My goal in this exercise is to produce an ordered ranking of how well the players are expected to perform in 2018. First I'll take a look at the dataset to get an idea of the different variables and their data types.

```
dataset <-  read.csv("TRACKMAN_DATASET_Baseball.csv")
```

My first idea for evaluating these players was to get an idea of how their PA outcomes compared. I calculated their AVG/OBP/SLG so that I could have a better understanding of their hitting profiles, as well their relative effectiveness. From here on, I attempted to keep the players in this same order with their anonymized names.

```
outcomes <- table(dataset$batter_id, dataset$outcome)
outcomes <- as.data.frame.matrix(outcomes)

# PA's are estimated to be the sum of outs, hits, walks, and HBP
outcomes$PA <- rowSums(outcomes[-c(7)])

#AB's are estimated to be the sum of outs, hits, and RBOE
outcomes$AB <- rowSums(outcomes[c(1,2,3,6,8)])

outcomes$H <- rowSums(outcomes[c(1,2,3,6)])
outcomes$TB <- with(outcomes, `1B`*1 + `2B`*2 + `3B`*3 + `HR` *4)

# AVG = H / AB
outcomes$AVG <- round(rowSums(outcomes[13] / outcomes[12]), 3)

# OBP estimated to be  (H+BB+HBP) / (AB+BB+HBP+OUT-SAC) I realize that OUT-SAC and SF are not
necessarily the same, but given the small number of them, and the limited information present
in the RESULT column, I chose to use SF and OUT-SAC interchangably here.

table(dataset$outcome=="OUT-SAC", dataset$result)

##
##             B   BIP  BUIP     F    FB    GB   HBP    LD    PU     S
##   FALSE 12874     4    21  5079  1403  2420    65  1279   411  8719
##   TRUE      0     0     1     0    38     0     0     8     1     0

outcomes$OBP <- round(rowSums(outcomes[c(4,5,13)]) / rowSums(outcomes[c(13,4,5,8)]),3)

# SLG = TB / AB
outcomes$SLG <- round(rowSums(outcomes[14] / outcomes[12]), 3)

outcomes

##                                   1B 2B 3B  BB HBP HR NULL  OUT OUT-SAC
## 26c2b53e65d15d855499513ae12fbb00 113 66  0 126  19 65 3722  784       9
## 35e5fc499d77f0cf1cab4e822f02a915 242 90  8 276  11 76 6226 1251      11
## 485ba029f9c20c2e7fa216b449ceefaa 216 69  1 133   6 37 3705  884       5
## 52f0a588b4443a0343fb29f63c97f53a 242 74  6 164  25 42 5051  986      13
## ede7784ac9ec41e851bc7539b7f2d94c 358 88  6 165   4 63 5581 1246      10
##                                 RBOE   PA   AB   H  TB   AVG   OBP   SLG
## 26c2b53e65d15d855499513ae12fbb00    4 1186 1028 244 505 0.237 0.332 0.491
## 35e5fc499d77f0cf1cab4e822f02a915   10 1975 1667 416 750 0.250 0.360 0.450
## 485ba029f9c20c2e7fa216b449ceefaa    6 1357 1207 323 505 0.268 0.343 0.418
## 52f0a588b4443a0343fb29f63c97f53a   12 1564 1350 364 576 0.270 0.359 0.427
## ede7784ac9ec41e851bc7539b7f2d94c   16 1956 1761 515 804 0.292 0.354 0.457
```

Next, I looked at the linear weights posted on fangraphs for the 2015-2017 seasons (http://www.fangraphs.com/guts.aspx?type=cn) and calculated the relative value of each players' PA outcomes with no context. I made a judgement call to apply the mean weight of the past three seasons here, rather than subdividing performance for all three seasons and applying those weights three times for each player. My reasoning is that the players couldn't know the relative value of the different plays and their context-neutral impact on the relative probability of scoring an additional run, and thus giving them the tiniest extra credit or debit would not provide additional insight.

