Lab12

Clayton Glenn

April 13, 2018

Table of Contents

# Task1

getwd()

## [1] "C:/Users/cglen/Desktop/Stat Methods/Labs/LAB12"

# Task2

## Set Data

set.seed(55);x1=rnorm(30,mean=25,sd=5)

## TTests

t.test(x1,mu=22)

##   
## One Sample t-test  
##   
## data: x1  
## t = 3.3863, df = 29, p-value = 0.002052  
## alternative hypothesis: true mean is not equal to 22  
## 95 percent confidence interval:  
## 23.30198 27.27320  
## sample estimates:  
## mean of x   
## 25.28759

t.test(x1,mu=23)

##   
## One Sample t-test  
##   
## data: x1  
## t = 2.3563, df = 29, p-value = 0.02543  
## alternative hypothesis: true mean is not equal to 23  
## 95 percent confidence interval:  
## 23.30198 27.27320  
## sample estimates:  
## mean of x   
## 25.28759

t.test(x1,mu=24)

##   
## One Sample t-test  
##   
## data: x1  
## t = 1.3263, df = 29, p-value = 0.1951  
## alternative hypothesis: true mean is not equal to 24  
## 95 percent confidence interval:  
## 23.30198 27.27320  
## sample estimates:  
## mean of x   
## 25.28759

t.test(x1,mu=25)

##   
## One Sample t-test  
##   
## data: x1  
## t = 0.29623, df = 29, p-value = 0.7692  
## alternative hypothesis: true mean is not equal to 25  
## 95 percent confidence interval:  
## 23.30198 27.27320  
## sample estimates:  
## mean of x   
## 25.28759

t.test(x1,mu=26)

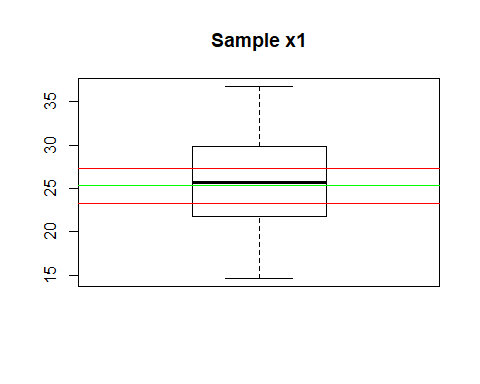
##   
## One Sample t-test  
##   
## data: x1  
## t = -0.7338, df = 29, p-value = 0.469  
## alternative hypothesis: true mean is not equal to 26  
## 95 percent confidence interval:  
## 23.30198 27.27320  
## sample estimates:  
## mean of x   
## 25.28759

## Boxplot with Confidence intervals and Mean

boxplot(x1, main="Sample x1")  
ci=t.test(x1,mu=22)$conf.int  
ci

## [1] 23.30198 27.27320  
## attr(,"conf.level")  
## [1] 0.95

abline(h=c(ci,mean(x1)),col=c("Red","Red","Green"))



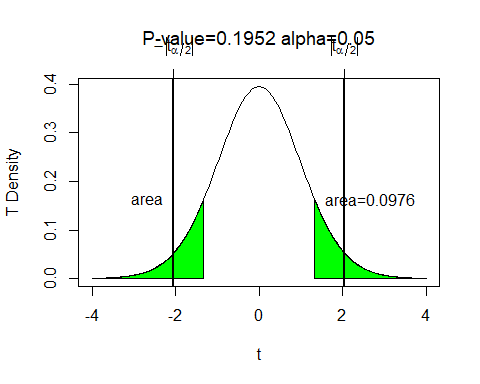
## TCalc

tcalc=(mean(x1)-24)/(sd(x1)/sqrt(30))  
tcalc

## [1] 1.326252

## P-Value Function

mypvalue=function(t0,xmax=4,n=20, alpha=0.05){  
   
 va=round(pt(-t0,df=n-1),4)  
 pv=2\*va  
   
 curve(dt(x,df=n-1),xlim=c(-xmax,xmax),ylab="T Density",xlab=expression(t),  
 main=substitute(paste("P-value=", pv, " alpha=", alpha)))  
   
