STAT 471: Homework 1

Ashley Clarke

Due: September 19, 2021 at 11:59pm

Contents

Instructions

Setup

Pull the latest version of this assignment from Github and set your working directory to `stat-471-fall-2021/homework/homework-1`. Consult the getting started guide if you need to brush up on `R` or `Git`.

Collaboration

The collaboration policy is as stated on the Syllabus:

> "Students are permitted to work together on homework assignments, but solutions must be written up and submitted individually. Students must disclose any sources of assistance they received; furthermore, they are prohibited from verbatim copying from any source and from consulting solutions to problems that may be available online and/or from past iterations of the course."

In accordance with this policy,

Please list anyone you discussed this homework with: Zach Bradlow and Paul Heysch de la Borde

Please list what external references you consulted (e.g. articles, books, or websites):

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**. Consult the preparing reports guide for guidance on compilation, creation of figures and tables, and presentation quality.

Programming

The `tidyverse` paradigm for data wrangling, manipulation, and visualization is strongly encouraged, but points will not be deducted for using base `R`.

Grading

The point value for each problem sub-part is indicated. Additionally, the presentation quality of the solution for each problem (as exemplified by the guidelines in Section 3 of the preparing reports guide 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 15 of which are for presentation.

Submission

Compile your writeup to PDF and submit to Gradescope.

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 `data/MLPayData_Total.csv`, 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 (30 points for correctness; 5 points for presentation)

1.1 Import (5 points)

- Import the data into a `tibble` called `mlb_raw` and print it.
- How many rows and columns does the data have?
- Does this match up with the data description given above?

```
mlb_raw <- read_csv("~/Desktop/STAT471/stat-471-fall-2021/data/MLPayData_Total.csv")
```

```
## Rows: 30 Columns: 54

## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (1): Team.name.2014
## dbl (53): payroll, avgwin, p1998, p1999, p2000, p2001, p2002, p2003, p2004, ...

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
print(mlb_raw)
```

```
## # A tibble: 30 x 54
##    payroll avgwin Team.name.2014 p1998 p1999 p2000 p2001 p2002 p2003 p2004 p2005
##      <dbl>  <dbl> <chr>          <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1   1.12   0.490 Arizona Diamo~  31.6  70.5  81.0  81.2 103.   80.6  70.2  63.0
##  2   1.38   0.553 Atlanta Braves  61.7  74.9  84.5  91.9  93.5 106.   88.5  85.1
##  3   1.16   0.454 Baltimore Ori~  71.9  72.2  81.4  72.4  60.5  73.9  51.2  74.6
##  4   1.97   0.549 Boston Red Sox  59.5  71.7  77.9 110.  108.   99.9 125.  121.
##  5   1.46   0.474 Chicago Cubs    49.8  42.1  60.5  64.0  75.7  79.9  91.1  87.2
##  6   1.32   0.511 Chicago White~  35.2  24.5  31.1  62.4  57.1  51.0  65.2  75.2
##  7   1.02   0.486 Cincinnati Re~  20.7  73.3  46.9  45.2  45.1  59.4  43.1  59.7
##  8   0.999  0.496 Cleveland Ind~  59.5  54.4  75.9  92.0  78.9  48.6  34.6  41.8
##  9   1.03   0.463 Colorado Rock~  47.7  55.4  61.1  71.1  56.9  67.2  64.6  47.8
## 10   1.43   0.482 Detroit Tigers  19.2  35.0  58.3  49.8  55.0  49.2  46.4  69.0
```

```
## # ... with 20 more rows, and 43 more variables: p2006 <dbl>, p2007 <dbl>,
## #   p2008 <dbl>, p2009 <dbl>, p2010 <dbl>, p2011 <dbl>, p2012 <dbl>,
## #   p2013 <dbl>, p2014 <dbl>, X2014 <dbl>, X2013 <dbl>, X2012 <dbl>,
## #   X2011 <dbl>, X2010 <dbl>, X2009 <dbl>, X2008 <dbl>, X2007 <dbl>,
## #   X2006 <dbl>, X2005 <dbl>, X2004 <dbl>, X2003 <dbl>, X2002 <dbl>,
## #   X2001 <dbl>, X2000 <dbl>, X1999 <dbl>, X1998 <dbl>, X2014.pct <dbl>,
## #   X2013.pct <dbl>, X2012.pct <dbl>, X2011.pct <dbl>, X2010.pct <dbl>, ...
```

[Hint: If your working directory is `stat-471-fall-2021/homework/homework-1`, then you can use a *relative path* to access the data at `../../data/MLPayData_Total.csv`.]

