Data Acquisition, Preparation and EDA

Introduction and Objectives

- **Data Science** connects statistics, computer science, and domain knowledge.
- We look for patterns & reasons for differences/changes in datasets.

Introduction and Objectives

- Datasets
  - design a study
    - lay out goals
    - apply domain knowledge (what factors are important?)
    - generate variable list
    - gather data (experiments, surveys, other studies)
    - account for study cost and feasibility
  - OR analyze existing data

Introduction and Objectives

- Once we have the data, we proceed to extract useful information
- BUT we must understand the data first
- In this lecture:
  - basic data acquisition/preparation
  - understand the nature of the data via **exploratory data analysis (EDA)**
  - explore plausible variable relationships
- *We defer formal modeling for later*

Introduction and Objectives

- Data mining tools:
  - expanding dramatically in the past 20 yrs
- R
  - popular among data scientists & in academia
  - open-source
  - most SOTA methods have R package implementations

Introduction and Objectives

- The number of studies is soaring (especially during COVID)
- BUT a significant number cannot be reproduced/replicated
  - this phenomenon is an ongoing crisis termed the **replicability crisis**
- In an effort to produce trustworthy and reproducible results, we use **R Markdown**.
  - It achieves many goals:
    1) Anyone can rerun our study to replicate the results.
    2) We can run our data analysis & produce reports at the same time.
- Communication between us and readers/decision-makers is essential.

Objectives

- This module will focus on **data preparation**, **data cleaning**, and **exploratory data analysis (EDA)**.
- R and R Markdown will be used.
  See advanced_R_tutorial.Rmd for extremely useful EDA tools such as `dplyr`, `ggplot`, `data.table`, and more.

Contents

0. Suggested extra readings/doing:
   - run and study Get_staRted.Rmd
   - run and study advanced_R_tutorial.Rmd and advanced_R_tutorial.html
   - read `50 years of data science` and `Teaching data science` available in Canvas. (skip this)
   - Data set: `MLPayData_Total.csv`
1. Case Study: Billion dollar Billy Beane
2. Study flow:
   - Study design
   - Gathering data
   - Process data (tidy data)
   - Exploratory Data Analysis (EDA)
   - Conclusion/Challenges
3. R functions
   - basic r functions
   - dplyr
   - ggplot

Handy Cheat Sheets

DPLYR Cheat Sheet:
http://www.rstudio.com/wp-content/uploads/2015/02/data-wrangling-cheatsheet.pdf

ggplot Cheat Sheet:
https://github.com/rstudio/cheatsheets/blob/master/data-visualization-2.1.pdf

1. Introduction and Objectives

2. Case Study: Baseball
   - Data Preparation

3. Exploratory Data Analysis (EDA)
   - Part I: Analyze aggregated variables
   - Part II: Analyze pay and winning percent over time and by team

4. Conclusions and Discussion

5. Appendix: Sample Statistics

Case Study: Baseball

Background

- Baseball is one of the most popular sports in the US.
- Major League Baseball (MLB) includes the highest level of baseball teams
  - 30 total teams including the American League and the National League
  - top teams include the New York Yankees, Boston Red Sox, Philadelphia Phillies, and Oakland Athletics
- Oakland A's
  - low budget team
  - recent rising star
  - General Manager (GM) Billy Beane
    is well known to apply statistics in his coaching

Case Study: Baseball

Background

- In the article Billion Dollar Billy Beane:
  - author: Benjamin Morris
  - studies a regression of performance vs. total payroll from all 30 teams over a 17 yr period
  - Oakland A's performance is comparable to the Boston Red Sox
  - therefore argues that Billy Beane is worth $12 million over 5 years
- Article link: https://fivethirtyeight.com/features/billion-dollar-billy-beane/

Case Study: Baseball

Objectives

1. Reproduce Benjamin's study.
   - Is Billy Beane (Oakland A's GM) worth 12.5 million dollars for a period of 5 years, as argued in the article?
   - We challenge Benjamin's reasoning behind his argument.
2. Explore general questions:
   - How does pay and performance relate to each other?
   - Will a team perform better when they are paid more?

Gathering Data

- We gathered information on payroll and performance by team from 1998 to 2014 and reassembled the data set from several websites.
  - (We could have easily *reproduced* all the analyses from the post if the article's data was available!)
- Some manual corrections were made.
  - i.e. consolidating team names for teams that underwent a name change.
- Example source: http://www.stevetheump.com/Payrolls.htm

Gathering Data

Data: `MLPayData_Total.csv`, consists of winning records and the payroll of all 30 ML teams from 1998 to 2014 (17 years). There are 162 games in each season.

The variables included are:

- `team name`: team names
- `p2014`: total pay in 2014 in **millions** and other years indicated by year
- `X2014`: number of games won in 2014 and other years labeled
- `X2014.pct`: percent winning in 2014 and other years (We only need one of the two variables from above.)

1. Introduction and Objectives

2. Case Study: Baseball
   - Data Preparation

3. Exploratory Data Analysis (EDA)
   - Part I: Analyze aggregated variables
   - Part II: Analyze pay and winning percent over time and by team

4. Conclusions and Discussion

5. Appendix: Sample Statistics

Data Preparation

Before we do any analysis, it is a **MUST** that we take a look at the data.

In particular, we will try to:

Tidy the data:

- Import data/Data preparation
- Data format
- Missing values/peculiarity
- Understand the variables: unit, format, unusual values, etc.
- Put data into a standard data format

Columns: variables
Rows: subjects

Data Preparation

Read the data:

The simplest way to import data is to use **`read.csv()`**
(or the faster `fread()` from the `data.table` package).

What is the current working directory?
R will find files or save files to the working directory.

```
getwd()
dir <- "/Users/lzhao/Dropbox/STAT471/Data"  # my laptop
setwd(dir)   #same as  setwd("/Users/lzhao/Dropbox/STAT471/Data")
getwd()
```

Alternatively, if the data is in the same folder as the .Rmd file then we can read data directly.

```
datapay <- read.csv("data/MLPayData_Total.csv", header=T, stringsAs
```

Data Preparation

What is in the dataset?

Take a quick look at the data. Pay attention to what is in the data, any missing values, and the variable format.

```
names(datapay)
```

```
##  [1] "Team.name.2014" "p1998"          "p1999"          "p2000"
##  [5] "p2001"          "p2002"          "p2003"          "p2004"
##  [9] "p2005"          "p2006"          "p2007"          "p2008"
## [13] "p2009"          "p2010"          "p2011"          "p2012"
## [17] "p2013"          "p2014"          "X2014"          "X2013"
## [21] "X2012"          "X2011"          "X2010"          "X2009"
## [25] "X2008"          "X2007"          "X2006"          "X2005"
## [29] "X2004"          "X2003"          "X2002"          "X2001"
## [33] "X2000"          "X1999"          "X1998"          "X2014.pct"
## [37] "X2013.pct"      "X2012.pct"      "X2011.pct"      "X2010.pct"
## [41] "X2009.pct"      "X2008.pct"      "X2007.pct"      "X2006.pct"
## [45] "X2005.pct"      "X2004.pct"      "X2003.pct"      "X2002.pct"
## [49] "X2001.pct"      "X2000.pct"      "X1999.pct"      "X1998.pct"
```

Is anything bothering you? We may want to change names of teams to a shorter, neater name.

