Data Analysis

by Sai Sreekar

03-06-2022

**This R-markdown-file documents all data analysis done in R for the case study**

## Load packages

For basic data wrangling, manipulation and plotting (via ggplot2) we install the tidyverse package that itself contains a lot of useful packages.The stargazer package enables us to give neat tables as output for functions on our data frames. It is especially useful for regression analysis output tables. You will need to install the tidyverse and stargazer packages manually via install.packages(“tidyverse”) and install.packages(“stargazer”), since doing so through a knitr document causes issues.

library(tidyverse)

## Attaching packages tidyverse 1.3.1

## v ggplot2 3.3.5 v purrr 0.3.4 ## v tibble 3.1.4 v dplyr 1.0.7 ## v tidyr 1.1.3 v stringr 1.4.0 ## v readr 2.0.1 v forcats 0.5.1

## Conflicts tidyverse\_conflicts()

## x dplyr::filter() masks stats::filter() ## x dplyr::lag() masks stats::lag()

library(stargazer)

## Check and set current working directory

If you want run this markdown file, it is important that you change the absolute file path to wherever you saved the data files!

## Importing and first look at data

df <- read.csv2("core\_data\_v3.csv")

Stargazer gives us nice output:

stargazer(df[c("time\_sec",

"team\_size",

"item\_weight\_kg", "item\_volume\_cc", "palette\_quantity", "palette\_number", "item\_time\_sec")],

type = "text",

title = "Descriptive Statistics", out = "table1.txt",

covariate.labels = c("Time per Palette (seconds)",

"Team Size",

"Item Weight (kilograms)",

"Item Volume (cubic centimeters)", "Item Quantity per Palette", "Number of Palettes",

"Time per Item (seconds)"))

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ## | Time per Palette (seconds) | 199 | 147.930 | 69.676 | 44 | 102 | 177.5 | 580 |
| ## | Team Size | 199 | 5.156 | 0.587 | 3 | 5 | 5 | 6 |
| ## | Item Weight (kilograms) | 199 | 6.633 | 2.129 | 3.040 | 4.350 | 8.840 | 8.840 |
| ## | Item Volume (cubic centimeters) | 199 | 59,942.750 | 18,573.130 | 30,723 | 41,538 | 85,800 | 85,800 |
| ## | Item Quantity per Palette | 199 | 23.668 | 8.184 | 14 | 14 | 24 | 50 |
| ## | Number of Palettes | 199 | 23.503 | 14.302 | 1 | 11.5 | 34 | 59 |
| ## | Time per Item (seconds) | 199 | 6.658 | 3.465 | 3.000 | 4.310 | 7.857 | 30.500 |

##

|  |  |
| --- | --- |
| ## |  |
| ## | Descriptive Statistics |
| ## | ========================================================================================= |
| ## | Statistic N Mean St. Dev. Min Pctl(25) Pctl(75) Max |
| ## |  |

Here is a first visualization for the variable of interest “time\_sec” as a histogram:

histo\_df1 <- df %>% ggplot() +

geom\_histogram(mapping = aes(x = time\_sec, fill = "red")) + guides(fill="none") +

labs(x = "Time per Palette \n(seconds)",

title = "How long does it take to fully stack a palette?",

subtitle = "distribution of palette completion times via histogramm", caption = "source: own data")

histo\_df1

# How long does it take to fully stack a palette?

### distribution of palette completion times via histogramm

30

20

count

10

0

200

### Time per Palette (seconds)

400

600

source: own data

**Variables “Timer per Palette (seconds)” and “Team Size” in relation as box plot, sample sizes annotated**

We define a function that returns sample sizes which will be annotated to the visualizations:

sample\_size\_finder <- **function**(variable, value){ df %>%

filter(.data[[variable]] == value) %>% count()

}

We build some string variables that will help us with the annotations to the plot:

str\_0<- "n ="

ss\_team\_a = sample\_size\_finder("team\_size", 3) ss\_team\_b = sample\_size\_finder("team\_size", 4) ss\_team\_c = sample\_size\_finder("team\_size", 5) ss\_team\_d = sample\_size\_finder("team\_size", 6)

