R Notebook

```
library(readr)
zurich <- read_csv("Desktop/AD 699/zurich_listings_699.csv")
```

```
## Rows: 2534 Columns: 75
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (30): listing_url, last_scraped, source, name, description, neighborhood...
## dbl (37): id, scrape_id, host_id, host_listings_count, host_total_listings_c...
## lgl  (8): host_is_superhost, host_has_profile_pic, host_identity_verified, b...
##
## 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.
```

```
colSums(is.na(zurich))
```

```
##                                           id
##                                            0
##                                  listing_url
##                                            0
##                                    scrape_id
##                                            0
##                                 last_scraped
##                                            0
##                                       source
##                                            0
##                                         name
##                                            0
##                                  description
##                                           45
##                        neighborhood_overview
##                                         1334
##                                  picture_url
##                                            0
##                                      host_id
##                                            0
##                                     host_url
##                                            0
##                                    host_name
##                                            0
##                                   host_since
##                                            0
##                                host_location
##                                          442
##                                   host_about
##                                          992
##                           host_response_time
##                                            0
##                           host_response_rate
##                                            0
##                         host_acceptance_rate
##                                            0
##                            host_is_superhost
##                                           42
##                           host_thumbnail_url
##                                            0
##                             host_picture_url
##                                            0
##                           host_neighbourhood
##                                         2499
##                          host_listings_count
##                                            0
##                    host_total_listings_count
##                                            0
##                           host_verifications
##                                            0
##                         host_has_profile_pic
##                                            0
##                       host_identity_verified
##                                            0
##                                neighbourhood
##                                         1334
##                       neighbourhood_cleansed
##                                            0
##                 neighbourhood_group_cleansed
##                                            0
##                                     latitude
##                                            0
##                                    longitude
##                                            0
##                                property_type
##                                            0
##                                    room_type
##                                            0
##                                 accommodates
##                                            0
##                                    bathrooms
##                                         2534
##                               bathrooms_text
##                                            2
##                                     bedrooms
##                                          795
##                                         beds
##                                           44
##                                    amenities
##                                             0
##                                         price
##                                             0
##                                minimum_nights
##                                             0
##                                maximum_nights
##                                             0
##                        minimum_minimum_nights
##                                             0
##                        maximum_minimum_nights
##                                             0
##                        minimum_maximum_nights
##                                             0
##                        maximum_maximum_nights
##                                             0
##                        minimum_nights_avg_ntm
##                                             0
##                        maximum_nights_avg_ntm
##                                             0
##                              calendar_updated
##                                          2534
##                              has_availability
##                                             0
##                               availability_30
##                                             0
##                               availability_60
##                                             0
##                               availability_90
##                                             0
##                              availability_365
##                                             0
##                         calendar_last_scraped
##                                             0
##                             number_of_reviews
##                                             0
##                         number_of_reviews_ltm
##                                             0
##                        number_of_reviews_l30d
##                                             0
##                                  first_review
##                                           553
##                                   last_review
##                                           553
##                          review_scores_rating
##                                           553
##                        review_scores_accuracy
##                                           561
##                     review_scores_cleanliness
##                                           561
##                         review_scores_checkin
##                                          561
##                  review_scores_communication
##                                          561
##                       review_scores_location
##                                          561
##                          review_scores_value
##                                          561
##                                      license
##                                         2534
##                             instant_bookable
##                                            0
##               calculated_host_listings_count
##                                            0
##  calculated_host_listings_count_entire_homes
##                                            0
## calculated_host_listings_count_private_rooms
##                                            0
##  calculated_host_listings_count_shared_rooms
##                                            0
##                            reviews_per_month
##                                          553
```

```
#Prepare the data for my clustering
data <- zurich[, c("price","number_of_reviews", "bedrooms", "review_scores_rating", "
latitude", "longitude" )]
str(data)
```

```
## tibble [2,534 × 6] (S3: tbl_df/tbl/data.frame)
##  $ price               : chr [1:2534] "$100.00" "$60.00" "$200.00" "$79.00" ...
##  $ number_of_reviews   : num [1:2534] 49 9 0 235 31 1 4 336 11 13 ...
##  $ bedrooms            : num [1:2534] 1 NA NA NA 2 1 1 2 3 3 ...
##  $ review_scores_rating: num [1:2534] 4.78 4.89 NA 4.6 4.97 5 5 4.48 4.6 4.92 ...
##  $ latitude            : num [1:2534] 47.4 47.4 47.4 47.3 47.4 ...
##  $ longitude           : num [1:2534] 8.52 8.53 8.48 8.54 8.52 ...
```

```
data$price <- as.numeric(gsub("[\\$,]", "", zurich$price))
```

```
colSums(is.na(data))
```

```
##                price    number_of_reviews             bedrooms
##                    0                    0                  795
## review_scores_rating             latitude            longitude
##                  553                    0                    0
```

```
get_mode <- function(v) {
   uniqv <- unique(v)
   uniqv[which.max(tabulate(match(v, uniqv)))]
}

mode_bedrooms <- get_mode(data$bedrooms[!is.na(data$bedrooms)])
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
#clean variable "bedroom"
data$bedrooms[is.na(data$bedrooms)] <- mode_bedrooms

lower_limit <- 0
upper_limit <- 5

# Filter out outliers
data <- data %>%
  filter(bedrooms >= lower_limit & bedrooms <= upper_limit)
summary(data$bedrooms)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   1.000   1.265   1.000   5.000
```

```
#clean "review score rating"
median_rating <- median(data$review_scores_rating, na.rm = TRUE)

# Impute missing values with the median
data$review_scores_rating[is.na(data$review_scores_rating)] <- median_rating
summary(data$review_scores_rating)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   4.730   4.860   4.764   5.000   5.000
```