The tibble has 30 rows and 54 columns. These dimensions match the data description becsaue each row corresponds to one of the 30 teams. The 54 columns correspond to team name, payroll, average winning percentage, 17 years of payroll, 17 years of wins, and 17 years of win percentages

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?

[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.]

```
mlb_aggregate <- mlb_raw %>%
  # selects team name, payroll, and avgwin columns
  select(Team.name.2014, payroll, avgwin) %>%
  # renames columns
  rename("team" = "Team.name.2014", "payroll_aggregate" = "payroll",
         "pct_wins_aggregate" = "avgwin")

#formats the chart
mlb_aggregate %>%
  kable(format = "latex", row.names = NA,
      booktabs = TRUE, digits = 2,
      caption = "MLB Team Aggregate Statistics") %>%
  kable_styling(position = "center")
```

mlb__aggregate contains 30 rows and 3 columns because there are 30 teams and 3 columns (2 aggregate stats and the team name)

Table 1: MLB Team Aggregate Statistics

team	payroll_aggregate	pct_wins_aggregate
Arizona Diamondbacks	1.12	0.49
Atlanta Braves	1.38	0.55
Baltimore Orioles	1.16	0.45
Boston Red Sox	1.97	0.55
Chicago Cubs	1.46	0.47
Chicago White Sox	1.32	0.51
Cincinnati Reds	1.02	0.49
Cleveland Indians	1.00	0.50
Colorado Rockies	1.03	0.46
Detroit Tigers	1.43	0.48
Houston Astros	1.06	0.47
Kansas City Royals	0.82	0.43
Los Angeles Angels	1.56	0.55
Los Angeles Dodgers	1.74	0.53
Miami Marlins	0.67	0.48
Milwaukee Brewers	0.98	0.47
Minnesota Twins	0.97	0.50
New York Mets	1.59	0.49
New York Yankees	2.70	0.58
Oakland Athletics	0.84	0.54
Philadelphia Phillies	1.63	0.52
Pittsburgh Pirates	0.73	0.44
San Diego Padres	0.84	0.48
San Francisco Giants	1.42	0.53
Seattle Mariners	1.31	0.49
St. Louis Cardinals	1.37	0.56
Tampa Bay Rays	0.71	0.47
Texas Rangers	1.27	0.50
Toronto Blue Jays	1.13	0.49
Washington Nationals	0.92	0.47

```r
mlb_yearly <- mlb_raw %>%
  # pivots to create a new column that is filled with previous yearly column names
  pivot_longer(-c(payroll, avgwin, Team.name.2014), names_to = 'column_name',
               values_to = "stat") %>%
   #removes payroll and avgwin columns
  select(-c(payroll, avgwin)) %>%
  #sepertes into three columns, where middle column includes characters 2-5
  separate(column_name, c("prefix", "year", "suffix"), sep=c(1,5)) %>%
  #combines prefix and suffix columns
  mutate(tidy_col_name = paste(prefix, suffix)) %>%
  #recodes the factors to allign with new column names
  mutate(tidy_col_name = recode(tidy_col_name, "p " = "payroll",
                                "X .pct" = "pct_wins" , "X "= "num_wins")) %>%
  # removes prefix and suffix columns
  select(-c("prefix", "suffix")) %>%
  #creates seperate columns for each of the tidy column variable names
  pivot_wider(names_from = tidy_col_name, values_from = stat) %>% #
  # renames the team column
  rename("team" = "Team.name.2014")

#prints tibble
print(mlb_yearly)
```

```
## # A tibble: 510 x 5
##    team                 year  payroll num_wins pct_wins
##    <chr>                <chr>   <dbl>    <dbl>    <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
##  7 Arizona Diamondbacks 2004     70.2       51    0.315
##  8 Arizona Diamondbacks 2005     63.0       77    0.475
##  9 Arizona Diamondbacks 2006     59.7       76    0.469
## 10 Arizona Diamondbacks 2007     52.1       90    0.556
## # ... with 500 more rows
```

mlb_yearly contains 510 rows and 5 columns because there are 30 teams and 17 years of data for each team (30*17 = 510). The columns correspond to team name and 4 metrics (year, payroll, num_wins, pct_wins)

