part1

Akarsh Gaonkar

2025-01-04

```r
# Importing Libraries
library(readr)
library(ggplot2)
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
```

```r
library(tidyr)
library(maps)
```

```
## Warning: package 'maps' was built under R version 4.3.3
```

```r
library(mapdata)
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.3.3
```

```
## corrplot 0.95 loaded
```

```r
## Task 1: DATA EXPLORATION AND PREPROCESSING


# Load Dataset
data <- read.csv("~/Desktop/data_analysis_restaurant_data/Dataset.csv")

# View top 10 rows of the dataset
head(data, 10)
```

```
##    Restaurant.ID                          Restaurant.Name Country.Code
## 1        6317637                         Le Petit Souffle          162
## 2        6304287                         Izakaya Kikufuji          162
## 3        6300002                   Heat - Edsa Shangri-La          162
## 4        6318506                                     Ooma          162
## 5        6314302                              Sambo Kojin          162
## 6       18189371                             Din Tai Fung          162
## 7        6300781                               Buffet 101          162
## 8        6301290                                  Vikings          162
## 9        6300010 Spiral - Sofitel Philippine Plaza Manila          162
## 10       6314987                                 Locavore          162
##                City
## 1       Makati City
## 2       Makati City
## 3  Mandaluyong City
## 4  Mandaluyong City
## 5  Mandaluyong City
## 6  Mandaluyong City
## 7        Pasay City
## 8        Pasay City
## 9        Pasay City
## 10       Pasig City
##                                                                                 Address
## 1              Third Floor, Century City Mall, Kalayaan Avenue, Poblacion, Makati City
## 2                  Little Tokyo, 2277 Chino Roces Avenue, Legaspi Village, Makati City
## 3                             Edsa Shangri-La, 1 Garden Way, Ortigas, Mandaluyong City
## 4               Third Floor, Mega Fashion Hall, SM Megamall, Ortigas, Mandaluyong City
## 5                     Third Floor, Mega Atrium, SM Megamall, Ortigas, Mandaluyong City
## 6              Ground Floor, Mega Fashion Hall, SM Megamall, Ortigas, Mandaluyong City
## 7  Building K, SM By The Bay, Sunset Boulevard, Mall of Asia Complex (MOA), Pasay City
## 8    Building B, By The Bay, Seaside Boulevard, Mall of Asia Complex (MOA), Pasay City
## 9                Plaza Level, Sofitel Philippine Plaza Manila, CCP Complex, Pasay City
## 10                 Brixton Technology Center, 10 Brixton Street, Kapitolyo, Pasig City
##                                           Locality
## 1        Century City Mall, Poblacion, Makati City
## 2       Little Tokyo, Legaspi Village, Makati City
## 3       Edsa Shangri-La, Ortigas, Mandaluyong City
## 4           SM Megamall, Ortigas, Mandaluyong City
## 5           SM Megamall, Ortigas, Mandaluyong City
## 6           SM Megamall, Ortigas, Mandaluyong City
## 7  SM by the Bay, Mall of Asia Complex, Pasay City
## 8  SM by the Bay, Mall of Asia Complex, Pasay City
## 9      Sofitel Philippine Plaza Manila, Pasay City
## 10                                       Kapitolyo
##                                                Locality.Verbose Longitude
## 1        Century City Mall, Poblacion, Makati City, Makati City  121.0275
## 2       Little Tokyo, Legaspi Village, Makati City, Makati City  121.0141
## 3  Edsa Shangri-La, Ortigas, Mandaluyong City, Mandaluyong City  121.0568
## 4      SM Megamall, Ortigas, Mandaluyong City, Mandaluyong City  121.0565
## 5      SM Megamall, Ortigas, Mandaluyong City, Mandaluyong City  121.0575
## 6      SM Megamall, Ortigas, Mandaluyong City, Mandaluyong City  121.0563
## 7   SM by the Bay, Mall of Asia Complex, Pasay City, Pasay City  120.9797
## 8   SM by the Bay, Mall of Asia Complex, Pasay City, Pasay City  120.9793
## 9       Sofitel Philippine Plaza Manila, Pasay City, Pasay City  120.9801
## 10                                        Kapitolyo, Pasig City  121.0565
##    Latitude                           Cuisines Average.Cost.for.two
## 1  14.56544         French, Japanese, Desserts                 1100
## 2  14.55371                           Japanese                 1200
## 3  14.58140   Seafood, Asian, Filipino, Indian                 4000
## 4  14.58532                    Japanese, Sushi                 1500
## 5  14.58445                   Japanese, Korean                 1500
## 6  14.58376                            Chinese                 