DSC1107:

Formative Assessment 2

Report

Far Eastern University
DSC1107: Data Mining & Wrangling
Mr. Frederick Gella
February 26, 2024

ROSALES, Frances Aneth C.

DSC1107: Formative Assessment 2
Frances Aneth Rosales
Due: February 26, 2024 at 11:59pm

Contents

Instructions

Materials

The allowed materials are as stated on the syllabus:

> Students may consult all course materials, including course textbooks, for all assignments and assessments. For programming-based assignments (homeworks and exams), students may also consult the internet (e.g. Stack Overflow) for help with general programming tasks (e.g. how to add a dashed line to a plot). Students may not search the internet for help with specific questions or specific datasets on any homework or exam. In particular, students may not use solutions to problems that may be available online and/or from past iterations of the course."

Writeup

Use this document as a starting point for your writeup, adding your solutions after "**Solution**". Add your R code using code chunks and add your text answers using **bold text**. Be sure that your compilation, creation of figures and tables, and presentation possess high quality. In particular, if the instructions ask you to "print a table", you should use `kable`. If the instructions ask you to "print a tibble", you should not use `kable` and instead print the tibble directly.

Programming

The *tidyverse* paradigm for data visualization, manipulation, and wrangling is required. No points will be awarded for code written in base R.

Grading

The point value for each problem sub-part is indicated. Additionally, the presentation quality of the solution for each problem will be evaluated on a per-problem basis (e.g. in this homework, there are three problems). There are 100 points possible on this homework, 85 of which are for correctness and 11 of which are for presentation. Your total score will be converted to a total 50 points, per formative assessment policy that FAs should have lower total points than SAs.

Submission

Compile your writeup to PDF and submit to Canvas.

Case study: Major League Baseball

What is the relationship between payroll and wins among Major League Baseball (MLB) teams? In this homework, we'll find out by wrangling, exploring, and modeling the dataset in `MLPayData_Total.rdata`, which contains the winning records and the payroll data of all 30 MLB teams from 1998 to 2014.

The dataset has the following variables:

- `payroll`: total team payroll (in billions of dollars) over the 17-year period
- `avgwin`: the aggregated win percentage over the 17-year period
- `Team.name.2014`: the name of the team
- `p1998`, …, `p2014`: payroll for each year (in millions of dollars)
- `X1998`, …, `X2014`: number of wins for each year
- `X1998.pct`, …, `X2014.pct`: win percentage for each year

We'll need to use the following `R` packages:

```
library(tidyverse)  # tidyverse

library(ggrepel)     # for scatter plot point labels
library(kableExtra) # for printing tables
library(cowplot)     # for side by side plots
```

1 Wrangle (35 points for correctness; 5 points for presentation)

1.1 Import (5 points)

- Import the data into a `tibble` called `mlb_raw` and print it.

Hide

```
load("ml_pay.rdata")
library(tibble)
mlb_raw <- as_tibble(ml_pay)
print(mlb_raw)
```

```
## # A tibble: 30 × 54
##    payroll avgwin Team.n…¹ p1998 p1999 p2000 p2001 p2002 p2003 p2004 p2005 p2006
##      <dbl>  <dbl> <fct>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1   1.12   0.490 Arizona…  31.6  70.5  81.0  81.2 103.   80.6  70.2  63.0  59.7
##  2   1.38   0.553 Atlanta…  61.7  74.9  84.5  91.9  93.5 106.   88.5  85.1  90.2
##  3   1.16   0.454 Baltimo…  71.9  72.2  81.4  72.4  60.5  73.9  51.2  74.6  72.6
##  4   1.97   0.549 Boston …  59.5  71.7  77.9 110.  108.   99.9 125.  121.  120.
##  5   1.46   0.474 Chicago…  49.8  42.1  60.5  64.0  75.7  79.9  91.1  87.2  94.4
##  6   1.32   0.511 Chicago…  35.2  24.5  31.1  62.4  57.1  51.0  65.2  75.2 103.
##  7   1.02   0.486 Cincinn…  20.7  73.3  46.9  45.2  45.1  59.4  43.1  59.7  60.9
##  8   0.999  0.496 Clevela…  59.5  54.4  75.9  92.0  78.9  48.6  34.6  41.8  56.0
##  9   1.03   0.463 Colorad…  47.7  55.4  61.1  71.1  56.9  67.2  64.6  47.8  41.2
## 10   1.43   0.482 Detroit…  19.2  35.0  58.3  49.8  55.0  49.2  46.4  69.0  82.6
## # … with 20 more rows, 42 more variables: p2007 <dbl>, p2008 <dbl>,
## #   p2009 <dbl>, p2010 <dbl>, p2011 <dbl>, p2012 <dbl>, p2013 <dbl>,
## #   p2014 <dbl>, X2014 <int>, X2013 <int>, X2012 <int>, X2011 <int>,
## #   X2010 <int>, X2009 <int>, X2008 <int>, X2007 <int>, X2006 <int>,
## #   X2005 <int>, X2004 <int>, X2003 <int>, X2002 <int>, X2001 <int>,
## #   X2000 <int>, X1999 <int>, X1998 <int>, X2014.pct <dbl>, X2013.pct <dbl>,
## #   X2012.pct <dbl>, X2011.pct <dbl>, X2010.pct <dbl>, X2009.pct <dbl>, …
```

- How many rows and columns does the data have?

