

# Airbnb\_project

leonard

2024-02-24

## Data Analytics Case Study with R : airbnb data analysis

### Introduction

Welcome to my Airbnb data analysis case study! In this case study, I will perform real-world tasks of a data analyst. And in order to answer some key business questions, I will follow the steps of the data analysis process: ask, prepare, process, analyze, share, and act.

### Scenario of the study

Imagine a real estate management company wants to determine if they should expand into a new geographic area. Use public data from Airbnb to answer questions about the vacation rental market in a specific area. Investigate how neighborhoods or amenities influence Airbnb prices, what listings are rented most frequently, and where super hosts are located

### Identifying the business task and questions to be answered (ask phase)

#### 1. Impact of Neighborhoods on Airbnb Prices:

Question: How do different neighborhoods influence Airbnb prices? Answering: Analyze the average rental prices across different neighborhoods, conduct statistical tests to compare price distributions, and visualize price variations on a map.

2. Influence of Amenities on Airbnb Prices:

Question: What amenities have the most significant impact on Airbnb prices? Answering: Perform regression analysis to identify amenities (such as pool, parking, Wi-Fi) that correlate with higher prices and quantify their impact on rental rates.

3. Analysis of Most Frequently Rented Listings:

Question: Which listings are rented most frequently, and what factors contribute to their popularity? Answering: Calculate rental frequency metrics for listings, examine listing attributes (e.g., location, size, amenities), and analyze guest reviews to identify key factors driving demand.

4. Location of Super Hosts:

Question: Where are super hosts located within the target area, and what distinguishes their listings? Answering: Map the distribution of super hosts, analyze their listing characteristics (e.g., price, ratings, response rate), and compare them to regular hosts to understand the factors contributing to super host status.

## **Statement of Business Task:**

The real estate management company seeks to evaluate the feasibility of expanding into a new geographic area within the vacation rental market. Utilizing public data from Airbnb, the company aims to answer key questions regarding the rental market in the target area. The analysis will focus on understanding the impact of neighborhoods and amenities on Airbnb prices, identifying the most frequently rented listings, and pinpointing the locations of super hosts within the area. By investigating these factors, the company aims to make data-driven decisions regarding the potential expansion and optimize their market entry strategy for maximum success.

### **key stakeholders**

**Real Estate Management Company:** The company seeking to expand into the vacation rental market is the primary stakeholder. They are responsible for making strategic decisions based on the analysis of Airbnb data to determine the feasibility and potential success of expanding into the new geographic area.

**Investors and Shareholders:** Investors and shareholders of the real estate management company have a vested interest in the company's expansion plans. They rely on the analysis of Airbnb data to assess the potential return on investment and make decisions regarding funding and resource allocation.

**Property Owners and Hosts:** Property owners and hosts who list their properties on Airbnb are important stakeholders. They may be impacted by the company's expansion into the new geographic area, as it could affect rental demand, pricing dynamics, and competition within the market.

**Local Community and Residents:** The local community and residents of the new geographic area are stakeholders who may be affected by the company's expansion. Changes in the vacation rental market can have implications for housing affordability, neighborhood dynamics, and community relations.

**Regulatory Authorities and Government Agencies:** Regulatory authorities and government agencies responsible for overseeing the vacation rental market may also be stakeholders. They may be interested in the company's expansion plans and the potential impact on local regulations, zoning laws, and taxation policies.

**Tourism and Hospitality Industry:** Stakeholders within the broader tourism and hospitality industry, including hotels, restaurants, and tourist attractions, may be impacted by the company's expansion. Changes in the vacation rental market can affect tourism patterns, visitor spending, and overall industry dynamics.

## **Preparing the Data (Prepare Phase)**

In this phase, I will download and Import the dataset. Then make sure all the data is organized and credible. And I will sort and filter the data.

### **install packages**

Now, I'm going to Install some R packages that will help me in my analysis. And I'm using in my code the options message=FALSE and warning=FALSE, to save space. And to prevent printing of the execution of the R code generated and the warning messages. And I will add some data cleaning packages as well (last 3 packages)

### **load packages**

Now, I'm going to load these packages. And I'm using in my code the options message=FALSE and warning=FALSE, to save space. And to prevent printing of the execution of the R code generated and the warning

messages.

```
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 4.3.2

## Warning: package 'ggplot2' was built under R version 4.3.2

## Warning: package 'tidyverse' was built under R version 4.3.2

## Warning: package 'dplyr' was built under R version 4.3.2

## Warning: package 'stringr' was built under R version 4.3.2

## Warning: package 'lubridate' was built under R version 4.3.2

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr     1.1.4      v readr     2.1.4
## vforcats   1.0.0      v stringr   1.5.1
## v ggplot2   3.5.0      v tibble    3.2.1
## v lubridate 1.9.3      v tidyverse 1.3.1
## v purrr    1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(lubridate)
library(dplyr)
library(ggplot2)
library(tidyverse)
library(here)
```

```
## Warning: package 'here' was built under R version 4.3.2

## here() starts at C:/Users/HP/Documents/R STUDIO/case_studies
```

```
library(skimr)
```

```
## Warning: package 'skimr' was built under R version 4.3.2
```

```
library(janitor)
```

```
## Warning: package 'janitor' was built under R version 4.3.2
```

```
##
## Attaching package: 'janitor'
##
## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

## Importing dataset

Now, I'm going to Import all dataset. Then VIEW, CLEAN, FORMAT, and ORGANIZE the data. After reviewing all the dataset, I decided to make some assumptions and work only with these data for my analysis:

```
amsterdam_weekdays <- read_csv("~/R STUDIO/case_studies/airbnb/amsterdam_weekdays.csv")
```

```
## New names:  
## Rows: 1103 Columns: 20  
## -- Column specification  
## ----- Delimiter: "," chr  
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,  
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost  
## i Use 'spec()' to retrieve the full column specification for this data. i  
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## * ' ' -> '...1'
```

```
head(amsterdam_weekdays)
```

```
## # A tibble: 6 x 20  
##   ...1 realSum room_type    room_shared room_private person_capacity  
##   <dbl>    <dbl> <chr>        <lgl>      <lgl>           <dbl>  
## 1     0    194. Private room FALSE       TRUE            2  
## 2     1    344. Private room FALSE       TRUE            4  
## 3     2    264. Private room FALSE       TRUE            2  
## 4     3    434. Private room FALSE       TRUE            4  
## 5     4    486. Private room FALSE       TRUE            2  
## 6     5    553. Private room FALSE       TRUE            3  
## # i 14 more variables: host_is_superhost <lgl>, multi <dbl>, biz <dbl>,  
## #   cleanliness_rating <dbl>, guest_satisfaction_overall <dbl>, bedrooms <dbl>,  
## #   dist <dbl>, metro_dist <dbl>, attr_index <dbl>, attr_index_norm <dbl>,  
## #   rest_index <dbl>, rest_index_norm <dbl>, lng <dbl>, lat <dbl>
```

```
colnames(amsterdam_weekdays)
```

```
## [1] "...1"                      "realSum"  
## [3] "room_type"                  "room_shared"  
## [5] "room_private"                "person_capacity"  
## [7] "host_is_superhost"          "multi"  
## [9] "biz"                         "cleanliness_rating"  
## [11] "guest_satisfaction_overall" "bedrooms"  
## [13] "dist"                        "metro_dist"  
## [15] "attr_index"                 "attr_index_norm"  
## [17] "rest_index"                 "rest_index_norm"  
## [19] "lng"                         "lat"
```

```
str(amsterdam_weekdays)
```

```
## spc_tbl_ [1,103 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)  
## $ ...1             : num [1:1103] 0 1 2 3 4 5 6 7 8 9 ...  
## $ realSum          : num [1:1103] 194 344 264 434 486 ...
```

```

## $ room_type : chr [1:1103] "Private room" "Private room" "Private room" "Private room"
## $ room_shared : logi [1:1103] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private : logi [1:1103] TRUE TRUE TRUE TRUE TRUE TRUE ...
## $ person_capacity : num [1:1103] 2 4 2 4 2 3 2 4 4 2 ...
## $ host_is_superhost : logi [1:1103] FALSE FALSE FALSE FALSE TRUE FALSE ...
## $ multi : num [1:1103] 1 0 0 0 0 0 0 0 0 1 ...
## $ biz : num [1:1103] 0 0 1 1 0 0 0 0 0 0 ...
## $ cleanliness_rating : num [1:1103] 10 8 9 9 10 8 10 10 9 10 ...
## $ guest_satisfaction_overall: num [1:1103] 93 85 87 90 98 100 94 100 96 88 ...
## $ bedrooms : num [1:1103] 1 1 1 2 1 2 1 3 2 1 ...
## $ dist : num [1:1103] 5.023 0.488 5.748 0.385 0.545 ...
## $ metro_dist : num [1:1103] 2.539 0.239 3.652 0.44 0.319 ...
## $ attr_index : num [1:1103] 78.7 631.2 75.3 493.3 552.8 ...
## $ attr_index_norm : num [1:1103] 4.17 33.42 3.99 26.12 29.27 ...
## $ rest_index : num [1:1103] 98.3 837.3 95.4 875 815.3 ...
## $ rest_index_norm : num [1:1103] 6.85 58.34 6.65 60.97 56.81 ...
## $ lng : num [1:1103] 4.91 4.9 4.98 4.89 4.9 ...
## $ lat : num [1:1103] 52.4 52.4 52.4 52.4 52.4 ...
## - attr(*, "spec")=
##   .. cols(
##     ... .1 = col_double(),
##     ... realSum = col_double(),
##     ... room_type = col_character(),
##     ... room_shared = col_logical(),
##     ... room_private = col_logical(),
##     ... person_capacity = col_double(),
##     ... host_is_superhost = col_logical(),
##     ... multi = col_double(),
##     ... biz = col_double(),
##     ... cleanliness_rating = col_double(),
##     ... guest_satisfaction_overall = col_double(),
##     ... bedrooms = col_double(),
##     ... dist = col_double(),
##     ... metro_dist = col_double(),
##     ... attr_index = col_double(),
##     ... attr_index_norm = col_double(),
##     ... rest_index = col_double(),
##     ... rest_index_norm = col_double(),
##     ... lng = col_double(),
##     ... lat = col_double()
##   ... )
## - attr(*, "problems")=<externalptr>

```

```
amsterdam_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/amsterdam_weekends.csv")
```

```

## New names:
## Rows: 977 Columns: 20
## -- Column specification
##   ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use `spec()` to retrieve the full column specification for this data. i
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## * ' ' -> '...1'

```

```
head(amsterdam_weekends)
```

```
## # A tibble: 6 x 20
##   ...1 realSum room_type      room_shared room_private person_capacity
##   <dbl>    <dbl> <chr>        <lgl>       <lgl>                  <dbl>
## 1     0     320. Private room FALSE        TRUE                   2
## 2     1     348. Private room FALSE        TRUE                   2
## 3     2     483. Private room FALSE        TRUE                   4
## 4     3     486. Private room FALSE        TRUE                   2
## 5     4    2772. Entire home/apt FALSE       FALSE                  4
## 6     5    1002. Entire home/apt FALSE       FALSE                  4
## # i 14 more variables: host_is_superhost <lgl>, multi <dbl>, biz <dbl>,
## #   cleanliness_rating <dbl>, guest_satisfaction_overall <dbl>, bedrooms <dbl>,
## #   dist <dbl>, metro_dist <dbl>, attr_index <dbl>, attr_index_norm <dbl>,
## #   rest_index <dbl>, rest_index_norm <dbl>, lng <dbl>, lat <dbl>
```

```
colnames(amsterdam_weekends)
```

```
## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"
```

```
str(amsterdam_weekends)
```

```
## #> #> spc_tbl_ [977 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## #> #> $ ...1                      : num [1:977] 0 1 2 3 4 5 6 7 8 9 ...
## #> #> $ realSum                  : num [1:977] 320 348 483 486 2772 ...
## #> #> $ room_type                 : chr [1:977] "Private room" "Private room" "Private room" "Private room"
## #> #> $ room_shared                : logi [1:977] FALSE FALSE FALSE FALSE FALSE FALSE ...
## #> #> $ room_private               : logi [1:977] TRUE TRUE TRUE TRUE FALSE FALSE ...
## #> #> $ person_capacity            : num [1:977] 2 2 4 2 4 4 2 2 2 2 ...
## #> #> $ host_is_superhost          : logi [1:977] FALSE FALSE FALSE TRUE TRUE FALSE ...
## #> #> $ multi                     : num [1:977] 0 0 0 0 0 0 1 0 1 ...
## #> #> $ biz                        : num [1:977] 1 1 1 0 0 0 0 0 0 ...
## #> #> $ cleanliness_rating          : num [1:977] 9 9 9 10 10 9 10 10 10 10 ...
## #> #> $ guest_satisfaction_overall: num [1:977] 88 87 90 98 100 96 99 88 96 97 ...
## #> #> $ bedrooms                  : num [1:977] 1 1 2 1 3 2 1 1 1 1 ...
## #> #> $ dist                       : num [1:977] 4.763 5.748 0.385 0.545 1.687 ...
## #> #> $ metro_dist                 : num [1:977] 0.852 3.652 0.44 0.319 1.458 ...
## #> #> $ attr_index                 : num [1:977] 110.9 75.3 493.3 552.8 208.8 ...
## #> #> $ attr_index_norm            : num [1:977] 5.87 3.99 26.12 29.27 11.06 ...
## #> #> $ rest_index                 : num [1:977] 137 95.4 875.1 815.3 272.3 ...
## #> #> $ rest_index_norm            : num [1:977] 11.94 8.32 76.29 71.07 23.74 ...
## #> #> $ lng                        : num [1:977] 4.85 4.98 4.89 4.9 4.88 ...
## #> #> $ lat                        : num [1:977] 52.3 52.4 52.4 52.4 52.4 ...
```

```

## - attr(*, "spec")=
##   .. cols(
##     ... .1 = col_double(),
##     ... realSum = col_double(),
##     ... room_type = col_character(),
##     ... room_shared = col_logical(),
##     ... room_private = col_logical(),
##     ... person_capacity = col_double(),
##     ... host_is_superhost = col_logical(),
##     ... multi = col_double(),
##     ... biz = col_double(),
##     ... cleanliness_rating = col_double(),
##     ... guest_satisfaction_overall = col_double(),
##     ... bedrooms = col_double(),
##     ... dist = col_double(),
##     ... metro_dist = col_double(),
##     ... attr_index = col_double(),
##     ... attr_index_norm = col_double(),
##     ... rest_index = col_double(),
##     ... rest_index_norm = col_double(),
##     ... lng = col_double(),
##     ... lat = col_double()
##   ... )
## - attr(*, "problems")=<externalptr>

```

```
athens_weekdays <- read_csv("~/R STUDIO/case_studies/airbnb/athens_weekdays.csv")
```

