**# dataset : https://www.kaggle.com/hugodarwood/epirecipes/data**

**# Perform the below given activities:**

**# a. apply K-means clustering to identify similar recipies**

**# b. apply K-means clustering to identify similar attributes**

**# c. how many unique recipies that people order often**

**# d. what are their typical profiles**

**# Import Zip File**

getwd()

setwd("E:\\R\\Assignment\\Assignment 22")

data <- read.csv(unzip("epi\_r.zip"))

View(data)

dim(data)

str(data)

**# Preprocessing the data set**

colnames(data)

library(maps)

data("world.cities")

world.cities$country.etc <- toupper(world.cities$country.etc)

raw <- colnames(data) # column names tored in a vector

raw <- gsub("[[:punct:]\n]","",raw) # Removing punctuation

raw <- strsplit(raw, " ") # Split data at word boundaries

raw <- toupper(raw) # convert to upper case

length(raw)

**# Match on country / cities in world.countries**

CountryList\_raw <- (lapply(raw, function(x)x[which(x %in% world.cities$country.etc)]))

colnames(data) <- raw

# check for NA

sum(is.na(data))

sort(sapply(data, function(x) sum(is.na(x))))

**# impute missing values**

library(mice)

imputed = mice(data[,c("CALORIES", "SODIUM", "PROTEIN", "FAT")], method='cart', m=5)

imputed <- mice::complete(imputed)

# replacing NAs with imputed values

data$CALORIES <- imputed$CALORIES

data$PROTEIN <- imputed$PROTEIN

data$SODIUM <- imputed$SODIUM

data$FAT <- imputed$FAT

sum(is.na(data))

**# checking for outliers**

library(ggplot2)

ggplot(reshape2::melt(data[,c("CALORIES", "SODIUM", "PROTEIN", "FAT")]),

aes(x= variable, value, fill = variable))+

geom\_boxplot()+facet\_wrap(~variable, scales = 'free\_y')

# yes there are outliers

**# removing these outliers**

df <- outliers::rm.outlier(data[,c("CALORIES", "SODIUM", "PROTEIN", "FAT")], fill = TRUE)

data$CALORIES <- df$CALORIES

data$PROTEIN <- df$PROTEIN

data$SODIUM <- df$SODIUM

data$FAT <- df$FAT

dim(data)

TITLE <- data$TITLE

# Load required libraries

library(tidyverse) # data manipulation

library(cluster) # clustering algorithms

library(factoextra) # clustering algorithms & visualization

**# a. apply K-means clustering to identify similar recipies**

**# preparing data set for receipe**

set.seed(123)

data\_recipe <- data[,-c(1,unlist(which(raw %in% world.cities$country.etc)))]

data\_recipe <- scale(data\_recipe)

**# Compute k-means clustering with k = 5**

final\_recipe <- kmeans(data\_recipe, 5, nstart = 25)

summary(final\_recipe)

table(final\_recipe$cluster) # cluster for similar recipes

fviz\_cluster(final\_recipe, data = data\_recipe)

**# b. apply K-means clustering to identify similar attributes**

**# preparing data set for receipe**

set.seed(123)

data\_att <- data[,c(unlist(which(raw %in% world.cities$country.etc)))]

**# Compute k-means clustering with k = 2**

final\_att <- kmeans(data\_att, 2, nstart = 25)

summary(final\_att)

table(final\_att$cluster) # cluster for similar attributes

fviz\_cluster(final\_att, data = data\_att)

**# c. how many unique recipies that people order often**

df$Clusters <- final\_recipe$cluster

df$TITLE <- TITLE

by\_cluster <- df %>% group\_by(Clusters) %>% summarise\_all("length") %>% select(Clusters, TITLE)

by\_cluster

max(by\_cluster$TITLE)

**# d. what are their typical profiles**

profile <- (df[,-6] %>% group\_by(Clusters) %>% summarise\_all("mean") %>%

select("CALORIES", "SODIUM", "PROTEIN", "FAT"))[1,] profile