1000
## 7  14.53133                    Asian, European                 2000
## 8  14.54000 Seafood, Filipino, Asian, European                 2000
## 9  14.55299            European, Asian, Indian                 6000
## 10 14.57204                           Filipino                 1100
##            Currency Has.Table.booking Has.Online.delivery Is.delivering.now
## 1  Botswana Pula(P)               Yes                  No                No
## 2  Botswana Pula(P)               Yes                  No                No
## 3  Botswana Pula(P)               Yes                  No                No
## 4  Botswana Pula(P)                No                  No                No
## 5  Botswana Pula(P)               Yes                  No                No
## 6  Botswana Pula(P)                No                  No                No
## 7  Botswana Pula(P)               Yes                  No                No
## 8  Botswana Pula(P)               Yes                  No                No
## 9  Botswana Pula(P)               Yes                  No                No
## 10 Botswana Pula(P)               Yes                  No                No
##    Switch.to.order.menu Price.range Aggregate.rating Rating.color Rating.text
## 1                    No           3              4.8   Dark Green   Excellent
## 2                    No           3              4.5   Dark Green   Excellent
## 3                    No           4              4.4        Green   Very Good
## 4                    No           4              4.9   Dark Green   Excellent
## 5                    No           4              4.8   Dark Green   Excellent
## 6                    No           3              4.4        Green   Very Good
## 7                    No           4              4.0        Green   Very Good
## 8                    No           4              4.2        Green   Very Good
## 9                    No           4              4.9   Dark Green   Excellent
## 10                   No           3              4.8   Dark Green   Excellent
##    Votes
## 1    314
## 2    591
## 3    270
## 4    365
## 5    229
## 6    336
## 7    520
## 8    677
## 9    621
## 10   532
```

```r
# Explore the dataset and identify the number of rows and columns

# Checking number of rows and columns of the dataset
cat("Number of rows:", nrow(data), "\n")
```

```
## Number of rows: 9551
```

```r
cat("Number of columns:", ncol(data), "\n")
```

```
## Number of columns: 21
```

```r
# Dataset Duplicate Value Count
dup <- sum(duplicated(data))
cat("Number of duplicate rows:", dup)
```

```
## Number of duplicate rows: 0
```

```r
# Check for missing values in each column and handle them accordingly

# Check for missing values
missing_values <- sum(is.na(data))

# Check for empty values
empty_values <- sum(data == "")

cat("Missing values count:", missing_values, "\n")
```

```
## Missing values count: 0
```

```r
cat("Empty values count:", empty_values, "\n")
```

```
## Empty values count: 9
```

```r
# There are 9 empty values, let's find out which column/columns has it
empty_values_count <- colSums(data == "")
cat("Empty Values Count:\n")
```

```
## Empty Values Count:
```

```r
print(empty_values_count)
```

```
##        Restaurant.ID      Restaurant.Name         Country.Code
##                    0                    0                    0
##                 City              Address             Locality
##                    0                    0                    0
##      Locality.Verbose            Longitude             Latitude
##                    0                    0                    0
##             Cuisines Average.Cost.for.two             Currency
##                    9                    0                    0
##    Has.Table.booking  Has.Online.delivery     Is.delivering.now
##                    0                    0                    0
## Switch.to.order.menu          Price.range     Aggregate.rating
##                    0                    0                    0
##         Rating.color          Rating.text                Votes
##                    0                    0                    0
```

```
# The Cuisines column has 9 empty values. Since it's not many, let's remove these rows
data <- data[!(data$Cuisines == ""), , drop = FALSE]


# Check for empty values after Removing
empty_values <- sum(data == "")
cat("Empty values count:", empty_values, "\n")
```

```
## Empty values count: 0
```

```
# Display basic information about the dataset to check various data types
str(data)
```

