Capstone Data Project

Chia-Yu Wei

2023-2-15

## Data

We use the data from [Sean Lahaman’s Website](http://www.seanlahman.com/baseball-archive/statistics/). It’s a useful score for baseball statistics.

# Import data  
batting <- read.csv('batting.csv')  
head(batting)

## playerID yearID stint teamID lgID G G\_batting AB R H X2B X3B HR RBI SB CS  
## 1 aardsda01 2004 1 SFN NL 11 11 0 0 0 0 0 0 0 0 0  
## 2 aardsda01 2006 1 CHN NL 45 43 2 0 0 0 0 0 0 0 0  
## 3 aardsda01 2007 1 CHA AL 25 2 0 0 0 0 0 0 0 0 0  
## 4 aardsda01 2008 1 BOS AL 47 5 1 0 0 0 0 0 0 0 0  
## 5 aardsda01 2009 1 SEA AL 73 3 0 0 0 0 0 0 0 0 0  
## 6 aardsda01 2010 1 SEA AL 53 4 0 0 0 0 0 0 0 0 0  
## BB SO IBB HBP SH SF GIDP G\_old  
## 1 0 0 0 0 0 0 0 11  
## 2 0 0 0 0 1 0 0 45  
## 3 0 0 0 0 0 0 0 2  
## 4 0 1 0 0 0 0 0 5  
## 5 0 0 0 0 0 0 0 NA  
## 6 0 0 0 0 0 0 0 NA

It’s a data with 24 variables.

str(batting)

## 'data.frame': 97889 obs. of 24 variables:  
## $ playerID : chr "aardsda01" "aardsda01" "aardsda01" "aardsda01" ...  
## $ yearID : int 2004 2006 2007 2008 2009 2010 2012 1954 1955 1956 ...  
## $ stint : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ teamID : chr "SFN" "CHN" "CHA" "BOS" ...  
## $ lgID : chr "NL" "NL" "AL" "AL" ...  
## $ G : int 11 45 25 47 73 53 1 122 153 153 ...  
## $ G\_batting: int 11 43 2 5 3 4 NA 122 153 153 ...  
## $ AB : int 0 2 0 1 0 0 NA 468 602 609 ...  
## $ R : int 0 0 0 0 0 0 NA 58 105 106 ...  
## $ H : int 0 0 0 0 0 0 NA 131 189 200 ...  
## $ X2B : int 0 0 0 0 0 0 NA 27 37 34 ...  
## $ X3B : int 0 0 0 0 0 0 NA 6 9 14 ...  
## $ HR : int 0 0 0 0 0 0 NA 13 27 26 ...  
## $ RBI : int 0 0 0 0 0 0 NA 69 106 92 ...  
## $ SB : int 0 0 0 0 0 0 NA 2 3 2 ...  
## $ CS : int 0 0 0 0 0 0 NA 2 1 4 ...  
## $ BB : int 0 0 0 0 0 0 NA 28 49 37 ...  
## $ SO : int 0 0 0 1 0 0 NA 39 61 54 ...  
## $ IBB : int 0 0 0 0 0 0 NA NA 5 6 ...  
## $ HBP : int 0 0 0 0 0 0 NA 3 3 2 ...  
## $ SH : int 0 1 0 0 0 0 NA 6 7 5 ...  
## $ SF : int 0 0 0 0 0 0 NA 4 4 7 ...  
## $ GIDP : int 0 0 0 0 0 0 NA 13 20 21 ...  
## $ G\_old : int 11 45 2 5 NA NA NA 122 153 153 ...

### Let’s create the new statistic! For example, for Batting Average, here’s the formula for batting average:

Which means that the Batting Average is equal to H (Hits) divided by AB (At Base). So we’ll do the following to create a new column called **BA** and add it to our data frame:

batting$AVG <- batting$H / batting$AB  
tail(batting$AVG)

## [1] 0.0000000 0.1230769 0.2746479 0.1470588 0.2745098 0.2138728

### Now, calculate On Base Percentage (OBP) and Slugging Percentage (SLG).

#On Base Percentage  
attach(batting)  
batting$OBP <- (H + BB + HBP) / (AB + BB + HBP + SF)

batting$X1B <- H - X2B - X3B - HR  
  
#Slugging Percentage  
batting$SLG <- ((1 \* batting$X1B) + (2 \* X2B) + (3 \* X3B) + (4 \* HR)) / AB

str(batting)

