

DataScience_FinalProject

Jan Steinwender & Paul Pavlis

2020-10-28

Clean the environment variables (*Temporary*)

Remove this whole header before handing the project in. This is just so that working with the document is easier

```
rm(list = ls())
```

Load needed libraries

```
library(tidyverse)

## -- Attaching packages ----- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr    0.3.4
## v tibble   3.0.3     v dplyr    1.0.2
## v tidyr    1.1.2     v stringr  1.4.0
## v readr    1.3.1     vforcats  0.5.0

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()   masks stats::lag()

library(ggplot2)
```

Data wrangling

Data import from csv file

Die Daten werden mit `read.csv` eingelesen, da dadurch alle Spalten richtig eingelesen werden, was aufgrund der Spalte **Cuisine.Style** anders nicht so leicht geht, da diese Spalte ein Python List object enthält, was als Trennzeichen das Gleiche Trennzeichen verwendet wie bei der Trennung der verschiedenen Spalten.

Außerdem werden leere Einträge durch NA Werte ersetzt, da Funktionen mit diesen besonders umgehen können.

```

restaurant_data = read.csv(file = "restaurants_data.csv")
# Set empty data entries to NA
restaurant_data = restaurant_data %>% na_if("")
```

Clean the data

Rename and remove columns

Die Spaltennamen werden auf angenehmerer Art dargestellt und die Spalten welche keine benötigten Infos liefern e.g. **X**, **URL_TA** & **ID_TA** werden entfernt.

```

restaurant_data = as_tibble(restaurant_data)

restaurant_data = restaurant_data %>%
  rename("Cuisine_Style" = Cuisine.Style) %>%
  rename("Price_Range" = Price.Range) %>%
  rename("Review_Count" = Number.of.Reviews)

# Remove useless columns
restaurant_data = restaurant_data %>%
  mutate(X = NULL, URL_TA = NULL, ID_TA = NULL)
```

Correct the column types

Die Spalte **City** wird von einer Textvariable zu einer kategorialen Faktorvariable geändert, da die Städte öfter vorkommen und dadurch mehr Infos aus der Spalte geholt werden können. Diese Variable besitzt 31 Ausprägungen

Die Spalten **Price_Range** und **Rating** werden aus den selben Gründen ebenso in einen Faktor umgewandelt. Die Variable **Price_Range** besitzt 3 Ausprägungen, die für leichtere Verständlichkeit umbenannt werden in: *low*, *medium* & *high*. Die Variable **Rating** besitzt 9 Ausprägungen.

```

# Change from character to factor
restaurant_data = restaurant_data %>% mutate(City = as_factor(City))

# Change from character to factor
restaurant_data = restaurant_data %>% mutate(Price_Range = as_factor(Price_Range))
# Rename the levels
restaurant_data = restaurant_data %>% mutate(Price_Range = fct_recode(
  Price_Range,
  "high" = "$$$$",
  "medium" = "$$ - $$",
  "low" = "$"
))

# Set invalid entries with a rating of -1 to NA
restaurant_data$Rating = restaurant_data$Rating %>% na_if("-1")
# Change from character to factor
restaurant_data = restaurant_data %>% mutate(Rating = as_factor(Rating))
```

Check duplicated and NULL values

Ursprünglich enthält der Datensatz 286 Datensätze die doppelt vorkommen. Diese werden entfernt.

Insgesamt enthält der Datensatz 123992 NA Werte. Diese sind wie folgt auf die einzelnen Spalten aufgeteilt:

```
duplicated(restaurant_data) %>% sum()

## [1] 289

# Remove duplicates
restaurant_data = restaurant_data %>% distinct()

is.na(restaurant_data) %>% sum()

## [1] 124022

sapply(restaurant_data, function(x) sum(is.na(x)))

##          Name          City Cuisine_Style      Ranking       Rating
##          0            0        31222        9370        9389
##  Price_Range  Review_Count     Reviews
##        47642         17062        9337
```

Summary

Der Datensatz handelt von TripAdvisor Bewertungen vieler Restaurants von 31 europäischen Städten.