```
## ’data.frame':    9542 obs. of  21 variables:
##  $ Restaurant.ID       : int  6317637 6304287 6300002 6318506 6314302 18189371 6300781 6301290 63000
##  $ Restaurant.Name     : chr  "Le Petit Souffle" "Izakaya Kikufuji" "Heat - Edsa Shangri-La" "Ooma"
##  $ Country.Code        : int  162 162 162 162 162 162 162 162 162 162 ...
##  $ City                : chr  "Makati City" "Makati City" "Mandaluyong City" "Mandaluyong City" ...
##  $ Address             : chr  "Third Floor, Century City Mall, Kalayaan Avenue, Poblacion, Makati Ci
##  $ Locality            : chr  "Century City Mall, Poblacion, Makati City" "Little Tokyo, Legaspi Vil
##  $ Locality.Verbose    : chr  "Century City Mall, Poblacion, Makati City, Makati City" "Little Tokyo
##  $ Longitude           : num  121 121 121 121 121 ...
##  $ Latitude            : num  14.6 14.6 14.6 14.6 14.6 ...
##  $ Cuisines            : chr  "French, Japanese, Desserts" "Japanese" "Seafood, Asian, Filipino, Indi
##  $ Average.Cost.for.two: int  1100 1200 4000 1500 1500 1000 2000 2000 6000 1100 ...
##  $ Currency            : chr  "Botswana Pula(P)" "Botswana Pula(P)" "Botswana Pula(P)" "Botswana Pula
##  $ Has.Table.booking   : chr  "Yes" "Yes" "Yes" "No" ...
##  $ Has.Online.delivery : chr  "No" "No" "No" "No" ...
##  $ Is.delivering.now   : chr  "No" "No" "No" "No" ...
##  $ Switch.to.order.menu: chr  "No" "No" "No" "No" ...
##  $ Price.range         : int  3 3 4 4 4 3 4 4 4 3 ...
##  $ Aggregate.rating    : num  4.8 4.5 4.4 4.9 4.8 4.4 4 4.2 4.9 4.8 ...
##  $ Rating.color        : chr  "Dark Green" "Dark Green" "Green" "Dark Green" ...
##  $ Rating.text         : chr  "Excellent" "Excellent" "Very Good" "Excellent" ...
##  $ Votes               : int  314 591 270 365 229 336 520 677 621 532 ...
```