```

## New names:
## Rows: 2653 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * `--> `...1'

```

```
head(athens_weekdays)
```

```

## # A tibble: 6 x 20
##   ...1 realSum room_type      room_shared room_private person_capacity
##   <dbl>    <dbl> <chr>        <lgl>       <lgl>           <dbl>
## 1     0    130. Entire home/apt FALSE      FALSE            4
## 2     1    139. Entire home/apt FALSE      FALSE            4
## 3     2    156. Entire home/apt FALSE      FALSE            3
## 4     3    91.6 Entire home/apt FALSE      FALSE            4
## 5     4    74.1 Private room   FALSE      TRUE             2
## 6     5    114. Entire home/apt FALSE      FALSE            6
## # i 14 more variables: host_is_superhost <lgl>, multi <dbl>, biz <dbl>,
## #   cleanliness_rating <dbl>, guest_satisfaction_overall <dbl>, bedrooms <dbl>,
## #   dist <dbl>, metro_dist <dbl>, attr_index <dbl>, attr_index_norm <dbl>,
## #   rest_index <dbl>, rest_index_norm <dbl>, lng <dbl>, lat <dbl>

```

```
colnames(athens_weekdays)
```

```
## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"           "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                  "attr_index_norm"
## [17] "rest_index"                  "rest_index_norm"
## [19] "lng"                         "lat"
```

```
str(athens_weekdays)
```

```
## spc_tbl_ [2,653 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1
## $ realSum
## $ room_type
## $ room_shared
## $ room_private
## $ person_capacity
## $ host_is_superhost
## $ multi
## $ biz
## $ cleanliness_rating
## $ guest_satisfaction_overall
## $ bedrooms
## $ dist
## $ metro_dist
## $ attr_index
## $ attr_index_norm
## $ rest_index
## $ rest_index_norm
## $ lng
## $ lat
## - attr(*, "spec")=
##   .. cols(
##     ..   ...1 = col_double(),
##     ..   realSum = col_double(),
##     ..   room_type = col_character(),
##     ..   room_shared = col_logical(),
##     ..   room_private = col_logical(),
##     ..   person_capacity = col_double(),
##     ..   host_is_superhost = col_logical(),
##     ..   multi = col_double(),
##     ..   biz = col_double(),
##     ..   cleanliness_rating = col_double(),
##     ..   guest_satisfaction_overall = col_double(),
##     ..   bedrooms = col_double(),
##     ..   dist = col_double(),
##     ..   metro_dist = col_double(),
```

```

## .. attr_index = col_double(),
## .. attr_index_norm = col_double(),
## .. rest_index = col_double(),
## .. rest_index_norm = col_double(),
## .. lng = col_double(),
## .. lat = col_double()
## ...
## - attr(*, "problems")=<externalptr>

```

```
athens_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/athens_weekends.csv")
```

```

## New names:
## Rows: 2627 Columns: 20
## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(athens_weekends)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"               "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(athens_weekends)
```

```

## spc_tbl_ [2,627 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1                      : num [1:2627] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum                    : num [1:2627] 139 91.6 76.6 151.9 98.7 ...
## $ room_type                  : chr [1:2627] "Entire home/apt" "Entire home/apt" "Private room" "Enti...
## $ room_shared                 : logi [1:2627] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private                : logi [1:2627] FALSE FALSE TRUE FALSE FALSE FALSE ...
## $ person_capacity              : num [1:2627] 4 4 2 4 2 4 3 6 5 4 ...
## $ host_is_superhost           : logi [1:2627] TRUE TRUE FALSE TRUE TRUE TRUE ...
## $ multi                       : num [1:2627] 1 1 0 0 1 1 0 1 0 0 ...
## $ biz                          : num [1:2627] 0 0 0 1 0 0 0 0 1 ...
## $ cleanliness_rating           : num [1:2627] 10 10 10 10 10 10 10 10 10 9 ...
## $ guest_satisfaction_overall: num [1:2627] 96 99 100 100 95 97 95 98 97 88 ...
## $ bedrooms                     : num [1:2627] 1 1 1 2 1 1 2 2 3 1 ...
## $ dist                         : num [1:2627] 0.407 4.367 2.194 2.509 2.741 ...
## $ metro_dist                   : num [1:2627] 0.305 0.297 0.385 0.563 0.725 ...
## $ attr_index                   : num [1:2627] 240.3 39.8 78.7 68.8 62.9 ...

```

```

## $ attr_index_norm      : num [1:2627] 9.05 1.5 2.97 2.59 2.37 ...
## $ rest_index           : num [1:2627] 407.2 58.7 113.3 101.2 92.6 ...
## $ rest_index_norm      : num [1:2627] 6.081 0.877 1.692 1.511 1.383 ...
## $ lng                  : num [1:2627] 23.7 23.7 23.7 23.7 23.7 ...
## $ lat                  : num [1:2627] 38 38 38 38 38 ...
## - attr(*, "spec")=
##   .. cols(
##     .. .1 = col_double(),
##     .. realSum = col_double(),
##     .. room_type = col_character(),
##     .. room_shared = col_logical(),
##     .. room_private = col_logical(),
##     .. person_capacity = col_double(),
##     .. host_is_superhost = col_logical(),
##     .. multi = col_double(),
##     .. biz = col_double(),
##     .. cleanliness_rating = col_double(),
##     .. guest_satisfaction_overall = col_double(),
##     .. bedrooms = col_double(),
##     .. dist = col_double(),
##     .. metro_dist = col_double(),
##     .. attr_index = col_double(),
##     .. attr_index_norm = col_double(),
##     .. rest_index = col_double(),
##     .. rest_index_norm = col_double(),
##     .. lng = col_double(),
##     .. lat = col_double()
##   .. )
## - attr(*, "problems")=<externalptr>

barcelona_weekdays <- read_csv("~/R STUDIO/case_studies/airbnb/barcelona_weekdays.csv")

## New names:
## Rows: 1555 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

colnames(barcelona_weekdays)

```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"               "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(barcelona_weekdays)
```

```
## spc_tbl_ [1,555 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1 : num [1:1555] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum : num [1:1555] 474 170 162 368 197 ...
## $ room_type : chr [1:1555] "Entire home/apt" "Private room" "Private room" "Entire home/apt" ...
## $ room_shared : logi [1:1555] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private : logi [1:1555] FALSE TRUE TRUE FALSE TRUE FALSE ...
## $ person_capacity : num [1:1555] 4 2 4 3 3 3 3 2 4 2 ...
## $ host_is_superhost : logi [1:1555] FALSE TRUE FALSE FALSE FALSE FALSE ...
## $ multi : num [1:1555] 0 1 0 0 1 0 0 0 1 1 ...
## $ biz : num [1:1555] 1 0 1 1 0 1 1 0 0 0 ...
## $ cleanliness_rating : num [1:1555] 10 10 9 10 9 9 9 9 9 9 ...
## $ guest_satisfaction_overall: num [1:1555] 91 88 88 91 91 100 86 96 84 91 ...
## $ bedrooms : num [1:1555] 1 1 1 1 1 2 1 1 1 1 ...
## $ dist : num [1:1555] 1.11 1.75 1.67 1.48 1.86 ...
## $ metro_dist : num [1:1555] 0.6305 0.124 0.0803 0.0931 0.2725 ...
## $ attr_index : num [1:1555] 526 320 344 400 346 ...
## $ attr_index_norm : num [1:1555] 17.9 10.9 11.7 13.6 11.8 ...
## $ rest_index : num [1:1555] 916 794 841 947 792 ...
## $ rest_index_norm : num [1:1555] 20.2 17.5 18.5 20.8 17.4 ...
## $ lng : num [1:1555] 2.18 2.15 2.15 2.17 2.15 ...
## $ lat : num [1:1555] 41.4 41.4 41.4 41.4 41.4 ...
## - attr(*, "spec")=
## .. cols(
## ..   ...1 = col_double(),
## ..   realSum = col_double(),
## ..   room_type = col_character(),
## ..   room_shared = col_logical(),
## ..   room_private = col_logical(),
## ..   person_capacity = col_double(),
## ..   host_is_superhost = col_logical(),
## ..   multi = col_double(),
## ..   biz = col_double(),
## ..   cleanliness_rating = col_double(),
## ..   guest_satisfaction_overall = col_double(),
## ..   bedrooms = col_double(),
## ..   dist = col_double(),
## ..   metro_dist = col_double(),
## ..   attr_index = col_double(),
## ..   attr_index_norm = col_double(),
## ..   rest_index = col_double(),
## ..   rest_index_norm = col_double(),
## ..   lng = col_double(),
## ..   lat = col_double()
## .. )
## - attr(*, "problems")=<externalptr>
```

```
barcelona_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/barcelona_weekends.csv")
```

```
## New names:
## Rows: 1278 Columns: 20
```

```

## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' -> '...1'

```

```
colnames(barcelona_weekends)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                  "attr_index_norm"
## [17] "rest_index"                  "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(barcelona_weekends)
```

```

## spc_tbl_ [1,278 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1
##   : num [1:1278] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum
##   : num [1:1278] 532 170 197 141 715 ...
## $ room_type
##   : chr [1:1278] "Entire home/apt" "Private room" "Private room" "Private room" ...
## $ room_shared
##   : logi [1:1278] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private
##   : logi [1:1278] FALSE TRUE TRUE TRUE FALSE TRUE ...
## $ person_capacity
##   : num [1:1278] 4 2 2 3 4 2 4 5 2 2 ...
## $ host_is_superhost
##   : logi [1:1278] FALSE TRUE FALSE FALSE TRUE FALSE ...
## $ multi
##   : num [1:1278] 0 1 1 0 0 0 1 0 1 1 ...
## $ biz
##   : num [1:1278] 1 0 0 1 1 0 0 1 0 0 ...
## $ cleanliness_rating
##   : num [1:1278] 10 10 9 9 9 9 9 9 10 9 ...
## $ guest_satisfaction_overall
##   : num [1:1278] 91 88 90 86 95 96 84 87 98 91 ...
## $ bedrooms
##   : num [1:1278] 1 1 1 1 2 1 1 2 1 1 ...
## $ dist
##   : num [1:1278] 1.112 1.752 0.622 1.648 0.694 ...
## $ metro_dist
##   : num [1:1278] 0.6305 0.124 0.2975 0.0894 0.3577 ...
## $ attr_index
##   : num [1:1278] 526 320 503 343 538 ...
## $ attr_index_norm
##   : num [1:1278] 20.3 12.4 19.4 13.2 20.8 ...
## $ rest_index
##   : num [1:1278] 916 794 1242 821 1082 ...
## $ rest_index_norm
##   : num [1:1278] 20.1 17.4 27.3 18 23.8 ...
## $ lng
##   : num [1:1278] 2.18 2.15 2.16 2.15 2.17 ...
## $ lat
##   : num [1:1278] 41.4 41.4 41.4 41.4 41.4 ...
## - attr(*, "spec")=
## .. cols(
##   .. ...1 = col_double(),
##   .. realSum = col_double(),
##   .. room_type = col_character(),
##   .. room_shared = col_logical(),
##   .. room_private = col_logical(),
##   .. person_capacity = col_double(),
##   .. host_is_superhost = col_logical(),

```

```

## .. multi = col_double(),
## .. biz = col_double(),
## .. cleanliness_rating = col_double(),
## .. guest_satisfaction_overall = col_double(),
## .. bedrooms = col_double(),
## .. dist = col_double(),
## .. metro_dist = col_double(),
## .. attr_index = col_double(),
## .. attr_index_norm = col_double(),
## .. rest_index = col_double(),
## .. rest_index_norm = col_double(),
## .. lng = col_double(),
## .. lat = col_double()
## ...
## - attr(*, "problems")=<externalptr>

```

```
berlin_weekdays <- read_csv("~/R STUDIO/case_studies/airbnb/berlin_weekdays.csv")
```

```

## New names:
## Rows: 1284 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(berlin_weekdays)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"               "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(berlin_weekdays)
```

```

## spc_tbl_ [1,284 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1                      : num [1:1284] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum                    : num [1:1284] 186 195 176 208 151 ...
## $ room_type                  : chr [1:1284] "Private room" "Private room" "Private room" ...
## $ room_shared                 : logi [1:1284] FALSE FALSE FALSE FALSE FALSE ...
## $ room_private                : logi [1:1284] TRUE TRUE TRUE TRUE TRUE ...
## $ person_capacity             : num [1:1284] 2 5 2 3 2 2 6 2 2 3 ...
## $ host_is_superhost           : logi [1:1284] TRUE FALSE FALSE TRUE FALSE FALSE ...
## $ multi                       : num [1:1284] 0 0 0 0 1 0 0 1 1 ...

```

```

## $ biz : num [1:1284] 0 1 0 0 0 0 0 0 0 0 ...
## $ cleanliness_rating : num [1:1284] 10 9 9 10 10 8 10 10 10 10 ...
## $ guest_satisfaction_overall: num [1:1284] 98 86 91 97 99 87 97 96 100 98 ...
## $ bedrooms : num [1:1284] 1 1 1 1 1 1 2 1 1 1 ...
## $ dist : num [1:1284] 3.582 3.525 3.802 0.982 8.87 ...
## $ metro_dist : num [1:1284] 0.175 0.512 0.281 0.706 2.187 ...
## $ attr_index : num [1:1284] 105.1 75.3 73.7 133.2 39.9 ...
## $ attr_index_norm : num [1:1284] 16.02 11.49 11.23 20.31 6.08 ...
## $ rest_index : num [1:1284] 149 106 105 198 51 ...
## $ rest_index_norm : num [1:1284] 30.7 21.9 21.7 40.9 10.5 ...
## $ lng : num [1:1284] 13.4 13.5 13.5 13.4 13.5 ...
## $ lat : num [1:1284] 52.5 52.5 52.5 52.5 52.5 ...
## - attr(*, "spec")=
##   .. cols(
##     .. .1 = col_double(),
##     .. realSum = col_double(),
##     .. room_type = col_character(),
##     .. room_shared = col_logical(),
##     .. room_private = col_logical(),
##     .. person_capacity = col_double(),
##     .. host_is_superhost = col_logical(),
##     .. multi = col_double(),
##     .. biz = col_double(),
##     .. cleanliness_rating = col_double(),
##     .. guest_satisfaction_overall = col_double(),
##     .. bedrooms = col_double(),
##     .. dist = col_double(),
##     .. metro_dist = col_double(),
##     .. attr_index = col_double(),
##     .. attr_index_norm = col_double(),
##     .. rest_index = col_double(),
##     .. rest_index_norm = col_double(),
##     .. lng = col_double(),
##     .. lat = col_double()
##   .. )
## - attr(*, "problems")=<externalptr>

```

```
berlin_weekends <- read_csv("~/R STUDIO0/case_studies/airbnb/berlin_weekends.csv")
```

```

## New names:
## Rows: 1200 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): .1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' -> '...