 xcurve=seq(t0,xmax,length=1000)  
 ycurve=dt(xcurve,df=n-1)  
   
 xlcurve=seq(-t0,-xmax,length=1000)  
 ylcurve=dt(xcurve,df=n-1)  
   
 polygon(c(t0,xcurve,xmax),c(0,ycurve,0),col="green")  
 polygon(c(-t0,xlcurve,-xmax),c(0,ylcurve,0),col="green")  
   
 q=qt(1-alpha/2,n-1)  
 abline( v=c(q,-q),lwd=2)   
 axis(3,c(q,-q),c(expression(abs(t[alpha/2])),expression(-abs(t[alpha/2]))))  
   
 text(0.5\*(t0+xmax),max(ycurve),substitute(paste(area, "=",va)))  
 text(-0.5\*(t0+xmax),max(ycurve),expression(area))  
   
 return(list(q=q,pvalue=pv))  
}  
mypvalue(tcalc,n=30,alpha=0.05)

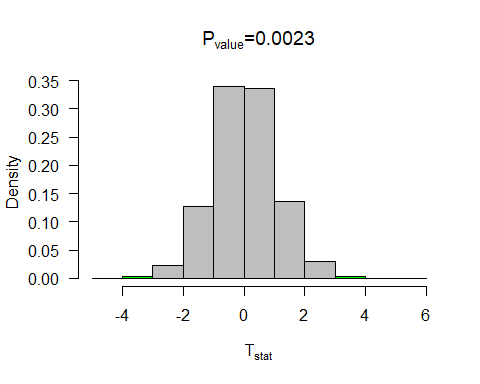


## $q  
## [1] 2.04523  
##   
## $pvalue  
## [1] 0.1952

P-Values rejection region is between -2,2 and the pvalue should be greater than 0.05 for the null hypothesis to be accepted.

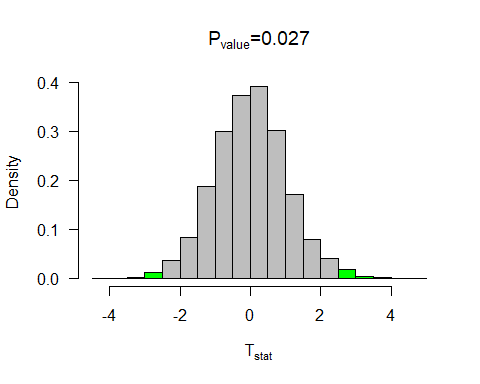
## Bootstrap Function

bootpval<-function(x,conf.level=0.95,iter=3000,mu0=0, test="two"){  
 n=length(x)  
 y=x-mean(x)+mu0   
 rs.mat<-c()   
 xrs.mat<-c()  
 for(i in 1:iter){   
 rs.mat<-cbind(rs.mat,sample(y,n,replace=TRUE))   
 xrs.mat<-cbind(xrs.mat,sample(x,n,replace=TRUE))   
   
 }  
   
 tstat<-function(z){   
 sqrt(n)\*(mean(z)-mu0)/sd(z)  
 }  
   
 tcalc=tstat(x)   
 ytstat=apply(rs.mat,2,tstat)   
 xstat=apply(xrs.mat,2,mean)   
 alpha=1-conf.level   
 ci=quantile(xstat,c(alpha/2,1-alpha/2))  
 pvalue=ifelse(test=="two",length(ytstat[ytstat>abs(tcalc) | ytstat < -abs(tcalc)])/iter,  
 ifelse(test=="upper",length(ytstat[ytstat>tcalc])/iter,  
 length(ytstat[ytstat<xstat])/iter))  
   
