# STAT 471: Homework 1 Solutions

Eugene Katsevich

Due: September 15, 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:*

*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?

[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`.]

**Solution.**

```
mlb_raw = read_csv("../../data/MLPayData_Total.csv")
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>, ...
```

**We see that the data contain 30 rows and 54 columns. These dimensions match up with the data description given. Indeed, there are 30 teams and one row per team. For each team, there are 3 + 17 + 17 + 17 = 54 features.**

## 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.]

**Solution.**

```
# create tidy aggregate data
mlb_aggregate = mlb_raw %>%
  select(Team.name.2014, payroll, avgwin) %>% # select aggregate columns
  rename(team = Team.name.2014,               # rename columns
         payroll_aggregate = payroll,
         pct_wins_aggregate = avgwin)

mlb_aggregate                                 # print the tibble
```

```
## # A tibble: 30 x 3
##    team                  payroll_aggregate pct_wins_aggregate
##    <chr>                             <dbl>              <dbl>
##  1 Arizona Diamondbacks              1.12               0.490
##  2 Atlanta Braves                    1.38               0.553
##  3 Baltimore Orioles                 1.16               0.454
##  4 Boston Red Sox                    1.97               0.549
##  5 Chicago Cubs                      1.46               0.474
##  6 Chicago White Sox                 1.32               0.511
##  7 Cincinnati Reds                   1.02               0.486
##  8 Cleveland Indians                 0.999              0.496
##  9 Colorado Rockies                  1.03               0.463
## 10 Detroit Tigers                    1.43               0.482
## # ... with 20 more rows
```

```
# create tidy yearly data
mlb_yearly = mlb_raw %>%
  select(-payroll, -avgwin) %>%               # remote aggregate columns
  rename(team = Team.name.2014) %>%           # rename team name column
  pivot_longer(-team,                         # pivot all columns except team
               names_to = "col_name",         #  into a longer format
               values_to = "value") %>%       #  for processing
  separate("col_name",                        # separate column names into a
```

```r
         into = c("prefix",                   #  prefix, year, and suffix
                  "year",
                  "suffix"),
         sep = c(1,5),
         convert = TRUE) %>%
  mutate(tidy_col_name =                      # create new column names based
           case_when(prefix == "p"            #  on prefix and suffix
                     ~ "payroll",
                     prefix == "X" & suffix == ""
                     ~ "num_wins",
                     prefix == "X" & suffix == ".pct"
                     ~ "pct_wins")) %>%
  select(-prefix, -suffix) %>%                # remove prefix and suffix columns
  pivot_wider(names_from = "tidy_col_name",   # pivot the columns back into a
              values_from = "value")          #  wider format

mlb_yearly                                    # print the tibble
```

```
## # A tibble: 510 x 5
##    team                  year payroll num_wins pct_wins
##    <chr>                <int>   <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_aggregate` contains 30 rows, one per team. `mlb_yearly` contains 510 = 30x17 rows, one per team per year.**

### 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.

**Solution.**

```r
# compute aggregate statistics based on yearly data
mlb_aggregate_computed = mlb_yearly %>%
  group_by(team) %>%                       # group by team
  summarise(payroll_aggregate_computed =
              sum(payroll)/1000,           # sum payroll and convert to billions
            pct_wins_aggregate_computed =
              mean(pct_wins))              # average the wins pcts per year

# join the computed and provided aggregate statistics
mlb_aggregate_joined = full_join(mlb_aggregate,
                                 mlb_aggregate_computed,
                                 by = "team")

# plot provided versus computed aggregate payroll
p1 = mlb_aggregate_joined %>%
  ggplot(aes(x = payroll_aggregate_computed,
             y = payroll_aggregate)) +
  geom_point() +                                 # create scatter plot
  geom_abline(slope = 1,                         # add 45 degree line
              color = "red",
              linetype = "dashed") +
  labs(x = "Aggregate payroll (computed)",       # add informative axis titles
       y = "Aggregate payroll (provided)") +
  theme_bw()

