|  |
| --- |
| Photo displaying partial image of two pie charts on a canvas-textured page |
| Airbnb Berlin & Videogames sales – Data Insights with R  Programming for Big Data |
| |  |  |  | | --- | --- | --- | | Teresa Ventaja | 26/04/2020 | [Course title] | |

INDEX

[Methodology 2](#_Toc38725686)

[DATASET 1 2](#_Toc38725687)

[**Literature review** 2](#_Toc38725688)

[**Objectives** 3](#_Toc38725689)

[**Analysis** 3](#_Toc38725690)

[Selection & Dataset Description 3](#_Toc38725691)

[Pre-processing 4](#_Toc38725692)

[Transformation 4](#_Toc38725693)

[Data Mining 5](#_Toc38725694)

[Interpretation/Evaluation 5](#_Toc38725695)

[DATASET 2 8](#_Toc38725696)

[**Literature review** 8](#_Toc38725697)

[**Objectives** 9](#_Toc38725698)

[**Analysis** 9](#_Toc38725699)

[Selection & Dataset Description 9](#_Toc38725700)

[Pre-processing 10](#_Toc38725701)

[Transformation 10](#_Toc38725702)

[Data Mining 10](#_Toc38725703)

[Interpretation/Evaluation 11](#_Toc38725704)

[Challenges & Mechanisms to Overcome them 13](#_Toc38725705)

[Annex – R Code 14](#_Toc38725706)

[Bibliography 40](#_Toc38725707)

# Methodology

I based this project on Knowledge Discovery in Databases (KDD)[[1]](#footnote-1) methodology, due to its simplicity and its flexibility to build up some new conclusions starting with prior knowledge on the topic. I designed the research of each dataset starting with literature review to help building the objectives. Then I divided the analysis in the 5 method stages.

I performed descriptive statistical calculations, and one predictive test of hypothesis (t-test). I used multiple graphical representations created in both, R and Excel to help visualize and interpret the results. In both datasets I deleted some columns from the start, so that I could focus only in the data that interested me[[2]](#footnote-2).

# DATASET 1

## **Literature review**

For the purpose of this project, I am focusing this review on describing the importance of timely communications in Airbnb. I will then switch to regional factors affecting the demand of Airbnb accommodation in Berlin. These topics have been selected because they are related with the analysis that I am going to do.

Given that the average cost of spending a night on an Airbnb accommodation in Berlin is almost half the cost of a night on a hotel, it is no surprise that more and more travellers are choosing such kind of P2P platform for their trips[[3]](#footnote-3). But for the platform to attract users, it needs to build trust among them. In that sense, timely communication is effective to facilitate trust between users. Hosts unable to meet with that expectations results on guest’s perception of lack of interactional justice (unexpected losses of time and money)[[4]](#footnote-4). Airbnb messages between guests and hosts are normally notified by email. Time expectations for email responses is 1 day for an average of 43% users[[5]](#footnote-5). Even though 44% would expect a shorter time, Airbnb uses that rationale to allow 1 day to hosts to respond to guests without penalties[[6]](#footnote-6).

Switching the topic to regional factors, governments are aware of the negative impact of Airbnb listings on the rental market and hotels demand. As such, we can see that an increasing number of city authorities are imposing restrictions on Airbnb. In Berlin, in April 2016, landlords renting more than 50% of the property could have incurred in a fine worth up to 100.000€[[7]](#footnote-7). The restriction was partially lifted on May 2018, allowing landlords to rent on Airbnb if they acquired a permission, but rising the penalties if they broke the rules. In terms of the most popular Berlin neighbourhoods on Airbnb, Mitte and Friedrichshain-Kreuzberg are on the lead, followed by Charlotten-Wilmersdorf, and Tempelhof-Schöneberg[[8]](#footnote-8).

## **Objectives**

1. Explore what a good price is for an Airbnb in Berlin.

2. Explore how hosts and superhosts behave in terms of the time they take to respond to reviews.

3. Explore how location is valued.

3.1. Check which neighbourhoods are best rated by location

3.2. Check if there is a significant difference between the mean overall rating and the mean location rating

4. Explore if users submit more reviews on certain weekdays

## **Analysis**

### Selection & Dataset Description

I decided to research on Airbnb data because I work for this company and I am familiar with how the platform works. I was interested on knowing how analytical tasks look like within the company. Also, I picked Berlin data due to availability on the source and significance of data. Berlin is a European capital city and the dataset was large enough to gather some interesting insights.

I selected a dataset containing detailed data on Airbnb reviews in Berlin from 20/06/2009 to 14/05/2019. It originally contained 456.978 rows and 47 columns, but I decided to get rid of those that I was sure I was not interested in. Dataset description details can be found below, and dataset link clickable on the footnotes[[9]](#footnote-9):



### Pre-processing

Since my original dataset is large and less than 5% of the rows are blank, I simply deleted rows containing empty cells in Excel using “*Find and Select*”, selecting blanks and removing the rows associated with them.

### Transformation

1. Since my dataset contains a list of reviews, not categories of listings, I had multiple reviews per listing. To calculate descriptive statistics, first I needed to create a data frame in R and write a .csv file (see price\_listing.csv).

2. Since my dataset contains a list of reviews, not categories of listings, I had duplicate rows per listing. I got rid of the duplicates and created subsets for the categories I needed: host/superhost and review response time.

3.1. I got rid of the duplicate listing rows and created subsets for the categories I needed: listing ID, neighbourhood and location rating (by neighbourhood).

3.2. I got rid of the duplicate listing rows and created a data frame with the columns that I needed: listing ID, overall rating and location rating.