 h=hist(ytstat,plot=FALSE)  
 mid=h$mid  
 if(test=="two"){  
 ncoll=length(mid[mid<= -abs(tcalc)])  
 ncolr=length(mid[mid>= abs(tcalc)])  
 col=c(rep("Green",ncoll),rep("Gray",length(mid)-ncoll-ncolr),rep("Green",ncolr))  
 }  
 if(test=="upper"){  
 ncolr=length(mid[mid>= abs(tcalc)])  
 col=c(rep("Gray",length(mid)-ncolr),rep("Green",ncolr))  
 }  
   
 if(test=="lower"){  
 ncoll=length(mid[mid<= -abs(tcalc)])  
 col=c(rep("Green",ncoll),rep("Gray",length(mid)-ncoll))  
 }  
 hist(ytstat,col=col,freq=FALSE,las=1,main="",xlab=expression(T[stat]))  
 segments(ci[1],0,ci[2],0,lwd=2)  
 pround=round(pvalue,4)  
 title(substitute(paste(P[value],"=",pround)))  
 return(list(pvalue=pvalue,tcalc=tcalc,n=n,x=x,test=test,ci=ci))  
}  
bootpval(x=x1,mu0=22,test="two")



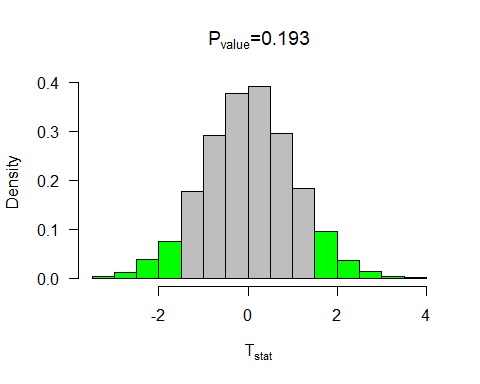
## $pvalue  
## [1] 0.002333333  
##   
## $tcalc  
## [1] 3.386302  
##   
## $n  
## [1] 30  
##   
## $x  
## [1] 25.60070 15.93812 25.75791 19.40389 25.00954 30.94259 22.47328  
## [8] 24.50383 26.52677 25.99205 24.75545 20.78383 14.62365 23.19618  
## [15] 21.81155 23.16861 36.77682 30.46689 26.42921 29.96829 17.40364  
## [22] 32.48559 29.09808 30.33025 28.66878 29.80172 21.53910 32.02999  
## [29] 16.83231 26.30915  
##   
## $test  
## [1] "two"  
##   
## $ci  
## 2.5% 97.5%   
## 23.36578 27.13615

bootpval(x=x1,mu0=23,test="two")



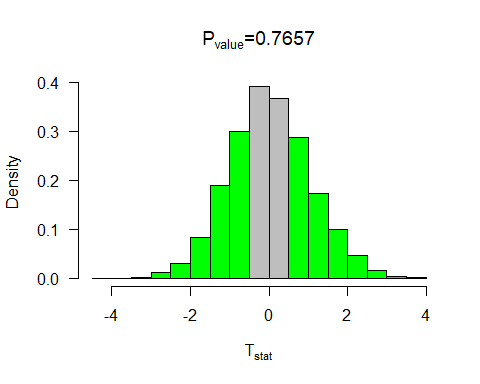
## $pvalue  
## [1] 0.027  
##   
## $tcalc  
## [1] 2.356277  
##   
## $n  
## [1] 30  
##   
## $x  
## [1] 25.60070 15.93812 25.75791 19.40389 25.00954 30.94259 22.47328  
## [8] 24.50383 26.52677 25.99205 24.75545 20.78383 14.62365 23.19618  
## [15] 21.81155 23.16861 36.77682 30.46689 26.42921 29.96829 17.40364  
## [22] 32.48559 29.09808 30.33025 28.66878 29.80172 21.53910 32.02999  
## [29] 16.83231 26.30915  
##   
## $test  
## [1] "two"  
##   
## $ci  
## 2.5% 97.5%   
## 23.35295 27.15016

bootpval(x=x1,mu0=24,test="two")



