Essential R Markdown Exercises for MATH513 Big Data and Social Network Visualization

Lecturer: Luciana Dalla Valle
Notes: Julian Stander
Student: Alex Mounsey (10570155)

October 2020

1 Exercise 1

Reproduce all the material that appears here using RMarkdown.

To save you typing, most of the text is given in the file TEXT_ONLY_of_Tutorial_8_R_Markdown.txt.

This material is meant to be instructive.

Please proceed step by step!

Check that your formatting works as you go along.

Your task is to:
- Re-create this document

1.1 Slides

It is possible to produce slides using RMarkdown. The following steps will get you started:

- Go to 'File', then
- 'New File', then
- 'R Markdown'...

Choose:

- Presentation
- Give the presentation a **Title**, such as *'First Presentation'*, and specify the **Author(s)**
- Specify **PDF (Beamer)**, then **OK**. Please note that this assumes that a **working version of LaTeX is installed**

RStudio should open an example R Markdown Beamer Presentation document:

- Run the document by clocking the **Knit** or **Knit PDF** button

Please make sure that you have saved the document to a suitable directory, with a sensible file name.

- Have a look at what is produced. You should see slides containing R code, together with the output that it produces, including a figure

Warning: There may be problems with creating slides
You may have to keep pressing the **'Knit PDF'** button, replying **'Install'** (or similar) each time. This is because not all the underlying LaTeX packages have been properly installed. This is a nuisance, but only has to be done **once** for each R session.

1.2 Some Topics that We Will See in this Module

1. R Packages: Create and develop your R package as a collection of R functions and datasets
2. Social Media Sentiment Analysis: How to associate user sentiments to social media text data

1.3 Statistical Tests

By the end of the module, we will have studied the following:

- Tests on the shape of a simple linear regression model
- Test on means:
  - Comparing two means: the i-test
  - Comparing more than two means: the analysis of variance (**ANOVA**)

1.4 The Analysis of Variance

Here is an example of initial analysis from:

Anderson, D, R., Sweeney, D. J., Williams, T. A., Freeman, J. and Shoesmith, E. (2010). Statistics for Business and Economics, Second Edition. South-Western CENGAGE Learning.

National Computer Products (NCP) manufactures printers at plants located in Ayr, Dusseldorf and Stockholm. To measure how much employees at these plants know about total quality management, a random sample of six employees was selected from each plant and given a quality awareness examination.

Here is one way to analyze this data:

- First, input the data into R:

```
ayr <- c(85, 85, 82, 76, 71, 85)
dusseldorf <- c(71, 75, 73, 74, 69, 82)
stockholm <- c(59, 64, 62, 69, 75, 67)
```

- Put these vectors into a dataframe:

```
df <- data.frame(ayr, dusseldorf, stockholm)
df
```

```
##   ayr dusseldorf stockholm
## 1  85         71        59
## 2  85         75        64
## 3  82         73        62
## 4  76         74        69
## 5  71         69        75
## 6  85         82        67
```

- Use functionality from `tidyr` to convert the dataframe to the long format so that all of the scores are in one column:

```
require(tidyr)

df_2 <- df %>%
  gather(Location, Score, 1:3)
df_2
```

```
##      Location Score
## 1         ayr    85
## 2         ayr    85
## 3         ayr    82
## 4         ayr    76
## 5         ayr    71
## 6         ayr    85
## 7  dusseldorf    71
## 8  dusseldorf    75
## 9  dusseldorf    73
## 10 dusseldorf    74
## 11 dusseldorf    69
## 12 dusseldorf    82
## 13  stockholm    59
## 14  stockholm    64
## 15  stockholm    62
## 16  stockholm    69
## 17  stockholm    75
## 18  stockholm    67
```

