#Before Starting

# Install packages

install.packages(c("dplyr", "ggplot2", "gridExtra", "tidyr", "car", "lmtest", "corrplot"))

# Load packages

library(readr) # Reading and writing data

library(dplyr) # Data manipulation

library(corrplot) # Correlation matrix visualization

library(car) # Linear model diagnostics

library(gmodels) # Contingency tables and hypothesis tests for categorical data

library(lmtest) # Hypothesis testing in linear regression models

library(DescTools) # Descriptive statistics

library(tidyverse) # Collection of packages for data manipulation

library(ggplot2) # Data visualization

library(gridExtra) # Arrangement of multiple grid-based figures

library(tidyr) # Data tidying

setwd("/Volumes/Academic/6th semester/Computing/lab/sample question solution") # ekhane toi tor csv file jekhane rakhcht tar path bosabi

file.choose() # eta dile file select kore nibi

# Packages a giye base package a tik mark de

#1. Read the given dataset (ObesityData.csv) into R.

# Load the dataset

obes\_data <-obes\_data <- read.csv("ObesityData.csv", header = TRUE)

obes\_data

# Display the structure of the dataset

str(obes\_data)

#2. Calculate frequencies, percentages and cumulative frequencies of all the categorical variables.

# At 1st, extract only the categorical variables from the dataset

cate <- obes\_data[sapply(obes\_data, is.factor) | sapply(obes\_data, is.character)]

# Display the structure of the extracted categorical variables

str(cate)

for (c in names(cate)) {

tab <- table(cate[[c]])

cat("Variable:", c, "\n")

print(cbind(freq = tab, perc = prop.table(tab) \* 100, cumfreq = cumsum(tab)))

cat("\n")

}

#Q3 Present the categorical variables using suitable graph (e.g., horizontal and vertical bardagram, pie chart, etc.). [Figures must be clear and contain all the necessary components such as title of the figure, titles of the horizontal and vertical axis, values in each axis,legend, etc.]

# Set up the layout for multiple plots with larger margins

par(mfrow = c(2, 4), mar = c(3, 3, 2, 1))

# Create a varticle bar plots for each specified categorical variable

for (c in names(cate)) {

# Create a vertical bar plot

barplot(table(cate[[c]]), main =c, xlab = "Category", ylab = "Frequency", col = rainbow(length(unique(cate[[c]]))))

}

for (c in names(cate)) {

# Create a horizontal bar plot each specified categorical variable

barplot(table(cate[[c]]), main = c, xlab = "Frequency", ylab = "Category", horiz = TRUE, col = rainbow(length(unique(cate[[c]]))))

}

# Create pie charts for each specified categorical variable

par(mfrow=c(3,3),mar=c(4,4,2,1)) # lay out abar bole ditaci sondor ekta pie er shape asar jnno

for (c in names(cate)) {

pie(table(cate[[c]]),main = c, col=rainbow(length(unique(cate[[c]]))))

}

# Reset the layout

par(mfrow = c(1, 1))

#Q4 Calculate appropriate summary statistics (e.g., minimum, maximum, mean, median, mode, 1st quartile, 3rd quartile, standard deviation, variance, coefficient of variation, interquartile range, coefficient of quartile deviation, etc.) for each of the numerical variables.

# Calculate summary statistics for each numerical variable

#summary statistics mone rakhich always numerical variable er jnno hy, category er jnno hy na

# Extract only the numerical variables from the dataset and rename to 'num'

num <- obes\_data[sapply(obes\_data, is.numeric)]

#1st a tor kaj hocche ekta user define function create korbi, jar pete thakbe mean, meadian, .... funtion gola jegola question dekhaite bolbe

user\_def<- function(x){

c(

mean = mean(x, na.rm = TRUE),

median = median(x, na.rm = TRUE),

mode = as.numeric(names(sort(table(x), decreasing = TRUE)[1])) # etar formula parle mone rakhbi naile baddhi