```

```
colnames(berlin_weekends)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"

```

```

## [5] "room_private"           "person_capacity"
## [7] "host_is_superhost"      "multi"
## [9] "biz"                     "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                    "metro_dist"
## [15] "attr_index"              "attr_index_norm"
## [17] "rest_index"              "rest_index_norm"
## [19] "lng"                     "lat"

str(berlin_weekends)

## #> #> spc_tbl_ [1,200 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## #> #>   $ ...1                  : num [1:1200] 0 1 2 3 4 5 6 7 8 9 ...
## #> #>   $ realSum               : num [1:1200] 186 387 195 172 208 ...
## #> #>   $ room_type              : chr [1:1200] "Private room" "Entire home/apt" "Private room" "Private ...
## #> #>   $ room_shared             : logi [1:1200] FALSE FALSE FALSE FALSE FALSE ...
## #> #>   $ room_private            : logi [1:1200] TRUE FALSE TRUE TRUE TRUE TRUE ...
## #> #>   $ person_capacity         : num [1:1200] 2 6 5 2 3 2 3 2 2 2 ...
## #> #>   $ host_is_superhost       : logi [1:1200] TRUE FALSE FALSE FALSE TRUE FALSE ...
## #> #>   $ multi                  : num [1:1200] 0 0 0 0 0 0 0 0 0 0 ...
## #> #>   $ biz                     : num [1:1200] 0 1 1 0 0 0 0 0 1 0 ...
## #> #>   $ cleanliness_rating       : num [1:1200] 10 10 9 9 10 10 10 10 9 10 ...
## #> #>   $ guest_satisfaction_overall: num [1:1200] 98 93 86 91 97 99 96 98 78 96 ...
## #> #>   $ bedrooms                : num [1:1200] 1 2 1 1 1 1 1 1 1 1 ...
## #> #>   $ dist                     : num [1:1200] 3.582 6.082 3.525 3.802 0.982 ...
## #> #>   $ metro_dist               : num [1:1200] 0.175 0.481 0.512 0.281 0.706 ...
## #> #>   $ attr_index               : num [1:1200] 105.1 52.9 75.3 73.7 133.2 ...
## #> #>   $ attr_index_norm          : num [1:1200] 16.01 8.06 11.48 11.23 20.3 ...
## #> #>   $ rest_index               : num [1:1200] 148.9 66.9 106.4 105.4 198.2 ...
## #> #>   $ rest_index_norm          : num [1:1200] 31.5 14.2 22.5 22.3 41.9 ...
## #> #>   $ lng                      : num [1:1200] 13.4 13.5 13.5 13.5 13.4 ...
## #> #>   $ lat                      : num [1:1200] 52.5 52.5 52.5 52.5 52.5 ...
## #> - attr(*, "spec")=
## #> .. cols(
## #> ..   ...1 = col_double(),
## #> ..   realSum = col_double(),
## #> ..   room_type = col_character(),
## #> ..   room_shared = col_logical(),
## #> ..   room_private = col_logical(),
## #> ..   person_capacity = col_double(),
## #> ..   host_is_superhost = col_logical(),
## #> ..   multi = col_double(),
## #> ..   biz = col_double(),
## #> ..   cleanliness_rating = col_double(),
## #> ..   guest_satisfaction_overall = col_double(),
## #> ..   bedrooms = col_double(),
## #> ..   dist = col_double(),
## #> ..   metro_dist = col_double(),
## #> ..   attr_index = col_double(),
## #> ..   attr_index_norm = col_double(),
## #> ..   rest_index = col_double(),
## #> ..   rest_index_norm = col_double(),
## #> ..   lng = col_double(),
## #> ..   lat = col_double()

```

```

## .. )
## - attr(*, "problems")=<externalptr>

budapest_weekdays <- read_csv("~/R STUDIO/case_studies/airbnb/budapest_weekdays.csv")

## New names:
## Rows: 2074 Columns: 20
## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * `--> '...1'

colnames(budapest_weekdays)

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"               "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"

str(budapest_weekdays)

## #> spc_tbl_ [2,074 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## #> $ ...1                      : num [1:2074] 0 1 2 3 4 5 6 7 8 9 ...
## #> $ realSum                    : num [1:2074] 239 301 162 118 134 ...
## #> $ room_type                  : chr [1:2074] "Entire home/apt" "Entire home/apt" "Entire home/apt" ...
## #> $ room_shared                 : logi [1:2074] FALSE FALSE FALSE FALSE FALSE FALSE ...
## #> $ room_private                : logi [1:2074] FALSE FALSE FALSE FALSE FALSE FALSE ...
## #> $ person_capacity             : num [1:2074] 6 6 4 2 4 4 4 4 6 4 ...
## #> $ host_is_superhost           : logi [1:2074] TRUE FALSE TRUE FALSE TRUE FALSE ...
## #> $ multi                       : num [1:2074] 0 0 0 0 1 1 0 0 0 0 ...
## #> $ biz                          : num [1:2074] 1 1 0 0 0 0 0 1 1 0 ...
## #> $ cleanliness_rating           : num [1:2074] 10 9 10 9 10 9 6 9 10 10 ...
## #> $ guest_satisfaction_overall: num [1:2074] 99 98 98 92 99 91 80 92 88 96 ...
## #> $ bedrooms                     : num [1:2074] 1 2 1 1 2 2 2 1 2 1 ...
## #> $ dist                         : num [1:2074] 0.359 0.929 2.451 1.559 1.114 ...
## #> $ metro_dist                  : num [1:2074] 0.353 0.2 0.279 0.478 0.27 ...
## #> $ attr_index                  : num [1:2074] 404 1677 164 192 199 ...
## #> $ attr_index_norm              : num [1:2074] 24.12 100 9.76 11.43 11.84 ...
## #> $ rest_index                  : num [1:2074] 893 453 192 326 636 ...
## #> $ rest_index_norm              : num [1:2074] 67.7 34.3 14.5 24.7 48.1 ...
## #> $ lng                          : num [1:2074] 19.1 19 19 19.1 19.1 ...
## #> $ lat                          : num [1:2074] 47.5 47.5 47.5 47.5 47.5 ...
## - attr(*, "spec")=
```

```

## .. cols(
## ..   ...1 = col_double(),
## ..   realSum = col_double(),
## ..   room_type = col_character(),
## ..   room_shared = col_logical(),
## ..   room_private = col_logical(),
## ..   person_capacity = col_double(),
## ..   host_is_superhost = col_logical(),
## ..   multi = col_double(),
## ..   biz = col_double(),
## ..   cleanliness_rating = col_double(),
## ..   guest_satisfaction_overall = col_double(),
## ..   bedrooms = col_double(),
## ..   dist = col_double(),
## ..   metro_dist = col_double(),
## ..   attr_index = col_double(),
## ..   attr_index_norm = col_double(),
## ..   rest_index = col_double(),
## ..   rest_index_norm = col_double(),
## ..   lng = col_double(),
## ..   lat = col_double()
## .. )
## - attr(*, "problems")=<externalptr>

```

```
budapest_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/budapest_weekends.csv")
```

```

## New names:
## Rows: 1948 Columns: 20
## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(budapest_weekends)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"           "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                  "attr_index_norm"
## [17] "rest_index"                  "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(budapest_weekends)
```

```
## spc_tbl_ [1,948 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
```

```

## $ ...1 : num [1:1948] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum : num [1:1948] 332 331 244 207 238 ...
## $ room_type : chr [1:1948] "Entire home/apt" "Entire home/apt" "Entire home/apt" "Entire ...
## $ room_shared : logi [1:1948] FALSE FALSE FALSE FALSE FALSE ...
## $ room_private : logi [1:1948] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ person_capacity : num [1:1948] 6 6 6 5 4 2 3 4 5 2 ...
## $ host_is_superhost : logi [1:1948] TRUE FALSE FALSE TRUE FALSE FALSE ...
## $ multi : num [1:1948] 0 0 1 1 1 1 1 0 0 1 ...
## $ biz : num [1:1948] 1 1 0 0 0 0 0 1 1 0 ...
## $ cleanliness_rating : num [1:1948] 10 9 10 10 9 10 9 9 9 10 ...
## $ guest_satisfaction_overall: num [1:1948] 99 98 95 99 84 100 93 92 91 95 ...
## $ bedrooms : num [1:1948] 1 2 2 2 0 0 1 0 1 0 ...
## $ dist : num [1:1948] 0.359 0.929 0.674 1.338 0.836 ...
## $ metro_dist : num [1:1948] 0.353 0.2 0.315 0.614 0.176 ...
## $ attr_index : num [1:1948] 404 1675 430 483 536 ...
## $ attr_index_norm : num [1:1948] 24.1 100 25.7 28.8 32 ...
## $ rest_index : num [1:1948] 893 453 699 372 534 ...
## $ rest_index_norm : num [1:1948] 78.1 39.6 61.1 32.5 46.7 ...
## $ lng : num [1:1948] 19.1 19 19 19 19 ...
## $ lat : num [1:1948] 47.5 47.5 47.5 47.5 47.5 ...
## - attr(*, "spec")=
##   .. cols(
##     .. ...1 = col_double(),
##     .. realSum = col_double(),
##     .. room_type = col_character(),
##     .. room_shared = col_logical(),
##     .. room_private = col_logical(),
##     .. person_capacity = col_double(),
##     .. host_is_superhost = col_logical(),
##     .. multi = col_double(),
##     .. biz = col_double(),
##     .. cleanliness_rating = col_double(),
##     .. guest_satisfaction_overall = col_double(),
##     .. bedrooms = col_double(),
##     .. dist = col_double(),
##     .. metro_dist = col_double(),
##     .. attr_index = col_double(),
##     .. attr_index_norm = col_double(),
##     .. rest_index = col_double(),
##     .. rest_index_norm = col_double(),
##     .. lng = col_double(),
##     .. lat = col_double()
##   .. )
## - attr(*, "problems")=<externalptr>

```

```
lisbon_weekdays <- read_csv("~/R STUDIO/case_studies/airbnb/lisbon_weekdays.csv")
```

```

## New names:
## Rows: 2857 Columns: 20
## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i

```

```
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## * `` -> '...1'
```

```
colnames(lisbon_weekdays)
```

```
## [1] "...1"                      "realSum"  
## [3] "room_type"                  "room_shared"  
## [5] "room_private"                "person_capacity"  
## [7] "host_is_superhost"          "multi"  
## [9] "biz"                         "cleanliness_rating"  
## [11] "guest_satisfaction_overall" "bedrooms"  
## [13] "dist"                        "metro_dist"  
## [15] "attr_index"                 "attr_index_norm"  
## [17] "rest_index"                 "rest_index_norm"  
## [19] "lng"                         "lat"
```

```
str(lisbon_weekdays)
```

```
## #> #> spc_tbl_ [2,857 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)  
## #> #> $ ...1                      : num [1:2857] 0 1 2 3 4 5 6 7 8 9 ...  
## #> #> $ realSum                  : num [1:2857] 138 124 194 192 328 ...  
## #> #> $ room_type                 : chr [1:2857] "Private room" "Private room" "Private room" "Entire home..."  
## #> #> $ room_shared                : logi [1:2857] FALSE FALSE FALSE FALSE FALSE ...  
## #> #> $ room_private               : logi [1:2857] TRUE TRUE TRUE FALSE FALSE TRUE ...  
## #> #> $ person_capacity            : num [1:2857] 2 2 4 4 6 2 2 2 3 4 ...  
## #> #> $ host_is_superhost         : logi [1:2857] TRUE TRUE TRUE FALSE FALSE FALSE ...  
## #> #> $ multi                     : num [1:2857] 1 1 1 0 1 0 0 0 0 1 ...  
## #> #> $ biz                        : num [1:2857] 0 0 0 1 0 1 1 1 1 0 ...  
## #> #> $ cleanliness_rating         : num [1:2857] 10 10 10 9 9 6 9 9 10 10 ...  
## #> #> $ guest_satisfaction_overall: num [1:2857] 98 97 87 89 93 40 88 80 95 95 ...  
## #> #> $ bedrooms                   : num [1:2857] 1 1 2 1 2 1 1 1 1 1 ...  
## #> #> $ dist                        : num [1:2857] 4.328 4.466 4.475 0.851 0.667 ...  
## #> #> $ metro_dist                 : num [1:2857] 0.298 0.294 0.168 0.589 0.53 ...  
## #> #> $ attr_index                 : num [1:2857] 74.2 72.6 72.5 409.7 537 ...  
## #> #> $ attr_index_norm             : num [1:2857] 2.45 2.4 2.39 13.53 17.73 ...  
## #> #> $ rest_index                 : num [1:2857] 177 174 178 806 776 ...  
## #> #> $ rest_index_norm             : num [1:2857] 7.91 7.78 7.96 36.07 34.69 ...  
## #> #> $ lng                         : num [1:2857] -9.14 -9.14 -9.14 -9.13 -9.13 ...  
## #> #> $ lat                         : num [1:2857] 38.8 38.8 38.8 38.7 38.7 ...  
## #> - attr(*, "spec")=  
## #> .. cols(  
## #> ..   ...1 = col_double(),  
## #> ..   realSum = col_double(),  
## #> ..   room_type = col_character(),  
## #> ..   room_shared = col_logical(),  
## #> ..   room_private = col_logical(),  
## #> ..   person_capacity = col_double(),  
## #> ..   host_is_superhost = col_logical(),  
## #> ..   multi = col_double(),  
## #> ..   biz = col_double(),  
## #> ..   cleanliness_rating = col_double(),  
## #> ..   guest_satisfaction_overall = col_double(),  
## #> ..   bedrooms = col_double(),
```

```

## .. dist = col_double(),
## .. metro_dist = col_double(),
## .. attr_index = col_double(),
## .. attr_index_norm = col_double(),
## .. rest_index = col_double(),
## .. rest_index_norm = col_double(),
## .. lng = col_double(),
## .. lat = col_double()
## ...
## - attr(*, "problems")=<externalptr>

lisbon_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/lisbon_weekends.csv")

## New names:
## Rows: 2906 Columns: 20
## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

colnames(lisbon_weekends)

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"           "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                  "attr_index_norm"
## [17] "rest_index"                  "rest_index_norm"
## [19] "lng"                         "lat"

str(lisbon_weekends)

## #> spc_tbl_ [2,906 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## #> $ ...1                      : num [1:2906] 0 1 2 3 4 5 6 7 8 9 ...
## #> $ realSum                     : num [1:2906] 138 124 193 326 174 ...
## #> $ room_type                   : chr [1:2906] "Private room" "Private room" "Private room" "Entire home...
## #> $ room_shared                 : logi [1:2906] FALSE FALSE FALSE FALSE FALSE ...
## #> $ room_private                : logi [1:2906] TRUE TRUE TRUE FALSE TRUE FALSE ...
## #> $ person_capacity              : num [1:2906] 2 2 4 6 3 4 4 3 4 2 ...
## #> $ host_is_superhost           : logi [1:2906] TRUE TRUE TRUE FALSE FALSE TRUE ...
## #> $ multi                        : num [1:2906] 1 1 1 1 0 1 1 0 0 1 ...
## #> $ biz                           : num [1:2906] 0 0 0 0 1 0 0 1 1 0 ...
## #> $ cleanliness_rating            : num [1:2906] 10 10 10 9 10 10 10 10 10 10 ...
## #> $ guest_satisfaction_overall: num [1:2906] 98 97 87 93 96 95 96 91 96 99 ...
## #> $ bedrooms                      : num [1:2906] 1 1 2 2 1 1 1 2 1 1 ...
## #> $ dist                          : num [1:2906] 4.328 4.465 4.475 0.667 4.889 ...