- Use functionality from `dplyr` in order to turn 'Location' into a factor with suitable labels:

```
require(dplyr)

df_3 <- df_2 %>%
  mutate(Location_f = factor(Location,
                             levels = c("ayr", "dusseldorf", "stockholm"),
                             labels = c("Plant 1 Ayr",
                                        "Plant 2 Dusseldof",
                                        "Plant 3 Stockholm")))
df_3
```

```
##      Location Score        Location_f
## 1         ayr    85       Plant 1 Ayr
## 2         ayr    85       Plant 1 Ayr
## 3         ayr    82       Plant 1 Ayr
## 4         ayr    76       Plant 1 Ayr
## 5         ayr    71       Plant 1 Ayr
## 6         ayr    85       Plant 1 Ayr
## 7  dusseldorf    71 Plant 2 Dusseldof
## 8  dusseldorf    75 Plant 2 Dusseldof
## 9  dusseldorf    73 Plant 2 Dusseldof
## 10 dusseldorf    74 Plant 2 Dusseldof
```

```
## 11 dusseldorf    69 Plant 2 Dusseldof
## 12 dusseldorf    82 Plant 2 Dusseldof
## 13  stockholm    59 Plant 3 Stockholm
## 14  stockholm    64 Plant 3 Stockholm
## 15  stockholm    62 Plant 3 Stockholm
## 16  stockholm    69 Plant 3 Stockholm
## 17  stockholm    75 Plant 3 Stockholm
## 18  stockholm    67 Plant 3 Stockholm
```

- Now, compute the sample mean, the sample median, and the sample standard deviation score for each location:

```
df_3 %>%
  group_by(Location) %>%
  summarise(mean = mean(Score),
            median = median(Score),
            stdev = sd(Score))
```

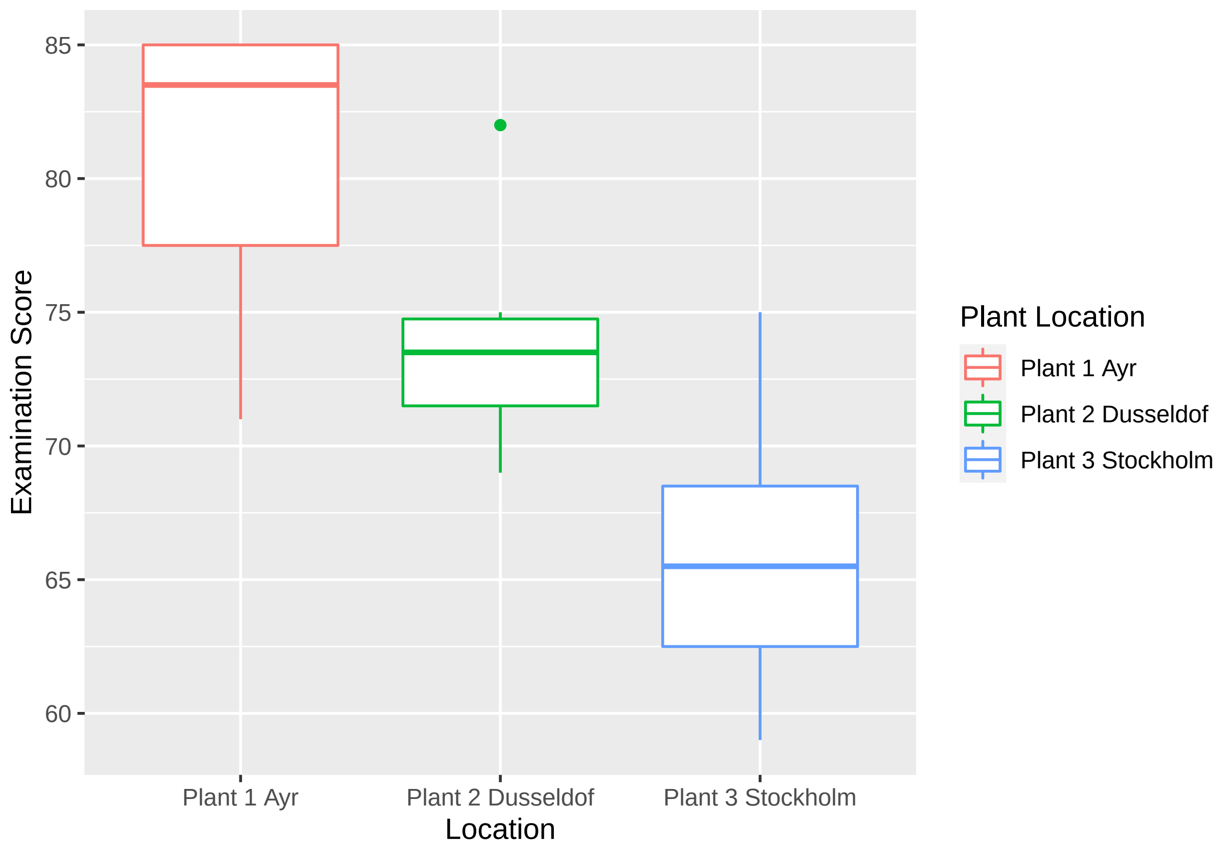
```
## # A tibble: 3 x 4
##   Location    mean median stdev
##   <chr>      <dbl>  <dbl> <dbl>
## 1 ayr         80.7   83.5  5.89
## 2 dusseldorf  74     73.5  4.47
## 3 stockholm   66     65.5  5.66
```