I also wanted to know who had the highest maximum exit velocity on a batted ball (exit_velo). Since it's a relatively new metric that does not have a great deal of research behind it in the public sphere, I don't want to rely on average_exit_velo.Specifically, I'm skeptical that it is not confounded by hard hit groundballs or weakly hit pop-ups, skewing it away from players who hit hard fly balls more often. Maximum exit velocity will instead, hopefully, give a more direct look at the power tool.

```
outcomes$wOBA <- round(with(outcomes, (`BB`*.69 + `HBP`*.72 + `1B`*.879 + `2B`*1.24 + `3B`*1.57
+ `HR`*2.02) / `PA`), 3)
outcomes$max_EV <- aggregate(exit_velo ~ batter_id, data = dataset, max)[,2]

outcomes

##                                   1B 2B 3B  BB HBP HR NULL  OUT OUT-SAC
## 26c2b53e65d15d855499513ae12fbb00 113 66  0 126  19 65 3722  784       9
## 35e5fc499d77f0cf1cab4e822f02a915 242 90  8 276  11 76 6226 1251      11
## 485ba029f9c20c2e7fa216b449ceefaa 216 69  1 133   6 37 3705  884       5
## 52f0a588b4443a0343fb29f63c97f53a 242 74  6 164  25 42 5051  986      13
## ede7784ac9ec41e851bc7539b7f2d94c 358 88  6 165   4 63 5581 1246      10
##                                  RBOE   PA   AB   H  TB   AVG   OBP   SLG
## 26c2b53e65d15d855499513ae12fbb00    4 1186 1028 244 505 0.237 0.332 0.491
## 35e5fc499d77f0cf1cab4e822f02a915   10 1975 1667 416 750 0.250 0.360 0.450
## 485ba029f9c20c2e7fa216b449ceefaa    6 1357 1207 323 505 0.268 0.343 0.418
## 52f0a588b4443a0343fb29f63c97f53a   12 1564 1350 364 576 0.270 0.359 0.427
## ede7784ac9ec41e851bc7539b7f2d94c   16 1956 1761 515 804 0.292 0.354 0.457
##                                   wOBA  max_EV
## 26c2b53e65d15d855499513ae12fbb00 0.348 114.893
## 35e5fc499d77f0cf1cab4e822f02a915 0.349 112.384
## 485ba029f9c20c2e7fa216b449ceefaa 0.330 111.530
## 52f0a588b4443a0343fb29f63c97f53a 0.339 109.368
## ede7784ac9ec41e851bc7539b7f2d94c 0.346 118.046
```

Since the range between these 5 players batting outcomes over the 3 years is relatively small (max wOBA = 0.349, min wOBA = 0.330) I would like to consider other variables to help determine a hierarchy of my preference, based on a set of values.