Der endgültige Datensatz mit dem wir in diesem Projekt arbeiten werden besteht aus *acht* Variablen (Spalten) mit 125.238 Observationen (Zeilen). Diese beinhalten:

Name: Name des Restaurants - Textvariable (unique)

City: Stadt in der sich das Restaurant befindet - Kategoriale Faktorvariable mit 31 Ausprägungen (London mit 18113 Einträgen, Paris mit 14867 Einträgen, ...)

Cuisine_Style: Essensrichtungen des Restaurants - Textvariable (grundsätzlich besteht diese aus mehreren Faktoren innerhalb eines Python list Objektes / 31222 NA Werte)

Ranking: Rang des Restaurants im Vergleich zu allen anderen Restaurants in der Stadt - Diskrete Variable (Min: 1, Mean: 3658, Median: 2256, Max: 16444 / 9370 NA Werte)

Rating: Bewertung des Restaurants von 1-5 in 0.5 Schritten - Kategoriale Faktorvariable mit 9 Ausprägungen (Bewertung 4 mit 39841 Einträgen, 4.5 mit 31325 Einträgen, ... / 9389 NA Werte)

Price_Range: Preisbewertung - Kategoriale Faktorvariable mit 3 Ausprägungen (medium mit 54302 Einträgen, high mit 4306 Einträgen und low mit 18988 Einträgen / 47642 NA Werte)

Review_Count: Anzahl der Reviews - Diskrete Variable (Min: 2, Mean: 125.2, Median: 32, Max: 16478 / 17062 NA Werte)

Reviews: Zwei Reviews des Restaurants und die Daten, an dem die Reviews geschrieben wurden - Textvariable (als Python list Objekt abgespeichert)

```
restaurant_data
```

```
## # A tibble: 125,238 x 8
##   Name     City   Cuisine_Style Ranking Rating Price_Range Review_Count Reviews
##   <chr>    <fct>  <chr>        <dbl> <fct>  <fct>        <dbl> <chr>
## 1 Marti~ Amste~ ['French', 'D~      1 5     medium          136 [['Just~
## 2 De Si~ Amste~ ['Dutch', 'Eu~     2 4.5    high           812 [['Grea~
## 3 La Ri~ Amste~ ['Mediterrane~  3 4.5    high           567 [['Sati~
## 4 Vinke~ Amste~ ['French', 'E~     4 5     high           564 [['True~
## 5 Libri~ Amste~ ['Dutch', 'Eu~    5 4.5    high           316 [['Best~
## 6 Ciel ~ Amste~ ['Contemporar~  6 4.5    high           745 [['A tr~
## 7 Zaza's Amste~ ['French', 'I~    7 4.5    medium         1455 [['40th~
## 8 Blue ~ Amste~ ['Asian', 'In~   8 4.5    high           675 [['Grea~
## 9 Teppa~ Amste~ ['Japanese', ~  9 4.5    high           923 [['Grea~
## 10 Rob W~ Amste~ ['Dutch', 'Se~  10 4.5   low            450 [['Exce~
## # ... with 125,228 more rows
```

```
str(restaurant_data)
```

```
## tibble [125,238 x 8] (S3: tbl_df/tbl/data.frame)
## $ Name      : chr [1:125238] "Martine of Martine's Table" "De Silveren Spiegel" "La Rive" "Vinke...
## $ City       : Factor w/ 31 levels "Amsterdam","Athens",...: 1 1 1 1 1 1 1 1 1 1 ...
## $ Cuisine_Style: chr [1:125238] "[French", "Dutch", "European"]" "[Dutch", "European", "Vegetarian...
## $ Ranking     : num [1:125238] 1 2 3 4 5 6 7 8 9 10 ...
## $ Rating      : Factor w/ 9 levels "1","1.5","2",...: 9 8 8 9 8 8 8 8 8 8 ...
## $ Price_Range  : Factor w/ 3 levels "medium","high",...: 1 2 2 2 2 2 1 2 2 3 ...
## $ Review_Count : num [1:125238] 136 812 567 564 316 ...
## $ Reviews     : chr [1:125238] "[['Just like home', 'A Warm Welcome to Wintry Amsterdam'], ['01/03...
```

```
summary(restaurant_data)
```

```
##      Name             City   Cuisine_Style      Ranking
## Length:125238     London     :18113  Length:125238     Min.   : 1
## Class :character  Paris      :14867  Class :character  1st Qu.: 965
## Mode  :character  Madrid     : 9524   Mode  :character  Median : 2256
##                  Barcelona: 8390                    Mean   : 3658
##                  Berlin     : 7073                    3rd Qu.: 5237
##                  Milan     : 6680                    Max.   :16444
##                  (Other)    :60591                   NA's   :9370
##      Rating          Price_Range      Review_Count      Reviews
## 4     :39841   medium:54302   Min.   : 2.0  Length:125238
## 4.5   :31325   high   :4306   1st Qu.: 9.0  Class :character
## 3.5   :19744   low    :18988  Median : 32.0  Mode  :character
## 5     :11257   NA's    :47642   Mean   : 125.2
## 3     : 8522                3rd Qu.: 114.0
## (Other): 5160                Max.   :16478.0
## NA's  : 9389                NA's   :17062
```

