STAT 471: Homework 1

Nico Melton

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:

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.

```
# read data using 'read_csv'
mlb_raw <- read_csv("../../data/MLPayData_Total.csv")
# print data using 'print'
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>, ...
```

This data matches the data description above because it contains 30 MLB teams and has all variables listed above.

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_aggregate` 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.

```r
# use 'select' to select and rename columns `team`, `payroll_aggregate`, `pct_wins_aggregate` from mlb_
mlb_aggregate <- mlb_raw %>% select(team = Team.name.2014, payroll_aggregate = payroll, pct_wins_aggrega

# use 'select' to select and rename team and all yearly payroll, percent wins, and num wins variables f
mlb_yearly <- mlb_raw %>% select(team = Team.name.2014, !c("payroll", "avgwin")) # all vars except payr
# rename all win percentage variable to have prefix pctX and no suffix (to prep for 'pivot_longer')
mlb_yearly <- mlb_yearly %>% rename_with(~ str_sub(paste0("pct", .x), start = 1, end = 8), ends_with(".p
# use 'pivot_longer' to pivot the yearly data in each row to long format for payroll, pct_wins, and num
payroll_yearly <- mlb_yearly %>%
  select(team, matches("p\\d")) %>%
  pivot_longer(!team, names_to = "year", names_prefix = "p", values_to = "payroll")
pct_wins_yearly <- mlb_yearly %>%
  select(team, starts_with("pctX")) %>%
  pivot_longer(!team, names_to = "year", names_prefix = "pctX", values_to = "pct_wins")
num_wins_yearly <- mlb_yearly %>%
  select(team, starts_with("X")) %>%
  pivot_longer(!team, names_to = "year", names_prefix = "X", values_to = "num_wins")
# use 'merge' to combine each long format data set, override mlb_yearly
mlb_yearly <- payroll_yearly %>%
```

```
  merge(pct_wins_yearly, by = c("team", "year"), all = TRUE) %>%
  merge(num_wins_yearly, by = c("team", "year"), all = TRUE)
```

TODO: add description of solution?

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.

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`.
- 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.

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.

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.

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.

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.

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.