The sample standard deviations (spread) are similar for each location, while the sample means and medians seem rather different. Let's examine this graphically:

- Visualize the data by means of boxplots using `ggplot2`:

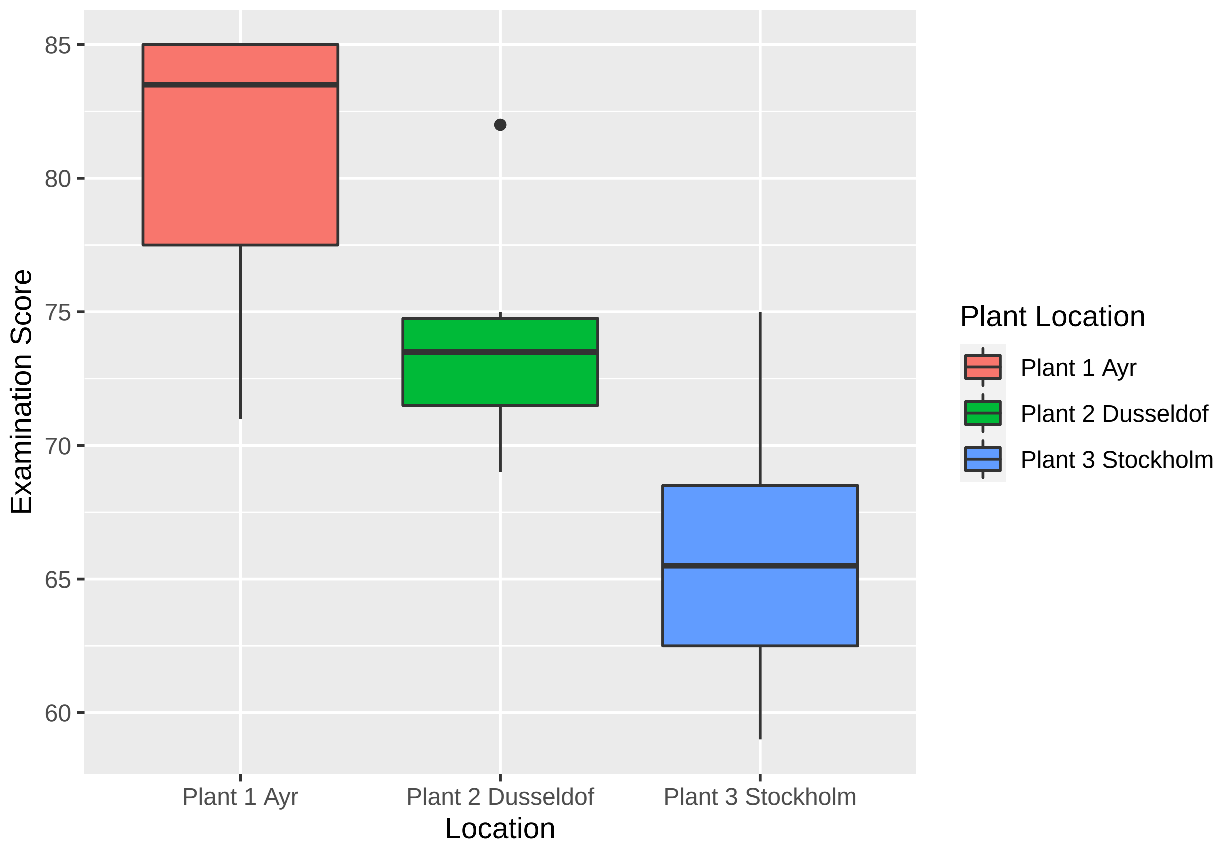
```
require(ggplot2)

ggplot(df_3, aes(x = Location_f, y = Score, col = Location_f)) +
  geom_boxplot() +
  labs(x = "Location", y = "Examination Score", col = "Plant Location")
```



or, even:

```
ggplot(df_3, aes(x = Location_f, y = Score, fill = Location_f)) +
  geom_boxplot() +
  labs(x = "Location", y = "Examination Score", fill = "Plant Location")
```



These plots show considerable differences in the examination score between plant locations.