,

min = min(x, na.rm = TRUE),

max = max(x, na.rm = TRUE),

q1 = quantile(x, 0.25, na.rm = TRUE),

q3 = quantile(x, 0.75, na.rm = TRUE),

sd = sd(x, na.rm = TRUE),

var = var(x, na.rm = TRUE),

cv = sd(x, na.rm = TRUE) / mean(x, na.rm = TRUE) \* 100, #cv er definition+formula google korich

iqr = IQR(x, na.rm = TRUE),

#coefficient of quantile deviation

cqd = IQR(x, na.rm = TRUE) / (quantile(x, 0.75, na.rm = TRUE) + quantile(x, 0.25, na.rm = TRUE))

)

}

apply(num, 2,user\_def) #apply funtion kivabe kaj kore messenger grp a link ache dekhich

# Q5 Present the numerical variables using suitable graph (e.g., histogram, frequency curve,histogram with normal probability curve, box plot, stem and leaf plot, etc.).

# Set up the layout for multiple plots

par(mfrow = c(2, 4), mar = c(3, 3, 2, 1))

# Create histograms for each numerical variable

for (col in names(num)) {

hist(num[[col]], main = paste("Histogram of", col), xlab = col, col = "lightblue")

#col a colar ektai taici jate mone thake exam a

}

# Reset the layout

par(mfrow = c(1, 1))

# Set up the layout for multiple plots

par(mfrow = c(2, 4), mar = c(3, 3, 2, 1))

# Frequency curve (density plot)

for (col in names(num)) {

plot(density(num[[col]]), main = paste("Density Plot of", col), col = "lightblue")

}

# Histogram with normal probability curve

for (col in names(num)) {

hist(num[[col]], main = paste("Histogram with Normal Curve of", col), xlab = col, col = "lightblue", prob = TRUE)

lines(density(num[[col]]), col = "black")

}

# Box plot

for (col in names(num)) {

boxplot(num[[col]], main = paste("Boxplot of", col), col = "lightblue", border = "black")

}

# Stem-and-leaf plot

for (col in names(num)) {

stem(num[[col]])

}

# Reset the layout

par(mfrow = c(1, 1))

#Q6 Explain which summary measures and graphs are appropriate to present each individual variable and why. [Hints: test normality for each of the numerical variables. If it follows normal distribution, present it using mean, standard deviation/variance, coefficient of variation, histogram, etc. and it does not follow normal distribution, present it using median, interquartile range, coefficient of quartile deviation, box plot, etc.]

# Extract only the numerical variables from the dataset and rename to 'num'

num <- obes\_data[sapply(obes\_data, is.numeric)]

# Test normality

for (col in names(num)) {

# Extract the variable from the data frame

tem\_var <- num[[col]]

# Perform the Shapiro-Wilk test

st <- shapiro.test(tem\_var)

# Print the p-value

print(paste("P-value for", col, ":", st))

#normal<-data.frame("X", "Y", "Z")

#p value dekhe dekhe normality mane emon variable nibi x,y,z er jaygay bosai bi bosabo manually

# p value 0.05 er cheye choto hole shapiro test, oi variable normality follow kore na

#j j variable normal

not\_normal<-data.frame("Age", "Height", "Weight", "FCVC", "NCP", "CH2O", "FAF", "TUE")

#ekhaneo toi ekta temporary funtion create kortecht,jate shortcut a man gola ber kora jay

non\_normal <- sapply(num, function(x) {

c(

median = median(x),

iqr = IQR(x),

cqd = IQR(x) / (quantile(x, 0.75) +quantile(x, 0.25))

)

})

non\_normal <- as.data.frame(non\_normal)

non\_normal

# #q7 Perform exploratory subgroup analysis: calculate summary statistics and draw graphs for

# each numerical variable by every categorical variable. For example, calculate minimum,

# maximum, mean, median, mode, 1st quartile, 3rd quartile, standard deviation, variance,

# coefficient of variation, interquartile range, coefficient of quartile deviation of age, height,

# weight, FCVC, CH2O , FAF, TUE separately for every level of the categorical variables

# such as separately for male and female, smoker and non-smoker, … so on.

# Extract categorical and numerical variables

cat <- c("Gender", "Family\_history\_with\_overweight", "CAEC", "SMOKE", "SCC", "CALC", "MTRANS")

num <- c("Age", "Height", "Weight", "FCVC", "CH2O", "FAF", "TUE")

# Loop over each categorical variable

for (cat\_var in cat) {

# Print the categorical variable name

cat("\n\nExploratory Analysis for", cat\_var, "\n")