Hide

```
mlb_rows <- nrow(mlb_raw)
mlb_columns <- ncol(mlb_raw)

cat("Number of rows of ML Pay:", mlb_rows, "\n")
```

```
## Number of rows of ML Pay: 30
```

Hide

```
cat("Number of columns of ML Pay:", mlb_columns, "\n")
```

```
## Number of columns of ML Pay: 54
```

- Does this match up with the data description given above?

In accordance of the data that I imported into the table with the given data description, it is indeed similar data as assumed.

Therefore, we can now analyze the data of Major League Baseball (MLB) teams' payroll.

Solution.

1.2 Tidy (15 points)

The raw data are in a messy format: Some of the column names are hard to interpret, we have data from different years in the same row, and both year-by-year and aggregate data are present.

- Tidy the data into two separate `tibbles`: one called `mlb_aggregate` containing the aggregate data and another called `mlb_yearly` containing the year-by-year data. `mlb_total` should contain columns named `team`, `payroll_aggregate`, `pct_wins_aggregate` and `mlb_yearly` should contain columns named `team`, `year`, `payroll`, `pct_wins`, `num_wins`. Comment your code to explain each step.

- Print these two `tibbles`. How many rows do `mlb_aggregate` and `mlb_yearly` contain, and why?

We need to change the variables containing the following

mlb_aggregate: aggregate data

mlb_yearly: year-by-year data

mlb_aggregate: columns named team,payroll_aggregate, pct_wins_aggregate

mlb_yearly: contain columns named team, year, payroll, pct_wins, num_wins

mlb_aggregate tibble

Hide

```
library(tidyverse)
mlb_raw <- mlb_raw %>%
  rename_all(str_to_lower) %>%
  rename(team = team.name.2014, avgwin = avgwin)

mlb_raw_aggre <- mlb_raw %>%
  rename_all(str_to_lower) %>%
  rename( the_payroll = payroll )
view(mlb_raw_aggre)



mlb_aggregate <- mlb_raw_aggre %>%
  select(team, starts_with("the_payroll"), matches("^X.*\\.pct$")) %>%
  rename_with(~ "payroll_aggregate", starts_with("the_payroll")) %>%
  mutate(pct_wins_aggregate = rowMeans(select(., matches("^X.*\\.pct$")))) %>%
  select(team, starts_with("payroll_aggregate"), pct_wins_aggregate)
print(mlb_aggregate)
```

```
## # A tibble: 30 × 3
##    team                 payroll_aggregate pct_wins_aggregate
##    <fct>                            <dbl>              <dbl>
##  1 Arizona Diamondbacks             1.12               0.492
##  2 Atlanta Braves                   1.38               0.563
##  3 Baltimore Orioles                1.16               0.457
##  4 Boston Red Sox                   1.97               0.551
##  5 Chicago Cubs                     1.46               0.475
##  6 Chicago White Sox                1.32               0.507
##  7 Cincinnati Reds                  1.02               0.491
##  8 Cleveland Indians                0.999              0.505
##  9 Colorado Rockies                 1.03               0.463
## 10 Detroit Tigers                   1.43               0.474
## # … with 20 more rows
```

