Computer Practical 3 : Solutions by Harleen Gulati. Student Number : 2101550

Q1 Pre-requisite : Loading data and replacing missing values

```
# load data
diabetes<-read.csv("diabetes_data.csv",header=T)

# replace missing values
missing<-function(var){
  med<-median(var[var>0])
  var[var==0]<-med
  return(var)
}
diabetes$Glucose<-missing(diabetes$Glucose)
diabetes$BloodPressure<-missing(diabetes$BloodPressure)
diabetes$Insulin<-missing(diabetes$Insulin)
diabetes$SkinThickness<-missing(diabetes$SkinThickness)
diabetes$BMI<-missing(diabetes$BMI)
```

Question 1.[2 marks] Use the intervals

$$(-\infty, 45],\ (45, 55],\ (55, 65],\ (65, 75],\ (75, 85],\ (85, 95],\ (95, \infty)$$

to quantize the data. You can do this using the following code:

```
BP<-diabetes$BloodPressure
breaks<-c(-Inf,seq(45,95,by=10),Inf)
Obs<-table(cut(BP,breaks))
```

Pearson's goodness of fit test

Hence, using Pearson's goodness of fit test, confirm that a Normal distribution is a valid assumption for the BloodPressure data, i.e. they are derived from $\mathcal{N}(\mu, \sigma^2)$ for an appropriate choice of the parameters μ and σ^2. You may find the handout on and the relevant case study helpful.

First we need an appropriate choice for μ and σ^2 which we can find by setting $\mu = \bar{x}$ and $\sigma^2 = s$ (using the sample mean and the sample variance).

Then, we have :

$$H_0 : BloodPressure\ data\ is\ derived\ from\ a\ N(\mu, \sigma^2)$$
$$H_1 : BloodPressure\ data\ is\ not\ derived\ from\ a\ N(\mu, \sigma^2)$$

Our test statistic is as follows :

$T(X) = \sum_{i=1}^{7} \frac{(O_j - E_j)^2}{E_j}$

Our observed test statistic is as follows :

$T(x) = \sum_{i=1}^{7} \frac{(o_j - e_j)^2}{e_j}$ where $o_j = n_j$ (the number of observations in category j) and $e_j = np_{0,j}$ (the expected number of observations in category j under H_0)

We then find our p-value range which is $\mathbb{P}(T(X) > T(x))$ under H_0 for $T(X) \sim \chi^2((7-1)-2) = \chi^2(4)$ for our lower bound and $\mathbb{P}(T(X) > T(x))$ under H_0 for $T(X) \sim \chi^2(7-1) = \chi^2(6)$ for our upper bound

We compute all the above in R as follows

```
degreeOfFLower <- 4
degreesOfFreedomUpper <- 6
# setting mean to be sample mean
sampleMean <- mean(diabetes$BloodPressure)
# setting variance to be sample variance
sampleVar <- var(diabetes$BloodPressure)
# storing the value of each category in an array (this represents nj)
nJ <- as.vector(Obs)
n <- sum(nJ)
# finding p0j (the probability of observing each interval) under H0
p0J <- rep(0, length(breaks)-1)
# observing the first interval which is just being less than 45 w.r.t H0
p0J[1] <- pnorm(breaks[2], sampleMean, sqrt(sampleVar))
# observing the last interval which is just being larger than 95 w.r.t H0
p0J[length(breaks)-1] <- 1-pnorm(breaks[length(breaks)-1], sampleMean, sqrt(sampleVa
r))
# observing the rest of the intervals w.r.t H0
for(i in 2:length(breaks)-2) {
  p0J[i] <- pnorm(breaks[i+1], sampleMean, sqrt(sampleVar)) - pnorm(breaks[i], sample
Mean, sqrt(sampleVar))
}
# finding np0J (the expected number of observations in category j)
np0J <- n*p0J
# we have the tools to calculate the observed test statistic now
obsTestStatGoodnessOfFit <- sum((nJ-np0J)^2/np0J)
# p-value
pValLow <- 1-pchisq(obsTestStatGoodnessOfFit, degreeOfFLower)
pValUp <- 1-pchisq(obsTestStatGoodnessOfFit, degreesOfFreedomUpper)
paste("The lower bound p-value is : " , pValLow)
```

```
## [1] "The lower bound p-value is :  0.135621717398305"
```

```
paste("The upper bound p-value is : ", pValUp)
```

```
## [1] "The upper bound p-value is :  0.320380641752077"
```

