# assignment2

2024-08-15

# R Markdown

```
library(baseballr)
library(dplyr)
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
library(ggplot2)
library(caTools)
library(knitr)
library(tibble)
library(RColorBrewer)
library(flux)
## This is flux 0.3-0.1
library(gridExtra)
##
## Attaching package: 'gridExtra'
## The following object is masked from 'package:dplyr':
##
##     combine
library(rockchalk)
##
## Attaching package: 'rockchalk'
## The following object is masked from 'package:dplyr':
##
##     summarize
```

```
library(tidyverse)
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✓ forcats   1.0.0     ✓ readr     2.1.5
## ✓ lubridate 1.9.3     ✓ stringr   1.5.1
## ✓ purrr     1.0.2     ✓ tidyr     1.3.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ tidyr::chop()          masks flux::chop()
## ✖ gridExtra::combine()   masks dplyr::combine()
## ✖ dplyr::filter()        masks stats::filter()
## ✖ dplyr::lag()           masks stats::lag()
## ✖ rockchalk::summarize() masks dplyr::summarize()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflict
s to become errors
```

Question #1 Determine the exponent in the Pythagorean Wins Formula that results in the lowest prediction error for games in recent history (you can decide how many years to include, last 5 years or 10 years, but include all major league games in that time frame). Are we performing better or worse than expected?

```
## ── MLB League data from MLB.com ──────────────────────────── baseballr 1.6.0 ──
## ℹ Data updated: 2024-08-16 00:26:47 CDT
## # A tibble: 116 × 40
##    league_id league_name       league_link league_abbreviation league_name_short
##  *     <int> <chr>             <chr>       <chr>               <chr>
##  1       103 American League   /api/v1/le… AL                  American
##  2       104 National League   /api/v1/le… NL                  National
##  3       114 Cactus League     /api/v1/le… CL                  Cactus
##  4       115 Grapefruit League /api/v1/le… GL                  Grapefruit
##  5       117 International Le… /api/v1/le… INT                 International
##  6       112 Pacific Coast Le… /api/v1/le… PCL                 Pacific Coast
##  7       138 American Associa… /api/v1/le… AA (1)              American Assoc.
##  8       139 American Associa… /api/v1/le… AA (2)              American Assoc.
##  9       113 Eastern League    /api/v1/le… EAS                 Eastern
## 10       111 Southern League   /api/v1/le… SOU                 Southern
## # ℹ 106 more rows
## # ℹ 35 more variables: league_season_state <chr>, league_has_wild_card <lgl>,
## #   league_has_split_season <lgl>, league_num_games <int>,
```

```
## #   league_has_playoff_points <lgl>, league_num_teams <int>,
## #   league_num_wildcard_teams <int>, league_season <chr>,
## #   league_org_code <chr>, league_conferences_in_use <lgl>,
## #   league_divisions_in_use <lgl>, league_sort_order <int>, …
# lets do some exponents


CleanStandings$exp1point2 <- ((CleanStandings$rsra)^1.2)/(((CleanStandings$rsra)^1.2)
+ 1)
CleanStandings$exp1point3 <- ((CleanStandings$rsra)^1.3)/(((CleanStandings$rsra)^1.3)
+ 1)
CleanStandings$exp1point4 <- ((CleanStandings$rsra)^1.4)/(((CleanStandings$rsra)^1.4)
+ 1)
CleanStandings$exp1point5 <- ((CleanStandings$rsra)^1.5)/(((CleanStandings$rsra)^1.5)
+ 1)
CleanStandings$exp1point6 <- ((CleanStandings$rsra)^1.6)/(((CleanStandings$rsra)^1.6)
+ 1)
CleanStandings$exp1point7 <- ((CleanStandings$rsra)^1.7)/(((CleanStandings$rsra)^1.7)
+ 1)
CleanStandings$exp1point8 <- ((CleanStandings$rsra)^1.8)/(((CleanStandings$rsra)^1.8)
+ 1)
CleanStandings$exp1point9 <- ((CleanStandings$rsra)^1.9)/(((CleanStandings$rsra)^1.9)
+ 1)
CleanStandings$exp2point0 <- ((CleanStandings$rsra)^2)/(((CleanStandings$rsra)^2) + 1)
CleanStandings$exp2point1 <- ((CleanStandings$rsra)^2.1)/(((CleanStandings$rsra)^2.1)
+ 1)
CleanStandings$exp2point2 <- ((CleanStandings$rsra)^2.2)/(((CleanStandings$rsra)^2.2)
+ 1)
CleanStandings$exp2point3 <- ((CleanStandings$rsra)^2.3)/(((CleanStandings$rsra)^2.3)
+ 1)
CleanStandings$exp2point4 <- ((CleanStandings$rsra)^2.4)/(((CleanStandings$rsra)^2.4)
+ 1)
```

Then we want to get absolute error, but there is a problem as the winning percentage is characterized as a character and not a number, so we have to fix that

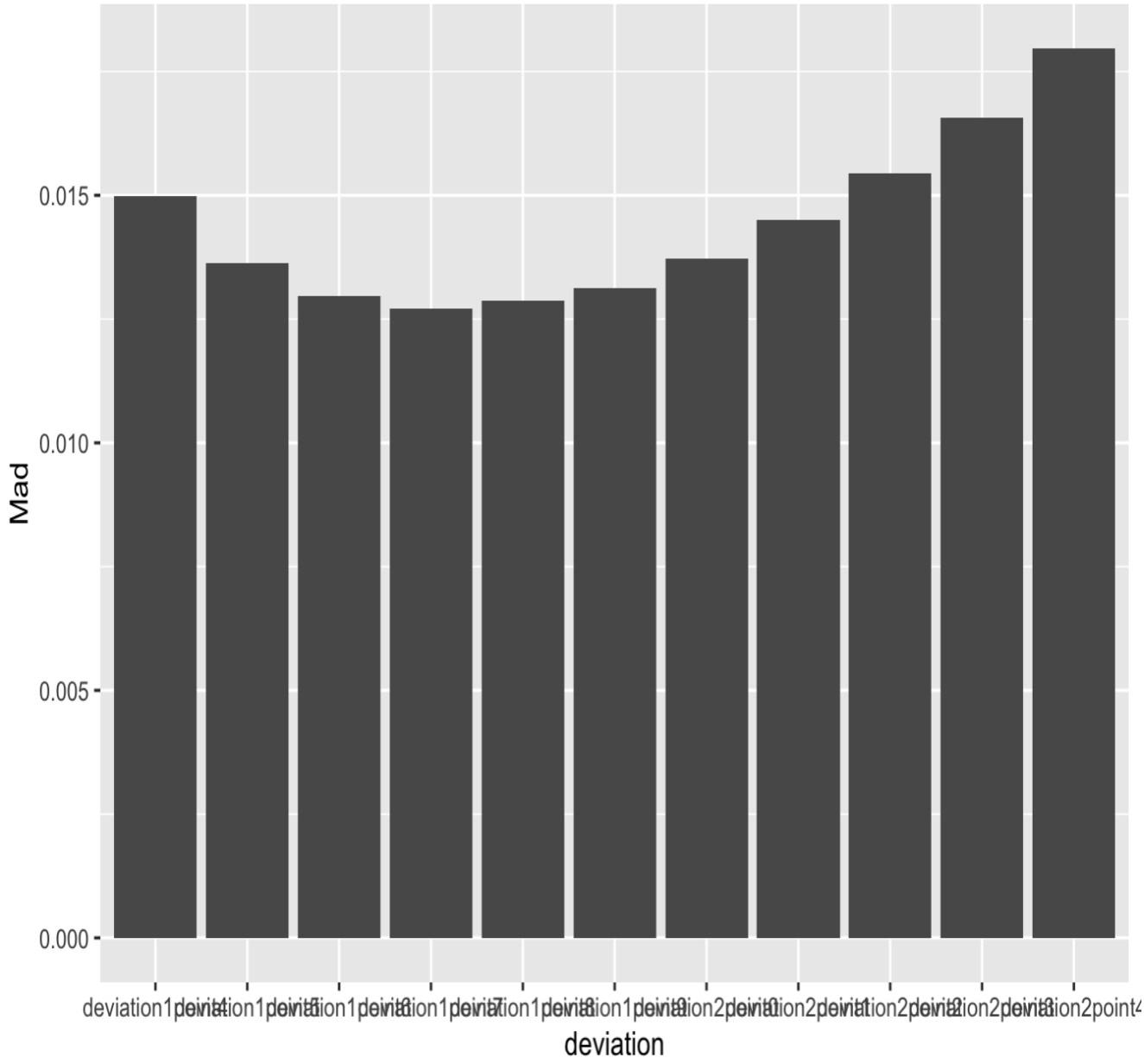
```
CleanStandings$team_records_league_record_pct <- as.numeric(CleanStandings$team_record
s_league_record_pct)


CleanStandings$absoluteerror1point2 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point2))
CleanStandings$absoluteerror1point3 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point3))
CleanStandings$absoluteerror1point4 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point4))
```

```
CleanStandings$absoluteerror1point5 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point5))

CleanStandings$absoluteerror1point6 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point6))

CleanStandings$absoluteerror1point7 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point7))

CleanStandings$absoluteerror1point8 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point8))

CleanStandings$absoluteerror1point9 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp1point9))

CleanStandings$absoluteerror2point0 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp2point0))

CleanStandings$absoluteerror2point1 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp2point1))

CleanStandings$absoluteerror2point2 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp2point2))

CleanStandings$absoluteerror2point3 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp2point3))

CleanStandings$absoluteerror2point4 <- abs((CleanStandings$team_records_league_record_
pct) - (CleanStandings$exp2point4))




View(CleanStandings)
```