```
# Analyze the distribution of the target variable ("Aggregate rating") and identify any class imbalance

# Distribution of the target variable ("Aggregate rating")
target_counts <- table(data$'Aggregate rating')

# Print the distribution
print("Distribution of target variable:")
```

```
## [1] "Distribution of target variable:"
```

```
print(target_counts)
```

```
## < table of extent 0 >
```

```r
# Check if the distribution is balanced
is_balanced <- all(target_counts >= mean(target_counts))
if (is_balanced) {
  print("The distribution of the target variable is balanced.")
} else {
  print("The distribution of the target variable is imbalanced.")
}
```

```
## [1] "The distribution of the target variable is balanced."
```

```r
## Task 2: DESCRIPTIVE ANALYSIS


# Basic statistical measures (mean, median, standard deviation, etc.) for numerical columns

# Select Numerical Columns
numeric_columns <- data[, sapply(data, is.numeric)]

# Calculate basic statistical measures using summary()
summary_stats <- summary(numeric_columns)
print(summary_stats)
```

```
##  Restaurant.ID       Country.Code      Longitude          Latitude
##  Min.   :      53   Min.   :  1.00   Min.   :-157.95   Min.   :-41.33
##  1st Qu.:  301931   1st Qu.:  1.00   1st Qu.:  77.08   1st Qu.: 28.48
##  Median : 6002726   Median :  1.00   Median :  77.19   Median : 28.57
##  Mean   : 9043301   Mean   : 18.18   Mean   :  64.28   Mean   : 25.85
##  3rd Qu.:18352604   3rd Qu.:  1.00   3rd Qu.:  77.28   3rd Qu.: 28.64
##  Max.   :18500652   Max.   :216.00   Max.   : 174.83   Max.   : 55.98
##  Average.Cost.for.two  Price.range    Aggregate.rating     Votes
##  Min.   :     0       Min.   :1.000   Min.   :0.000   Min.   :    0.0
##  1st Qu.:   250       1st Qu.:1.000   1st Qu.:2.500   1st Qu.:    5.0
##  Median :   400       Median :2.000   Median :3.200   Median :   31.0
##  Mean   :  1200       Mean   :1.805   Mean   :2.665   Mean   :  156.8
##  3rd Qu.:   700       3rd Qu.:2.000   3rd Qu.:3.700   3rd Qu.:  130.0
##  Max.   :800000       Max.   :4.000   Max.   :4.900   Max.   :10934.0
```

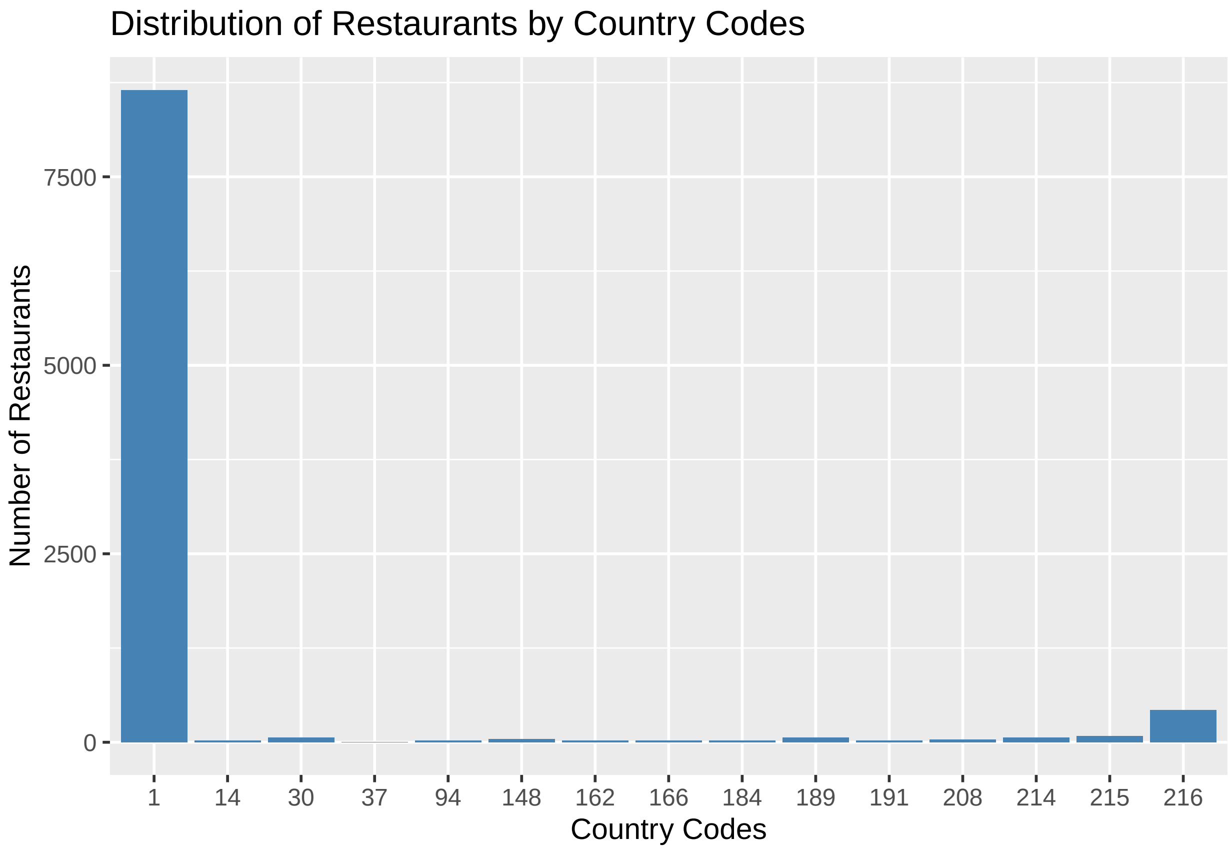
```r
# Calculate standard deviation for numerical columns
sds <- sapply(data[, sapply(data, is.numeric)], sd, na.rm = TRUE)


print("Standard deviation for numerical columns:")
```

```
## [1] "Standard deviation for numerical columns:"
```

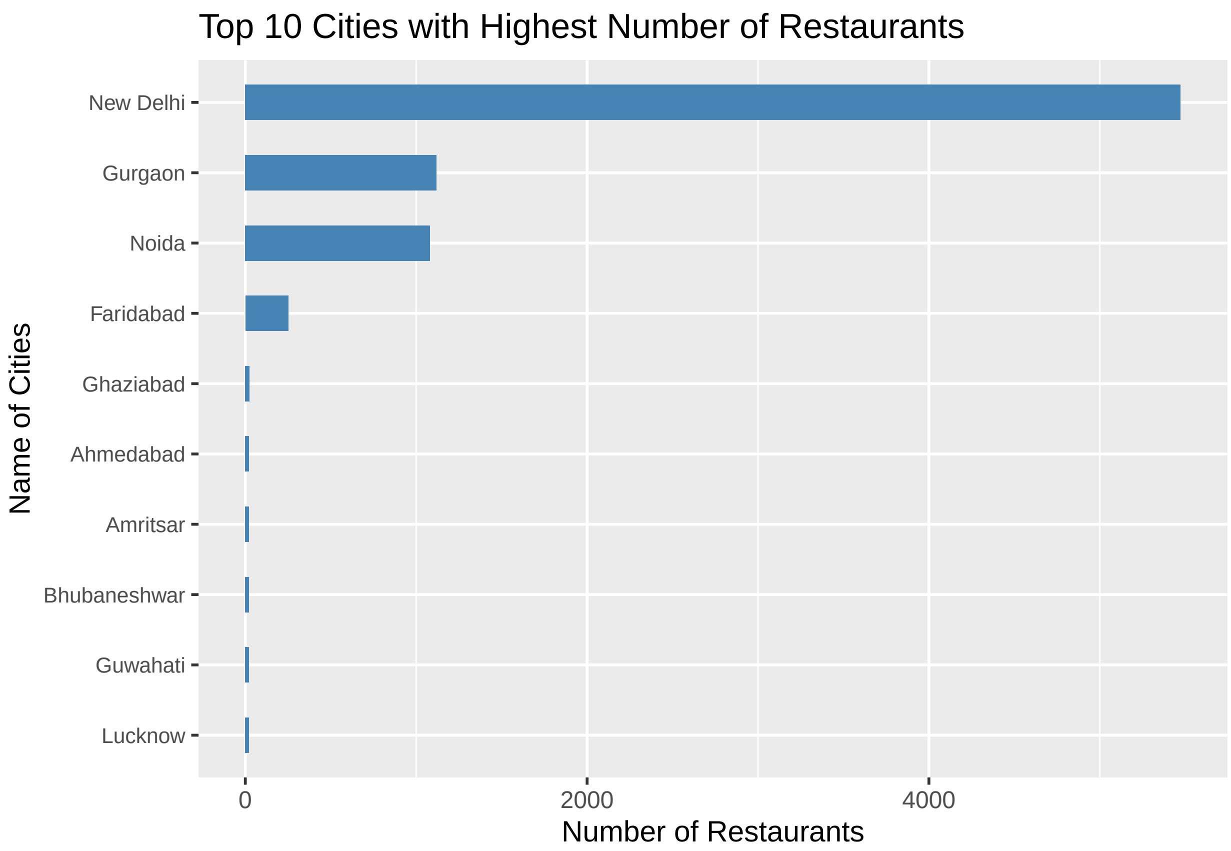
```r
print(sds)
```

```
##        Restaurant.ID         Country.Code            Longitude
##         8.791967e+06         5.645160e+01         4.119760e+01
##             Latitude Average.Cost.for.two          Price.range
##         1.101009e+01         1.612874e+04         9.055631e-01
##     Aggregate.rating                Votes
##         1.516588e+00         4.302033e+02
```

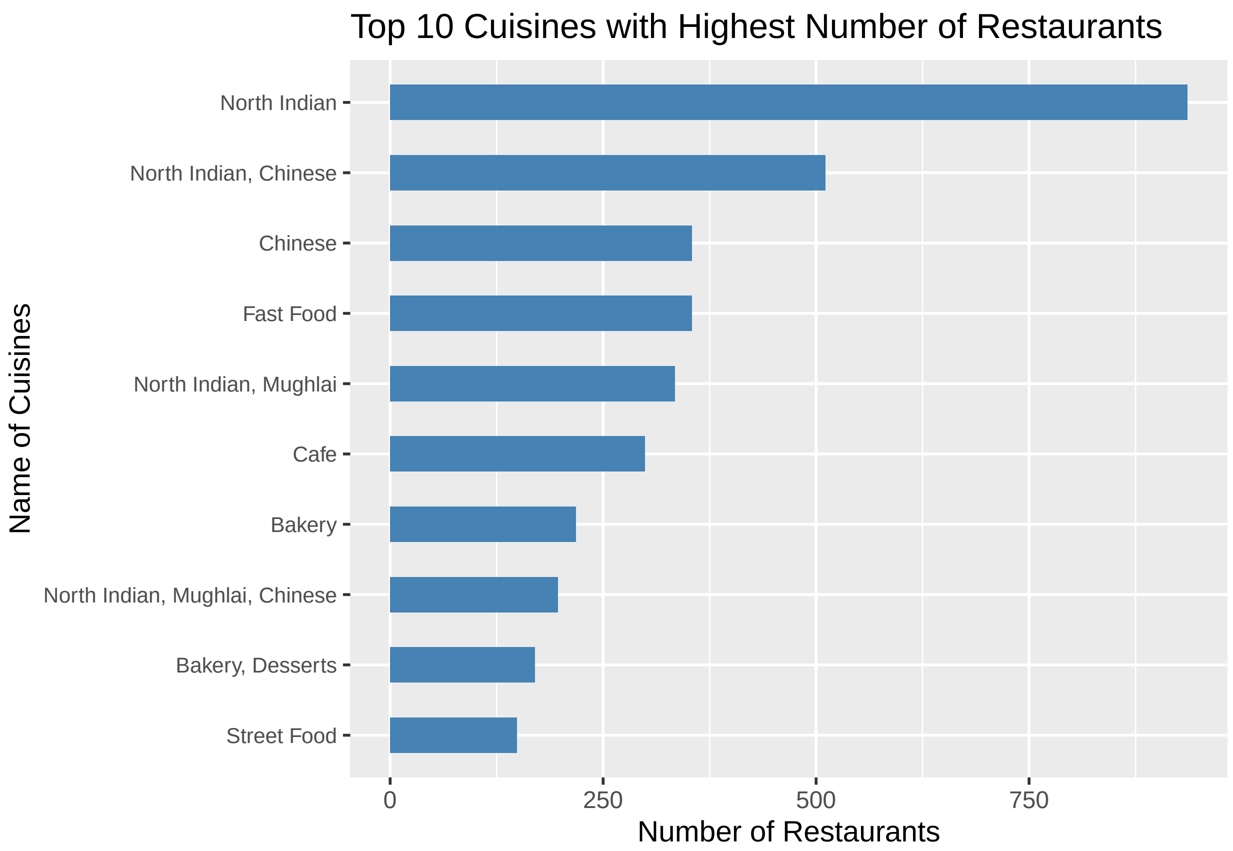
```r
# The Distribution of Categorical Variables like 'Country Code', 'City', and 'Cuisines'

# Create count plot for Country Code
ggplot(data, aes(x = factor(Country.Code))) +
  geom_bar(fill = "steelblue") +
  labs(title = "Distribution of Restaurants by Country Codes",
       x = "Country Codes", y = "Number of Restaurants")
```



```r