1.3 Quality control (10 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`.
- 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`.)
- 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.

```
mlb_aggregate_computed <- mlb_yearly %>%
  # groups by team
  group_by(team) %>%
  # calculates payroll in billions
  summarise(payroll_aggregate_computed = sum(payroll)/1000,
            # calculates average percent wins for all years
            pct_wins_aggregate_computed = mean(pct_wins))

# joins the aggregate and computed tibbles
mlb_aggregate_joined <- left_join(mlb_aggregate, mlb_aggregate_computed, by = "team")

# creates ggplot where x=provided and y = computed payroll
agg_payroll_plot <- mlb_aggregate_joined  %>%
  ggplot() +
  geom_point(mapping =
               aes(x = payroll_aggregate,
                   y = payroll_aggregate_computed)) +
  geom_abline(color = "blue") +
  labs(
    x = "Aggregate Payroll ($ Billions)",
    y = "Computed Aggregate Payroll ($ Billions)"
  ) +
  ggtitle("Computed vs. Provided Payroll") +
  theme(plot.title = element_text(size = 12, face = "bold"))

# creates ggplot where x=provided and y = computed pct_wins
pct_wins_plot <-mlb_aggregate_joined  %>%
  ggplot() +
  geom_point(mapping =
               aes(x = pct_wins_aggregate,
                   y = pct_wins_aggregate_computed)) +
  geom_abline(color = "blue") +
  labs(
    x = "Aggregate Percent Wins (%)",
    y = "Computed Aggregate Percent Wins (%)"
  ) +
  ggtitle("Computed vs. Provided Percent Wins") +
  theme(plot.title = element_text(size = 12, face = "bold"))

#plots grids next to each other
plot_grid(agg_payroll_plot, pct_wins_plot)
```

Payroll: The computed average payroll is close to the aggregate payroll. However, since most of the points are too the left of the 45 degree line, reported aggregate payroll is lower than computed payroll for every team.

Percent Wins: the computed aggregated winning percentage is slightly different than reported percent wins. Unliike payroll, percentage wins is not consistently too high or too low.

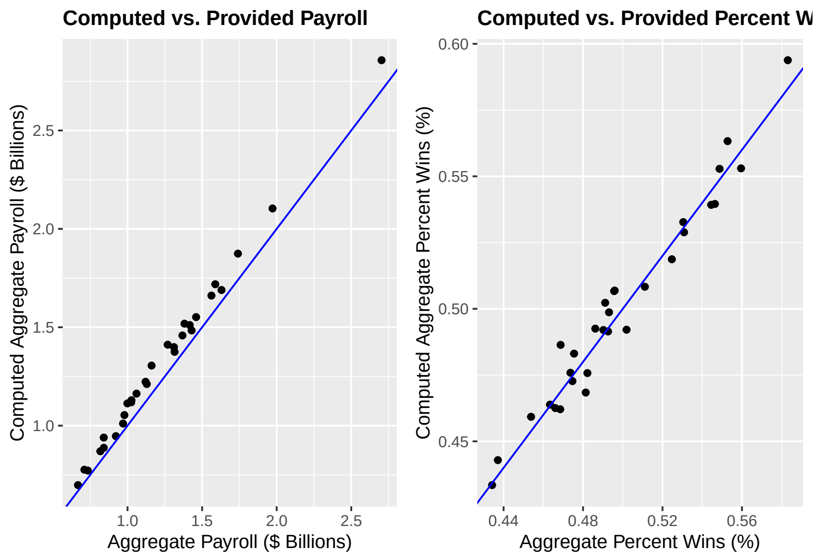

Figure 1: Computed vs. Provided Payroll and Percentage Wins

2 Explore (40 points for correctness; 7 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.

```
#calcualtes mean payroll for each team
payroll_mean <- mlb_yearly %>%
  group_by(team) %>%
  summarise(mean_val= mean(payroll))

#plots payroll by year for each team
mlb_yearly %>%
  ggplot() +
  geom_point(mapping =
               aes(x = year,
                   y = payroll)) +
  geom_hline(data = payroll_mean,
             aes(yintercept = mean_val),
            linetype = "dashed",
             color = "red") +
  # split into facets based on team
  facet_wrap(~ team,
             nrow = 5) +
  #formats the ggplot
  labs(
    x = "Year",
    y = "Payroll ($ Billions)"
  ) +
  scale_x_discrete(breaks = c("2000", "2004", "2008", "2012")) +
```

```
  scale_y_continuous(breaks = c(0, 100, 200)) +
  ggtitle("Team's Payroll by Year") +
  theme(
    plot.title = element_text(size = 12, face = "bold"),
    axis.text.x = element_text(size = 6),
    axis.text.y = element_text(size = 6),
    strip.text = element_text(size = 6, face = "bold"),
    panel.spacing.x = unit(1, "lines"))
```

