Muthukumarasamy.S

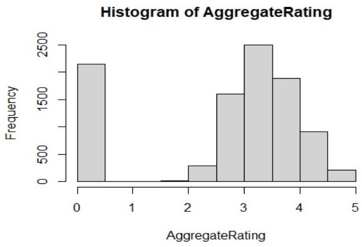
2024-06-07

Level 1

Task 1: Data Exploration and Preprocessing

Explore the dataset and identify the number of rows and columns. Check for missing values in each column and handle them accordingly. Perform data type conversion if necessary. Analyze the distribution of the target variable (“Aggregate rating”) and identify any class imbalances.

```
df=read.csv('E:/Virtual_Intern/Dataset .csv')
print("Number of Rows:")

## [1] "Number of Rows:"

print(nrow(df))

## [1] 9551

print("Number of Columns:")

## [1] "Number of Columns:"

print(ncol(df))

## [1] 21

print("Missing Values:")

## [1] "Missing Values:"

print(sum(is.na(df)))

## [1] 0

AggregateRating=df$Aggregate.rating
hist(AggregateRating)
```

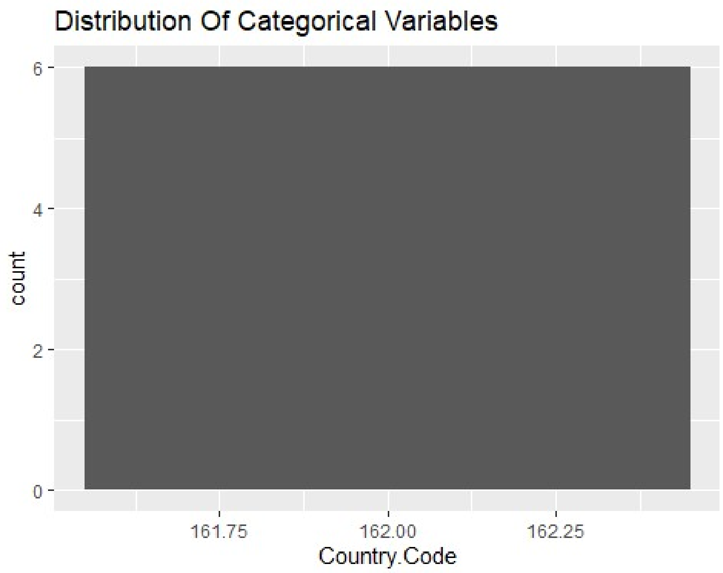


Task 2:Descriptive Analysis

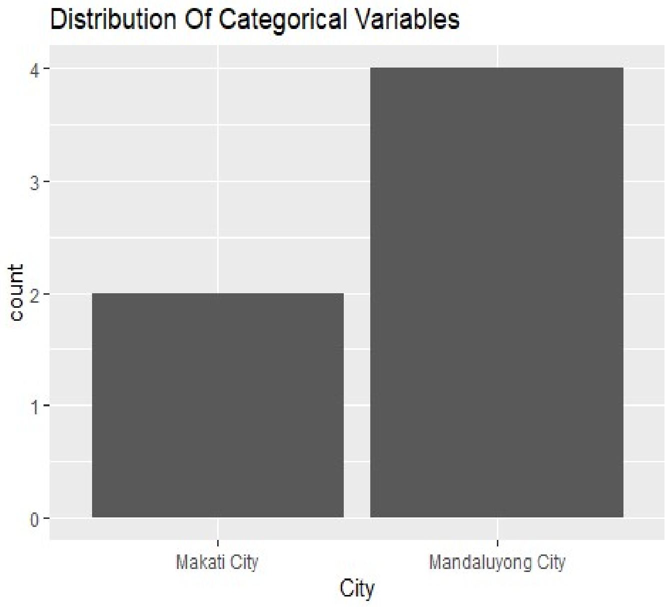
Calculate basic statistical measures (mean, median, standard deviation, etc.) for numerical columns. Explore the distribution of categorical variables like “Country Code,” “City,” and “Cuisines.” Identify the top cuisines and cities with the highest number of restaurants.

```r
print("Mean,Median and Standard Deviation Of Various Numerical Features")

## [1] "Mean,Median and Standard Deviation Of Various Numerical Features"

mean(df$Restaurant.ID,na.rm=TRUE)

## [1] 9051128

median(df$Restaurant.ID,na.rm=TRUE)

## [1] 6004089

sd(df$Restaurant.ID,na.rm=TRUE)

## [1] 8791521

mean(df$Country.Code,na.rm=TRUE)

## [1] 18.36562

median(df$Country.Code,na.rm=TRUE)

## [1] 1

sd(df$Country.Code,na.rm=TRUE)

## [1] 56.75055

mean(df$Longitude,na.rm=TRUE)

## [1] 64.12657

median(df$Longitude,na.rm=TRUE)

## [1] 77.19196

sd(df$Longitude,na.rm=TRUE)

## [1] 41.46706

mean(df$Latitude,na.rm=TRUE)

## [1] 25.85438

median(df$Latitude,na.rm=TRUE)

## [1] 28.57047
```