mlb_aggregate tibble

Hide

```
mlb_raw_yearly <- mlb_raw %>%
  rename_all(str_to_lower) %>%
  rename_with(~ paste0("pct_wins_", str_remove(., "^X")), matches("^X.*\\.pct$")) %>%
  rename(the_payroll = payroll)

mlb_yearly <- mlb_raw_yearly %>%
  select(team, the_payroll, starts_with("p"), starts_with("x"), starts_with("wins_aggregate")) %>%
  rename_with(
    ~ str_replace(., "^p(\\d+)$", "payroll_\\1"),
    starts_with("p")
  ) %>%
  rename_with(
    ~ str_replace(., "^x(\\d+)$", "num_wins_\\1"),
    starts_with("x")
  ) %>%
  rename_with(
    ~ str_replace(., "^wins_aggregate", "wins_aggregate"),
    starts_with("wins_aggregate")
  )
print(mlb_yearly)
```

```
## # A tibble: 30 × 53
##    team  the_p…¹ payro…² payro…³ payro…⁴ payro…⁵ payro…⁶ payro…⁷ payro…⁸ payro…⁹
##    <fct>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>
##  1 Ariz…   1.12     31.6    70.5    81.0    81.2   103.     80.6    70.2    63.0
##  2 Atla…   1.38     61.7    74.9    84.5    91.9    93.5   106.     88.5    85.1
##  3 Balt…   1.16     71.9    72.2    81.4    72.4    60.5    73.9    51.2    74.6
##  4 Bost…   1.97     59.5    71.7    77.9   110.    108.     99.9   125.    121.
##  5 Chic…   1.46     49.8    42.1    60.5    64.0    75.7    79.9    91.1    87.2
##  6 Chic…   1.32     35.2    24.5    31.1    62.4    57.1    51.0    65.2    75.2
##  7 Cinc…   1.02     20.7    73.3    46.9    45.2    45.1    59.4    43.1    59.7
##  8 Clev…   0.999    59.5    54.4    75.9    92.0    78.9    48.6    34.6    41.8
##  9 Colo…   1.03     47.7    55.4    61.1    71.1    56.9    67.2    64.6    47.8
## 10 Detr…   1.43     19.2    35.0    58.3    49.8    55.0    49.2    46.4    69.0
## # … with 20 more rows, 43 more variables: payroll_2006 <dbl>,
## #   payroll_2007 <dbl>, payroll_2008 <dbl>, payroll_2009 <dbl>,
## #   payroll_2010 <dbl>, payroll_2011 <dbl>, payroll_2012 <dbl>,
## #   payroll_2013 <dbl>, payroll_2014 <dbl>, pct_wins_x2014.pct <dbl>,
## #   pct_wins_x2013.pct <dbl>, pct_wins_x2012.pct <dbl>,
## #   pct_wins_x2011.pct <dbl>, pct_wins_x2010.pct <dbl>,
## #   pct_wins_x2009.pct <dbl>, pct_wins_x2008.pct <dbl>, …
```

Row Numbers

Hide

```
mlb_aggregaterows <- nrow(mlb_aggregate)
mlb_yearlyrows <- nrow(mlb_yearly)

cat("Number of rows of mlb_aggregate:", mlb_aggregaterows, "\n")
```

```
## Number of rows of mlb_aggregate: 30
```

Hide

```
cat("Number of rows of mlb_yearly  :", mlb_yearlyrows, "\n")
```

```
## Number of rows of mlb_yearly  : 30
```

[Hint: For mlb_yearly, the main challenge is to extract the information from the column names. To do so, you can pivot_longer all these column names into one column called column_name, separate this column into three called prefix, year, suffix, mutate prefix and suffix into a a new column called tidy_col_name that takes values payroll, num_wins, or pct_wins, and then pivot_wider to make the entries of tidy_col_name into column names.]

Solution.

1.3 Quality control (15 points)

It's always a good idea to check whether a dataset is internally consistent. In this case, we are given both aggregated and yearly data, so we can check whether these match. To this end, carry out the following steps:

- Create a new tibble called mlb_aggregate_computed based on aggregating the data in mlb_yearly, containing columns named team, payroll_aggregate_computed, and pct_wins_aggregate_computed.

mlb_aggregate_computed tibble

Hide

```
library(tidyverse)
mlb_aggregate_computed <- mlb_yearly %>%
  group_by(team) %>%
  summarise(
    payroll_aggregate_computed = sum(across(starts_with("payroll_")), na.rm = TRUE),  # Total team payroll
    pct_wins_aggregate_computed = mean(sum(across(starts_with("pct_wins_x")), na.rm = TRUE), na.rm = TRUE) / 17  )
```

- Ideally, mlb_aggregate_computed would match mlb_aggregate. To check whether this is the case, join these two tibbles into mlb_aggregate_joined (which should have five columns: team, payroll_aggregate, pct_wins_aggregate, payroll_aggregate_computed, and pct_wins_aggregate_computed.)

```
mlb_aggregate_joined <- mlb_aggregate %>%
  left_join(mlb_aggregate_computed, by = "team")
print(mlb_aggregate_joined)
```

```
## # A tibble: 30 × 5
##    team                  payroll_aggregate pct_wins_aggregate payroll_a…¹ pct_w…²
##    <fct>                             <dbl>              <dbl>       <dbl>   <dbl>
##  1 Arizona Diamondbacks              1.12               0.492       1223.   0.492
##  2 Atlanta Braves                    1.38               0.563       1518.   0.563
##  3 Baltimore Orioles                 1.16               0.457       1305.   0.457
##  4 Boston Red Sox                    1.97               0.551       2104.   0.551
##  5 Chicago Cubs                      1.46               0.475       1552.   0.475
##  6 Chicago White Sox                 1.32               0.507       1375.   0.507
##  7 Cincinnati Reds                   1.02               0.491       1119.   0.491
##  8 Cleveland Indians                 0.999              0.505       1113.   0.505
##  9 Colorado Rockies                  1.03               0.463       1129.   0.463
## 10 Detroit Tigers                    1.43               0.474       1484.   0.474
## # … with 20 more rows, and abbreviated variable names
## #   ¹payroll_aggregate_computed, ²pct_wins_aggregate_computed
```

