# anything at the same line after the symbol # will be ignored by R.

# So, we use the space after # to make a comment on/give an explanation for our code.

#Set the working directory by using the following command line with your own folder's path,

#or on the Menu Bar across the top of the screen, by clicking "Session" then "Set Working Directory" then "Choose Directory" to choose a folder.

setwd('/Users/haishu/Desktop/classes/Biostat for Public Health 2020\_Shu/assigments')

#read in the data file;

data=read.csv('Data1.csv',header=TRUE) #header=TRUE or header=T avoids putting the column names as the 1st row of the "data" table.

head(data) # shows the first few rows of the "data" table

#Ignore the 1st columns which are just row indices. You may delete the 1st column by

data=data[,-1]

data[1,] #the 1st row of the "data" table

data[,1] #the 1st column of the "data" table

data[2,3] #the cell at the 2nd row and the 3rd column of the table

data[2:5,4] # the 4 cells at the 2nd to 5th rows and also at the 4th column of the table

data[c(2,4,5),4] #the 3 celles at 2,4,5th rows and also at the 4th columns of the table

data[2:5,3:5] # the 4x3 subtable at the 2nd to 5th rows and the 3rd to 5th columns of the table

data[c(2,4,5),c(1,3,6)] #the 3x3 subtable at the 2,4,5th rows and also at 1,3,6th columns of the table

dim(data) # gives the dimensions (# of rows and columns) of the "data" table

dim(data)[1] # gives the number of rows in the "data" table

dim(data)[2] # gives the number of columns in the "data" table

summary(data)

# Gives min, max, mean, quartiles, and the number of missing values (if applicable) for continuous variables

# gives frequencies for continuous variables

############ Frequency ############

summary(data$RACE)#Frequency

n\_race=length(data$RACE)

n\_race #261

summary(data$RACE)/n\_race#Relative Frequency

data$RACE[1]#API

#what if there is a missing value?

#e.g., we let

data$RACE[1]=NA

data$RACE[1] # now we have changed the RACE column to have a missing value

summary(data$RACE)

n\_race\_noNA=length( na.omit(data$RACE) )

n\_race\_noNA

summary(data$RACE)/n\_race\_noNA #Valid Relative Frequency; ignore the final value for NA's

#put the original value back to data$RACE[1]

data$RACE[1]='API'

summary(data$RACE)

############Central Tendency############

median(data$AGE)

median(data[,1])#since AGE is the 1st column, data[,1] is equivalent to data$AGE

mean(data$LONELINESS) #if there exist missing values, it will gives NA

mean(data$LONELINESS,na.rm=TRUE) #compute the mean after removing missing values

#also try functions: min(), max()

###compute the mode of a variable

table(data$LONELINESS) # will automatically ignore missing values.

#the 1st line of the output shows the variable values

#the 2st line shows the frequencies.

table(data$AGE)

sort( table(data$AGE), decreasing = TRUE ) #To find the mode of AGE

############ Variability ############

range(data$AGE)#try range(data$LONELINESS,na.rm=T) for data$LONELINESS that has missing values

range(data$AGE)[2]-range(data$AGE)[1] #range

var(data$AGE)#variance

sd(data$AGE)#standard deviation

############ Plots ############

### Barplot

table(data$RACE)

barplot( table(data$RACE) )

barplot( table(data$RACE), ylab='Frequency', xlab='Race', main='Barplot of Race' ) # Add labels for axes.

barplot( table(data$AGE), ylab='Frequency', xlab='Age', main='Barplot of Age' )

### Histogram

hist(data$AGE)

hist(data$AGE, xlab='Age', main='Histogram of Age' )

hist(data$DEPRESSION)

hist(data$DEPRESSION,breaks=10)

hist(data$DEPRESSION,breaks=20)

hist(data$DEPRESSION,breaks=50)

hist(data$DEPRESSION,breaks=100)

###subtables

#To extract the subtable for males only

data\_male=data[data$GENDER=='male',] # use "==" not "="

mean(data\_male$TOTALDRUGS)