Now for absolute deviations

```
CleanStandings$deviation1point2 <- abs((CleanStandings$absoluteerror1point2) - (mean(C
leanStandings$absoluteerror1point2)))

CleanStandings$deviation1point3 <- abs((CleanStandings$absoluteerror1point3) - (mean(C
leanStandings$absoluteerror1point3)))

CleanStandings$deviation1point4 <- abs((CleanStandings$absoluteerror1point4) - (mean(C
leanStandings$absoluteerror1point4)))

CleanStandings$deviation1point5 <- abs((CleanStandings$absoluteerror1point5) - (mean(C
leanStandings$absoluteerror1point5)))

CleanStandings$deviation1point6 <- abs((CleanStandings$absoluteerror1point6) - (mean(C
leanStandings$absoluteerror1point6)))

CleanStandings$deviation1point7 <- abs((CleanStandings$absoluteerror1point7) - (mean(C
leanStandings$absoluteerror1point7)))

CleanStandings$deviation1point8 <- abs((CleanStandings$absoluteerror1point8) - (mean(C
leanStandings$absoluteerror1point8)))

CleanStandings$deviation1point9 <- abs((CleanStandings$absoluteerror1point9) - (mean(C
leanStandings$absoluteerror1point9)))

CleanStandings$deviation2point0 <- abs((CleanStandings$absoluteerror2point0) - (mean(C
leanStandings$absoluteerror2point0)))
```

```
CleanStandings$deviation2point1 <- abs((CleanStandings$absoluteerror2point1) - (mean(C
leanStandings$absoluteerror2point1)))

CleanStandings$deviation2point2 <- abs((CleanStandings$absoluteerror2point2) - (mean(C
leanStandings$absoluteerror2point2)))

CleanStandings$deviation2point3 <- abs((CleanStandings$absoluteerror2point3) - (mean(C
leanStandings$absoluteerror2point3)))

CleanStandings$deviation2point4 <- abs((CleanStandings$absoluteerror2point4) - (mean(C
leanStandings$absoluteerror2point4)))

Mad<- colMeans(CleanStandings[,37:47])

datamad <- data.frame(Mad)

datamad <- rownames_to_column(datamad, var = "deviation")


ggplot(datamad, aes(x = deviation, y = Mad)) +

  geom_bar(stat = 'identity', position = 'dodge')
```




from this we can see the lowest MAD is for the exponent 1.7, so we should use that for our team Pythagorean Wins Formula

```
CleanRecords <- select(CleanStandings, 1, 2, 5, 14)
```

We can compare the pythagorean win loss record with th 1.7 exponent to the actual win loss record and see that every year except for 2019 and 2023, the Mariners overperformed expectations

Question 2: Pick the three players with the most at-bats this year and determine where they rank at their position compared to the rest of the league. Pick 4 metrics to discuss with at least one coming from seasonal data, one coming from play-by-play data, and one coming from pitch-by-pitch data

```
batstat <- (fg_batter_leaders(startseason = 2023, endseason = 2023, pos = "np"))


marinersbatters <- batstat %>%
filter(team_name == "SEA")
(marinersbatters[order(marinersbatters$AB, decreasing = TRUE), ]   )
## ── MLB Player Batting Leaders data from FanGraphs.com ──────── baseballr 1.6.0 ──
## ℹ Data updated: 2024-08-16 00:26:55 CDT
## # A tibble: 17 × 349
##    Season team_name Bats  xMLBAMID PlayerNameRoute   PlayerName   playerid   Age
##     <int> <chr>     <chr>    <int> <chr>             <chr>           <int> <int>
##  1   2023 SEA       R       677594 Julio Rodriguez   Julio Rodrí…    23697    22
##  2   2023 SEA       R       606192 Teoscar Hernandez Teoscar Her…    13066    30
##  3   2023 SEA       R       553993 Eugenio Suarez    Eugenio Suá…    12552    31
##  4   2023 SEA       R       664034 Ty France         Ty France       17982    28
##  5   2023 SEA       L       641487 J.P. Crawford     J.P. Crawfo…    15491    28
##  6   2023 SEA       B       663728 Cal Raleigh       Cal Raleigh     21534    26
##  7   2023 SEA       L       672284 Jarred Kelenic    Jarred Kele…    22558    23
##  8   2023 SEA       R       676609 Jose Caballero    José Caball…    23401    26
##  9   2023 SEA       L       645801 Mike Ford         Mike Ford       15585    30
## 10   2023 SEA       R       608596 Tom Murphy        Tom Murphy      13499    32
## 11   2023 SEA       R       664238 Dylan Moore       Dylan Moore     18042    30
## 12   2023 SEA       B       664059 Sam Haggerty      Sam Haggerty    18054    29
## 13   2023 SEA       L       687799 Cade Marlowe      Cade Marlowe    25505    26
## 14   2023 SEA       L       666211 Taylor Trammell   Taylor Tram…    19960    25
## 15   2023 SEA       B       669450 Cooper Hummel     Cooper Humm…    19458    28
## 16   2023 SEA       L       600303 Tommy La Stella   Tommy La St…    12371    34
## 17   2023 SEA       R       657247 Brian O'Keefe     Brian O'Kee…    16680    29
## # ℹ 341 more variables: AgeRng <chr>, SeasonMin <int>, SeasonMax <int>,
## #   G <int>, AB <int>, PA <int>, H <int>, `1B` <int>, `2B` <int>, `3B` <int>,
## #   HR <int>, R <int>, RBI <int>, BB <int>, IBB <int>, SO <int>, HBP <int>,
## #   SF <int>, SH <int>, GDP <int>, SB <int>, CS <int>, AVG <dbl>, GB <int>,
## #   FB <int>, LD <int>, IFFB <int>, Pitches <int>, Balls <int>, Strikes <int>,
## #   IFH <int>, BU <int>, BUH <int>, BB_pct <dbl>, K_pct <dbl>, BB_K <dbl>,
## #   OBP <dbl>, SLG <dbl>, OPS <dbl>, ISO <dbl>, BABIP <dbl>, GB_FB <dbl>, …
```

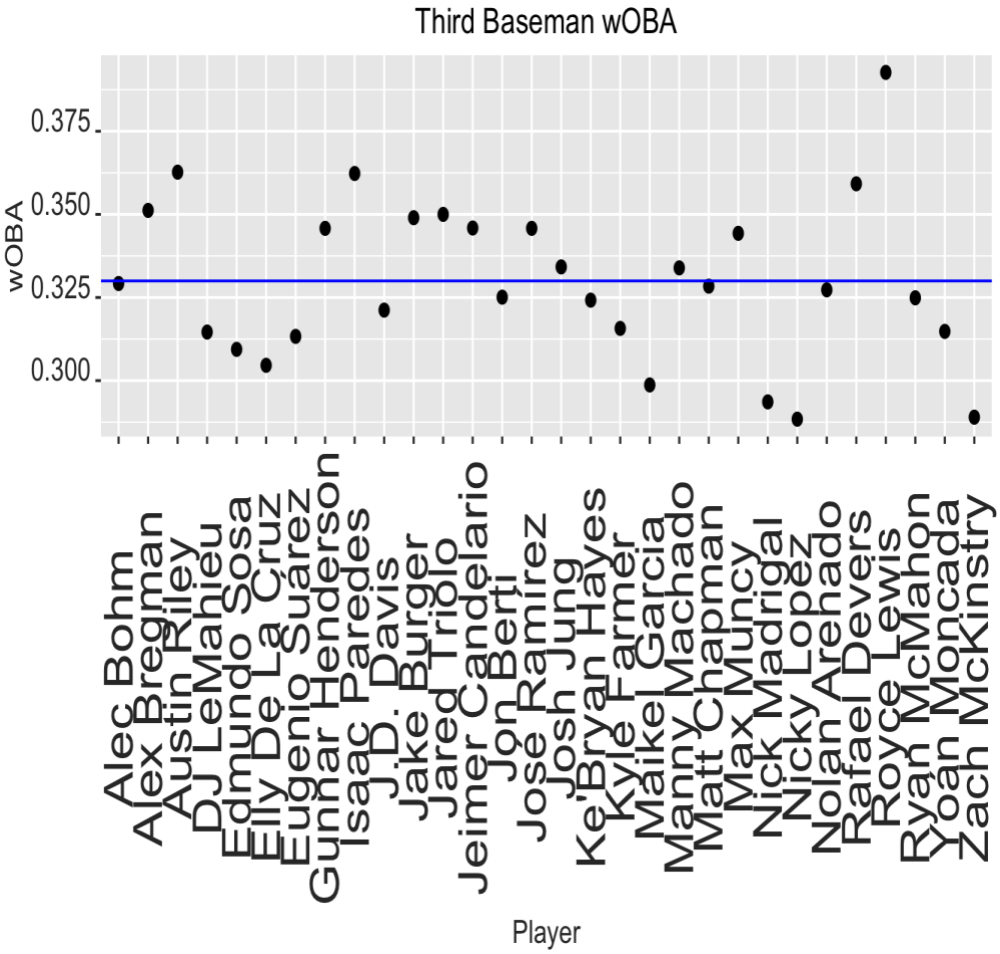
From this, we can see that Julio Rodriguez (Center Field), Teoscar Hernandez (Right Field), and Eugenio Suarez are the top three batters in at bats for the Seattle Mariners.

We want to compare with four metrics. We Will use wRC+, wOBA, WPA, and Contact percentage

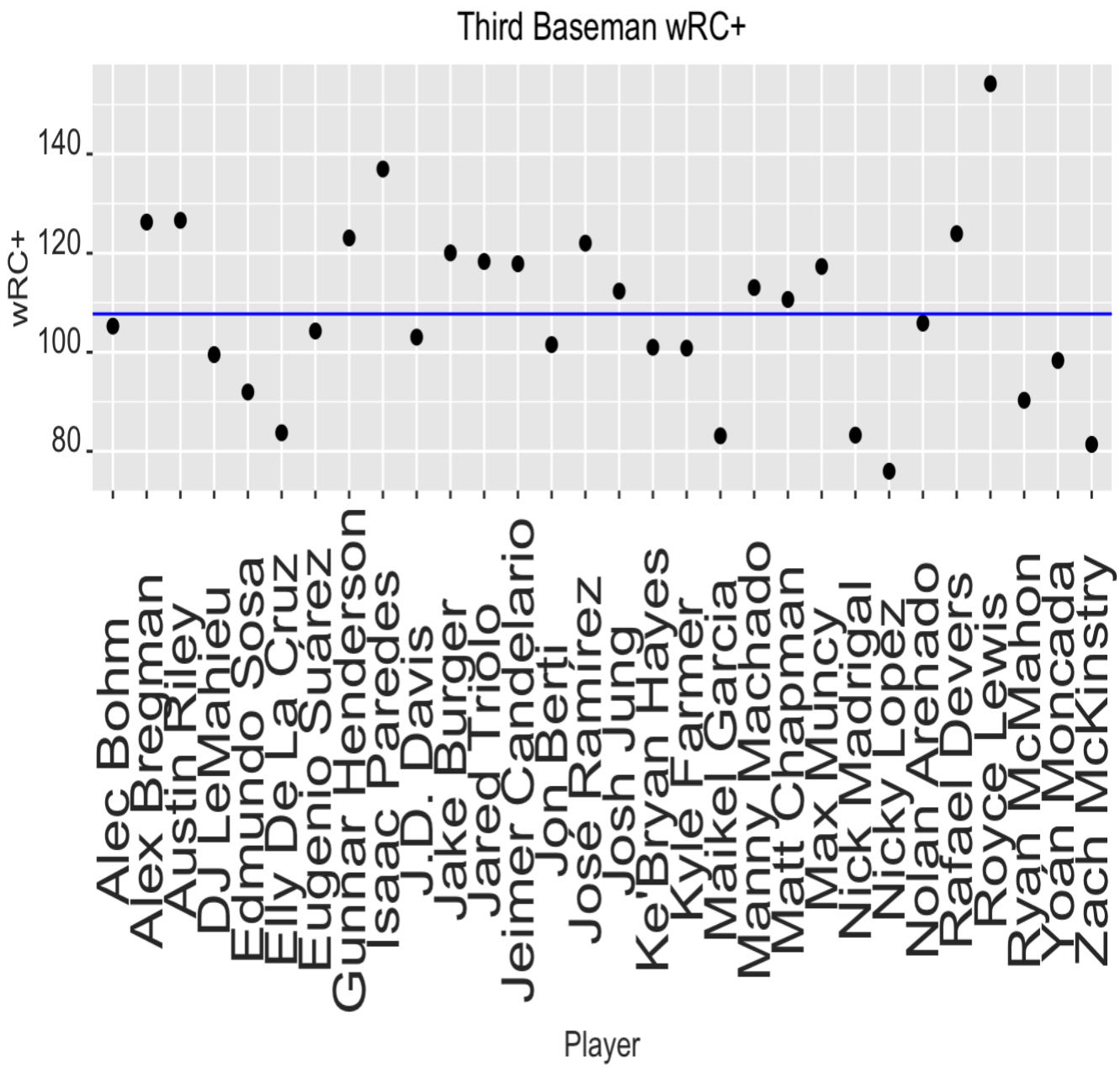
```
thirdbase <- fg_batter_leaders(startseason = "2023", endseason = "2023", pos = "3b")
thirdbaseclean <- (thirdbase[1:30, c(6, 7, 8, 12, 59,75, 77, 116)])
centerfield <- fg_batter_leaders(startseason = "2023", endseason = "2023", pos = "cf")
centerfieldclean <- (centerfield[1:30, c(6, 7, 8, 12, 59,75, 77, 116)])
rightfield <- fg_batter_leaders(startseason = "2023", endseason = "2023", pos = "rf")
rightfieldclean <- (rightfield[1:30, c(6, 7, 8, 12, 59,75, 77, 116)])
```