# plot provided versus computed aggregate win percentage
p2 = mlb_aggregate_joined %>%
  ggplot(aes(x = pct_wins_aggregate_computed,
             y = pct_wins_aggregate)) +
  geom_point() +                                 # create scatter plot
  geom_abline(slope = 1,                         # add 45 degree line
              color = "red",
              linetype = "dashed") +
  labs(x = "Aggregate win percentage (computed)", # add informative axis titles
       y = "Aggregate win percentage (provided)") +
  theme_bw()

# combine plots
plot_grid(p1, p2)
```

**Figure 1 shows a decent, but imperfect agreement between the provided and computed aggregate quantities. This is an artifact in the data that may warrant further investigation.**

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

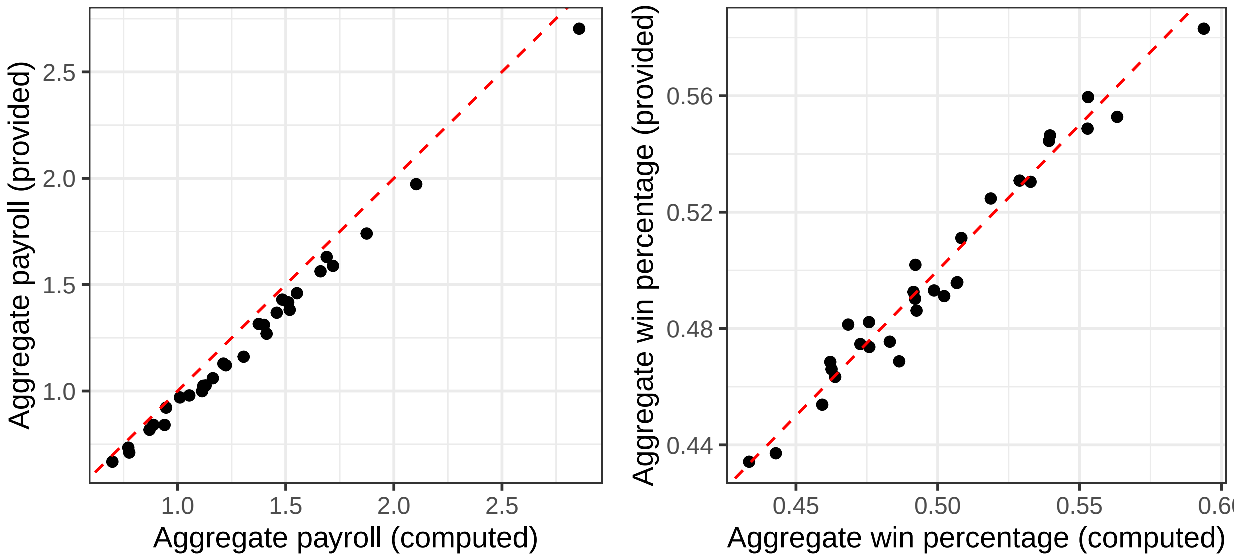


Figure 1: Comparing provided and computed aggregate payroll and win percentages. They are decently but not perfectly aligned.

- 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.
- 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.]

**Solution.**

```
# payroll versus year
mlb_yearly %>%
  ggplot(aes(x = year, y = payroll)) +
  geom_line() +                                         # create line plot
  geom_hline(aes(yintercept =                           # add horizontal line
                   payroll_aggregate_computed*1000/17), # convert to millions
             colour = "red",                            #  and avg. over years
             linetype = "dashed",
             data = mlb_aggregate_computed) +
  facet_wrap(team ~ .) +                                # one panel per team
  labs(x = "Year",                                      # informative titles
       y = "Total payroll (millions)") +
  theme_bw()
```