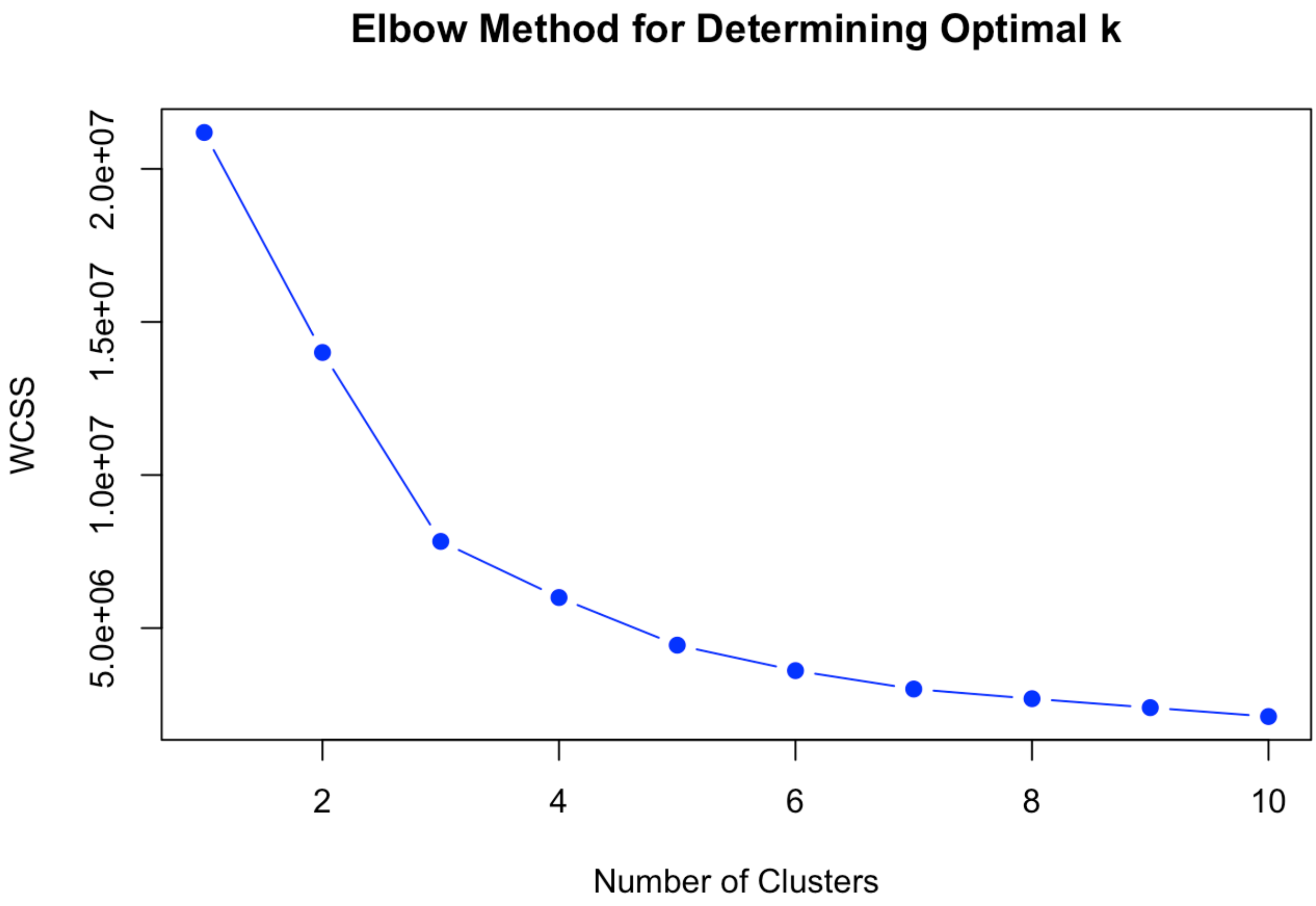
```
colSums(is.na(data))
```

```
##                price    number_of_reviews             bedrooms
##                    0                    0                    0
## review_scores_rating             latitude            longitude
##                    0                    0                    0
```

```
#clean "price"
IQR <- IQR(data$price)
upper_bound <- quantile(data$price, 0.75) + (1.5 * IQR)
lower_bound <- quantile(data$price, 0.25) - (1.5 * IQR)
data <- subset(data, data$price <= upper_bound & data$price >= lower_bound)
```

```
data_scaled <- scale(data)
View(data_scaled)
```

```
wcss <- sapply(1:10, function(k) {
  kmeans(data, centers = k, nstart = 20)$tot.withinss
})
plot(1:10, wcss, type = "b", pch = 19, col = "blue", xlab = "Number of Clusters", yla
b = "WCSS",
     main = "Elbow Method for Determining Optimal k")
```



```
#choose k=3 based on the Elbow chart
set.seed(123)
k <- 3
clusters <- kmeans(data_scaled, centers = k)
```

```
clusters$centers
```

```
##           price number_of_reviews   bedrooms review_scores_rating     latitude
## 1  1.154149589         0.1015333  1.9252636           0.02708672 -0.003293961
## 2 -0.005299418         0.1447875 -0.3942995          -0.13090767 -0.704010420
## 3 -0.399361870        -0.1450366 -0.3745334           0.08958236  0.533828829
##      longitude
## 1 -0.02469441
## 2  0.68495511
## 3 -0.50962910
```

