Ordinary Least-Squares Estimation

Contents

Pre-Requisite Knowledge

This lab assignment includes practical questions involving estimation in simple linear regression via ordinary least-squares. All its content is in a frequentist framework. Furthermore, this assignment would be the introductory practicum in the course **DSCI 561 (Regression I)**.

Besides the sample lecture, students specifically need to be familiar with the following courses and topics:

- **DSCI 551 (Descriptive Statistics and Probability for Data Science)**. Random variables, expected values (and their properties), and normality.
- **DSCI 552 (Statistical Inference and Computation I)**. Estimators, sampling distributions, hypothesis testing, and confidence intervals.
- **DSCI 531 (Data Visualization I)**. Data visualization via the package `ggplot2`.

Lab Assignment Settings

This assignment has the following characteristics:

- It is expected to be submitted as an `R` markdown along with its corresponding PDF file.
- The handout incorporates auto-graded items for instantaneous feedback. These items are built using Otter Grader.
- Moreover, the practicum is designed to be submitted to the online grading platform Gradescope.

Specific Learning Objectives

By the end of this lab assignment, students are expected to attain the following:

- Define linear regression models.
- Estimate their terms using `R` via a sample and interpret them.

Lab Assignment

This lab assignment will allow you to define and conceptualize the simple regression model via a practical case. Moreover, you will get familiar with the process of a typical statistical model involving (but not limited to!) a main statistical inquiry, exploratory data analysis (EDA), mathematical modelling, estimation, and data storytelling.

Setup

To solve this assignment, you need to load the packages below. If you fail to load any of them, you can install them and rerun the cell.

```
library(tidyverse)
library(broom)
library(tree)
library(digest)
library(testthat)
```

Rubric

This assignment is worth **17 points in total** plus 2 bonus points if you solve the optional question. Most of the questions are **auto-graded**. **Thus, you will need to pass all the corresponding auto-grading tests to get their full marks.** The rest of the points belong to **reasoning**.

The Facebook Dataset

It is time to explore another engaging dataset, such as the Spotify data we covered in our lecture. This time we will work with Facebook data. In their work related to data mining for predicting performance metrics of posts on Facebook pages linked to brands, Moro et al. (2016) provide a dataset related to post metrics on Facebook user engagement. This engagement data comes from 2014 on a Facebook page of a famous cosmetics brand. The original dataset has 500 observations, each belonging to specific classes of page posts. You can find the raw dataset in data.world. The `CSV` file to be used in this assignment is a modified version of this dataset with only 491 observations.

Moreover, it is essential to clarify that the raw dataset has 17 different continuous and discrete variables. Nevertheless, for this assignment, let us narrow them down to the following:

1. The continuous variable `total_engagement_percentage` is a **key variable** for any brand with a Facebook page. It tells us how engaged the Facebook users are with the company's posts, **regardless of whether they previously liked their Facebook page or not**. *The larger the percentage, the better the total engagement.* We compute it as follows:

$$\texttt{total\_engagement\_percentage} = \frac{\text{Lifetime Engaged Users}}{\text{Lifetime Post Total Reach}} \times 100\%$$

- **Lifetime Post Total Reach:** The number of overall *Facebook unique users* who *saw* the post.
- **Lifetime Engaged Users:** The number of overall *Facebook unique users* who *saw and clicked* on the post. This count is a subset of **Lifetime Post Total Reach**.

2. The continuous `share_percentage` is the percentage that the number of *shares* represents from the sum of *likes*, *comments*, and *shares* in each post. It is computed as follows:

$$\texttt{share\_percentage} = \frac{\text{Number of Shares}}{\text{Total Post Interactions}} \times 100\%$$

- **Total Post Interactions:** The sum of *likes*, *comments*, and *shares* in a given post.
- **Number of Shares:** The number of *shares* in a given post. This count is a subset of *Total Post Interactions.*

1. Main Statistical Inquiry

rubric={reasoning:2}

Suppose you are the sales manager of the cosmetics brand; you are interested in the following:

> Is the **mean** total engagement percentage dependent on the share percentage on our Facebook page? If so, by how much?

Suppose you want to use simple linear regression (SLR) to answer this inquiry. Hence, answer the following:

1. **In one sentence**, what would be the model's response?

Type your answer here, replacing this text.

2. **In one sentence**, what would be the model's regressor?

Type your answer here, replacing this text.