Graphs for third base

```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = wOBA)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(thirdbaseclean$wOBA), color="blue")+
  ggtitle(bquote('Third Baseman wOBA')) +
  theme(plot.title = element_text(hjust = 0.5))+
  scale_color_brewer(palette="YlOrRd") +
  ylab("wOBA") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),
        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),
        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```

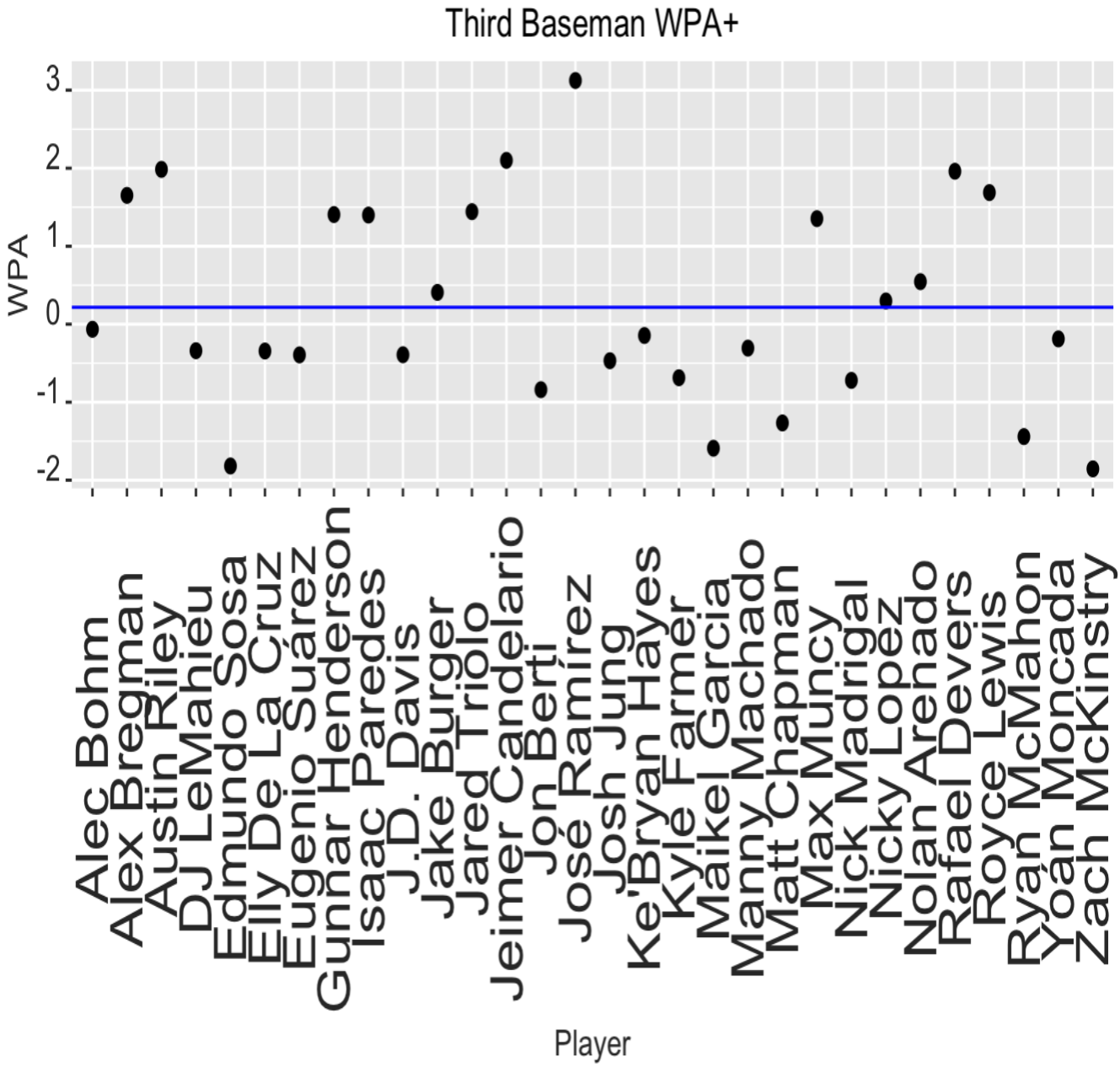



```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = wRC_plus)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(thirdbaseclean$wRC_plus), color="blue")+
  ggtitle(bquote('Third Baseman wRC+')) +
  theme(plot.title = element_text(hjust = 0.5))+
  ylab("wRC+") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),
        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),
        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



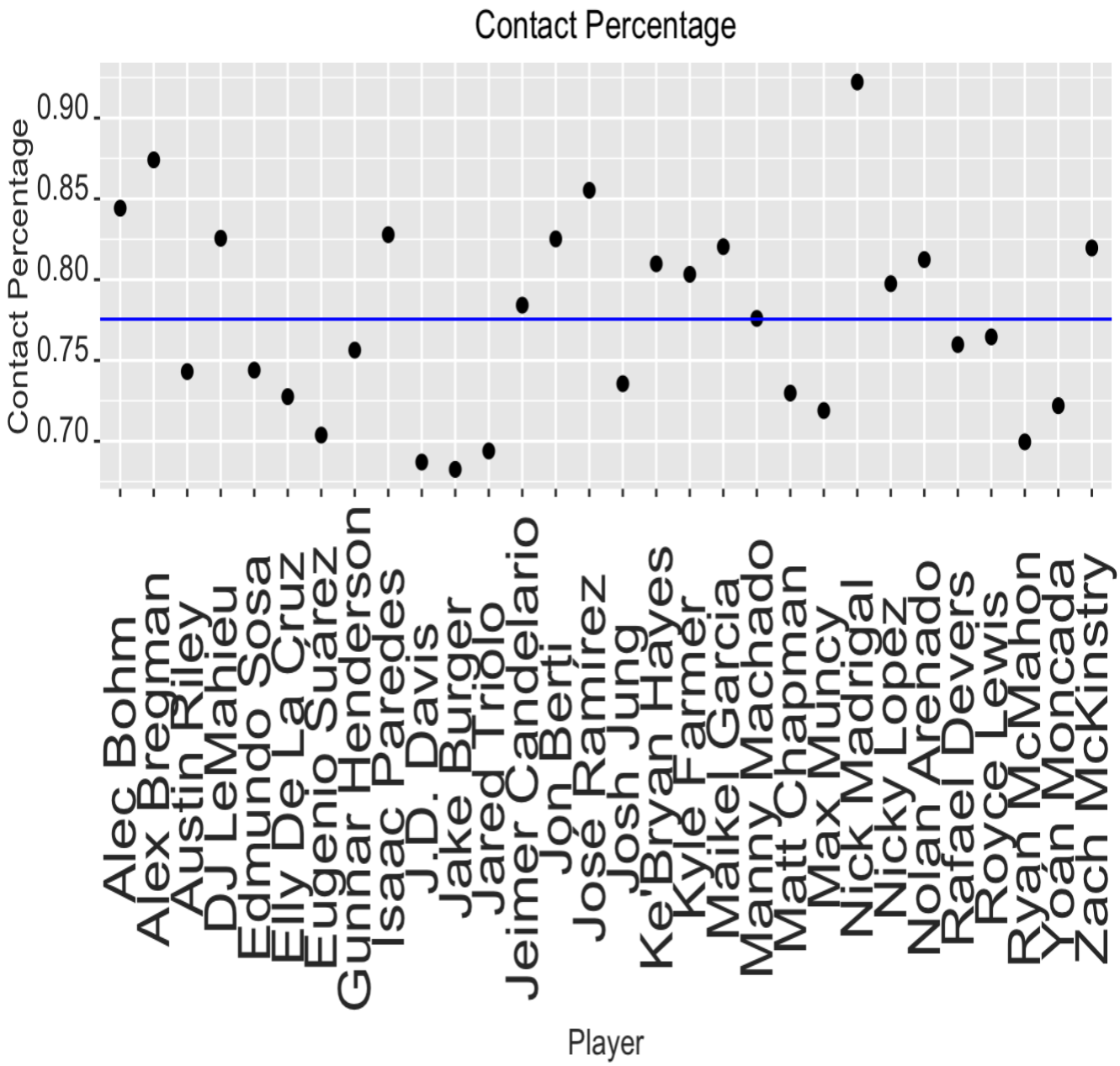

```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = WPA)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(thirdbaseclean$WPA), color="blue")+
  ggtitle(bquote('Third Baseman WPA+')) +
  theme(plot.title = element_text(hjust = 0.5))+
  ylab("WPA") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
```

```
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



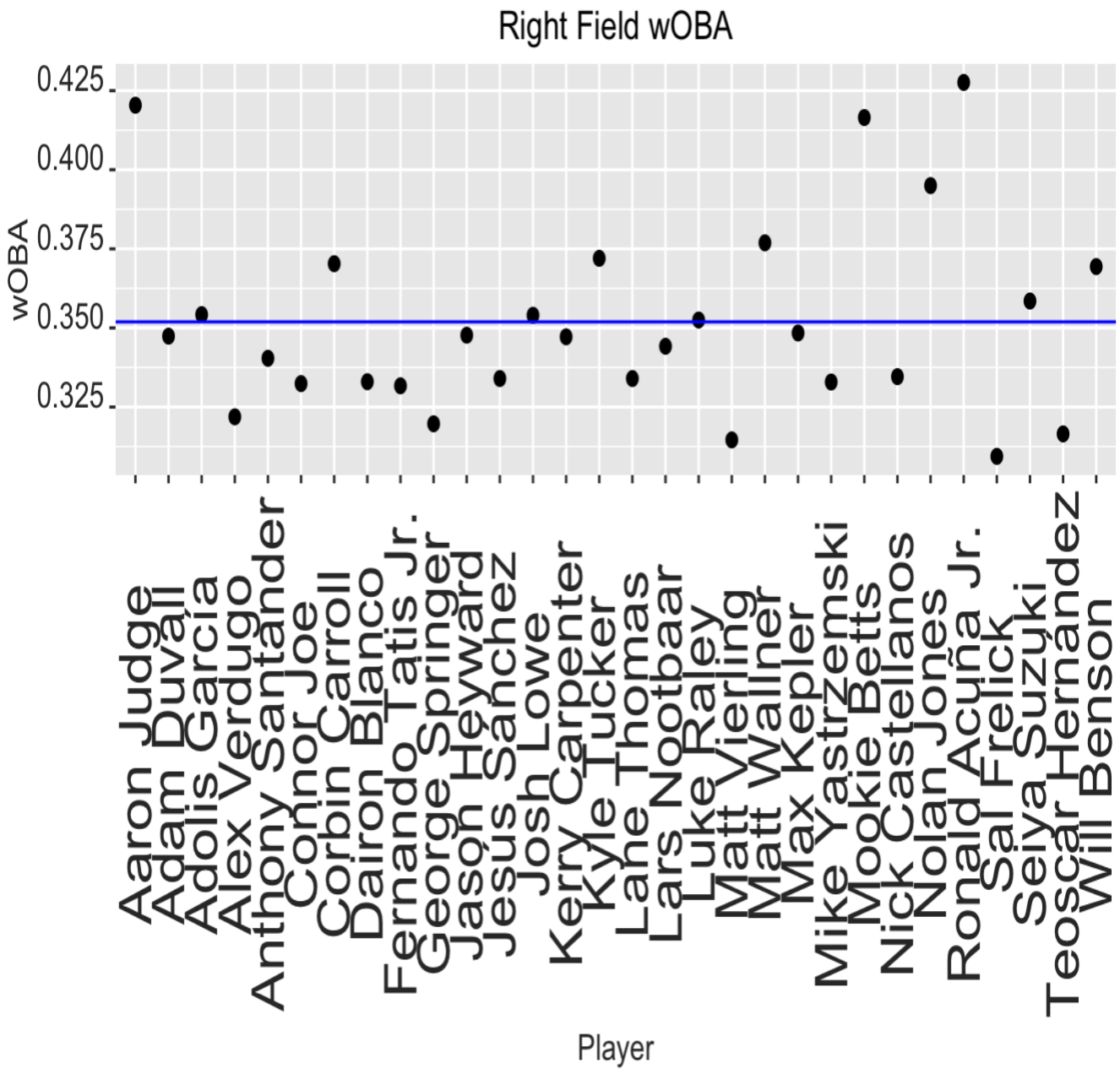