Data Preparation

Everything seems to be OK at the moment other than changing one variable name.

```
#summary(datapay)
summary(datapay)[1:10] # quick summary. missing values may be shown
```

```
##  [1] "Length:30          " "Class :character  " "Mode  :character  "
##  [4] NA                    NA                    NA
##  [7] "Min.   : 8.3  "      "1st Qu.:27.7  "      "Median :43.9  "
## [10] "Mean   :41.1  "
```

```
str(datapay) # data structure
```

```
## 'data.frame':    30 obs. of  52 variables:
##  $ Team.name.2014: chr  "Arizona Diamondbacks" "Atlanta Braves" "Baltimore Orioles" "Boston Red Sox" ...
##  $ p1998         : num  31.6 61.7 71.9 59.5 49.8 ...
##  $ p1999         : num  70.5 74.9 72.2 71.7 42.1 ...
##  $ p2000         : num  81 84.5 81.4 77.9 60.5 ...
##  $ p2001         : num  81.2 91.9 72.4 109.6 64 ...
##  $ p2002         : num  102.8 93.5 60.5 108.4 75.7 ...
##  $ p2003         : num  80.6 106.2 73.9 99.9 79.9 ...
##  $ p2004         : num  70.2 88.5 51.2 125.2 91.1 ...
##  $ p2005         : num  63 85.1 74.6 121.3 87.2 ...
##  $ p2006         : num  59.7 90.2 72.6 120.1 94.4 ...
##  $ p2007         : num  52.1 87.3 93.6 143 99.7 ...
##  $ p2008         : num  66.2 102.4 67.2 133.4 118.3 ...
##  $ p2009         : num  73.6 96.7 67.1 122.7 135.1 ...
##  $ p2010         : num  60.7 84.4 81.6 162.7 146.9 ...
##  $ p2011         : num  53.6 87 85.3 161.4 125.5 ...
##  $ p2012         : num  74.3 83.3 81.4 173.2 88.2 ...
##  $ p2013         : num  89.1 89.8 91 150.7 104.3 ...
##  $ p2014         : num  113 111 107 163 89 ...
##  $ X2014         : int  64 79 96 71 73 73 76 85 66 90 ...
```

Data Preparation

Let's update the team name variable.

```
# change variable name and also update the data file
datapay <- datapay %>% rename(team = Team.name.2014)
names(datapay)[1:5] # only show 5 names
```

```
## [1] "team"  "p1998" "p1999" "p2000" "p2001"
```

Data Preparation: Reshape the data

The original format of the dataset `MLPayData_Total.csv` is not in a desirable format. Each row lists multiple results. Also the variable `year` is missing.

```
datapay[1:4, 1:5]  # list a few lines (subsetting)
```

```
##                  team p1998 p1999 p2000 p2001
## 1 Arizona Diamondbacks  31.6  70.5  81.0  81.2
## 2       Atlanta Braves  61.7  74.9  84.5  91.9
## 3    Baltimore Orioles  71.9  72.2  81.4  72.4
## 4       Boston Red Sox  59.5  71.7  77.9 109.6
```

```
#datapay$team # get variables
```

Data Preparation: Reshape the data

We would like to reshape the data into the following table format:

- columns (variables) contain all variables
- each row records one result(s)

In our case we have four variables: team, year, pay, win_number and win_percentage. Let's rearrange the data into the following form:

team | year | payroll | win_number | win_percentage

Data Preparation: Reshape the data

Let us do this using `dplyr::pivot_longer()`.
First we create the `payroll` and `year` variables:

```
payroll <- datapay %>%   # first create variable: payroll and year
  select(team, p1998:p2014) %>%
  pivot_longer(cols = starts_with("p"),
               names_to = "year",
               names_prefix = "p",
               values_to = "payroll")
payroll[1:3, 1:3] # show a few rows
```

Data Preparation: Reshape the data

Let's create the other variables.

```
win_num <- datapay %>%  # create variable: win_num and year
  select(team, X1998:X2014) %>%
  pivot_longer(cols = X1998:X2014,
               names_to = "year",
               names_prefix = "X",
               values_to = "win_num")

win_pct <- datapay %>%  # create variable: win_pct and year
  select(team, X1998.pct:X2014.pct) %>%
  pivot_longer(cols = X1998.pct:X2014.pct,
               names_to = "year",
               names_prefix = "X",
               values_to = "win_pct") %>%
  mutate(year = substr(year, 1, 4))
```

Data Preparation: Reshape the data

Finally, we join the tables into team, year, payroll, win_num, and win_pct.

```
datapay_long <- payroll %>%
  inner_join(win_num, by = c("team", "year")) %>%
  inner_join(win_pct, by = c("team", "year"))
head(datapay_long, 2)  # see first 2 rows
```

```
## # A tibble: 2 x 5
##   team                 year  payroll win_num win_pct
##   <chr>                <chr>   <dbl>   <int>   <dbl>
## 1 Arizona Diamondbacks 1998     31.6      65   0.401
## 2 Arizona Diamondbacks 1999     70.5     100   0.617
```

Data Preparation: Reshape the data

Take a quick look at the newly formed data file `datapay_long`.

```
names(datapay_long) #names(datapay) new vs. old data files
```

```
## [1] "team"    "year"    "payroll" "win_num" "win_pct"
```

Quick summary of the new data:

```
head(datapay_long) # shows the first 6 rows
```

```
## # A tibble: 6 x 5
##   team                 year  payroll win_num win_pct
##   <chr>                <chr>   <dbl>   <int>   <dbl>
## 1 Arizona Diamondbacks 1998     31.6      65   0.401
## 2 Arizona Diamondbacks 1999     70.5     100   0.617
## 3 Arizona Diamondbacks 2000     81.0      85   0.525
## 4 Arizona Diamondbacks 2001     81.2      92   0.568
## 5 Arizona Diamondbacks 2002    103.       98   0.605
## 6 Arizona Diamondbacks 2003     80.6      84   0.519
```

More ways to summarize the data:

```
# dim(datapay_long)
# str(datapay_long)
# summary(datapay_long)
# skimr::skim(datapay_long)
```

Data Preparation: Output the cleaned data file

After processing, save this cleaned data file into a new table called `baseball.csv`. Let's output this table to the /data folder in our working folder. From now on we will only use the data file `baseball`.

```
write.csv(datapay_long, "data/baseball.csv", row.names = F)
```

Remark

The above data prep process should be put into a separate .r or .rmd file. There is no need to rerun the above data prep portion each time we work on the project. We put the whole project into one file for the purpose of demonstration.

Exploratory Data Analysis (EDA)

- All the analyses done in this lecture will be exploratory.
- The goal is to see what information we might be able to extract so that it will support the goal of our study. This is an extremely important first step of the data analyses.
- We try to understand the data, summarize the data, then finally explore the relationships among the variables through useful visualization.

Part I: Analyze aggregated variables

In this section:

- Try to use aggregated information such as:
  - the total pay for each team
  - average performance
- Look for the relationship between performance and the payroll as suggested in Morris's post.