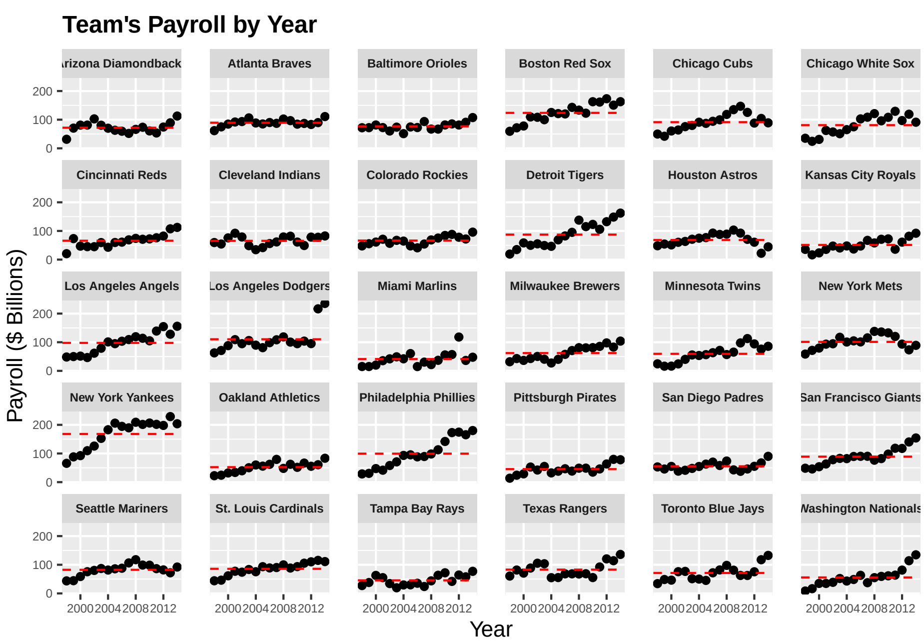

Figure 2: Payroll by Team and Year

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

```
top_3_payroll <- mlb_aggregate_computed %>%
  arrange(desc(payroll_aggregate_computed)) %>% #descending order
  select(team, payroll_aggregate_computed) %>% #selects columns
  head(3) #picks top 3

#formats table
top_3_payroll %>%
  kable(format = "latex", row.names = NA,
      booktabs = TRUE, digits = 2,
      caption = "Top 3 Teams: Aggregate Payroll Computed") %>%
  kable_styling(position = "center", latex_options = "HOLD_position")
```

Table 2: Top 3 Teams: Aggregate Payroll Computed

team	payroll_aggregate_computed
New York Yankees	2.86
Boston Red Sox	2.10
Los Angeles Dodgers	1.87

- 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.

```
payroll_1998 <- mlb_yearly %>%
  filter(year ==  "1998") %>% #selects only 1998 data
  # pivots to create a column for the year
  pivot_wider(names_from = year, values_from = payroll) %>%
  rename(payroll_1998 = "1998")

payroll_2014 <- mlb_yearly %>%
  filter(year ==  "2014") %>%  #selects only 2014 data
  # pivots to create a column for the year
  pivot_wider(names_from = year, values_from = payroll) %>%
  rename(payroll_2014= "2014")

top_3_payroll_inc <- left_join(payroll_1998, payroll_2014, by = "team") %>%
  select("team", "payroll_1998", "payroll_2014") %>%
  # crates new variable for percent increase
  mutate(pct_increase = ((payroll_2014 - payroll_1998)/payroll_1998)*100) %>%
  arrange(desc(pct_increase)) %>% #descending order
  head(3) #selects top 3

top_3_payroll_inc %>%
  kable(format = "latex", row.names = NA,
      booktabs = TRUE, digits = 2,
      caption = "Top 3 Teams: Percent Increase Payroll") %>%
  kable_styling(position = "center", latex_options = "HOLD_position")
```

Table 3: Top 3 Teams: Percent Increase Payroll

team	payroll_1998	payroll_2014	pct_increase
Washington Nationals	8.32	135	1520
Detroit Tigers	19.24	162	743
Philadelphia Phillies	28.62	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?