For the clustering analysis of Airbnb properties in Zurich, I employed the K-means clustering algorithm to categorize Airbnb properties in Zurich, categorize the listings based on four key attributes: price, number of bedrooms, review scores, and the number of reviews. This approach aims to segment the properties into distinct groups, each representing a different type of accommodation based on their pricing, size, guest satisfaction, and popularity.

This segmentation provides valuable insights into the diverse offerings within the Zurich Airbnb market, enabling potential guests to make informed decisions that align with their preferences and budget.

Cluster 1: “Premium and Popular” - This cluster might represent more expensive listings due to the high price among all the clusters, with a larger number of bedrooms, which could suggest more spacious or luxury accommodations. They also have a relatively higher number of reviews, which might indicate popularity.

Cluster 2: “Economical and Cozy” - This cluster could represent more budget-friendly and smaller listings. The review_scores_rating is slightly below the mean, suggesting average satisfaction.

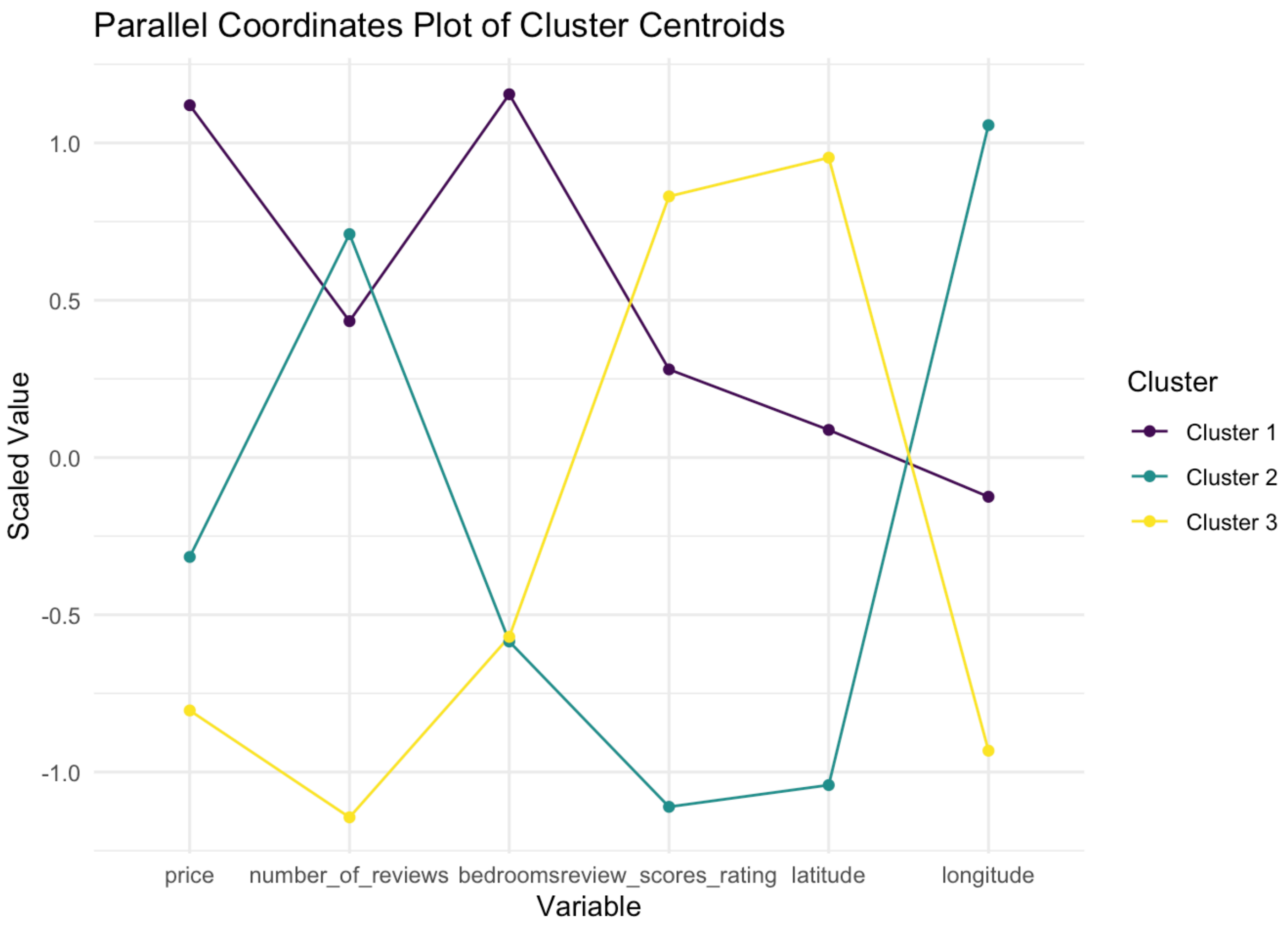
Cluster 3: “Moderate and Satisfactory” - This cluster has very negative values for price, indicating lower prices, and an average number of bedrooms. The review_scores_rating is above the mean, indicating that these listings are well-reviewed, possibly offering a balance between quality and cost.

```
library(GGally)
```