## 'data.frame': 97889 obs. of 28 variables:  
## $ playerID : chr "aardsda01" "aardsda01" "aardsda01" "aardsda01" ...  
## $ yearID : int 2004 2006 2007 2008 2009 2010 2012 1954 1955 1956 ...  
## $ stint : int 1 1 1 1 1 1 1 1 1 1 ...  
## $ teamID : chr "SFN" "CHN" "CHA" "BOS" ...  
## $ lgID : chr "NL" "NL" "AL" "AL" ...  
## $ G : int 11 45 25 47 73 53 1 122 153 153 ...  
## $ G\_batting: int 11 43 2 5 3 4 NA 122 153 153 ...  
## $ AB : int 0 2 0 1 0 0 NA 468 602 609 ...  
## $ R : int 0 0 0 0 0 0 NA 58 105 106 ...  
## $ H : int 0 0 0 0 0 0 NA 131 189 200 ...  
## $ X2B : int 0 0 0 0 0 0 NA 27 37 34 ...  
## $ X3B : int 0 0 0 0 0 0 NA 6 9 14 ...  
## $ HR : int 0 0 0 0 0 0 NA 13 27 26 ...  
## $ RBI : int 0 0 0 0 0 0 NA 69 106 92 ...  
## $ SB : int 0 0 0 0 0 0 NA 2 3 2 ...  
## $ CS : int 0 0 0 0 0 0 NA 2 1 4 ...  
## $ BB : int 0 0 0 0 0 0 NA 28 49 37 ...  
## $ SO : int 0 0 0 1 0 0 NA 39 61 54 ...  
## $ IBB : int 0 0 0 0 0 0 NA NA 5 6 ...  
## $ HBP : int 0 0 0 0 0 0 NA 3 3 2 ...  
## $ SH : int 0 1 0 0 0 0 NA 6 7 5 ...  
## $ SF : int 0 0 0 0 0 0 NA 4 4 7 ...  
## $ GIDP : int 0 0 0 0 0 0 NA 13 20 21 ...  
## $ G\_old : int 11 45 2 5 NA NA NA 122 153 153 ...  
## $ AVG : num NaN 0 NaN 0 NaN ...  
## $ OBP : num NaN 0 NaN 0 NaN ...  
## $ X1B : int 0 0 0 0 0 0 NA 85 116 126 ...  
## $ SLG : num NaN 0 NaN 0 NaN ...

## Merging Salary Data with Batting Data

Let’s merge data about salary with the player data above.

salary <- read.csv('Salaries.csv')  
head(salary)

## yearID teamID lgID playerID salary  
## 1 1985 BAL AL murraed02 1472819  
## 2 1985 BAL AL lynnfr01 1090000  
## 3 1985 BAL AL ripkeca01 800000  
## 4 1985 BAL AL lacyle01 725000  
## 5 1985 BAL AL flanami01 641667  
## 6 1985 BAL AL boddimi01 625000

Since our salary data starts from 1985 but batting data is starting from 1871, we should select the data after 1985.

batting <- subset(batting, yearID >= 1985)  
summary(batting)