Now we can construct the box plot:

box\_df1 <- df %>% ggplot() +

geom\_boxplot(mapping = aes(group = team\_size, x = team\_size, y = time\_sec)) + scale\_x\_discrete(limits = c(3, 4, 5, 6)) +

labs(x = "Team Size",

y = "Time per Palette \n(seconds)",

title = "How does the team size impact palette completion time?",

subtitle = "comparison via box plot \nn = sample size", caption = "source: own data") +

annotate("text", x = 3, y = 75, label = paste(str\_0, ss\_team\_a)) + annotate("text", x = 4, y = 50, label = paste(str\_0, ss\_team\_b)) + annotate("text", x = 5, y = 25, label = paste(str\_0, ss\_team\_c)) + annotate("text", x = 6, y = 1, label = paste(str\_0, ss\_team\_d))

box\_df1

# How does the team size impact palette completion time?

### comparison via box plot n = sample size

600

n = 3

n = 12

n = 135

n = 49

400

Time per Palette (seconds)

200

0

3 4 5 6

### Team Size

source: own data

**Show means of “Time per Palette (seconds)” for different values of “Team Sizes”**

We define a function that returns number of observations for any variable at any value:

add\_observations\_time\_sec <- **function**(variable, value){ df %>%

filter(.data[[variable]] == value) %>% select(time\_sec) %>%

sum()

}

Let’s gather the number of observations for given values for team sizes:

time\_sec\_team\_size\_3 = add\_observations\_time\_sec("team\_size", 3) time\_sec\_team\_size\_4 = add\_observations\_time\_sec("team\_size", 4) time\_sec\_team\_size\_5 = add\_observations\_time\_sec("team\_size", 5) time\_sec\_team\_size\_6 = add\_observations\_time\_sec("team\_size", 6)

Finally, we calculate the mean of “Time per Palette (seconds)” for the individual team sizes:

mean\_team\_3 = time\_sec\_team\_size\_3/ss\_team\_a mean\_team\_4 = time\_sec\_team\_size\_4/ss\_team\_b mean\_team\_5 = time\_sec\_team\_size\_5/ss\_team\_c mean\_team\_6 = time\_sec\_team\_size\_6/ss\_team\_d

mean\_team <- c(mean\_team\_3, mean\_team\_4, mean\_team\_5, mean\_team\_6)

Create a data frame consisting of the calculated means:

help\_df <- data.frame(team\_size = c(3:6),

mean\_time\_sec = as.double(mean\_team))

help\_df

## team\_size mean\_time\_sec

|  |  |  |
| --- | --- | --- |
| ## 1 | 3 | 208.6667 |
| ## 2 | 4 | 150.2500 |
| ## 3 | 5 | 158.3111 |
| ## 4 | 6 | 115.0408 |

## Transform “Team Size” data and export into csv file for further visualization

Define a function that returns number of observations for all observations of a variable:

sample\_size\_finder\_2 <- **function**(variable){ df %>%

select(.data[[variable]]) %>% count()

}

Gather the number of observations for the different team sizes:

ss\_team\_size <- sample\_size\_finder\_2("team\_size")

Build a vector of sample sizes per team size:

team\_size\_stats <- c(ss\_team\_size, ss\_team\_a, ss\_team\_b, ss\_team\_c, ss\_team\_d) team\_size\_stats

## $n

## [1] 199 ##

## $n

## [1] 3 ##

## $n

## [1] 12 ##

## $n

## [1] 135 ##

## $n

## [1] 49

Finally, we export team\_size\_stats as csv file for visualization in MS-Excel (You probably won’t need to do this step):

write.csv2(team\_size\_stats, "pie\_chart\_team\_size.csv")

## Variables “Time per Palette (seconds)” and “Item Volume (cubic centimeter)” in relation as box plot, sample sizes annotated

Gather the number of observations for given values of “Item Volume (cubic centimeter)”, which we will anotate to the box plot:

ss\_volume\_a <- sample\_size\_finder("item\_volume\_cc", 30723.00) ss\_volume\_b <- sample\_size\_finder("item\_volume\_cc", 41538.00) ss\_volume\_c <- sample\_size\_finder("item\_volume\_cc", 43987.50) ss\_volume\_d <- sample\_size\_finder("item\_volume\_cc", 64275.75) ss\_volume\_e <- sample\_size\_finder("item\_volume\_cc", 85800.00)