2 The Analysis of Variance - Advanced

Here is, again, an example taken from:

Anderson, D, R., Sweeney, D. J., Williams, T. A., Freeman, J. and Shoesmith, E. (2010). Statistics for Business and Economics, Second Edition. South-Western CENGAGE Learning.

The examination scores for the 18 employees are listed in the following LaTeX table:

Plant 1 Ayr	Plant 2 Dusseldorf	Plant 3 Stockholm
85	71	59
75	75	64
82	73	62
76	74	69
71	69	75
85	82	67

We can perform a test to see whether the *underlying* mean examination scores for the three manufacturing plants are the same or not. We are **not** asking whether the means of the six examination scores from each plant are different; we know this from summary statistics calculated elsewhere. We are asking a **more profound** question: are the means of **all possible scores** from the plants different?

First, let us write down the analysis of variance model:

$$
\begin{aligned}
y_i &= \mu_A + \epsilon_i \text{ for Plant 1 Ayr} \\
y_i &= \mu_D + \epsilon_i \text{ for Plant 2 Dusseldorf} \\
y_i &= \mu_S + \epsilon_i \text{ for Plant 3 Stocholm}
\end{aligned}
$$

In which the errors $\epsilon_i \sim N(0, a^2$ independently

- To answer this question about the underlying means, we formulate two hypotheses.

The *null hypotheses* is $H_0 : \epsilon_A = \epsilon_D = \epsilon_S$, in which $\epsilon_A/\epsilon_D/\epsilon_S$ are the underlying mean scores from Plant 1 Ayr/Plant 2 Dusseldorf/Plant 3 Stockholm.

The *alternative* hypothesis is H_1: underlying means are not all equal.

- We now perform an Analysis of Variance of **ANOVA** test. **ANOVA** is an example of a linear model.

First, we have to manipulate the data into a suitable format, as we have done before:

```r
require(tidyr)
require(dplyr)
df <- data.frame(ayr, dusseldorf, stockholm)
df_2 <- df %>%
        gather(Location, Score,  1:3)
df_3 <- df_2 %>% mutate(Location_f =
                        factor(Location,
                         levels = c("ayr",
                                    "dusseldorf",
                                    "stockholm"),
                         labels = c("Plant 1 Ayr",
                                    "Plant 2 Dusseldorf",
                                    "Plant 3 Stockholm")))
df_3
```

```
##      Location Score         Location_f
## 1         ayr    85        Plant 1 Ayr
## 2         ayr    85        Plant 1 Ayr
## 3         ayr    82        Plant 1 Ayr
## 4         ayr    76        Plant 1 Ayr
## 5         ayr    71        Plant 1 Ayr
## 6         ayr    85        Plant 1 Ayr
## 7  dusseldorf    71 Plant 2 Dusseldorf
## 8  dusseldorf    75 Plant 2 Dusseldorf
## 9  dusseldorf    73 Plant 2 Dusseldorf
## 10 dusseldorf    74 Plant 2 Dusseldorf
## 11 dusseldorf    69 Plant 2 Dusseldorf
## 12 dusseldorf    82 Plant 2 Dusseldorf
## 13  stockholm    59  Plant 3 Stockholm
## 14  stockholm    64  Plant 3 Stockholm
## 15  stockholm    62  Plant 3 Stockholm
## 16  stockholm    69  Plant 3 Stockholm
## 17  stockholm    75  Plant 3 Stockholm
## 18  stockholm    67  Plant 3 Stockholm
```

Now, we perform the **ANOVA** test, using the `lm` function:

```
m <- lm(Score ~ Location_f, data = df_3)
anova(m)
```

```
## Analysis of Variance Table
##
## Response: Score
##            Df Sum Sq Mean Sq F value   Pr(>F)
## Location_f  2 647.11  323.56    11.2 0.001057 **
## Residuals  15 433.33   28.89
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

We can extract the p-value:

```
p_value <- anova(m)$"Pr(>F)"[1]
p_value
```

```
## [1] 0.001057176
```

The p-value is 0.0010572. As this is less than 0.05, we reject the null hypothesis H_0 and conclude that there is a difference in the underlying mean examination scores from the three manufacturing plants.