```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = Contact_pct)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(thirdbaseclean$Contact_pct), color="blue")+
```

```
  ggtitle(bquote('Contact Percentage')) +

  theme(plot.title = element_text(hjust = 0.5))+

  ylab("Contact Percentage") +

  xlab(bquote('Player')) +

  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



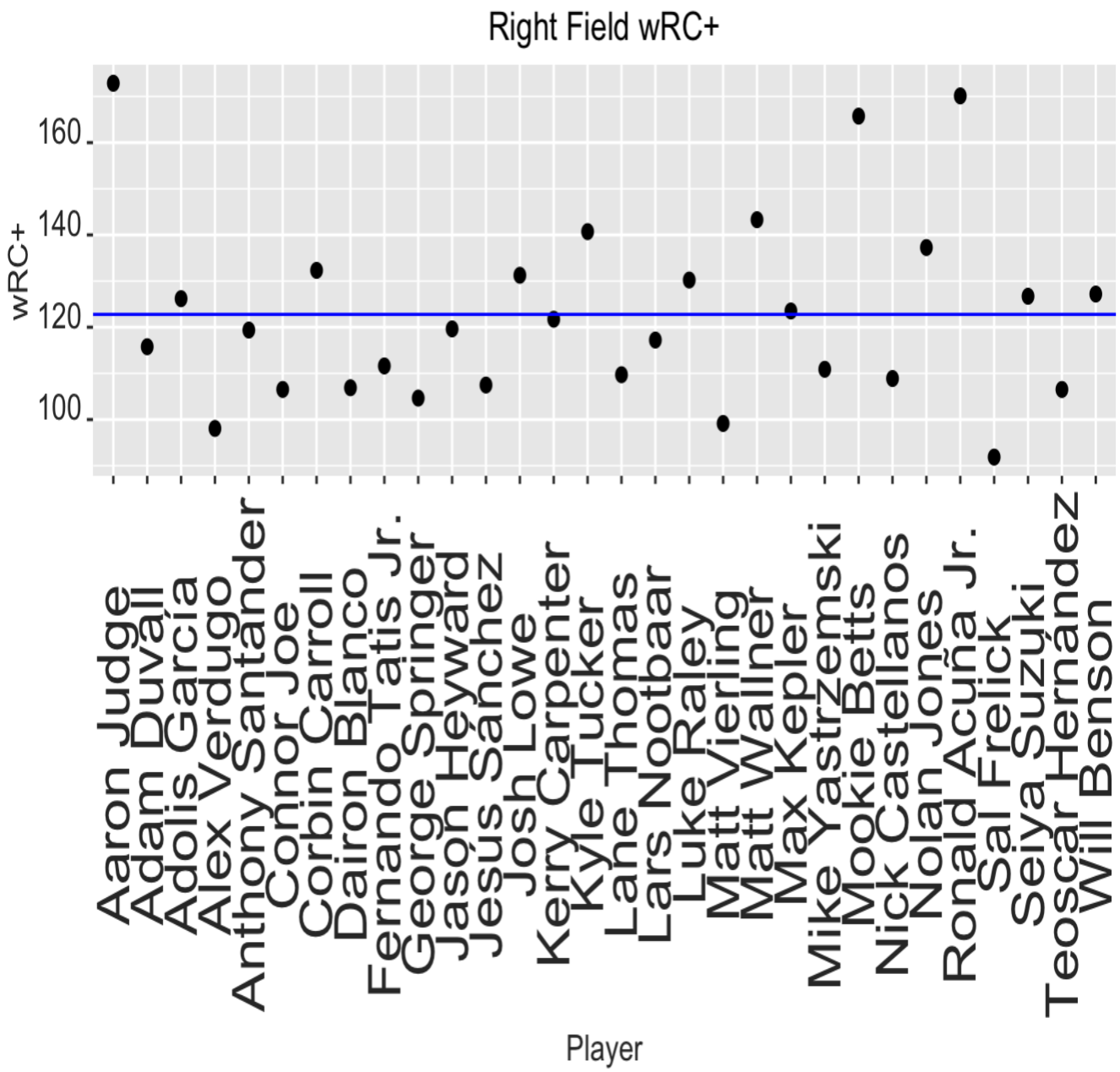

```
ggplot(data = rightfieldclean, aes(x = PlayerName, y = wOBA)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(rightfieldclean$wOBA), color="blue")+
  ggtitle(bquote('Right Field wOBA')) +
  theme(plot.title = element_text(hjust = 0.5))+
  scale_color_brewer(palette="YlOrRd") +
  ylab("wOBA") +
  xlab(bquote('Player')) +
```

```
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



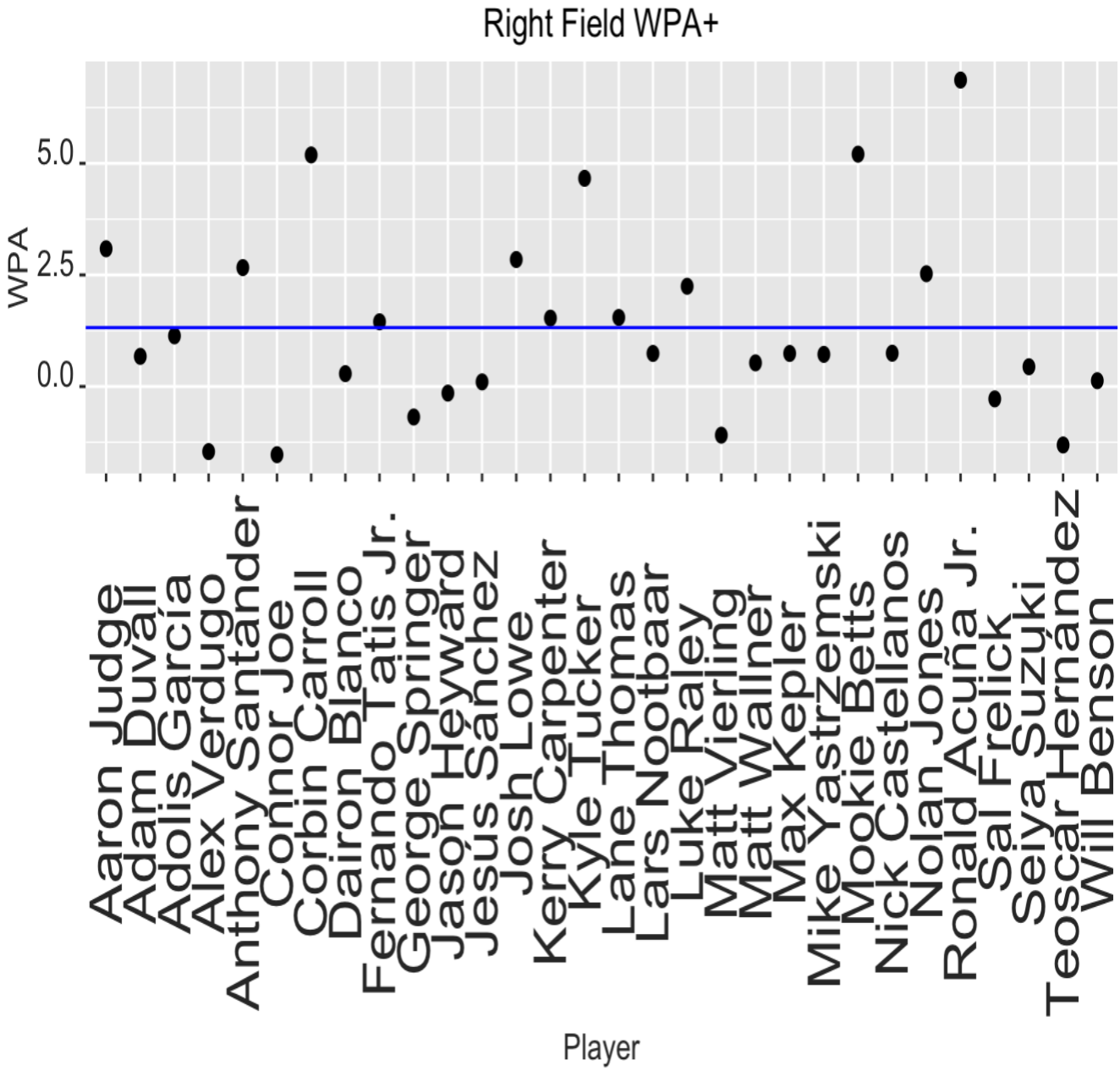