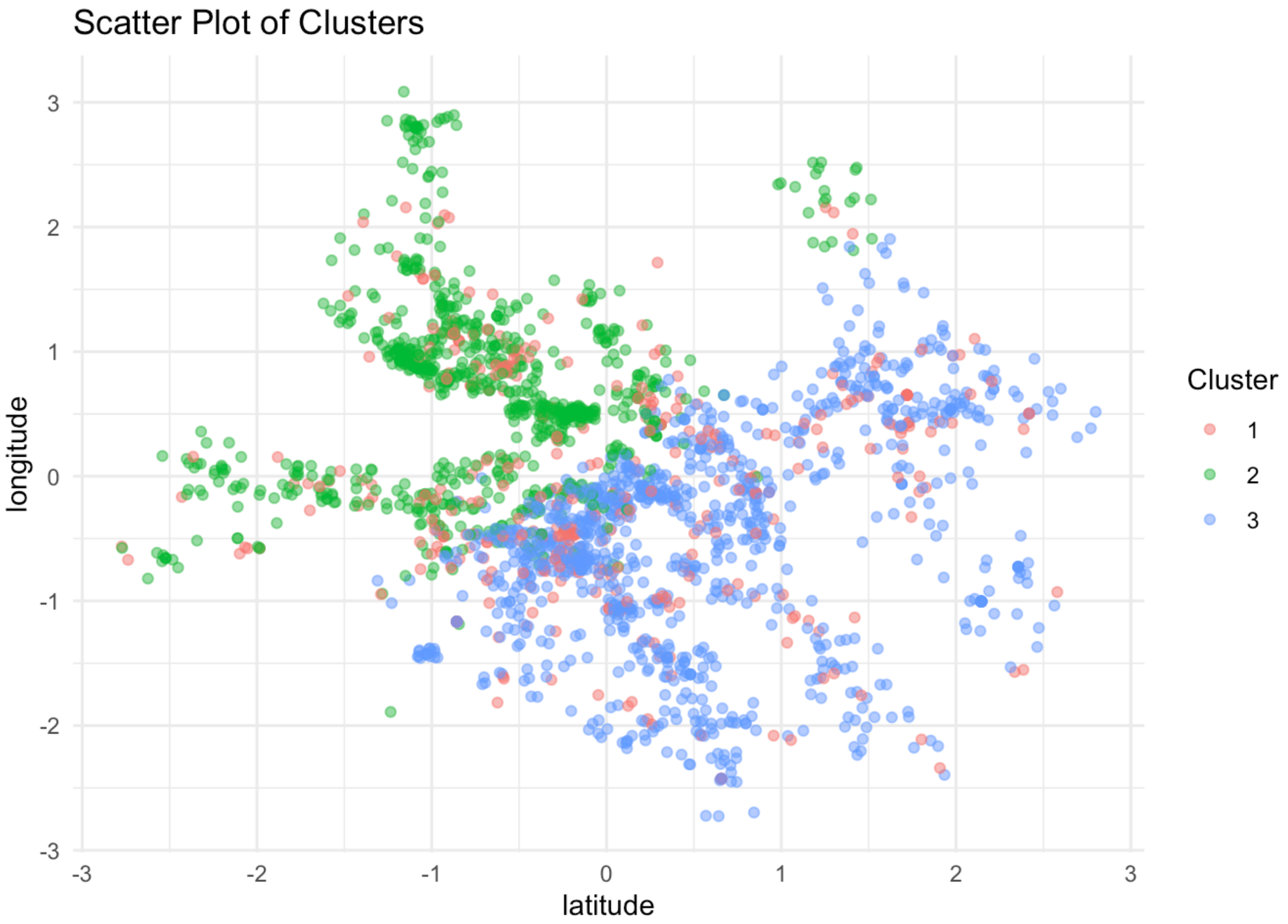
```
## Loading required package: ggplot2
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

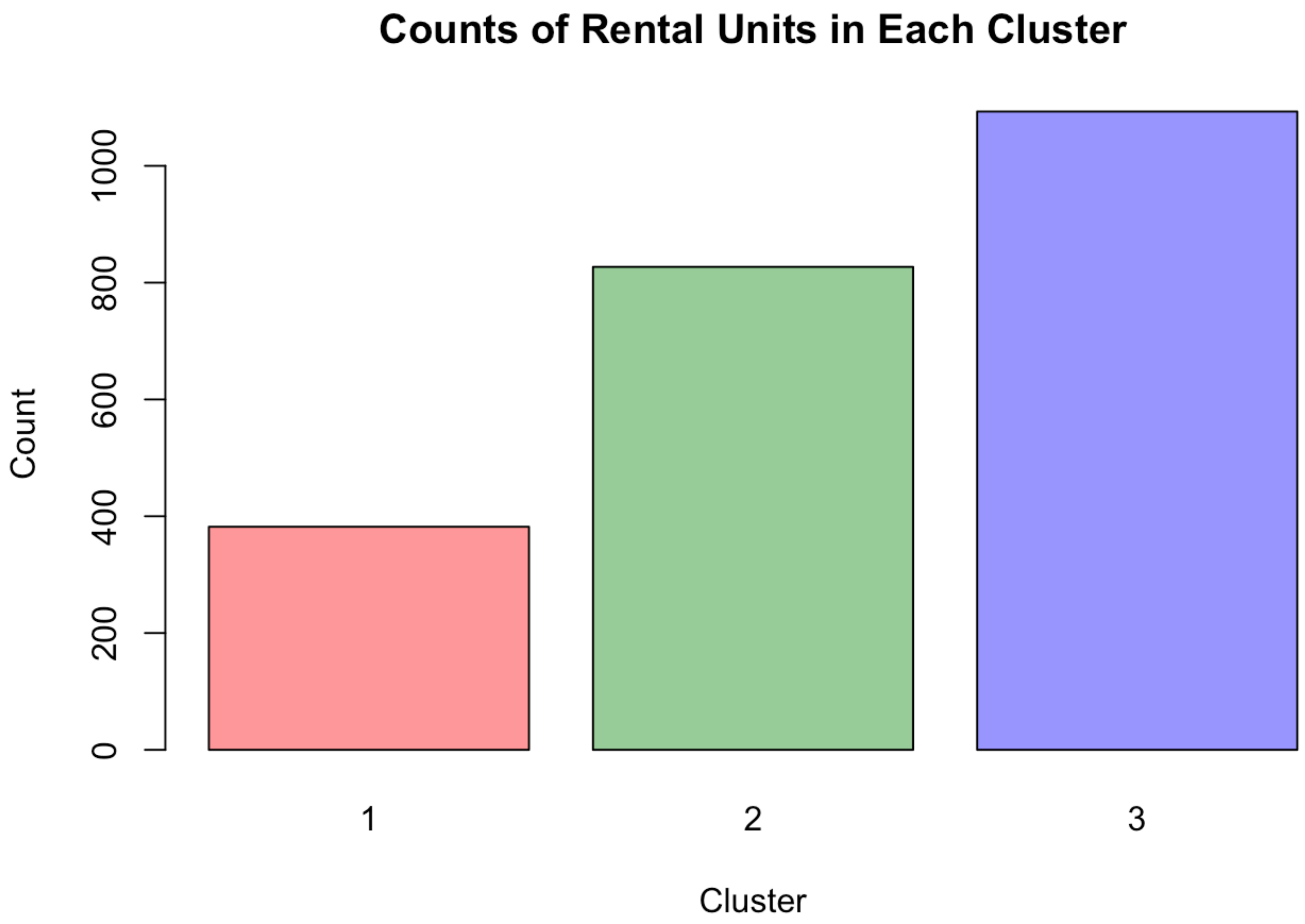
```
library(ggdendro)
centroids <- data.frame(clusters$centers)
centroids['Cluster'] = paste('Cluster', seq(1, nrow(centroids)))
ggparcoord(centroids, columns=1:6, groupColumn='Cluster', showPoints=TRUE) +
  scale_color_viridis_d() +
  labs(x='Variable', y='Scaled Value') +
  theme_minimal() +
  ggtitle("Parallel Coordinates Plot of Cluster Centroids")
```



