Effect of Errors in Analysis*

Qi Er (Emma) Teng

February 27, 2024

This essay explores the impact of data generation and cleaning errors on statistical analysis. We simulate an instrument memory error leading to observation overwriting and a data processing mistake that alters numerical values. The study assesses the effects of these errors on the reliability of linear regression outcomes.MOreover, the essay concludes with recommendations for error detection and prevention, aiming to enhance the integrity of empirical research.

Table of contents

Introduction

In this essay, we delve into the implications of data collection and cleaning errors on statistical analysis through a simulated examination of two specific errors. We generate an instrument

*Code is available at https://github.com/dwz92/Effect-of-Errors-in-Analysis

memory limitation leading to data overwriting, and two alterations during data cleaning that modify data values. These simulations are designed to simulate the potential distortions these errors can cause in data interpretation, particularly within the framework of linear regression. By investigating these disruptions, we aim to highlight the necessity of stringent data verification procedures to ensure the accuracy and integrity of statistical outcomes.

Data Simulation

In this section, we will simulate several errors in the data generation process to create a dataset of errors using R(R Core Team 2020), and R libraries of dplyr(Wickham et al. 2023), and tidyverse(Wickham et al. 2019).

Data Generation

Allow that the true data generating process is a Normal distribution with mean of one, and standard deviation of 1. We obtain a sample of 1,000 observations using some instrument.

```
generated_data <- rnorm(1000, mean = 1, sd = 1)
```

First Situation

The first situation we will simulate is an instrument error:

Unknown to us, the instrument has a mistake in it, which means that it has a maximum memory of 900 observations, and begins over-writing at that point, so the final 100 observations are actually a repeat of the first 100.

```
situation1 <- generated_data
tail(situation1)
```

```
[1] 3.00294100 1.12703654 3.13209993 0.03567124 0.78255041 0.98782680
```

```
situation1[901:1000] <- situation1[1:100]
tail(situation1)
```

```
[1] 1.4538217 2.3429203 0.1719302 2.5596646 1.1211245 2.6384068
```

Second Situation

The second situation we will simulate is a data error:

`We employ a research assistant to clean and prepare the dataset. During the process of doing this, unknown to us, they accidentally change half of the negative draws to be positive.`

```
situation2 <- situation1
head(situation2)
```

```
[1]  1.433187  1.818221 -1.027639  1.690692  0.653155  3.271659
```

```
situation2[which(situation2 < 0)[1:(0.5*length(which(situation2 < 0)))]] <- abs(situation2

head(situation2)
```

```
[1] 1.433187 1.818221 1.027639 1.690692 0.653155 3.271659
```

Third Situation

The third situation we will simulate is another data error:

`They additionally, accidentally, change the decimal place on any value between 1 and 1.1, so that, for instance 1 becomes 0.1, and 1.1 would become 0.11.`

```
situation3 <- situation2
head(situation3)
```

```
[1] 1.433187 1.818221 1.027639 1.690692 0.653155 3.271659
```

```
situation3[1 <= situation3 & situation3 <= 1.1] <- (0.1*situation3[1 <= situation3 & situa

head(situation3)
```

```
[1] 1.4331870 1.8182211 0.1027639 1.6906920 0.6531550 3.2716587
```

Analysis

In this section, we will study the mean of the true data generating process in the last situation:

`You finally get the cleaned dataset and are interested in understanding whether the mean of the true data generating process is greater than 0.`

We will start with finding the true mean of the data.

```
error_mean <- mean(situation3)
error_mean
```

```
[1] 1.042586
```

With our newly calculated mean, we will conduct a t-test to get the test statistic for our mean. Note that we are conducting the test to compare the true mean with the null mean of 0.

```
t_stat <- t.test(situation3, mu=0, alternative="greater")

t_stat
```

```
    One Sample t-test

data:  situation3
t = 35.95, df = 999, p-value < 2.2e-16
alternative hypothesis: true mean is greater than 0
95 percent confidence interval:
 0.9948394       Inf
sample estimates:
mean of x
 1.042586
```

From the result of the test, we can notice that the null mean of 0 should be rejected intuitively. This is because the the p-value is smaller than 0.05, which indicates the true mean is significantly greater than 0. Hence, we can conclude the mean of the true data generating process is in fact greater than 0.

Conclusion

The effect of these errors in the data results in a corrupted dataset. During the first error, there will be clear data repetition and compromised inferential statistics from the reduce in actual length of dataset.

The second and third errors creates a skewed distribution and misleading data precision. These effect are from the sign change and right shift of decimal place.

To prevent or notice such complicated errors, we could visualize the data to detect unexpected skew that could possibly be sign changes in the dataset. Moreover, it is important to check descriptive statistics such as median and min to make sure there aren't drastic and unexpected changes in the dataset.

These procedures may not prevent errors entirely, but are effective to notice distorted change in the dataset.

References

R Core Team. 2020. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley, Mara Averick, Jennifer Bryan, Winston Chang, Lucy D'Agostino McGowan, Romain François, Garrett Grolemund, et al. 2019. "Welcome to the tidyverse." *Journal of Open Source Software* 4 (43): 1686. https://doi.org/10.21105/joss.01686.

Wickham, Hadley, Romain François, Lionel Henry, Kirill Müller, and Davis Vaughan. 2023. *Dplyr: A Grammar of Data Manipulation*. https://dplyr.tidyverse.org.