- Create scatter plots of `payroll_aggregate_computed` versus `payroll_aggregate` and `pct_wins_aggregate_computed` versus `pct_wins_aggregate`, including a 45° line in each. Display these scatter plots side by side, and comment on the relationship between the computed and provided aggregate statistics.

Hide

```r
library(ggplot2)

plot_payroll <- ggplot(mlb_aggregate_joined, aes(x = payroll_aggregate, y = payroll_aggregate_computed)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, linetype = "dashed", color = "red") +
  labs(title = "Scatter Plot of Payroll Aggregate",
       x = "Payroll Aggregate",
       y = "Computed Payroll Aggregate")


plot_pct_wins <- ggplot(mlb_aggregate_joined, aes(x = pct_wins_aggregate, y = pct_wins_aggregate_computed)) +
  geom_point() +
  geom_abline(intercept = 0, slope = 1, linetype = "dashed", color = "blue") +
  labs(title = "Scatter Plot of Win Percentage Aggregate",
       x = "Win Percentage Aggregate",
       y = "Computed Win Percentage Aggregate")
library(grid)

common_title <- textGrob("Figure 1.1", gp = gpar(fontsize = 12, fontface = "bold"))

grid.arrange(
  arrangeGrob(plot_payroll, plot_pct_wins, ncol = 2),
  common_title,
  heights = c(4, 0.5)  # Adjust the heights as needed
)
```

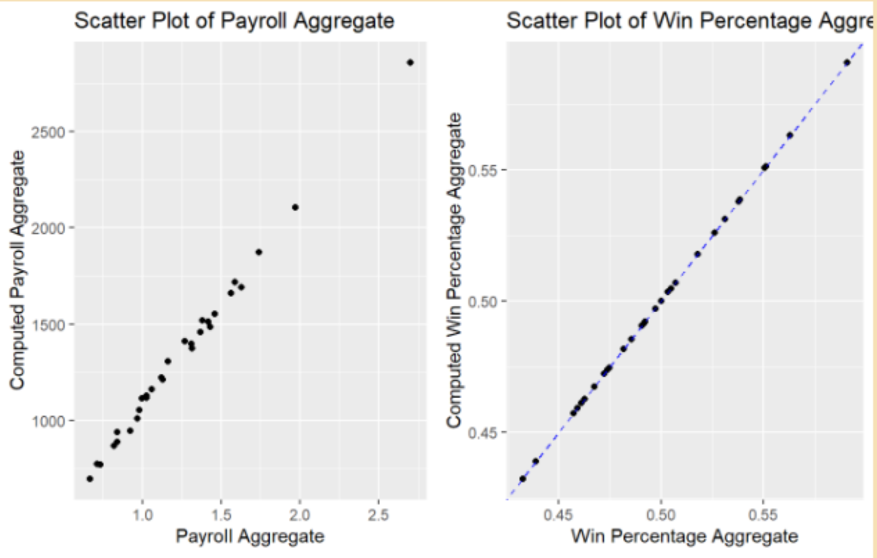


Figure 1.1

Using function ggplot, getting the plot of **payroll_aggregate_computed versus payroll_aggregate** and **pct_wins_aggregate_computed versus pct_wins_aggregate** , we have concluded the plotting shows a linear model which pertains into a proportion increment of data. This will also that the data is close to similar to each other as desired.

Solution.

2 Explore (50 points for correctness; 10 points for presentation)

Now that the data are in tidy format, we can explore them by producing visualizations and summary statistics.

2.1 Payroll across years (15 points)

- Plot `payroll` as a function of `year` for each of the 30 teams, faceting the plot by `team` and adding a red dashed horizontal line for the mean payroll across years of each team.

