aug = c(18.3,13.3,16.5,12.6,9.5,13.6,8.1,8.9,10.0,8.3,7.9,8.1,13.4)

nov = c(12.7,11.1,15.3,12.7,10.5,15.6,11.2,14.2,16.3,15.5,19.9,20.4,36.8)

Diff = aug - nov

#boxplot(Diff)

hist(Diff,

col="gray",

main="Histogram of differences",

xlab="Difference")

#Paired t-test

sort(aug)

sort(nov)

t.test(aug, nov,

paired=TRUE,

conf.level=0.95)

#Wilcoxon signed-rank test

require(packHV)

par(mfrow=c(1,2))

qqnorm(Diff)

qqline(Diff, col='red')

#wilcox.test(aug,nov,alternative="less",paired=TRUE)

wilcox.test(Diff,conf.int=TRUE)

library(BSDA)

SIGN.test(Diff,alternative= "two.sided")

#Permutation test

R <- 99 #number of replicates

z <- c(aug, nov) #pooled sample

K <- 1:26 #n=14, m=12

difmedian <- numeric(R) #storage for replicates

t0 <- median(aug)-median(nov)

p.permutation=c()

pnorm.permutation=c()

n=1

while(n<99){

for (i in 1:R) {

gp<-rep("y",26) # initialize factors

#generate indices k for the first sample

k <- sample(K, size = 13, replace = FALSE)

difmedian[i] <- median(z[k])-median(z[-k]) # difference in medians

}

p.permutation[n]=sum(abs(c(t0, difmedian)) >= t0)/1000

pnorm.permutation[n]=2\*pnorm(-abs((t0-mean(c(difmedian,t0)))/sd(c(difmedian,t0))))

n=n+1

}

hist(c(t0,difmedian),xlab="difference in medians for permutation samples",main="Difference of medians in permutation samples \n with original difference indicated")

points(t0,0, cex = 1, pch = 16)

pval <- 2\*mean(c(t0, difmedian) >= t0)

pval

quantile(pnorm.permutation,c(.025,.975))

#bootstrap

R <- 99 #number of replicates

z <- c(aug, nov) #pooled sample

n = length(aug)

m = length(nov)

#(package boot):

library(boot)

camp=data.frame(group=rep(c(1,2),c(n,m)),z)

dif.median=function(data,i) {

d=data[i,]

n1=n+1

m1=n+m

median(d$y[1:n])-median(d$y[n1:m1]) }

dif.boot=boot(camp,dif.median,R=99, strata=camp$group)

boot.ci(dif.boot, conf =0.95, type="bca")

[1] "All values of t are equal to NaN \n Cannot calculate confidence intervals"

class = c(1,1,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,1,1,0)

FST = c(-0.006,-0.005,-0.005,-0.002,0.003,0.004,0.006,0.015,0.016,0.016,0.024,0.041,0.044,0.049,0.053,0.058,0.066,0.095,0.116,0.163)

data = data.frame(class,FST)

xyz = split(data,data$class)

dna = xyz$'0'

protein = xyz$'1'

hist(dna$FST)

hist(protein$FST)

a = median(dna$FST)

b = median(protein$FST)

diff = a - b

a

b

diff

boxplot(FST~class)

kruskal.test(data$class, data$FST)

x = data$FST

bootmed = apply(matrix(sample(x, rep = TRUE, 10^4\*length(x)), nrow = 10\*4),1,median)

quantile(bootmed,c(0.025,0.975))s

#Question 4

Height = c(63.75,68.25,62.25,67.25,64.75,67.5,64.75,66.5,68.5,64.25,64.5,66,65.25,64.75,67.5)

Age = c("20-29","20-29","20-29","20-29","30-39","30-39","30-39","30-39","40-49","40-49","40-49","40-49","50-59","50-59","50-59")

data = data.frame(Age,Height)

height = split(data,data$Age)

one = height$'20-29'

library(lattice)

H = sort(Height)

histogram(~H|Age,layout=c(1,4),data=data)

boxplot(Height~Age, ylab="Height",

xlab="Age", main = "Height grouped by Age", col = c("Red","Green","Blue","Yellow"))

qqnorm(data$Height,ylab="Height",

xlab="Normal",

main="Normal Probability plot"

)

qqline(data$Height)

kruskal.test(Height ~ Age, data)

res.aov2 = aov(Height ~ Age,data = data)

summary(res.aov2)

TukeyHSD(res.aov2)

Kruskal-Wallis rank sum test

data: Height by Age

Kruskal-Wallis chi-squared = 0.39835, df = 3, p-value = 0.9406

Df Sum Sq Mean Sq F value Pr(>F)

Age 3 0.64 0.215 0.052 0.984

Residuals 11 45.46 4.133

Tukey multiple comparisons of means

95% family-wise confidence level

Fit: aov(formula = Height ~ Age, data = data)

$Age

diff lwr upr p adj

30-39-20-29 0.50000000 -3.826349 4.826349 0.9847774

40-49-20-29 0.43750000 -3.888849 4.763849 0.9896658

50-59-20-29 0.45833333 -4.214658 5.131324 0.9905467

40-49-30-39 -0.06250000 -4.388849 4.263849 0.9999685

50-59-30-39 -0.04166667 -4.714658 4.631324 0.9999926

50-59-40-49 0.02083333 -4.652158 4.693824 0.9999991

**Null hypothesis:** Null hypothesis assumes that the Height (groups) are from identical populations.

#question 7

x1 = rt(1000, df = 10)

x2 = rt(1000, df = 10)

ks.test(x1,x2,"pnorm",0, 1)

Two-sample Kolmogorov-Smirnov test

data: x1 and x2

D = 0.042, p-value = 0.341

alternative hypothesis: two-sided

20

Two-sample Kolmogorov-Smirnov test

data: x1 and x2

D = 0.039, p-value = 0.4324

alternative hypothesis: two-sided

100

Two-sample Kolmogorov-Smirnov test

data: x1 and x2

D = 0.058, p-value = 0.06919

alternative hypothesis: two-sided

1000

Two-sample Kolmogorov-Smirnov test

data: x1 and x2

D = 0.046, p-value = 0.2406

alternative hypothesis: two-sided

set.seed(96)

x1 = rt(10, df = 40)

x2 = rt(10, df = 40)

ks.test(x1,x2,"pnorm",0, 1)

Two-sample Kolmogorov-Smirnov test

data: x1 and x2

D = 0.3, p-value = 0.7869

alternative hypothesis: two-sided