4. I only used one column from my original data sample, converted numerical dates into weekdays and loaded them into R.

### Data Mining

1. I used data analysis tool in excel to calculate descriptive statistics. I will not take the mean price as a reference, because the data is not normally distributed, and it does not reflect the average price as such. I will consider that the range between -10% of the mode to +10% of the median is a good price, because it does reflect what hosts decide to use the most as a price, therefore if we assume they want to maximise sales, that is a good reference. Average price will be 27-54€, more than that is expensive, less is cheap.

2. Once I had my subsets, I had the counting of listings that belonged to each category. Based on that, I created a matrix, and then exported the data as a .csv file.

3.1. I calculated the mean rating for each neighbourhood. I needed to fix the data for those that were not numeric (2 subsets), removing NA rows. I created a matrix, exported it as a .csv file and created the chart in Excel.

3.2. I created 2 vectors for overall ratings and location ratings and executed t-test in R for them.

4. I created subsets so I could get the totals for each weekday more easily. Then I created a matrix with my results and exported them into a .csv file.

### Interpretation/Evaluation

1. I created a calculator in R to display whether a price is good or not. It is represented in the chart below:

A close up of a map

Description automatically generated

2. I created a graph so we can see graphically the differences in response time:

We can deduct that in Berlin, both hosts and superhosts typically take about one hour to respond to the review. However, it is relevant to notice that some superhosts do not respond at all, and it makes sense if we consider that they tend to have more positive reviews, so they do not need to. Also, few superhosts take more than a few hours to respond, whereas a significant number of hosts take one day or more.

3.1. We can find below R Mat Plot and a complementary graph with more details on neighbourhoods:

A picture containing table, photo, water, skiing

Description automatically generated

We can see that the rating matches only partially what we covered on the literature review. While Mitte and Friedrichshain-Kreuzberg are top rated, Airbnb users also rank very highly Charlotten-Wilmersdorf and Pankow (that one not mentioned in the review). Also, we can confirm that users are inclined to leave positive reviews, influenced by the acquiescence bias and other factors[[10]](#footnote-10).

3.2. P value equals 2.2e-16 (close to 0), so I reject the null hypothesis. T-test indicates that, on 95% confidence level, there is significant evidence that the mean of overall rating is different than the mean of location rating.

4. I created a pie chart in excel, showing the percentage of reviews from the total, by weekday:

I can clearly infer that in some days the number of reviews is almost double than others. Sundays are Mondays are the days where most users post reviews, whereas Tuesday to Saturday there are significantly less reviews.

# DATASET 2

## **Literature review**

As an introductory reference, it catches my curiosity to discover the factors influencing positively the performance of a games company. It was been investigated that such factors are user satisfaction, time to market, monetization strategy, market orientation, and brand name strategy[[11]](#footnote-11). However, I am going to focus my study on topics directly associated with my dataset, but also making sure that are matters on public discussion nowadays.

Firstly, I will dig into historical hits on video games, which video games made history[[12]](#footnote-12). This is relevant since classic games are in high demand nowadays[[13]](#footnote-13). Another widely discussed topic is on the influence of violent video games on behaviour, specially among children and adolescents[[14]](#footnote-14), since their cognitive structure are still developing. In this paper, I am not going to check behavioural characteristic, but I will explore if the demand is high.

I am also interested in knowing which platform is leading the demand of video games in recent years. Since modern consoles launched on 2005, I will check the data for units sold and compare it with the data covering a wider period[[15]](#footnote-15). In terms of geographical distribution, it seems that APAC is the leading market, followed by North America and EMEA[[16]](#footnote-16)[[17]](#footnote-17). I will explore my dataset to catch some insights on it.

## **Objectives**

1. Video games hits: explore which video games sold the most units on launching year (only counting the first game on the saga).

2. Myth or reality: are violent games more demanded?

3. Lead in hardware ownership for games launched Jan 2005- Jan 2017: explore which platform sold more units

4. Compare percentage of units sold by region

## **Analysis**

### Selection & Dataset Description

I selected the topic because video games are a hobby of mine, so I am naturally curious about it. My dataset is large enough and covers a wide variety of videogames across regions and timeline. The dataset contains video games launched from 1976 to 2017. It originally contained 17416 rows and 15 columns, but I got rid of 5 columns that I was not interested in. Dataset description details can be found below, and link to the dataset on the footnotes[[18]](#footnote-18):



### Pre-processing

The original dataset had 8 N/A rows on the year of release. I made a little research to find out the year of release to replace them with real data. Also, I deleted 15 rows in total containing URLs and prototypes on the “Name” column. These rows represent less than 0.1% of the total, so the impact on my analysis is minimum.

### Transformation

1. I had to do an initial research on the literature review to find out the names and dates of historical hits. Then I uploaded my .csv and worked on it.

2. I worked directly on my .csv dataset.

3. First I filtered for games whose year of release is greater or equal to 2005. I isolated only the columns that I needed (Platform/Global\_sales).

4. I worked directly on my .csv dataset.

### Data Mining

1. I filtered on my dataset to find the global sales for each video game, and created a bar plot in R.

2. I calculated the total global sales for all genres, and the totals for each genre separately. Then I calculated the percentage of sales for each genre, and created a matrix in R, exporting the data into a new .csv file. I did not create the pie chart on R because the visual appearance would not have been detailed enough.

3. I calculated the total global unit sales by platform, discarding Nintendo 64 and Dreamcast games, because these consoles launched far earlier than 2005. I also discarded PC games because they are not strictly compatible with a single platform. Then I created a lollipop plot in R.

4. I calculated the total units sold by region, and then created a Tree map in R (‘treemap’ package).

### Interpretation/Evaluation

1. I created a bar plot in R:

A screenshot of a cell phone

Description automatically generated

We can see that the historical video games that sold the most units are “*Super Mario Bros*.” (1985), “*Tetris” (*1992) and “*Super Mario 64”* (1996). Only “*Wii Sports*” has surpassed that version of “*Super Mario Bros”* in global sales.

