DATA VISUALIZATION

MoneyBall Project

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).

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.

Code

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

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

# print the first 6 rows to check the data

print(head(batting))


# str function to check the structure of dataframe

str(batting)
```

Feature Engineering

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

1: Batting Average

2: On Base Percentage

3: Slugging Percentage

The formulae can be seen from the internet.

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

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:

```
#find BA by formulae BA=H/AB

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

print(tail(batting$BA))
```

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 the data frame. However it 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

```
#on base percentage

#  H+BB+HBP / AB+BB+HBP+SF
```

```
# Hits+ basesonballs+ hitbypitch / atballs+ sacrifisefiles+ BB+ HBP

batting$OBP <- (batting$H + batting$BB + batting$HBP)/(batting$AB + batting$BB + batting$HBP + batting$SF)

batting$X1B <- batting$H - batting$X2B - batting$X3B - batting$HR

batting$SLG <- ((1*batting$X1B)+(2*batting$X2B)+(3*batting$X3B)+(4*batting$HR))/batting$AB

#doing str(batting) shows that we have sucessfully added 4 new variables
```

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'.

```
# now we will merge the sal and batting data

# get sal data into a data frame

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
```

```
# merge batting data with year > 1985
batting <- subset(batting, yearID >= 1985)
#join them by a vector playerID, yearID
#on checking summary the salary data is added
combo <- merge(batting,sal,by=c('playerID','yearID'))
summary(combo)
```

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').

```
#analyzing the lost players
#create a subset withe 3 player id
lost_players <- subset(combo,playerID %in%
c('giambja01','damonjo01','saenzol01'))
print(lost_players)
# the list of all the players will be printed, it is not shown
here due to the size.
# we need players from the year 2001, so we need to filter
our data
#get the data for yead 2001
```

```
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

print(head(lost_players))
```

Replacement Players

Now we have all the information we need!

The final task is to Find Replacement Players for the key three players we lost!

However, we have three constraints:

1: The total combined salary of the three players can not exceed 15 million dollars.

2: Their combined number of At Bats (AB) needs to be equal to or greater than the lost players.

3: Their mean OBP had to equal to or greater than the mean OBP of the lost players

We will use the combo dataframe previously created as the source of information! and will do this by creating a plot

```
# install the dplyr library

install.packages("dplyr")

# calling the library

library(dplyr)

# data frame avail.players

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

```
# plot a graph to see where to cut-off for salary in respect to OBP

# install and call the ggplot2 library

install.packages("ggplot2")

library(ggplot2)

# set x axis as OBP and y as salary

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


# Looks like there is no point in paying above 8 million.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.

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.

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

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

# Store it in a df

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

# getting the columns I am interested in

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

print(possible)


# we can not choose giamja01 again as we already lost him

# from the data it is clear that 2, 3 and 4 look good
```

```
# they follow all of our cinstrainsts.
# I choose them
print(possible[2:4,])
```

Great, looks like I just saved the 2001 Oakland A's a lot of money!

If only I could go back in time to do it I could have made a lot of money in 2001 picking players and later asked Brad Pitt to play me inthe MoneyBall movie.

Source: wikipedia/udemy