[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.]

The top three teams with the highest payroll increase are: the Washington Nationals, Detriot Tigers, and the Philadelphia Phillies. The Yankees, Red Sox, and Dodgers had the highest

payroll amounts across all years.None of the teams in the top 3 payroll increase category are found within the top 3 teams in the aggregate computed payroll category.

The teams with the highest `payroll_aggregate_computed` can be identified in the plot above by looking for the 3 teams with the highest mean payrolls. The teams with the highest `pct_increase` have the highest slopes in the plot above

2.2 Win percentage across years (10 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.

```
# calcultes mean pct wins by team
pct_wins_mean <- mlb_yearly %>%
  group_by(team) %>%
  summarise(mean_val= mean(pct_wins))

#creates percent wins by year plot for each team
mlb_yearly %>%                     # pipe in the data
  ggplot() +
  geom_point(mapping =
               aes(x = year,
                   y = pct_wins)) +
  geom_hline(data = pct_wins_mean, #adds aline for the mean
             aes(yintercept = mean_val),
             linetype = "dashed",
             color = "red") +
  facet_wrap(~ team,               # split into facets based on team
             nrow = 5)  +
  labs(
    x = "Year",
    y = "Percentage Wins"
  ) +
  scale_x_discrete(breaks = c("2000", "2004", "2008", "2012")) +
  scale_y_continuous(breaks = c(.25, .5, .75)) +
  ggtitle("Team's Percentage Wins by Year") +
  theme(
    plot.title = element_text(size = 12, face = "bold"),
    axis.text.x = element_text(size = 6),
    axis.text.y = element_text(size = 6),
    strip.text = element_text(size = 6, face = "bold"),
    panel.spacing.x = unit(1, "lines"))
```

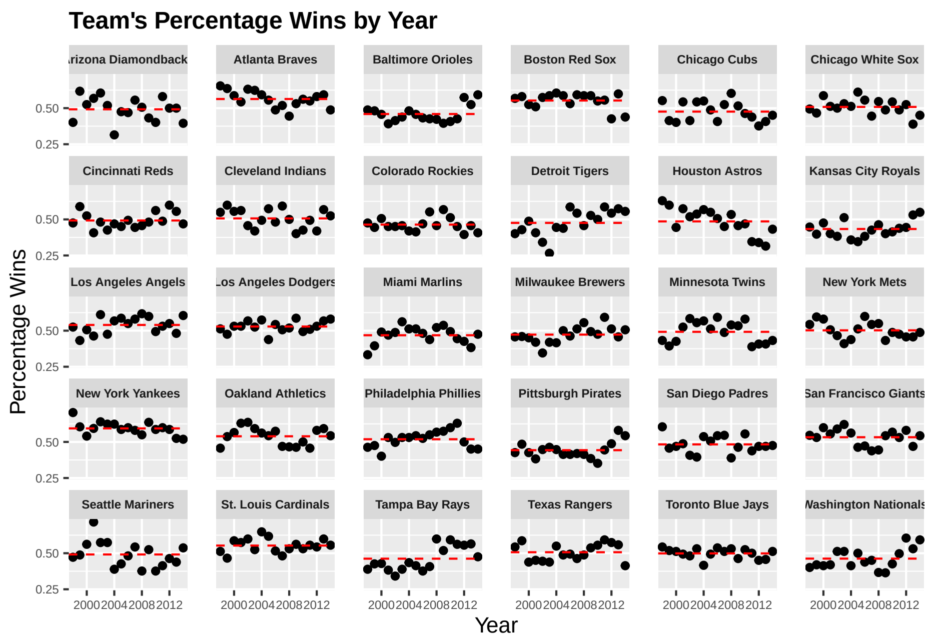

Figure 3: Percentage Wins by Team and Year

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

```
top_3_pct_wins <- mlb_aggregate %>%
  arrange(desc(pct_wins_aggregate)) %>% #descending order
  select(team, pct_wins_aggregate) %>% #column selection
  head(3) #top 3

top_3_pct_wins %>%
  kable(format = "latex", row.names = NA,
      booktabs = TRUE, digits = 2,
      caption = "Top 3 Teams: Percent Wins Aggregate") %>%
  kable_styling(position = "center", latex_options = "HOLD_position")
```