```
ggplot(data = rightfieldclean, aes(x = PlayerName, y = wRC_plus)) +

  geom_point(size = 2) +
```

```
  geom_hline(yintercept = mean(rightfieldclean$wRC_plus), color="blue")+
  ggtitle(bquote('Right Field wRC+')) +
  theme(plot.title = element_text(hjust = 0.5))+
  ylab("wRC+") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),
        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),
        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



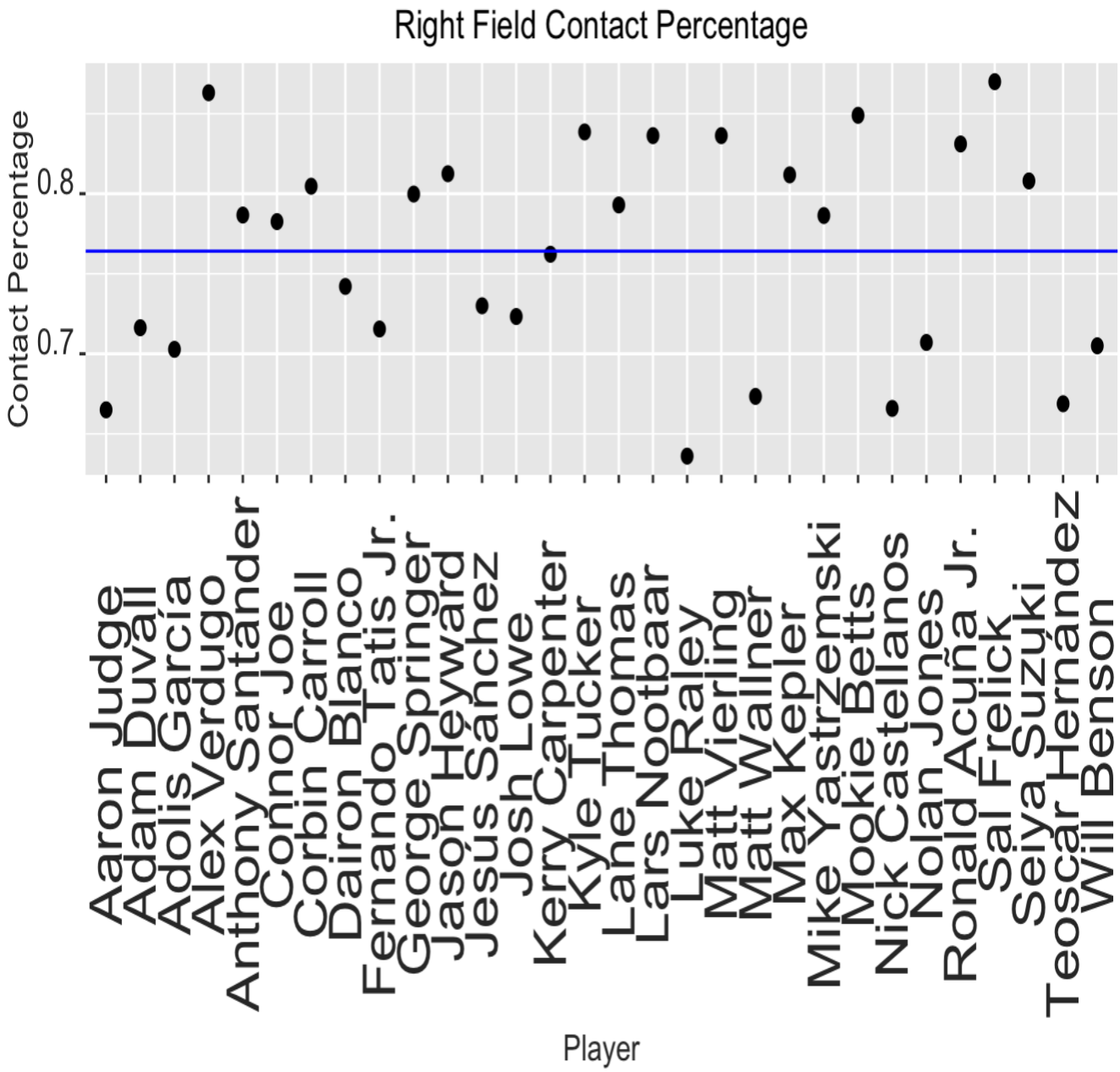

```
ggplot(data = rightfieldclean, aes(x = PlayerName, y = WPA)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(rightfieldclean$WPA), color="blue")+
  ggtitle(bquote('Right Field WPA+')) +
  theme(plot.title = element_text(hjust = 0.5))+
  ylab("WPA") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
```

```
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



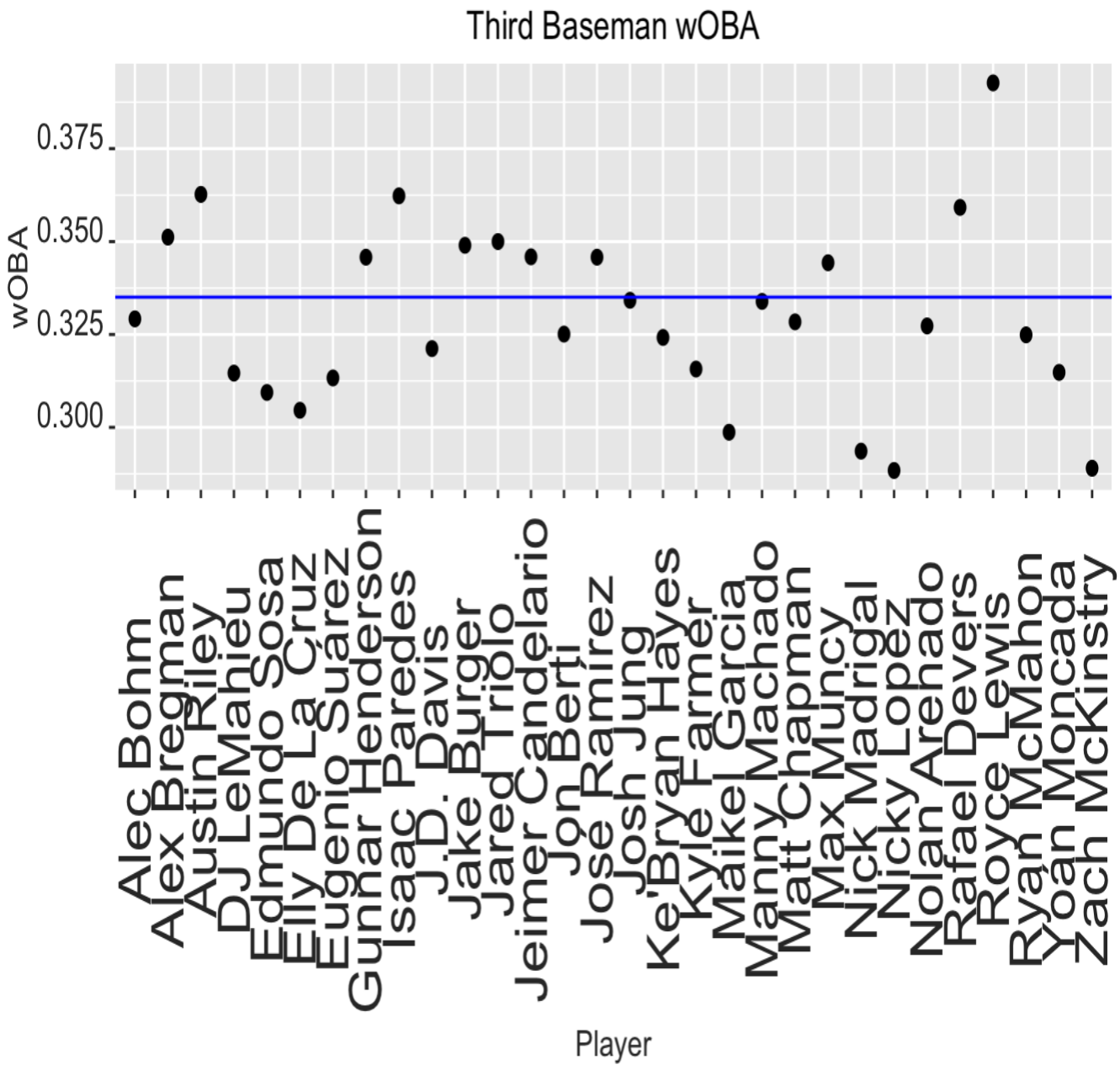

```
ggplot(data = rightfieldclean, aes(x = PlayerName, y = Contact_pct)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(rightfieldclean$Contact_pct), color="blue")+
```