Visualisation

Anmerkung an Jan: Ich habe hier nirgendswo die Farben geändert, da ich es beim ersten auch ned hinbekommen habe.

Anzahl der Restaurants pro Stadt

Hier zu sehen ist die Verteilung der Anzahl der Restaurants pro Stadt absteigend geordnet. Zusätzlich ist noch die Verteilung der Preis Einteilung pro Stadt zu sehen.

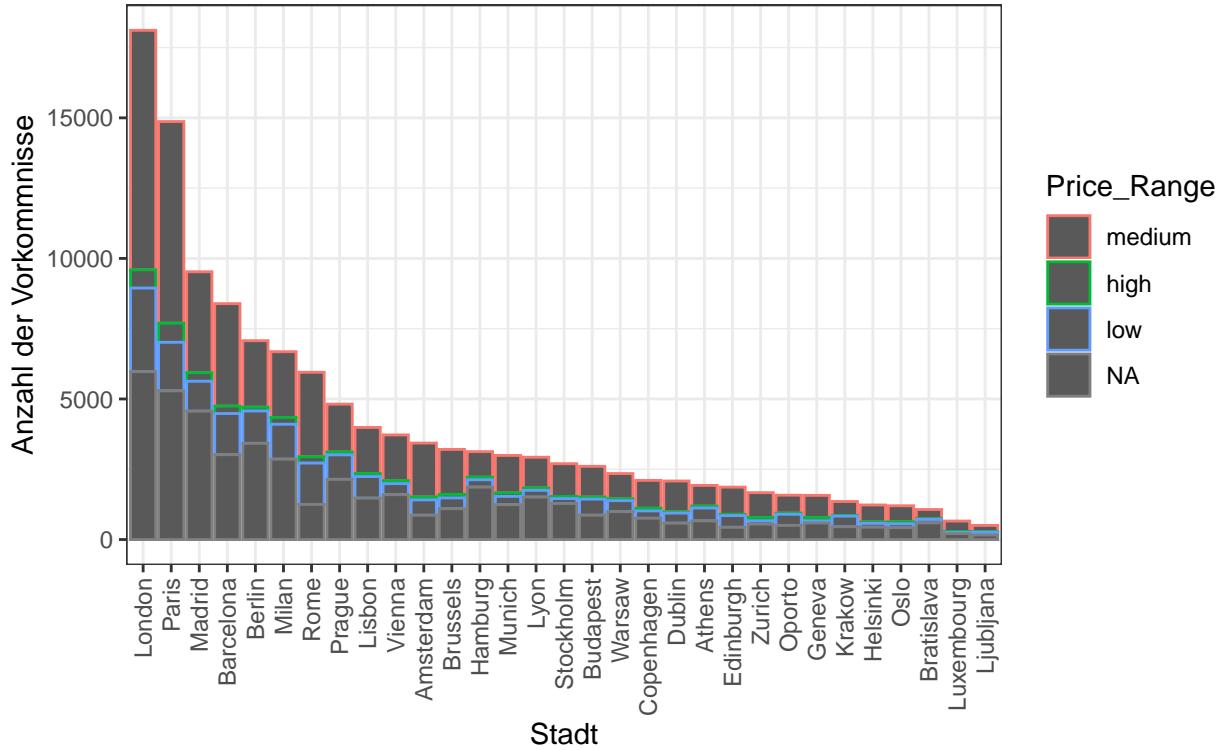
Gut zu sehen ist, dass in dem Datensatz Städte wie *London*, *Paris* sehr viele Restaurants besitzen. (Bereich ~15.000+) Danach ist ein stärkerer Abfall an Vorkommnissen zu sehen und die Städte *Luxenburg* und *Ljubljana* besitzen am wenigsten Restaurants hier. (<1000)