Create the box plot and show it:

box\_df2 <- df %>% ggplot() +

geom\_boxplot(mapping = aes(group = item\_volume\_cc, x = item\_volume\_cc, y = time\_sec)) + labs(title = "How does the volume of an item impact palette completion time?",

subtitle = "comparison via box plot \nn = sample size", caption = "source: own data",

x = "Item Volume \n(cubic centimeter)", y = "Time per Palette \n(seconds)") +

annotate("text", x = 30723.00, y = 150, label = paste(str\_0, ss\_volume\_a)) + annotate("text", x = 37500.00, y = 120, label = paste(str\_0, ss\_volume\_b)) + annotate("text", x = 47500.00, y = 75, label = paste(str\_0, ss\_volume\_c)) + annotate("text", x = 64275.75, y = 50, label = paste(str\_0, ss\_volume\_d)) + annotate("text", x = 85800.00, y = 20, label = paste(str\_0, ss\_volume\_e))

box\_df2

# How does the volume of an item impact palette completion time?

### comparison via box plot n = sample size

600

n = 10

n = 43

n = 34

n = 59

n = 53

400

Time per Palette (seconds)

200

0

30000

40000

50000

60000

### Item Volume

70000

80000

### (cubic centimeter)

source: own data

**Multiple linear regression, regular and Z-standardized**

We do a regular multiple linear regression. Let’s define the model with our controls and execute it:

mlr\_df <- lm(time\_sec ~team\_size + item\_weight\_kg + item\_volume\_cc + palette\_quantity + palette\_number,

data = df)

Get results from the regular regression as a nicely formatted table via stargazer library:

stargazer(mlr\_df,

type = "text",

title = "Multiple Linear Regression", out = "tablex.txt",

covariate.labels = c("Team Size",

"Item Weight (kilograms)",

"Item Volume (cubic centimeters)", "Item Quantity per Palette", "Number of Palette"))

##

## Multiple Linear Regression

## ===========================================================

## Dependent variable:

## ---------------------------

## time\_sec

##

## Team Size -18.726

|  |  |
| --- | --- |
| ## ## | (14.276) |
| ## | Item Weight (kilograms) 12.571 |
| ## | (8.002) |
| ## |  |
| ## | Item Volume (cubic centimeters) -0.001 |
| ## | (0.001) |
| ## |  |
| ## | Item Quantity per Palette 1.662 |
| ## | (1.082) |
| ## |  |
| ## | Number of Palette 0.375 |
| ## | (0.382) |
| ## |  |
| ## | Constant 191.219\*\*\* |
| ## | (63.680) |
| ## |  |
| ## |  |
| ## | Observations 199 |
| ## | R2 0.111 |
| ## | Adjusted R2 0.088 |
| ## | Residual Std. Error 66.529 (df = 193) |
| ## | F Statistic 4.836\*\*\* (df = 5; 193) |
| ## | =========================================================== |
| ## | Note: \*p<0.1; \*\*p<0.05; \*\*\*p<0.01 |

Data transformation for z-standardized multiple linear regression via the scale() function:

z\_time\_sec <- scale(df$time\_sec) z\_team\_size <- scale(df$team\_size)

z\_item\_weight\_kg <- scale(df$item\_weight\_kg) z\_item\_volume\_cc <- scale(df$item\_volume\_cc) z\_palette\_quantity <- scale(df$palette\_quantity) z\_palette\_number <- scale(df$palette\_number)

Build a new data frame from the z-standardized data:

dfz <- data.frame(z\_time\_sec,

z\_team\_size, z\_item\_weight\_kg, z\_item\_volume\_cc, z\_palette\_quantity, z\_palette\_number)

Run the model and show results:

Z\_mlr\_df <- lm(z\_time\_sec ~ z\_team\_size +

z\_item\_weight\_kg + z\_item\_volume\_cc + z\_palette\_quantity + z\_palette\_number, data = dfz)

Run this, if you want neater output for the z-regression:

stargazer(Z\_mlr\_df, type = "text",

title = "Z-Standardized Multiple Linear Regression", out = "tablez.txt",

covariate.labels = c("Team Size",

"Item Weight (kilograms)",

"Item Volume (cubic centimeters)", "Item Quantity per Palette", "Number of Palette"))

##

## Z-Standardized Multiple Linear Regression

## ===========================================================

## Dependent variable:

## ---------------------------

## z\_time\_sec

##

## Team Size -0.158

## (0.120)

##

## Item Weight (kilograms) 0.384

## (0.244)

##

## Item Volume (cubic centimeters) -0.348

## (0.334)

##

## Item Quantity per Palette 0.195

## (0.127)

##

## Number of Palette 0.077

## (0.079)

##

## Constant 0.000

## (0.068)

##

##

## Observations 199

## R2 0.111

## Adjusted R2 0.088

## Residual Std. Error 0.955 (df = 193)

## F Statistic 4.836\*\*\* (df = 5; 193)

## ===========================================================

## Note: \*p<0.1; \*\*p<0.05; \*\*\*p<0.01