Table 4: Top 3 Teams: Percent Wins Aggregate

team	pct_wins_aggregate
New York Yankees	0.58
St. Louis Cardinals	0.56
Atlanta Braves	0.55

- 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`.

```
top_3_pct_win_sd <- mlb_yearly %>%
  group_by(team) %>% #groups by team
  summarise(pct_wins_sd = sd(pct_wins)) %>% #creates sd variable
  arrange(desc(pct_wins_sd)) %>% #descending order
  head(3) #selects top 3
```

```r
top_3_pct_win_sd %>%
  kable(format = "latex", row.names = NA,
      booktabs = TRUE, digits = 2,
      caption = "Top 3 Teams: Standard Deviation of Percent Wins") %>%
  kable_styling(position = "center", latex_options = "HOLD_position")
```

Table 5: Top 3 Teams: Standard Deviation of Percent Wins

team	pct_wins_sd
Houston Astros	0.09
Detroit Tigers	0.09
Seattle Mariners	0.09

- How are the metrics `payroll_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?

The three teams with the highest payroll_aggregate_computed are the Yankees, Cardinals, and Braves. In the plot above, these teams have the highest mean percentage win value. The horizontal line on the graph shows the average of a teams wins across all years. Thus, the team with the highest line should also have the highest payroll_aggregate_computed value

The three teams with the highest pct_wins_sd are the Astros, Tigers, and Mariners. In the plot above, the Astros, Tigers, and Mariners all have points that are soread out vertically. This happens because pct_wins_sd can be seen by how far the points are away from the horizontal line. If a team has points that are very spread out, their standard deviation will be higher

2.3 Win percentage versus payroll (10 points)

The analysis goal is to study 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.

```r
ggplot(mlb_aggregate, aes(x=payroll_aggregate, y= pct_wins_aggregate)) +
  geom_point() +
  #adjusts labels so they do nto overlap
  geom_text_repel(aes(label = team), size = 2.5, hjust = .1) +
  #adds a line of best fit that is blue and transparent
  geom_line(stat="smooth",method = "lm", formula = y ~ x, se = FALSE,
            color= "blue", alpha = 0.4) +
  labs(
    x = "Payroll ($ Billions)",
    y = "Percentage Wins"
  )
```

- 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 positive. I would expect this to happen because, in theory, successful teams have more money to pay players and better performing players are paid higher. Therefore, more money should translate into more wins.

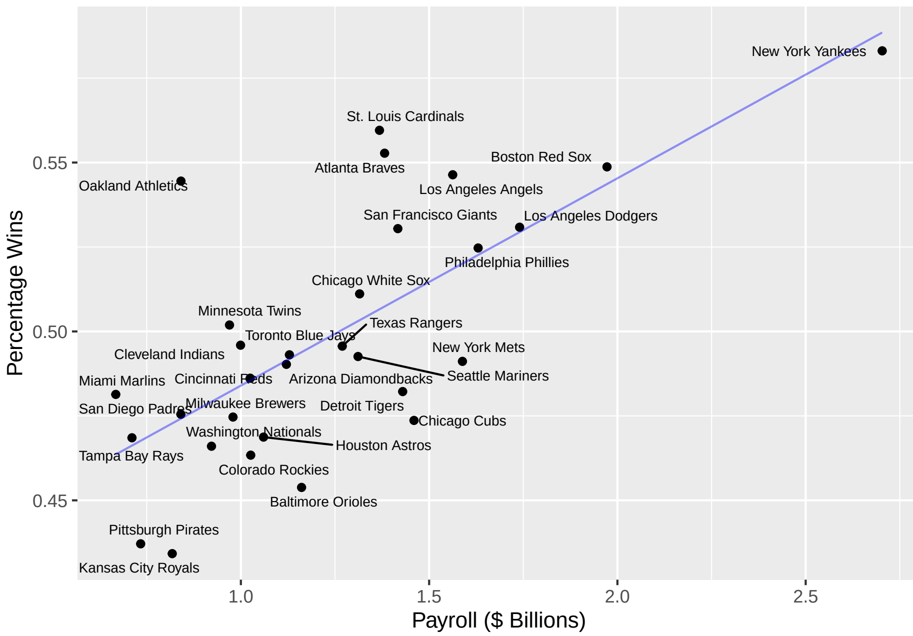

Figure 4: Percentage Wins by Payroll

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` and `payroll_aggregate`.
- In what sense do these three teams appear efficient in the previous plot?