```
library(ggplot2)
ggplot(data_scaled, aes(x = latitude, y = longitude, color = as.factor(clusters$clust
er))) +
  geom_point(alpha = 0.5) +
  labs(color = 'Cluster') +
  ggtitle('Scatter Plot of Clusters') +
  theme_minimal()
```

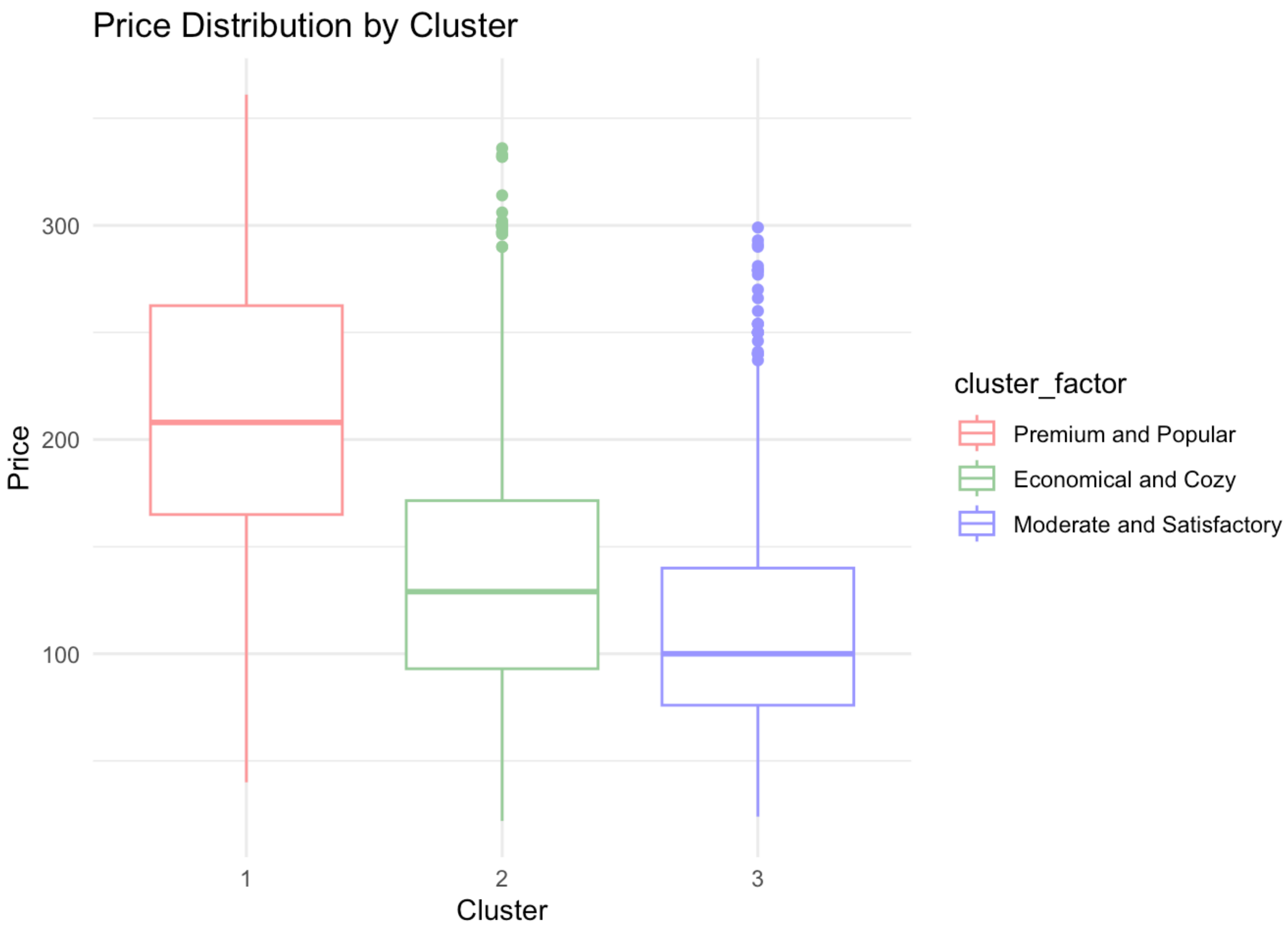