2. I created a pie chart in Excel to compare results:

Since we do not have a “violent” category, we must make some assumptions. Assuming that fighting games, shooters, action and strategy tend to be more violent, they make a total of 39%. However, such assumption is not accurate, because some action games are not violent per se, and some adventure or role-playing have a certain degree of violence. Therefore, we cannot conclude that 2 out of 5 games sold are violent, but we can say that shooters and action games are the lead in sales.

3. I created a lollipop plot in R to compare results:

A picture containing group, standing

Description automatically generated

X-Box 360 was the most popular platform in units of video games sold as of January 2017. It is followed very closely by PlayStation 3 and Wii. In portable format, Nintendo DS has the greatest popularity. In terms of revenue, we do not have data in this dataset to make conclusions, but we can say that Sony consoles sold more video game units than other manufacturers.

4. I created a Tree map in R representing the results:

A screenshot of a cell phone

Description automatically generated

We can clearly see that North America is the lead in units sold by far, and the amount of unit sold out of Japan, the EU and North America is very low.

# Challenges & Mechanisms to Overcome them

The fist remarkable challenge I can recall is on how to know if my objectives are appropriate enough. My concern was not covering what I am expected to do. In other words, I wanted to add as much complexity as possible and utilise a wide variety of techniques. I struggled specially to find topics to do a hypothesis testing on. I finally decided to include it on Objective 3.2 of my first dataset, however, it took me a while to decide which test was appropriate for my data.

Another difficulty was dealing with different data structures in R. I used vectors, data frames, subsets, matrices, etc., and the combination of them is not as simple as it seems. Some operations needed conversions or using of different indexing techniques to overcome error results. Last but least, the biggest challenge for me was using a scientist paper structure, such as IEEE. This time I did not have enough time to get ready to use a scientist convention, but it will definitely be my goal for the final project of this course. Limiting my word count was another little challenge, but finally I have not exceed widely 2500, so I am happy enough in that sense.

# Annex – R Code

#Dataset 1

# Read csv Airbnb file

airbnb <- read.csv(file="airbnb\_berlin\_cl.csv",head=TRUE,sep=",")

airbnb

# Objective 1

#Install package so I can select different categories of listings

install.packages('dplyr')

# Create a data frame with the price for objective 1

airbnb\_price <- data.frame(airbnb$listing\_id, airbnb$price)

airbnb\_price

# Create a vector including only different categories of listings

# Need to do it because there are multiple duplicate listings

# Because each row of the dataset is for one review, not for one listing

library(dplyr)

single\_prices <- distinct(airbnb\_price)

single\_prices

# Write a CSV file that I can manipulate

write.csv(single\_prices,"C:\\Users\\Owner\\Desktop\\datasets\\airbnb\\price\_listing.csv", row.names = FALSE)

# Create a calculator that displays whether a price is cheap, expensive or average

price <- as.integer(readline(prompt="Enter a price(number): "))

for (val in price) {

if (price > 54) {

print("This Airbnb listing is too expensive");

print("Watch out your savings!");

} else if (price < 27) {

print("This Airbnb listing is specially cheap");

print("Be careful with quality standards!");

} else {

print("This price is average for an Airbnb listing in Berlin");

print("Good selection!");

}

}

price <- as.integer(readline(prompt="Enter a price(number): "))

# Objective 2

# Create a data frame for objective 2

response\_time <- data.frame(airbnb$listing\_id, airbnb$host\_response\_time, airbnb$is\_superhost)

response\_time

# Create a vector including only different categories of listings

# Need to do it because there are multiple duplicate listings

# Because each row of the dataset is for one review, not for one listing

remove\_duplicate\_times <- distinct(response\_time)

remove\_duplicate\_times

# Create subset with different categories

# Creating subsets is going to tell me in the environment how many observations

# there are for each subset. I will use these numbers to create a matrix

superhostNA <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="TRUE"

& airbnb.host\_response\_time=="N/A",

select=airbnb.listing\_id:airbnb.is\_superhost)

summary(remove\_duplicate\_times$airbnb.is\_superhost=='TRUE'

& remove\_duplicate\_times$airbnb.host\_response\_time=="N/A")

superhostFH <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="TRUE"

& airbnb.host\_response\_time=="within a few hours",

select=airbnb.listing\_id:airbnb.is\_superhost)

superhostAH <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="TRUE"

& airbnb.host\_response\_time=="within an hour",

select=airbnb.listing\_id:airbnb.is\_superhost)

superhostAD <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="TRUE"

& airbnb.host\_response\_time=="within a day",

select=airbnb.listing\_id:airbnb.is\_superhost)

superhostFD <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="TRUE"

& airbnb.host\_response\_time=="a few days or more",

select=airbnb.listing\_id:airbnb.is\_superhost)

hostNA <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="FALSE"

& airbnb.host\_response\_time=="N/A",

select=airbnb.listing\_id:airbnb.is\_superhost)

hostFH <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="FALSE"

& airbnb.host\_response\_time=="within a few hours",

select=airbnb.listing\_id:airbnb.is\_superhost)

hostAH <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="FALSE"

& airbnb.host\_response\_time=="within an hour",

select=airbnb.listing\_id:airbnb.is\_superhost)

hostAD <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="FALSE"

& airbnb.host\_response\_time=="within a day",

select=airbnb.listing\_id:airbnb.is\_superhost)

hostFD <- subset(remove\_duplicate\_times, airbnb.is\_superhost=="FALSE"

& airbnb.host\_response\_time=="a few days or more",

select=airbnb.listing\_id:airbnb.is\_superhost)

# Creating a matrix

times\_HS <- matrix(c(266, 0, 733, 1870, 2136, 4088, 328, 1999, 6, 411), nrow=2, ncol=5)

rownames (times\_HS) <- c("Superhost", "Host")

colnames (times\_HS) <- c("No\_response", "Few\_hours", "One\_hour", "One\_day", "More\_than\_one\_day")

times\_HS

# Write a CSV file that I can manipulate

write.csv(times\_HS,"C:\\Users\\Owner\\Desktop\\datasets\\airbnb\\time\_host\_superhost.csv", row.names = TRUE)