Input the data

First, input the clean data `baseball` and quickly explore the data. Everything seems fine: no missing values, names of variables are good. The class of each variable matches its nature. (numeric, factor, characters...)

```
baseball <- read.csv("data/baseball.csv", header = TRUE, stringsAsFactors = F)
names(baseball)
str(baseball)
summary(baseball)
#View(baseball)
```

```
## [1] "team"    "year"    "payroll" "win_num" "win_pct"
## 'data.frame':    510 obs. of  5 variables:
##  $ team   : chr  "Arizona Diamondbacks" "Arizona Diamondbacks" "Arizona Diamondbacks" "Arizona Diamondbacks"
##  $ year   : int  1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 ...
##  $ payroll: num  31.6 70.5 81 81.2 102.8 ...
##  $ win_num: int  65 100 85 92 98 84 51 77 76 90 ...
##  $ win_pct: num  0.401 0.617 0.525 0.568 0.605 ...
##      team                year         payroll          win_num
##  Length:510         Min.   :1998   Min.   :  8.3   Min.   : 43
##  Class :character   1st Qu.:2002   1st Qu.: 51.3   1st Qu.: 72
##  Mode  :character   Median :2006   Median : 73.3   Median : 81
##                     Mean   :2006   Mean   : 78.1   Mean   : 81
##                     3rd Qu.:2010   3rd Qu.: 95.0   3rd Qu.: 90
##                     Max.   :2014   Max.   :235.3   Max.   :116
##     win_pct
##  Min.   :0.265
##  1st Qu.:0.444
##  Median :0.500
##  Mean   :0.500
##  3rd Qu.:0.556
##  Max.   :0.716
```

Create a new data table

For convenience, we create a new table which only contains the total payroll and average winning percentage for each team. We name them `team`, `payroll_total`, and `win_pct_ave`. We will change the unit of `payroll_total` from million to billion.

```
# create total and average winning percentage for each team
data_agg <-baseball %>%
  group_by(team) %>%
  summarise(
    payroll_total = sum(payroll)/1000,
    win_pct_ave = mean(win_pct))
str(data_agg)
summary(data_agg)
```

```
## tibble [30 x 3] (S3: tbl_df/tbl/data.frame)
##  $ team         : chr [1:30] "Arizona Diamondbacks" "Atlanta Braves" "Baltimore Orioles" "Boston Red Sox" ..
##  $ payroll_total: num [1:30] 1.22 1.52 1.31 2.1 1.55 ...
##  $ win_pct_ave  : num [1:30] 0.492 0.563 0.459 0.553 0.476 ...
##      team           payroll_total    win_pct_ave
##  Length:30          Min.   :0.698   Min.   :0.433
##  Class :character   1st Qu.:1.022   1st Qu.:0.473
##  Mode  :character   Median :1.264   Median :0.492
##                     Mean   :1.328   Mean   :0.500
##                     3rd Qu.:1.517   3rd Qu.:0.526
##                     Max.   :2.857   Max.   :0.594
```

Descriptive statistics

To summarize a continuous variable (such as `payroll_total` or `win_pct_ave`), we use the following measurements:

- **Center**: sample mean/median
- **Spread**: sample standard deviation
- **Range**: minimum and maximum
- **Distribution**: quantiles

Descriptive statistics

First, let us take a look at `payroll_total`.

Base `R` way:

```
mean(data_agg$payroll_total)
sd(data_agg$payroll_total)
quantile(data_agg$payroll_total, prob = seq(0, 1, 0.25))
median(data_agg$payroll_total)
max(data_agg$payroll_total)
min(data_agg$payroll_total)
summary(data_agg$payroll_total)
```

```
## [1] 1.33
## [1] 0.45
##    0%   25%   50%   75%  100%
## 0.698 1.022 1.264 1.517 2.857
## [1] 1.26
## [1] 2.86
## [1] 0.698
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.698   1.022   1.264   1.328   1.517   2.857
```

Descriptive statistics

First, let us take a look at `payroll_total`.

dplyr way:

```
data_agg %>% select(payroll_total) %>%
  summarise(
    mean = mean(payroll_total),
    sd   = sd(payroll_total),
    max = max(payroll_total),
    min = min(payroll_total),
    "0%" = quantile(payroll_total)[1],
    "25%" = quantile(payroll_total)[2],
    "50%" = quantile(payroll_total)[3],
    "75%" = quantile(payroll_total)[4],
    "100%" = quantile(payroll_total)[5]
  )
```

```
## # A tibble: 1 x 9
##    mean    sd   max   min  `0%` `25%` `50%` `75%` `100%`
##   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>
## 1  1.33 0.450  2.86 0.698 0.698  1.02  1.26  1.52   2.86
```

Descriptive statistics

Find the team with the max/min payroll.

Base `R` way:

```
data_agg$team[which.max(data_agg$payroll_total)]
```

```
## [1] "New York Yankees"
```

```
data_agg$team[which.min(data_agg$payroll_total)]
```

```
## [1] "Miami Marlins"
```

Descriptive statistics

Rearrange the data to see the ranks of team by `payroll`.

But we can easily rearrange the whole data set `data_agg` by ordering one variable, say `payroll_total`.

Base `R` way:

```
#To rank teams by payroll in decreasing order
arrange(data_agg, desc(payroll_total))[1:5,] #default decs=T
```

```
## # A tibble: 5 x 3
##   team                  payroll_total win_pct_ave
##   <chr>                         <dbl>       <dbl>
## 1 New York Yankees               2.86       0.594
## 2 Boston Red Sox                 2.10       0.553
## 3 Los Angeles Dodgers            1.87       0.529
## 4 New York Mets                  1.72       0.502
## 5 Philadelphia Phillies          1.69       0.519
```

```
#arrange(data_agg, win_pct_ave) # default???
#arrange(data_agg, -desc(payroll_total))[1:5,]
```

Descriptive statistics

Rearrange the data to see the ranks of team by `payroll`.

`dplyr` way:

```
data_agg %>% select(team,payroll_total) %>% filter(payroll_total == max(payroll_total))
data_agg %>% select(team,payroll_total) %>% filter(payroll_total == min(payroll_total))
```

```
## # A tibble: 1 x 2
##   team             payroll_total
##   <chr>                    <dbl>
## 1 New York Yankees          2.86
## # A tibble: 1 x 2
##   team          payroll_total
##   <chr>                 <dbl>
## 1 Miami Marlins         0.698
```

`dplyr` way:

```
data_agg %>%
  arrange(payroll_total) %>%
  slice(1:5)  # select first 5 rows
```