```
cluster_counts <- table(factor(clusters$cluster))
barplot(cluster_counts,
        main="Counts of Rental Units in Each Cluster",
        xlab="Cluster",
        ylab="Count",
        col=c("#FF9999", "#99CC99", "#9999FF"))
```



Among all the clusters, cluster 3 (Moderate and Satisfactory) has the largest number, which means majority people choose airbnb in Zurich more focus on the moderate price, well-reviewed place to stay; cluster 1 has the fewest number due to higher price and more bedrooms, suggesting that these higher-priced, larger accommodations are less common or in lesser demand compared to other types of listings. While Cluster 2 is in the middle, indicating a good availability of budget-friendly accommodations. This popularity can be linked to travelers prioritizing affordability without sacrificing comfort, appealing especially to those who plan extended stays or are budget-conscious.

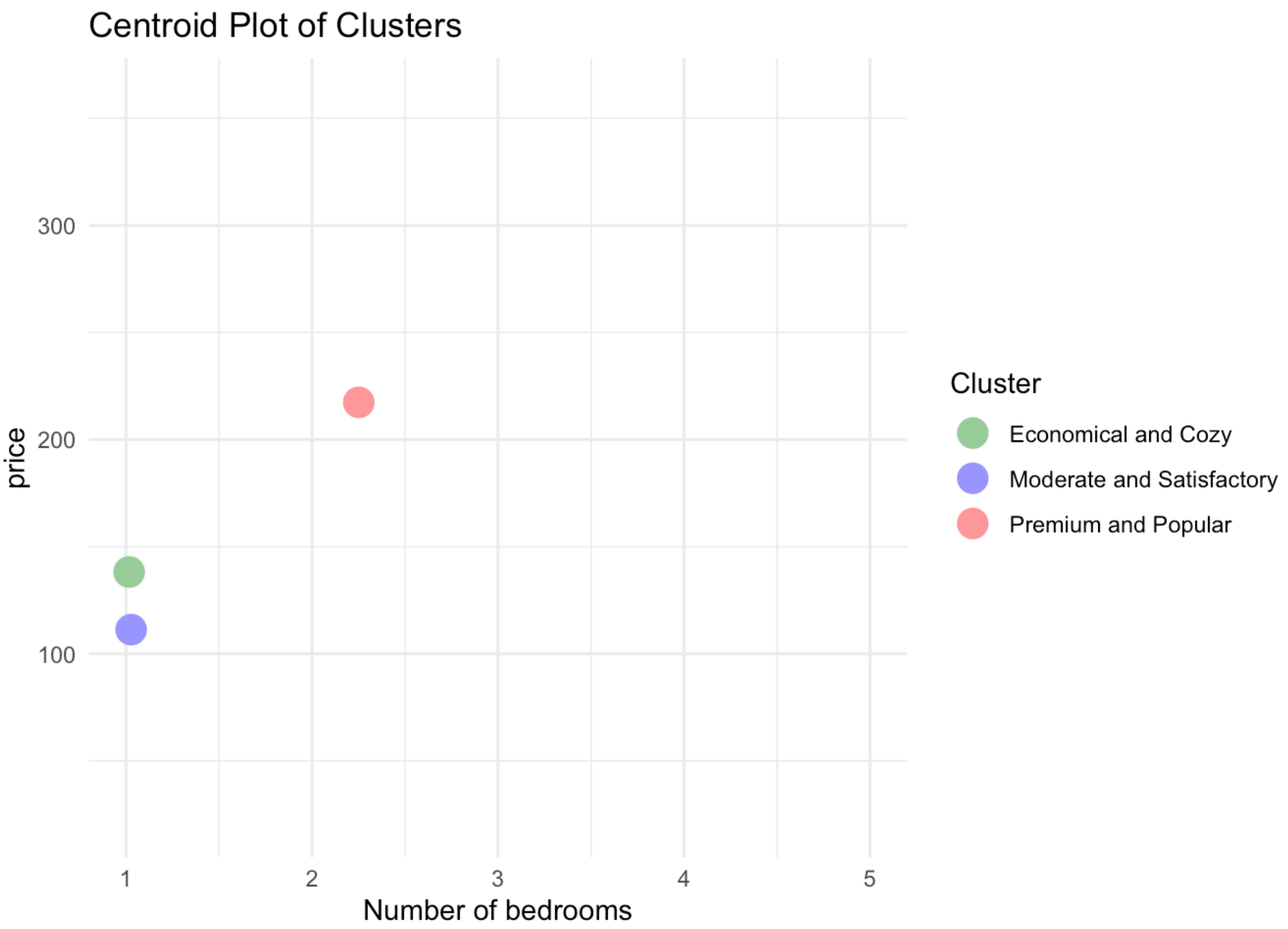
```
library(ggplot2)
data$cluster_factor <- factor(clusters$cluster, labels = c('Premium and Popular', 'Economical and Cozy', 'Moderate and Satisfactory'))