Auch zu erkennen ist, dass die meisten Restaurants in so gut wie allen Städten in die Preis Klasse *medium* fallen. Am wenigsten Restaurants sind in der Klasse *high*. Der Anteil an nicht klassifizierter Restaurants (NA) ist sehr hoch und ist ein beträchtlicher Anteil.

```
city_price_graph <-  
  mutate(restaurant_data, City = fct_infreq(City)) %>%  
  ggplot(aes(x = City)) + geom_bar(aes(col = Price_Range))  
  
city_price_graph + ggtitle("Anzahl der Restaurants pro Stadt",  
                           subtitle = paste0("Anzahl der Datensätze: ",  
                           length(which(  
                           !is.na(restaurant_data$City)  
                           )))) +  
  xlab("Stadt") +  
  ylab("Anzahl der Vorkommnisse") +  
  theme_bw() +  
  theme(axis.text.x = element_text(  
    angle = 90,  
    vjust = 0.5,  
    hjust = 1  
  ))
```

Anzahl der Restaurants pro Stadt

Anzahl der Datensätze: 125238



Anzahl der Restaurants Ratings

Hier zu sehen ist die Verteilung der Anzahl der Rating Stufen.

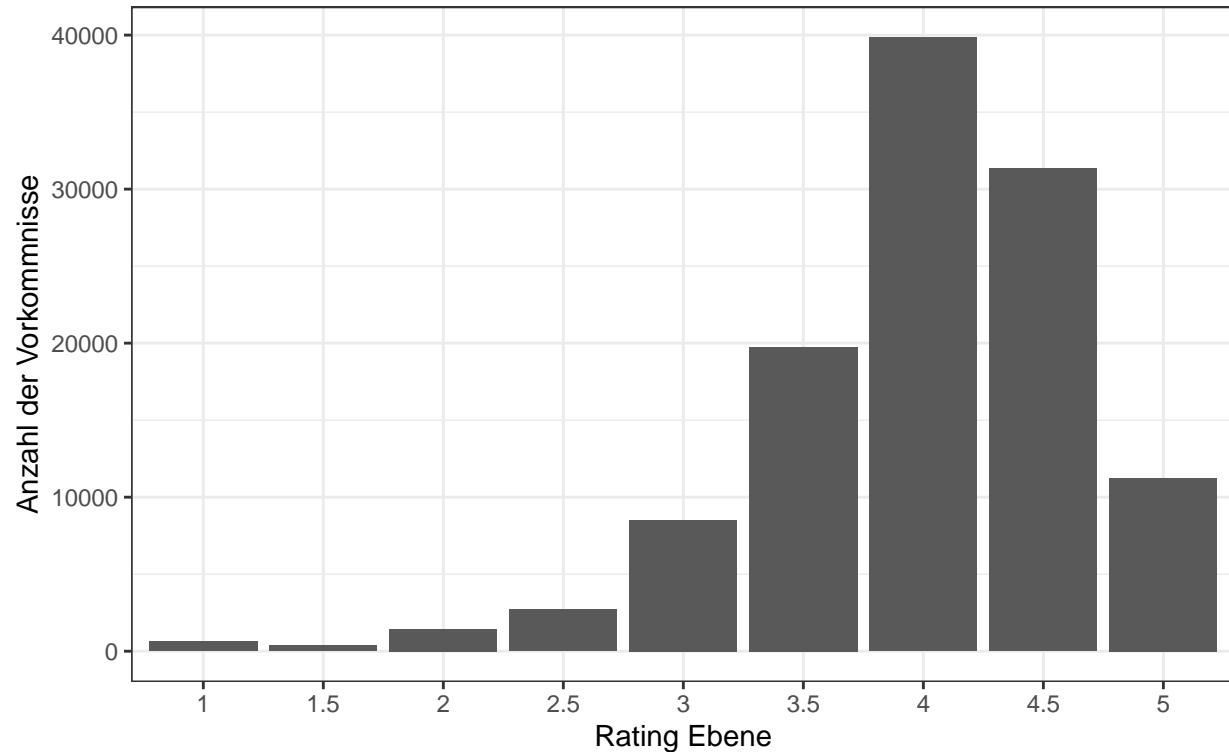
Gut zu erkennen ist, dass die Ratings 4 und 4.5 am häufigsten gegeben werden und Ratings wie 3 oder 5 viel seltener. Ratings <= 2.5 kommen am seltensten vor. Dies könnte sich dadurch erklären lassen, dass diese Restaurants sich nicht besonders gut halten werden und wahrscheinlich öfters schließen müssen als Restaurants mit besseren Bewertungen.

```
ratings_graph = restaurant_data %>%
  filter(!is.na(Rating)) %>%
  ggplot(aes(x = Rating)) + geom_bar()

ratings_graph + ggtitle("Anzahl der Restaurant Ratings Stufen",
                        subtitle = paste0("Anzahl der Datensätze: ", length(which(
                          !is.na(restaurant_data$Rating)
                        )))) +
  xlab("Rating Ebene") +
  ylab("Anzahl der Vorkommnisse") +
  theme_bw()
```