2. Loading Data

Now, let us load the data.

```
facebook_data <- read_csv("data/facebook_data.csv")
head(facebook_data)
```

```
## # A tibble: 6 x 2
##   total_engagement_percentage share_percentage
##                         <dbl>            <dbl>
## 1                        6.47            17
## 2                       13.9             17.7
## 3                        7.34            17.5
## 4                        4.41             8.27
## 5                        9.26            12.5
## 6                       11.4             17.7
```

3. Exploratory Data Analysis (EDA)

rubric={autograde:3,reasoning:2}

Using the variables of interest stored in `facebook_data`, create the **proper visualization** with the regressor and response on the x and y-axes, respectively. Add appropriate axes labels and titles. Store the plot in the variable `facebook_plot`.

Furthermore, **comment on your findings in one to two sentences**.

Type your answer here, replacing this text.

```
facebook_plot <- NULL

# YOUR CODE HERE

facebook_plot
```

```
## NULL
```

```
. = ottr::check("tests/Q3.R")
```

```
## Test Q3 - 1 passed
##
##
## Test Q3 - 2 failed:
## the variable used for the x-axis is incorrect
## "share_percentage" == rlang::get_expr(properties$x) is not TRUE
##
##   `actual`:
##   `expected`: TRUE
##
## Test Q3 - 3 failed:
## the variable used for the y-axis is incorrect
## "total_engagement_percentage" == rlang::get_expr(properties$y) is not TRUE
##
```

```
##   ‘actual‘:
##   ‘expected‘: TRUE
##
## Test Q3 - 4 failed:
## the plot type is incorrect
## "GeomPoint" %in% class(facebook_plot$layers[[1]]$geom) is not TRUE
##
##   ‘actual‘:   FALSE
##   ‘expected‘: TRUE
##
## Test Q3 - 5 failed:
## you should use a human-readable name for the x-axis label
## (facebook_plot$labels$x) == "share_percentage" is not FALSE
##
##   ‘actual‘:
##   ‘expected‘: FALSE
##
## Test Q3 - 6 failed:
## you should use a human-readable name for the y-axis label
## (facebook_plot$labels$y) == "total_engagement_percentage" is not FALSE
##
##   ‘actual‘:
##   ‘expected‘: FALSE
##
## Test Q3 - 7 failed:
## your plot should have a title
## is.null(facebook_plot$labels$title) is not FALSE
##
##   ‘actual‘:   TRUE
##   ‘expected‘: FALSE
```

4. Data Modelling

Once we have our EDA, let us proceed to the SLR modelling. Our training set (i.e., `facebook_data`) has a size of $n = 491$. For the ith observation in our training set ($i = 1, \ldots, 491$), the regression equation is the following:

$$\underbrace{Y_i}_{\text{Response}} = \underbrace{\beta_0 + \beta_1 X_i}_{\text{Systematic Component}} + \underbrace{\varepsilon_i}_{\text{Random Component}}$$

Recall that X_i is the regressor, whereas β_0 and β_1 are the unknown regression intercept and coefficient, respectively.

5. Estimation

Now, let us start with the model estimation. We will break down this stage into four questions.

5.1 rubric={autograde:2}

As seen during lecture time, we will use ordinary least-squares (OLS) estimation to obtain $\hat{\beta}_0$ and $\hat{\beta}_1$:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$

Using the response and regressor from the training set `facebook_data`, compute $\hat{\beta}_0$ and $\hat{\beta}_1$ via the formulas above **by hand**. Bind your results to the **numeric vector-type** variable `beta_0_hat` for $\hat{\beta}_0$ and `beta_1_hat` for $\hat{\beta}_1$.

```
beta_0_hat <- NULL
beta_1_hat <- NULL

# YOUR CODE HERE

beta_0_hat
```

```
## NULL
```

```
beta_1_hat
```

```
## NULL
```

```
. = ottr::check("tests/Q5.1.R")
```

```
## Test Q5.1 - 1 failed:
## beta_0_hat should be vector and numeric
## "numeric" %in% class(beta_0_hat) is not TRUE
##
##   `actual`:   FALSE
##   `expected`: TRUE
##
## Test Q5.1 - 2 failed:
## beta_1_hat should be vector and numeric
## "numeric" %in% class(beta_1_hat) is not TRUE
##
##   `actual`:   FALSE
##   `expected`: TRUE
##
## Test Q5.1 - 3 failed:
## beta_0_hat computation is wrong
## non-numeric argument to mathematical function
##
## Test Q5.1 - 4 failed:
## beta_1_hat computation is wrong
## non-numeric argument to mathematical function
```