ggplot(data, aes(x = cluster_factor, y = price, color = cluster_factor)) +
  geom_boxplot() +
  labs(title = "Price Distribution by Cluster", x = "Cluster", y = "Price") +
  scale_color_manual(values = c('Premium and Popular' = '#FF9999', 'Economical and Cozy' = '#99CC99', 'Moderate and Satisfactory' = '#9999FF')) +
  scale_x_discrete(breaks = c('Premium and Popular', 'Economical and Cozy', 'Moderate and Satisfactory'), labels = c('1', '2', '3')) +
  theme_minimal()
```



Cluster is characterized by a higher median price range and a broader spread in prices, including several higher outliers, suggesting that these listings are generally more expensive, likely due to superior amenities or locations; Cluster 2 Displays a lower and tighter price range with a slightly higher median than Cluster 3, this cluster represents more budget-friendly options that cater to travelers looking for affordable accommodations; Cluster 3 features a moderate price range with a substantial number of outliers on the higher end, indicating a mix of moderately priced listings generally viewed as offering a balance between cost and comfort.

```
#centroid plot of cluster
centroids <- aggregate(cbind(bedrooms, price) ~ clusters$cluster, data, mean)
cluster_labels <- c('Premium and Popular', 'Economical and Cozy', 'Moderate and Satis
factory')
centroids$cluster_name <- cluster_labels[centroids$cluster]
```