```
  ggtitle(bquote('Right Field Contact Percentage')) +

  theme(plot.title = element_text(hjust = 0.5))+

  ylab("Contact Percentage") +

  xlab(bquote('Player')) +

  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



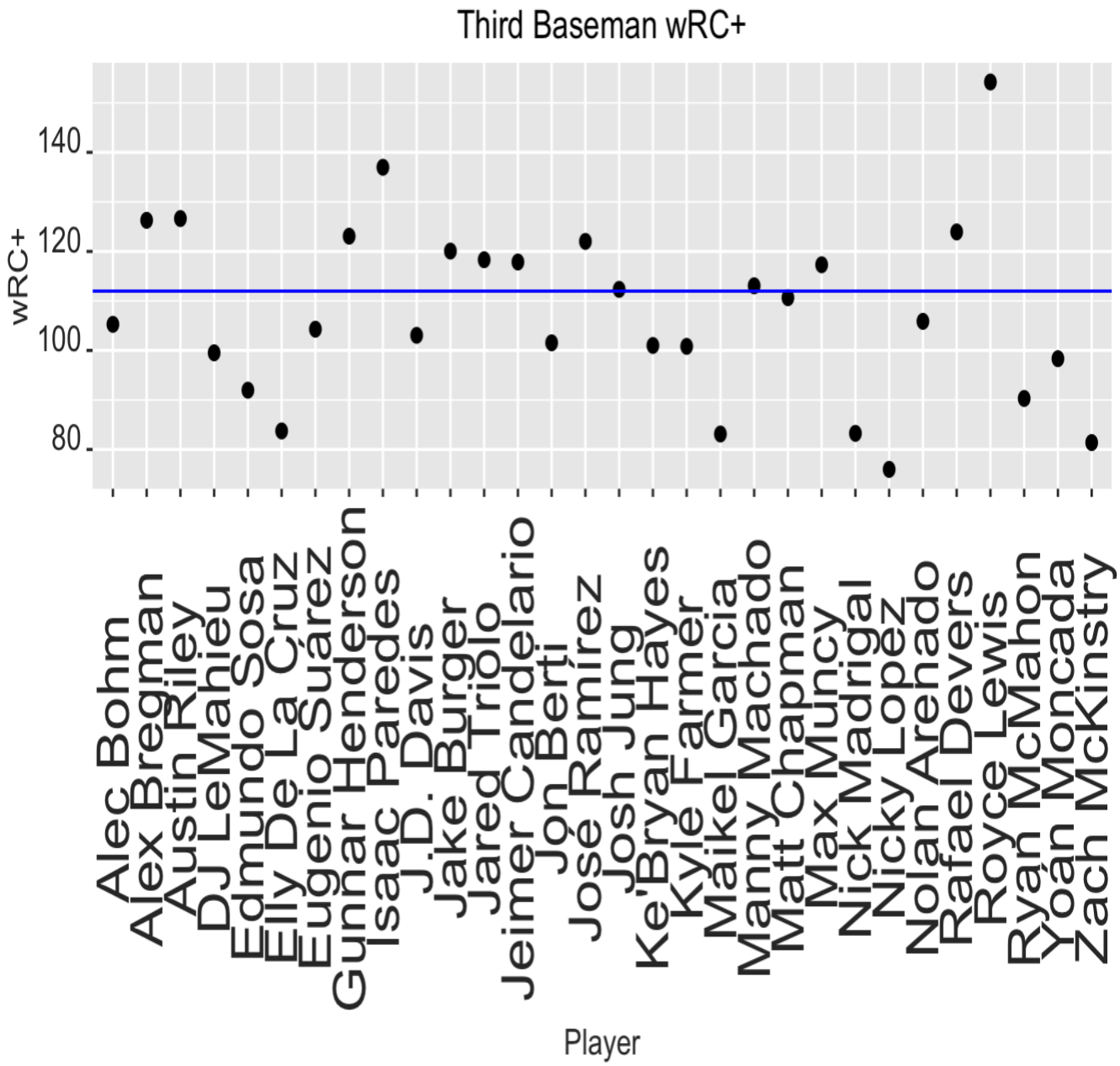

```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = wOBA)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(centerfieldclean$wOBA), color="blue")+
  ggtitle(bquote('Third Baseman wOBA')) +
  theme(plot.title = element_text(hjust = 0.5))+
  scale_color_brewer(palette="YlOrRd") +
  ylab("wOBA") +
  xlab(bquote('Player')) +
```

```
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



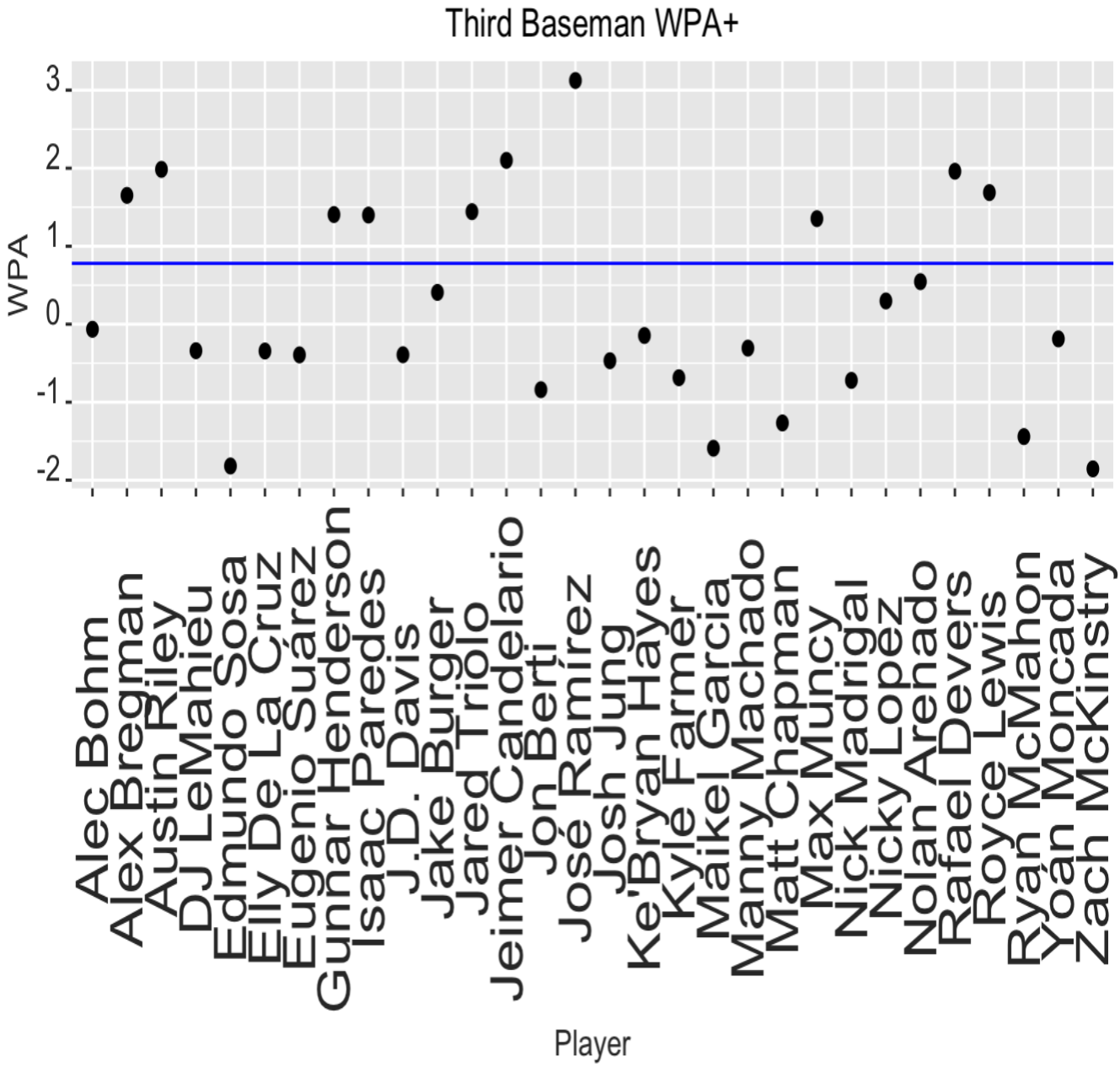