# Loop over each numerical variable

for (num\_var in num) {

# Check if the current variable is in the dataset

if (!(cat\_var %in% names(obes\_data) && num\_var %in% names(obes\_data))) {

cat("Skipping undefined columns:", cat\_var, num\_var, "\n")

next

}

# Extract data for the current subgroup

subgroup\_data <- obes\_data[, c(cat\_var, num\_var), drop = FALSE]

# Print variable name

cat("\nVariable:", num\_var, "\n")

# Calculate summary statistics

summary\_stats <- tapply(subgroup\_data[[num\_var]], subgroup\_data[[cat\_var]], function(x) {

c(

min = min(x, na.rm = TRUE),

max = max(x, na.rm = TRUE),

mean = mean(x, na.rm = TRUE),

median = median(x, na.rm = TRUE),

mode = as.numeric(names(sort(table(x), decreasing = TRUE)[1])),

q1 = quantile(x, 0.25, na.rm = TRUE),

q3 = quantile(x, 0.75, na.rm = TRUE),

sd = sd(x, na.rm = TRUE),

var = var(x, na.rm = TRUE),

cv = sd(x, na.rm = TRUE) / mean(x, na.rm = TRUE) \* 100,

iqr = IQR(x, na.rm = TRUE),

cqd = IQR(x, na.rm = TRUE) / (quantile(x, 0.75, na.rm = TRUE) - quantile(x, 0.25, na.rm = TRUE))

)

})

# Convert the result to a data frame

summary\_stats\_df <- as.data.frame(do.call(rbind, summary\_stats))

summary\_stats\_df

}

}

# #q8 Draw a scatter diagram between (i) age and weight, (ii) age and height and (iii) height and

# weight. Draw respective regression lines on the scatterplots. Also, draw scatterplot matrix

# of all the numerical variables. Comment on your results.

# Scatter diagram and regression line for Age and Weight

plot(obes\_data$Age, obes\_data$Weight, main = "Scatterplot: Age vs Weight", xlab = "Age", ylab = "Weight")

abline(lm(Weight ~ Age, data = obes\_data), col = "red")

# Scatter diagram and regression line for Age and Height

plot(obes\_data$Age, obes\_data$Height, main = "Scatterplot: Age vs Height", xlab = "Age", ylab = "Height")

abline(lm(Height ~ Age, data = obes\_data), col = "red")

# Scatter diagram and regression line for Height and Weight

plot(obes\_data$Height, obes\_data$Weight, main = "Scatterplot: Height vs Weight", xlab = "Height", ylab = "Weight")

abline(lm(Weight ~ Height, data = obes\_data), col = "red")

# Scatterplot matrix for all numerical variables

pairs(obes\_data[, num], main = "Scatterplot Matrix")

# #q9 Recode the variable MTRANS into MTRANS\_RC in which both the ‘Walking’ and ‘Bike’

# categories will be ‘Ownself’ and all other categories will be ‘Car’. Also, convert the

# variable FCVC to a factor (name as FCVC\_factor) by labeling 1 as ‘Never’, 2 as

# ‘Sometimes’ and 3 as ‘Always’

# Recode MTRANS variable

obes\_data$MTRANS\_RC <- ifelse(obes\_data$MTRANS %in% c("Walking", "Bike"), "Ownself", "Car") #ifelse diye kivabe record kore dekhe nich, lecture 2 teo chio eta

obes\_data$MTRANS\_RC

# Convert FCVC to a factor with labeled levels

obes\_data$FCVC\_factor <- factor(obes\_data$FCVC, levels =c(1,2,3), labels = c("Never", "Sometimes", "Always"))

obes\_data$MTRANS\_RC

# #q10 Calculate body mass index (BMI) by using the formula, BMI = Weight/(Height)2. Make a

# categorical variable (name as BMI\_cat) using the following categorization of BMI values:

# Less than 18.5 as ‘Underweight’; 18.5 to 24.9 as ‘Normal’; 25.0 to 29.9 as ‘Overweight’;

# Greater than 30 as ‘Obesity’.

# Calculate BMI

obes\_data$BMI <- obes\_data$Weight / (obes\_data$Height)^2

View(obes\_data) #eta diye dekh tor BMI name ekta column create hyeche

# Create BMI\_cat based on categorization

obes\_data$BMI\_cat <- ifelse(obes\_data$BMI < 18.5, "Underweight",

ifelse(obes\_data$BMI <= 24.9, "Normal",

ifelse(obes\_data$BMI <= 29.9, "Overweight", "Obesity")))