## playerID yearID stint teamID   
## Length:35652 Min. :1985 Min. :1.00 Length:35652   
## Class :character 1st Qu.:1993 1st Qu.:1.00 Class :character   
## Mode :character Median :2000 Median :1.00 Mode :character   
## Mean :2000 Mean :1.08   
## 3rd Qu.:2007 3rd Qu.:1.00   
## Max. :2013 Max. :4.00   
##   
## lgID G G\_batting AB   
## Length:35652 Min. : 1.0 Min. : 0.00 Min. : 0.0   
## Class :character 1st Qu.: 14.0 1st Qu.: 4.00 1st Qu.: 3.0   
## Mode :character Median : 34.0 Median : 27.00 Median : 47.0   
## Mean : 51.7 Mean : 46.28 Mean :144.7   
## 3rd Qu.: 77.0 3rd Qu.: 77.00 3rd Qu.:241.0   
## Max. :163.0 Max. :163.00 Max. :716.0   
## NA's :1406 NA's :4377   
## R H X2B X3B   
## Min. : 0.00 Min. : 0.00 Min. : 0.000 Min. : 0.000   
## 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: 0.000 1st Qu.: 0.000   
## Median : 4.00 Median : 8.00 Median : 1.000 Median : 0.000   
## Mean : 19.44 Mean : 37.95 Mean : 7.293 Mean : 0.824   
## 3rd Qu.: 30.00 3rd Qu.: 61.00 3rd Qu.:11.000 3rd Qu.: 1.000   
## Max. :152.00 Max. :262.00 Max. :59.000 Max. :23.000   
## NA's :4377 NA's :4377 NA's :4377 NA's :4377   
## HR RBI SB CS   
## Min. : 0.000 Min. : 0.00 Min. : 0.000 Min. : 0.000   
## 1st Qu.: 0.000 1st Qu.: 0.00 1st Qu.: 0.000 1st Qu.: 0.000   
## Median : 0.000 Median : 3.00 Median : 0.000 Median : 0.000   
## Mean : 4.169 Mean : 18.41 Mean : 2.811 Mean : 1.219   
## 3rd Qu.: 5.000 3rd Qu.: 27.00 3rd Qu.: 2.000 3rd Qu.: 1.000   
## Max. :73.000 Max. :165.00 Max. :110.000 Max. :29.000   
## NA's :4377 NA's :4377 NA's :4377 NA's :4377   
## BB SO IBB HBP   
## Min. : 0.00 Min. : 0.00 Min. : 0.000 Min. : 0.000   
## 1st Qu.: 0.00 1st Qu.: 1.00 1st Qu.: 0.000 1st Qu.: 0.000   
## Median : 3.00 Median : 12.00 Median : 0.000 Median : 0.000   
## Mean : 14.06 Mean : 27.03 Mean : 1.171 Mean : 1.273   
## 3rd Qu.: 21.00 3rd Qu.: 42.00 3rd Qu.: 1.000 3rd Qu.: 1.000   
## Max. :232.00 Max. :223.00 Max. :120.000 Max. :35.000   
## NA's :4377 NA's :4377 NA's :4378 NA's :4387   
## SH SF GIDP G\_old   
## Min. : 0.000 Min. : 0.000 Min. : 0.00 Min. : 0.0   
## 1st Qu.: 0.000 1st Qu.: 0.000 1st Qu.: 0.00 1st Qu.: 11.0   
## Median : 0.000 Median : 0.000 Median : 1.00 Median : 32.0   
## Mean : 1.465 Mean : 1.212 Mean : 3.25 Mean : 49.7   
## 3rd Qu.: 2.000 3rd Qu.: 2.000 3rd Qu.: 5.00 3rd Qu.: 77.0   
## Max. :39.000 Max. :17.000 Max. :35.00 Max. :163.0   
## NA's :4377 NA's :4378 NA's :4377 NA's :5189   
## AVG OBP X1B SLG   
## Min. :0.000 Min. :0.000 Min. : 0.00 Min. :0.000   
## 1st Qu.:0.136 1st Qu.:0.188 1st Qu.: 0.00 1st Qu.:0.167   
## Median :0.233 Median :0.296 Median : 6.00 Median :0.333   
## Mean :0.205 Mean :0.262 Mean : 25.66 Mean :0.304   
## 3rd Qu.:0.274 3rd Qu.:0.342 3rd Qu.: 42.00 3rd Qu.:0.423   
## Max. :1.000 Max. :1.000 Max. :225.00 Max. :4.000   
## NA's :8905 NA's :8821 NA's :4377 NA's :8905

### Let’s merge the salary data and batting data.

combo <- merge(batting, salary, by = c('playerID', 'yearID'))  
head(combo)

## playerID yearID stint teamID.x lgID.x G G\_batting AB R H X2B X3B HR RBI  
## 1 aardsda01 2004 1 SFN NL 11 11 0 0 0 0 0 0 0  
## 2 aardsda01 2007 1 CHA AL 25 2 0 0 0 0 0 0 0  
## 3 aardsda01 2008 1 BOS AL 47 5 1 0 0 0 0 0 0  
## 4 aardsda01 2009 1 SEA AL 73 3 0 0 0 0 0 0 0  
## 5 aardsda01 2010 1 SEA AL 53 4 0 0 0 0 0 0 0  
## 6 aardsda01 2012 1 NYA AL 1 NA NA NA NA NA NA NA NA  
## SB CS BB SO IBB HBP SH SF GIDP G\_old AVG OBP X1B SLG teamID.y lgID.y salary  
## 1 0 0 0 0 0 0 0 0 0 11 NaN NaN 0 NaN SFN NL 300000  
## 2 0 0 0 0 0 0 0 0 0 2 NaN NaN 0 NaN CHA AL 387500  
## 3 0 0 0 1 0 0 0 0 0 5 0 0 0 0 BOS AL 403250  
## 4 0 0 0 0 0 0 0 0 0 NA NaN NaN 0 NaN SEA AL 419000  
## 5 0 0 0 0 0 0 0 0 0 NA NaN NaN 0 NaN SEA AL 2750000  
## 6 NA NA NA NA NA NA NA NA NA NA NA NA NA NA NYA AL 500000