Principle #1: More recent performance is more useful in predicting future performance. Principle #2: Some stats (BB%, Contact%) are more consistent on a year-to-year basis than others (AVG, LD%) Principle #3: Larger samples are stronger samples, both because the information contained is more reliable and also because it is indicative of future playing time. Without supplementary injury / roster information, things like PA and G are the best indicators in this dataset of player health.

```
# Running the same code just for the 2015 season.

fifteen <- dataset[grep("2015", dataset$gamedate), ]
fifteen <- table(fifteen$batter_id, fifteen$outcome)
fifteen <- as.data.frame.matrix(fifteen)

fifteen$PA <- rowSums(fifteen[-c(7,10)])
```

```
fifteen$AB <- rowSums(fifteen[c(1,2,3,6,8)])
fifteen$H <- rowSums(fifteen[c(1,2,3,6)])
fifteen$TB <- with(fifteen, `1B`*1 + `2B`*2 + `3B`*3 + `HR` *4)
fifteen$AVG <- round(rowSums(fifteen[13] / fifteen[12]), 3)
fifteen$OBP <- round(rowSums(fifteen[c(4,5,13)]) / rowSums(fifteen[c(13,4,5,8)]),3)
fifteen$SLG <- round(rowSums(fifteen[14] / fifteen[12]), 3)
fifteen$wOBA <- round(with(fifteen, (`BB`*.69 + `HBP`*.72 + `1B`*.879 + `2B`*1.24 + `3B`*1.57 +
`HR`*2.02) / `PA`), 3)

fifteen$wOBA

## [1] 0.360 0.333 0.318 0.357 0.338

# Running the same code just for the 2016 season

sixteen <- dataset[grep("2016", dataset$gamedate), ]
sixteen <- table(sixteen$batter_id, sixteen$outcome)
sixteen <- as.data.frame.matrix(sixteen)

sixteen$PA <- rowSums(sixteen[-c(7,10)])
sixteen$AB <- rowSums(sixteen[c(1,2,3,6,8)])
sixteen$H <- rowSums(sixteen[c(1,2,3,6)])
sixteen$TB <- with(sixteen, `1B`*1 + `2B`*2 + `3B`*3 + `HR` *4)
sixteen$AVG <- round(rowSums(sixteen[13] / sixteen[12]), 3)
sixteen$OBP <- round(rowSums(sixteen[c(4,5,13)]) / rowSums(sixteen[c(13,4,5,8)]),3)
sixteen$SLG <- round(rowSums(sixteen[14] / sixteen[12]), 3)
sixteen$wOBA <- round(with(sixteen, (`BB`*.69 + `HBP`*.72 + `1B`*.879 + `2B`*1.24 + `3B`*1.57 +
`HR`*2.02) / `PA`), 3)

sixteen$wOBA

## [1] 0.298 0.368 0.303 0.340 0.327

# Running the same code just for the 2017 season

seventeen <- dataset[grep("2017", dataset$gamedate), ]
seventeen <- table(seventeen$batter_id, seventeen$outcome)
seventeen <- as.data.frame.matrix(seventeen)

seventeen$PA <- rowSums(seventeen[-c(7,10)])
seventeen$AB <- rowSums(seventeen[c(1,2,3,6,8)])
seventeen$H <- rowSums(seventeen[c(1,2,3,6)])
seventeen$TB <- with(seventeen, `1B`*1 + `2B`*2 + `3B`*3 + `HR` *4)
seventeen$AVG <- round(rowSums(seventeen[13] / seventeen[12]), 3)
seventeen$OBP <- round(rowSums(seventeen[c(4,5,13)]) / rowSums(seventeen[c(13,4,5,8)]),3)
seventeen$SLG <- round(rowSums(seventeen[14] / seventeen[12]), 3)
seventeen$wOBA <- round(with(seventeen, (`BB`*.69 + `HBP`*.72 + `1B`*.879 + `2B`*1.24 +
`3B`*1.57 + `HR`*2.02) / `PA`), 3)

seventeen$wOBA

## [1] 0.355 0.348 0.371 0.323 0.382

wOBA_df <- data.frame(row.names(outcomes), fifteen$PA, fifteen$wOBA, sixteen$PA, sixteen$wOBA,
seventeen$PA, seventeen$wOBA, outcomes$wOBA)
wOBA_df
```

```
##                row.names.outcomes. fifteen.PA fifteen.wOBA sixteen.PA
## 1 26c2b53e65d15d855499513ae12fbb00        562        0.360        159
## 2 35e5fc499d77f0cf1cab4e822f02a915        624        0.333        696
## 3 485ba029f9c20c2e7fa216b449ceefaa        371        0.318        490
## 4 52f0a588b4443a0343fb29f63c97f53a        574        0.357        543
## 5 ede7784ac9ec41e851bc7539b7f2d94c        669        0.338        633
##   sixteen.wOBA seventeen.PA seventeen.wOBA outcomes.wOBA
## 1        0.298          461          0.355         0.348
## 2        0.368          645          0.348         0.349
## 3        0.303          490          0.371         0.330
## 4        0.340          435          0.323         0.339
## 5        0.327          638          0.382         0.346
```

I additionally wanted to know whether some players were being used as platoon players, which could be shown by whether they were facing opposite handed pitching much more than one would expect a regular player to

```
table(dataset$batter_id, dataset$pitcher_hand, dataset$batter_hand)
```

```
## , ,  = LHB
##
##
##                                     LHP  RHP
##   26c2b53e65d15d855499513ae12fbb00 1134 3774
##   35e5fc499d77f0cf1cab4e822f02a915    0 5562
##   485ba029f9c20c2e7fa216b449ceefaa  827 4235
##   52f0a588b4443a0343fb29f63c97f53a    0    0
##   ede7784ac9ec41e851bc7539b7f2d94c 2500 5037
##
## , ,  = RHB
##
##
##                                     LHP  RHP
##   26c2b53e65d15d855499513ae12fbb00    0    0
##   35e5fc499d77f0cf1cab4e822f02a915 2632    7
##   485ba029f9c20c2e7fa216b449ceefaa    0    0
##   52f0a588b4443a0343fb29f63c97f53a 1980 4635
##   ede7784ac9ec41e851bc7539b7f2d94c    0    0
```

On a year-to-year basis we see a lot more variation between players' wOBA than the average across all 3. The 5th player produced a very strong 2017 wOBA in a large PA count, which would improve his 2018 forecast. The 4th player has declined in wOBA as well as PAs, both of which would hurt his production forecast for next season.