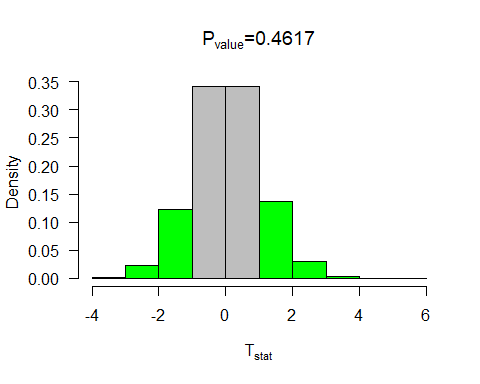
## $pvalue  
## [1] 0.193  
##   
## $tcalc  
## [1] 1.326252  
##   
## $n  
## [1] 30  
##   
## $x  
## [1] 25.60070 15.93812 25.75791 19.40389 25.00954 30.94259 22.47328  
## [8] 24.50383 26.52677 25.99205 24.75545 20.78383 14.62365 23.19618  
## [15] 21.81155 23.16861 36.77682 30.46689 26.42921 29.96829 17.40364  
## [22] 32.48559 29.09808 30.33025 28.66878 29.80172 21.53910 32.02999  
## [29] 16.83231 26.30915  
##   
## $test  
## [1] "two"  
##   
## $ci  
## 2.5% 97.5%   
## 23.42505 27.10562

bootpval(x=x1,mu0=25,test="two")



## $pvalue  
## [1] 0.7656667  
##   
## $tcalc  
## [1] 0.2962266  
##   
## $n  
## [1] 30  
##   
## $x  
## [1] 25.60070 15.93812 25.75791 19.40389 25.00954 30.94259 22.47328  
## [8] 24.50383 26.52677 25.99205 24.75545 20.78383 14.62365 23.19618  
## [15] 21.81155 23.16861 36.77682 30.46689 26.42921 29.96829 17.40364  
## [22] 32.48559 29.09808 30.33025 28.66878 29.80172 21.53910 32.02999  
## [29] 16.83231 26.30915  
##   
## $test  
## [1] "two"  
##   
## $ci  
## 2.5% 97.5%   
## 23.41679 27.24529

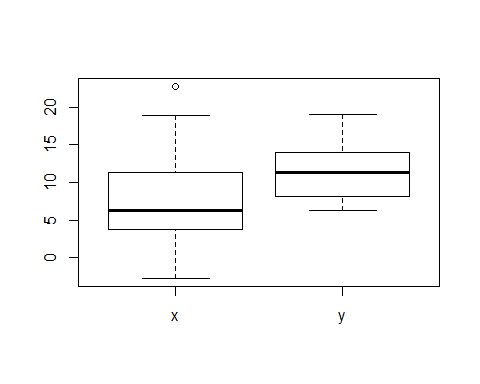
bootpval(x=x1,mu0=26,test="two")



## $pvalue  
## [1] 0.4616667  
##   
## $tcalc  
## [1] -0.7337985  
##   
## $n  
## [1] 30  
##   
## $x  
## [1] 25.60070 15.93812 25.75791 19.40389 25.00954 30.94259 22.47328  
## [8] 24.50383 26.52677 25.99205 24.75545 20.78383 14.62365 23.19618  
## [15] 21.81155 23.16861 36.77682 30.46689 26.42921 29.96829 17.40364  
## [22] 32.48559 29.09808 30.33025 28.66878 29.80172 21.53910 32.02999  
## [29] 16.83231 26.30915  
##   
## $test  
## [1] "two"  
##   
## $ci  
## 2.5% 97.5%   
## 23.44446 27.21564

# Task3

set.seed(30);x=rnorm(15,mean=10,sd=7)   
set.seed(40);y=rnorm(20,mean=12,sd=4)  
  
boxplot(list(x=x,y=y))



var.test(x,y)