Since the lower bounded p-value is larger than any appropriate α level test, we can conclude that we have insufficient evidence to reject H_0 thus we confirm that a normal distribution is indeed a valid assumption for the BloodPressure data.

Question 2 [2 marks] Assuming that the BloodPessure measurements are realizations of i.i.d. random variables from $\mathcal{N}(\mu, 12)$, perform the following test for an appropriate (UMP) test statistic

$$H_0 : \mu = 70 \quad vs \quad H_1 : \mu > 70$$

reporting the p-value of the test. State clearly the test statistic used and justify your choice appropriately.

$$Let\ X \overset{iid}{\sim} N(\mu, 12)$$

$$Then\ calculate\ the\ Neyman\ Pearson\ test\ statistic\ (NP)\ for\ the\ below\ simple\ hypothesis$$

$$H_0 : \mu = 70\ vs\ H_1 : \mu = \mu_1$$

$$where\ u_1 > \mu$$

$$We\ get\ T_{NP}(\mathbf{x}) = \sum_{i=1}^{n} \frac{f(x_i;\mu_1)}{f(x_i;\mu)}$$

$$= \prod_{i=1}^{n} \frac{\frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x_i-\mu_1)^2}{2\sigma^2}}}{\frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(x_i-\mu)^2}{2\sigma^2}}}$$

$$= \prod_{i=1}^{n} e^{\frac{(x_i-\mu)^2}{2\sigma^2} - \frac{(x_i-\mu_1)^2}{2\sigma^2}}$$

$$= e^{\frac{1}{2\sigma^2} \sum_{i=1}^{n} [x_i^2 - 2x_i\mu + u^2 - (x_i^2 - 2x_i\mu_1 + \mu_1^2)]}$$

$$= e^{\frac{1}{2\sigma^2} \sum_{i=1}^{n} [2x_i(\mu_1 - \mu) + \mu^2 - \mu_1^2]}$$

$$= e^{\sum_{i=1}^{n} [\frac{x_i(\mu_1-\mu)}{\sigma^2} + \frac{\mu^2}{2\sigma^2} - \frac{\mu_1^2}{2\sigma^2}]}$$

$$= e^{\sum_{i=1}^{n} \frac{x_i(\mu_1-\mu)}{\sigma^2} + \frac{n\mu^2}{2\sigma^2} - \frac{n\mu_1^2}{2\sigma^2}}$$

$$= e^{\sum_{i=1}^{n} \frac{x_i(\mu_1-\mu)}{\sigma^2}} e^{\frac{n\mu^2}{2\sigma^2}} e^{\frac{-n\mu_1^2}{2\sigma^2}}$$

$$\propto e^{\sum_{i=1}^{n} \frac{x_i(\mu_1-\mu)}{\sigma^2}}$$

Since $\mu_1 > \mu$ then $T_{NP}(\mathbf{x})$ is increasing with $S_n(\mathbf{x}) := \sum_{i=1}^n x_i$. Thus an equivalent test statistic to $T_{NP}(\mathbf{x})$ is $T(\mathbf{x}) := S_n(\mathbf{x})$ Since $T(\mathbf{x})$ does not depend on μ or μ_1 then by Theorem 20.1 $T(\mathbf{x})$ is the uniformly most powerful test statistic (UMP).