## Analyzing the Lost Player

From story about *The 2022 Oakland A’s*, the Oakland A’s lost 3 key players during the off-season. We’ll want to get their stats to see what we have to replace. The players lost were: first baseman 2000 AL MVP Jason Giambi (giambja01) to the New York Yankees, outfielder Johnny Damon (damonjo01) to the Boston Red Sox and infielder Rainer Gustavo “Ray” Olmedo (‘saenzol01’).

### Use the subset() function to get a data frame called lost\_players from the combo data frame consisting of those 3 players. Hint: Try to figure out how to use %in% to avoid a bunch of or statements!

lost\_player <- subset(combo,  
 playerID %in% c('giambja01','damonjo01','saenzol01'))  
  
  
#44x31  
dim(lost\_player)

## [1] 44 31

We only get the data which yearID is 2001, and reduce the variables to what we need.

lost\_player <- subset(lost\_player, yearID == 2001)  
new\_lost <- lost\_player[, c('playerID', 'H', 'X2B', 'X3B', 'HR', 'OBP', 'SLG', 'AB')]  
head(new\_lost)

## playerID H X2B X3B HR OBP SLG AB  
## 5141 damonjo01 165 34 4 9 0.3235294 0.3633540 644  
## 7878 giambja01 178 47 2 38 0.4769001 0.6596154 520  
## 20114 saenzol01 67 21 1 9 0.2911765 0.3836066 305

## Task

Now we have all the information we need! Here is your final task - Find Replacement Players for the key three players we lost! However, you have three constraints:

* The total combined salary of the three players can not exceed 15 million dollars.
* Their combined number of At Bats (AB) needs to be equal to or greater than the lost players(minimum is 520 in lost platers) ).
* Their mean OBP had to equal to or greater than the mean OBP of the lost players

library(dplyr)

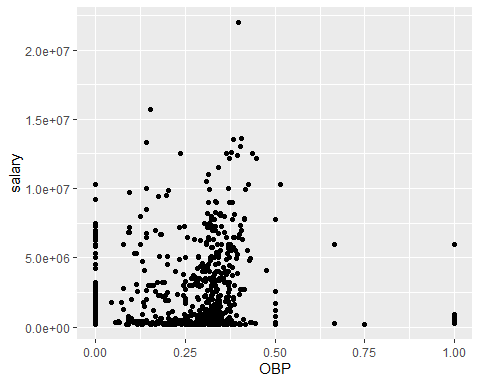
##   
## 載入套件：'dplyr'

## 下列物件被遮斷自 'package:stats':  
##   
## filter, lag

## 下列物件被遮斷自 'package:base':  
##   
## intersect, setdiff, setequal, union

library(ggplot2)  
avail.players <- filter(combo, yearID == 2001)  
ggplot(avail.players, aes(x = OBP, y = salary)) + geom\_point()

## Warning: Removed 168 rows containing missing values (`geom\_point()`).



avail.players <- filter(avail.players, salary < 8000000, OBP > 0, AB >= 520)  
candidate <- head(arrange(avail.players,desc(OBP)),10)  
(possible <- candidate[,c('playerID','OBP','AB','salary')])

## playerID OBP AB salary  
## 1 giambja01 0.4769001 520 4103333  
## 2 heltoto01 0.4316547 587 4950000  
## 3 berkmla01 0.4302326 577 305000  
## 4 gonzalu01 0.4285714 609 4833333  
## 5 thomeji01 0.4161491 526 7875000  
## 6 alomaro01 0.4146707 575 7750000  
## 7 gilesbr02 0.4035608 576 7333333  
## 8 pujolal01 0.4029630 590 200000  
## 9 olerujo01 0.4011799 572 6700000  
## 10 bagweje01 0.3974895 600 6500000

We select some players who is satisfied with our condition, and sort the data by *OBP*. The first choice is ‘giambja01’, but we can not choose him agian. Therefore, we should choose player rank2 to 4.

possible[2:4, ]

## playerID OBP AB salary  
## 2 heltoto01 0.4316547 587 4950000  
## 3 berkmla01 0.4302326 577 305000  
## 4 gonzalu01 0.4285714 609 4833333