```
#Non-Parametric Two Sample Test

#Ho: The average radius for both the groups Malignant and Benign is same, there is no different in cancer

cancer <- read.csv("C:/UCONN Classes/Statistics in Business Analytics/Project/cancer.csv")

Cancer= cancer

View(Cancer)

g= Cancer$radius_se

Cancer1 = Cancer[Cancer$diagnosis=="M",]

g1= Cancer1$radius_se

View(g1)

Cancer2 = Cancer[Cancer$diagnosis=="B",]

g2= Cancer2$radius_se

View(g2)


tstat = abs(mean(g1)-mean(g2))

tstat

#shuffle randomly vector g

sample(g)


length(g)

f1 = function()

{

 s= sample(g)

 length(s)

 v1= mean(s[1:284])

 v2=mean(s[284:569])

 return(abs(v1-v2))


}
```

```
f1()

sdist = replicate(10000,f1())
plot(density(sdist),xlim=c(0,0.35))
polygon(density(sdist),col="green",xlim=c(0,0.35))
abline(v=tstat,lwd=2)
rset = sdist[sdist>=tstat]
p_value = length(rset)/length(sdist)
p_value

#Test to evaluate the average radius for Benign is greater than malignant
t.test(g1,g2)

#Quantile Test
data=cancer
View(data)

## 75th quantile value for radius_mean ##

#Computation for Malignant cancer types

s1=data[data$diagnosis=="M",3]
View(s1)
t1=s1
View(t1)
quantile(t1,probs = 0.75)

#Output
# quantile(t1,probs = 0.75)
```

```
# 75%
# 19.59
#=>75% of the time the mean radius is around 19.59 when the cancer is malignant.

#Computation for Benign cancer types

s2=data[data$diagnosis=="B",3]
View(s2)
t2=s2
View(t2)
quantile(t2,probs = 0.75)

#Output
# quantile(t2,probs = 0.75)
# 75%
# 13.37
#=>75% of the time the mean radius is around 13.37 when the cancer is benign.

#By comparing the mean radius for malignant and bening cancer nuclei, we can conclude that
#malignant cancer nuclei have larger radius on an average.


#75th quantile for smoothness

#Computation for Malignant cancer types

s3=data[data$diagnosis=="M",7]
View(s3)
t3=s3
```

```
View(t3)
quantile(t3,probs = 0.75)

#Output
# quantile(t3,probs = 0.75)
# 75%
# 0.110925
#=>75% of the time the mean smoothness is around 0.110925 when the cancer is malignant.

#Computation for Benign cancer types

s4=data[data$diagnosis=="B",7]
View(s4)
t4=s4
View(t4)
quantile(t4,probs = 0.75)

#Output
# quantile(t4,probs = 0.75)
# 75%
# 0.1007
#=>75% of the time the mean smoothness is around 0.1007 when the cancer is benign.

#By comparing the mean smoothness for malignant and bening cancer nuclei, we can conclude
#that malignant cancer nuclei are smoother on an average.

