MoneyBall Project - Solutions

Rules of Baseball

You don't need to know much about Baseball to complete this exercise. If you're totally unfamiliar with Baseball, check out this useful explanatory video!

Background

Source: Wikipedia

The 2002 Oakland A's

The Oakland Athletics' 2002 season was the team's 35th in Oakland, California. It was also the 102nd season in franchise history. The Athletics finished first in the American League West with a record of 103-59.

The Athletics' 2002 campaign ranks among the most famous in franchise history. Following the 2001 season, Oakland saw the departure of three key players (the lost boys). Billy Beane, the team's general manager, responded with a series of under-the-radar free agent signings. The new-look Athletics, despite a comparative lack of star power, surprised the baseball world

by besting the 2001 team's regular season record. The team is most famous, however, for winning 20 consecutive games between August 13 and September 4, 2002.[1] The Athletics' season was the subject of Michael Lewis' 2003 book Moneyball: The Art of Winning an Unfair Game (as Lewis was given the opportunity to follow the team around throughout that season)

This project is based off the book written by Michael Lewis (later turned into a movie).

Moneyball Book

The central premise of book *Moneyball* is that the collective wisdom of baseball insiders (including players, managers, coaches, scouts, and the front office) over the past century is subjective and often flawed. Statistics such as stolen bases, runs batted in, and batting average, typically used to gauge players, are relics of a 19th-century view of the game and the statistics available at that time. The book argues that the Oakland A's' front office took advantage of more analytical gauges of player performance to field a team that could better compete against richer competitors in Major League Baseball (MLB).

Rigorous statistical analysis had demonstrated that on-base percentage and slugging percentage are better indicators of offensive success, and the A's became convinced that these qualities were cheaper to obtain on the open market than more historically valued qualities such as speed and contact. These observations often flew in the face of conventional baseball wisdom and the beliefs of many baseball scouts and executives.

By re-evaluating the strategies that produce wins on the field, the 2002 Athletics, with approximately US 44 million dollars in salary, were competitive with larger market teams such as the New York Yankees, who spent over US$125 million in payroll that same season.

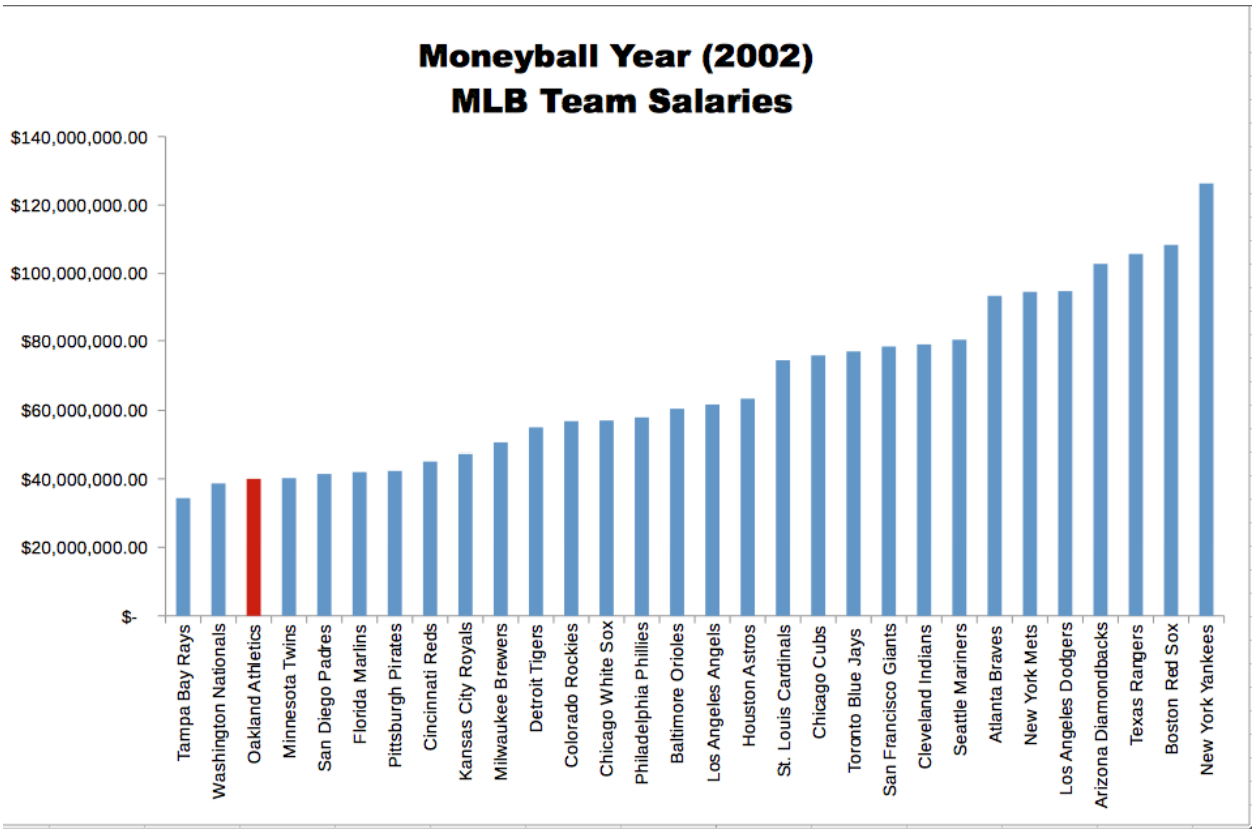
Because of the team's smaller revenues, Oakland is forced to find players undervalued by the market, and their system for finding value in undervalued players has proven itself thus far. This approach brought the A's to the playoffs in 2002 and 2003.