##   
## F test to compare two variances  
##   
## data: x and y  
## F = 3.3363, num df = 14, denom df = 19, p-value = 0.01596  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 1.260446 9.544255  
## sample estimates:  
## ratio of variances   
## 3.33631

t.test(x,y, mu=0,var.equal=FALSE)

##   
## Welch Two Sample t-test  
##   
## data: x and y  
## t = -1.7945, df = 20.248, p-value = 0.08768  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -7.9632156 0.5950525  
## sample estimates:  
## mean of x mean of y   
## 7.990343 11.674425

t.test(x,y, mu=2,var.equal=FALSE)

##   
## Welch Two Sample t-test  
##   
## data: x and y  
## t = -2.7687, df = 20.248, p-value = 0.01176  
## alternative hypothesis: true difference in means is not equal to 2  
## 95 percent confidence interval:  
## -7.9632156 0.5950525  
## sample estimates:  
## mean of x mean of y   
## 7.990343 11.674425

According to var test, we should show the variances unequal with the p value. In Summary, as mean1 - mean2 = 0, we do not have sufficient evidence to reject the null, but as mean1-mean2=2, we can reject the null hypothesis. This seems hand in hand since estimated mean1 < estimated mean2 and they vary significantly.

# Task4

set.seed(30);x=rnorm(15,mean=10,sd=4)   
set.seed(40);y=rnorm(20,mean=12,sd=4)  
  
var.test(x,y)

##   
## F test to compare two variances  
##   
## data: x and y  
## F = 1.0894, num df = 14, denom df = 19, p-value = 0.8454  
## alternative hypothesis: true ratio of variances is not equal to 1  
## 95 percent confidence interval:  
## 0.4115743 3.1164915  
## sample estimates:  
## ratio of variances   
## 1.089407

t.test(x,y, mu=0,var.equal=FALSE)

##   
## Welch Two Sample t-test  
##   
## data: x and y  
## t = -2.0494, df = 29.581, p-value = 0.04939  
## alternative hypothesis: true difference in means is not equal to 0  
## 95 percent confidence interval:  
## -5.637517565 -0.008082348  
## sample estimates:  
## mean of x mean of y   
## 8.851625 11.674425

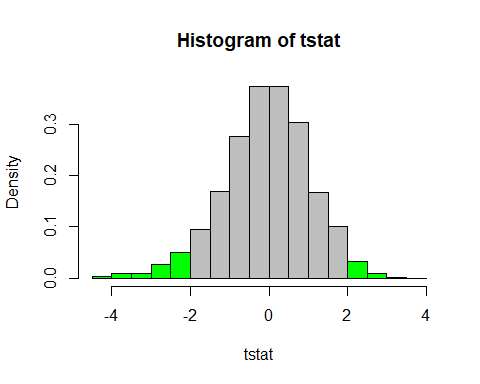
t.test(x,y, mu=2,var.equal=FALSE)

##   
## Welch Two Sample t-test  
##   
## data: x and y  
## t = -3.5014, df = 29.581, p-value = 0.001491  
## alternative hypothesis: true difference in means is not equal to 2  
## 95 percent confidence interval:  
## -5.637517565 -0.008082348  
## sample estimates:  
## mean of x mean of y   
## 8.851625 11.674425

According to var test, we should show the variances equal with the p value. In Summary, as mean1 - mean2 = 0 and as mean1-mean2=2, we can reject the null hypothesis. This seems hand in hand since estimated means are closer quartiles.