Next, I wanted to produce a MARCEL-style projection for the upcoming 2018 season. This would provide a simplistic outlook on future performance based on past performance, weighted for recency (3*2015, 4*2016, 5*2017), regressed to league rates of performance. League rate data acquired from http://www.fangraphs.com/leaders.aspx?pos=all&stats=bat&lg=all&qual=0&type=0&season=2017&month=0&season1=2015&ind=0&team=0,ss&rost=0&age=0&filter=&players=0

Methodology for this individualized MARCEL forecasting can be found here: http://www.tangotiger.net/archives/stud0346.shtml

```
league <- read.csv("C:/Users/jonathan.gruber/Downloads/FanGraphs Leaderboard (1).csv")

year <- c(2015, 2016, 2017)
weight <- c(3,4,5)
marcel <- data.frame(year, weight)
```

```r
# Establishing coefficients for different PA outcomes weighted for recency and league-wide rates of each event
marcel$HRco <- marcel$weight * league$HR / league$PA
marcel$`1Bco` <- marcel$weight * league$X1B / league$PA
marcel$`2Bco` <- marcel$weight * league$X2B / league$PA
marcel$`3Bco` <- marcel$weight * league$X3B / league$PA
marcel$BBco <- marcel$weight * league$BB / league$PA
marcel$HBPco <- marcel$weight * league$HBP / league$PA

eighteen <- data.frame(unique(dataset$batter_id))

eighteen$PA345 <- 3 * fifteen$PA + 4 * sixteen$PA + 5 * seventeen$PA
eighteen$HR345 <- 3 * fifteen$HR + 4 * sixteen$HR + 5 * seventeen$HR
eighteen$`1B345` <- 3 * fifteen$`1B` + 4 * sixteen$`1B` + 5 * seventeen$`1B`
eighteen$`2B345` <- 3 * fifteen$`2B` + 4 * sixteen$`2B` + 5 * seventeen$`2B`
eighteen$`3B345` <- 3 * fifteen$`3B` + 4 * sixteen$`3B` + 5 * seventeen$`3B`
eighteen$BB345 <- 3 * fifteen$BB + 4 * sixteen$BB + 5 * seventeen$BB
eighteen$HBP345 <- 3 * fifteen$HBP + 4 * sixteen$HBP + 5 * seventeen$HBP

eighteen$lg1B <- marcel$`1Bco`[1] * wOBA_df$fifteen.PA + marcel$`1Bco`[2] * wOBA_df$sixteen.PA + marcel$`1Bco`[3] * wOBA_df$seventeen.PA
eighteen$lg2B <- marcel$`2Bco`[1] * wOBA_df$fifteen.PA + marcel$`2Bco`[2] * wOBA_df$sixteen.PA + marcel$`2Bco`[3] * wOBA_df$seventeen.PA
eighteen$lg3B <- marcel$`3Bco`[1] * wOBA_df$fifteen.PA + marcel$`3Bco`[2] * wOBA_df$sixteen.PA + marcel$`3Bco`[3] * wOBA_df$seventeen.PA
eighteen$lgHR <- marcel$HRco[1] * wOBA_df$fifteen.PA + marcel$HRco[2] * wOBA_df$sixteen.PA + marcel$HRco[3] * wOBA_df$seventeen.PA
eighteen$lgHBP <- marcel$HBPco[1] * wOBA_df$fifteen.PA + marcel$HBPco[2] * wOBA_df$sixteen.PA + marcel$HBPco[3] * wOBA_df$seventeen.PA
eighteen$lgBB <- marcel$BBco[1] * wOBA_df$fifteen.PA + marcel$BBco[2] * wOBA_df$sixteen.PA + marcel$BBco[3] * wOBA_df$seventeen.PA

eighteen$projPA <- 0.1 * wOBA_df$sixteen.PA + 0.5 * wOBA_df$seventeen.PA + 200