View(obes\_data)

# #q11 Calculate BMI for respondents (i) whose age > 30 years, (ii) who is non-smoker, have

# physical activity of 2 days and drinks more than 1 liter of water daily.

# (i) Calculate BMI for respondents whose age > 30 years

subset1 <- obes\_data[obes\_data$Age > 30, ]

subset1$BMI <- subset1$Weight / (subset1$Height)^2

# (ii) Calculate BMI for non-smokers with physical activity of 2 days and daily water consumption > 1 liter

View(obes\_data) # eta diye age dekh smoke,faf r ch2o ei 3 column er data gola kivabe ache. then condition diye subset kore nibo

subset2 <- obes\_data[obes\_data$SMOKE == "no" & obes\_data$FAF == 2 & obes\_data$CH2O > 1, ]

subset2$BMI <- subset2$Weight / (subset2$Height)^2

View(obes\_data)

# View the calculated BMI for the subsets

head(subset1[, c("Age", "Weight", "Height", "BMI")])

head(subset2[, c("SMOKE", "FAF", "CH2O", "Weight", "Height", "BMI")])

# subset1: This refers to the subset of the obes\_data dataframe that you created earlier using the condition obes\_data$Age > 30. It contains the rows where the age is greater than 30.

# [, c("Age", "Weight", "Height", "BMI")]: This part of the code is used to select specific columns from the dataframe. The comma in the square brackets is used to subset the data. The c("Age", "Weight", "Height", "BMI") part specifies the column names you want to select.

# head(...): The head() function is used to show the first few rows of the resulting dataframe. By default, it displays the first 6 rows, but you can specify a different number inside the parentheses to show a different number of rows.

#

# #q12 Create a new dataset (name as: obesity\_sub) by taking the respondents whose height is

# more than 1.8 meter and who eat high caloric food frequently. Calculate mean and standard

# deviation of BMI using the obesity\_sub dataset.

# Create obesity\_sub dataset

View(obes\_data)

obesity\_sub <- obes\_data[obes\_data$Height > 1.8 & obes\_data$FAVC == "yes", ]

#ekhane FAVC ken nilam question upore description dekh

#In this case, only row conditions are specified. tai "yes" er ekta comma

obesity\_sub

# Calculate BMI for obesity\_sub dataset

obesity\_sub$BMI <- obesity\_sub$Weight / (obesity\_sub$Height)^2

# Calculate mean and standard deviation of BMI

mean\_bmi <- mean(obesity\_sub$BMI, na.rm = TRUE)

sd\_bmi <- sd(obesity\_sub$BMI, na.rm = TRUE)

mean\_bmi

sd\_bmi

# #q13 Calculate correlation between (i) age and weight, (ii) age and height and (iii) height and

# weight. Compare and contrast these results with the results of the task number 8.

# (i) Calculate correlation between age and weight

cor\_age\_weight <- cor(obes\_data$Age, obes\_data$Weight, use = "complete.obs")

cor\_age\_weight

# (ii) Calculate correlation between age and height

cor\_age\_height <- cor(obes\_data$Age, obes\_data$Height, use = "complete.obs")

cor\_age\_height

# (iii) Calculate correlation between height and weight

cor\_height\_weight <- cor(obes\_data$Height, obes\_data$Weight, use = "complete.obs")

cor\_height\_weight

# #q14 Calculate correlation between age and BMI and comment on the relationship. Does this

# correlation significantly differ from zero? Calculate correlation matrix of all the numerical

# variables. Comment on your results.

# Calculate correlation between age and BMI

cor\_age\_bmi <- cor(obes\_data$Age, obes\_data$BMI, use = "complete.obs")

cor\_age\_bmi

# Null hypothesis: There is no correlation between age and BMI (correlation = 0)

# Alternative hypothesis: There is a correlation between age and BMI (correlation ≠ 0)

# Test whether the correlation is significantly different from zero

cor\_test\_result <- cor.test(obes\_data$Age, obes\_data$BMI)

cor\_test\_result

# # Create a heatmap of the correlation matrix

# corrplot(cor\_matrix, method = "color")

#null hypo reject korbi jokhon p value 0.05 thek choto hbe ekhane p value=2.2e-16