Computing the p-value of the test :

$$p := \mathbb{P}(T(\mathbf{X}) \geq T(\mathbf{x}); \mu)$$

$$= \mathbb{P}(\sum_{i=1}^{n} X_i \geq T(\mathbf{x}); \mu)$$

$$= \mathbb{P}(Y \geq T(\mathbf{x}); \mu)$$

$$where\ Y \sim N(n\mu, 12n)$$

$$= 1 - \mathbb{P}(Y < T(\mathbf{x}); \mu)$$

$$which\ we\ now\ compute\ in\ R\ as\ follows$$

```r
# stores value
n = 768
mu = 70
variance = 12
newVariance = variance*n
# calculate observed test statistic value
obsTestStat <- sum(diabetes$BloodPressure)
# calculates the p-value
pValue <- 1-pnorm(obsTestStat, n*mu, sqrt(newVariance))
paste("The P-Value of the test is as follows : " , pValue)
```

```
## [1] "The P-Value of the test is as follows :  0"
```

This P-Value suggests it is impossible to see values as large as our observed test statistic under H_0, thus suggesting it is unlikely that obsTestStat is a realization of Y thus giving us evidence to reject H_0 which suggests $\mu > 70$.

Question 3 [1 mark] Show that for $\pi_i = \mathbb{P}(Y_i = 1)$

$$\log \frac{\pi_i}{1-\pi_i} = \sum_{j=1}^{d} \theta_j x_{ij}.$$

Note $\pi_i = \mathbb{P}(Y_i = 1) = \sigma(\theta^T x_i)$ (by definition of Bernoulli)

Consider

$$\begin{gathered}
log\frac{\pi_i}{1-\pi_i} \\
= log(\pi_i) - log(1-\pi_i) \\
= log(\sigma(\theta^T x_i)) - log(1-\sigma(\theta^T x_i)) \\
= log[\frac{1}{1+exp(-\theta^T x_i)}] - log[1 - \frac{1}{1+exp(-\theta^T x_i)}] \\
= log[\frac{1}{1+exp(-\theta^T x_i)}] - log[\frac{1+exp(-\theta^T x_i)-1}{1+exp(-\theta^T x_i)}] \\
= log[\frac{1}{1+exp(-\theta^T x_i)}] - log[\frac{exp(-\theta^T x_i)}{1+exp(-\theta^T x_i)}] \\
= log[\frac{1}{1+exp(-\theta^T x_i)} * \frac{1+exp(-\theta^T x_i)}{exp(-\theta^T x_i)}] \\
= log[\frac{1}{exp(-\theta^T x_i)}] \\
= log[1] - log[exp(-\theta^T x_i)] \\
= 0 - (-\theta^T x_i) \\
= \theta^T x_i \\
= \theta_1 x_{i1} + \theta_2 x_{i2} + ... + \theta_d x_{id} = \sum_{i=1}^{d} \theta_j x_{ij} \\
as\ required
\end{gathered}$$

Question 4 [2 marks] Use the generalized likelihood ratio test to decide whether the variables BloodPressure, SkinThickness, Insulin and Age are statistically significant for the development of diabetes.

To do this start by forming the matrices that correspond to the restricted model under the null hypothesis (X_rest) and to the full model (X_full).

```
# all the data
X_full<-cbind(1,as.matrix(diabetes[,1:8]))
# data not dependent on BloodPressure, SkinThickness, Age and Insulin
X_rest<-cbind(1,as.matrix(diabetes[,c(1,2,6,7)]))
# data stating if you have diabetes or not
Y<-diabetes[,9]
```

Functions required for Q4

```
sigma <- function(v) {
  1/(1+exp(-v))
}

# returns the log-likelihood function
ell <- function(theta, X, y) {
  p <- as.vector(sigma(X%*%theta))
  sum(y*log(p) + (1-y)*log(1-p))
}

# calculates the score vector
score <- function(theta, X, y) {
  p <- as.vector(sigma(X%*%theta))
  as.vector(t(X)%*%(y-p))
}



# computes the ML estimate
maximize.ell <- function(ell, score, X, y, theta0) {
  optim.out <- optim(theta0, fn=ell, gr=score, X=X, y=y, method="BFGS",
                     control=list(fnscale=-1, maxit=1000, reltol=1e-16))
return(list(theta=optim.out$par, value=optim.out$value))
}
```