eighteen$`1Brate` <- (eighteen$`1B345` + eighteen$lg1B * 1200/eighteen$PA345) / (eighteen$PA345 + 1200)
eighteen$`2Brate` <- (eighteen$`2B345` + eighteen$lg2B * 1200/eighteen$PA345) / (eighteen$PA345 + 1200)
eighteen$`3Brate` <- (eighteen$`3B345` + eighteen$lg3B * 1200/eighteen$PA345) / (eighteen$PA345 + 1200)
eighteen$BBrate <- (eighteen$BB345 + eighteen$lgBB * 1200/eighteen$PA345) / (eighteen$PA345 + 1200)
eighteen$HBPrate <- (eighteen$HBP345 + eighteen$lgHBP * 1200/eighteen$PA345) / (eighteen$PA345 + 1200)
eighteen$HRrate <- (eighteen$HR345 + eighteen$lgHR * 1200/eighteen$PA345) / (eighteen$PA345 + 1200)

eighteen$proj1B <- eighteen$`1Brate` * eighteen$projPA
eighteen$proj2B <- eighteen$`2Brate` * eighteen$projPA
eighteen$proj3B <- eighteen$`3Brate` * eighteen$projPA
eighteen$projBB <- eighteen$BBrate * eighteen$projPA
eighteen$projHBP <- eighteen$HBPrate * eighteen$projPA
eighteen$projHR <- eighteen$HRrate * eighteen$projPA
```

```
eighteen
```

```
##          unique.dataset.batter_id. PA345 HR345 1B345 2B345 3B345 BB345
## 1 ede7784ac9ec41e851bc7539b7f2d94c  4627   262   426   257     0   495
## 2 485ba029f9c20c2e7fa216b449ceefaa  7881   306   978   370    33  1087
## 3 35e5fc499d77f0cf1cab4e822f02a915  5523   170   862   282     3   555
## 4 52f0a588b4443a0343fb29f63c97f53a  6069   157   935   285    22   673
## 5 26c2b53e65d15d855499513ae12fbb00  7729   261  1438   349    21   667
##   HBP345     lg1B     lg2B     lg3B     lgHR    lgHBP     lgBB projPA
## 1     69  686.9707 208.5750 21.51917 140.3470 42.32892 378.1089  446.4
## 2     47 1169.4676 354.6814 36.57694 240.9167 71.91829 646.5401  592.1
## 3     23  818.1456 248.6412 25.47331 170.0473 50.55238 454.8095  494.0
## 4     89  902.7405 272.9951 28.41033 183.7109 55.14520 495.3202  471.8
## 5     13 1147.3831 347.8844 35.92522 235.7134 70.50897 633.2972  582.3
##      1Brate     2Brate       3Brate     BBrate     HBPrate     HRrate
## 1 0.1036835 0.05338825 0.0009577722 0.10177819 0.013725397 0.05120965
## 2 0.1273063 0.04669150 0.0042472619 0.13054128 0.006381526 0.03773628
## 3 0.1546573 0.04998112 0.0012694732 0.09725092 0.005054838 0.03078190
## 4 0.1531841 0.04663341 0.0037993487 0.10605830 0.013743795 0.02659574
## 5 0.1809992 0.04513521 0.0029765628 0.08571232 0.002681956 0.03332923
##      proj1B   proj2B    proj3B   projBB  projHBP   projHR
## 1  46.28433 23.83252 0.4275495 45.43379 6.127017 22.85999
## 2  75.37809 27.64604 2.5148038 77.29349 3.778501 22.34365
## 3  76.40071 24.69067 0.6271198 48.04195 2.497090 15.20626
## 4  72.27227 22.00164 1.7925327 50.03830 6.484323 12.54787
## 5 105.39585 26.28224 1.7332525 49.91029 1.561703 19.40761
```