# Objective 3.1

# Create a data frame for objective 3.1

locations <- data.frame(airbnb$listing\_id, airbnb$neighborhood\_group, airbnb$location\_rating)

locations

# Create a vector including only different categories of listings

# Need to do it because there are multiple duplicate listings

# Because each row of the dataset is for one review, not for one listing

remove\_duplicate\_locations <- distinct(locations)

remove\_duplicate\_locations

# Create subset with different neighbourhood

# Creating subsets is going to help me to calculate the average rating per neighbourhood

CW <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Charlottenburg-Wilm.",

select=airbnb.listing\_id:airbnb.location\_rating)

FK <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Friedrichshain-Kreuzberg",

select=airbnb.listing\_id:airbnb.location\_rating)

LI <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Lichtenberg",

select=airbnb.listing\_id:airbnb.location\_rating)

MH <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Marzahn - Hellersdorf",

select=airbnb.listing\_id:airbnb.location\_rating)

MI <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Mitte",

select=airbnb.listing\_id:airbnb.location\_rating)

NE <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="NeukÃ¶lln",

select=airbnb.listing\_id:airbnb.location\_rating)

PA <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Pankow",

select=airbnb.listing\_id:airbnb.location\_rating)

RE <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Reinickendorf",

select=airbnb.listing\_id:airbnb.location\_rating)

SP <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Spandau",

select=airbnb.listing\_id:airbnb.location\_rating)

SZ <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Steglitz - Zehlendorf",

select=airbnb.listing\_id:airbnb.location\_rating)

TS <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Tempelhof - SchÃ¶neberg",

select=airbnb.listing\_id:airbnb.location\_rating)

TK <- subset(remove\_duplicate\_locations,

airbnb.neighborhood\_group=="Treptow - KÃ¶penick",

select=airbnb.listing\_id:airbnb.location\_rating)

# Create a vector for the overall\_rating column of each subset

Rating\_CW <- CW$airbnb.location\_rating

Rating\_FK <- FK$airbnb.location\_rating

Rating\_LI <- LI$airbnb.location\_rating

Rating\_MH <- MH$airbnb.location\_rating

Rating\_MI <- MI$airbnb.location\_rating

Rating\_NE <- NE$airbnb.location\_rating

Rating\_PA <- PA$airbnb.location\_rating

Rating\_RE <- RE$airbnb.location\_rating

Rating\_SP <- SP$airbnb.location\_rating

Rating\_SZ <- SZ$airbnb.location\_rating

Rating\_TS <- TS$airbnb.location\_rating

Rating\_TK <- TK$airbnb.location\_rating

# Calculate the mean with 2 decimal places

CW\_mean <- round(mean(Rating\_CW), digits=2)

FK\_mean <- round(mean(Rating\_FK), digits=2)

LI\_mean <- round(mean(Rating\_LI), digits=2)

MH\_mean <- round(mean(Rating\_MH), digits=2)

MI\_mean <- round(mean(Rating\_MI), digits=2)

NE\_mean <- round(mean(Rating\_NE), digits=2)

PA\_mean <- round(mean(Rating\_PA), digits=2)

RE\_mean <- round(mean(Rating\_RE), digits=2)

SP\_mean <- round(mean(Rating\_SP), digits=2)

SZ\_mean <- round(mean(Rating\_SZ), digits=2)

TK\_mean <- round(mean(Rating\_TK), digits=2)

TS\_mean <- round(mean(Rating\_TS), digits=2)

# Since FK\_mean and PA\_mean returned NA, I am exporting them as a CSV file to troubleshoot

write.csv(Rating\_FK,"C:\\Users\\Owner\\Desktop\\datasets\\airbnb\\FK.csv", row.names = FALSE)

write.csv(Rating\_PA,"C:\\Users\\Owner\\Desktop\\datasets\\airbnb\\PA.csv", row.names = FALSE)

# Remove 1 NA value in Excel on each file

# Import new data

FK\_new <- read.csv(file="FK.csv",head=FALSE,sep=",")

PA\_new <- read.csv(file="PA.csv",head=FALSE,sep=",")

# Replace faulty vectors with my new data

Rating\_FK <- FK\_new$V1

Rating\_PA <- PA\_new$V1

# Calculate the mean

FK\_mean <- round(mean(Rating\_FK), digits=2)

PA\_mean <- round(mean(Rating\_PA), digits=2)

# Creating a matrix

rating\_location <- matrix(c(9.55, 9.7, 9.14, 8.84, 9.53, 9.46, 9.64, 9.11, 8.88, 9.5, 9.29, 9.53), byrow=TRUE, nrow=1, ncol=12)

rownames (rating\_location) <- c("Mean Location Rating")

colnames (rating\_location) <- c("Charlottenburg-Wilmersdorf", "Friedrichshain-Kreuzberg",

"Lichtenberg", "Marzahn-Hellersdorf", "Mitte",

"Neukölln", "Pankow", "Reinickendorf", "Spandau",

"Steglitz-Zehlendorf", "Tempelhof-Schöneberg", "Treptow-Köpenick")

rating\_location

# Creating a Mat Plot with my matrix, where I can compare the results

# Since I have too many neighbourhood with long names, a leyend does not look good

# This is why I am adding a complementary Excel graph on the report

matplot(t(rating\_location), type="b", pch=15:18, col=c(4),

xlab = 'Neighbourhood', ylab = 'Rating')

# Write a CSV file to add a complementary graph

write.csv(rating\_location,"C:\\Users\\Owner\\Desktop\\datasets\\airbnb\\rating\_by\_neighborhood.csv", row.names = TRUE)