# Task5

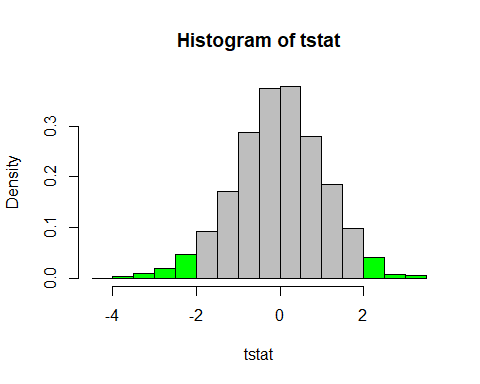
boot2pval<-function(x1,x2,conf.level=0.95,iter=3000,mudiff=0, test="two"){  
 n1=length(x1)  
 n2=length(x2)  
 y1=x1-mean(x1)+mean(c(x1,x2)) # transform the data so that it is centered at the NULL  
 y2=x2-mean(x2)+mean(c(x1,x2))  
 y1rs.mat<-c() #rs.mat will be come a resample matrix -- now it is an empty vector  
 x1rs.mat<-c()  
 y2rs.mat<-c()  
 x2rs.mat<-c()  
 for(i in 1:iter){ # for loop - the loop will go around iter times  
 y1rs.mat<-cbind(y1rs.mat,sample(y1,n1,replace=TRUE)) #sampling from y cbind -- column bind -- binds the vectors together by columns  
 y2rs.mat<-cbind(y2rs.mat,sample(y2,n2,replace=TRUE))  
   
 }  
 x1rs.mat<-y1rs.mat+mean(x1)-mean(c(x1,x2))  
 x2rs.mat<-y2rs.mat+mean(x2)-mean(c(x1,x2))  
   
 xbar1=mean(x1)  
 xbar2=mean(x2)  
 sx1sq=var(x1)  
 sx2sq=var(x2)  
   
 tcalc=(xbar1-xbar2-mudiff)/sqrt(sx1sq/n1+sx2sq/n2)  
   
 sy1sq=apply(y1rs.mat,2,var)  
 sy2sq=apply(y2rs.mat,2,var)   
 y1bar=apply(y1rs.mat,2,mean)  
 y2bar=apply(y2rs.mat,2,mean)  
   
 tstat=(y1bar-y2bar-mudiff)/sqrt(sy1sq/n1+sy2sq/n2)  
   
   
 alpha=1-conf.level  
 pvalue=ifelse(test=="two",length(tstat[tstat>abs(tcalc) | tstat < -abs(tcalc)])/iter,  
 ifelse(test=="upper",length(tstat[tstat>tcalc])/iter,  
 length(ytstat[tstat<tcalc])/iter))  
   
 h=hist(tstat,plot=FALSE)  
 mid=h$mid  
 if(test=="two"){  
 ncoll=length(mid[mid<= -abs(tcalc)])  
 ncolr=length(mid[mid>= abs(tcalc)])  
 col=c(rep("Green",ncoll),rep("Gray",length(mid)-ncoll-ncolr),rep("Green",ncolr))  
 }  
 hist(tstat,col=col,freq=FALSE)  
  
 return(list(pvalue=pvalue))  
}  
set.seed(30);x=rnorm(15,mean=10,sd=7)   
set.seed(40);y=rnorm(20,mean=12,sd=4)  
boot2pval(x1=x,x2=y)



## $pvalue  
## [1] 0.104

# Task6

set.seed(30);x=rnorm(15,mean=10,sd=4)   
set.seed(40);y=rnorm(20,mean=12,sd=4)  
boot2pval(x1=x,x2=y)



## $pvalue  
## [1] 0.06033333

# Task7

## A

Line A is a t test with one sample with the null hypothesis as 23.

## B

Line B shows that the t test is only associated with 1 set of data

## C

Line C Shows the T calc, the degrees of freedom and the p value of the data with the null hypothesis. P-Value getting to small, kick H-not out the door.

## D

Line D shows the functions interpretation of the P-Value with the mean.

## E

The t test is over a 95% confidence interval that shows we are 95% confident that the mean lies within this region.

## F

Line F shows the lower and upper quartiles that the mean should 95% certainly lie between.

## G

Line G shows according to the t test, the mean of the data should be estimated around 25.28759.