# Create a subset of the data containing only the top 10 cities
top_10_cities <- head(names(sort(table(data$City), decreasing = TRUE)), 10)
data_top_10_cities <- data[data$City %in% top_10_cities, ]

# Create count plot for top 10 cities (horizontal bar plot)
ggplot(data = data_top_10_cities, aes(y = factor(City, levels = rev(top_10_cities)))) +
  geom_bar(fill = "steelblue", width = 0.5, stat = "count") +
  labs(title = "Top 10 Cities with Highest Number of Restaurants",
       x = "Number of Restaurants", y = "Name of Cities") +
  theme(axis.text.y = element_text(size = 8))
```



```r
# Subset the data to include only the top 10 cuisines
top_10_cuisines <- head(names(sort(table(data$Cuisines), decreasing = TRUE)), 10)
data_top_10 <- data[data$Cuisines %in% top_10_cuisines, ]

# Create count plot for top 10 cuisines (horizontal bar plot)
ggplot(data = data_top_10, aes(y = factor(Cuisines, levels = rev(top_10_cuisines)))) +
  geom_bar(fill = "steelblue", width = 0.5, stat = "count") +
  labs(title = "Top 10 Cuisines with Highest Number of Restaurants",
       x = "Number of Restaurants", y = "Name of Cuisines") +
  theme(axis.text.y = element_text(size = 8))
```



```r
# The top cuisines and cities with the highest number of restaurants

# Identify the top 10 cuisines and their counts
top_cuisines <- head(sort(table(data$Cuisines), decreasing = TRUE), 10)

# Create a dataframe with cuisine names and counts
top_cuisines_df <- data.frame(Cuisine = names(top_cuisines), Count = as.numeric(top_cuisines))

# Display the dataframe
print("Top 10 Cuisines with the Highest Number of Restaurants:")
```

```
## [1] "Top 10 Cuisines with the Highest Number of Restaurants:"
```