#Chi Square test
#H0: Any radius mean over 14 is malignent and below is benign
```

```
p_diag = prop.table(table(data$diagnosis))
p_rad = prop.table(table(data$X))

p_diag
p_rad
p = p_diag %*% t(p_rad)
p
n = nrow(data)
E = p*n
E
O = table(data$diagnosis,data$X)
O
tstat = sum((O-E)^2/E)
tstat

f1 = function()
{
  s1 = sample(x = c('M',"B"),size = n,replace = T,prob = p_diag)
  s2 = sample(x = c("0","1"),size = n,replace = T,prob = p_rad)
  O = table(s1,s2)
  return(sum((O-E)^2/E))
}
f1()
sdist = replicate(10000,f1())
plot(density(sdist))
polygon(density(sdist),col="green")
abline(v=tstat,lwd=2)
rset = sdist[sdist>=tstat]
p_value = length(rset)/length(sdist)
```

```
p_value
#.0000000004 reject null hypothesis, the point of this test is to highlight that we cannot
#simply create cut off points to say cancer or no cancer so we need to do a logistic regressionto predict



library(bbmle)
tmp <- cor(data)
tmp[upper.tri(tmp)] <- 0
diag(tmp) <- 0
CancerNew <- Cancer[,!apply(tmp,2,function(x) any(x > 0.90))]

plot(data$diagnosis)
plot(data$diagnosis,data$radius_mean)
plot(data$diagnosis,data$texture_mean)
plot(data$diagnosis,data$smoothness_mean)
plot(data$diagnosis,data$compactness_mean)



#Regression
cancer = cancer[-c(1,2)]
View(cancer)
head(cancer)
install.packages("bbmle")
library(bbmle)
tmp <- cor(cancer)
tmp[upper.tri(tmp)] <- 0
diag(tmp) <- 0
CancerNew <- cancer[,!apply(tmp,2,function(x) any(x > 0.90))]
cancer = CancerNew
```

```
length(cancer)


cor(cancer)

# Reduce highly related vatibles

cancer[,c("compactness_mean","area_worst","compactness_se","concavity_se","concave
points_se","compactness_worst","concave points_worst","fractal_dimension_se")] <-NULL

View(cancer)


#reg1 AIC: 92.902

reg1 = glm(diagnosis_b~.,family = "binomial",data=cancer)

summary(reg1)

coef(reg1)

exp(coef(reg1))

res = step(reg1,~.^2)

res$anova


#reg2 AIC: 88.891

reg2 = glm(diagnosis_b ~.+ smoothness_mean:texture_se -
smoothness_worst,data=cancer,family="binomial")

summary(reg2)


library(MASS)

step <- stepAIC(reg2, direction="both")

step$anova


#reg3 AIC: 85.7

reg3 = glm(diagnosis_b ~ smoothness_mean + texture_se + area_se + smoothness_se +

        concave.points_se + texture_worst + concavity_worst + concave.points_worst +

        symmetry_worst + fractal_dimension_worst +
smoothness_mean:texture_se,data=cancer,family="binomial")
```

```
summary(reg3)


AIC(reg1,reg2,reg3)
sort(exp(coef(reg3)))


install.packages("regclass")
library(regclass)
confusion_matrix(reg3)
# Accuracy(reg3)  = (353+206)/569 = 0.9824


#Maximum likelihood Estimation
cancer <- read.csv("~/Desktop/Rstudio/Project/cancer.csv")
cancer = cancer[-c(1,2)]
View(cancer)
head(cancer)
install.packages("bbmle")
library(bbmle)
tmp <- cor(cancer)
tmp[upper.tri(tmp)] <- 0
diag(tmp) <- 0
CancerNew <- cancer[,!apply(tmp,2,function(x) any(x > 0.90))]
cancer = CancerNew
length(cancer)


f1 = function(b0,b1,b2,b3,b4,b5,b6,b7,b8,b9,b10,b11,b12,b13,b14,b15,b16,b17,b18,b19,b20)
{
  X =
b0+b1*cancer$smoothness_mean+b2*cancer$compactness_mean+b3*cancer$symmetry_mean+b4*ca
ncer$fractal_dimension_mean+b5*cancer$texture_se+b6*cancer$area_se+b7*cancer$smoothness_se+
b8*cancer$compactness_se+b9*cancer$concavity_se+b10*cancer$concave.points_se+b11*cancer$sym
```

```
metry_se+b12*cancer$fractal_dimension_se+b13*cancer$texture_worst+b14*cancer$area_worst+b15
*cancer$smoothness_worst+b16*cancer$compactness_worst+b17*cancer$concavity_worst+b18*canc
er$concave.points_worst+b19*cancer$symmetry_worst+b20*cancer$fractal_dimension_worst

  p = exp(X)/(1+exp(X))

  L = ifelse(cancer$diagnosis_b==1,p,1-p)

  LL = sum(log(L))

  return(-1*LL)

}

res = mle2(minuslogl =
f1,start=list(b0=0,b1=0,b2=0,b3=0,b4=0,b5=0,b6=0,b7=0,b8=0,b9=0,b10=0,b11=0,b12=0,b13=0,b14=0,b
15=0,b16=0,b17=0,b18=0,b19=0,b20=0))

summary(res)

AIC(res)
```