Hide

```
library(ggplot2)

the_payroll_plot_now <-mlb_yearly %>%
  gather(key = "year", value = "payroll", starts_with("payroll_")) %>%
  ggplot(aes(x = as.numeric(sub("payroll_", "", year)), y = payroll)) +
  geom_point() +
  facet_wrap(~ team, scales = "free_y") +
  geom_hline(aes(yintercept = mean(payroll, na.rm = TRUE)), linetype = "dashed", color = "red") +
  labs(title = "Payroll as a Function of Year for Each Team",
       x = "Year",
       y = "Payroll") +
  theme_minimal()


library(grid)

payroll_plot_title <- textGrob("Figure 1.2", gp = gpar(fontsize = 12, fontface = "bold"))

grid.arrange( the_payroll_plot_now,
  payroll_plot_title,
  heights = c(4, 0.5)
)
```

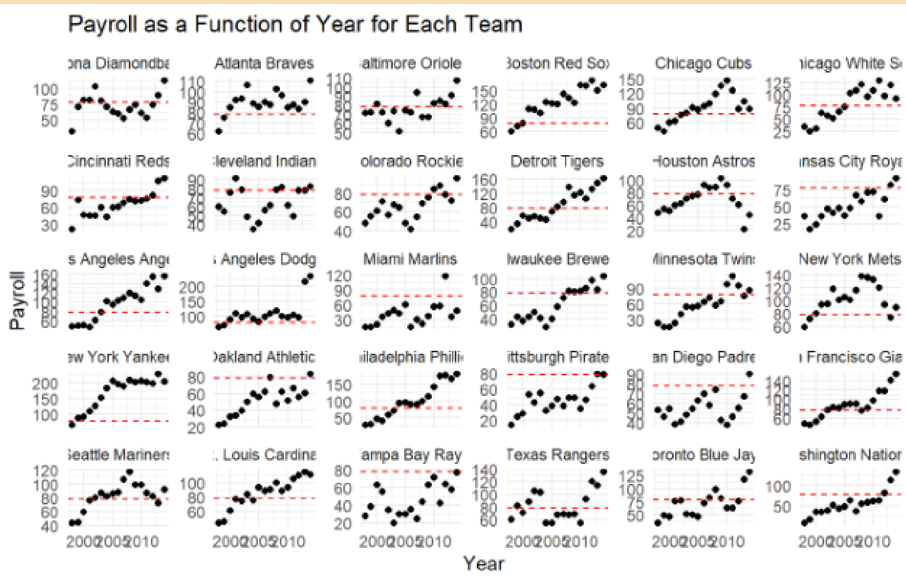


Figure 1.2

Using function ggplot and gather, getting the plot of payroll as a function of year for each of the 30 teams. We can now analyze together the plot of payroll of each team over the years of 1998 to 2014. We can also say that the mean of the payroll of each team are not equal together as the red horizontal line varied for each team.

- Using `dplyr`, identify the three teams with the greatest `payroll_aggregate_computed`, and print a table of these teams and their `payroll_aggregate_computed`.

```
library(dplyr)


top_teams_table <- mlb_aggregate_joined %>%
  top_n(3, payroll_aggregate_computed) %>%
  select(team, payroll_aggregate_computed)

print(top_teams_table)
```

```
## # A tibble: 3 × 2
##   team                payroll_aggregate_computed
##   <chr>                                    <dbl>
## 1 Boston Red Sox                           2104.
## 2 Los Angeles Dodgers                      1874.
## 3 New York Yankees                         2857.
```

- Using `dplyr`, identify the three teams with the greatest percentage increase in payroll from 1998 to 2014 (call it `pct_increase`), and print a table of these teams along with `pct_increase` as well as their payroll figures from 1998 and 2014.

```
library(dplyr)

mlb_yearly_increase <- mlb_yearly %>%
  select(team, matches("^payroll_")) %>%
  rename_with(~ gsub("^payroll_", "", .), matches("^payroll_"))

mlb_top_teams <- mlb_yearly_increase %>%
  group_by(team) %>%
  summarise(
    payroll_1998 = first(`1998`),
    payroll_2014 = first(`2014`),
    pct_increase = ((payroll_2014 - payroll_1998) / payroll_1998) * 100
  ) %>%
  arrange(desc(pct_increase)) %>%
  head(3)

print(mlb_top_teams)
```

```
## # A tibble: 3 × 4
##   team                  payroll_1998 payroll_2014 pct_increase
##   <fct>                        <dbl>        <dbl>        <dbl>
## 1 Washington Nationals          8.32         135.        1520.
## 2 Detroit Tigers               19.2          162.         743.
## 3 Philadelphia Phillies        28.6          180.         529.
```

- How are the metrics `payroll_aggregate_computed` and `pct_increase` reflected in the plot above, and how can we see that the two sets of teams identified above are the top three in terms of these metrics?

As mentioned in our initial plotting of payroll of each teams are distinct to each other as its horizontal line varied to each other, which implicates a different mean of payrolls. The **Boston Red Sox, Los Angeles Dodgers, and New York Yankees** are identified using dplyr as the teams with the **highest payroll_aggregate_computed values**, characterized by high and varying payroll figures over the years.