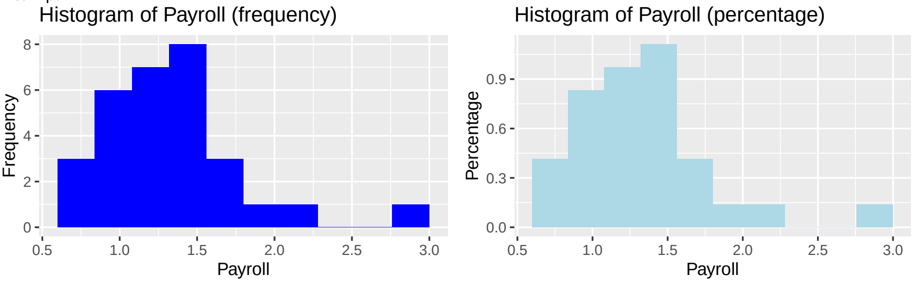
```
## # A tibble: 5 x 3
##   team                payroll_total win_pct_ave
##   <chr>                       <dbl>       <dbl>
## 1 Miami Marlins               0.698       0.468
## 2 Pittsburgh Pirates          0.772       0.443
## 3 Tampa Bay Rays              0.776       0.462
## 4 Kansas City Royals          0.870       0.433
## 5 Oakland Athletics           0.888       0.539
```

Displaying variables: Histograms

`ggplot` plots:

```
p1 <- ggplot(data_agg) +
  geom_histogram(aes(x = payroll_total), bins = 10, fill = "blue") +
  labs( title = "Histogram of Payroll (frequency)", x = "Payroll" , y = "Frequency")

p2 <- ggplot(data_agg) +
  geom_histogram(aes(x = payroll_total, y = ..density..), bins = 10, fill = "light blue") +
  labs( title = "Histogram of Payroll (percentage)", x = "Payroll" , y = "Percentage")

grid.arrange(p1, p2, ncol = 2) # facet the two plots  side by side
```

hist-1.pdf



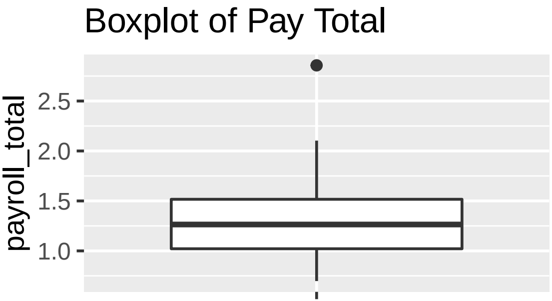
Notice, the two plots above look identical but with different `y-scale`.

Displaying variables: Boxplots

A **boxplot** captures the spread by showing median, quantiles and outliers:

`ggplot` plots:

```
ggplot(data_agg) +
  geom_boxplot(aes(x="", y=payroll_total)) +
  labs(title="Boxplot of Pay Total", x="")
```



Normal variables

When would the sample mean and sample standard deviation help us to describe the distribution of a variable? As an exercise, let us summarize the variable `win_pct_ave`.

```
mean(data_agg$win_pct_ave) # sort(data_agg$win_pct_ave)
sd(data_agg$win_pct_ave)
```

```
## [1] 0.5
## [1] 0.0376
```

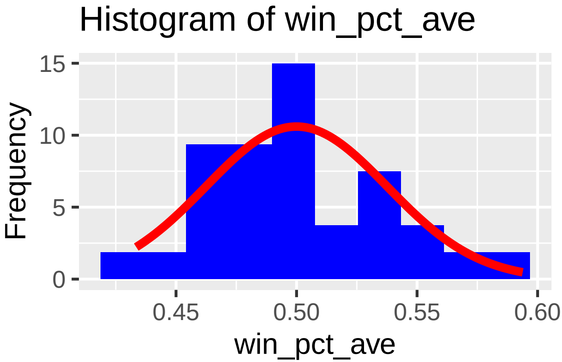
We see that `win` on average is 0.5 with a SD being 0.038. How would the mean and sd be useful in describing the distribution of `win`? **Only if the histogram looks like a bell curve**!

Normal variables

Take a look at the histogram of `win`. Here we impose a **normal curve** with the center being 0.5 and the spread, sd $= 0.038$.

`ggplot` way:

```
ggplot(data_agg) +
  geom_histogram(aes(x=win_pct_ave, y = ..density..), bins=10, fill= "blue" ) +
  stat_function(fun = dnorm, args = list(mean = mean(data_agg$win_pct_ave),
                                         sd = sd(data_agg$win_pct_ave)), colour = "red",
  labs( title = "Histogram of win_pct_ave", x = "win_pct_ave" , y = "Frequency")
```



w normal-1.pdf

Normal variables

The smoothed normal curve captures the shape of the histogram of win. Or we will say that the variable `win` follows a normal distribution approximately. Then we can describe the distribution of `win` using the two numbers: mean and sd.

Roughly speaking:

- 68% of teams with `win` to be within one sd from the mean.

$$0.5 \pm 0.038 = [0.462, 0.538]$$

- 95% of the teams with `win` to be within 2 sd from the mean:

$$0.5 \pm 2 * 0.038 = [0.425, 0.575]$$

- 2.5% of the teams with `win` to be higher 2.5 times of sd above the mean:

$$> 0.5 + 2 * 0.038 = 0.575$$

Explore variable relationships

- **Scatter plots** show the relationship between x variable `payroll_total` and y variable `win_pct_ave`.
- We are looking for patterns between the two variables, such as a linear or quadratic relationship.

Explore variable relationships: Scatter plots

`ggplot` plots (shown on next slide)

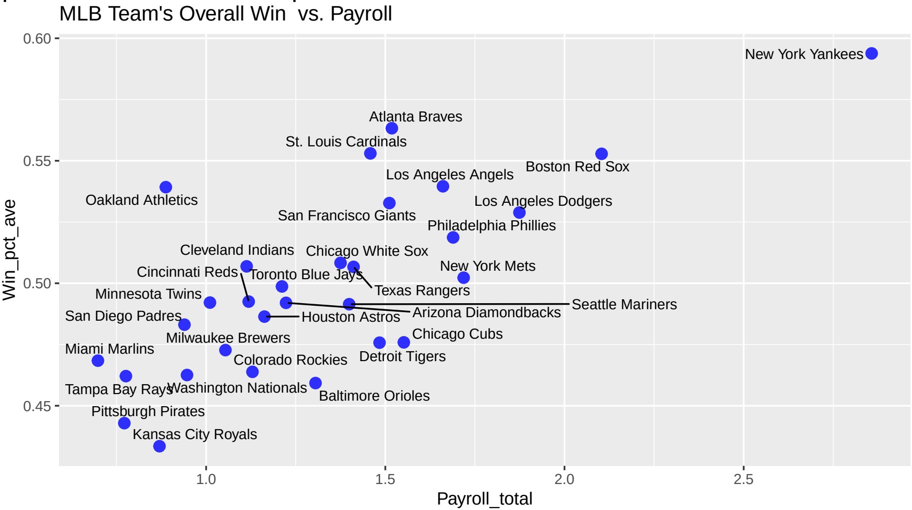
```
data_agg %>%
  ggplot(aes(x = payroll_total, y = win_pct_ave)) +
  # geometric options: color, size, shape, alpha: transparency (range: 0 to 1)
  geom_point(color = "blue", size= 3, alpha = .8) +
  geom_text_repel(aes(label = team), size = 3) +
  labs(title = "MLB Team's Overall Win  vs. Payroll",
       x = "Payroll_total",
       y = "Win_pct_ave")
```

Explore variable relationships: Scatter plots

We notice the positive association:

when `payroll_total` increases, so does `win_pct_ave`.