```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = wRC_plus)) +

  geom_point(size = 2) +
```

```
  geom_hline(yintercept = mean(centerfieldclean$wRC_plus), color="blue")+

  ggtitle(bquote('Third Baseman wRC+')) +

  theme(plot.title = element_text(hjust = 0.5))+

  ylab("wRC+") +

  xlab(bquote('Player')) +

  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



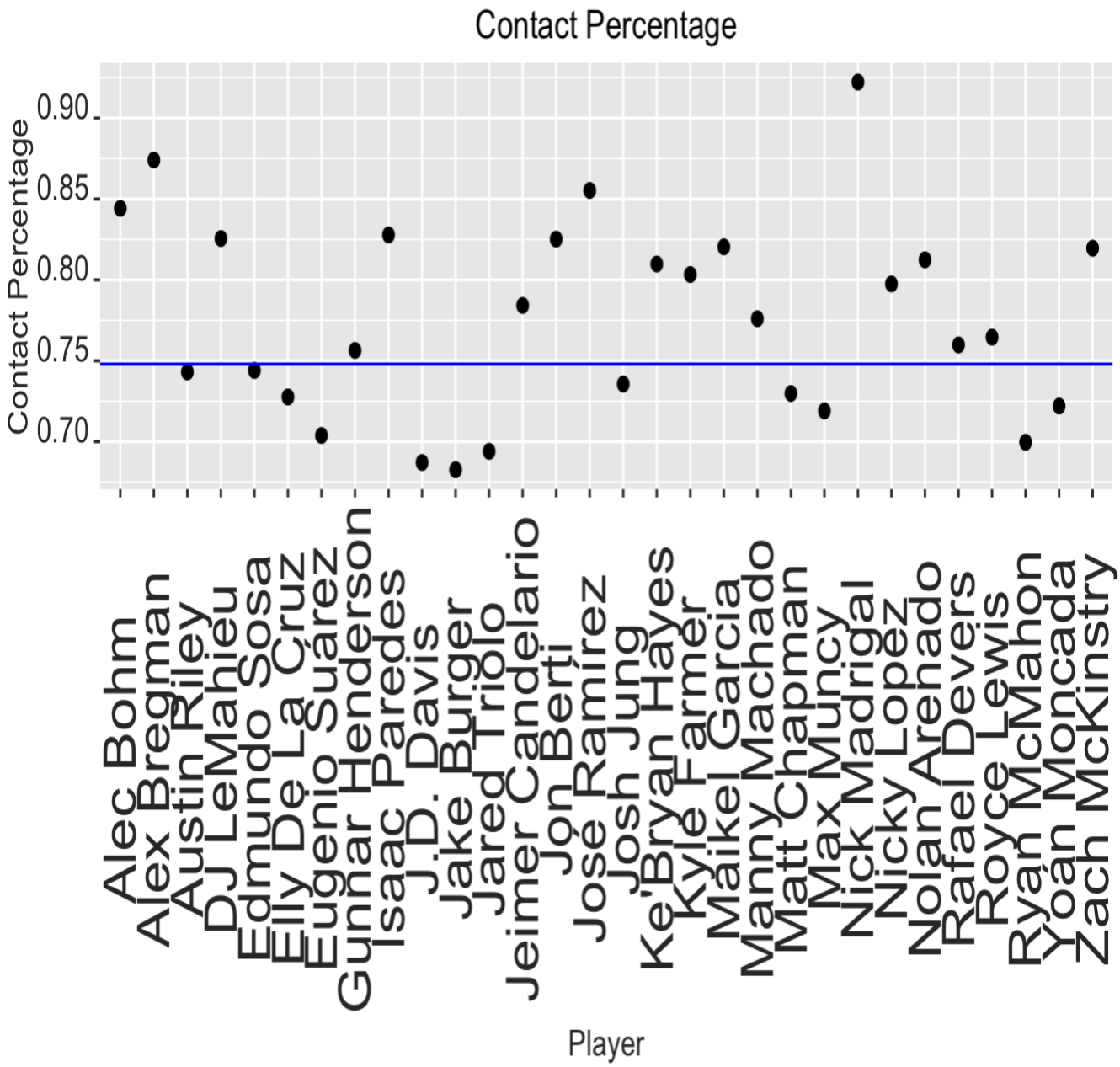

```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = WPA)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(centerfieldclean$WPA), color="blue")+
  ggtitle(bquote('Third Baseman WPA+')) +
  theme(plot.title = element_text(hjust = 0.5))+
  ylab("WPA") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
```

```
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



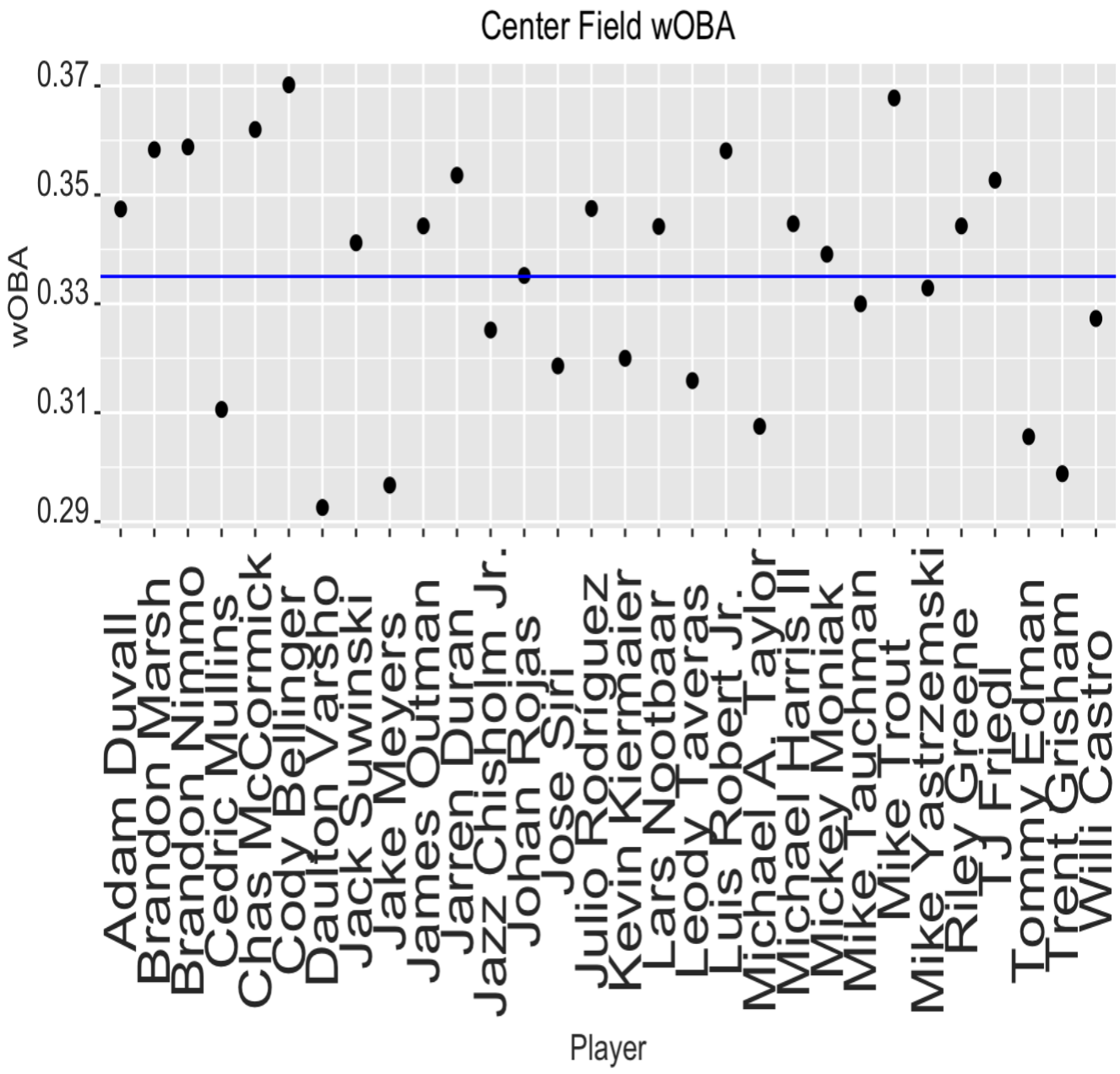

```
ggplot(data = thirdbaseclean, aes(x = PlayerName, y = Contact_pct)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(centerfieldclean$Contact_pct), color="blue")+
```

```
  ggtitle(bquote('Contact Percentage')) +

  theme(plot.title = element_text(hjust = 0.5))+

  ylab("Contact Percentage") +

  xlab(bquote('Player')) +

  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



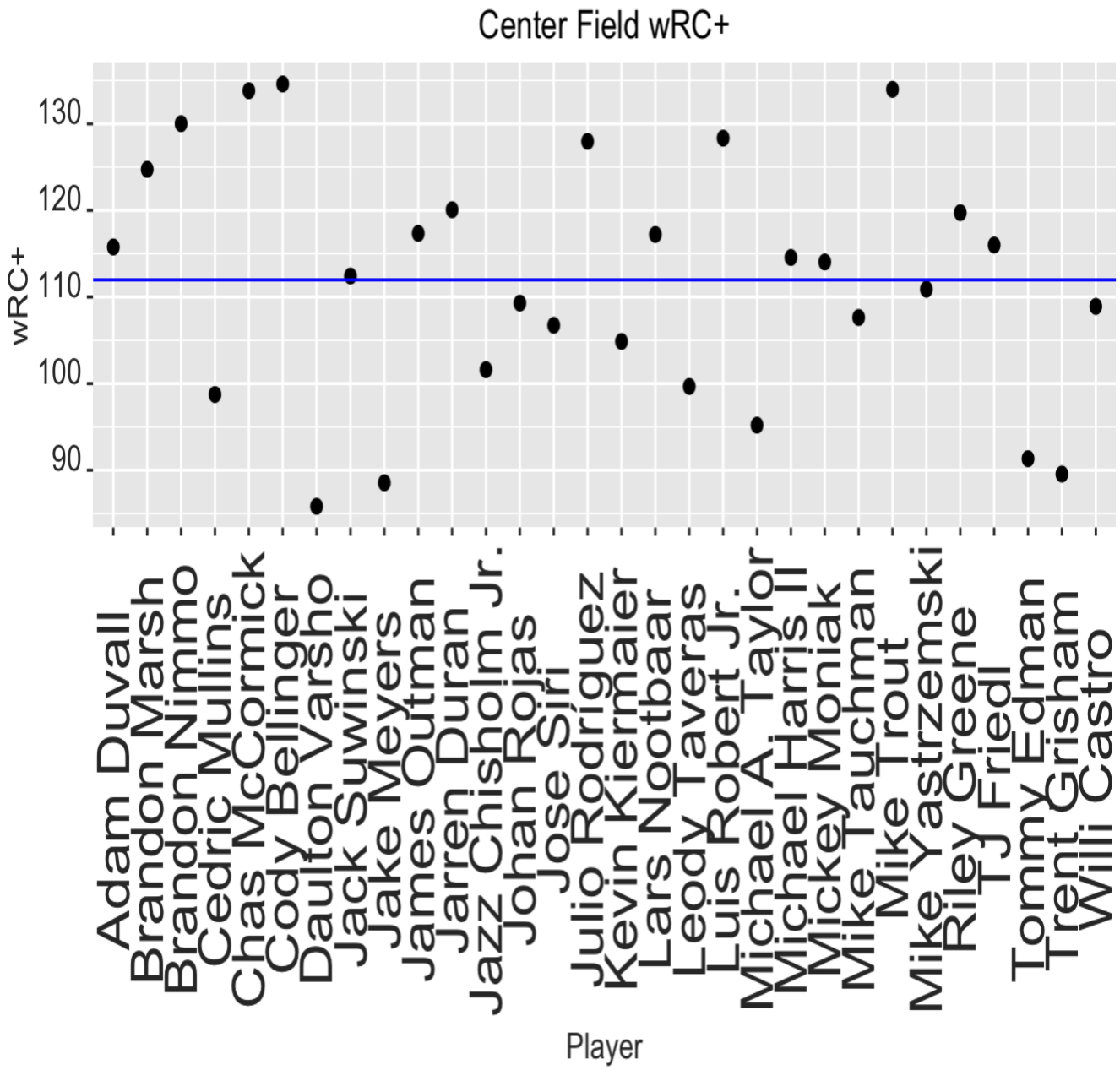

```
ggplot(data = centerfieldclean, aes(x = PlayerName, y = wOBA)) +
  geom_point(size = 2) +
geom_hline(yintercept = mean(centerfieldclean$wOBA), color="blue")+
  ggtitle(bquote('Center Field wOBA')) +
  theme(plot.title = element_text(hjust = 0.5))+
  scale_color_brewer(palette="YlOrRd") +
  ylab("wOBA") +
  xlab(bquote('Player')) +
```

```
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



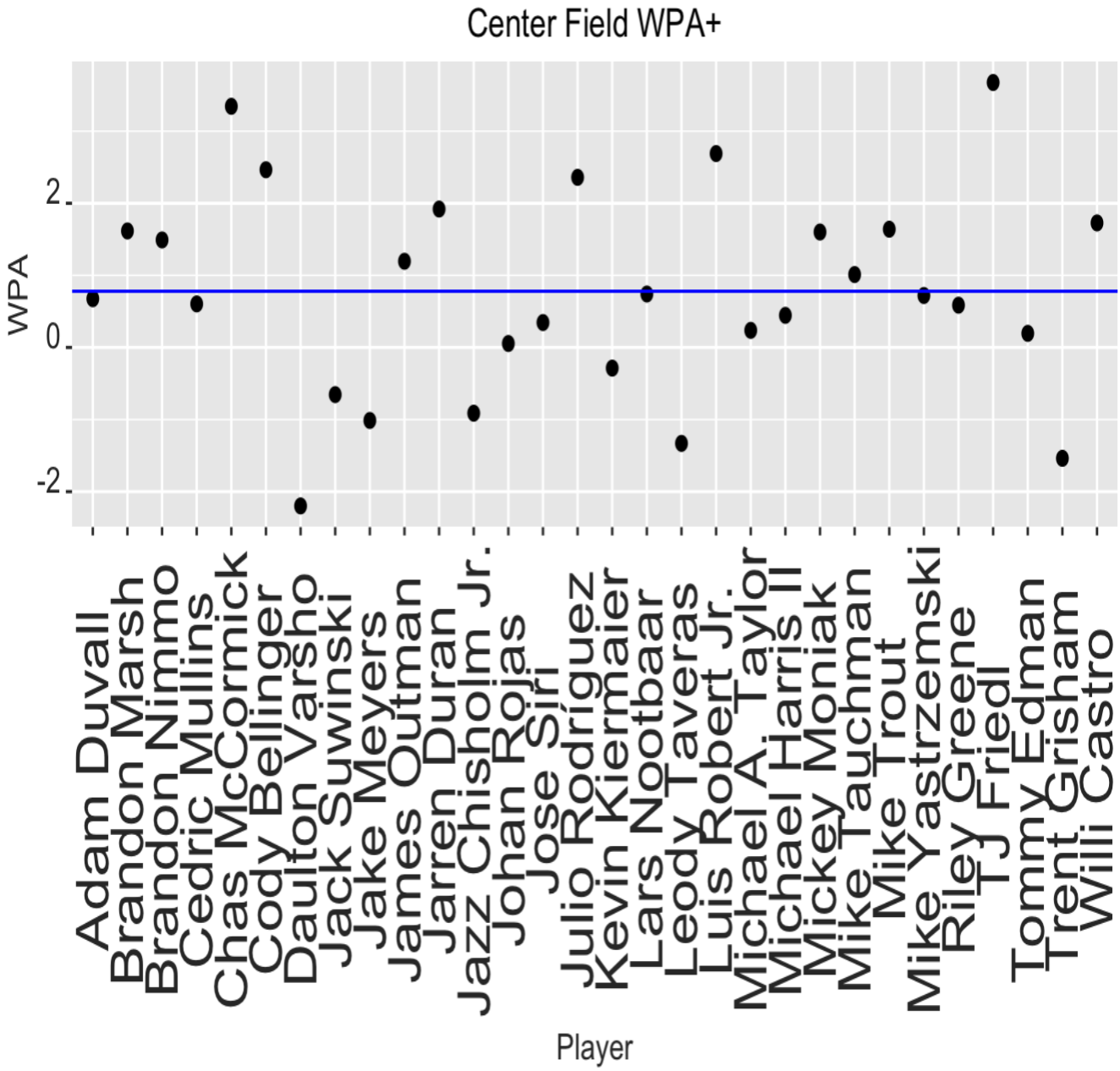