While, The pct_increase metric shows the percentage increase in payroll from 1998 to 2014, with the **Washington Nationals, Detroit Tigers, and Philadelphia Phillies** showing **the greatest percentage increases** , indicating substantial growth in payroll over the analyzed period.

The plot shows how team payroll fluctuates over time, with teams with higher payroll aggregate values or significant pct_increase values easily identified. This visual representation complements quantitative analysis using dplyr, providing a more comprehensive understanding of payroll dynamics in Major League Baseball, as teams with wider spreads identified.
Additionally, we have shown that the top 3 of high payroll over the years (highes payroll_aggregate_computed) is different with top 3 of the greatest percentage increases of payroll, thus as assumed the an implication of different mean of payrolls.

[Hint: To compute payroll increase, it's useful to `pivot_wider` the data back to a format where different years are in different columns. Use `names_prefix = "payroll_"` inside `pivot_wider` to deal with the fact column names cannot be numbers. To add different horizontal lines to different facets, see this webpage.]

Solution.

2.2 Win percentage across years (15 points)

- Plot `pct_wins` as a function of `year` for each of the 30 teams, faceting the plot by `team` and adding a red dashed horizontal line for the average `pct_wins` across years of each team.

Hide

```
library(ggplot2)

pct_wins_acc_year <- mlb_yearly %>%
  pivot_longer(cols = starts_with("pct_wins_"), names_to = "year", values_to = "pct_wins") %>%
  mutate(year = as.numeric(str_extract(year, "\\d+"))) %>%
  ggplot(aes(x = year, y = pct_wins)) +
  geom_point() +
  geom_hline(aes(yintercept = mean(pct_wins, na.rm = TRUE)), linetype = "dashed", color = "red") +
  facet_wrap(~team, scales = "free_y") +
  labs(title = "Pct Wins Across Years for Each Team",
       x = "Year",
       y = "Pct Wins") +
  theme_minimal()

pct_wins_acc_year_title <- textGrob("Figure 1.3", gp = gpar(fontsize = 12, fontface = "bold"))
grid.arrange( pct_wins_acc_year,
  pct_wins_acc_year_title,
  heights = c(4, 0.5)
)
```

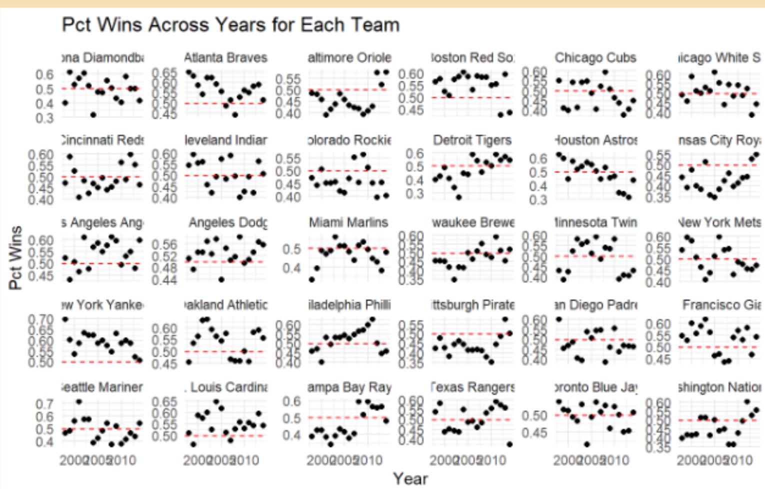


Figure 1.3

- Using dplyr, identify the three teams with the greatest pct_wins_aggregate_computed and print a table of these teams along with pct_wins_aggregate_computed.

Hide

```
library(ggplot2)
thee_top_teams <- mlb_aggregate_joined %>%
  top_n(3, pct_wins_aggregate_computed) %>%
  select(team, pct_wins_aggregate_computed)
print(thee_top_teams)
```

```
## # A tibble: 3 × 2
##   team              pct_wins_aggregate_computed
##   <chr>                                   <dbl>
## 1 Atlanta Braves                          0.563
## 2 Boston Red Sox                          0.551
## 3 New York Yankees                        0.591
```

- Using dplyr, identify the three teams with the most erratic pct_wins across years (as measured by the standard deviation, call it pct_wins_sd) and print a table of these teams along with pct_wins_sd.