#To extract the TOTALDRUGS column for males only

data\_male\_totaldrugs=data[data$GENDER=='male','TOTALDRUGS']#replace the 'TOTALDRUGS' by any column name of interest

mean(data\_male\_totaldrugs)

library(car)# which has the leveneTest() function

#Set the working directory by using the following command line with your own folder's path,

#or on the Menu Bar across the top of the screen, by clicking "Session" then "Set Working Directory" then "Choose Directory" to choose a folder.

setwd('/Users/haishu/Desktop/classes/Biostat for Public Health 2020\_Shu/assigments')

#read in the data file;

data=read.csv('Data1.csv',header=TRUE) #header=TRUE or header=T avoids putting the column names as the 1st row of the "data" table.

head(data) # shows the first few rows of the "data" table

#Ignore the 1st columns which are just row indices. You may delete the 1st column by

data=data[,-1]

##################################

#Independent samples t-test

##################################

#####To extract the TOTALDRUGS column for males only

data\_male\_totaldrugs=data$TOTALDRUGS[data$GENDER=='male']

mean(data\_male\_totaldrugs)

var(data\_male\_totaldrugs)#variance

length(data\_male\_totaldrugs)

#hist(data\_male\_totaldrugs)

#####To extract the TOTALDRUGS column for females only

data\_female\_totaldrugs=data$TOTALDRUGS[data$GENDER=='female']

mean(data\_female\_totaldrugs)

var(data\_female\_totaldrugs)

length(data\_female\_totaldrugs)

#hist(data\_female\_totaldrugs)

#####Levene's test for equal variance; leveneTest() from package 'car'

leveneTest(TOTALDRUGS~GENDER, data=data)

#####t-test

t.test(TOTALDRUGS~GENDER, data=data, alternative = "two.sided", var.equal = TRUE)

##################################

#One-way ANOVA

##################################

#test the association between race and totaldrugs

summary(data$RACE)

tapply(data$TOTALDRUGS,data$RACE,mean) # compute the mean for each group

tapply(data$TOTALDRUGS,data$RACE,sd) # compute the sd for each group

#####histogram

#extract the TOTALDRUGS information for RACE=='API'

data\_TOTALDRUGS\_API = data$TOTALDRUGS[data$RACE=='API']

hist(data\_TOTALDRUGS\_API)

#####boxplot

boxplot(TOTALDRUGS~RACE,data=data)

#####Levene's test for equal variance; leveneTest() from package 'car'

leveneTest(TOTALDRUGS~RACE, data=data)

#####One-way ANOVA

fit = aov(TOTALDRUGS~RACE, data=data)

summary(fit)

##################################

#Correlation test

##################################

plot(y=data$TOTALDRUGS, x=data$DEPRESSION, ylab='TOTALDRUGS',xlab='DEPRESSION',main='TOTALDRUGS vs DEPRESSION')

abline(lm(data$TOTALDRUGS~data$DEPRESSION), col="blue")

length(data$TOTALDRUGS)

hist(data$TOTALDRUGS, xlab='TOTALDRUGS', main='Histogram of TOTALDRUGS')

hist(data$DEPRESSION, breaks=30, xlab='DEPRESSION', main='Histogram of DEPRESSION')

cor(data$TOTALDRUGS,data$DEPRESSION)

cor.test(data$TOTALDRUGS,data$DEPRESSION)

summary(data$LONELINESS)

cor(data$TOTALDRUGS,data$LONELINESS,use="complete.obs")#due to missing values

cor.test(data$TOTALDRUGS,data$LONELINESS)

##################################

#Chi-square test

##################################

tbl=table(data$GENDER, data$ALCOHOL) #create the contingency table

tbl

ChisqTest=chisq.test(tbl, correct=FALSE) #conduct the chi-square test

ChisqTest

ChisqTest$expected

#Fisher test

FisherTest = fisher.test(tbl)

FisherTest