Anzahl der Restaurant Ratings Stufen

Anzahl der Datensätze: 115849



Anzahl der Restaurant Ratings pro Preisklasse

Hier zu sehen ist die Verteilung der Anzahl der Rating Stufen pro Preisklasse.

Es ist ersichtlich, dass Restaurants mit einer niedrigen Preisklasse tendenziell eher weniger hohe Bewertungen haben (im Vergleich zu den anderen Preisklassen) Restaurants in einer hohen Preisklasse hingegen besitzen höhere Ratings welche im Schnitt 4 oder höher sind.

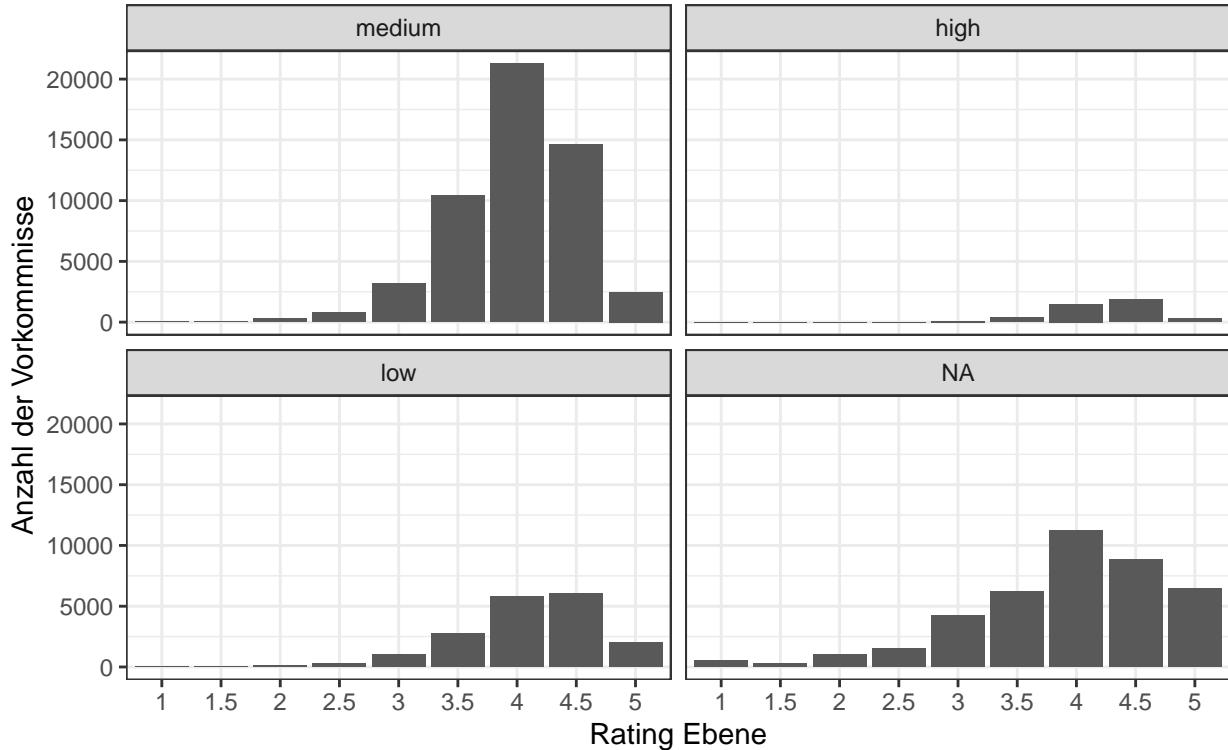
Ein weiteres interessantes Merkmal ist, dass Restaurants ohne Preisklasseneinstufung tendenziell viel mehr 5 Sterne Bewertungen haben. Dies kann entweder ein statistischer Zufall sein oder es könnte daran liegen, dass diese Restaurants nicht traditionell in Preisklassen eingeteilt werden können und dadurch Leute besser ansprechen (was aber eher weit hergeholt ist).