```
# arrange teams by descending aggregate payroll
mlb_aggregate_computed %>%
  arrange(desc(payroll_aggregate_computed)) %>%
  select(team, payroll_aggregate_computed) %>%
  rename(Team = team,
         `Aggregate payroll` = payroll_aggregate_computed) %>%
```

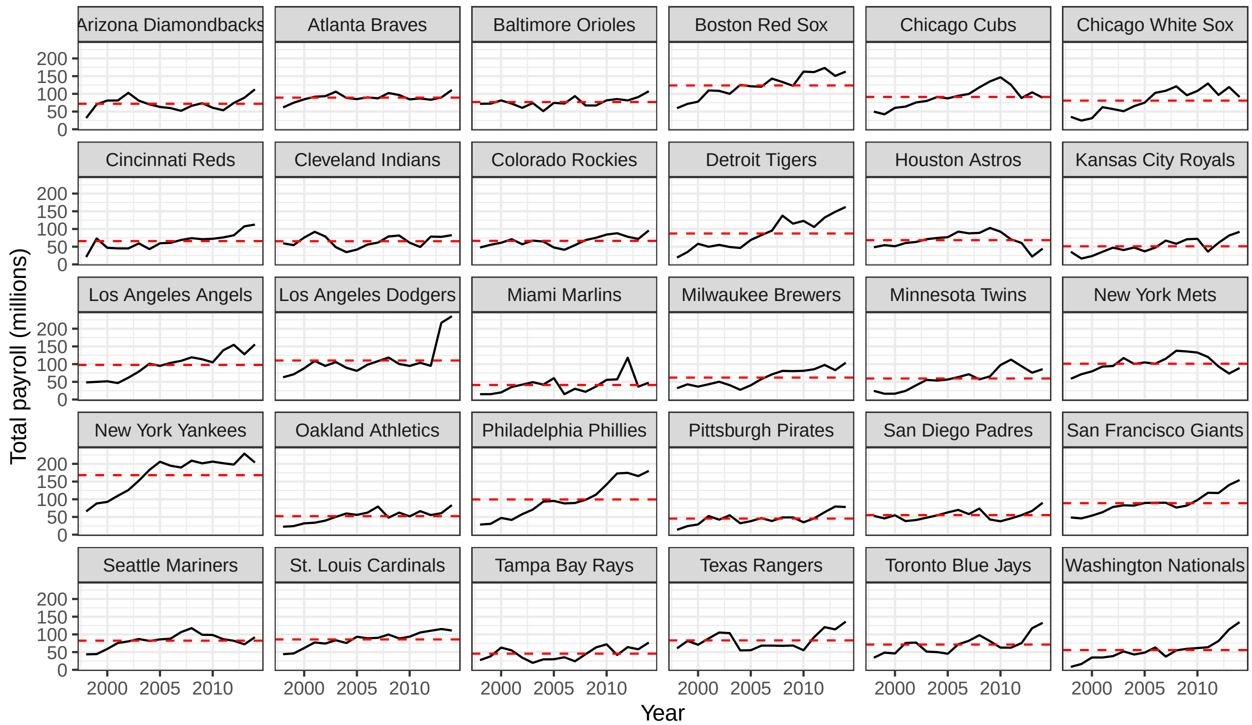


Figure 2: Payroll over time for 30 MLB teams. Red dashed lines denote mean payroll for each team.

Table 1: Top three teams by aggregate payroll (in billions of dollars).

| Team | Aggregate payroll |
|---|---|
| New York Yankees | 2.86 |
| Boston Red Sox | 2.10 |
| Los Angeles Dodgers | 1.87 |

```
  head(3) %>%
  kable(format = "latex", row.names = NA,
        booktabs = TRUE, digits = 2,
        caption = "Top three teams by aggregate payroll
        (in billions of dollars).") %>%
  kable_styling(position = "center")
```