```
ggplot(data, aes(x = bedrooms, y = price)) +
  geom_point(alpha = 0) +
  geom_point(data = centroids, aes(x = bedrooms, y = price, color = cluster_name), si
ze = 5) +
  scale_color_manual(values = c('Premium and Popular' = '#FF9999', 'Economical and Co
zy' = '#99CC99', 'Moderate and Satisfactory' = '#9999FF')) +
  labs(title = "Centroid Plot of Clusters",
       x = "Number of bedrooms",
       y = "price",
       color = "Cluster") +
  theme_minimal()
```



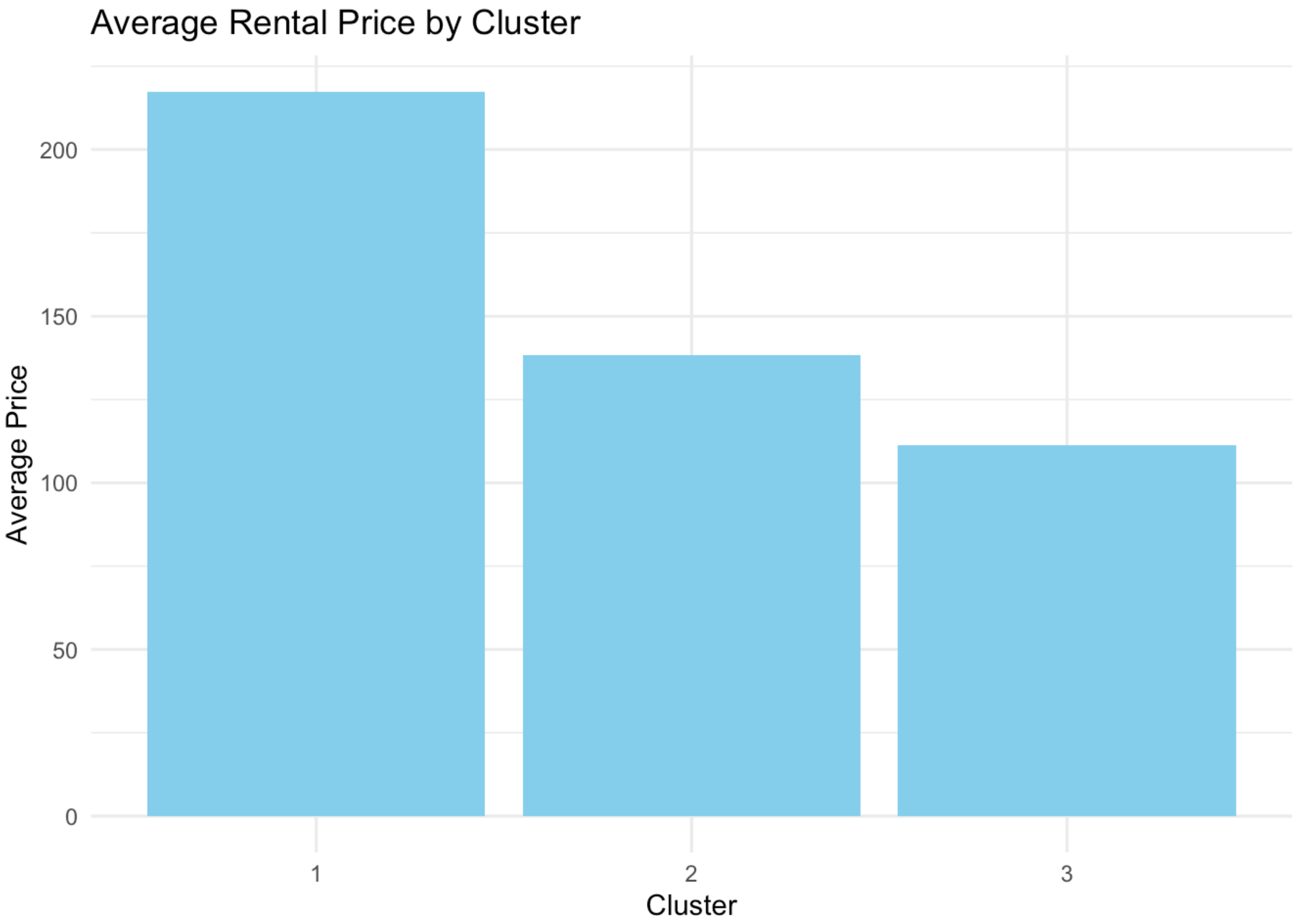
The positioning of the centroids clearly delineates the distinct offerings in the Zurich Airbnb market: from luxury and spacious accommodations to modest and budget-friendly options, catering to diverse traveler needs and preferences.

Cluster 1 Represented by the red dot, this cluster’s centroid is positioned at the highest price point, suggesting these listings are the most expensive. It’s also placed at a higher number of bedrooms, indicating that these listings typically offer more space, which could be a factor in their premium pricing; Cluster 2 Shown in green, this cluster’s centroid is at the lowest price point and near the lower end of the bedroom scale. This suggests that these listings are the most budget-friendly and generally have fewer bedrooms, which aligns with their description as economical and cozy. Cluster 3 represents purple, which is positioned moderately in terms of both price and bedrooms. This placement indicates that listings in this cluster offer a balance of affordability and comfort, providing a satisfactory option for a broad range of travelers.

```
data$cluster <- clusters$cluster
library(dplyr)
cluster_summary <- data %>%
  group_by(cluster) %>%
  summarize(
    avg_price = mean(price, na.rm = TRUE),
    avg_review_scores_rating = mean(review_scores_rating, na.rm = TRUE),
    bedrooms = mean(bedrooms, na.rm = TRUE)
  )

# Order clusters by average price
cluster_summary <- cluster_summary %>% arrange(desc(avg_price))

# See the summary
print(cluster_summary)
```

```
## # A tibble: 3 × 4
##   cluster avg_price avg_review_scores_rating bedrooms
##     <int>     <dbl>                    <dbl>    <dbl>
## 1       1      217.                     4.78     2.25
## 2       2      138.                     4.71     1.02
## 3       3      111.                     4.80     1.03
```

```
ggplot(cluster_summary, aes(x = factor(cluster), y = avg_price)) +
  geom_bar(stat = "identity", fill = 'skyblue') +
  theme_minimal() +
  labs(x = 'Cluster', y = 'Average Price', title = 'Average Rental Price by Cluster')
```


Average Rental Price by Cluster
Average Price
0
50
100
150
200
1
2
3
Cluster