# Objective 3.2

# Create a data frame for objective 3.2

ratings <- data.frame(airbnb$listing\_id, airbnb$overall\_rating, airbnb$location\_rating)

ratings

# Create a vector including only different categories of listings

# Need to do it because there are multiple duplicate listings

# because each row of the dataset is for one review, not for one listing

# Note that the ratings showing are not the ones rated by each review

# but the ratings by listing

library('dplyr')

rem\_duplicate\_ratings <- distinct(ratings)

rem\_duplicate\_ratings

# Create vectors for overall\_rating and location\_rating columns

rating\_data <- subset(rem\_duplicate\_ratings, select=airbnb.overall\_rating:airbnb.location\_rating)

rating\_location2 <- rating\_data$airbnb.location\_rating

rating\_overall2 <- rating\_data$airbnb.overall\_rating

# t-test

testing\_ratings <-t.test(rating\_location2, rating\_overall2)

testing\_ratings

# Objective 4

# Create a data frame for objective 4

review\_date <- data.frame(airbnb$review\_date)

review\_date

# Write a CSV file that I can manipulate

# I will convert date into weekday

write.csv(review\_date,"C:\\Users\\Owner\\Desktop\\datasets\\airbnb\\review\_dates.csv", row.names = FALSE)

# Load csv with data converted

review\_weekday <- read.csv(file="review\_dates.csv",head=FALSE,sep=",")

review\_weekday

# Obtaining the total observations for each weekday

summary(review\_weekday)

# Creating a matrix based on summary observations

review\_per\_weekday <- matrix(c(82746, 58545, 50721, 51244, 56254, 53364, 99250), byrow=TRUE, nrow=1, ncol=7)

rownames (review\_per\_weekday) <- c("Number of Reviews")

colnames (review\_per\_weekday) <- c("Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday", "Sunday")

review\_per\_weekday

# Write a CSV file that I can manipulate

write.csv(review\_per\_weekday,"C:\\Users\\Owner\\Desktop\\datasets\\airbnb\\review\_weekday.csv", row.names = TRUE)

#Dataset 2

# Read csv Airbnb file

video\_games <- read.csv(file="video\_games\_clean.csv",head=TRUE,sep=",")

video\_games

# Objective 1

# I identified the videogames that I want to find on my dataset

# I know the year of release and the name of each of them,

# I got this info from the literature review

# Now I am going to filter on my dataset to find the sales for each of them

Space\_game <- video\_games$Name == 'Space Invaders'

video\_games[Space\_game,]

# We have 2.53 Million Units for Space Invaders sold globally

# For the release year 1978

Kong\_game <- video\_games$Name == 'Donkey Kong'

video\_games[Kong\_game,]

# We have 3.07 Million Units for Donkey Kong sold globally

# For the release year 1981

Pac\_game <- video\_games$Name == 'Pac-Man'

video\_games[Pac\_game,]

# We have 7.81 Million Units for Pac-Man sold globally

# For the release year 1982

Mario\_game <- video\_games$Name == 'Super Mario Bros.'

video\_games[Mario\_game,]

# We have 40.24 Million Units for Super Mario Bros. sold globally

# For the release year 1985

Zelda\_game <- video\_games$Name == 'The Legend of Zelda'

video\_games[Zelda\_game,]

# We have 6.51 Million Units for The Legend of Zelda sold globally

# For the release year 1986

Metroid\_game <- video\_games$Name == 'Metroid'

video\_games[Metroid\_game,]

# We have 2.73 Million Units for Metroid sold globally

# For the release year 1986

Castle\_game <- video\_games$Name == 'Castlevania'

video\_games[Castle\_game,]

# We have 1.23 Million Units for Castlevania sold globally

# For the release year 1986

Sonic\_game <- video\_games$Name == 'Sonic the Hedgehog'

video\_games[Sonic\_game,]

# We have 4.34 Million Units for Sonic the Hedgehog sold globally

# For the release year 1991

Dragon\_game <- video\_games$Name == 'Dragon Quest V: Tenkuu no Hanayome'

video\_games[Dragon\_game,]

# We have 2.79 Million Units for Dragon Quest V: Tenkuu no Hanayome sold globally

# For the release year 1992

Tetris\_game <- video\_games$Name == 'Tetris'

video\_games[Tetris\_game,]

# We have 30.26 Million Units for Tetris sold globally

# For the release year 1992

Street\_game <- video\_games$Name == 'Street Fighter II: The World Warrior'

video\_games[Street\_game,]

# We have 6.3 Million Units for Street Fighter II: The World Warrior sold globally

# For the release year 1992

Kombat\_game <- video\_games$Name == 'Mortal Kombat'

video\_games[Kombat\_game,]

# We have 2.67 Million Units for Mortal Kombat sold globally

# For the release year 1992

Flight\_game <- video\_games$Name == 'Microsoft Flight Simulator'

video\_games[Flight\_game,]

# We have 5.12 Million Units for Microsoft Flight Simulator sold globally

# For the release year 1996

Mario2\_game <- video\_games$Name == 'Super Mario 64'

video\_games[Mario2\_game,]

# We have 11.89 Million Units for Super Mario 64 sold globally

# For the release year 1996

Spyros\_game <- video\_games$Name == 'Skylanders: Spyros Adventure'

video\_games[Spyros\_game,]

# We have 5.37 Million Units for Skylanders: Spyros Adventure sold globally

# For the release year 2011

Disney\_game <- video\_games$Name == 'Disney Infinity'

video\_games[Disney\_game,]

# We have 5 Million Units for Disney Infinity sold globally

# For the release year 2013

# Create a bar chart in R

# Create the data for the chart

H <- c(2.53,3.07,7.81,40.24,6.51, 2.73, 1.23, 4.34,

2.79, 30.26, 6.3, 2.67, 5.12, 11.89, 5.37, 5)