```
ratings_price_graph = restaurant_data %>%
  filter(!is.na(Rating)) %>%
  ggplot(aes(x = Rating)) + geom_bar() + facet_wrap(~Price_Range)

ratings_price_graph + ggtitle("Anzahl der Restaurant Ratings Stufen pro Preisklasse",
  subtitle = paste0("Anzahl der Datensätze: ", length(which(
    !is.na(restaurant_data$Rating)
  )))) +
  xlab("Rating Ebene") +
  ylab("Anzahl der Vorkommnisse") +
  theme_bw()
```

Anzahl der Restaurant Ratings Stufen pro Preisklasse

Anzahl der Datensätze: 115849



Review Anzahl

Hier zu sehen ist die Verteilung der Anzahl der Review Bewertungen.

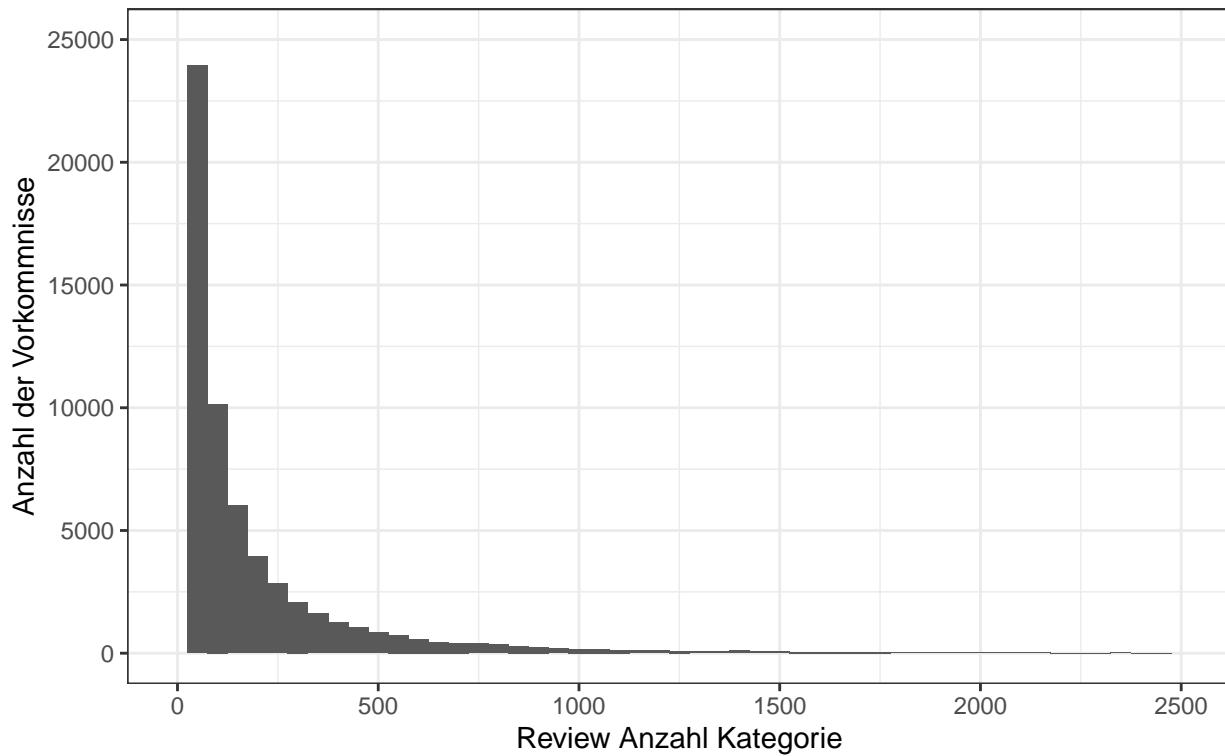
Dieser Graph gruppiert auf der x-Achse die Anzahl der Reviews in 50er Blöcke und stellt davon die Anzahl der Vorkommnisse auf der y-Achse dar.