```

```

## $ metro_dist : num [1:2906] 0.298 0.294 0.168 0.53 0.466 ...
## $ attr_index : num [1:2906] 74.2 72.6 72.5 537 68 ...
## $ attr_index_norm : num [1:2906] 2.45 2.39 2.39 17.71 2.24 ...
## $ rest_index : num [1:2906] 177 174 178 776 160 ...
## $ rest_index_norm : num [1:2906] 9.94 9.77 10 43.6 9 ...
## $ lng : num [1:2906] -9.14 -9.14 -9.14 -9.13 -9.14 ...
## $ lat : num [1:2906] 38.8 38.8 38.8 38.7 38.8 ...
## - attr(*, "spec")=
##   .. cols(
##     ... .1 = col_double(),
##     ... realSum = col_double(),
##     ... room_type = col_character(),
##     ... room_shared = col_logical(),
##     ... room_private = col_logical(),
##     ... person_capacity = col_double(),
##     ... host_is_superhost = col_logical(),
##     ... multi = col_double(),
##     ... biz = col_double(),
##     ... cleanliness_rating = col_double(),
##     ... guest_satisfaction_overall = col_double(),
##     ... bedrooms = col_double(),
##     ... dist = col_double(),
##     ... metro_dist = col_double(),
##     ... attr_index = col_double(),
##     ... attr_index_norm = col_double(),
##     ... rest_index = col_double(),
##     ... rest_index_norm = col_double(),
##     ... lng = col_double(),
##     ... lat = col_double()
##   ... )
## - attr(*, "problems")=<externalptr>

```

```
london_weekdays <- read_csv("~/R_STUDIO/case_studies/airbnb/london_weekdays.csv")
```

```

## New names:
## Rows: 4614 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(london_weekdays)
```

```

## [1] "...1"                   "realSum"
## [3] "room_type"              "room_shared"
## [5] "room_private"            "person_capacity"
## [7] "host_is_superhost"       "multi"
## [9] "biz"                     "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                    "metro_dist"

```

```

## [15] "attr_index"                      "attr_index_norm"
## [17] "rest_index"                       "rest_index_norm"
## [19] "lng"                             "lat"

str(london_weekdays)

## #> #> spc_tbl_ [4,614 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## #> #>   $ ...1                         : num [1:4614] 0 1 2 3 4 5 6 7 8 9 ...
## #> #>   $ realSum                      : num [1:4614] 570 298 337 227 256 ...
## #> #>   $ room_type                     : chr [1:4614] "Entire home/apt" "Private room" "Private ...
## #> #>   $ room_shared                   : logi [1:4614] FALSE FALSE FALSE FALSE FALSE ...
## #> #>   $ room_private                  : logi [1:4614] FALSE TRUE TRUE TRUE TRUE TRUE ...
## #> #>   $ person_capacity               : num [1:4614] 2 2 2 2 3 2 2 2 4 3 ...
## #> #>   $ host_is_superhost            : logi [1:4614] FALSE TRUE FALSE TRUE FALSE FALSE ...
## #> #>   $ multi                         : num [1:4614] 0 1 1 1 0 0 0 1 1 0 ...
## #> #>   $ biz                            : num [1:4614] 0 0 0 0 0 0 0 0 0 1 ...
## #> #>   $ cleanliness_rating             : num [1:4614] 10 10 10 10 9 10 10 10 9 8 ...
## #> #>   $ guest_satisfaction_overall: num [1:4614] 98 99 96 99 98 100 100 95 93 74 ...
## #> #>   $ bedrooms                      : num [1:4614] 1 1 1 1 1 1 1 1 1 0 ...
## #> #>   $ dist                           : num [1:4614] 5.3 2.2 2.32 5.71 3.26 ...
## #> #>   $ metro_dist                    : num [1:4614] 1.589 0.379 0.453 1.724 0.825 ...
## #> #>   $ attr_index                    : num [1:4614] 210 554 428 196 329 ...
## #> #>   $ attr_index_norm              : num [1:4614] 14.6 38.5 29.8 13.6 22.9 ...
## #> #>   $ rest_index                    : num [1:4614] 468 961 959 452 735 ...
## #> #>   $ rest_index_norm              : num [1:4614] 8.37 17.22 17.17 8.1 13.17 ...
## #> #>   $ lng                            : num [1:4614] -0.1603 -0.0968 -0.1055 -0.1658 -0.1206 ...
## #> #>   $ lat                            : num [1:4614] 51.5 51.5 51.5 51.5 51.5 ...
## #> #> - attr(*, "spec")=
## #> #> .. cols(
## #> #>   .. ...1 = col_double(),
## #> #>   .. realSum = col_double(),
## #> #>   .. room_type = col_character(),
## #> #>   .. room_shared = col_logical(),
## #> #>   .. room_private = col_logical(),
## #> #>   .. person_capacity = col_double(),
## #> #>   .. host_is_superhost = col_logical(),
## #> #>   .. multi = col_double(),
## #> #>   .. biz = col_double(),
## #> #>   .. cleanliness_rating = col_double(),
## #> #>   .. guest_satisfaction_overall = col_double(),
## #> #>   .. bedrooms = col_double(),
## #> #>   .. dist = col_double(),
## #> #>   .. metro_dist = col_double(),
## #> #>   .. attr_index = col_double(),
## #> #>   .. attr_index_norm = col_double(),
## #> #>   .. rest_index = col_double(),
## #> #>   .. rest_index_norm = col_double(),
## #> #>   .. lng = col_double(),
## #> #>   .. lat = col_double()
## #> #> )
## #> #> - attr(*, "problems")=<externalptr>

```

```
london_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/london_weekends.csv")
```

```
## New names:  
## Rows: 5379 Columns: 20  
## -- Column specification  
## ----- Delimiter: "," chr  
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,  
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost  
## i Use 'spec()' to retrieve the full column specification for this data. i  
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.  
## * ' ' -> '...1'
```

```
colnames(london_weekends)
```

```
## [1] "...1"                      "realSum"  
## [3] "room_type"                  "room_shared"  
## [5] "room_private"                "person_capacity"  
## [7] "host_is_superhost"          "multi"  
## [9] "biz"                         "cleanliness_rating"  
## [11] "guest_satisfaction_overall" "bedrooms"  
## [13] "dist"                        "metro_dist"  
## [15] "attr_index"                 "attr_index_norm"  
## [17] "rest_index"                 "rest_index_norm"  
## [19] "lng"                         "lat"
```

```
str(london_weekends)
```

```
## #> #> spc_tbl_ [5,379 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)  
## #> #> $ ...1                      : num [1:5379] 0 1 2 3 4 5 6 7 8 9 ...  
## #> #> $ realSum                   : num [1:5379] 121 196 193 180 406 ...  
## #> #> $ room_type                  : chr [1:5379] "Private room" "Private room" "Private room" "Private room" ...  
## #> #> $ room_shared                : logi [1:5379] FALSE FALSE FALSE FALSE FALSE ...  
## #> #> $ room_private               : logi [1:5379] TRUE TRUE TRUE TRUE FALSE FALSE ...  
## #> #> $ person_capacity            : num [1:5379] 2 2 3 2 3 2 2 2 4 2 ...  
## #> #> $ host_is_superhost         : logi [1:5379] FALSE FALSE FALSE FALSE FALSE FALSE ...  
## #> #> $ multi                     : num [1:5379] 0 1 1 1 0 0 0 0 0 1 ...  
## #> #> $ biz                        : num [1:5379] 0 0 0 0 1 1 1 1 1 0 ...  
## #> #> $ cleanliness_rating        : num [1:5379] 6 10 10 9 7 9 10 9 9 10 ...  
## #> #> $ guest_satisfaction_overall: num [1:5379] 69 96 95 87 65 93 97 88 87 97 ...  
## #> #> $ bedrooms                   : num [1:5379] 1 1 1 1 0 0 1 1 1 1 ...  
## #> #> $ dist                        : num [1:5379] 5.73 4.79 4.6 2.05 4.49 ...  
## #> #> $ metro_dist                 : num [1:5379] 0.437 1.464 0.45 0.133 0.354 ...  
## #> #> $ attr_index                 : num [1:5379] 223 235 269 472 318 ...  
## #> #> $ attr_index_norm            : num [1:5379] 15.5 16.4 18.7 32.8 22.1 ...  
## #> #> $ rest_index                 : num [1:5379] 470 530 549 1021 693 ...  
## #> #> $ rest_index_norm            : num [1:5379] 8.41 9.49 9.83 18.28 12.4 ...  
## #> #> $ lng                         : num [1:5379] -0.0498 -0.0848 -0.1459 -0.1061 -0.188 ...  
## #> #> $ lat                         : num [1:5379] 51.5 51.5 51.5 51.5 51.5 ...  
## #> - attr(*, "spec")=  
## #> .. cols(  
## #> ..   ...1 = col_double(),  
## #> ..   realSum = col_double(),
```

```

## .. room_type = col_character(),
## .. room_shared = col_logical(),
## .. room_private = col_logical(),
## .. person_capacity = col_double(),
## .. host_is_superhost = col_logical(),
## .. multi = col_double(),
## .. biz = col_double(),
## .. cleanliness_rating = col_double(),
## .. guest_satisfaction_overall = col_double(),
## .. bedrooms = col_double(),
## .. dist = col_double(),
## .. metro_dist = col_double(),
## .. attr_index = col_double(),
## .. attr_index_norm = col_double(),
## .. rest_index = col_double(),
## .. rest_index_norm = col_double(),
## .. lng = col_double(),
## .. lat = col_double()
## ...
## - attr(*, "problems")=<externalptr>

```

```
paris_weekdays <- read_csv("~/R_STUDIO/case_studies/airbnb/paris_weekdays.csv")
```

```

## New names:
## Rows: 3130 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(paris_weekdays)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"           "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                  "attr_index_norm"
## [17] "rest_index"                  "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(paris_weekdays)
```

```

## spc_tbl_ [3,130 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1                      : num [1:3130] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum                    : num [1:3130] 296 288 211 299 248 ...
## $ room_type                  : chr [1:3130] "Private room" "Private room" "Private room" "Entire hom...

```

```

## $ room_shared : logi [1:3130] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private : logi [1:3130] TRUE TRUE TRUE FALSE FALSE FALSE ...
## $ person_capacity : num [1:3130] 2 2 2 2 4 4 2 2 2 3 ...
## $ host_is_superhost : logi [1:3130] TRUE TRUE FALSE FALSE FALSE TRUE ...
## $ multi : num [1:3130] 0 0 0 0 0 0 0 0 0 ...
## $ biz : num [1:3130] 0 0 0 1 0 0 0 0 1 1 ...
## $ cleanliness_rating : num [1:3130] 10 10 10 9 7 10 10 10 9 10 ...
## $ guest_satisfaction_overall: num [1:3130] 97 97 94 91 82 93 90 95 94 91 ...
## $ bedrooms : num [1:3130] 1 1 1 1 1 1 1 1 0 1 ...
## $ dist : num [1:3130] 0.7 2.1 3.302 0.548 1.198 ...
## $ metro_dist : num [1:3130] 0.194 0.107 0.235 0.196 0.104 ...
## $ attr_index : num [1:3130] 518 873 445 542 407 ...
## $ attr_index_norm : num [1:3130] 25.2 42.5 21.6 26.4 19.8 ...
## $ rest_index : num [1:3130] 1219 1001 903 1199 1071 ...
## $ rest_index_norm : num [1:3130] 71.6 58.8 53.1 70.5 62.9 ...
## $ lng : num [1:3130] 2.35 2.32 2.32 2.36 2.36 ...
## $ lat : num [1:3130] 48.9 48.9 48.9 48.9 48.9 ...
## - attr(*, "spec")=
## .. cols(
## ..   ...1 = col_double(),
## ..   realSum = col_double(),
## ..   room_type = col_character(),
## ..   room_shared = col_logical(),
## ..   room_private = col_logical(),
## ..   person_capacity = col_double(),
## ..   host_is_superhost = col_logical(),
## ..   multi = col_double(),
## ..   biz = col_double(),
## ..   cleanliness_rating = col_double(),
## ..   guest_satisfaction_overall = col_double(),
## ..   bedrooms = col_double(),
## ..   dist = col_double(),
## ..   metro_dist = col_double(),
## ..   attr_index = col_double(),
## ..   attr_index_norm = col_double(),
## ..   rest_index = col_double(),
## ..   rest_index_norm = col_double(),
## ..   lng = col_double(),
## ..   lat = col_double()
## .. )
## - attr(*, "problems")=<externalptr>

```

```
paris_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/paris_weekends.csv")
```

```

## New names:
## Rows: 3558 Columns: 20
## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * `` -> ...1'

```

```
colnames(paris_weekends)
```

```
## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"           "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"
```

```
str(paris_weekends)
```

```
## spc_tbl_ [3,558 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1
## $ realSum
## $ room_type
## $ room_shared
## $ room_private
## $ person_capacity
## $ host_is_superhost
## $ multi
## $ biz
## $ cleanliness_rating
## $ guest_satisfaction_overall
## $ bedrooms
## $ dist
## $ metro_dist
## $ attr_index
## $ attr_index_norm
## $ rest_index
## $ rest_index_norm
## $ lng
## $ lat
## - attr(*, "spec")=
##   .. cols(
##     ..  ...1 = col_double(),
##     ..  realSum = col_double(),
##     ..  room_type = col_character(),
##     ..  room_shared = col_logical(),
##     ..  room_private = col_logical(),
##     ..  person_capacity = col_double(),
##     ..  host_is_superhost = col_logical(),
##     ..  multi = col_double(),
##     ..  biz = col_double(),
##     ..  cleanliness_rating = col_double(),
##     ..  guest_satisfaction_overall = col_double(),
##     ..  bedrooms = col_double(),
##     ..  dist = col_double(),
##     ..  metro_dist = col_double(),
```

```

## .. attr_index = col_double(),
## .. attr_index_norm = col_double(),
## .. rest_index = col_double(),
## .. rest_index_norm = col_double(),
## .. lng = col_double(),
## .. lat = col_double()
## ...
## - attr(*, "problems")=<externalptr>

```

```
rome_weekdays <- read_csv("~/R_STUDIO/case_studies/airbnb/rome_weekdays.csv")
```

```

## New names:
## Rows: 4492 Columns: 20
## -- Column specification
## ----- Delimiter: "," chr
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(rome_weekdays)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"               "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(rome_weekdays)
```

```

## spc_tbl_ [4,492 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1                      : num [1:4492] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum                    : num [1:4492] 157 173 278 445 131 ...
## $ room_type                  : chr [1:4492] "Private room" "Private room" "Entire home/apt" "Entire ...
## $ room_shared                 : logi [1:4492] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private                : logi [1:4492] TRUE TRUE FALSE FALSE TRUE FALSE ...
## $ person_capacity              : num [1:4492] 2 2 4 6 3 4 2 3 2 2 ...
## $ host_is_superhost           : logi [1:4492] TRUE FALSE FALSE FALSE FALSE FALSE ...
## $ multi                       : num [1:4492] 1 1 0 1 1 1 0 0 0 0 ...
## $ biz                          : num [1:4492] 0 0 1 0 0 0 1 0 0 1 ...
## $ cleanliness_rating           : num [1:4492] 10 9 9 9 9 9 9 9 9 10 ...
## $ guest_satisfaction_overall: num [1:4492] 95 80 90 92 91 89 90 92 94 96 ...
## $ bedrooms                     : num [1:4492] 1 1 1 2 1 2 1 1 1 0 ...
## $ dist                         : num [1:4492] 2.978 0.935 2.203 2.703 1.296 ...
## $ metro_dist                   : num [1:4492] 1.596 0.649 0.495 1.295 0.867 ...
## $ attr_index                   : num [1:4492] 281 483 692 806 317 ...