Firstly, begin by stating H_0 and H_1

$$H_0 : \theta_3 = \theta_4 = \theta_5 = \theta_8 = 0$$
$$H_1 : \textit{at least one of } \theta_3, \theta_4, \theta_5, \theta_8 \textit{ not equal to } 0$$

Consider our test statistic which is as follows :

$T(\mathbf{Y}) = -2\log\Lambda_n(\mathbf{Y}) = -2\{l(\hat{\boldsymbol{\theta}}_0; \mathbf{Y}) - l(\hat{\boldsymbol{\theta}}_{MLE}; \mathbf{Y})\} \sim \chi^2_4$ under H_0

Consider our observed test statistic which is as follows :

$$T(\mathbf{y}) = -2\log\Lambda_n(\mathbf{y}) = -2\{l(\hat{\boldsymbol{\theta}}_0; \mathbf{y}) - l(\hat{\boldsymbol{\theta}}_{MLE}; \mathbf{y})\}$$

To find $T(\mathbf{y})$ we first need to find $\hat{\boldsymbol{\theta}}_0$ which is the maximum likelihood estimator under the null hypothesis and which can be found using the maximise.ell function calling in the relevant parameters (which are ell, score, X_rest, Y and theta0).

Then we need to find $l(\hat{\boldsymbol{\theta}}_0; \mathbf{y})$ which can be found using the ell function (log-likelihood function) again passing in the relevant parameters ($\hat{\boldsymbol{\theta}}_0$, X_rest, Y)

Finally, we need to find $\hat{\boldsymbol{\theta}}_{MLE}$ which is the maximum likelihood estimator for the full model and can be found using maximise.ell where the only different parameter to last time is X_rest (which is replaced with X_full) and similarly we can find $l(\hat{\boldsymbol{\theta}}_{MLE}; \mathbf{y})$ using the ell function replacing the parameter X_rest with X_full and $\hat{\boldsymbol{\theta}}_0$ with $\hat{\boldsymbol{\theta}}_{MLE}$.

Once we have a value for $T(\mathbf{y})$ we can calculate $\mathbb{P}(Z > T(\mathbf{y}))$ for $Z \sim \chi^2_4$.

Below is the R Code which does the above :

```
parametersOmittedNum <- 4
totalParameters <- 9
degreesOfFreedom <- 4
testStatisticIs <- function(Y) {
  # finding theta hat 0
  thetahat0 <- maximize.ell(ell, score, X_rest, Y, rep(0, totalParameters-parametersO
mittedNum))
  thetahat0Thetas <- thetahat0$theta
  # finding log-likelihood of theta hat 0
  loglikelihoodthetahat0 <- ell(thetahat0Thetas, X_rest, Y)
  # finding theta hat MLE
  thetahatMLE <- maximize.ell(ell, score, X_full, Y, rep(0, totalParameters))
  thetahatMLEThetas <- thetahatMLE$theta
  # finding log-likelihood of theta hat MLE
  loglikelihoodthetahatMLE <- ell(thetahatMLEThetas, X_full, Y)
  # calculate observed test statistic
  observedT <- -2*(loglikelihoodthetahat0 - loglikelihoodthetahatMLE)
  return(observedT)
}
# calculates probability chi squared 4 greater than the observed test statistic
prob <- 1-pchisq(testStatisticIs(Y), degreesOfFreedom)
paste("The probabillity of seeing values as extreme as our observed test statistic un
der the null hypothesis is " , prob)
```

```
## [1] "The probabillity of seeing values as extreme as our observed test statistic u
nder the null hypothesis is  0.474285746036054"
```

This p-value gives us insufficient evidence to reject H_0 since it is larger than any appropriate α level test thus suggesting we can accept H_0 thus concluding $\theta_3 = \theta_4 = \theta_5 = \theta_8 = 0$ implying that the variables BloodPressure, SkinThickness, Insulin and Age are not statistically significant for the development of diabetes.