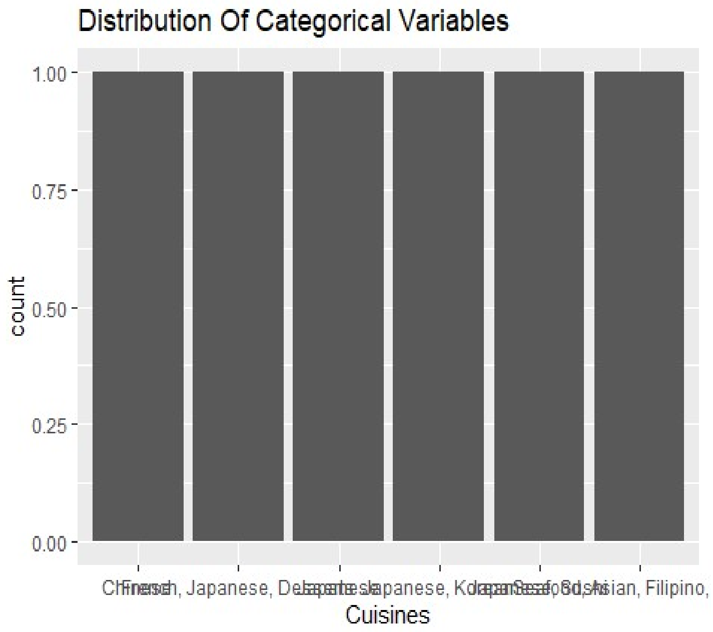
```
sd(df$Latitude,na.rm=TRUE)

## [1] 11.00794

mean(df$Average.Cost.for.two,na.rm=TRUE)

## [1] 1199.211

median(df$Average.Cost.for.two,na.rm=TRUE)

## [1] 400

sd(df$Average.Cost.for.two,na.rm=TRUE)

## [1] 16121.18

mean(df$Price.range,na.rm=TRUE)

## [1] 1.804837

median(df$Price.range,na.rm=TRUE)

## [1] 2

sd(df$Price.range,na.rm=TRUE)

## [1] 0.9056088

mean(df$Aggregate.rating,na.rm=TRUE)

## [1] 2.66637

median(df$Aggregate.rating,na.rm=TRUE)

## [1] 3.2

sd(df$Aggregate.rating,na.rm=TRUE)

## [1] 1.516378

mean(df$Votes,na.rm=TRUE)

## [1] 156.9097

median(df$Votes,na.rm=TRUE)

## [1] 31

sd(df$Votes,na.rm=TRUE)

## [1] 430.1691

library(ggplot2)
x1<-head(df)
ggplot()+
geom_bar(data=x1,mapping = aes(x=Country.Code))+labs(title = "Distribution Of
Categorical Variables")
```



```
ggplot()+
geom_bar(data=x1,mapping = aes(x=City))+labs(title = "Distribution Of
Categorical Variables")
```



```
ggplot()+
geom_bar(data=x1,mapping = aes(x=Cuisines))+labs(title = "Distribution Of
Categorical Variables")
```



```
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

ans<-df %>% group_by(City) %>% summarize(count=n())
print("Highest Number Of Restaurants with Top Cities")

## [1] "Highest Number Of Restaurants with Top Cities"

head(arrange(ans,desc(count)),10)

## # A tibble: 10 × 2
##    City         count
##    <chr>        <int>
##  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
```

```
ans1<-df %>% group_by(Cuisines) %>% summarize(count=n())
print("Highest Number Of Restaurants with Top Cuisines")
```

```
## [1] "Highest Number Of Restaurants with Top Cuisines"
```

```
head(arrange(ans1,desc(count)),10)
```

```
## # A tibble: 10 × 2
##    Cuisines                       count
##    <chr>                          <int>
##  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
```