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."

```
top_3_efficiency <- mlb_aggregate %>%
  #creates efficency variable
  summarise(team, efficiency = pct_wins_aggregate/payroll_aggregate,
            pct_wins_aggregate, payroll_aggregate) %>%
  arrange(desc(efficiency)) %>% #descending order
  head(3) #selects top 3

top_3_efficiency %>%
  kable(format = "latex", row.names = NA,
      booktabs = TRUE, digits = 2,
      caption = "Top 3 Teams: Efficency (Percent Wins/Aggregate Payroll") %>%
  kable_styling(position = "center", latex_options = "HOLD_position")
```

Table 6: Top 3 Teams: Efficency (Percent Wins/Aggregate Payroll

team	efficiency	pct_wins_aggregate	payroll_aggregate
Miami Marlins	0.72	0.48	0.67
Tampa Bay Rays	0.66	0.47	0.71
Oakland Athletics	0.65	0.54	0.84

The Marlins, Rays, and Athletics are the most efficient teams. This is seen in the plot above because all three team are found significantly above the line of best fit, which means they have positive residuals. Therefore, the teams performed better than we would have expected them to given their payroll

3 Model (15 points for correctness; 3 points for presentation)

Finally, we build a predictive model for `pct_wins_aggregate` in terms of `payroll_aggregate` using the aggregate data `mlb_aggregate`.

3.1 Running a linear regression (5 points)

- Run a linear regression of `pct_wins_aggregate` on `payroll_aggregate` and print the regression summary.
- What is the coefficient of `payroll_aggregate`, and what is its interpretation?
- What fraction of the variation in `pct_wins_aggregate` is explained by `payroll_aggregate`?

```r
#creates linear regression that predicts pct_wins from payroll
win_pred_lm <- lm(pct_wins_aggregate ~ payroll_aggregate, mlb_aggregate)
summary(win_pred_lm)
```

```
##
## Call:
## lm(formula = pct_wins_aggregate ~ payroll_aggregate, data = mlb_aggregate)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.04003 -0.01749  0.00094  0.01095  0.07030
##
## Coefficients:
##                   Estimate Std. Error t value Pr(>|t|)
## (Intercept)         0.4226     0.0153   27.56  < 2e-16 ***
## payroll_aggregate   0.0614     0.0117    5.23  1.5e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.027 on 28 degrees of freedom
## Multiple R-squared:  0.494,  Adjusted R-squared:  0.476
## F-statistic: 27.4 on 1 and 28 DF,  p-value: 1.47e-05
```

The coefficient of `payroll_aggregate` is 0.0614. This means for every $1 billion increase in payroll, there is a 6% increase in the percentage of games won.

49.4% of the variation in pct_wins_aggregate can be explained by payroll_aggregate

3.2 Comparing Oakland Athletics to the linear trend (10 points)

- Given their payroll, what is the linear regression prediction for the winning percentage of the Oakland Athletics? What was their actual winning percentage?

```
oakland = mlb_aggregate %>%
  filter(team == "Oakland Athletics") #filters for only Oakland data
#predicts pct_wins using model
pct_wins_pred <-  predict(win_pred_lm, newdata = oakland)[[1]]

print(oakland$pct_wins_aggregate)
```

```
## [1] 0.545
```

```
print(pct_wins_pred)
```

```
## [1] 0.474
```

The Oakland Athletic's actual winning percentage was 54.5%, and the model predicted they would win 47.4% of their games based on their payroll.

- Now run a linear regression of `payroll_aggregate` on `pct_wins_aggregate`. What is the linear regression prediction for the `payroll_aggregate` of the Oakland Athletics? What was their actual payroll?

```
#creates linear regression that predicts payroll from pct_wins
payroll_pct_wins_lm <- lm(payroll_aggregate ~ pct_wins_aggregate, mlb_aggregate)
payroll_pred <- predict(payroll_pct_wins_lm, newdata =oakland)[[1]]

print(oakland$payroll_aggregate)
```

```
## [1] 0.841
```

```
print(payroll_pred)
```

```
## [1] 1.61
```

The Oakland Athletic's actual payroll was $0.841 billion, and the model predicted their payroll was $1.61 billion based on pcts_wins