```
# arrange teams by descending percentage increase in payroll
mlb_yearly %>%
  select(team, year, payroll) %>%          # select relevant variables
  pivot_wider(names_prefix = "payroll_",   # pivot so that payrolls are
              names_from = "year",         #  in separate columns per year
              values_from = "payroll") %>%
  mutate(pct_increase =                    # percent increase in payroll
         (payroll_2014 - payroll_1998)/payroll_1998*100) %>%
  select(team,                             # select relevant variables
         payroll_1998,
         payroll_2014,
```

Table 2: Top three teams by payroll increase (payroll indicated in millions of dollars).

| Team | Payroll (1998) | Payroll (2014) | Percent increase |
|---|---|---|---|
| Washington Nationals | 8.32 | 135 | 1520 |
| Detroit Tigers | 19.24 | 162 | 743 |
| Philadelphia Phillies | 28.62 | 180 | 529 |

```
         pct_increase) %>%
  arrange(desc(pct_increase)) %>%           # arrange in decreasing order
  head(3) %>%
  rename(Team = team,
         `Payroll (1998)` = payroll_1998,
         `Payroll (2014)` = payroll_2014,
         `Percent increase` = pct_increase) %>%
  kable(format = "latex", row.names = NA,
        booktabs = TRUE, digits = 2,
        caption = "Top three teams by payroll increase
        (payroll indicated in millions of dollars).") %>%
  kable_styling(position = "center")
```

**Based on Table 1, the three teams with the highest mean payrolls per year are the Yankees, Red Sox, and Dodgers. Based on Table 2, the three teams with the highest increase in payroll across the period of interest are the Nationals, Tigers, and Phillies. The red dashed lines in Figure 2 correspond to the mean payrolls and we see that the Yankees, Red Sox, and Dodgers appear to have the highest red dashed lines. The slopes of the lines connecting the left-most and right-most points correspond to the increase in payroll across the period of interest, and the Nationals, Tigers, and Phillies apear to have the highest slopes.**

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

**Solution.**

```
# win percentage versus year
mlb_yearly %>%
  ggplot(aes(x = year, y = pct_wins)) +
  geom_line() +                                # create line plot
  geom_hline(aes(yintercept =                  # add horizontal line
                   pct_wins_aggregate_computed),
             colour = "red",
             linetype = "dashed",
             data = mlb_aggregate_computed) +
  facet_wrap(team ~ .) +                       # one team per panel
```

Table 3: Top three teams by aggregate win percentage.

| Team | Aggregate win percentage |
|---|---:|
| New York Yankees | 0.59 |
| Atlanta Braves | 0.56 |
| St. Louis Cardinals | 0.55 |