Hier kann man stark betrachten, dass die Verteilung stark rechts fallend ist und die meisten Vorkommnisse in die Kategorie von 0-50 Reviews fallen. Dies bedeutet, dass die meisten Restaurants wenig Reviews haben und das nur noch ein sehr kleiner Anteil an Restaurants mehr als 500 Reviews haben.

```
reviewCount_graph = restaurant_data %>% filter(!is.na(Review_Count)) %>% ggplot(aes(x = Review_Count)) +  
  reviewCount_graph + ggtitle("Verteilung der Review Anzahl",  
    subtitle = paste0("Anzahl der Datensätze: ", length(which(  
      !is.na(restaurant_data$Review_Count)  
    )), " | Binwidth = 50")) +  
  xlab("Review Anzahl Kategorie") +  
  ylab("Anzahl der Vorkommnisse") +  
  theme_bw() +  
  xlim(0, 2500) + ylim(0, 25000)  
  
## Warning: Removed 250 rows containing non-finite values (stat_bin).  
  
## Warning: Removed 2 rows containing missing values (geom_bar).
```

Verteilung der Review Anzahl

Anzahl der Datensätze: 108176 | Binwidth = 50



Restaurant Ranking erklärt durch Review Anzahl

Hier zu sehen ist die Verteilung des Rankings von Restaurants erklärt durch die Review Anzahl.

Es ist eindeutig ersichtlich, dass Restaurants welche ein hohes Ranking in ihrer Stadt haben, eine dementsprechend höhere Anzahl an Reviews hat. Dies kann sich dadurch erklären lassen, dass Top-Restaurants öfters besucht werden und dadurch mehr Leute auch eine Bewertung schreiben.

```
ranking_reviewCount_graph = restaurant_data %>% filter(!is.na(Review_Count)) %>% filter(!is.na(Ranking))
  geom_point() + geom_smooth(method = "gam")

ranking_reviewCount_graph + ggtitle("Restaurant Ranking erklärt durch Review Anzahl",
                                     subtitle = paste0("Anzahl der Datensätze: ", length(which(
                                       !is.na(restaurant_data$Ranking)
                                       )))) +
  xlab("Review Anzahl") +
  ylab("Ranking des Restaurants") +
  theme_bw() +
  scale_y_continuous(trans = "reverse", limits = c(17000, 1))

## `geom_smooth()` using formula 'y ~ s(x, bs = "cs")'
```

Restaurant Ranking erklärt durch Review Anzahl

Anzahl der Datensätze: 115868