```

```

## $ attr_index_norm      : num [1:4492] 6.23 10.7 15.33 17.85 7.03 ...
## $ rest_index           : num [1:4492] 698 1252 1626 2036 837 ...
## $ rest_index_norm      : num [1:4492] 15.2 27.2 35.4 44.3 18.2 ...
## $ lng                  : num [1:4492] 12.5 12.5 12.5 12.5 12.5 ...
## $ lat                  : num [1:4492] 41.9 41.9 41.9 41.9 41.9 ...
## - attr(*, "spec")=
##   .. cols(
##     .. .1 = col_double(),
##     .. realSum = col_double(),
##     .. room_type = col_character(),
##     .. room_shared = col_logical(),
##     .. room_private = col_logical(),
##     .. person_capacity = col_double(),
##     .. host_is_superhost = col_logical(),
##     .. multi = col_double(),
##     .. biz = col_double(),
##     .. cleanliness_rating = col_double(),
##     .. guest_satisfaction_overall = col_double(),
##     .. bedrooms = col_double(),
##     .. dist = col_double(),
##     .. metro_dist = col_double(),
##     .. attr_index = col_double(),
##     .. attr_index_norm = col_double(),
##     .. rest_index = col_double(),
##     .. rest_index_norm = col_double(),
##     .. lng = col_double(),
##     .. lat = col_double()
##   .. )
## - attr(*, "problems")=<externalptr>

```

```
rome_weekends <- read_csv("~/R_STUDIO/case_studies/airbnb/rome_weekends.csv")
```

```

## New names:
## Rows: 4535 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(rome_weekends)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                  "attr_index_norm"
## [17] "rest_index"                  "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(rome_weekends)
```

```
## spc_tbl_ [4,535 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1 : num [1:4535] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum : num [1:4535] 173 157 196 299 131 ...
## $ room_type : chr [1:4535] "Private room" "Private room" "Private room" "Entire home ...
## $ room_shared : logi [1:4535] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private : logi [1:4535] TRUE TRUE TRUE FALSE TRUE FALSE ...
## $ person_capacity : num [1:4535] 2 2 2 4 3 6 4 2 4 3 ...
## $ host_is_superhost : logi [1:4535] FALSE TRUE FALSE FALSE FALSE FALSE FALSE ...
## $ multi : num [1:4535] 0 1 1 0 1 1 1 0 0 0 ...
## $ biz : num [1:4535] 0 0 0 1 0 0 0 1 1 0 ...
## $ cleanliness_rating : num [1:4535] 10 10 9 9 9 9 9 9 10 9 ...
## $ guest_satisfaction_overall: num [1:4535] 93 95 80 90 91 90 89 90 94 92 ...
## $ bedrooms : num [1:4535] 1 1 1 1 1 2 2 1 1 1 ...
## $ dist : num [1:4535] 1.223 2.978 0.935 2.203 1.296 ...
## $ metro_dist : num [1:4535] 0.398 1.596 0.649 0.495 0.867 ...
## $ attr_index : num [1:4535] 550 281 483 692 317 ...
## $ attr_index_norm : num [1:4535] 12.19 6.23 10.69 15.33 7.02 ...
## $ rest_index : num [1:4535] 1075 698 1252 1626 837 ...
## $ rest_index_norm : num [1:4535] 23.4 15.2 27.3 35.4 18.2 ...
## $ lng : num [1:4535] 12.5 12.5 12.5 12.5 12.5 ...
## $ lat : num [1:4535] 41.9 41.9 41.9 41.9 41.9 ...
## - attr(*, "spec")=
## .. cols(
## ..   ...1 = col_double(),
## ..   realSum = col_double(),
## ..   room_type = col_character(),
## ..   room_shared = col_logical(),
## ..   room_private = col_logical(),
## ..   person_capacity = col_double(),
## ..   host_is_superhost = col_logical(),
## ..   multi = col_double(),
## ..   biz = col_double(),
## ..   cleanliness_rating = col_double(),
## ..   guest_satisfaction_overall = col_double(),
## ..   bedrooms = col_double(),
## ..   dist = col_double(),
## ..   metro_dist = col_double(),
## ..   attr_index = col_double(),
## ..   attr_index_norm = col_double(),
## ..   rest_index = col_double(),
## ..   rest_index_norm = col_double(),
## ..   lng = col_double(),
## ..   lat = col_double()
## .. )
## - attr(*, "problems")=<externalptr>
```

```
vienna_weekdays <- read_csv("~/R STUDIO/case_studies/airbnb/vienna_weekdays.csv")
```

```
## New names:
## Rows: 1738 Columns: 20
```

```

## -- Column specification
## ----- Delimiter: ","
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' -> '...1'

```

```
colnames(vienna_weekdays)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"                "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(vienna_weekdays)
```

```

## spc_tbl_ [1,738 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1
##   : num [1:1738] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum
##   : num [1:1738] 251 157 283 302 151 ...
## $ room_type
##   : chr [1:1738] "Entire home/apt" "Entire home/apt" "Entire home/apt" "Entire home/apt" ...
## $ room_shared
##   : logi [1:1738] FALSE FALSE FALSE FALSE FALSE FALSE ...
## $ room_private
##   : logi [1:1738] FALSE FALSE FALSE FALSE FALSE TRUE ...
## $ person_capacity
##   : num [1:1738] 3 3 5 4 2 2 3 2 4 2 ...
## $ host_is_superhost
##   : logi [1:1738] TRUE FALSE TRUE TRUE TRUE TRUE ...
## $ multi
##   : num [1:1738] 1 0 0 0 1 0 0 0 1 ...
## $ biz
##   : num [1:1738] 0 0 1 1 1 0 1 0 1 0 ...
## $ cleanliness_rating
##   : num [1:1738] 10 10 10 10 10 10 9 8 9 10 ...
## $ guest_satisfaction_overall
##   : num [1:1738] 99 97 95 94 96 99 91 92 86 97 ...
## $ bedrooms
##   : num [1:1738] 1 1 2 2 1 1 1 1 1 1 ...
## $ dist
##   : num [1:1738] 1.73 4.81 2.81 3.24 2.24 ...
## $ metro_dist
##   : num [1:1738] 0.213 0.228 0.244 0.61 1.12 ...
## $ attr_index
##   : num [1:1738] 118.6 80.1 101.2 92.5 124.3 ...
## $ attr_index_norm
##   : num [1:1738] 8.5 5.74 7.25 6.62 8.9 ...
## $ rest_index
##   : num [1:1738] 185.1 85.7 151.4 129.9 154.3 ...
## $ rest_index_norm
##   : num [1:1738] 4.42 2.05 3.62 3.1 3.69 ...
## $ lng
##   : num [1:1738] 16.4 16.3 16.3 16.3 16.4 ...
## $ lat
##   : num [1:1738] 48.2 48.2 48.2 48.2 48.2 ...
## - attr(*, "spec")=
## .. cols(
##   .. ...1 = col_double(),
##   .. realSum = col_double(),
##   .. room_type = col_character(),
##   .. room_shared = col_logical(),
##   .. room_private = col_logical(),
##   .. person_capacity = col_double(),
##   .. host_is_superhost = col_logical(),

```

```

## .. multi = col_double(),
## .. biz = col_double(),
## .. cleanliness_rating = col_double(),
## .. guest_satisfaction_overall = col_double(),
## .. bedrooms = col_double(),
## .. dist = col_double(),
## .. metro_dist = col_double(),
## .. attr_index = col_double(),
## .. attr_index_norm = col_double(),
## .. rest_index = col_double(),
## .. rest_index_norm = col_double(),
## .. lng = col_double(),
## .. lat = col_double()
## ...
## - attr(*, "problems")=<externalptr>

```

```
vienna_weekends <- read_csv("~/R STUDIO/case_studies/airbnb/vienna_weekends.csv")
```

```

## New names:
## Rows: 1799 Columns: 20
## -- Column specification
## -----
## (1): room_type dbl (16): ...1, realSum, person_capacity, multi, biz,
## cleanliness_rating, gu... lgl (3): room_shared, room_private, host_is_superhost
## i Use 'spec()' to retrieve the full column specification for this data. i
## Specify the column types or set 'show_col_types = FALSE' to quiet this message.
## * ' ' -> '...1'

```

```
colnames(vienna_weekends)
```

```

## [1] "...1"                      "realSum"
## [3] "room_type"                  "room_shared"
## [5] "room_private"               "person_capacity"
## [7] "host_is_superhost"          "multi"
## [9] "biz"                         "cleanliness_rating"
## [11] "guest_satisfaction_overall" "bedrooms"
## [13] "dist"                        "metro_dist"
## [15] "attr_index"                 "attr_index_norm"
## [17] "rest_index"                 "rest_index_norm"
## [19] "lng"                         "lat"

```

```
str(vienna_weekends)
```

```

## spc_tbl_ [1,799 x 20] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ ...1                      : num [1:1799] 0 1 2 3 4 5 6 7 8 9 ...
## $ realSum                    : num [1:1799] 325 151 196 108 196 ...
## $ room_type                  : chr [1:1799] "Entire home/apt" "Entire home/apt" "Entire home/apt" ...
## $ room_shared                 : logi [1:1799] FALSE FALSE FALSE FALSE FALSE ...
## $ room_private                : logi [1:1799] FALSE FALSE FALSE TRUE FALSE FALSE ...
## $ person_capacity              : num [1:1799] 4 2 3 2 4 2 2 2 4 ...
## $ host_is_superhost           : logi [1:1799] TRUE TRUE TRUE FALSE TRUE TRUE ...
## $ multi                       : num [1:1799] 0 0 0 0 1 0 0 0 0 ...

```

```

## $ biz : num [1:1799] 1 1 1 0 1 0 0 1 1 1 ...
## $ cleanliness_rating : num [1:1799] 10 10 9 8 9 10 10 9 10 10 ...
## $ guest_satisfaction_overall: num [1:1799] 94 96 91 92 94 97 94 90 100 100 ...
## $ bedrooms : num [1:1799] 2 1 1 1 1 1 1 1 1 1 ...
## $ dist : num [1:1799] 3.24 2.24 2.37 3.63 2.4 ...
## $ metro_dist : num [1:1799] 0.61 1.12 0.452 0.473 0.444 ...
## $ attr_index : num [1:1799] 92.5 124.3 125.4 62.5 124 ...
## $ attr_index_norm : num [1:1799] 6.62 8.9 8.98 4.47 8.88 ...
## $ rest_index : num [1:1799] 129.9 154.4 206.6 93.8 202.3 ...
## $ rest_index_norm : num [1:1799] 3.14 3.73 4.99 2.27 4.89 ...
## $ lng : num [1:1799] 16.3 16.4 16.3 16.4 16.3 ...
## $ lat : num [1:1799] 48.2 48.2 48.2 48.2 48.2 ...
## - attr(*, "spec")=
##   .. cols(
##     .. .1 = col_double(),
##     .. realSum = col_double(),
##     .. room_type = col_character(),
##     .. room_shared = col_logical(),
##     .. room_private = col_logical(),
##     .. person_capacity = col_double(),
##     .. host_is_superhost = col_logical(),
##     .. multi = col_double(),
##     .. biz = col_double(),
##     .. cleanliness_rating = col_double(),
##     .. guest_satisfaction_overall = col_double(),
##     .. bedrooms = col_double(),
##     .. dist = col_double(),
##     .. metro_dist = col_double(),
##     .. attr_index = col_double(),
##     .. attr_index_norm = col_double(),
##     .. rest_index = col_double(),
##     .. rest_index_norm = col_double(),
##     .. lng = col_double(),
##     .. lat = col_double()
##   .. )
## - attr(*, "problems")=<externalptr>

```

To simplify analysis, let's consolidate the data. Initially, I'll rename the initial column as 'City' and populate it with the corresponding city for each dataset. Subsequently, I'll introduce a new column named 'Weekdays/Weekends' and populate it with "Weekdays" for listings available during weekdays and "Weekends" for those available on weekends.

```

amsBnbDays <- amsterdam_weekdays
athBnbDays <- athens_weekdays
barBnbDays <- barcelona_weekdays
berBnbDays <- berlin_weekdays
budBnbDays <- budapest_weekdays

```

```

lisBnbDays <- lisbon_weekdays
lonBnbDays <- london_weekdays
parBnbDays <- paris_weekdays
romBnbDays <- rome_weekdays
vieBnbDays <- vienna_weekdays

amsBnbEnds <- amsterdam_weekends
athBnbEnds <- athens_weekends
barBnbEnds <- barcelona_weekends
berBnbEnds <- berlin_weekends
budBnbEnds <- budapest_weekends
lisBnbEnds <- lisbon_weekends
lonBnbEnds <- london_weekends
parBnbEnds <- paris_weekends
romBnbEnds <- rome_weekends
vieBnbEnds <- vienna_weekends

```

## Cities

```

city1 <- 'Amsterdam'
city2 <- 'Athens'
city3 <- 'Barcelona'
city4 <- 'Berlin'
city5 <- 'Budapest'
city6 <- 'Lisbon'
city7 <- 'London'
city8 <- 'Paris'
city9 <- 'Rome'
city10 <- 'Vienna'