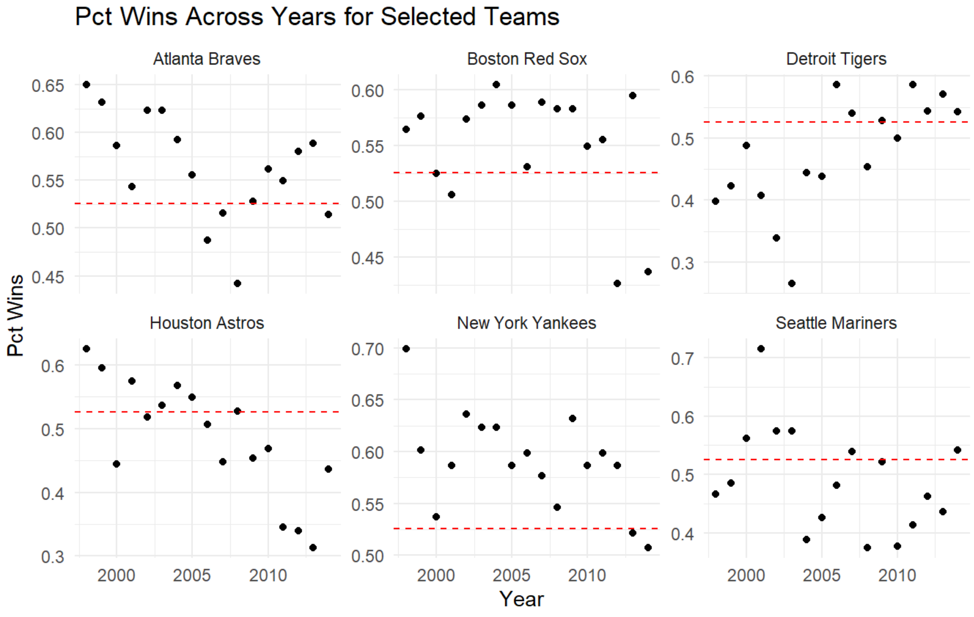
Hide

```
top_erratic_teams <- mlb_yearly %>%
  gather(key = "year", value = "pct_wins", starts_with("pct_wins_x")) %>%
  group_by(team) %>%
  summarise(pct_wins_sd = sd(pct_wins, na.rm = TRUE)) %>%
  top_n(3, pct_wins_sd) %>%
  select(team, pct_wins_sd)

print(top_erratic_teams)
```

```
## # A tibble: 3 × 2
##   team              pct_wins_sd
##   <fct>                   <dbl>
## 1 Detroit Tigers         0.0898
## 2 Houston Astros         0.0914
## 3 Seattle Mariners       0.0892
```

- How are the metrics `pct_wins_aggregate_computed` and `pct_wins_sd` reflected in the plot above, and how can we see that the two sets of teams identified above are the top three in terms of these metrics?



Plot of Top (pct_wins_sd) and (pct_wins_aggregate_computed)

The **"Atlanta Braves", "Boston Red Sox", "New York Yankees"** are identified using dplyr as the teams with the **highest pct_wins_aggregate_computed values**, characterized by high and varying wins figures over the years.

While, The pct_wins_sd metric shows the standard deviation in percentage win from 1998 to 2014, with the **"Detroit Tigers","Houston Astros", "Seattle Mariners"** showing **the greatest standard deviation in percentage win** .

As we plot the Top 3 teams of 2 different characteristic, we can see as shown that the plotting implicates a plotting **higher than 0.50**.

Thus, indeed we can see why the team top the rank.

Solution.

2.3 Win percentage versus payroll (15 points)

Let us investigate the relationship between win percentage and payroll.

- Create a scatter plot of `pct_wins` versus `payroll` based on the aggregated data, labeling each point with the team name using `geom_text_repel` from the `ggrepel` package and adding the least squares line.