```
  labs(x = "Year",                              # informative axis titles
       y = "Win percentage") +
  theme_bw()
```

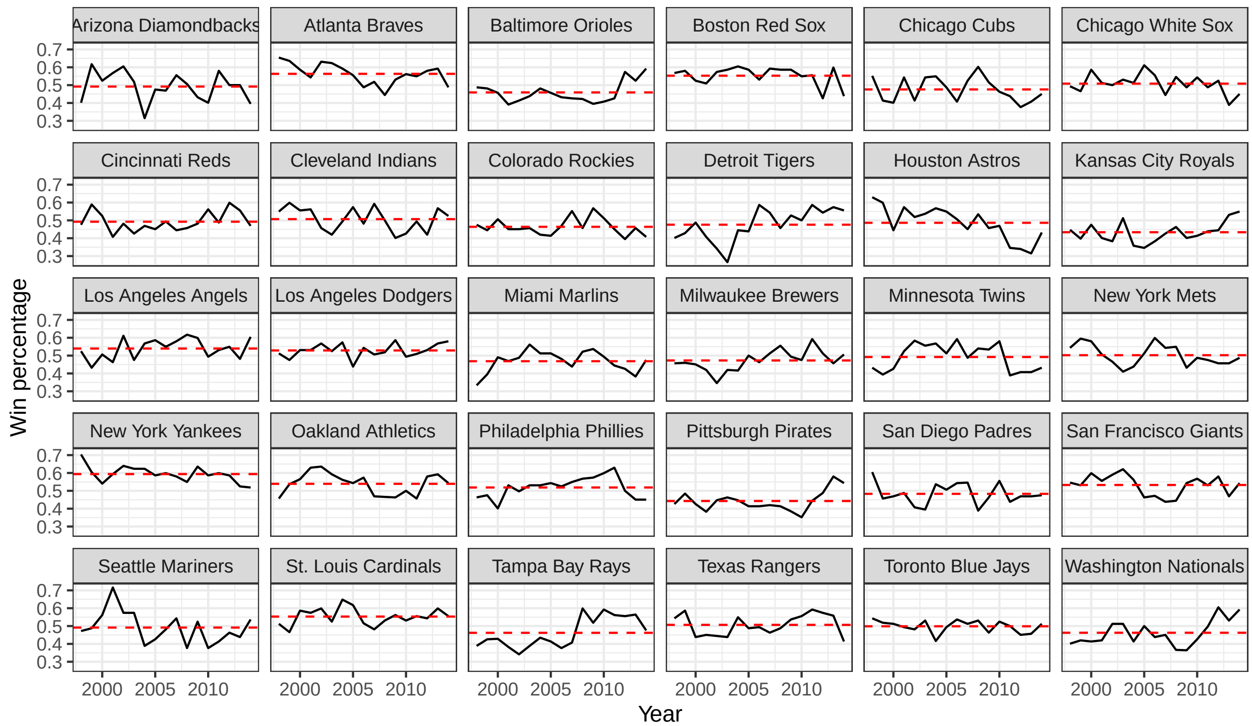


Figure 3: Win percentage over time for 30 MLB teams. Red dashed lines denote mean win percentage for each team.

```
# arrange teams by descending win percentage
mlb_aggregate_computed %>%
  select(team, pct_wins_aggregate_computed) %>%
  arrange(desc(pct_wins_aggregate_computed)) %>%
  rename(Team = team,
         `Aggregate win percentage` = pct_wins_aggregate_computed) %>%
  head(3) %>%
  kable(format = "latex", row.names = NA,
        booktabs = TRUE, digits = 2,
        caption = "Top three teams by aggregate win percentage.") %>%
  kable_styling(position = "center")
```

Table 4: Top three teams by win percentage standard deviation over time.

| Team | Win percentage standard deviation |
|---|---|
| Houston Astros | 0.09 |
| Detroit Tigers | 0.09 |
| Seattle Mariners | 0.09 |

```
# arrange teams in descending order of pct_wins standard deviation
mlb_yearly %>%
  select(team, year, pct_wins) %>%           # select relevant variables
  group_by(team) %>%                         # group by team
  summarise(pct_wins_sd = sd(pct_wins)) %>%  # compute standard deviation
  arrange(desc(pct_wins_sd)) %>%             # arrange by standard deviation
  head(3) %>%
  rename(Team = team,
         `Win percentage standard deviation` = pct_wins_sd) %>%
  kable(format = "latex", row.names = NA,
        booktabs = TRUE, digits = 2,
        caption = "Top three teams by win
        percentage standard deviation over time.") %>%
  kable_styling(position = "center")
```

**Table 3 shows that the three teams with the highest mean win percentage per year are the Yankees, Braves, and Cardinals. Table 4 shows that the three teams with the most erratic win percentage across the period of interest are the Astros, Tigers, and Mariners. Figure 3 produced above supports these conclusions in the sense that the Yankees, Braves, and Cardinals appear to have the highest red dashed lines (corresponding to mean win percentage) and the Astros, Tigers, and Mariners apear to have the highest variation in win percentage across years (corresponding to how erratically a team performs).**

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

**Solution.**

