133 -1.0945514 -1.1382337 -0.88723411
```

```
predictpcr4 <- predict(m4)[,,6]
predictpcr4
```

```
##       Jose Altuve  Michael Brantley      Ozzie Albies    Yordan Alvarez
```

```
##        -13.716995          7.683874         20.697259         32.981945
##      Alex Bregman   Freddie Freeman     Carlos Correa      Austin Riley
##          7.957841          3.828418          2.958350          5.362086
##     Eddie Rosario      Yuli Gurriel       Kyle Tucker       Adam Duvall
##         16.929607         26.968686          6.956351         34.358492
##   Travis d'Arnaud    Dansby Swanson  Martín Maldonado      Joc Pederson
##         17.594900         27.431844         32.174239         63.129102
##       Jorge Soler    Chas McCormick         Jose Siri      Jason Castro
##         100.803924        119.556448        143.737577        180.096235
##      Aledmys Díaz Guillermo Heredia
##         168.908388        161.601430
```

```
#RMSE using 6 PCs to predict ranksum
sqrt(mean((predictpcr4-wsp$ranksum)^2))
```

```
## [1] 23.51245
```

Using 6 PCs (using all numeric predictors) to maximize the significant portion of variance explained by the model with minimal complexity, we see that Plate appearances/At-bats, total bases, and complete games in a single position are more influential to the predicted rank sum. These variables from both batting and fielding notably attest to a player's consistency and versatility more than what aspects of their game make them stand out as position players. However, the model had an RMSE of 23. 5, so on average our predictions are off by 23.5 ranks on the scale of summed ranks. Our ranks can range from (2-257 (lowest rank in world series players dataset is 108 for batting, 149 for fielding)), so deviating 23.5 is not terrible, especially since ranksums tend to fall into clusters/ranges rather than increasing/decreasing smoothly.

Our model really likes Carlos Correa and Freddie Freeman, modeling at the low rank sums (2.96 and 3.83, respectively). Based on their post season stats, our model would nominate those players as ALCS/NLCS MVPs.

(e) RECOMMENDATIONS/FINDINGS

Starting with PCR prediction: Our PCR predicted Carlos Correa (HOU) as the player with the biggest case to win the ALCS. The actual winner was Yordan Alvarez (HOU). The model overlooked Alvares, predicting a ranksum of (~33) for him. However, Correa won the AL gold glove for SS, was named to All-star reserves, and was 7th in the AL for runs scored. According to Wikipedia, the Houston chapter of the Baseball Writers' Association of America (BBWAA) named Correa the Astros' team Most Valuable Player in 2021. Though our PCR model was wrong about MLB's ALCS MVP choice, we can still say that the model did a good job at identifying important combinations of features that make an valuable player.

Freddie Freeman was a favorite for MVP according to our PCR and mixed model predictions. He is a crazy player and was probably in talks surround most influential players during the 2021 post season. In 2021, he won a Silver Slugger Award Babe Ruth Award and had key HRs during the post season. Again, our models were still able to identify key players in the post season.

Our linear mixed-effects model: Eddie Rosario was the actual NLCS MVP choice, and I was rooting for him as I had watched him play for Cleveland for a majority of the regular season prior. Our linear mixed-effects model did a good job in highlighting his role that season, and our PCR model also gave him a relatively low rank sum.

The actual MVP of the World Series was Jorge Soler which both key models overlooked. This isn't too surprising as our data only consisted of post season averages, so there was likely to be uncaptured nuance not only from specific series, but specific plays in key games. For Soler (Braves), he hit three go-ahead home

runs the the World Series Game 6, and the Braves ended up winning the WS that game. Such contributions are difficult to pull out from a pool of other consistent and high-performing players of the course of many games and series.

Overall, our rank sum variable is an imperfect proxy for who deserved to win MVP, but our model predictions could still be useful to nominate players for the many awards that the MLB awards at the end of each season (ie gold gloves, all stars) for consistent and impactful performance. Determining MVP highly considers situational nuance and subjective comparison in context that are more nuanced than our models.