Data Analysis

by Matthias Langenhan

03.10.2021

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)
```

```
##
## Please cite as:
```

```
##  Hlavac, Marek (2018). stargazer: Well-Formatted Regression and Summary Statistics Tables.
```

```
##  R package version 5.2.2. https://CRAN.R-project.org/package=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",
```

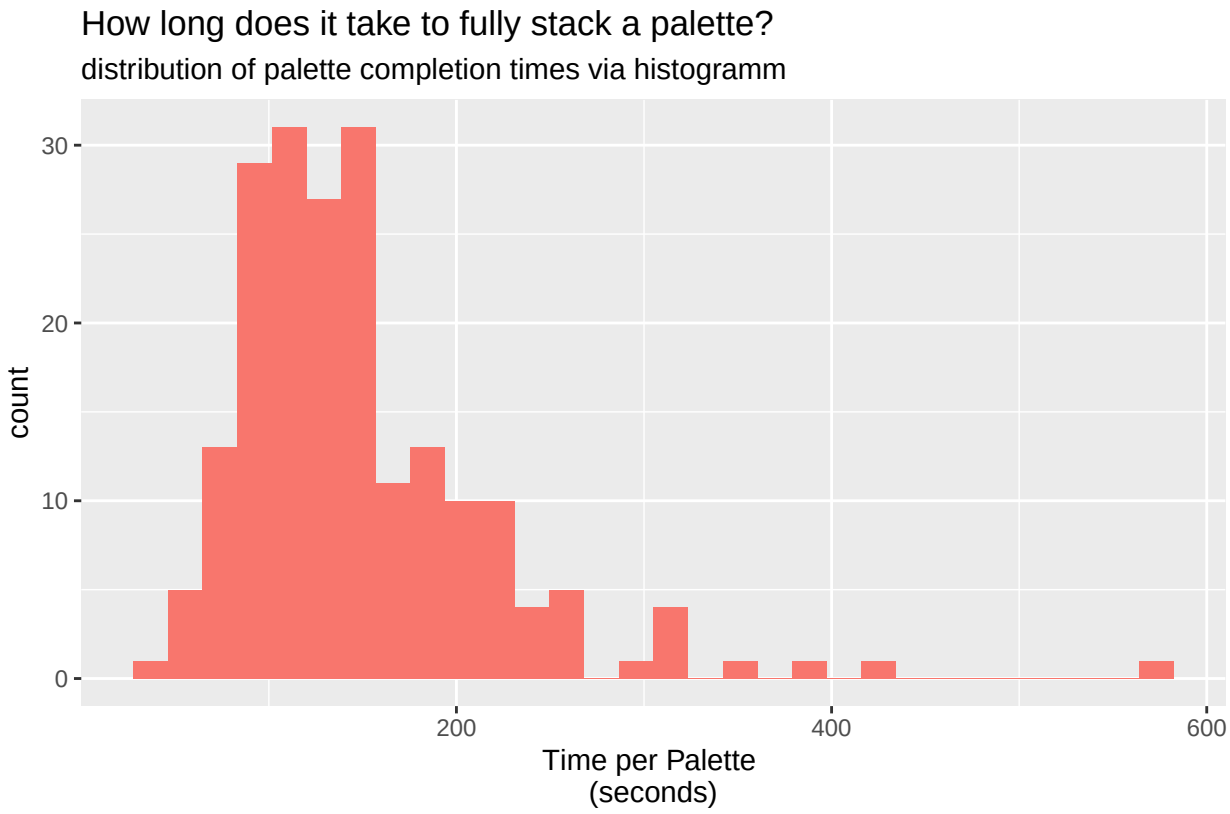
```
                "item_weight_kg",
                "item_volume_cc",
                "palette_quantity",
                "palette_number",
                "item_time_sec")],
          type = "text",
          title = "Descriptive Statistics",
          out = "table1.txt",
          covariate.labels = 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)"))
```

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

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
```



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:

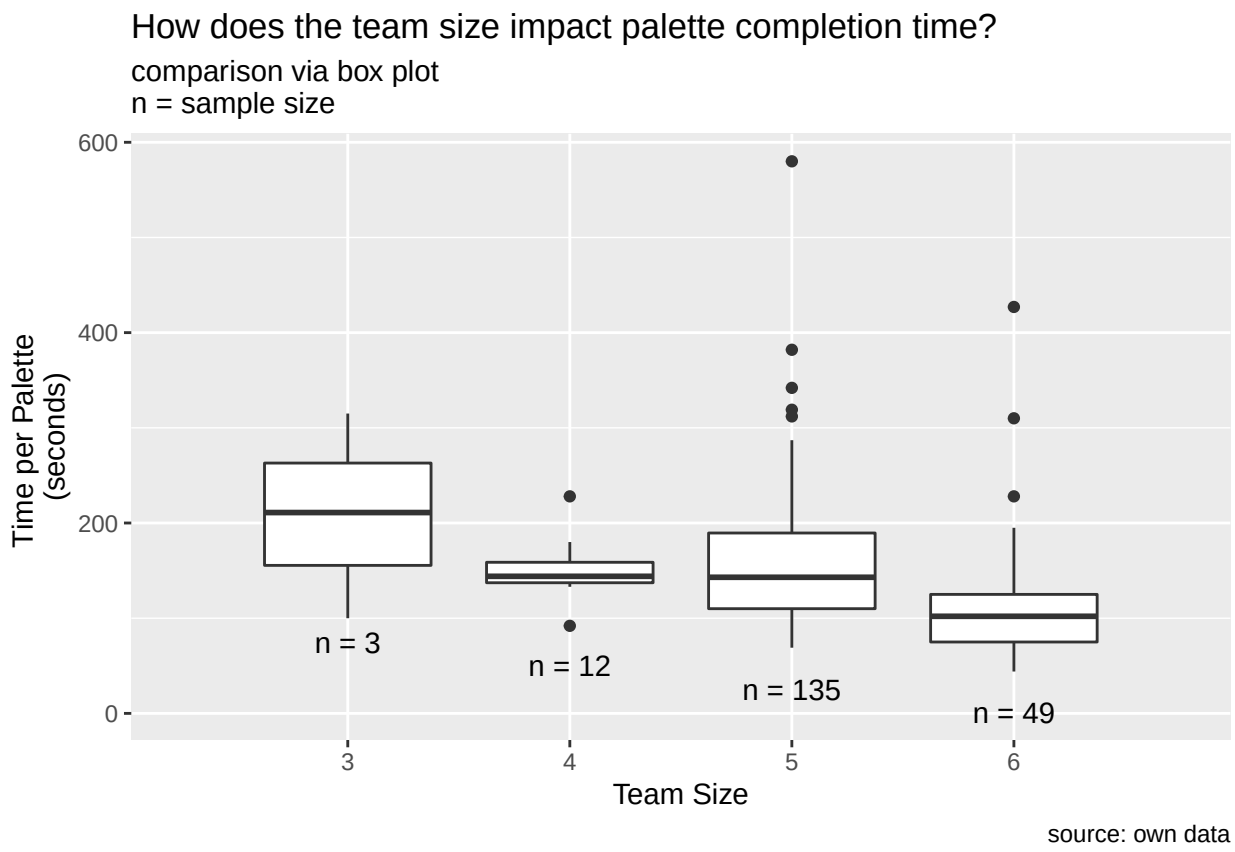
```
str_0<- "n ="
ss_team_a = sample_size_finder("team_size", 3)
ss_team_b = sample_size_finder("team_size", 4)
ss_team_c = sample_size_finder("team_size", 5)
ss_team_d = sample_size_finder("team_size", 6)
```

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
```



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

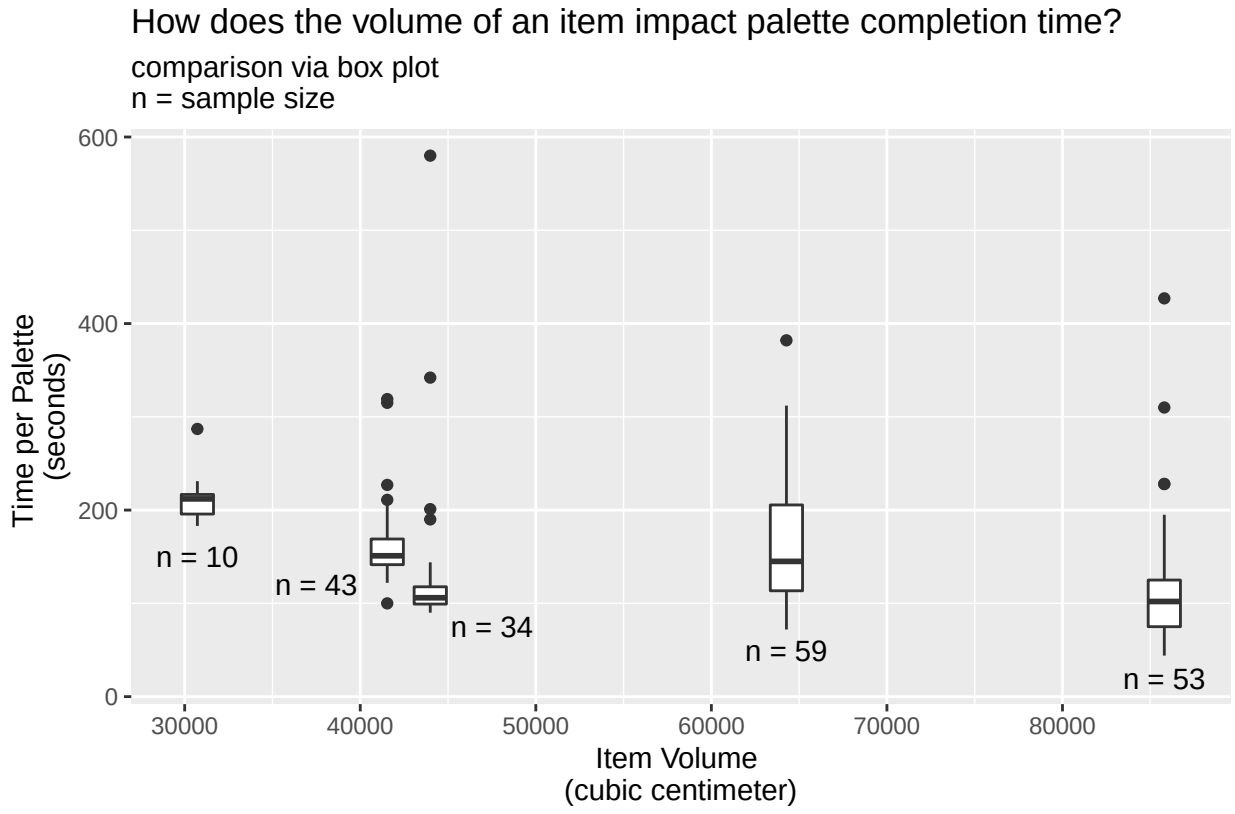
Gather the number of observations for given values of "Item Volume (cubic centimeter)", which we will anotate to the box plot:

```
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
```



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
```