In this project we'll work with some data and with the goal of trying to find replacement players for the ones lost at the start of the off-season - During the 2001–02 offseason, the team lost three key free agents to larger market teams: 2000 AL MVP Jason Giambi to the New York Yankees, outfielder Johnny Damon to the Boston Red Sox, and closer Jason Isringhausen to the St. Louis Cardinals.

The main goal of this project is for you to feel comfortable working with R on real data to try and derive actionable insights!

Let's get started!

Follow the steps outlined in bold below using your new R skills and help the Oakland A's recruit under-valued players!



Data

We'll be using data from [Sean Lahaman's Website] a very useful source for baseball statistics. The documentation for the csv files is located in the **readme2013.txt** file. You may need to reference this to understand what acronyms stand for.

Use R to open the Batting.csv file and assign it to a dataframe called batting using *read.csv*

In [1]:

```
batting <- read.csv('Batting.csv')
```

Use head() to check out the batting

In [2]:

```
head(batting)
```

Out[2]:

	playerID	yearID	stint	teamID	lgID	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old
1	aardsda01	2004	1	SFN	NL	11	11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11

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

Use str() to check the structure. Pay close attention to how columns that start with a number get an 'X' in front of them! You'll need to know this to call those columns!

In [3]:

```
str(batting)
```

```
'data.frame':  97889 obs. of  24 variables:
 $ playerID : Factor w/ 18107 levels "aardsda01","aaronha01",..: 1 1
1 1 1 1 1 2 2 2 ...
 $ 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   : Factor w/ 149 levels "ALT","ANA","ARI",..: 117 35 33 16
116 116 93 80 80 80 ...
 $ lgID     : Factor w/ 6 levels "AA","AL","FL",..: 4 4 2 2 2 2 2 4 4
4 ...
 $ 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 ...
```

Make sure you understand how to call the columns by using the $ symbol.

Call the head() of the first five rows of AB (At Bats) column

In [4]:

```
head(batting$AB)
```

Out[4]:

```
0 2 0 1 0 0
```

Call the head of the doubles (X2B) column

In [5]:

```
head(batting$X2B)
```

Out[5]:

```
0 0 0 0 0 0
```

Quick Note: If you used fread() to use data.table, then you won't need to worry about these X in front of numbers, instead you would use something like:

```
batting[,'2B',with=FALSE]
```

There's a few more ways of doing detailed here.

Alright! Let's move on!

Feature Engineering

We need to add three more statistics that were used in Moneyball! These are:

- Batting Average
- On Base Percentage
- Slugging Percentage

Click on the links provided and search the wikipedia page for the formula for creating the new statistic! For example, for Batting Average, you'll need to scroll down until you see:

$$ AVG = \frac{H}{AB} $$

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:

In [6]:

```
batting$BA <- batting$H / batting$AB
```

After doing this operation, check the last 5 entries of the BA column of your data frame and it should look like this:

In [7]:

```
tail(batting$BA,5)
```

Out[7]:

0.123076923076923 0.274647887323944 0.147058823529412 0.274509803921569 0.213872832369942

Now do the same for some new columns! On Base Percentage (OBP) and Slugging Percentage (SLG). Hint: For SLG, you need 1B (Singles), this isn't in your data frame. However you can calculate it by subtracting doubles,triples, and home runs from total hits (H): 1B = H-2B-3B-HR

- **Create an OBP Column**
- **Create an SLG Column**

In [8]:

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

In [9]:

```
# Creating X1B (Singles)
batting$X1B <- batting$H - batting$X2B - batting$X3B - batting$HR
```

In [10]:

```
# Creating Slugging Average (SLG)
batting$SLG <- ((1 * batting$X1B) + (2 * batting$X2B) + (3 *
batting$X3B) + (4 * batting$HR) ) / batting$AB
```

Check the structure of your data frame using str()

In [11]:

```
str(batting)
```

```
'data.frame':  97889 obs. of  28 variables:
 $ playerID : Factor w/ 18107 levels "aardsda01","aaronha01",..: 1 1
1 1 1 1 1 2 2 2 ...
 $ 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   : Factor w/ 149 levels "ALT","ANA","ARI",..: 117 35 33 16
116 116 93 80 80 80 ...
 $ lgID     : Factor w/ 6 levels "AA","AL","FL",..: 4 4 2 2 2 2 2 4 4
4 ...
 $ 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 ...
$ BA       : 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

We know we don't just want the best players, we want the most *undervalued* players, meaning we will also need to know current salary information! We have salary information in the csv file 'Salaries.csv'.

Complete the following steps to merge the salary data with the player stats!

Load the Salaries.csv file into a dataframe called sal using read.csv

In [12]:

```
sal <- read.csv('Salaries.csv')
```

Use summary to get a summary of the batting data frame and notice the minimum year in the yearID column. Our batting data goes back to 1871! Our salary data starts at 1985, meaning we need to remove the batting data that occured before 1985.

Use subset() to reassign batting to only contain data from 1985 and onwards

In [13]:

```
summary(batting)
```

Out[13]:

```
      playerID         yearID         stint           teamID
lgID
```

```
 mcguide01:   31   Min.   :1871   Min.   :1.000   CHN    : 4720   AA
: 1890
 henderi01:   29   1st Qu.:1931   1st Qu.:1.000   PHI    : 4621   AL
:44369
 newsobo01:   29   Median :1970   Median :1.000   PIT    : 4575   FL
:  470
 johnto01 :   28   Mean   :1962   Mean   :1.077   SLN    : 4535   NL
:49944
 kaatji01 :   28   3rd Qu.:1995   3rd Qu.:1.000   CIN    : 4393   PL
:  147
 ansonca01:   27   Max.   :2013   Max.   :5.000   CLE    : 4318   UA
:  332
 (Other)  :97717                                  (Other):70727
NA's:  737
       G            G_batting           AB              R
 Min.   :  1.00   Min.   :  0.00   Min.   :  0.0   Min.   :  0.00
 1st Qu.: 13.00   1st Qu.:  7.00   1st Qu.:  9.0   1st Qu.:  0.00
 Median : 35.00   Median : 32.00   Median : 61.0   Median :  5.00
 Mean   : 51.65   Mean   : 49.13   Mean   :154.1   Mean   : 20.47
 3rd Qu.: 81.00   3rd Qu.: 81.00   3rd Qu.:260.0   3rd Qu.: 31.00
 Max.   :165.00   Max.   :165.00   Max.   :716.0   Max.   :192.00
                  NA's   :1406     NA's   :6413    NA's   :6413
       H              X2B             X3B              HR
 Min.   :  0.00   Min.   : 0.0   Min.   : 0.000   Min.   : 0.000
 1st Qu.:  1.00   1st Qu.: 0.0   1st Qu.: 0.000   1st Qu.: 0.000
 Median : 12.00   Median : 2.0   Median : 0.000   Median : 0.000
 Mean   : 40.37   Mean   : 6.8   Mean   : 1.424   Mean   : 3.002
 3rd Qu.: 66.00   3rd Qu.:10.0   3rd Qu.: 2.000   3rd Qu.: 3.000
 Max.   :262.00   Max.   :67.0   Max.   :36.000   Max.   :73.000
 NA's   :6413     NA's   :6413   NA's   :6413     NA's   :6413
     RBI              SB               CS              BB
 Min.   :  0.00   Min.   :  0.000   Min.   : 0.000   Min.   :  0.00
 1st Qu.:  0.00   1st Qu.:  0.000   1st Qu.: 0.000   1st Qu.:  0.00
 Median :  5.00   Median :  0.000   Median : 0.000   Median :  4.00
 Mean   : 18.47   Mean   :  3.265   Mean   : 1.385   Mean   : 14.21
 3rd Qu.: 28.00   3rd Qu.:  2.000   3rd Qu.: 1.000   3rd Qu.: 21.00
 Max.   :191.00   Max.   :138.000   Max.   :42.000   Max.   :232.00
 NA's   :6837     NA's   :7713      NA's   :29867    NA's   :6413
       SO             IBB              HBP              SH
 Min.   :  0.00   Min.   :  0.00   Min.   : 0.000   Min.   : 0.000
 1st Qu.:  2.00   1st Qu.:  0.00   1st Qu.: 0.000   1st Qu.: 0.000
 Median : 11.00   Median :  0.00   Median : 0.000   Median : 1.000
 Mean   : 21.95   Mean   :  1.28   Mean   : 1.136   Mean   : 2.564
 3rd Qu.: 31.00   3rd Qu.:  1.00   3rd Qu.: 1.000   3rd Qu.: 3.000
 Max.   :223.00   Max.   :120.00   Max.   :51.000   Max.   :67.000
 NA's   :14251    NA's   :42977    NA's   :9233     NA's   :12751
       SF             GIDP            G_old             BA
 Min.   : 0.0    Min.   : 0.00   Min.   :  0.00   Min.   :0.000
 1st Qu.: 0.0    1st Qu.: 0.00   1st Qu.: 11.00   1st Qu.:0.148
 Median : 0.0    Median : 1.00   Median : 34.00   Median :0.231
```

```
 Mean   : 1.2    Mean   : 3.33   Mean   : 50.99   Mean   :0.209
 3rd Qu.: 2.0    3rd Qu.: 5.00   3rd Qu.: 82.00   3rd Qu.:0.275
 Max.   :19.0    Max.   :36.00   Max.   :165.00   Max.   :1.000
 NA's   :42446   NA's   :32521   NA's   :5189     NA's   :13520
      OBP             X1B             SLG
 Min.   :0.00    Min.   :  0.00   Min.   :0.000
 1st Qu.:0.19    1st Qu.:  1.00   1st Qu.:0.179
 Median :0.29    Median :  9.00   Median :0.309
 Mean   :0.26    Mean   : 29.14   Mean   :0.291
 3rd Qu.:0.34    3rd Qu.: 48.00   3rd Qu.:0.397
 Max.   :1.00    Max.   :225.00   Max.   :4.000
 NA's   :49115   NA's   :6413     NA's   :13520
```

In [14]:

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

Now use summary again to make sure the subset reassignment worked, your yearID min should be 1985

In [15]:

```
summary(batting)
```

Out[15]:

```
      playerID         yearID         stint          teamID      lgID
 moyerja01:   27   Min.   :1985   Min.   :1.00   SDN    : 1313   AA:
0
 mulhote01:   26   1st Qu.:1993   1st Qu.:1.00   CLE    : 1306
AL:17226
 weathda01:   26   Median :2000   Median :1.00   PIT    : 1299   FL:
0
 maddugr01:   25   Mean   :2000   Mean   :1.08   NYN    : 1297
NL:18426
 sierrru01:   25   3rd Qu.:2007   3rd Qu.:1.00   BOS    : 1279   PL:
0
 thomeji01:   25   Max.   :2013   Max.   :4.00   CIN    : 1279   UA:
0
 (Other)  :35498                                (Other):27879
       G           G_batting           AB              R
 Min.   :  1.0   Min.   :  0.00   Min.   :  0.0   Min.   :  0.00
 1st Qu.: 14.0   1st Qu.:  4.00   1st Qu.:  3.0   1st Qu.:  0.00
 Median : 34.0   Median : 27.00   Median : 47.0   Median :  4.00
 Mean   : 51.7   Mean   : 46.28   Mean   :144.7   Mean   : 19.44
 3rd Qu.: 77.0   3rd Qu.: 77.00   3rd Qu.:241.0   3rd Qu.: 30.00
 Max.   :163.0   Max.   :163.00   Max.   :716.0   Max.   :152.00
                 NA's   :1406     NA's   :4377    NA's   :4377
       H              X2B             X3B              HR
 Min.   :  0.00   Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
 1st Qu.:  0.00   1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.: 0.000
 Median :  8.00   Median : 1.000   Median : 0.000   Median : 0.000
 Mean   : 37.95   Mean   : 7.293   Mean   : 0.824   Mean   : 4.169
 3rd Qu.: 61.00   3rd Qu.:11.000   3rd Qu.: 1.000   3rd Qu.: 5.000
```

```
Max.   :262.00   Max.   :59.000   Max.   :23.000   Max.   :73.000
NA's   :4377     NA's   :4377     NA's   :4377     NA's   :4377
      RBI               SB               CS               BB
Min.   :  0.00   Min.   :  0.000   Min.   : 0.000   Min.   :  0.00
1st Qu.:  0.00   1st Qu.:  0.000   1st Qu.: 0.000   1st Qu.:  0.00
Median :  3.00   Median :  0.000   Median : 0.000   Median :  3.00
Mean   : 18.41   Mean   :  2.811   Mean   : 1.219   Mean   : 14.06
3rd Qu.: 27.00   3rd Qu.:  2.000   3rd Qu.: 1.000   3rd Qu.: 21.00
Max.   :165.00   Max.   :110.000   Max.   :29.000   Max.   :232.00
NA's   :4377     NA's   :4377      NA's   :4377     NA's   :4377
      SO               IBB               HBP               SH
Min.   :  0.00   Min.   :  0.000   Min.   : 0.000   Min.   : 0.000
1st Qu.:  1.00   1st Qu.:  0.000   1st Qu.: 0.000   1st Qu.: 0.000
Median : 12.00   Median :  0.000   Median : 0.000   Median : 0.000
Mean   : 27.03   Mean   :  1.171   Mean   : 1.273   Mean   : 1.465
3rd Qu.: 42.00   3rd Qu.:  1.000   3rd Qu.: 1.000   3rd Qu.: 2.000
Max.   :223.00   Max.   :120.000   Max.   :35.000   Max.   :39.000
NA's   :4377     NA's   :4378      NA's   :4387     NA's   :4377
      SF               GIDP             G_old             BA
Min.   : 0.000   Min.   : 0.00   Min.   :  0.0   Min.   :0.000
1st Qu.: 0.000   1st Qu.: 0.00   1st Qu.: 11.0   1st Qu.:0.136
Median : 0.000   Median : 1.00   Median : 32.0   Median :0.233
Mean   : 1.212   Mean   : 3.25   Mean   : 49.7   Mean   :0.205
3rd Qu.: 2.000   3rd Qu.: 5.00   3rd Qu.: 77.0   3rd Qu.:0.274
Max.   :17.000   Max.   :35.00   Max.   :163.0   Max.   :1.000
NA's   :4378     NA's   :4377    NA's   :5189    NA's   :8905
     OBP              X1B              SLG
Min.   :0.000   Min.   :  0.00   Min.   :0.000
1st Qu.:0.188   1st Qu.:  0.00   1st Qu.:0.167
Median :0.296   Median :  6.00   Median :0.333
Mean   :0.262   Mean   : 25.66   Mean   :0.304
3rd Qu.:0.342   3rd Qu.: 42.00   3rd Qu.:0.423
Max.   :1.000   Max.   :225.00   Max.   :4.000
NA's   :8821    NA's   :4377     NA's   :8905
```

Now it is time to merge the batting data with the salary data! Since we have players playing multiple years, we'll have repetitions of playerIDs for multiple years, meaning we want to merge on *both* players and years.

Use the merge() function to merge the batting and sal data frames by c('playerID','yearID'). Call the new data frame combo

In [16]:

```
combo <- merge(batting,sal,by=c('playerID','yearID'))
```

Use summary to check the data

In [17]:

```
summary(combo)
```

Out[17]:

```
      playerID         yearID         stint          teamID.x
lgID.x
 moyerja01:   27   Min.   :1985   Min.   :1.000   LAN    :  940   AA:
0
 thomeji01:   25   1st Qu.:1993   1st Qu.:1.000   PHI    :  937
AL:12292
 weathda01:   25   Median :1999   Median :1.000   BOS    :  935   FL:
0
 vizquom01:   24   Mean   :1999   Mean   :1.098   NYA    :  928
NL:13105
 gaettga01:   23   3rd Qu.:2006   3rd Qu.:1.000   CLE    :  920   PL:
0
 griffke02:   23   Max.   :2013   Max.   :4.000   SDN    :  914   UA:
0
 (Other)  :25250                                  (Other):19823
       G            G_batting           AB              R
 Min.   :  1.00   Min.   :  0.00   Min.   :  0.0   Min.   :  0.00
 1st Qu.: 26.00   1st Qu.:  8.00   1st Qu.:  5.0   1st Qu.:  0.00
 Median : 50.00   Median : 42.00   Median : 85.0   Median :  9.00
 Mean   : 64.06   Mean   : 57.58   Mean   :182.4   Mean   : 24.71
 3rd Qu.:101.00   3rd Qu.:101.00   3rd Qu.:336.0   3rd Qu.: 43.00
 Max.   :163.00   Max.   :163.00   Max.   :716.0   Max.   :152.00
                  NA's   :906      NA's   :2661    NA's   :2661
       H              X2B              X3B              HR
 Min.   :  0.00   Min.   : 0.000   Min.   : 0.000   Min.   : 0.000
 1st Qu.:  1.00   1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.: 0.000
 Median : 19.00   Median : 3.000   Median : 0.000   Median : 1.000
 Mean   : 48.18   Mean   : 9.276   Mean   : 1.033   Mean   : 5.369
 3rd Qu.: 87.25   3rd Qu.:16.000   3rd Qu.: 1.000   3rd Qu.: 7.000
 Max.   :262.00   Max.   :59.000   Max.   :23.000   Max.   :73.000
 NA's   :2661     NA's   :2661     NA's   :2661     NA's   :2661
      RBI              SB               CS              BB
 Min.   :  0.00   Min.   :  0.000   Min.   : 0.00   Min.   :  0.00
 1st Qu.:  0.00   1st Qu.:  0.000   1st Qu.: 0.00   1st Qu.:  0.00
 Median :  8.00   Median :  0.000   Median : 0.00   Median :  6.00
 Mean   : 23.56   Mean   :  3.568   Mean   : 1.54   Mean   : 17.98
 3rd Qu.: 39.00   3rd Qu.:  3.000   3rd Qu.: 2.00   3rd Qu.: 29.00
 Max.   :165.00   Max.   :110.000   Max.   :29.00   Max.   :232.00
 NA's   :2661     NA's   :2661      NA's   :2661    NA's   :2661
       SO              IBB              HBP              SH
 Min.   :  0.00   Min.   :  0.000   Min.   : 0.000   Min.   : 0.000
 1st Qu.:  2.00   1st Qu.:  0.000   1st Qu.: 0.000   1st Qu.: 0.000
 Median : 20.00   Median :  0.000   Median : 0.000   Median : 0.000
 Mean   : 33.52   Mean   :  1.533   Mean   : 1.614   Mean   : 1.786
 3rd Qu.: 55.00   3rd Qu.:  2.000   3rd Qu.: 2.000   3rd Qu.: 2.000
 Max.   :223.00   Max.   :120.000   Max.   :35.000   Max.   :39.000
 NA's   :2661     NA's   :2662      NA's   :2670     NA's   :2661
       SF              GIDP             G_old             BA
 Min.   : 0.000   Min.   : 0.000   Min.   :  0.00   Min.   :0.000
 1st Qu.: 0.000   1st Qu.: 0.000   1st Qu.: 20.00   1st Qu.:0.160
```

```
 Median : 0.000   Median : 2.000   Median : 47.00   Median :0.242
 Mean   : 1.554   Mean   : 4.127   Mean   : 61.43   Mean   :0.212
 3rd Qu.: 2.000   3rd Qu.: 7.000   3rd Qu.:101.00   3rd Qu.:0.276
 Max.   :17.000   Max.   :35.000   Max.   :163.00   Max.   :1.000
 NA's   :2662     NA's   :2661     NA's   :3414     NA's   :5618
      OBP              X1B              SLG           teamID.y
lgID.y
 Min.   :0.000   Min.   :  0.0   Min.   :0.000   CLE    :  935
AL:12304
 1st Qu.:0.208   1st Qu.:  0.0   1st Qu.:0.200   PIT    :  932
NL:13093
 Median :0.305   Median : 13.0   Median :0.351   PHI    :  931
 Mean   :0.270   Mean   : 32.5   Mean   :0.317   SDN    :  923
 3rd Qu.:0.346   3rd Qu.: 59.0   3rd Qu.:0.432   LAN    :  921
 Max.   :1.000   Max.   :225.0   Max.   :4.000   CIN    :  912
 NA's   :5562    NA's   :2661    NA's   :5618    (Other):19843
     salary
 Min.   :       0
 1st Qu.:  255000
 Median :  550000
 Mean   : 1879256
 3rd Qu.: 2150000
 Max.   :33000000
```

Analyzing the Lost Players

As previously mentioned, 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!

In [18]:

```
lost_players <- subset(combo,playerID %in%
c('giambja01','damonjo01','saenzol01') )
```

In [19]:

```
lost_players
```

Out[19]:

	playerID	yearID	stint	teamID.x	lgID.x	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old	BA	OBP	X1B	SLG	teamID.y	lgID.y	salary
51355	damonjo01	1995	1	KCA	AL	47	47	188	32	53	11	5	3	23	7	0	12	22	0	1	2	3	2	47	0.2819149	0.3235294	34	0.4414894	KCA	AL	1090000
51366	damonjo01	1996	1	KCA	AL	145	145	517	61	140	22	5	6	50	25	5	31	64	3	3	10	5	4	145	0.2707093	0.3129496	107	0.3675048	KCA	AL	1800000
51377	damonjo01	1997	1	KCA	AL	146	146	472	70	130	12	8	8	48	16	10	42	70	2	3	6	1	3	146	0.2754237	0.3378378	102	0.3855932	KCA	AL	2400000
51388	damonjo01	1998	1	KCA	AL	161	161	642	104	178	30	10	18	66	26	12	58	84	4	4	3	3	4	161	0.2772586	0.3394625	120	0.4392523	KCA	AL	4600000
51399	damonjo01	1999	1	KCA	AL	145	145	583	101	179	39	9	14	77	36	6	67	50	5	3	3	4	13	145	0.3070326	0.3789954	117	0.4768439	KCA	AL	21000000
51400	damonjo01	2000	1	KCA	AL	159	159	655	136	214	42	10	16	88	46	9	65	60	4	1	8	12	7	159	0.3267176	0.3819918	146	0.4946565	KCA	AL	40000000
514	damonj	200	1	OAK	AL	155	155	644	108	165	34	4	9	49	27	12	61	70	1	5	5	4	7	155	0.2562	0.3235	118	0.3633	OAK	AL	7100000

	playerID	yearID	stint	teamID.x	lgID.x	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old	BA	OBP	X1B	SLG	teamID.y	lgID.y	salary
1	o01	1																							112	294		54			0
51422	damonjo01	2002	1	BOS	AL	154	154	623	118	178	34	11	14	63	31	6	65	70	5	6	3	5	4	154	0.28557143	0.35623232	119	0.44430177	BOS	AL	72500000
51443	damonjo01	2003	1	BOS	AL	145	145	608	103	166	32	6	12	67	30	6	68	74	4	2	6	6	5	145	0.27330263	0.34550292	116	0.40446053	BOS	AL	75000000
51444	damonjo01	2004	1	BOS	AL	150	150	621	123	189	35	6	20	94	19	8	76	71	1	2	0	3	8	150	0.30343478	0.38034199	128	0.47666506	BOS	AL	80000000
51445	damonjo01	2005	1	BOS	AL	148	148	624	117	197	35	6	10	75	18	1	53	69	3	2	0	9	5	148	0.31557051	0.36627911	146	0.43910226	BOS	AL	82500000
51446	damonjo01	2006	1	NYA	AL	149	149	593	115	169	35	5	24	80	25	10	67	85	1	4	2	5	4	149	0.28499166	0.35874444	105	0.48229334	NYA	AL	13000000
51447	damonjo01	2007	1	NYA	AL	141	141	533	93	144	27	2	12	63	27	3	66	79	1	2	1	3	4	141	0.27016689	0.35093934	103	0.39587244	NYA	AL	13000000

	playerID	yearID	stint	teamID.x	lgID.x	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old	BA	OBP	X1B	SLG	teamID.y	lgID.y	salary
51448	damonjo01	2008	1	NYA	AL	143	143	555	95	168	27	5	17	71	29	8	64	82	0	1	2	1	5	143	0.30027027	0.37752013	119	0.46126133	NYA	AL	13000000
51449	damonjo01	2009	1	NYA	AL	143	143	550	107	155	36	3	24	82	12	0	71	98	1	2	2	1	9	NA	0.28181822	0.36538466	92	0.48909099	NYA	AL	13000000
51450	damonjo01	2010	1	DET	AL	145	145	539	81	146	36	5	8	51	11	1	69	90	2	2	2	1	5	NA	0.27708072	0.35515555	97	0.40074021	DET	AL	8000000
51551	damonjo01	2011	1	TBA	AL	150	150	582	79	152	29	7	16	73	19	6	51	92	1	7	2	5	4	150	0.26111684	0.32555814	100	0.41752588	TBA	AL	5250000
78722	giambja01	1995	1	OAK	AL	54	54	176	27	45	7	0	6	25	2	1	28	31	0	3	1	2	4	54	0.25568188	0.36363644	32	0.39777273	OAK	AL	1090000
78733	giambja01	1996	1	OAK	AL	140	140	536	84	156	40	1	20	79	0	1	51	95	3	5	1	5	15	140	0.29104488	0.35510889	95	0.48134333	OAK	AL	1200000
7877	giambja	199	1	OAK	AL	142	142	519	66	152	41	2	20	81	0	1	55	89	3	6	0	8	11	142	0.29228	0.3622	89	0.49451	OAK	AL	2050000

	playerID	yearID	stint	teamID.x	lgID.x	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old	BA	OBP	X1B	SLG	teamID.y	lgID.y	salary
4	01	7																							709	449		83			
7875	giambja01	1998	1	OAK	AL	153	153	562	92	166	28	0	27	110	2	2	81	102	7	5	0	9	16	153	0.29553737	0.38835616	111	0.48932388	OAK	AL	3150000
7876	giambja01	1999	1	OAK	AL	158	158	575	115	181	36	1	33	123	1	1	105	106	6	7	0	8	11	158	0.31478826	0.42158827	111	0.55304335	OAK	AL	21033333