# Calculate the correlation matrix of all numerical variables

cor\_matrix <- cor(obes\_data[, c("Age", "Weight", "Height", "BMI", "FCVC", "CH2O", "FAF", "TUE")], use = "complete.obs")

cor\_matrix

# #Q15 Using appropriate method, test whether the respondent’s (i) average age is equal to 30

# years, (ii) average height is greater than to 1.7 meters, (iii) average consumption of water

# daily (CH2O) is equal to 2 liters and (iv) average BMI is less than 30. [Hints on appropriate

# method: among parametric and non-parametric tests, which is applicable in each individual

# question? If parametric, then z-test or t-test? Why?]

# (i) Average Age is Equal to 30 Years

# Appropriate Test: One-sample t-test

# Null Hypothesis (H0): Average age is equal to 30 years (μ = 30).

# Alternative Hypothesis (H1): Average age is not equal to 30 years (μ ≠ 30).

# One-sample t-test for average age

t\_test\_age <- t.test(obes\_data$Age, mu = 30)

t\_test\_age

# (ii) Average Height is Greater than 1.7 Meters

# Appropriate Test: One-sample t-test

# Null Hypothesis (H0): Average height is equal to or less than 1.7 meters (μ ≤ 1.7).

# Alternative Hypothesis (H1): Average height is greater than 1.7 meters (μ > 1.7).

#mone rakhich sob somoy, greater/less diye kono condition dile egola alternative hypo te jabe

# One-sample t-test for average height

t\_test\_height <- t.test(obes\_data$Height, mu = 1.7, alternative = "greater")

t\_test\_height

# (iii) Average Consumption of Water Daily (CH2O) is Equal to 2 Liters

# Appropriate Test: One-sample t-test

# Null Hypothesis (H0): Average CH2O is equal to 2 liters (μ = 2).

# Alternative Hypothesis (H1): Average CH2O is not equal to 2 liters (μ ≠ 2).

# One-sample t-test for average CH2O

t\_test\_ch2o <- t.test(obes\_data$CH2O, mu = 2)

print(t\_test\_ch2o)

# (iv) Average BMI is Less than 30

# Appropriate Test: One-sample t-test

# Null Hypothesis (H0): Average BMI is equal to or greater than 30 (μ ≥ 30).

# Alternative Hypothesis (H1): Average BMI is less than 30 (μ < 30).

# One-sample t-test for average BMI

t\_test\_bmi <- t.test(obes\_data$BMI, mu = 30, alternative = "less")

t\_test\_bmi

# #Q16 Using appropriate method, test whether the average age, height and BMI of the respondents

# significantly differ between (i) male and female, and (ii) smoker and non-smoker. Also,

# test whether the BMI value is higher for the respondents with family history of overweight

# and lower for those who monitor their calorie consumption. [Hints on appropriate method:

# among parametric and non-parametric tests, which is applicable in each individual

# question? If parametric, then z-test or t-test? Why?]

#question na bojle bolich, eta type kore bolla ta tough hbe

# (i) Difference in Average Age, Height, and BMI Between Male and Female

# Appropriate Test: Independent samples t-test (assuming normal distribution of data)

# Null Hypothesis (H0): There is no difference in average age/height/BMI between male and female.

# Alternative Hypothesis (H1): There is a difference in average age/height/BMI between male and female.

# Independent samples t-test for average age between male and female

t\_test\_age\_gender <- t.test(Age ~ Gender, data = obes\_data)

t\_test\_age\_gender

# Independent samples t-test for average height between male and female

t\_test\_height\_gender <- t.test(Height ~ Gender, data = obes\_data)

t\_test\_height\_gender

# Independent samples t-test for average BMI between male and female

t\_test\_bmi\_gender <- t.test(BMI ~ Gender, data = obes\_data)