```
ggplot(data = centerfieldclean, aes(x = PlayerName, y = wRC_plus)) +

  geom_point(size = 2) +
```

```
geom_hline(yintercept = mean(centerfieldclean$wRC_plus), color="blue")+
  ggtitle(bquote('Center Field wRC+')) +
  theme(plot.title = element_text(hjust = 0.5))+
  ylab("wRC+") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),
        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),
        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```



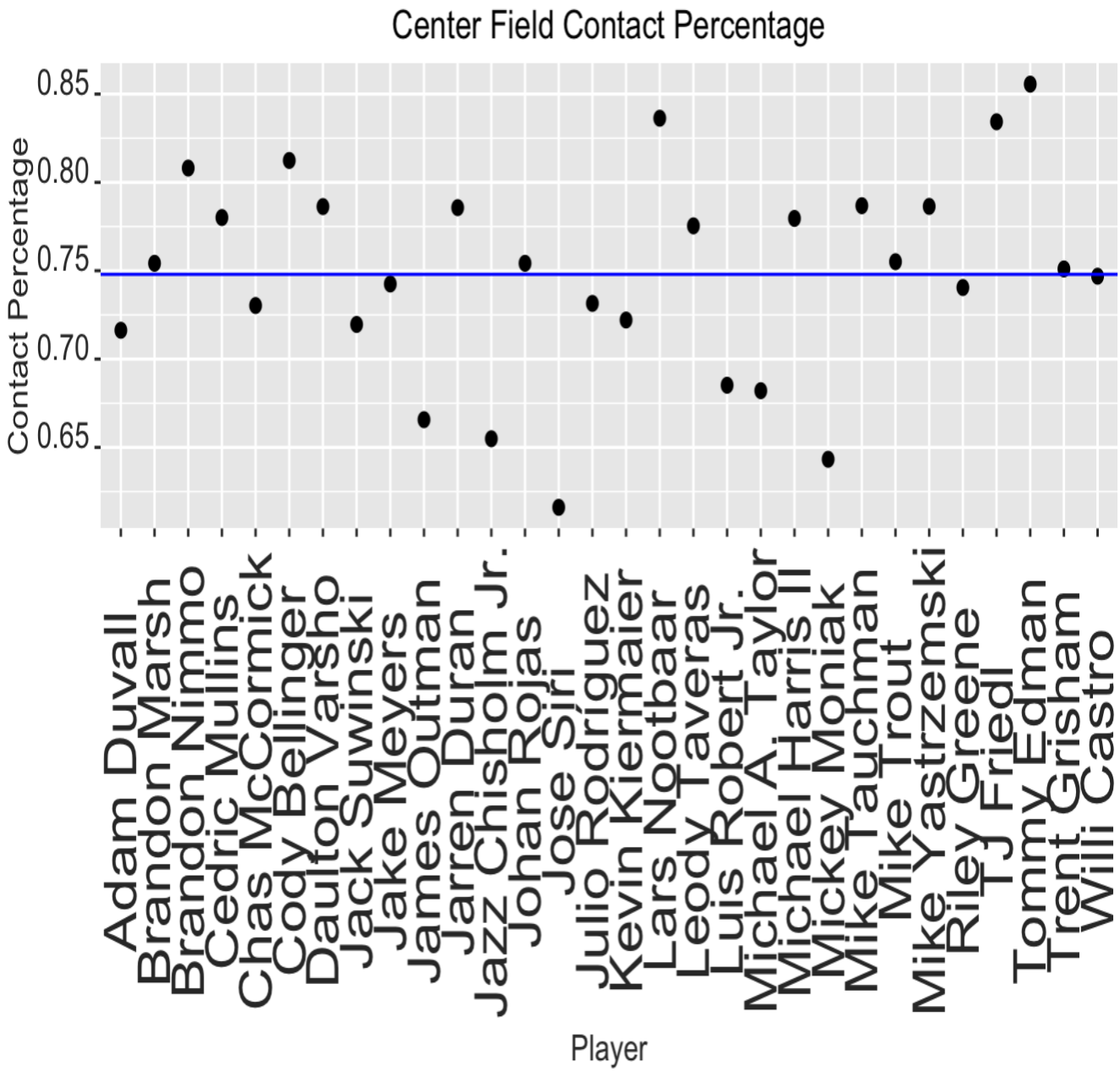

```
ggplot(data = centerfieldclean, aes(x = PlayerName, y = WPA)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(centerfieldclean$WPA), color="blue") +
  ggtitle(bquote('Center Field WPA+')) +
  theme(plot.title = element_text(hjust = 0.5))+
  ylab("WPA") +
  xlab(bquote('Player')) +
  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),
```

```
        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```




```
ggplot(data = centerfieldclean, aes(x = PlayerName, y = Contact_pct)) +
  geom_point(size = 2) +
  geom_hline(yintercept = mean(centerfieldclean$Contact_pct), color="blue") +
```

```
  ggtitle('Center Field Contact Percentage') +

  theme(plot.title = element_text(hjust = 0.5))+

  ylab("Contact Percentage") +

  xlab(bquote('Player')) +

  theme(axis.text.x = element_text(color = "grey20", size = 20, angle = 90, hjust = .5
, vjust = .5, face = "plain"),

        axis.text.y = element_text(color = "grey20", size = 12, angle = 0, hjust = 1,
vjust = 0, face = "plain"),

        axis.title.x = element_text(color = "grey20", size = 12, angle = 0, hjust = .5
, vjust = 0, face = "plain"),

        axis.title.y = element_text(color = "grey20", size = 12, angle = 90, hjust = .
5, vjust = .5, face = "plain"))
```




Determine the additional runs we could expect based on seasonal data and possibly through simulation. You can follow the approach outlined in Mathletics (Chapters 3 and 4) or Chapter 9 of Analyzing Baseball Data with R. You may also attempt to come up with a novel approach on your own.

Lets switch Teoscar Hernandez and Isaac Paredes, and start by looking at Hernandez's runs created.

```
totalbases <- (rightfield$'1B') + ((rightfield$'2B' * 2 )) + ((rightfield$'3B' * 3 ))
+ ((rightfield$HR * 4 ))


rightfield$totalbases <- totalbases
```

```
Hernandezstats <- filter(rightfield, PlayerName == "Teoscar Hernández")



Hernandezouts <- (((0.982)*Hernandezstats$AB) - Hernandezstats$H+ Hernandezstats$GDP +
Hernandezstats$SF + Hernandezstats$SB + Hernandezstats$CS) / 26.72


HernandezRunsCreated <- ((Hernandezstats$H + Hernandezstats$BB + Hernandezstats$HBP) *
(Hernandezstats$totalbases)) / ((Hernandezstats$AB) + Hernandezstats$BB + Hernandezsta
ts$HBP)


Hernandezrunscreatedpergame <- HernandezRunsCreated/Hernandezouts
```

Now for Nolan Jones

```
Jonesstats <- filter(rightfield, PlayerName == "Nolan Jones")


Jonesouts <- (((0.982)*Jonesstats$AB) - Jonesstats$H+ Jonesstats$GDP + Jonesstats$SF +
Jonesstats$SB + Jonesstats$CS) / 26.72


JonesRunsCreated <- ((Jonesstats$H + Jonesstats$BB + Jonesstats$HBP) * (Jonesstats$tot
albases)) / ((Jonesstats$AB) + Jonesstats$BB + Jonesstats$HBP)


Jonesrunscreatedpergame <- JonesRunsCreated/Jonesouts


head(Jonesrunscreatedpergame)

## [1] 7.477149

head(Hernandezrunscreatedpergame)

## [1] 4.722677
```

We can see from this that Jones is creating 7.48 runs per game, compared to Hernandez, who was creating 4.72 runs per game