7877	giambja01	2000	1	OAK	AL	152	152	510	108	170	29	1	43	137	2	0	137	96	6	9	0	8	9	152	0.33333333	0.47590336	97	0.64706588	OAK	AL	31033333
7878	giambja01	2001	1	OAK	AL	154	154	520	109	178	47	2	38	120	2	0	129	83	24	13	0	9	17	154	0.34230707	0.47690001	91	0.65961544	OAK	AL	41033333
7879	giambja01	2002	1	NYA	AL	155	155	560	120	176	34	1	41	122	2	2	109	112	4	15	0	5	18	155	0.31428857	0.43541336	100	0.59821443	NYA	AL	10428571
7880	giambja01	2003	1	NYA	AL	156	156	535	97	134	25	0	41	107	2	1	129	140	9	21	0	5	9	156	0.25046673	0.41159442	68	0.52710288	NYA	AL	11428571

	playerID	yearID	stint	teamID.x	lgID.x	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old	BA	OBP	X1B	SLG	teamID.y	lgID.y	salary
78811	giambja01	2004	1	NYA	AL	80	80	264	33	55	9	0	12	40	0	1	47	62	1	8	0	3	5	80	0.208333333	0.341614149	34	0.378787879	NYA	AL	12428571
78812	giambja01	2005	1	NYA	AL	139	139	417	74	113	14	0	32	87	0	0	108	109	5	19	0	1	7	139	0.270983832	0.444036767	67	0.534772722	NYA	AL	13428571
78813	giambja01	2006	1	NYA	AL	139	139	446	92	113	25	0	37	113	2	0	110	106	12	16	0	7	10	139	0.253336332	0.412728077	51	0.558296	NYA	AL	20428571
78814	giambja01	2007	1	NYA	AL	83	83	254	31	60	8	0	14	39	1	0	40	66	2	8	0	1	1	83	0.236622055	0.356435656	38	0.433307079	NYA	AL	23428571
78815	giambja01	2008	1	NYA	AL	145	145	458	68	113	19	1	32	96	2	1	76	111	5	22	0	9	6	145	0.246672499	0.373451313	61	0.502183434	NYA	AL	23428571
78816	giambja01	2009	1	OAK	AL	83	83	269	39	52	13	0	11	40	0	0	50	72	1	7	0	2	6	NA	0.193330866	0.332331711	28	0.364343123	OAK	AL	40000000
78817	giambja	200	2	COL	NL	19	19	24	4	7	1	0	2	11	0	0	7	8	0	0	0	0	0	NA	0.29167	0.45167	4	0.58333	OAK	AL	40000000

	playerID	yearID	stint	teamID.x	lgID.x	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old	BA	OBP	X1B	SLG	teamID.y	lgID.y	salary
7	01	9																							667	129		333			0
7888	giambja01	2010	1	COL	NL	87	87	176	17	43	9	0	6	35	2	0	35	47	5	6	0	5	5	NA	0.24431822	0.37837843	28	0.39777273	COL	NL	17500000
7889	giambja01	2011	1	COL	NL	64	64	131	20	34	6	0	13	32	0	0	17	45	0	3	0	1	1	64	0.25595542	0.35526322	15	0.60305334	COL	NL	10000000
7890	giambja01	2012	1	COL	NL	60	NA	89	7	20	4	0	1	8	0	0	20	24	2	2	0	2	4	NA	0.22471911	0.37168144	15	0.30333708	COL	NL	10000000
7891	giambja01	2013	1	CLE	AL	71	71	186	21	34	8	0	9	31	0	1	23	56	0	4	0	3	8	NA	0.18279577	0.28240744	17	0.37096767	CLE	AL	7500000
2011 12	saenzol01	1999	1	OAK	AL	97	97	255	41	70	18	0	11	41	1	1	22	47	1	15	0	3	6	97	0.27450908	0.36271119	41	0.47450908	OAK	AL	2400000
2011 13	saenzol01	2000	1	OAK	AL	76	76	214	40	67	12	2	9	33	1	0	25	40	2	7	0	1	6	76	0.31308841	0.40080907	44	0.51401187	OAK	AL	2600000

	playerID	yearID	stint	teamID.x	lgID.x	G	G_batting	AB	R	H	X2B	X3B	HR	RBI	SB	CS	BB	SO	IBB	HBP	SH	SF	GIDP	G_old	BA	OBP	X1B	SLG	teamID.y	lgID.y	salary
2011014	saenzol01	2001	1	OAK	AL	106	106	305	33	67	21	1	9	32	0	1	19	64	1	13	1	3	9	106	0.2196721	0.2911765	36	0.3836066	OAK	AL	2900000
2011015	saenzol01	2002	1	OAK	AL	68	68	156	15	43	10	1	6	18	1	1	13	31	1	7	0	2	2	68	0.2756541	0.3539326	26	0.4679487	OAK	AL	8000000
2011016	saenzol01	2005	1	LAN	NL	109	109	319	39	84	24	0	15	63	0	1	27	63	1	3	0	2	12	109	0.2633229	0.3247863	45	0.4796238	LAN	NL	6500000
2011017	saenzol01	2006	1	LAN	NL	103	103	179	30	53	15	0	11	48	0	0	14	47	1	7	0	4	4	103	0.2960894	0.3627451	27	0.5642458	LAN	NL	10000000
2011018	saenzol01	2007	1	LAN	NL	92	92	110	9	21	5	0	4	18	0	0	16	25	0	2	0	4	5	92	0.1909091	0.2954545	12	0.3454545	LAN	NL	10000000

Since all these players were lost in after 2001 in the offseason, let's only concern ourselves with the data from 2001.

Use subset again to only grab the rows where the yearID was 2001.

In [20]:

```
lost_players <- subset(lost_players,yearID == 2001)
```

Reduce the lost_players data frame to the following columns: playerID,H,X2B,X3B,HR,OBP,SLG,BA,AB

In [21]:

```
lost_players <-
lost_players[,c('playerID','H','X2B','X3B','HR','OBP','SLG','BA','AB'
)]
```

In [22]:

```
head(lost_players)
```

Out[22]:

	playerID	H	X2B	X3B	HR	OBP	SLG	BA	AB
5141	damonjo01	165	34	4	9	0.3235294	0.363354	0.2562112	644
7878	giambja01	178	47	2	38	0.4769001	0.6596154	0.3423077	520
20114	saenzol01	67	21	1	9	0.2911765	0.3836066	0.2196721	305

Replacement Players

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.
- Their mean OBP had to equal to or greater than the mean OBP of the lost players

Use the combo dataframe you previously created as the source of information! Remember to just use the 2001 subset of that dataframe. There's lost of different ways you can do this, so be creative! It should be relatively simple to find 3 players that satisfy the requirements, note that there are many correct combinations available!

Helpful info on sorting data frames

Example Solution

Note: There are lots of correct answers and ways to solve this!

First only grab available players from year 2001

In [38]:

```
library(dplyr)
```

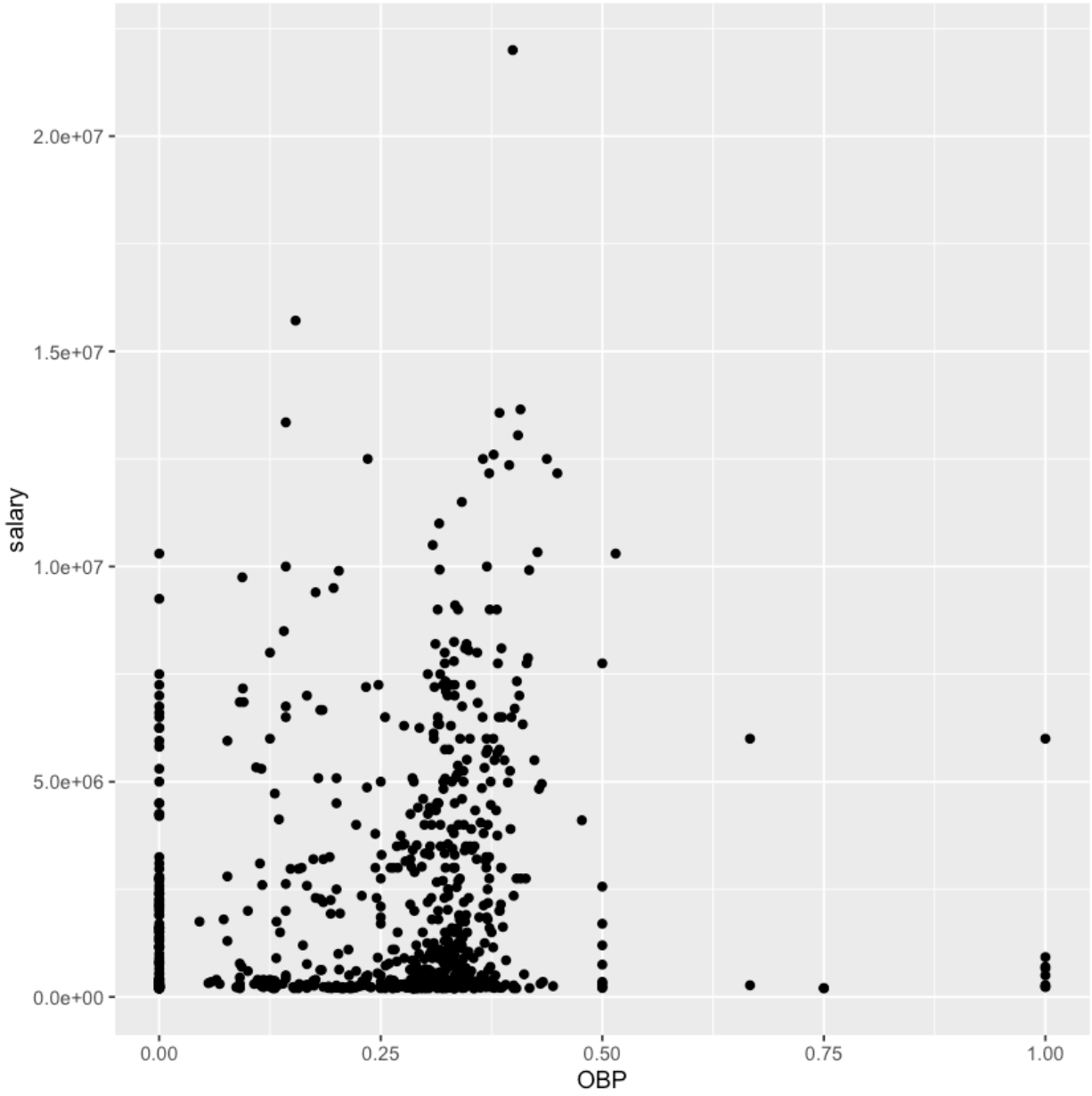
```
avail.players <- filter(combo,yearID==2001)
```

Then I made a quick plot to see where I should cut-off for salary in respect to OBP:

In [39]:

```
library(ggplot2)
ggplot(avail.players,aes(x=OBP,y=salary)) + geom_point()
```

```
Warning message:
: Removed 168 rows containing missing values (geom_point).
```



Looks like there is no point in paying above 8 million or so (I'm just eyeballing this number). I'll choose that as a cutt off point. There are also a lot of players with OBP==0. Let's get rid of them too.

In [41]:

```
avail.players <- filter(avail.players,salary<8000000,OBP>0)
```

The total AB of the lost players is 1469. This is about 1500, meaning I should probably cut off my avail.players at 1500/3= 500 AB.

In [42]:

```
avail.players <- filter(avail.players,AB >= 500)
```

Now let's sort by OBP and see what we've got!

In [44]:

```
possible <- head(arrange(avail.players,desc(OBP)),10)
```

Grab columns I'm interested in:

In [45]:

```
possible <- possible[,c('playerID','OBP','AB','salary')]
```

In [46]:

```
possible
```

Out[46]:

	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	edmonji01	0.4102142	500	6333333
8	gilesbr02	0.4035608	576	7333333
9	pujolal01	0.402963	590	200000
10	olerujo01	0.4011799	572	6700000

Can't choose giambja again, but the other ones look good (2-4). I choose them!

In [47]:

```
possible[2:4,]
```

Out[47]:

	playerID	OBP	AB	salary

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

AMAN SINGH

PRIME -ROG

https://github.com/prime-rog/Capstone-Project.git