```

## weekdays

```

amsBnbDays$...1 <- city1
colnames(amsBnbDays)[1] <- "city"
amsBnbDays$`Weekdays/Weekends` <- 'Weekdays'
View(amsBnbDays)

athBnbDays$...1 <- city2
colnames(athBnbDays)[1] <- "city"
athBnbDays$`Weekdays/Weekends` <- 'Weekdays'

barBnbDays$...1 <- city3
colnames(barBnbDays)[1] <- "city"
barBnbDays$`Weekdays/Weekends` <- 'Weekdays'

berBnbDays$...1 <- city4
colnames(berBnbDays)[1] <- "city"

```

```

berBnbDays$`Weekdays/Weekends` <- 'Weekdays'

budBnbDays$...1 <- city5
colnames(budBnbDays)[1] <- "city"
budBnbDays$`Weekdays/Weekends` <- 'Weekdays'

lisBnbDays$...1 <- city6
colnames(lisBnbDays)[1] <- "city"
lisBnbDays$`Weekdays/Weekends` <- 'Weekdays'

lonBnbDays$...1 <- city7
colnames(lonBnbDays)[1] <- "city"
lonBnbDays$`Weekdays/Weekends` <- 'Weekdays'

parBnbDays$...1 <- city8
colnames(parBnbDays)[1] <- "city"
parBnbDays$`Weekdays/Weekends` <- 'Weekdays'

romBnbDays$...1 <- city9
colnames(romBnbDays)[1] <- "city"
romBnbDays$`Weekdays/Weekends` <- 'Weekdays'

vieBnbDays$...1 <- city10
colnames(vieBnbDays)[1] <- "city"
vieBnbDays$`Weekdays/Weekends` <- 'Weekdays'

```

## weekends

```

amsBnbEnds$...1 <- city1
colnames(amsBnbEnds)[1] <- "city"
amsBnbEnds$`Weekdays/Weekends` <- 'Weekends'

athBnbEnds$...1 <- city2
colnames(athBnbEnds)[1] <- "city"
athBnbEnds$`Weekdays/Weekends` <- 'Weekends'

barBnbEnds$...1 <- city3
colnames(barBnbEnds)[1] <- "city"
barBnbEnds$`Weekdays/Weekends` <- 'Weekends'

berBnbEnds$...1 <- city4
colnames(berBnbEnds)[1] <- "city"
berBnbEnds$`Weekdays/Weekends` <- 'Weekends'

budBnbEnds$...1 <- city5

```

```

colnames(budBnbEnds)[1] <- "city"
budBnbEnds$`Weekdays/Weekends` <- 'Weekends'

lisBnbEnds$...1 <- city6
colnames(lisBnbEnds)[1] <- "city"
lisBnbEnds$`Weekdays/Weekends` <- 'Weekends'

lonBnbEnds$...1 <- city7
colnames(lonBnbEnds)[1] <- "city"
lonBnbEnds$`Weekdays/Weekends` <- 'Weekends'

parBnbEnds$...1 <- city8
colnames(parBnbEnds)[1] <- "city"
parBnbEnds$`Weekdays/Weekends` <- 'Weekends'

romBnbEnds$...1 <- city9
colnames(romBnbEnds)[1] <- "city"
romBnbEnds$`Weekdays/Weekends` <- 'Weekends'

vieBnbEnds$...1 <- city10
colnames(vieBnbEnds)[1] <- "city"
vieBnbEnds$`Weekdays/Weekends` <- 'Weekends'

```

## Renaming the columns

```

newColumns <- c('City', 'Listing Price', 'Room Type', 'Shared Room', 'Private Room',
               'Room Capacity', 'Superhost', 'Multiple Listings', 'Business Listings',
               'Cleanliness Rating', 'Overall Guest Satisfaction', 'No. of Bedrooms', 'Distance from City',
               'Distance from Metro', 'Attraction Index', 'Attraction Index (Normalised)', 'Restaurant Index',
               'Restaurant Index (Normalised)', 'Longitude', 'Latitude', 'Weekdays/Weekends')

colnames(amsBnbDays) <- newColumns
colnames(athBnbDays) <- newColumns
colnames(barBnbDays) <- newColumns
colnames(berBnbDays) <- newColumns
colnames(budBnbDays) <- newColumns
colnames(lisBnbDays) <- newColumns
colnames(lonBnbDays) <- newColumns
colnames(parBnbDays) <- newColumns
colnames(romBnbDays) <- newColumns
colnames(vieBnbDays) <- newColumns

colnames(amsBnbEnds) <- newColumns
colnames(athBnbEnds) <- newColumns
colnames(barBnbEnds) <- newColumns
colnames(berBnbEnds) <- newColumns
colnames(budBnbEnds) <- newColumns
colnames(lisBnbEnds) <- newColumns
colnames(lonBnbEnds) <- newColumns

```

```
colnames(parBnbEnds) <- newColumns  
colnames(romBnbEnds) <- newColumns  
colnames(vieBnbEnds) <- newColumns
```

## Combining the data for each city

```
amsBnbAll <- merge(amsBnbDays, amsBnbEnds, all = TRUE)  
athBnbAll <- merge(athBnbDays, athBnbEnds, all = TRUE)  
barBnbAll <- merge(barBnbDays, barBnbEnds, all = TRUE)  
berBnbAll <- merge(berBnbDays, berBnbEnds, all = TRUE)  
budBnbAll <- merge(budBnbDays, budBnbEnds, all = TRUE)  
lisBnbAll <- merge(lisBnbDays, lisBnbEnds, all = TRUE)  
lonBnbAll <- merge(lonBnbDays, lonBnbEnds, all = TRUE)  
parBnbAll <- merge(parBnbDays, parBnbEnds, all = TRUE)  
romBnbAll <- merge(romBnbDays, romBnbEnds, all = TRUE)  
vieBnbAll <- merge(vieBnbDays, vieBnbEnds, all = TRUE)
```

## merging all the rows together

```
eurobnb_trips <- bind_rows(amsBnbDays, amsBnbEnds, athBnbDays, athBnbEnds, barBnbDays, barBnbEnds, berBnbDays, berBnbEnds, budBnbDays, budBnbEnds, lisBnbDays, lisBnbEnds, lonBnbDays, lonBnbEnds, parBnbDays, parBnbEnds, romBnbDays, romBnbEnds, vieBnbDays, vieBnbEnds)
```

## Does listing availability influence guess satisfaction score?

now to answer some of the questions

To determine if listing availability affects guest satisfaction scores, we'll analyze the 'Weekdays/Weekends' column created during data loading.

Initially, we'll categorize the 'Overall Guest Satisfaction' column into good and bad scores. Typically, satisfaction ratings ranging from 76 to 85 are deemed good, though this may vary. For this analysis, scores of 75 or lower will be classified as bad. To facilitate this, we'll create a function to label the scores as "Good" or "Bad".

## Dividing the dataset between high scores and low scores

```
GvB <- function(score) {  
  if (score > 75) {  
    return('Good')  
  } else {  
    return('Bad')  
  }  
}
```

```
eurobnb_trips$"Good/Bad Score" <- sapply(eurobnb_trips$`Overall Guest Satisfaction`, GvB)
```

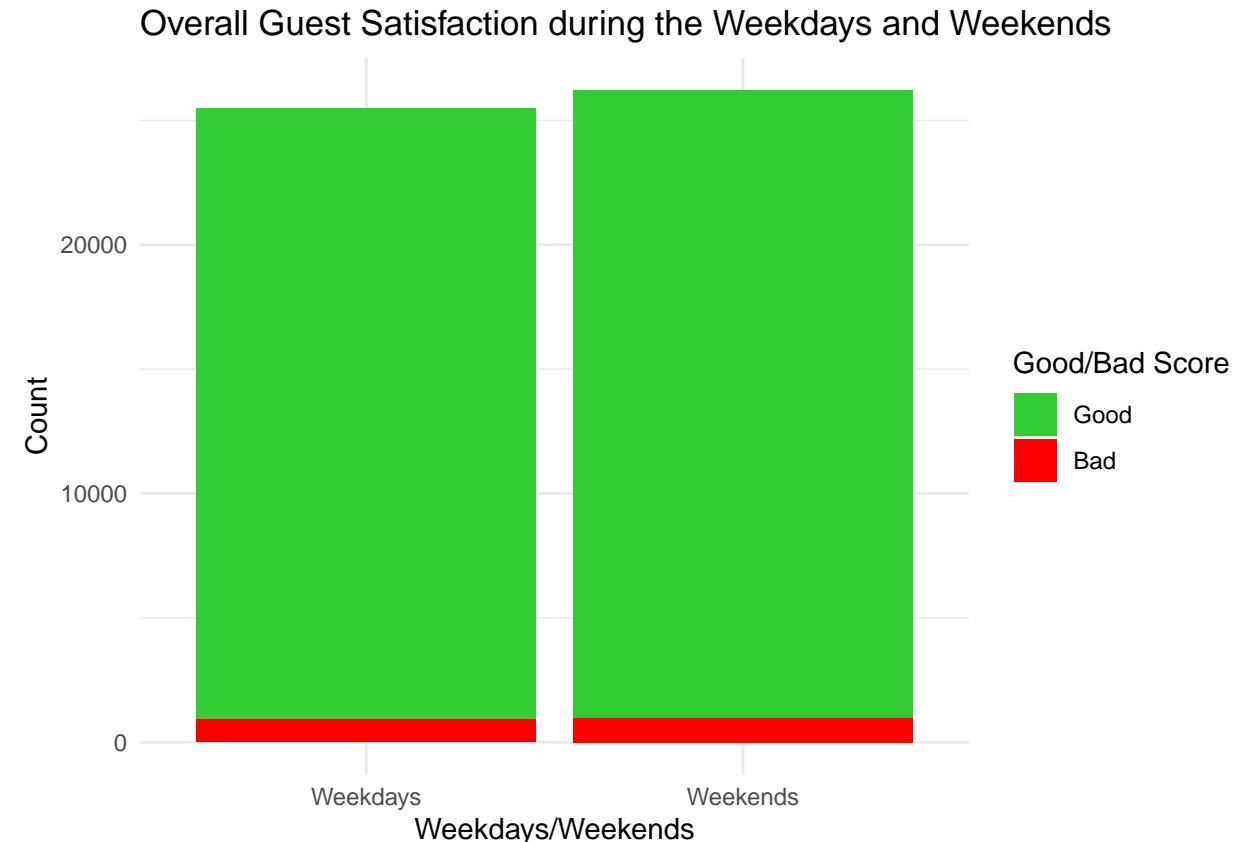
## Group the data and calculate the counts

```
deGvB_Grouped <- eurobnb_trips %>%
  group_by(`Weekdays/Weekends`, `Good/Bad Score`) %>%
  summarise(Count = n(), .groups = "drop")
```

```
deGvB_Grouped$`Good/Bad Score` <- factor(deGvB_Grouped$`Good/Bad Score`, levels = c("Good", "Bad"))
```

## Create the stacked bar plot using ggplot2

```
ggplot(deGvB_Grouped, aes(x = factor(`Weekdays/Weekends`), y = Count, fill = `Good/Bad Score`)) +
  geom_bar(position = "stack", stat = "identity") + # Use stat = "count" to automatically count the number of observations
  labs(title = "Overall Guest Satisfaction during the Weekdays and Weekends",
       x = "Weekdays/Weekends",
       y = "Count") +
  theme_minimal() + # Set theme (optional)
  scale_fill_manual(values = c("limegreen", "red"))
```



## Changing the values in ‘Superhost’ from True and False to ‘Superhost’ and ‘Regular Host’ respectively

```
SvsR <- function(host) {  
  if (host == TRUE) {  
    return('Superhost')  
  } else {  
    return('Regular Host')  
  }  
}
```

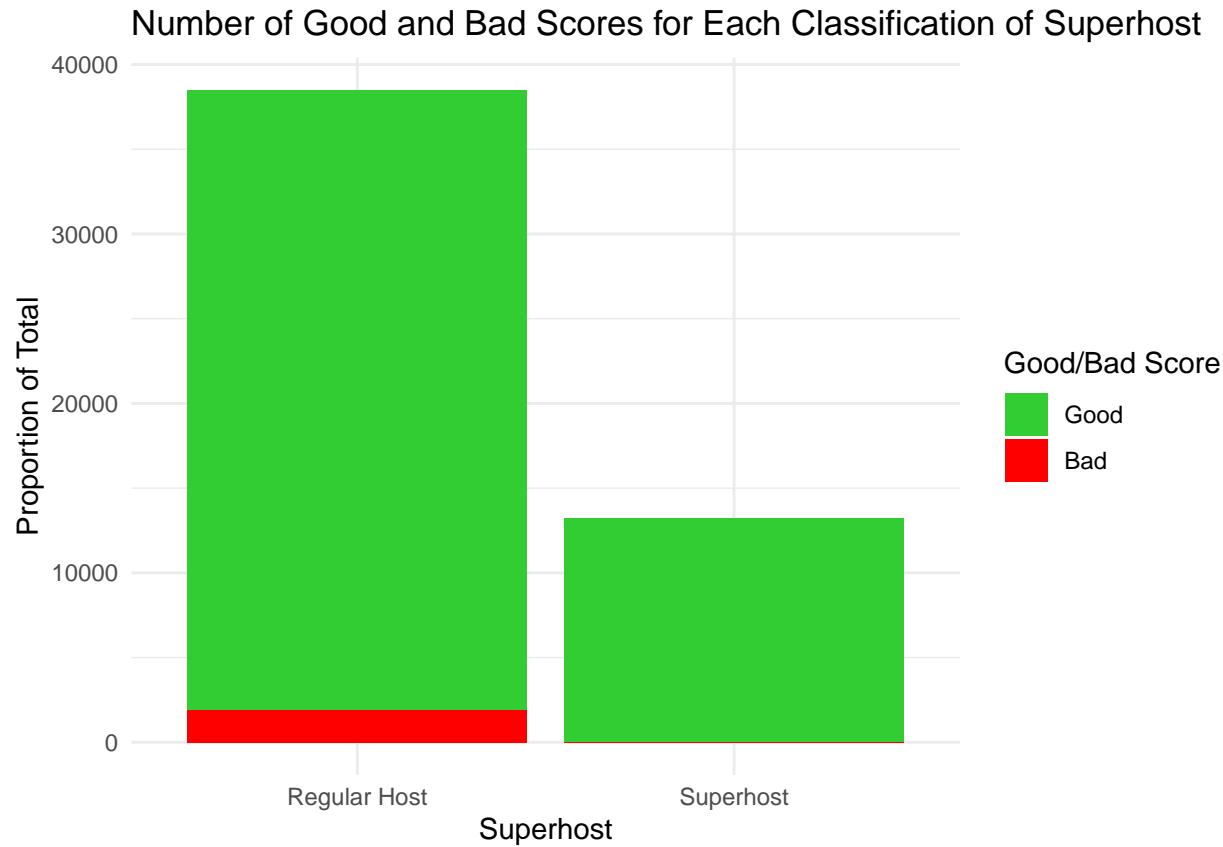
applying the function to the ‘Superhost’ column.

```
eurobnb_trips$Superhost <- ifelse(eurobnb_trips$Superhost == TRUE, "Superhost", "Regular Host")
```

With that done, I'll make a bar graph to illustrate the number of good and bad scores for each classification of Superhost.

```
superhostGvBGrouped <- eurobnb_trips %>%  
  group_by(Superhost, `Good/Bad Score`) %>%  
  summarise(Count = n(), .groups = "drop") %>%  
  ungroup()
```

```
superhostGvBGrouped$`Good/Bad Score` <- factor(superhostGvBGrouped$`Good/Bad Score`, levels = c("Good",  
  "Bad"))  
  
ggplot(superhostGvBGrouped, aes(x = Superhost, y = Count, fill = `Good/Bad Score`)) +  
  geom_bar(position = "stack", stat = "identity") +  
  labs(title = "Number of Good and Bad Scores for Each Classification of Superhost",  
    x = "Superhost",  
    y = "Proportion of Total") +  
  scale_fill_manual(values = c("limegreen", "red")) + # Customize fill colors  
  theme_minimal() # Set theme (optional)
```



This visualization indicates a striking disparity: listings from superhosts with poor guest satisfaction ratings are exceedingly rare. Conversely, there is a notable presence of listings with unfavorable scores from regular hosts. Given that superhosts are recognized for their exceptional hospitality, it's more probable for listings from regular hosts to receive lower ratings in comparison to those from superhosts.