```
mlb_aggregate %>%
  ggplot(aes(x = payroll_aggregate,
             y = pct_wins_aggregate,
             label = team)) +
  geom_point() +                             # create scatter plot
  geom_smooth(method = "lm", se = FALSE) +   # add least squares line
  ggrepel::geom_text_repel() +               # add labels to points
  labs(x = "Aggregate payroll (billions of dollars)",
       y = "Aggregate win percentage") +     # add informative axis titles
  theme_bw()
```

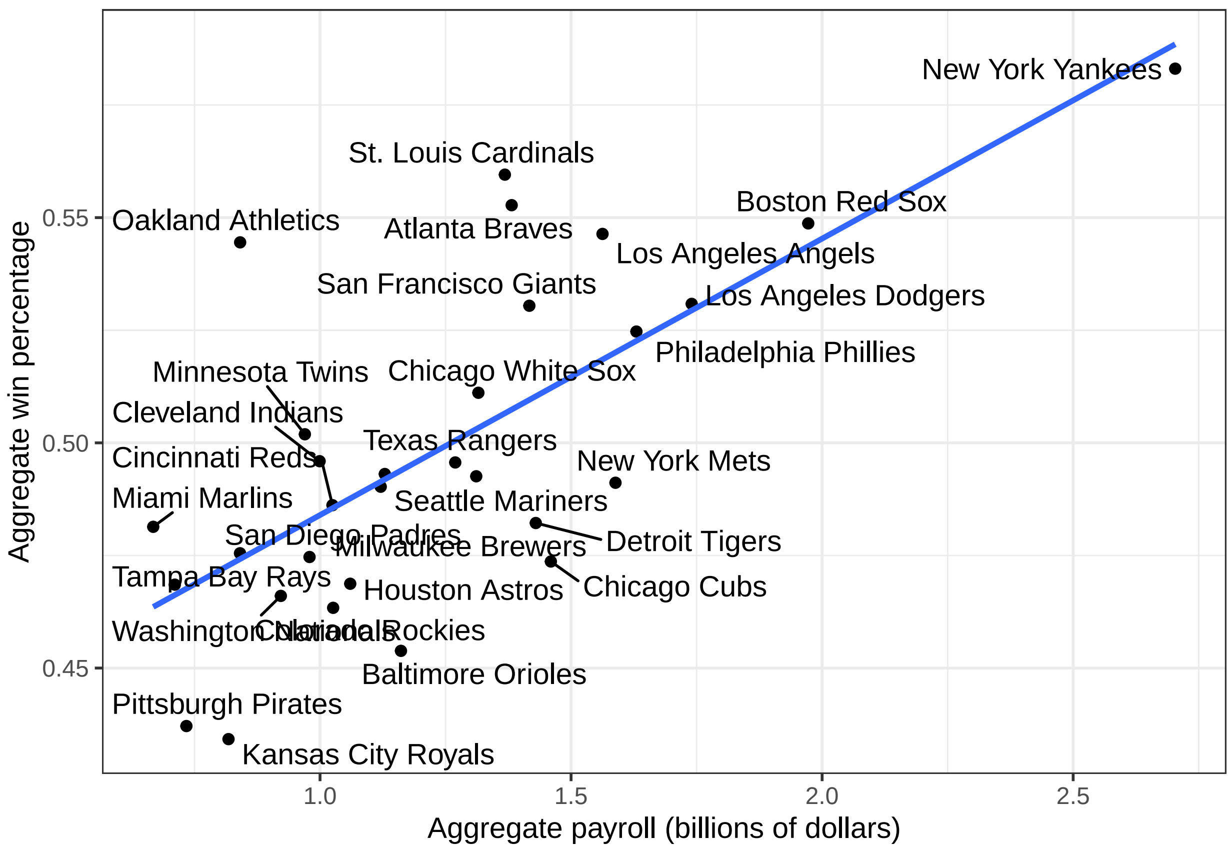


Figure 4: Aggregate win percentage as a function of aggregate payroll. We observe a generally positive relationship between these two variables.

Table 5: Top three teams by efficiency.

| Team | Aggregate payroll | Aggregate win percent | Efficiency |
|---|---|---|---|
| Miami Marlins | 0.67 | 0.48 | 0.72 |
| Tampa Bay Rays | 0.71 | 0.47 | 0.66 |
| Oakland Athletics | 0.84 | 0.54 | 0.65 |

**Based on the shape of the scatter plot and the positive slope of the least squares line in Figure 4, the relationship between `payroll` and `pct_wins` appears positive. This makes sense because better players tend to earn higher salaries.**

### 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."

**Solution.**