M <- c("SI 78","DK 81","PM 82","MB 85","Lz 86",

"Me 86", "CV 86", "So 91",

"DQ 92", "Te 92", "SF 92", "MK 92",

"FS 96", "M64 96", "Sp 11", "DI 13"

)

# Give the chart file a name

png(file = "historical\_videogames.png")

# Plot the bar chart

barplot(H, names.arg=M ,xlab="Game", ylab="Units (Millions)", col="blue",

main="Historical Video Games", border="red")

# Objective 2

# I am going to calculate the total sales for all genres

# Then I will get the total sales for each genre

# And compare the results

# Total sales for all genres

Total\_sales <- sum(video\_games$Global\_Sales)

# Total\_sales gives me a result of 8987.1 Million units as of Jan 2017

# Getting the total sum of sales for Action games

Action\_games <- video\_games$Genre == 'Action'

video\_games[Action\_games,10]

Action\_sales <- sum(video\_games[Action\_games,10])

Action\_sales

# Getting the total sum of sales for Adventure games

Adventure\_games <- video\_games$Genre == 'Adventure'

video\_games[Adventure\_games,10]

Adventure\_sales <- sum(video\_games[Adventure\_games,10])

Adventure\_sales

# Getting the total sum of sales for Fighting games

Fighting\_games <- video\_games$Genre == 'Fighting'

video\_games[Fighting\_games,10]

Fighting\_sales <- sum(video\_games[Fighting\_games,10])

Fighting\_sales

# Getting the total sum of sales for Misc games

Misc\_games <- video\_games$Genre == 'Misc'

video\_games[Misc\_games,10]

Misc\_sales <- sum(video\_games[Misc\_games,10])

Misc\_sales

# Getting the total sum of sales for Platform games

Platform\_games <- video\_games$Genre == 'Platform'

video\_games[Platform\_games,10]

Platform\_sales <- sum(video\_games[Platform\_games,10])

Platform\_sales

# Getting the total sum of sales for Puzzle games

Puzzle\_games <- video\_games$Genre == 'Puzzle'

video\_games[Puzzle\_games,10]

Puzzle\_sales <- sum(video\_games[Puzzle\_games,10])

Puzzle\_sales

# Getting the total sum of sales for Racing games

Racing\_games <- video\_games$Genre == 'Racing'

video\_games[Racing\_games,10]

Racing\_sales <- sum(video\_games[Racing\_games,10])

Racing\_sales

# Getting the total sum of sales for Role-Playing games

Role\_games <- video\_games$Genre == 'Role-Playing'

video\_games[Role\_games,10]

Role\_sales <- sum(video\_games[Role\_games,10])

Role\_sales

# Getting the total sum of sales for Shooter games

Shooter\_games <- video\_games$Genre == 'Shooter'

video\_games[Shooter\_games,10]

Shooter\_sales <- sum(video\_games[Shooter\_games,10])

Shooter\_sales

# Getting the total sum of sales for Simulation games

Simulation\_games <- video\_games$Genre == 'Simulation'

video\_games[Simulation\_games,10]

Simulation\_sales <- sum(video\_games[Simulation\_games,10])

Simulation\_sales

# Getting the total sum of sales for Sports games

Sports\_games <- video\_games$Genre == 'Sports'

video\_games[Sports\_games,10]

Sports\_sales <- sum(video\_games[Sports\_games,10])

Sports\_sales

# Getting the total sum of sales for Strategy games

Strategy\_games <- video\_games$Genre == 'Strategy'

video\_games[Strategy\_games,10]

Strategy\_sales <- sum(video\_games[Strategy\_games,10])

Strategy\_sales

# Action - 1757.7 Million units sold as of Jan 2017, percentage:

acpe <- (1757.7\*100)/8987.1

acpe # 19.56

# Adventure - 241.55 Million units sold as of Jan 2017, percentage:

adpe <- (241.55\*100)/8987.1

adpe # 2.69

# Fighting - 449.17 Million units sold as of Jan 2017, percentage:

fipe <- (449.17\*100)/8987.1

fipe # 5

# Miscellany - 808.8 Million units sold as of Jan 2017, percentage:

mipe <- (808.8\*100)/8987.1

mipe # 9

# Platform - 831.74 Million units sold as of Jan 2017, percentage:

plpe <- (831.74\*100)/8987.1

plpe # 9.25

# Puzzle - 243.76 Million units sold as of Jan 2017, percentage:

pupe <- (243.76\*100)/8987.1

pupe # 2.71

# Racing - 731.67 Million units sold as of Jan 2017, percentage:

rape <- (731.67\*100)/8987.1

rape # 8.14

# Role-Playing - 945.85 Million units sold as of Jan 2017, percentage:

rppe <- (945.85\*100)/8987.1

rppe # 10.52

# Shooter - 1067.3 Million units sold as of Jan 2017, percentage:

shpe <- (1067.3\*100)/8987.1

shpe # 11.88

# Simulation - 392.84 Million units sold as of Jan 2017, percentage:

sipe <- (392.84\*100)/8987.1

sipe # 4.37

# Sports - 1341 Million units sold as of Jan 2017, percentage:

sppe <- (1341\*100)/8987.1

sppe # 14.92

# Strategy - 175.72 Million units sold as of Jan 2017, percentage:

stpe <- (175.72\*100)/8987.1

stpe # 1.96

# Creating a matrix with the data

sales\_by\_genre <- matrix(c(19.56, 2.69, 5, 9, 9.25, 2.71, 8.14, 10.52,

11.88, 4.37, 14.92, 1.96), byrow=TRUE, nrow=1, ncol=12)

rownames (sales\_by\_genre) <- c("% units sold")

colnames (sales\_by\_genre) <- c("Action", "Adventure", "Fighting",

"Miscellany", "Platform", "Puzzle",

"Racing", "Role-Playing", "Shooter",

"Simulation", "Sports", "Strategy")

sales\_by\_genre

# Write a CSV file that I can manipulate

write.csv(sales\_by\_genre,"C:\\Users\\Owner\\Desktop\\datasets\\sales\_genre.csv", row.names = TRUE)

