Lab 3: Descriptive Statistics

Code

AUTHOR
Maxim Dokukin

PUBLISHED
February 13, 2024

Follow the instructions below and use R Markdown to create a pdf document with your code and answers to the following questions on Gradescope. You may find a template file by clicking "Code" in the top right corner of this page.

Your final submission should clearly include all code needed to generate your answers and should be formatted according to the guidelines outlined in class. In particular, make sure:

1. Code and output are clearly organized by question.
2. Unnecessary messages, warning, and output are removed.

You may collaborate with your classmates and consult external resources, but you should write and submit your own answer. **Any classmates with whom you collaborate should be credited at the top of your submission. Similarly, if you consult any external references, you should cite them clearly and explicitly.**

A. Weather Forecast Data

1. For this lab, we'll be using data on weather forecasts gathered by student at Saint Louis University. You can read about the dataset here. Download the weather forecasts data using the following code:

```
weather_forecasts <- readr::read_csv('https://raw.githubusercon
summary(weather_forecasts)
```

```
      date                 city              state          
high_or_low       
 Min.   :2021-01-30   Length:651968      Length:651968     
Length:651968     
 1st Qu.:2021-05-31   Class :character   Class :character  
Class :character  
 Median :2021-09-30   Mode  :character   Mode  :character  
Mode  :character  
 Mean   :2021-09-30                                        
 3rd Qu.:2022-01-30  
 Max.   :2022-06-01  
                                                                  
 forecast_hours_before observed_temp    forecast_temp    
observed_precip 
 Min.   :12            Min.   :-47.00   Min.   :-41.00   Min.   
: 0.00  
 1st Qu.:21            1st Qu.: 42.00   1st Qu.: 42.00   1st 
Qu.: 0.00  
 Median :30            Median : 59.00   Median : 59.00   Median 
: 0.00  
 Mean   :30            Mean   : 57.56   Mean   : 57.36   Mean   
: 0.10  
 3rd Qu.:39            3rd Qu.: 74.00   3rd Qu.: 74.00   3rd 
Qu.: 0.02  
 Max.   :48            Max.   :122.00   Max.   :118.00   Max.   
:12.40  
                       NA's   :47744    NA's   :37313    NA's   
:50416  
 forecast_outlook   possible_error    
 Length:651968      Length:651968     
 Class :character   Class :character  
 Mode  :character   Mode  :character  
```

2. How many rows are in this dataset? How many columns?

```r
dim(weather_forecasts)
```

```
[1] 651968     10
```

651968 rows, 10 clos.

3. How many cities are represented in this dataset?

```r
length(unique(weather_forecasts$city))
```

```
[1] 160
```

160 cities = size of array with unique cities.

4. Create a new data frame containing only the forecasts for San Jose. You may have to explore the values for the `city` variable.

```
san_jose_data = weather_forecasts[weather_forecasts$city == 'SAN
```

15 min wasted on realizing its san_jose, not san jose.

5. Compute the mean absolute error between `observed_temp` and `forecast_temp` for San Jose.

```
san_jose_data$absolute_error <- abs(san_jose_data$observed_temp
                                    san_jose_data$forecast_temp
mean_error <- mean(san_jose_data$absolute_error, na.rm = TRUE)
print(mean_error)
```

```
[1] 2.169762
```

2.169762

6. Compute the mean absolute error between `observed_temp` and `forecast_temp` for San Jose using only forecasts made 48 hours in advance.

```
san_jose_48adv_data = san_jose_data[san_jose_data$forecast_hours

san_jose_48adv_data$absolute_error <- abs(san_jose_48adv_data$ob
                                          san_jose_48adv_data$
mean_error_48 <- mean(san_jose_48adv_data$absolute_error, na.rm
print(mean_error_48)
```

```
[1] 2.262544
```

2.262544

7. Compute the mean absolute error between `observed_temp` and `forecast_temp` for San Jose using only forecasts made 12 hours in advance.

```
san_jose_12adv_data = san_jose_data[san_jose_data$forecast_hours

san_jose_12adv_data$absolute_error <- abs(san_jose_12adv_data$ob
                                          san_jose_12adv_data$
mean_error_12 <- mean(san_jose_12adv_data$absolute_error, na.rm
print(mean_error_12)
print(mean_error_12)
```

```
[1] 2.0553
```

2.0553

8. Compare your answers to 6 and 7. What do you notice? How does this compare to your expectation?

24h advance forecasts have smaller error. This makes sense, as it is easier to make more accurate predictions in the short term.