```
mlb_aggregate %>%
  mutate(efficiency =             # calculate efficiency
           pct_wins_aggregate/payroll_aggregate) %>%
  arrange(desc(efficiency)) %>%   # arrange by decreasing efficiency
  head(3) %>%
  rename(Team = team,
         `Aggregate payroll` = payroll_aggregate,
         `Aggregate win percent` = pct_wins_aggregate,
         Efficiency = efficiency) %>%
  kable(format = "latex", row.names = NA,
        booktabs = TRUE, digits = 2,
        caption = "Top three teams by efficiency.") %>%
  kable_styling(position = "center")
```

**Based on Table 5, the three most efficient teams are the Marlins, Rays, and Athletics. Figure 4 supports this conclusion in the sense that these three teams have relatively high win percentage and relatively low 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`?

**Solution.**

```
lm_fit = lm(pct_wins_aggregate ~ payroll_aggregate, data = mlb_aggregate)
summary(lm_fit)
```

```
##
## 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
```

**Based on the regression summary, the coefficient of `payroll` is 0.06, which means that a 1 billion increase in aggregate payroll is associated with a 0.06 increase in winning percentage. The $R^2$ is 0.494, so 49.4% of the variation in `pct_wins` is explained by `payroll`.**

### 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?
- 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?

**Solution.**

```
# extract data for Athletics
aggregate_athletics = mlb_aggregate %>%
  filter(team == "Oakland Athletics")
# predict on Athletics using fitted model
pct_wins_prediction = predict(lm_fit,  newdata = aggregate_athletics)
# extract actual win percentage
pct_wins_athletics = aggregate_athletics %>% pull(pct_wins_aggregate)

# print a table with the results
tibble("Predicted win percentage" = pct_wins_prediction,
       "Actual win percentage" = pct_wins_athletics) %>%
  kable(format = "latex", row.names = NA,
        booktabs = TRUE, digits = 2,
        caption = "Predicted versus actual win percentages
        for the Oakland Athletics.") %>%
```

Table 6: Predicted versus actual win percentages for the Oakland Athletics.

| Predicted win percentage | Actual win percentage |
| ---: | ---: |
| 0.47 | 0.54 |

Table 7: Predicted versus actual aggregate payrolls (in billions of dollars) for the Oakland Athletics.

| Predicted payroll | Actual payroll |
| ---: | ---: |
| 1.61 | 0.84 |

```
  kable_styling(position = "center")
```

**Given their payroll, we would have expected the Oakland Athletics to have a winning percentage of 47%, whereas they actually had a winning percentage of 54.5% (Table 6).**

```
# run the reverse regression
lm_fit_reverse = lm(payroll_aggregate ~ pct_wins_aggregate,
                    data = mlb_aggregate)
# predict on Athletics using fitted model
payroll_prediction = predict(lm_fit_reverse,
                             newdata = aggregate_athletics)
# extract actual payroll
payroll_athletics = aggregate_athletics %>% pull(payroll_aggregate)
# print a table with the results
tibble("Predicted payroll" = payroll_prediction,
       "Actual payroll" = payroll_athletics) %>%
  kable(format = "latex", row.names = NA,
        booktabs = TRUE, digits = 2,
        caption = "Predicted versus actual
        aggregate payrolls (in billions of dollars)
        for the Oakland Athletics.") %>%
  kable_styling(position = "center")
```

**Given their winning percentage, we would have expected the Oakland Athletics to spend $1.6 billion, whereas they actually only spent $0.84 billion (Table 7).**