# Objective 3

# I need to filter games whose year of release is greater or equal to 2005

Modern\_games <- video\_games$Year\_of\_Release >= '2005'

# Next I get only the columns I need (Platform, Global\_sales)

Sales\_platform <- video\_games[Modern\_games, c(2, 10)]

# I now need to calculate the total global sales by platform

# I am going to discard Nintendo 64 and Dreamcast games

# Because these consoles are too old for this analysis

# I am also discarding PC because it is not a single platform 'per se'

# Nintendo 3DS

TDS\_sales <- Sales\_platform$Platform == "3DS"

TDS\_total\_sales <- sum(Sales\_platform[TDS\_sales,2])

TDS\_total\_sales # Total of 270.94

# Nintendo DS

DS\_sales <- Sales\_platform$Platform == "DS"

DS\_total\_sales <- sum(Sales\_platform[DS\_sales,2])

DS\_total\_sales # Total of 791.52

# Game Boy Advanced

GBA\_sales <- Sales\_platform$Platform == "GBA"

GBA\_total\_sales <- sum(Sales\_platform[GBA\_sales,2])

GBA\_total\_sales # Total of 43.82

# Nintendo Game Cube

GC\_sales <- Sales\_platform$Platform == "GC"

GC\_total\_sales <- sum(Sales\_platform[GC\_sales,2])

GC\_total\_sales # Total of 39.95

# PlayStation 2

PS2\_sales <- Sales\_platform$Platform == "PS2"

PS2\_total\_sales <- sum(Sales\_platform[PS2\_sales,2])

PS2\_total\_sales # Total of 432.86

# PlayStation 3

PS3\_sales <- Sales\_platform$Platform == "PS3"

PS3\_total\_sales <- sum(Sales\_platform[PS3\_sales,2])

PS3\_total\_sales # Total of 941.27

# PlayStation 4

PS4\_sales <- Sales\_platform$Platform == "PS4"

PS4\_total\_sales <- sum(Sales\_platform[PS4\_sales,2])

PS4\_total\_sales # Total of 340.79

# PSP

PSP\_sales <- Sales\_platform$Platform == "PSP"

PSP\_total\_sales <- sum(Sales\_platform[PSP\_sales,2])

PSP\_total\_sales # Total of 288.46

# PS Vita

PSV\_sales <- Sales\_platform$Platform == "PSV"

PSV\_total\_sales <- sum(Sales\_platform[PSV\_sales,2])

PSV\_total\_sales # Total of 57.03

# Wii

Wii\_sales <- Sales\_platform$Platform == "Wii"

Wii\_total\_sales <- sum(Sales\_platform[Wii\_sales,2])

Wii\_total\_sales # Total of 910.18

# WiiU

WiiU\_sales <- Sales\_platform$Platform == "WiiU"

WiiU\_total\_sales <- sum(Sales\_platform[WiiU\_sales,2])

WiiU\_total\_sales # Total of 84.93

# X-box

X\_sales <- Sales\_platform$Platform == "X"

X\_total\_sales <- sum(Sales\_platform[X\_sales,2])

X\_total\_sales # Total of 60.43

# X-box 360

X360\_sales <- Sales\_platform$Platform == "X360"

X360\_total\_sales <- sum(Sales\_platform[X360\_sales,2])

X360\_total\_sales # Total of 973.39

# X-box One

XOne\_sales <- Sales\_platform$Platform == "XOne"

XOne\_total\_sales <- sum(Sales\_platform[XOne\_sales,2])

XOne\_total\_sales # Total of 173.82

# Creating a data frame with results

Platforms <- c('3DS', 'DS', 'GBA', 'GC', 'PS2', 'PS3','PS4',

'PSP', 'PSV', 'Wii', 'WiiU', 'X-box','X-360', 'X-One')

PSales <- c(270.94, 791.52, 43.82, 39.95, 432.86, 941.27, 340.79, 288.46,

57.03, 910.18, 84.93, 60.43, 973.39, 173.82)

platform\_data <- data.frame(Platforms, PSales)

platform\_data

# Creating a lollipop plot

library('ggplot2')

ggplot(platform\_data, aes(x=Platforms, y=PSales)) +

geom\_point() +

geom\_segment( aes(x=Platforms, xend=Platforms, y=0, yend=Sales))

# Objective 4

# Getting the total sales by region

NA\_sales <- sum(video\_games[,6])

NA\_sales # Result 4429.49

EU\_sales <- sum(video\_games[,7])

EU\_sales # Result 2448.77

JP\_sales <- sum(video\_games[,8])

JP\_sales # Result 1305.2

Other\_sales <- sum(video\_games[,9])

Other\_sales # Result 798.76

# Install package to create a Treemap

install.packages('treemap')

# Use package

library('treemap')

# Create data

group <- c("NA","EU","JP", "Other")

UnitsSold <- c(4429.49, 2448.77, 1305.2,798.76)

Region\_sales <- data.frame(group,UnitsSold)

# Treemap

treemap(Region\_sales,

index="group",

vSize="UnitsSold",

type="index"

)