p with team names 2-1.pdf



Explore variable relationships: Scatter plots

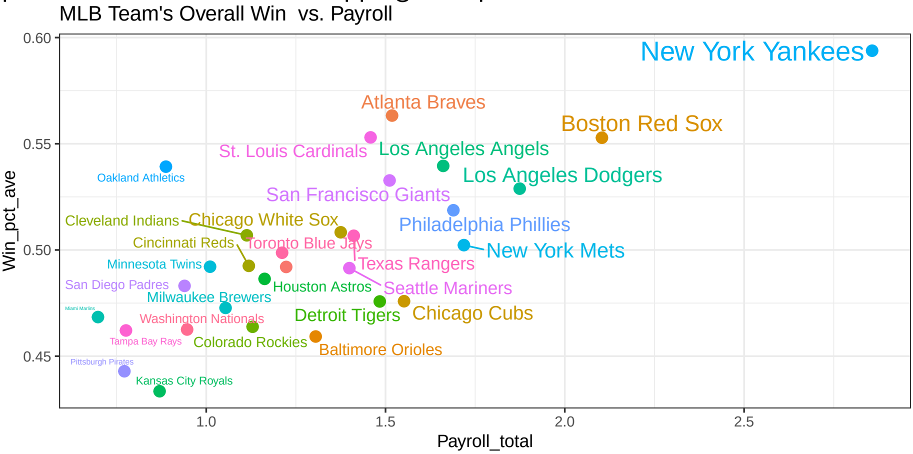
We can bring in other variables to adjust the color, size, and alpha of the scatter plot via **aesthetic mapping**.

```
data_agg %>%
  ggplot(aes(x = payroll_total, y = win_pct_ave)) +
  # geometric options with aes mapping:
  # color, size, alpha as a function of a variable
  geom_point(aes(color = team), size = 3) +
  geom_text_repel(aes(color = team, label = team, size = payroll_total)) +
  labs(title = "MLB Team's Overall Win  vs. Payroll",
       x = "Payroll_total",
       y = "Win_pct_ave") +
  theme_bw() +
  theme(legend.position = "none")
```

Explore variable relationships: Scatter plots

We can bring in other variables to adjust the color, size, and alpha of the scatter plot via **aesthetic mapping**.

p with team names with mappings 2-1.pdf



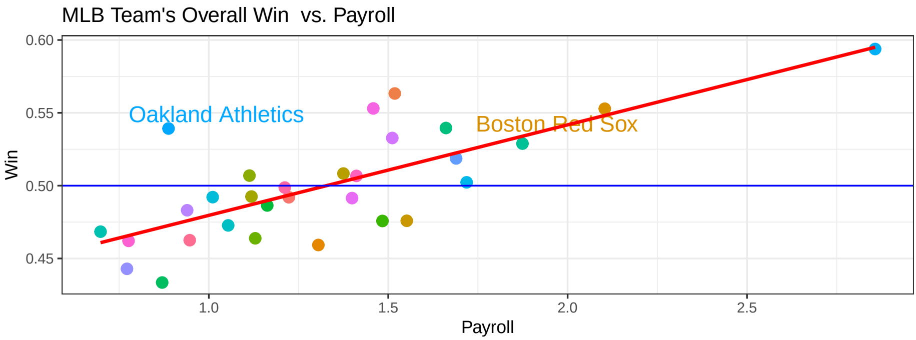
Explore variable relationships

Least Squared Lines

- The simplest function to capture the relationship between pay and performance is through the linear model.
- We impose the least squared equation on top of the scatter plot using `ggplot()` with `geom_smooth()`.
- We also annotate the two teams `Oakland Athletics` and `Boston Red Sox`.

Explore variable relationships: Least squared lines

```
selected_teams <- c("Oakland Athletics", "Boston Red Sox")

data_agg %>%
  ggplot(aes(x = payroll_total, y = win_pct_ave)) +
  geom_point(aes(color = team), size = 3) +
  # only show names of selected_teams
  geom_text_repel(data = subset(data_agg, team %in% selected_teams),
                  aes(label = team, color = team), size = 5) +
  geom_smooth(method = "lm", formula = y ~ x, se = F,color = "red") +
  geom_hline(aes(yintercept = mean(win_pct_ave)), color = "blue") +
  labs(title = "MLB Team's Overall Win  vs. Payroll", x = "Payroll", y = "Win") +
  theme_bw() +
  theme(legend.position = "none")
```



Conclusions/Discussions

Answer to Question 1:

HERE is how the article concludes that Beane is worth as much as the GM in Red Sox. By looking at the above plot, Oakland A's win pct is more or less the same as that of Red Sox, so based on the LS equation, the team should have paid 2 billion!

Do you agree with this argument? Why or why not?

Answer to Question 2:

From this regression line, we see a clear upward trend. Or precisely the least squared equation has a positive coefficient. Consequently, the more a team is paid the better performance we expect the team has.

Conclusions/Discussions

Questions for you:

- Do you agree with the conclusions made based on a regression analysis shown above?
- How would you carry out a study which may have done a better job? In what way?

Part II: Analyze pay and winning percent over time and by team

- Payroll and performance varies depending on teams and years.
- We investigate changes over time and by teams to see how payroll relates to performance.

Compare payroll and performance

We can compare summary statistics of payrolls and performance among teams.

```
baseball %>% group_by(team) %>%
  summarise(payroll_mean = mean(payroll),
            win_pct_mean = mean(win_pct)) %>%
  arrange(-payroll_mean) %>%
  slice(1:5)
```

```
## # A tibble: 5 x 3
##   team                  payroll_mean win_pct_mean
##   <chr>                        <dbl>        <dbl>
## 1 New York Yankees             168.         0.594
## 2 Boston Red Sox               124.         0.553
## 3 Los Angeles Dodgers          110.         0.529
## 4 New York Mets                101.         0.502
## 5 Philadelphia Phillies         99.4        0.519
```

We see that *New York Yankees* has the highest payroll. *Boston Red Sox* is the next highest paid team. The effect of time is not included here.

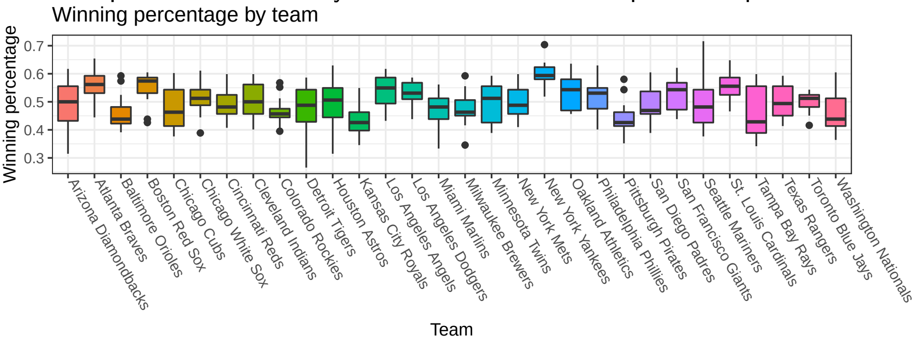
Compare payroll and performance

Summary statistics can not describe the distributions of either payroll or performances. Back to back boxplots of payroll or winning percentage would capture the variability in details.

```
baseball %>%
  ggplot(aes(x = team, y = win_pct, fill = team)) +
  geom_boxplot() +
  xlab("Team") +
  ylab("Winning percentage") +
  ggtitle("Winning percentage by team") +
  theme_bw() +
  theme(legend.position = "none",
        # adjust for margins around the plot; t: top; r: right; b: bottom; l: left
        plot.margin = margin(t = 5, r = 50, b = 5, l = 0, unit = "pt"),
        axis.text.x = element_text(angle = -60, vjust = 0, hjust = 0))
```

Compare payroll and performance

Summary statistics can not describe the distributions of either payroll or performances. Back to back boxplots of payroll or winning percentage would capture the variability in details. to back boxplots 2-1.pdf



We see clearly that the medians/means and spreads are very different. Is there a more informative way to display this?