```r
print(top_cuisines_df)
```

```
##                            Cuisine Count
## 1                     North Indian   936
## 2            North Indian, Chinese   511
## 3                          Chinese   354
## 4                        Fast Food   354
## 5            North Indian, Mughlai   334
## 6                             Cafe   299
## 7                           Bakery   218
## 8  North Indian, Mughlai, Chinese   197
## 9                 Bakery, Desserts   170
## 10                    Street Food   149
```

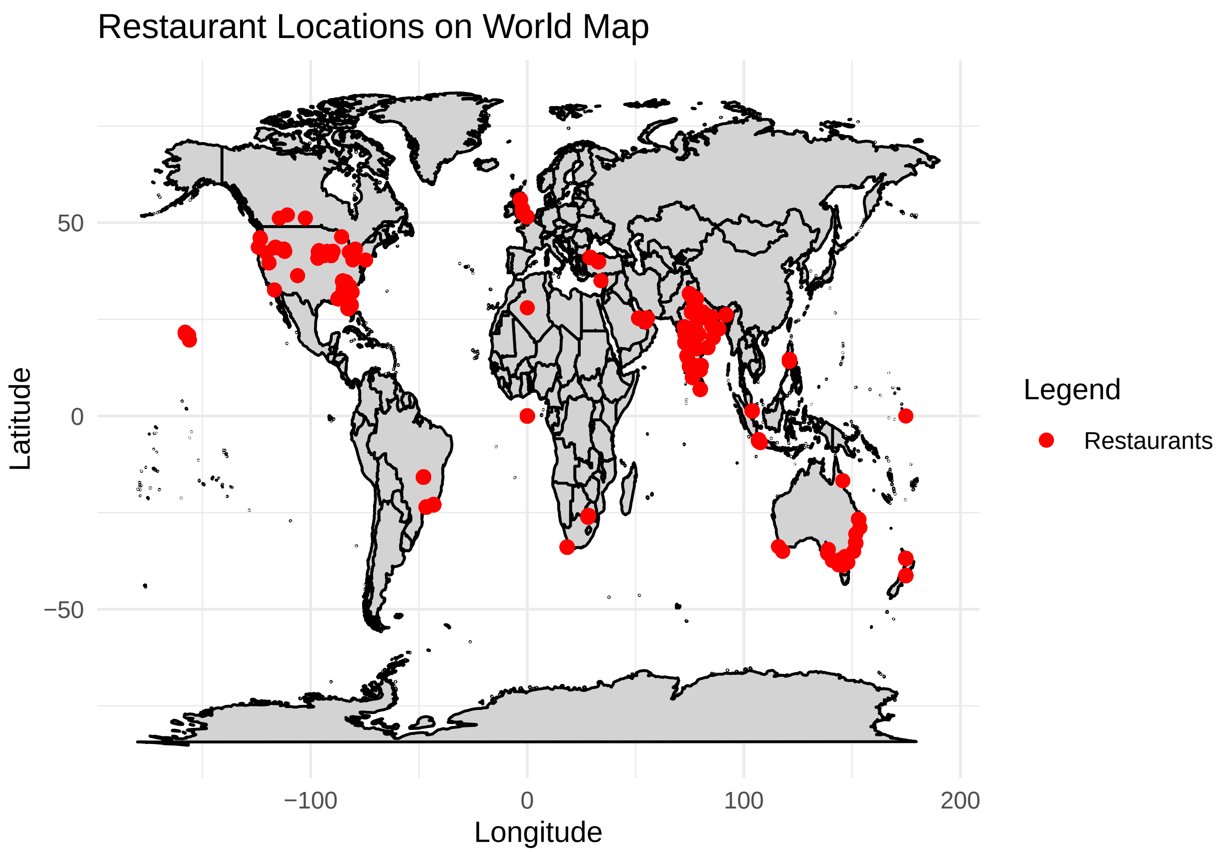
```
# Identify the top 10 city and their counts
top_city <- head(sort(table(data$City), decreasing = TRUE), 10)

# Create a dataframe with city names and counts
top_city_df <- data.frame(City = names(top_city), Count = as.numeric(top_city))

# Display the dataframe
print("Top 10 Cities with the Highest Number of Restaurants:")
```

```
## [1] "Top 10 Cities with the Highest Number of Restaurants:"
```

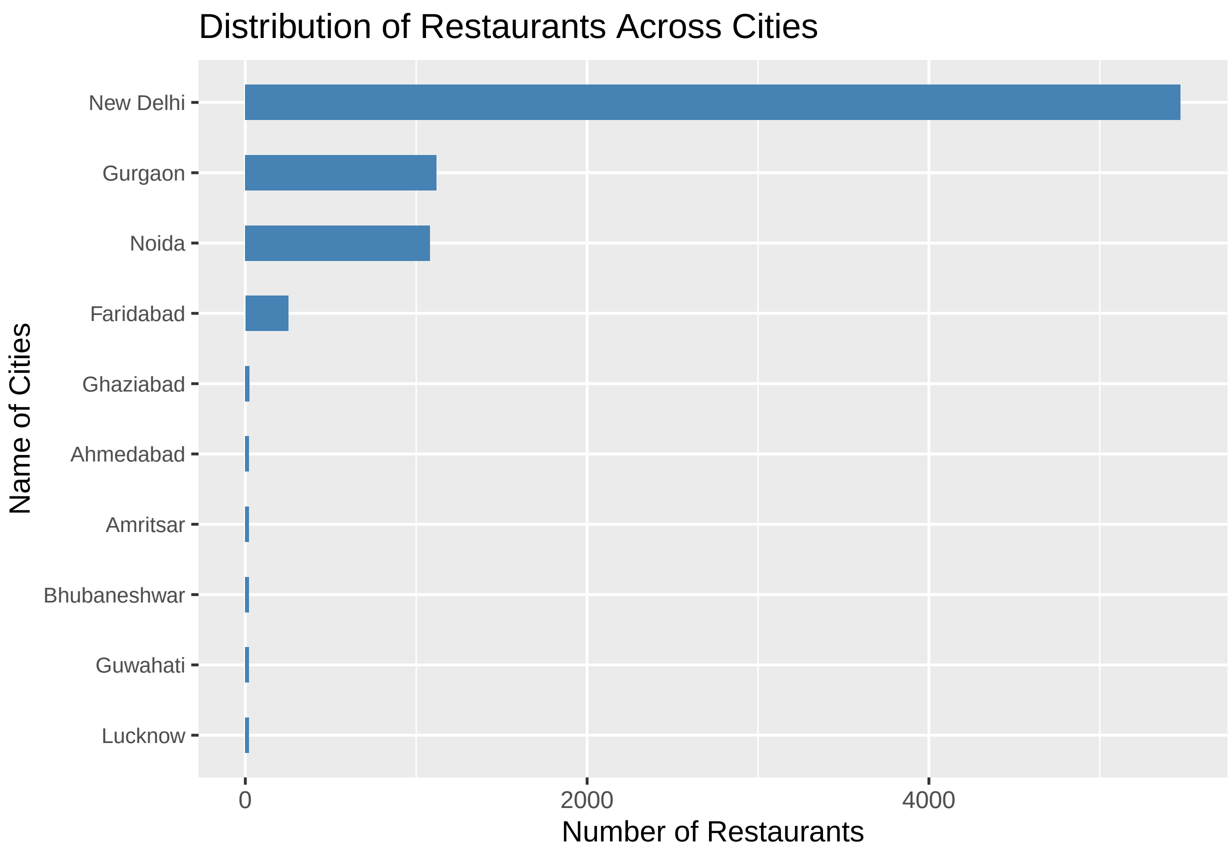
```
print(top_city_df)
```

```
##            City Count
## 1     New Delhi  5473
## 2       Gurgaon  1118
## 3         Noida  1080
## 4     Faridabad   251
## 5     Ghaziabad    25
## 6     Ahmedabad    21
## 7      Amritsar    21
## 8  Bhubaneshwar    21
## 9      Guwahati    21
## 10      Lucknow    21
```

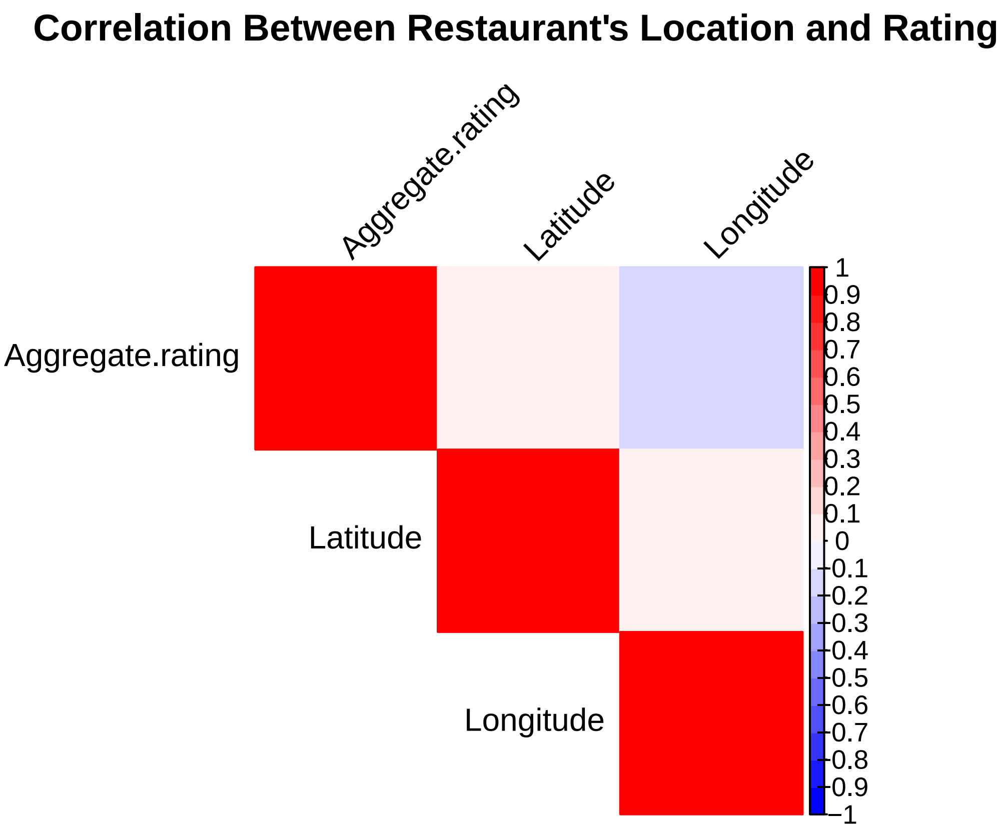
```
## Task 3: GEOSPATIAL ANALYSIS


# Visualize the locations of restaurants on a map

# Create a map of the world
world_map <- map_data("world")

# Plot restaurant locations on the map
ggplot() +
  geom_polygon(data = world_map, aes(x = long, y = lat, group = group), fill = "lightgrey", color = "bla