# Bibliography

1. **DATASET 1:** Yan, J. (2019): makeovermonday 2019/W25, *data.world*. Available at: <https://data.world/makeovermonday/2019w25/> [Accessed 25/04/2020] DATA SOURCE: <http://insideairbnb.com/get-the-data.html>
2. **DATASET 2:** Hill, Z. (2017): “project3”, *data.world*. Available at: <https://data.world/zacharius/project3/> [Accessed 25/04/20]
3. Azevedo, A. and Santos, M. F. (2008): “KDD, semma and CRISP-DM: A parallel overview”, *IADIS European Conference on Data Mining 2008, Amsterdam, The Netherlands, July 24-26, 2008*, Available at <https://www.researchgate.net/publication/220969845_KDD_semma_and_CRISP-DM_A_parallel_overview> [Accessed 25/04/20]
4. Wong, A (2016): “Airbnb y hoteles: ¿cuál es el más económico?”, *busbud*, Available at: <https://www.busbud.com/blog/es/airbnb-y-hoteles-precio-comparacion/> [Accessed 25/04/20)
5. Sthapit, E, and Björk, P. (2019): “Sources of distrust: Airbnb guests' perspectives”, *Tourism Management Perspectives, Volume 31, July 2019,* Pages 245-253 [Online]. Available at: <https://www.sciencedirect.com/science/article/pii/S2211973619300637> [Accessed 25/04/20]
6. Toister, J. (2018): Are customer expectations of email response times unreasonable?, *MYcustomer*. Available at: <https://www.mycustomer.com/service/channels/are-customer-expectations-of-email-response-times-unreasonable> [Accessed 25/04/20]
7. *What are response rate and response time and how are they calculated?,* 2020, Airbnb Help Center. Available at: <https://www.airbnb.com/help/article/430/what-are-response-rate-and-response-time-and-how-are-they-calculated> [Accessed 25/04/20]
8. Allibert, B (2016): “The Impact Of The AirBnB Ban In Berlin – 6 Months After”, *Settle in Berlin*. Available at: <https://www.settle-in-berlin.com/airbnb-ban-berlin/> [Accessed 25/04/20]
9. Skowronnek, A., Vogel, L. and Parnow, J. (2015): AIRBNB vs. BERLIN What data tell us, *FH Potsdam, Design Faculty*. Available at: <http://airbnbvsberlin.com/> [Accessed 25/04/20]
10. Ranson, P. (2019): How an Airbnb host can use behavioural science to influence guests review scores, *Medium*. Available at: <https://medium.com/@paulranson/how-an-airbnb-host-can-use-behavioural-science-to-influence-guests-review-scores-553a407caf78> [Accessed 25/04/20]
11. Aleem, S., Capretz, L. F. and Ahmed, F. (2016): “Empirical investigation of key business factors for digital game performance”, *Entertainment Computing. Volume 13, Pages 25-36* [Online]. Available at: <https://arxiv.org/ftp/arxiv/papers/1511/1511.04422.pdf> [Accessed 25/04/20]
12. History.com Editors (2019): Video Game History, *A&E Television Networks*. Available at: <https://www.history.com/topics/inventions/history-of-video-games> [Accessed 25/04/20]
13. Ballard, J. (2018): “5 Trends Explain the Growth of the Video-Game Industry”, *The Motley Fool*. Available at: <https://www.fool.com/investing/2018/11/09/5-trends-explain-the-growth-of-the-video-game-indu.aspx> [accessed 25/04/20]
14. Anderson, C. A. (2003): “Violent Video Games: Myths, Facts, and Unanswered Questions”, American Psychological Association. Available at: <https://www.apa.org/science/about/psa/2003/10/anderson> [Accessed 25/04/20]
15. Richter, F. (2020): “PlayStation 2 Is the Best-Selling Console of All Time”, *statista*. Available at: <https://www.statista.com/chart/18903/video-game-console-sales/> [Accessed 25/04/20]
16. McDonald, E. (2017): “The Global Games Market Will Reach $108.9 Billion in 2017 With Mobile Taking 42%”, *newzoo*. Available at: <https://newzoo.com/insights/articles/the-global-games-market-will-reach-108-9-billion-in-2017-with-mobile-taking-42/> [Accessed 25/04/20]
17. Gough, C. (2019): “Games market revenue share worldwide 2019, by region”, *statista*. Available at: <https://www.statista.com/statistics/983204/global-video-games-revenue-region/> [Accessed 25/04/20]

1. Azevedo and Santos (2008, p. 183) [↑](#footnote-ref-1)
2. You can find detailed information on the exact columns that I deleted on the dataset description tables. [↑](#footnote-ref-2)
3. Wong (2016) [↑](#footnote-ref-3)
4. Sthapit and Björk (2019, p. 245-253) [↑](#footnote-ref-4)
5. Toister (2018) [↑](#footnote-ref-5)
6. Airbnb Help Center (2020). Available at: <https://www.airbnb.com/help/article/430/what-are-response-rate-and-response-time-and-how-are-they-calculated> [Accessed 25/04/20] [↑](#footnote-ref-6)
7. Allibert (2016) [↑](#footnote-ref-7)
8. Skowronnek, Vogel and Parnow (2015) [↑](#footnote-ref-8)
9. Yan, J. (2019): makeovermonday 2019/W25, *data.world*. Available at: <https://data.world/makeovermonday/2019w25/> [Accessed 25/04/2020] DATA SOURCE: <http://insideairbnb.com/get-the-data.html>. [↑](#footnote-ref-9)
10. Ranson (2019) [↑](#footnote-ref-10)
11. Aleem, Capretz and Ahmed (2016) [↑](#footnote-ref-11)
12. History.com Editors (2019): Video Game History, *A&E Television Networks*. Available at: <https://www.history.com/topics/inventions/history-of-video-games> [Accessed 25/04/20] [↑](#footnote-ref-12)
13. Ballard (2018) [↑](#footnote-ref-13)
14. Anderson (2003) [↑](#footnote-ref-14)
15. Richter (2020) [↑](#footnote-ref-15)
16. McDonald (2017) [↑](#footnote-ref-16)
17. Gough (2019) [↑](#footnote-ref-17)
18. Hill, Z. (2017): “project3”, *data.world*. Available at: <https://data.world/zacharius/project3/> [Accessed 25/04/20] [↑](#footnote-ref-18)