Compare payroll and performance

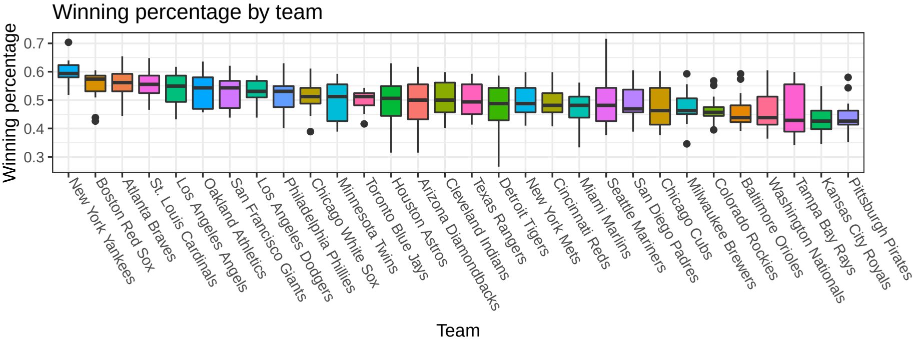
For example, we probably want to display the comparison by ranking the median:

```
boxplot_theme <-
  theme_bw() +
  theme(legend.position = "none",
        # adjust for margins around the plot; t: top; r: right; b: bottom; l: left
        plot.margin = margin(t = 5, r = 50, b = 5, l = 0, unit = "pt"),
        axis.text.x = element_text(angle = -60, vjust = 0, hjust = 0))
```

```
baseball %>%
  ggplot(aes(x = forcats::fct_reorder(team, -win_pct, .fun = median), #order win_pct in a decreasing order
             y = win_pct, fill = team)) +
  geom_boxplot() +
  xlab("Team") +
  ylab("Winning percentage") +
  ggtitle("Winning percentage by team") +
  boxplot_theme
```

Compare payroll and performance

We probably want to display the comparison by ranking the median for example:



We see that `NY Yankees` and `Red Sox` are consistently good teams while `Oakland A's` has a good overall team performance but the performance varies.

Compare payroll and performance

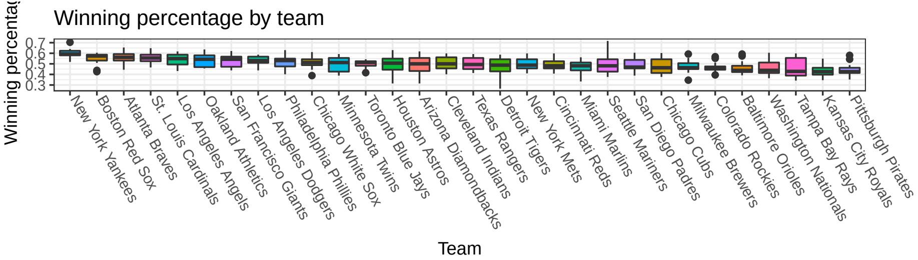
- Next: compare both `payroll` and `win_pct` by teams.
- Let us try to line up two back to back boxplots together.
- Notice that we tried to rank one variable while carrying the other variable in the same order.
- The hope is to reveal the relationship between `payroll` and `performance`.

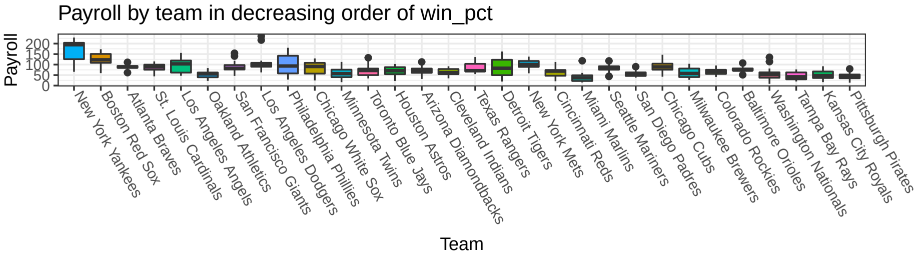
Compare payroll and performance

```r
# use reorder_within() and scale_x_reordered() from tidytext to order boxplot within each facet
library(tidytext)
p_win_pct <- baseball %>%
  ggplot(aes(x = forcats::fct_reorder(team, -win_pct, .fun = median), #order win_pct in a decreasing order
             y = win_pct, fill = team)) +
  geom_boxplot() +
  xlab("Team") +
  ylab("Winning percentage") +
  ggtitle("Winning percentage by team") +
  boxplot_theme

p_payroll <- baseball %>%
  ggplot(aes(x = forcats::fct_reorder(team, -win_pct, .fun = median), #order win_pct in a decreasing order
             y = payroll, fill = team)) +
  geom_boxplot() +
  xlab("Team") +
  ylab("Payroll") +
  ggtitle("Payroll by team in decreasing order of win_pct") +
  boxplot_theme

gridExtra::grid.arrange(p_win_pct, p_payroll, ncol=1)
```

```r
# ggpubr::ggarrange(p_win_pct, p_payroll, ncol = 1)
```

Compare payroll and performance





Bingo! While `Oakland A's` payroll are consistently lower than that of `Red Sox`, they have similar performance!!!