Hide

```
library(ggplot2)
library(ggrepel)

mlb_aggregate <- mlb_raw_aggre %>%
  select(team, starts_with("the_payroll"), matches("^X.*\\.pct$")) %>%
  rename_with(~ "payroll_aggregate", starts_with("the_payroll")) %>%
  mutate(pct_wins_aggregate = rowMeans(select(., matches("^X.*\\.pct$")))) %>%
  select(team, starts_with("payroll_aggregate"), pct_wins_aggregate)

scatter_plot <- ggplot(mlb_aggregate, aes(x = payroll_aggregate, y = pct_wins_aggregate)) +
  geom_point() +
  geom_text_repel(aes(label = team), box.padding = 0.5) +
  geom_smooth(method = "lm", se = FALSE, color = "blue") +
  labs(title = "Scatter Plot of pct_wins_aggregate vs. payroll_aggregate",
       x = "Payroll Aggregate",
       y = "Pct Wins Aggregate") +
  theme_minimal()


scatter_acc_year_title <- textGrob("Figure 1.4", gp = gpar(fontsize = 12, fontface = "bold"))
grid.arrange( scatter_plot,
  scatter_acc_year_title,
  heights = c(4, 0.5)
)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: ggrepel: 7 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

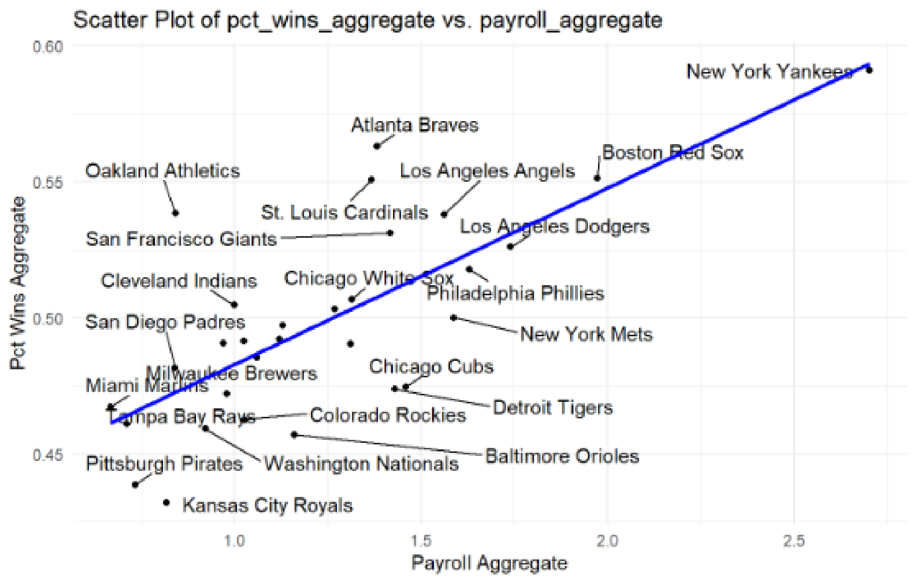


Figure 1.4

- Is the relationship between payroll and pct_wins positive or negative? Is this what you would expect, and why?

The relationship between payroll and pct_wins is posssitive as shown in the plot.
We can see that the plotting in Figure 1.4 that the New York Yankees, Boston Red Sox, and Los Angeles Dodgers continuously increases as Payroll and Pct_Win proportionally increases. Just like what we have on the Top 3 in our **payroll_aggregate_computed** which are also New York Yankees, Boston Red Sox, and Los Angeles Dodgers.
However, we cannot really say that there's a relationship between with the payroll and pct_wins only if we use more tool like a Simple Linear Model, etc., to see if there is a relationship that would make the data indeed proportionally into each other.

Solution.

2.4 Team efficiency (5 points)

Define a team's *efficiency* as the ratio of the aggregate win percentage to the aggregate payroll—more efficient teams are those that win more with less money.

- Using dplyr, identify the three teams with the greatest efficiency, and print a table of these teams along with their efficiency, as well as their pct_wins_aggregate_computed and payroll_aggregate_computed.

Hide

```
mlb_aggregate_computedz <- mlb_aggregate_joined %>%
  group_by(team) %>%
  summarise(
    payroll_aggregate_computed = sum(across(starts_with("payroll_aggregate_")), na.rm = TRUE),
    pct_wins_aggregate_computed = mean(sum(across(starts_with("pct_wins_aggregate_")), na.rm = TRUE), na.rm = TRUE) / 17,
    efficiency = pct_wins_aggregate_computed / payroll_aggregate_computed
  )

top_efficiency_teams <- mlb_aggregate_computedz %>%
  top_n(3, wt = efficiency) %>%
  arrange(desc(efficiency))

print(top_efficiency_teams)
```

```
## # A tibble: 3 × 4
##   team              payroll_aggregate_computed pct_wins_aggregate_comp…¹ effic…²
##   <chr>                                  <dbl>                     <dbl>   <dbl>
## 1 Miami Marlins                           698.                    0.0275 3.94e-5
## 2 Oakland Athletics                       888.                    0.0317 3.57e-5
## 3 Tampa Bay Rays                          776.                    0.0271 3.49e-5
## # … with abbreviated variable names ¹pct_wins_aggregate_computed, ²efficiency
```

- In what sense do these three teams appear efficient in the previous plot?

In accordance with our previous plot Figure 1.4, to say that a team is efficient, the quotient of pct_wins_aggregate_computed divided by payroll_aggregate_computed would be big. In other words, their payroll might not be big however, the percentage of them winning is great.

As seen in Figure 1.4 again, the most obvious team in the plot is Oakland Athletics as the x-axis (Payroll) of team may be low, however the y-axis (Percentafe Wins) is high. In which can appear as well into our Top Three Teams which are Miami Marlins, Oakland Athletics, and lastly, Tampa Bay Rays.

Side note: The movie "Moneyball" portrays "Oakland A's general manager Billy Beane's successful attempt to assemble a baseball team on a lean budget by employing computer-generated analysis to acquire new players."

Solution.