9. Pick two cities in this dataset. Investigate whether the forecast accuracy is better for one city than for the other, using an appropriate statistic. Discuss your findings.

```
nyc_data = weather_forecasts[weather_forecasts$city == 'NEW_YOR
nyc_data$absolute_error <- abs(nyc_data$observed_temp - nyc_dat
nyc_mean_error <- mean(nyc_data$absolute_error, na.rm = TRUE)
print(nyc_mean_error)
```

```
[1] 2.182927
```

```
gf_data = weather_forecasts[weather_forecasts$city == 'GREAT_FA
gf_data$absolute_error <- abs(gf_data$observed_temp - gf_data$f
gf_mean_error <- mean(gf_data$absolute_error, na.rm = TRUE)
print(gf_mean_error)
```

```
[1] 3.05578
```

Great Falls accuracy is significantly lower than in NYC. Maybe there are more variables that effect weather there? Yet, the best advice to avoid rain is the same, ‘Take the umbrella’

B. Find your own data

For this component, pick a Tidy Tuesday dataset and complete the following activity.

10. Provide a brief description of your dataset. Identify at least two questions you could try to answer using this dataset.

```
library(tidyverse)

big_tech_stock_prices <- readr::read_csv('https://raw.githubuser

summary(big_tech_stock_prices)
```

```
 stock_symbol            date                 open
high
 Length:45088       Min.   :2010-01-04   Min.   :  1.076   Min.
:  1.109
 Class :character   1st Qu.:2013-05-30   1st Qu.: 25.670   1st
Qu.: 25.930
 Mode  :character   Median :2016-08-09   Median : 47.930
Median : 48.460
                    Mean   :2016-08-03   Mean   : 89.267   Mean
: 90.370
                    3rd Qu.:2019-10-21   3rd Qu.:128.662   3rd
Qu.:129.849
                    Max.   :2023-01-24   Max.   :696.280   Max.
:700.990
      low                close           adj_close
volume
 Min.   :  0.9987   Min.   :  1.053   Min.   :  1.053   Min.
:5.892e+05
 1st Qu.: 25.3600   1st Qu.: 25.660   1st Qu.: 22.076   1st
Qu.:9.629e+06
 Median : 47.4650   Median : 47.970   Median : 45.377   Median
:2.646e+07
 Mean   : 88.1119   Mean   : 89.271   Mean   : 85.210   Mean
:5.298e+07
 3rd Qu.:127.2539   3rd Qu.:128.641   3rd Qu.:113.672   3rd
Qu.:5.840e+07
 Max.   :686.0900   Max.   :691.690   Max.   :691.690   Max.
:1.881e+09
```

Stock prices from 2010 to 2023 for big tech.

- What was the most growing stock from Jan 10 2014 to Jan 10 2022 (%)?
- What was the least growing stock from Jan 10 2012 to Jan 10 2022 (%)?
- What is the average (median) growth from Jan 10 2014 to Jan 10

2022 (%)?

PS I was trying Jan 1st, but realized stocks dont trade this day because of the holiday. Jan 10 2015 had no data either. Meta has not been on data set in 2010… had to adjust my date range to make sure I have all the numbers.

11. Open your dataset in R and compute one or more descriptive statistics that shed light on your questions. Discuss your findings.

```
companies <- unique(big_tech_stock_prices$stock_symbol)
prices_2014 <- big_tech_stock_prices[big_tech_stock_prices$date
prices_2022 <- big_tech_stock_prices[big_tech_stock_prices$date

prices_change <- data.frame(
  companies,
  prices_2014$close,
  prices_2022$close
)

prices_change$percent_change = (prices_change$prices_2022.close
                                   prices_change$prices_2014.cl
                                   prices_change$prices_2014.cl
colnames(prices_change) <- c('company', 'price_2014', 'price_20
```

Best performer 2014-2022

```
prices_change[which.max(prices_change$percent_change), ][1,]
```

```
   company price_2014 price_2022 percent_change
12    NVDA     3.9325        274       6867.578
```

Worst performer 2014-2022

```
prices_change[which.min(prices_change$percent_change), ]
```

```
  company price_2014 price_2022 percent_change
7     IBM   179.0249     135.03      -24.57472
```

Median growth rate 2014 - 2022

```
median(prices_change$percent_change)
```

```
[1] 589.2025
```

12. Are there any limitations of your analysis? Could additional data or more complicated methods improve your analysis? Discuss.

I wanted to do Jan 1 2010 to Jan 1 2020. But, for Jan 1 there is no data. META also had no data for the early years. I had to adjust my date range to fit these limitations. Using the last value in the previous year before Jan 1, could have made possible data analysis in the range Jan 1, YYYY - Jan 1, YYYY