## Does the number of listings the host has influence guest satisfaction?

The Airbnb dataset categorizes listings based on the number of properties offered by hosts. These categories include hosts with a single listing, hosts with 2 to 4 listings (referred to as “Multiple Listings”), and hosts with more than 4 listings (referred to as “Business Listings”). To represent these classifications more conveniently, I introduced a new column in the dataset.

## Creating a function that will make classification easier

```
MvsB <- function(m, b) {
  if (m == 0) {
    if (b == 1) {
      return('Business Listings')
    } else if (b == 0) {
      return('One Listing')
    }
  }
}
```

```

    }
} else if (m == 1) {
  return('Multiple Listings')
}
}

```

applying the function to the columns ‘Multiple Listings’ and ‘Business Listings’.

```

eurobnb_trips$Multiple_Business <- apply(eurobnb_trips, 1, function(x) {
  MvsB(x[['Multiple Listings']], x[['Business Listings']])
})

```

```

MBGvBGroupedTry <- eurobnb_trips %>%
  group_by(Multiple_Business, `Good/Bad Score`) %>%
  summarise(Count = n(), .groups = "drop") %>%
  ungroup()

```

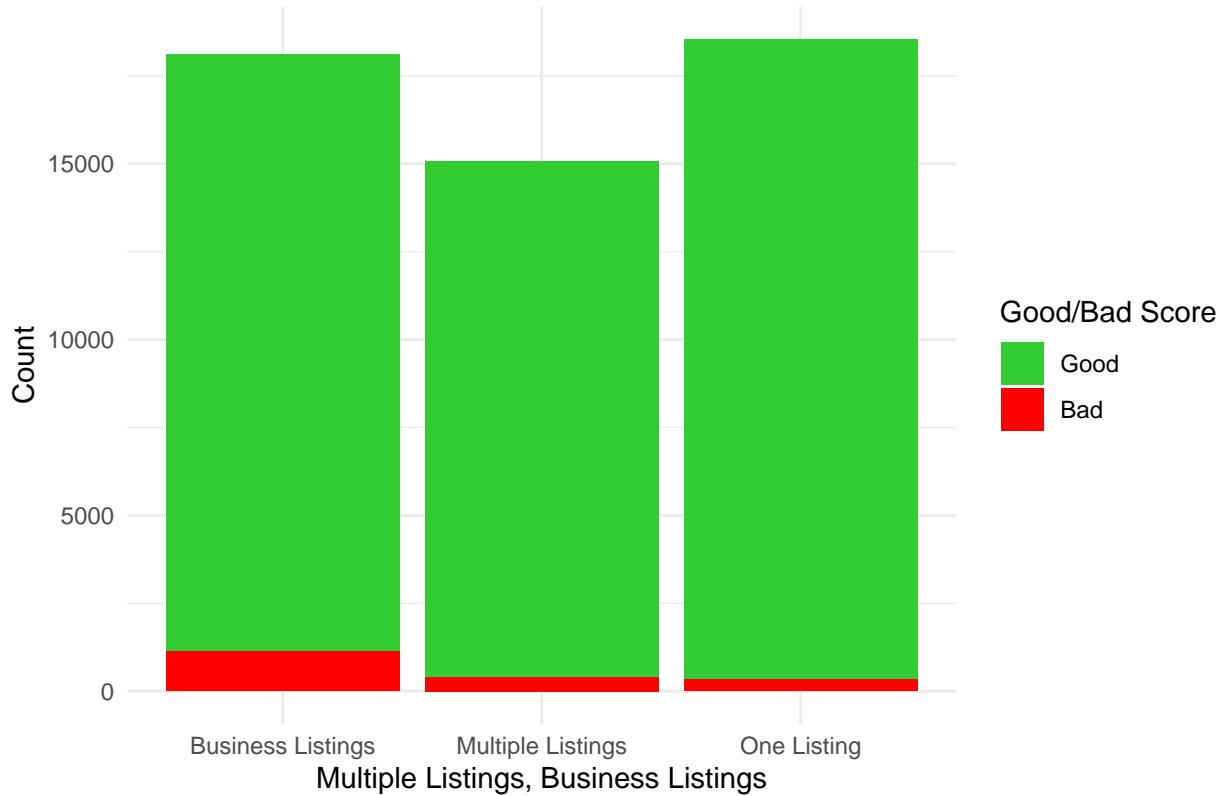
```

MBGvBGroupedTry$`Good/Bad Score` <- factor(MBGvBGroupedTry$`Good/Bad Score`, levels = c("Good", "Bad"))

ggplot(MBGvBGroupedTry, aes(x = Multiple_Business, y = Count, fill = `Good/Bad Score`)) +
  geom_bar( position = "stack", stat = "identity") +
  labs(title = "Guest Satisfaction Scores by Multiple Listings and Business Listings",
       x = "Multiple Listings, Business Listings",
       y = "Count") +
  theme_minimal() +
  scale_fill_manual(values = c("limegreen", "red"))

```

## Guest Satisfaction Scores by Multiple Listings and Business Listings

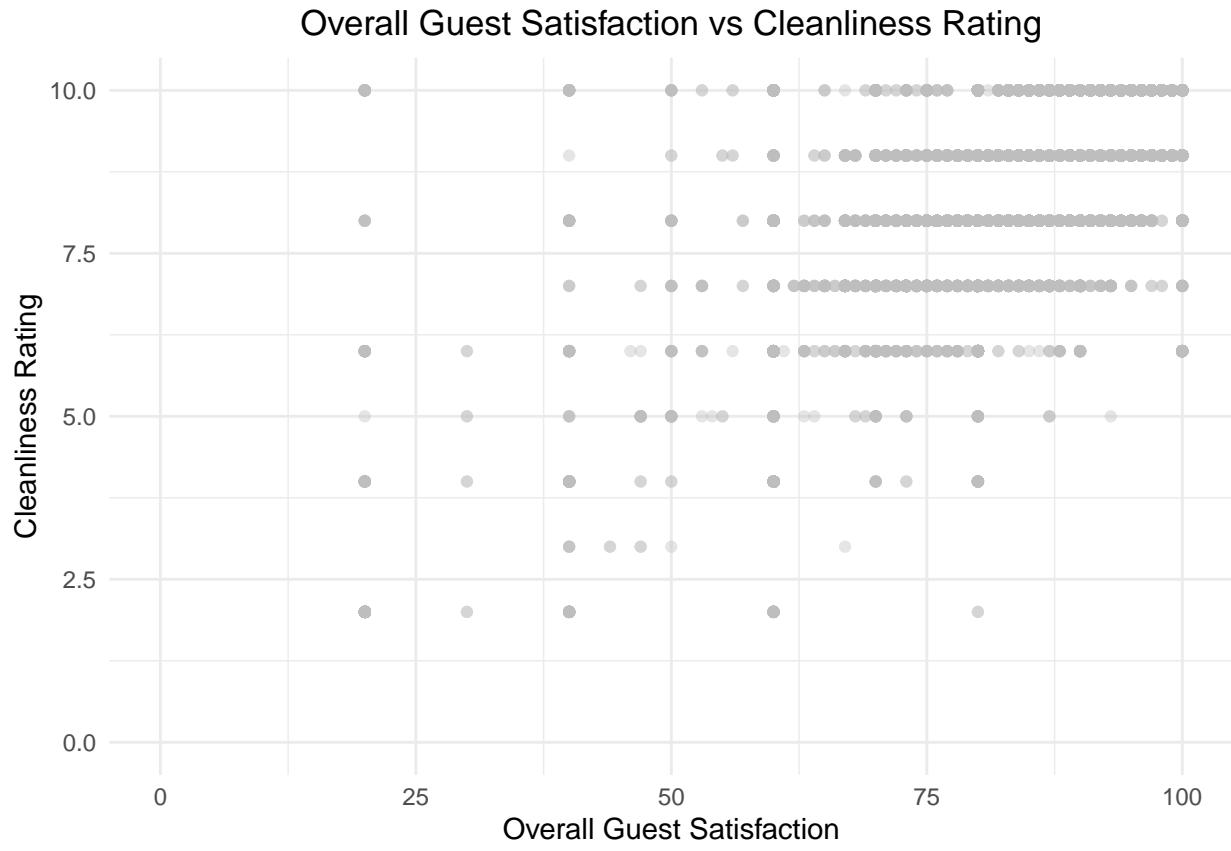


This bar graph indicates that properties managed by hosts with at least four other listings received the highest number of poor scores. This trend might be attributed to the increased workload of hosts managing multiple properties.

## Is there a correlation between cleanliness rating and guest satisfaction?

Cleanliness rating is scored from 1 to 10. To find the correlation, I will illustrate it using a scatter plot. To make overlapping points much easier to distinguish, I increased the transparency of each point.

```
plot <- ggplot(eurobnb_trips, aes(x = `Overall Guest Satisfaction`, y = `Cleanliness Rating`)) +  
  geom_point(alpha = 0.4, color = 'gray') +  
  labs(title = 'Overall Guest Satisfaction vs Cleanliness Rating',  
       x = 'Overall Guest Satisfaction',  
       y = 'Cleanliness Rating') +  
  theme_minimal() +  
  theme(plot.title = element_text(hjust = 0.5)) +  
  scale_x_continuous(limits = c(0, NA)) +  
  scale_y_continuous(limits = c(0, NA))  
  
print(plot)
```



Based on the plot, numerous listings receiving high scores also exhibit a cleanliness rating of 6 or higher. Conversely, listings with guest satisfaction scores below 75 display a range of cleanliness ratings. Interestingly, some listings with cleanliness ratings exceeding 6 received the lowest satisfaction scores. This suggests that while cleanliness remains crucial in this domain, other factors may contribute more significantly to lower satisfaction ratings.

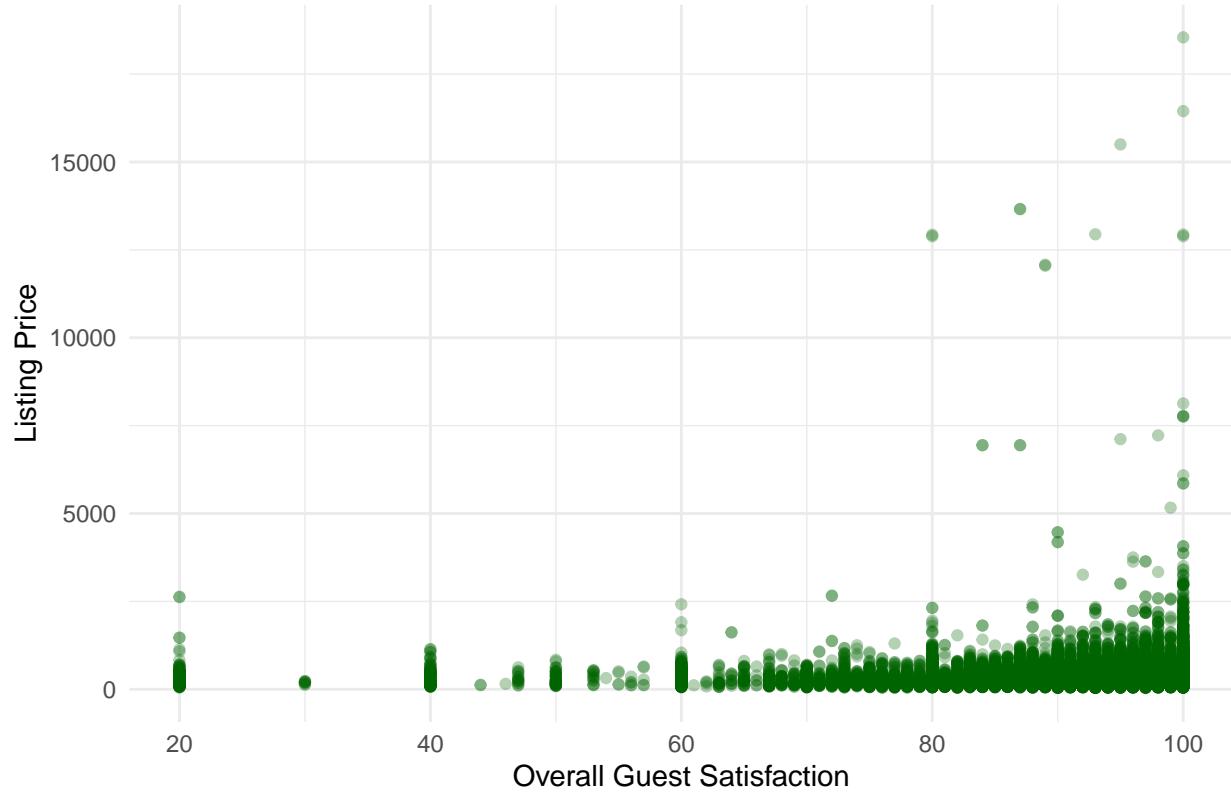
### Is there a correlation between listing price and guest satisfaction?

Again, I use a scatter plot to illustrate the correlation between listing price and guest satisfaction.

```
plot1 <- ggplot(eurobnb_trips, aes(x = `Overall Guest Satisfaction`, y = `Listing Price`)) +
  geom_point(alpha = 0.3, color = 'darkgreen') +
  labs(title = 'Overall Guest Satisfaction vs Listing Price',
       x = 'Overall Guest Satisfaction',
       y = 'Listing Price') +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5)) +
  coord_cartesian(ylim = c(0, NA)) +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme(legend.position = "none")
```

```
print(plot1)
```

## Overall Guest Satisfaction vs Listing Price



Listing prices are mostly below 5000, so while there are AirBnBs with low listing prices and bad scores, it is not the factor that leads to low guest satisfaction.