Compare payroll and performance

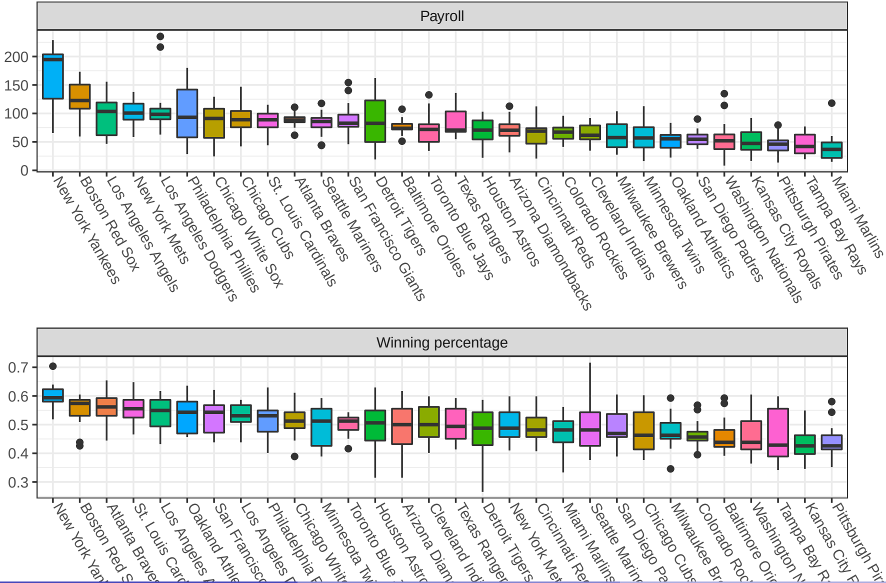
Alternative boxplot faceting

```
# use reorder_within() and scale_x_reordered() from tidytext to order boxplot within each facet
library(tidytext)
# facet names
facet_names <- c("payroll" = "Payroll",
                 "win_pct" = "Winning percentage")
baseball %>%
  select(-win_num) %>%
  pivot_longer(cols = c("payroll", "win_pct"),
               names_to = "variable") %>%
  ggplot(aes(x = reorder_within(team, -value, variable, fun = median),
             y = value, fill = team)) +
  geom_boxplot() +
  scale_x_reordered() +
  facet_wrap(~ variable, ncol = 1, scales = "free",
             labeller = as_labeller(facet_names)) +
  xlab("Team") + ylab("") +
  ggtitle("Payroll and winning percentage by team") +
  boxplot_theme
```

Compare payroll and performance

Alternative boxplot faceting facet b-b plots 2-1.pdf

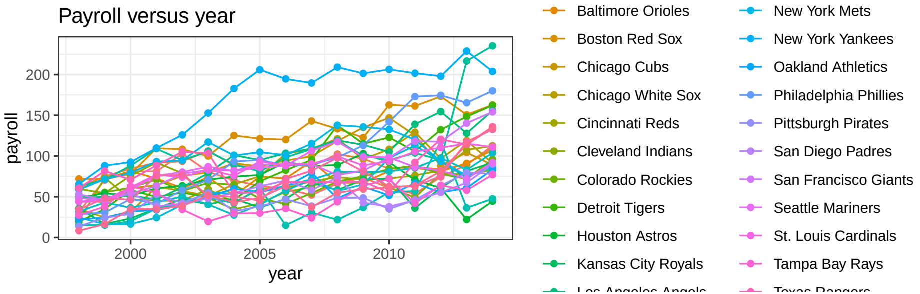
Payroll and winning percentage by team



Comparing performance as a function of time

A time series of performance may reveal patterns of performance over the years to see if some teams are consistently better or worse.

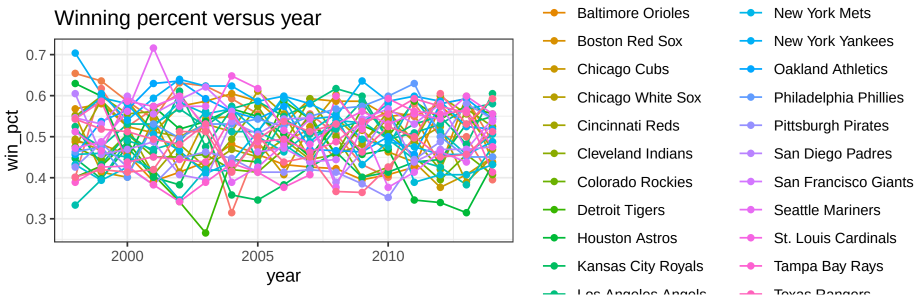
```
payroll_plot <- baseball %>%
  ggplot(aes(x = year, y = payroll, group = team, col = team)) +
  geom_line() +
  geom_point() +
  theme_bw() +
  ggtitle("Payroll versus year")
payroll_plot
```



Comparing performance as a function of time

A time series of performance may reveal patterns of performance over the years to see if some teams are consistently better or worse.

```
win_pct_plot <- baseball %>%
  ggplot(aes(x = year, y = win_pct, group = team, col = team)) +
  geom_line() +
  geom_point() +
  theme_bw() +
  ggtitle("Winning percent versus year")
win_pct_plot
```



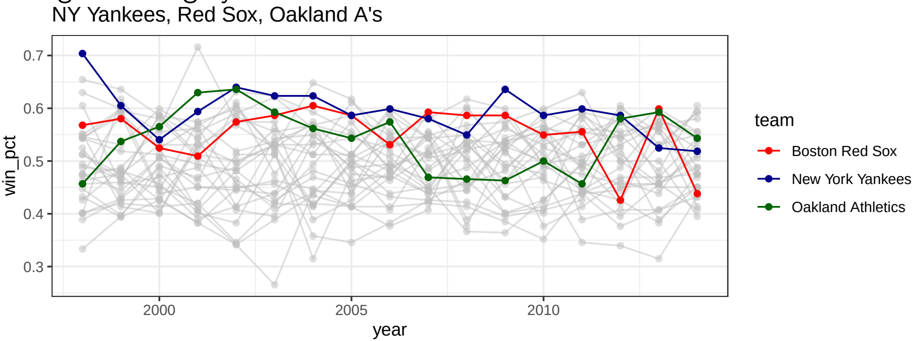
Comparing performance as a function of time

Winning pct plot with only `NY Yankees` (blue), `Boston Red Sox` (red) and `Oakland Athletics` (green) while keeping all other teams as background in gray.

```
selected_teams <- c("New York Yankees", "Boston Red Sox", "Oakland Athletics")

win_pct_plot <- baseball %>%
  ggplot(aes(x = year, y = win_pct, group = team)) +
  geom_line(col = "grey", alpha = .5) +
  geom_point(col = "grey", alpha = .5) +
  geom_line(data = subset(baseball, team %in% selected_teams),
            aes(col = team)) +
  geom_point(data = subset(baseball, team %in% selected_teams),
            aes(col = team)) +
  scale_color_manual(values = c("red", "darkblue", "darkgreen")) +
  theme_bw() +
  ggtitle("NY Yankees, Red Sox, Oakland A's")
  # layout(legend = list(x = 0.35, y = 0.99, orientation = 'h'))
win_pct_plot
```

Comparing performance as a function of time

Winning pct plot with only `NY Yankees` (blue), `Boston Red Sox` (red) and `Oakland Athletics` (green) while keeping all other teams as background in gray.



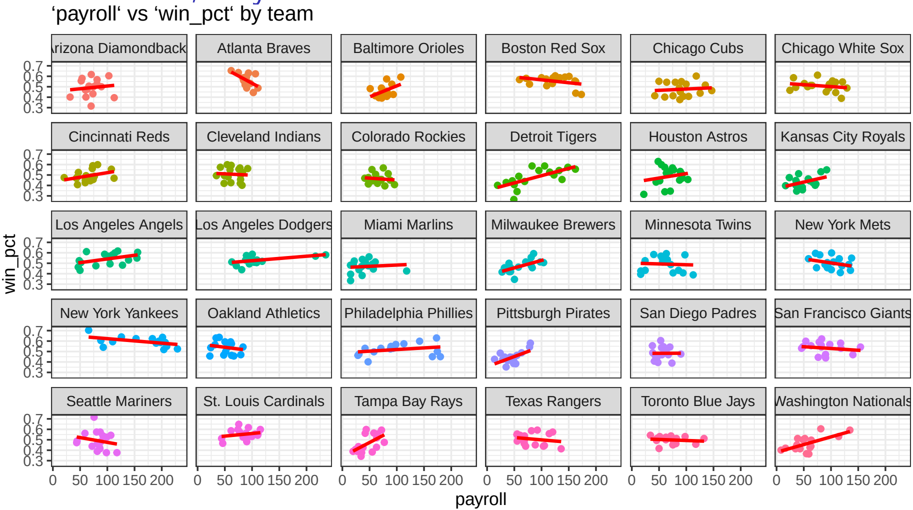
Now we see that `Red Sox` seems to perform better most of the time compared to the `Oakland A's`.