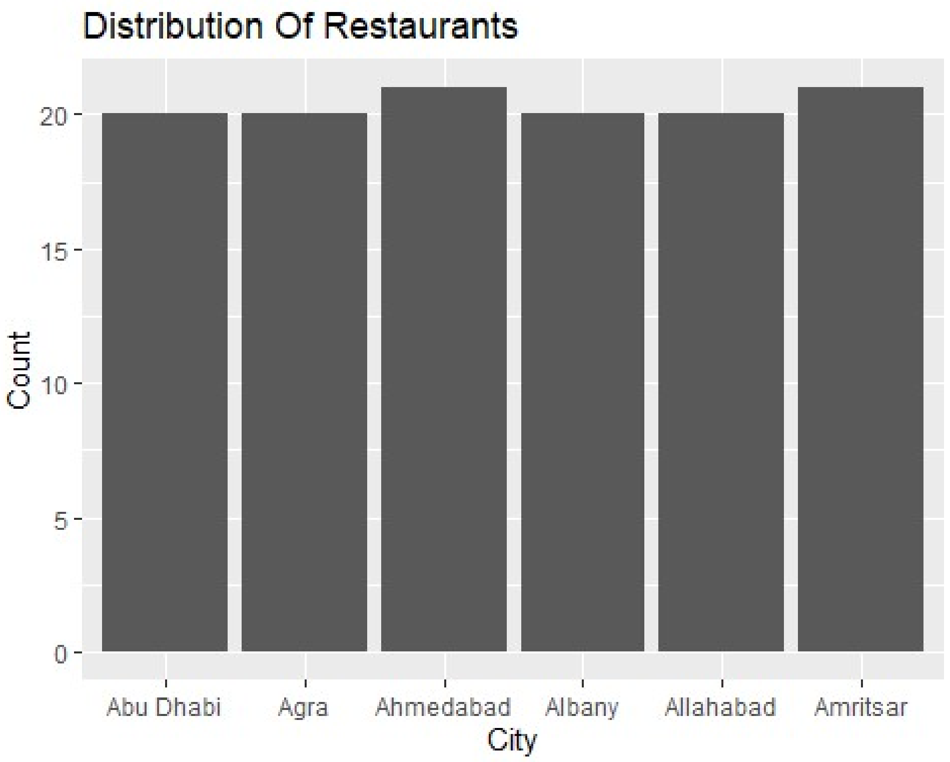
Task 3: Geospatial Analysis

Visualize the locations of restaurants on a map using latitude and longitude information. Analyze the distribution of restaurants across different cities or countries. Determine if there is any correlation between the restaurant's location and its rating.

```
library(leaflet)
map<-leaflet(df) %>% addTiles() %>% setView(lng = mean(df$Longitude),lat =
mean(df$Latitude),zoom=4)
map<-map %>% addCircleMarkers(lng = ~Longitude,lat =
~Latitude,popup=~paste("Locality:",`Locality`),radius = 3,color =
'red',stroke = FALSE,fillOpacity = 0.6)
library(htmlwidgets)

saveWidget(map,'restaurant.html',selfcontained = TRUE)
print("Map is Saved")
```

```
## [1] "Map is Saved"
```

```
group_Restaurant<- df %>% group_by(Restaurant.ID)
Group_City<- group_Restaurant %>% group_by(City) %>% summarize(Count=n())
top_Restaurant<-head(Group_City)

ggplot(data = top_Restaurant)+geom_bar(mapping=aes(x=City,y=Count),stat =
"identity")+labs(title = "Distribution Of Restaurants")
```



```
category_to_numeric<-as.numeric(factor(df$Locality))
cor_val<-cor(category_to_numeric,df$Aggregate.rating)
if(cor_val<0){
  print("Datas(Locality and Aggregate Rating) are -vely Correlated")
}else{
  print("Datas(Locality and Aggregate Rating) are Positively Correlated")
}

## [1] "Datas(Locality and Aggregate Rating) are -vely Correlated"
```