t\_test\_bmi\_gender

# (ii) Difference in Average Age, Height, and BMI Between Smoker and Non-Smoker

# Appropriate Test: Independent samples t-test (assuming normal distribution of data)

# Null Hypothesis (H0): There is no difference in average age/height/BMI between smokers and non-smokers.

# Alternative Hypothesis (H1): There is a difference in average age/height/BMI between smokers and non-smokers.

# Independent samples t-test for average age between smokers and non-smokers

t\_test\_age\_smoke <- t.test(Age ~ SMOKE, data = obes\_data)

t\_test\_age\_smoke

# Independent samples t-test for average height between smokers and non-smokers

t\_test\_height\_smoke <- t.test(Height ~ SMOKE, data = obes\_data)

t\_test\_height\_smoke

# Independent samples t-test for average BMI between smokers and non-smokers

t\_test\_bmi\_smoke <- t.test(BMI ~ SMOKE, data = obes\_data)

t\_test\_bmi\_smoke

# (iii) BMI Differences for Family History and Caloric Consumption Monitoring (considering "yes" only with SCC variable):

# Appropriate Test: Independent samples t-test or Mann-Whitney U test (non-parametric) depending on the normality of the data.

# Null Hypothesis (H0): There is no difference in average BMI for respondents with and without family history of overweight and for those who monitor and do not monitor their caloric consumption (considering only "yes" cases).

# Alternative Hypothesis (H1): There is a significant difference in average BMI.

# Filter the data for respondents with caloric consumption monitoring equal to "yes"

obes\_data\_yes\_scc <- subset(obes\_data, SCC == "yes")

obes\_data\_yes\_scc

# Independent samples t-test for BMI between family history groups (considering only "yes" cases)

t\_test\_family\_history\_yes\_scc <- t.test(BMI ~ Family\_history\_with\_overweight, data = obes\_data\_yes\_scc)

t\_test\_family\_history\_yes\_scc

# 17. Using appropriate method, test whether there are significant variations in the average age,

# height and BMI of the respondents among different groups of CAEC, CALC and

# MTRANS. Also, identify which pairs of the group means of each variable significantly

# differ (perform multiple comparison). [Hints on appropriate method: among parametric

# and non-parametric tests, which is applicable in each individual question and why?]

#1st a toke decide korte hbe toi parametric test korbi naki non parametric korbi

# r voy paich na ekhane same concept bar bar apply korteche for different variable er jnno

# Non-Parametric Test (Kruskal-Wallis):

# If the assumptions for ANOVA are violated or the data are not normally distributed:

# Kruskal-Wallis test for Age and CAEC

kruskal\_age\_caec <- kruskal.test(Age ~ CAEC, data = obes\_data)

kruskal\_age\_caec

# Kruskal-Wallis test for Height and CAEC

kruskal\_height\_caec <- kruskal.test(Height ~ CAEC, data = obes\_data)

kruskal\_height\_caec

# Kruskal-Wallis test for BMI and CAEC

kruskal\_bmi\_caec <- kruskal.test(BMI ~ CAEC, data = obes\_data)

kruskal\_bmi\_caec

# Kruskal-Wallis test for Age and CAEC

kruskal\_age\_caec <- kruskal.test(Age ~ CAEC, data = obes\_data)

kruskal\_age\_caec

# Kruskal-Wallis test for Height and CAEC

kruskal\_height\_caec <- kruskal.test(Height ~ CAEC, data = obes\_data)

kruskal\_height\_caec

# Kruskal-Wallis test for BMI and CAEC

kruskal\_bmi\_caec <- kruskal.test(BMI ~ CAEC, data = obes\_data)

kruskal\_bmi\_caec

# Kruskal-Wallis test for Age and MTRANS

kruskal\_age\_mtrans <- kruskal.test(Age ~ MTRANS, data = obes\_data)

kruskal\_age\_mtrans

# Kruskal-Wallis test for Height and MTRANS

kruskal\_height\_mtrans <- kruskal.test(Height ~ MTRANS, data = obes\_data)

kruskal\_height\_mtrans

# Kruskal-Wallis test for BMI and MTRANS

kruskal\_bmi\_mtrans <- kruskal.test(BMI ~ MTRANS, data = obes\_data)

kruskal\_bmi\_mtrans

# interpretation to interpret the p-values from these tests. If the p-value is less than the significance level (commonly set at 0.05), it suggests that there are significant differences among groups.