  geom_point(data = data, aes(x = Longitude, y = Latitude, color = "Restaurants"), size = 2) +
  scale_color_manual(name = "Legend", values = c(Restaurants = "red")) +
  labs(title = "Restaurant Locations on World Map", x = "Longitude", y = "Latitude") +
  theme_minimal()
```



```r
# Analyze the distribution of restaurants across different cities or countries

# Create a subset of the data containing only the top 10 cities
top_10_cities <- head(names(sort(table(data$City), decreasing = TRUE)), 10)
data_top_10_cities <- data[data$City %in% top_10_cities, ]

# Create a plot of the distribution of restaurants across cities
ggplot(data = data_top_10_cities, aes(y = factor(City, levels = rev(top_10_cities)))) +
  geom_bar(fill = "steelblue", width = 0.5, stat = "count") +
  labs(title = "Distribution of Restaurants Across Cities",
       x = "Number of Restaurants", y = "Name of Cities") +
  theme(axis.text.y = element_text(size = 8))
```



```r
# Determine if there is any correlation between the restaurant's location and its rating

# Calculate the correlation matrix
correlation_matrix <- cor(data[c("Latitude", "Longitude", "Aggregate.rating")])

# Create a heatmap to visualize the correlation
corrplot(correlation_matrix, method="color", col=colorRampPalette(c("blue", "white", "red"))(20), type=
         order="hclust", tl.col="black", tl.srt=45, title="Correlation Between Restaurant's Location an
         mar=c(0, 0, 3, 1))
```


Correlation Between Restaurant's Location and Rating
Aggregate.rating
Latitude
Longitude
Aggregate.rating
Latitude
Longitude
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.7
−0.8
−0.9
−1