I wanted to know how the different projected wOBA's would compare for the 5 players. Because the run-values-environment is not yet established, I will use the same linear weights that I relied on in the previous wOBA calculator.

```
eighteen$wOBA <- round(with(eighteen, (`projBB`*.69 + `projHBP`*.72 + `proj1B`*.879 +
`proj2B`*1.24 + `proj3B`*1.57 + `projHR`*2.02) / `projPA`), 3)
```

I wanted to try different PA projection formula, since the one above by MARCEL is a bit less elegant and does not as closely resemble the other coefficients used. This new PA projection (projPA3yr) would account for all three seasons, weighted for recency with a similar coefficient to the 3/4/5 one, without the need for a base of 200 PA.

```
eighteen$projPA3yr <- (.75 * wOBA_df$fifteen.PA +  wOBA_df$sixteen.PA + 1.25 *
wOBA_df$seventeen.PA) / 3

eighteen$proj1B <- eighteen$`1Brate` * eighteen$projPA3yr
eighteen$proj2B <- eighteen$`2Brate` * eighteen$projPA3yr
eighteen$proj3B <- eighteen$`3Brate` * eighteen$projPA3yr
eighteen$projBB <- eighteen$BBrate * eighteen$projPA3yr
eighteen$projHBP <- eighteen$HBPrate * eighteen$projPA3yr
eighteen$projHR <- eighteen$HRrate * eighteen$projPA3yr

eighteen$wOBAPA3 <- round(with(eighteen, (`projBB`*.69 + `projHBP`*.72 + `proj1B`*.879 +
`proj2B`*1.24 + `proj3B`*1.57 + `projHR`*2.02) / `projPA3yr`), 3)

wOBA_df$eighteen <- eighteen$wOBA
wOBA_df$projPA <- eighteen$projPA
wOBA_df$eighteenPA3 <- eighteen$wOBAPA3
wOBA_df$projPA3 <- eighteen$projPA3yr
```

```
wOBA_df

##                 row.names.outcomes. fifteen.PA fifteen.wOBA sixteen.PA
## 1 26c2b53e65d15d855499513ae12fbb00        562        0.360        159
## 2 35e5fc499d77f0cf1cab4e822f02a915        624        0.333        696
## 3 485ba029f9c20c2e7fa216b449ceefaa        371        0.318        490
## 4 52f0a588b4443a0343fb29f63c97f53a        574        0.357        543
## 5 ede7784ac9ec41e851bc7539b7f2d94c        669        0.338        633
##   sixteen.wOBA seventeen.PA seventeen.wOBA outcomes.wOBA eighteen projPA
## 1        0.298          461          0.355         0.348    0.342  446.4
## 2        0.368          645          0.348         0.349    0.347  592.1
## 3        0.303          490          0.371         0.330    0.333  494.0
## 4        0.340          435          0.323         0.339    0.335  471.8
## 5        0.327          638          0.382         0.346    0.348  582.3
##   eighteenPA3  projPA3
## 1       0.342 385.5833
## 2       0.347 656.7500
## 3       0.333 460.2500
## 4       0.335 505.7500
## 5       0.348 644.0833
```

Since wOBA is calculated on a rate basis, the number of PAs projected would not change it. However, from a player ranking standpoint, a player who has sustained performance over a longer period is preferrable, as it is more predictive of future durability. As with the wOBA of the combined three seasons, the spread in projected 2018 wOBA is very small here (min = .333, max = .348). There are many additional factors that could be considered, such as strength of opponent, how well splits (home/away, RHP/LHP, weather, etc.) favoured players differently, age, and many others that I have not included here. Nonetheless, here is my final ranking, with a short explanation for each player:

1: 35e5fc499d77f0cf1cab4e822f02a91 [second line]

This player produced the highest wOBA from the combined previous three seasons, despite a weaker 2017, and is a close 2nd for the 2018 wOBA projection. I really like this player's durability and walk rate. Both of which are known to be important skills in projecting future performance. He has the most walks by a large margin, enough to give him the highest OBP despite the 4th highest AVG, and the most PAs by a smaller one.

2: ede7784ac9ec41e851bc7539b7f2d94c [fifth line]

This player has also provided a lot of durability, and projects to have the top wOBA next season, after posting the strongest 2017 of the group. He has seems to rely on his BABIP a bit more than the others, which would make his performance appear to be a little more erratic. I also like his max_EV, which is quite high and leads the group. I don't know that it's predictive of future power, but it at least indicates that he has raw power that may eventually translate into more game power.

3: 26c2b53e65d15d855499513ae12fbb00 [first line]

This player produced the largest SLG and had a strong wOBA and max_EV. It appears that his production dropped off after a strong 2015, including a lot of missed time in 2016. The MARCEL PA projection is more forgiving of this, but even then he's still projected for the fewest PAs of the group. With the lowest AVG and OBP I am a bit concerned about the batting profile, particularly given the nature of how slugging has seemingly changed across the league since the middle part of the 2015 season.

4: 485ba029f9c20c2e7fa216b449ceefaa [third line]

This player seems to be used as a LHB platoon hitter, but has provided a consistent number of PAs despite facing RHP a disproportionate amount of the time (4235 vs 827). Because of this, I am inclined to think this player's durability is stronger than the projected PAs would believe, since he is likely being benched while healthy when facing LHP. Nevertheless, he posted the weakest wOBA of the group over the past 3 seasons in the second fewest PAs, even given the disproportionate PAs with a platoon advantage.

5: 52f0a588b4443a0343fb29f63c97f53a [fourth line]

This player has produced declining PAs and wOBAs in each of the last 3 seasons. I don't see the same upside with this player that I do with certain aspects of the other players' performance. Though I think his strong OBP would protect him from having a terrible 2018, I see little upside in either his durability (indicated by PAs) or ability to hit for power. He also produced the lowest max_EV of the group, which hurts the potential for more game power to appear, although the margin is small.