# #q18 Using appropriate method, test whether there is significant association between BMI and

# (i) gender, (ii) family history with overweight, (iii) smoking, (iv) FAVC, (v) CAEC, (vi)

# SCC, (vii) CALC and (viii) MTRANS. Report the contingency tables, row/column/total

# percentages and table of expected cell count. [Hints on appropriate method: among

# asymptotic and exact tests, which is applicable in each individual question and why?]

# Assuming you have a dataset named 'my\_data' and variables 'BMI\_cat' and 'Gender'

for (c in names(cate)) {

print(paste("Crosstab of BMI\_cat by", c))

CrossTable(obes\_data$BMI\_cat, obes\_data[[c]],

prop.r = TRUE, prop.c = TRUE, prop.t = TRUE,

prop.chisq = FALSE, fisher = FALSE, expected = TRUE)

}

View(obes\_data)

# #q19 (i) From the Obesity Dataset (ObesityData.csv), identify the significant factors affecting

# BMI. To do this, first, perform the model building/variable selection using suitable method.

# Then, present the final fitted model including estimates of the parameters, their confidence

# intervals, p-values. Comment on your findings. Comment goodness of fit of the final

# model/How well the model fitted with the data (F statistics with its p-value, R2, adjusted

# R2, And AIC Value)? Find the predicted values and the residuals.

# (ii) Perform model adequacy checking (check model assumptions) and examine whether

# each of the model assumptions is satisfied or not. Give justification of your answer.

# Propose alternative solution if any model assumption is not satisfied.

# (iii) Find if there is any outlier in the data. Also, discuss whether the outliers (if any) are

# influential or not.

# Check column names in the original dataset

colnames(obes\_data) # eta onek important, obes\_table a j j vabe varible gola ache sebae input dite hbe

# Create obesity\_small dataset

obesity\_small <- obes\_data[, c("Gender", "Age", "Height", "Weight", "family\_history\_with\_overweight", "FAVC", "CH2O", "SCC", "FAF", "TUE", "CALC")]

# Create obesity\_new dataset by dropping FAF, TUE, and CALC

obesity\_new <- obesity\_small[, c("Gender", "Age", "Height", "Weight", "family\_history\_with\_overweight", "FAVC", "CH2O", "SCC")]

# Fit multiple linear regression model

model <- lm(Weight ~ Gender + Age + Height + family\_history\_with\_overweight + FAVC + CH2O + SCC, data = obesity\_new)

# Print summary of the model

summary(model)

# Get multiple R^2

multiple\_r\_squared <- summary(model)$r.squared

print(paste("Multiple R^2:", multiple\_r\_squared))

# Predict weight for a specific respondent

new\_data <- data.frame(

Gender = "Male",

Age = 23,

Height = 1.77,

family\_history\_with\_overweight = "yes",

FAVC = "yes",

CH2O = 1,

SCC = "no"

)

predicted\_weight <- predict(model, newdata = new\_data)

print(paste("Predicted Weight:", predicted\_weight))

#Q20

# Define a threshold for identifying influential outliers

threshold <- 4 / nrow(obesity\_small)

#(i) Model Building and Summary

# Fit the linear model

model <- lm(BMI ~ . , data = data\_mod2)

# Stepwise variable selection

final\_model <- step(model)

# Display final model summary

summary(final\_model)

#(ii) Diagnostics

# Predictions and Residuals

predictions <- predict(final\_model)

residuals <- resid(final\_model)

# Linearity Check

plot(final\_model, which = 1)

# Normality Check

qqnorm(residuals)

qqline(residuals)

shapiro.test(residuals)

# Multicollinearity Check

library(car)

vif(final\_model)

# Autocorrelation Check

DurbinWatsonTest(final\_model)

# Homoscedasticity Check

plot(final\_model, which = 3)

library(lmtest)

bptest(final\_model)

#q21

# Set the seed using the last three digits of your registration number

seed\_number <- 045

set.seed(seed\_number)

# Draw a random sample of size n = 50

sample\_data <- obes\_data[sample(nrow(obes\_data), 50), ]

head(sample\_data)