Performance, Payroll and Year

We are trying to reveal the relationship between `performance` and `payroll`. But it depends on which team at a given year.

```
baseball %>%
  ggplot(aes(x=payroll, y=win_pct, group = team, color=team)) +
  geom_point()+
  geom_smooth(method="lm", formula=y~x, se=F,color = "red")+
  facet_wrap(~team) +
  theme_bw() +
  theme(legend.position = "none") +
  ggtitle("`payroll` vs `win_pct` by team")
```

Performance, Payroll and Year

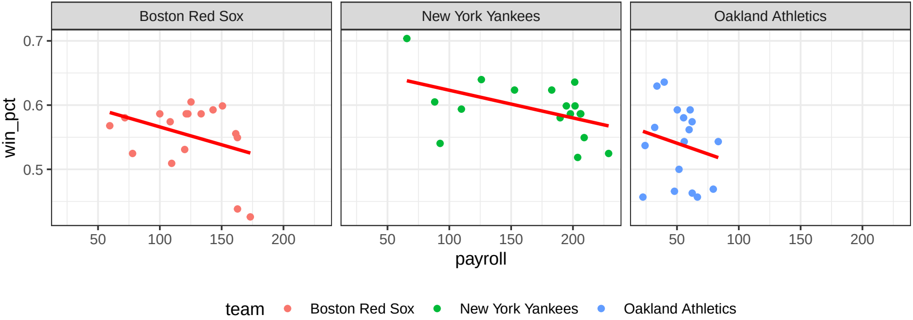


We see a discrepancy among teams for the relationship between `payroll` and `performance`. The positive trends vary from very positive to even negatively correlated.

Performance, Payroll and Year

If we zoom in on a few teams we see a clear negative correlation between payroll and performance. What is missing here?

```
baseball %>%
  filter(team %in% c("New York Yankees", "Boston Red Sox", "Oakland Athletics")) %>%
  ggplot(aes(x=payroll, y=win_pct, group = team, color=team)) +
  geom_point()+
  geom_smooth(method="lm", formula= y~x,  se=F,color = "red")+
  facet_wrap(~team) +
  theme_bw() +
  theme(legend.position = "bottom")
```

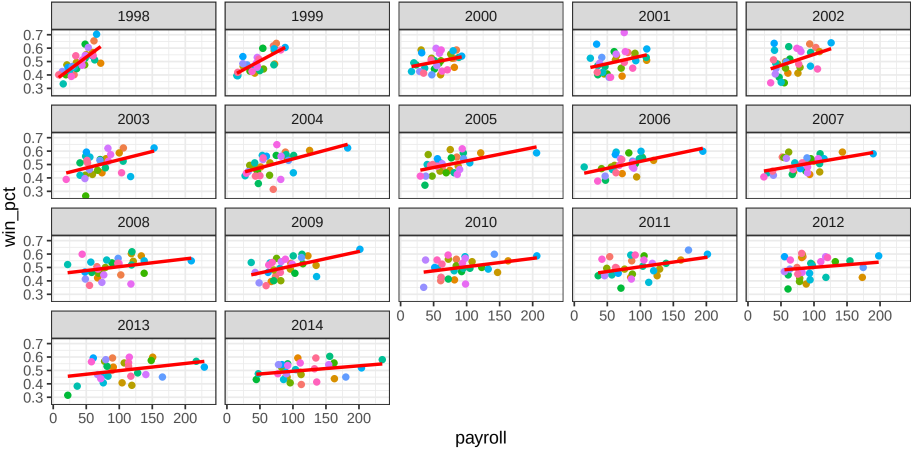


Performance, Payroll and Year

We have seen before, `payroll` increases over years. It will be better to examine `payroll` v.s. `win_pct` by `year`:

```
baseball %>%
  ggplot(aes(x=payroll, y=win_pct, group = year, color=team)) +
  geom_point()+
  geom_smooth(method="lm", formula=y~x, se=F,color = "red")+
  facet_wrap(~year) +
  theme_bw() +
  theme(legend.position = 0)
```

Performance, Payroll and Year



Now it seems to agree with our intuition, payroll and performance are indeed positively related for a given year. But the degree of relationship seems to change depending on which year and they are heavily controlled by some teams.

Performance, Payroll and Year

We can summarize the above three dimension plots via a movie that tracks dynamic changes!

See the plotly movie in the html file!

```
selected_team <- c("Oakland Athletics", "New York Yankees", "Boston Red Sox")

p <- baseball %>%
  ggplot(aes(x=payroll, y=win_pct, color=team, frame = year)) +
   theme(legend.position = 0) +
  geom_point() +
  geom_smooth(method="lm", formula=y~x, se=F, color = "red") +
  geom_text(data = subset(baseball, team %in% selected_team),
            aes(label = team),
            show.legend = FALSE) +
  theme_bw()

ggplotly(p)
```

Perhaps we do not see strong evidence that `Oakland A's` is comparable to `Red Sox` in performance.

Conclusions and Discussion

- We have shown the power of exploratory data analysis to reveal correlation between payroll and performance.
- Is payroll an important factor affecting the team performance if taking more factors into account?
  - Here we assembled a dataset containing only performance and payroll at the team level over a span of 17 yrs.
- Analysis via aggregated statistics can be misleading.
- Substantial variation can also exist within each team.
  - For example, payroll distribution is drastically different.
  - See this article on MLB income inequality: https://fivethirtyeight.com/features/good-mlb-teams-oppose-income-inequality/

Conclusions and Discussion

Questions remain:

1. Based on our current data,
   a) what model will you consider to capture effects of payroll, year and team over the performace?
   b) would you use other measurements as dependent variable, e.g. annual payroll increase?
2. If you are asked to run the study to find out what are the main factors affecting performance, how would you do it?
   To narrow down the scope of the first step of the study, what information you may gather?

Appendix: Sample Statistics

We remind readers of the definition of sample statistics here.

- Sample mean:

$$\bar{y} = \frac{1}{n}\sum_{i=1}^{n} y_i$$

- Sample variance:

$$s^2 = \frac{\sum_{i=1}^{n}(y_i - \bar{y})^2}{n-1}$$

Appendix: Sample Statistics

- Sample Standard Deviation:

$$s = \sqrt{\frac{\sum_{i=1}^{n}(y_i - \bar{y})^2}{n-1}}$$

- Sample correlation

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{s_x s_y}$$