5.2 rubric={autograde:2}

Now, using `lm()` with `facebook_data`, estimate a SLR called `facebook_SLR` to help determine the association of share percentage and page engagement percentage.

```r
facebook_SLR <- NULL

# YOUR CODE HERE

facebook_SLR
```

```
## NULL
```

```r
. = ottr::check("tests/Q5.2.R")
```

```
## Test Q5.2 - 1 failed:
## the correct fitting function is not being used
## "lm" %in% class(facebook_SLR) is not TRUE
##
##   `actual`:   FALSE
##   `expected`: TRUE
##
## Test Q5.2 - 2 failed:
## check the formula and data arguments in the fitting function
## digest(round(sum(facebook_SLR$coefficients), 2)) not equal to "a9f0cb4905810fd503591e0deb301798".
##   1/1 mismatches
##   x[1]: "908d1fd10b357ed0ceaaec823abf81bc"
##   y[1]: "a9f0cb4905810fd503591e0deb301798"
```

5.3 rubric={autograde:1}

Use `tidy()` from the `broom` package to obtain the estimated coefficients of `facebook_SLR`. Bind your results to the variable `tidy_SLR`. Your model estimates have to be equal to the manual computations `beta_0_hat` and `beta_1_hat`.

```r
tidy_SLR <- NULL

# YOUR CODE HERE

tidy_SLR
```

```
## NULL
```

```r
. = ottr::check("tests/Q5.3.R")
```

```
## Test Q5.3 - 1 failed:
## tidy_SLR should be a data frame
## "data.frame" %in% class(tidy_SLR) is not TRUE
##
##   `actual`:   FALSE
##   `expected`: TRUE
##
## Test Q5.3 - 2 failed:
## tidy_SLR does not have the right estimates
## digest(round(sum(tidy_SLR$estimate), 2)) not equal to "a9f0cb4905810fd503591e0deb301798".
##   1/1 mismatches
##   x[1]: "908d1fd10b357ed0ceaaec823abf81bc"
##   y[1]: "a9f0cb4905810fd503591e0deb301798"
```

5.4 rubric={autograde:1}

Compute the corresponding sum of squared residuals (SSR). You will need to use the training set `facebook_data` along with `beta_0_hat` and `beta_1_hat`.

$$S(\hat{\beta}_0, \hat{\beta}_1) = \sum_{i=1}^{n} \left[y_i - (\hat{\beta}_0 + \hat{\beta}_1 x_i) \right]^2$$

Bind your results to the **numeric vector-type** variable `SSR_facebook`.

```
SSR_facebook <- NULL

# YOUR CODE HERE

SSR_facebook
```

```
## NULL
```

```
. = ottr::check("tests/Q5.4.R")
```

```
## Test Q5.4 - 1 failed:
## SSR_facebook should be vector and numeric
## "numeric" %in% class(SSR_facebook) is not TRUE
##
##   `actual`:   FALSE
##   `expected`: TRUE
##
## Test Q5.4 - 2 failed:
## SSR_facebook computation is wrong
## non-numeric argument to mathematical function
```

(Optional) 5.5 rubric={autograde:2}

There is a way to automatically plot the estimated OLS regression line via `ggplot2`. Do it on top of `facebook_scatterplot`.

```
# YOUR CODE HERE

facebook_plot
```

```
## NULL
```

```
. = ottr::check("tests/Q5.5.R")
```

```
## Warning in formals(fun): argument is not a function
```

```
## Test Q5.5 - 1 failed:
## geom_smooth is missing
## "GeomSmooth" %in% class(facebook_plot$layers[[2]]$geom) is not TRUE
##
##   `actual`:   FALSE
```

```
##   ‘expected‘: TRUE
##
## Test Q5.5 - 2 failed:
## incorrect method in geom_smooth
## digest(tolower(method)) not equal to "0ebfb0ddc1a5ced965136ef1538883c6".
##   1/1 mismatches
##   x[1]: "5152ac13bdd09110d9ee9c169a3d9237"
##   y[1]: "0ebfb0ddc1a5ced965136ef1538883c6"
```

6. Conclusion

rubric={reasoning:2}

Run the cell below before continuing.

```
tidy_SLR
```

```
## NULL
```

Use this output to answer the main statistical inquiry **in one or two sentences**. Recall the following:

> Is the **mean** total engagement percentage dependent on the share percentage on our Facebook page? If so, by how much?

Type your answer here, replacing this text.

Submission

You are done with the assignment. Follow these final instructions:

- Knit the assignment to generate the `PDF` file.
- Submit both the `Rmd` AND the `PDF` files to Gradescope.

Reference

Moro, S., Rita, P., & Vala, B. (2016). Predicting social media performance metrics and evaluation of the impact on brand building: A data mining approach. Journal of Business Research, 69(9), 3341-3351.