Is there a correlation between the distance from the metro affect guest satisfaction?

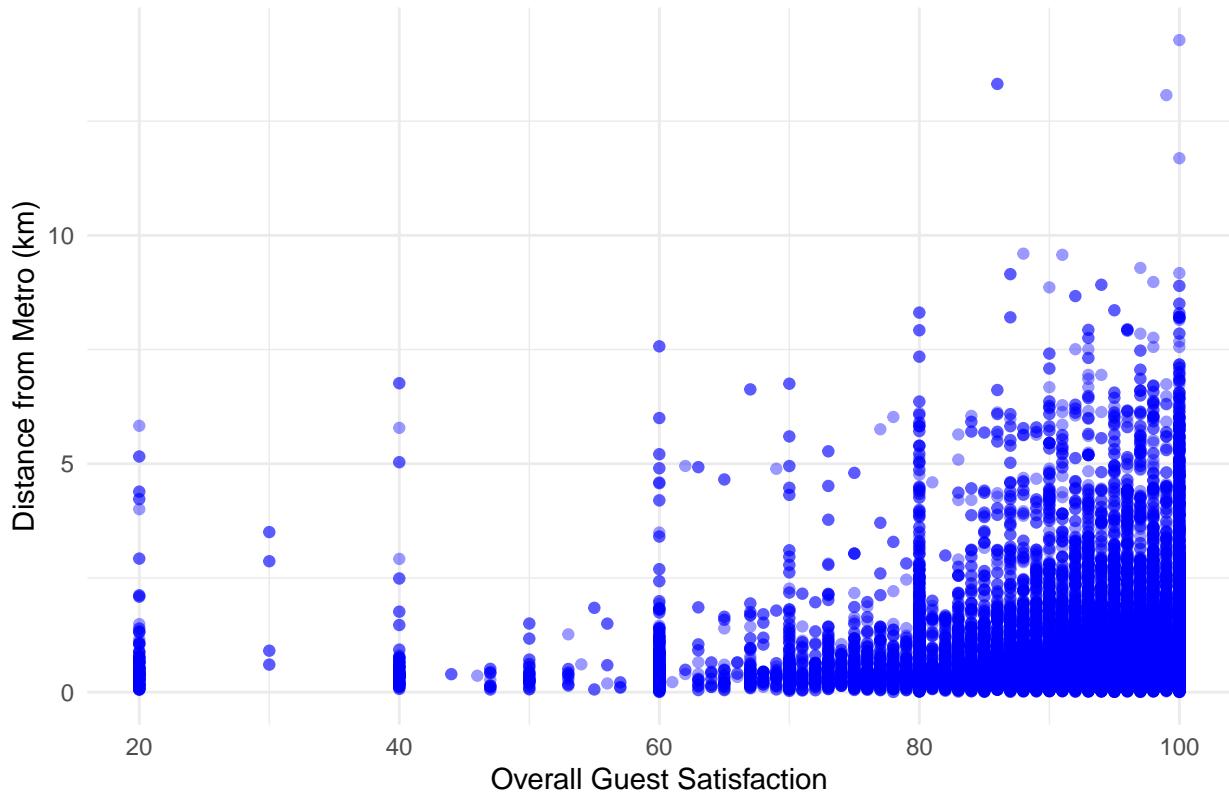
For this question, I am working under the assumption that listings closer to the metro are more favorable to guests. To illustrate the relationship between the distance and guest satisfaction, I again use a scatter plot.

## Looking for the correlation using a scatter plot

```
plot2 <- ggplot(eurobnb_trips, aes(x = `Overall Guest Satisfaction`, y = `Distance from Metro`)) +
  geom_point(alpha = 0.4, color = "blue") +
  labs(title = "Overall Guest Satisfaction vs Distance from Metro (km)",
       x = "Overall Guest Satisfaction",
       y = "Distance from Metro (km)") +
  theme_minimal()
```

```
print(plot2)
```

Overall Guest Satisfaction vs Distance from Metro (km)



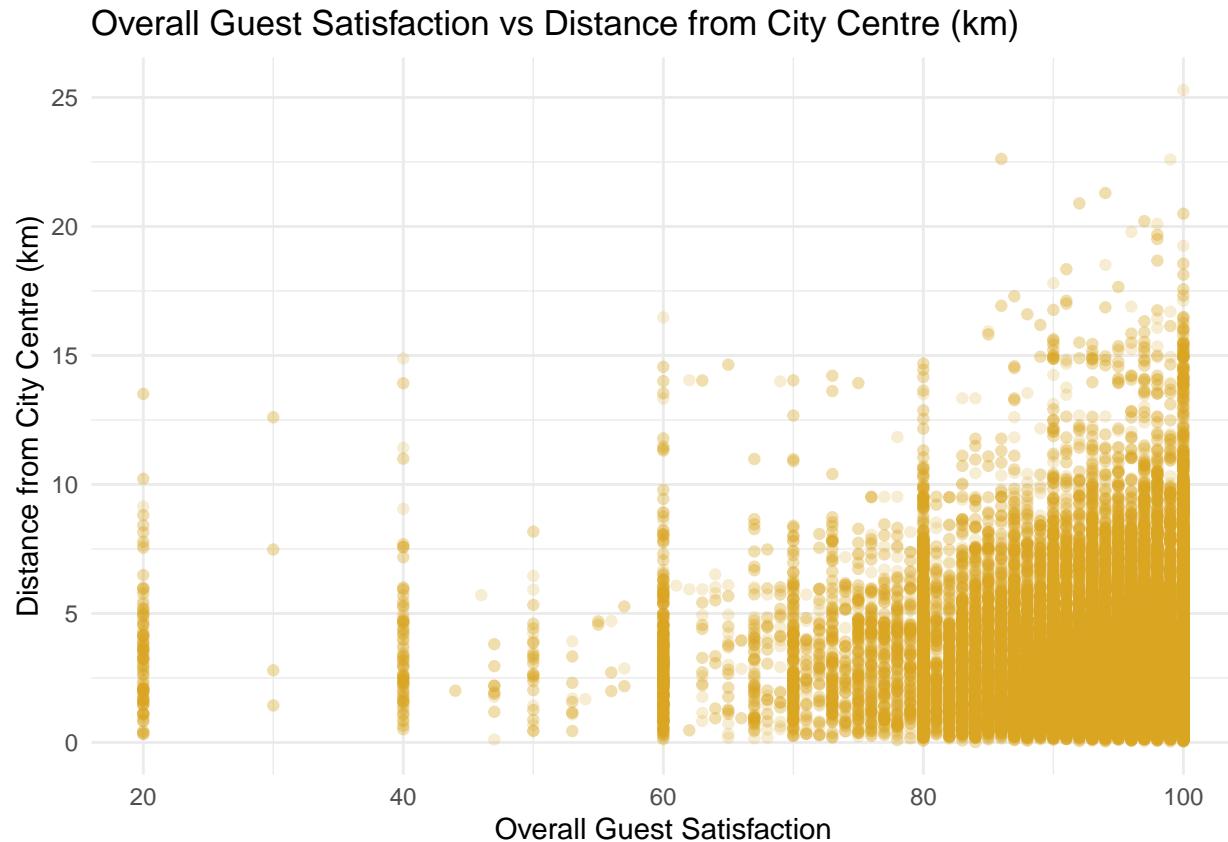
This plot illustrates that the majority of listings are typically situated within a distance of less than 6 km from the metro, with many of them receiving high satisfaction scores. Although there are some outliers suggesting that a greater distance from the metro could result in higher guest satisfaction, these instances are too infrequent to draw definitive conclusions. Regarding the listings with poor scores despite being within 6 km of the metro (a significant portion of which are within 2 km), it suggests that factors other than proximity to the metro may be influencing their scores negatively.

## Is there a correlation between the distance from the city centre affect guest satisfaction?

Similar to the previous inquiry, I operate under the assumption that guests generally prefer accommodations closer to the city center, where attractions, restaurants, and shops are typically located

```
plot3 <- ggplot(eurobnb_trips, aes(x = `Overall Guest Satisfaction`, y = `Distance from City Centre`)) +
  geom_point(alpha = 0.2, color = 'goldenrod') +
  labs(title = 'Overall Guest Satisfaction vs Distance from City Centre (km)',
       x = 'Overall Guest Satisfaction',
       y = 'Distance from City Centre (km)') +
  theme_minimal()
```

```
print(plot3)
```



This visualization illustrates that the majority of listings are typically located within 10 km of the city center, with many of them achieving high guest satisfaction scores. Although there are outliers suggesting that being farther from the city center could result in higher guest satisfaction, these instances are rare and not statistically significant. Additionally, listings with low scores despite their proximity to the city center may have other factors influencing their ratings.

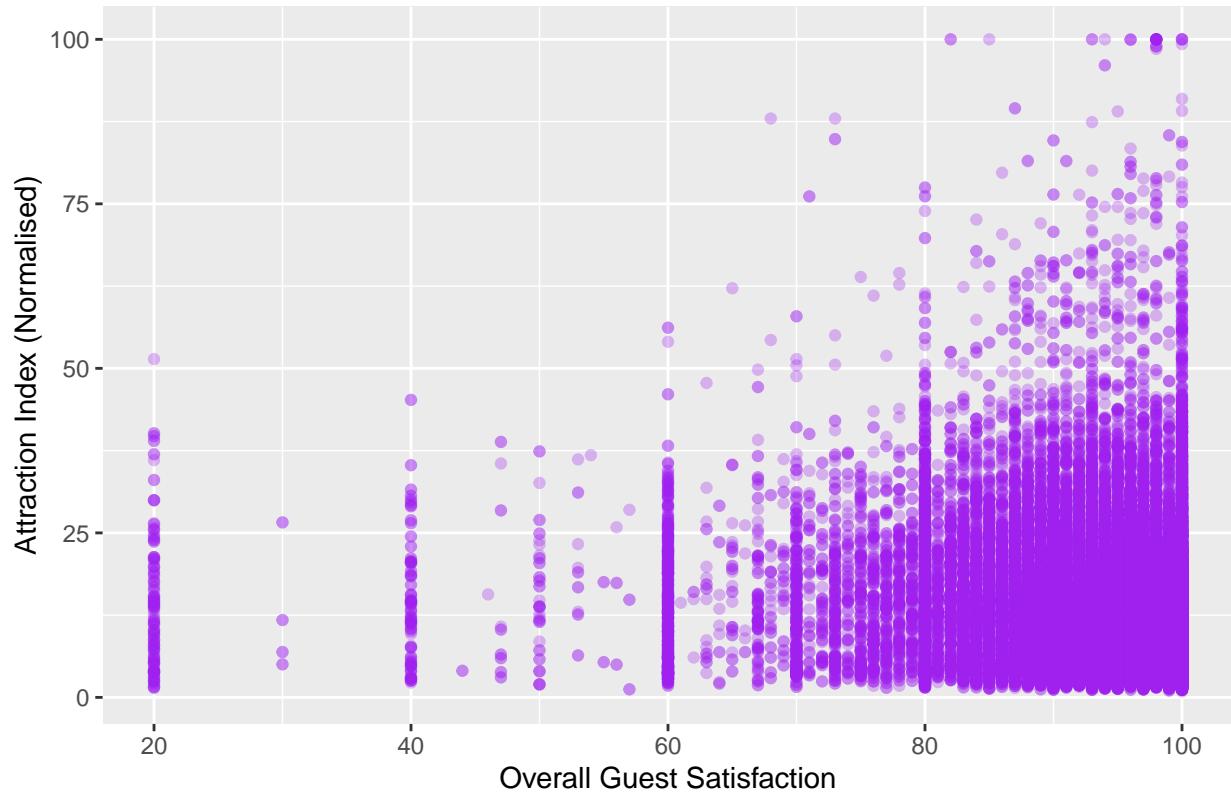
## Is there a correlation between the attraction index and guest satisfaction?

As per the dataset's origin, the attraction index quantifies the accessibility and popularity of attractions near a specific listing. A higher index value indicates that popular tourist attractions are closer to the listing.

```
plot4 <- ggplot ( eurobnb_trips, aes(x = `Overall Guest Satisfaction`, y = `Attraction Index (Normalised)`),
  geom_point(color = 'purple', alpha = 0.3) +
  labs(title = 'Overall Guest Satisfaction vs Attraction Index (Normalised)',
       x = 'Overall Guest Satisfaction',
       y = 'Attraction Index (Normalised)')
```

```
print(plot4)
```

## Overall Guest Satisfaction vs Attraction Index (Normalised)



The plot indicates that the majority of listings have attraction indexes around 40. However, there is a noticeable trend where listings with higher attraction indexes tend to have higher guest satisfaction scores. It's worth noting that low attraction indexes may not necessarily result in bad guest satisfaction scores, although it's notable that there are relatively few listings with attraction indexes exceeding 50.

## Is there a correlation between the restaurant index and guest satisfaction?

The restaurant index functions similarly to the attraction index but focuses on restaurants instead. A higher index value indicates closer proximity to popular restaurants relative to the listing.

## Create the scatter plot

```
plot5 <- ggplot( eurobnb_trips, aes(x = `Overall Guest Satisfaction`, y = `Restaurant Index (Normalised)`)) +  
  geom_point(alpha = 0.3, color = "coral") +  
  labs(title = "Overall Guest Satisfaction vs Restaurant Index (Normalised)") +  
  theme_minimal() +  
  theme(plot.title = element_text(hjust = 0.5)) +  
  xlim(0, NA) +  
  ylim(0, NA)
```

```
print(plot5)
```



According to the plot, there's a positive correlation between high restaurant indexes and higher guest satisfaction scores. While some listings with low guest satisfaction scores have high restaurant indexes, the majority of them tend to have indexes of 60 or lower. Notably, the lowest-rated listings generally have low restaurant indexes. Although other factors may contribute to low guest satisfaction, low restaurant indexes could be a significant factor to consider.

## conclusion

In conclusion, several factors were investigated to understand the reasons behind low guest satisfaction scores:

Low restaurant indexes. Low attraction indexes. Hosts with more than 4 listings simultaneously. Superhosts tend to avoid listing properties that receive low scores. Listing availability does not significantly impact guest satisfaction scores. Cleanliness is important, but some listings may still score poorly despite being clean. Listing price does not play a significant role in guest satisfaction. Regarding the distance from listings to the metro, it was initially assumed that proximity to the metro would enhance guest satisfaction. However, some listings scored poorly despite their proximity to the metro. Other factors, such as the absence of good restaurants or attractions nearby, may have influenced these scores.

Similarly, closeness to the city center was expected to satisfy guests due to the availability of restaurants, tourist attractions, and amenities. While many listings near the city center received high scores, others did not. This could be attributed to factors beyond the dataset's scope, such as the city's overall environment or specific property characteristics.

#END