Q5 Pre-requisite function

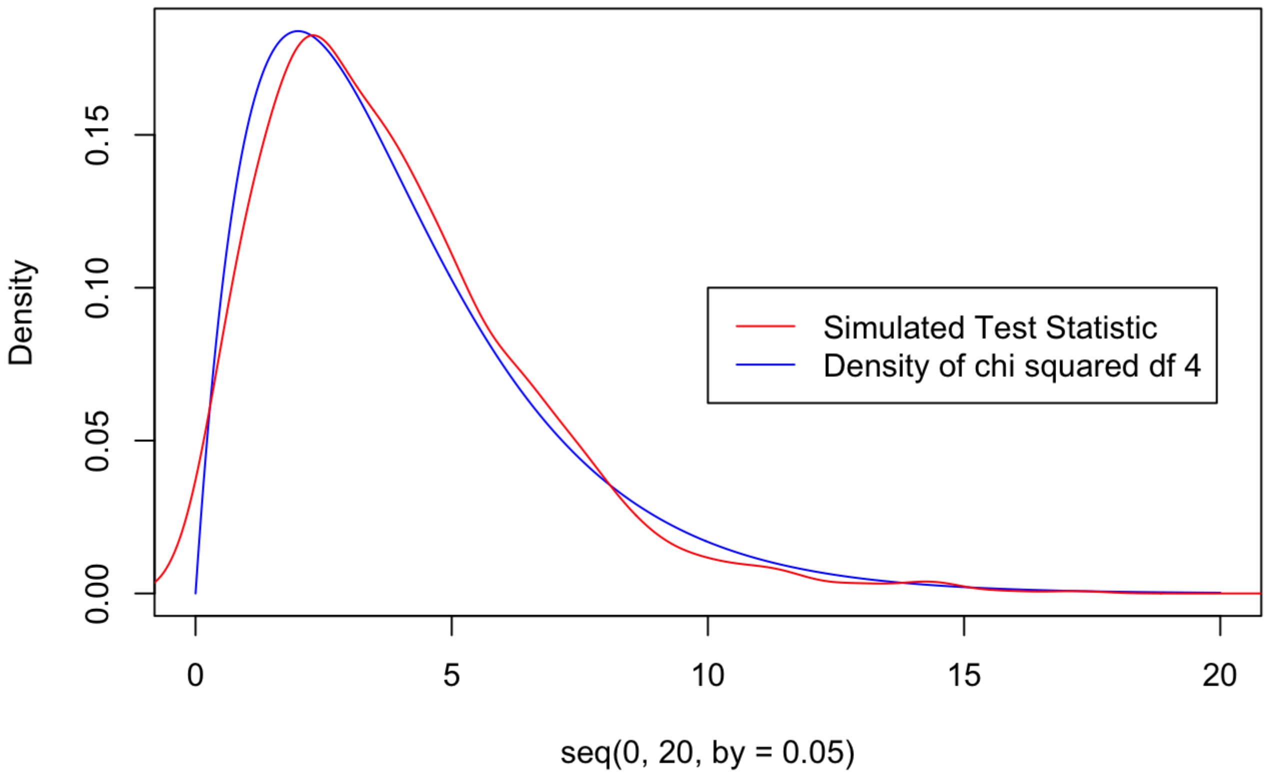
```
generate.ys <- function(X, theta) {
  n <- dim(X)[1]
  rbinom(n, size = 1, prob=sigma(X%*%theta))
}
```

Question 5 [3 marks] Consider the test in Question 4. By repeated experiments, simulate the test statistic $-2\log\Lambda_n$ using the appropriate maximum likelihood estimate for $\boldsymbol{\theta}$. Hence, verify that it has a $\mathcal{X}^2$ distribution with the underlying degrees of freedom.

Question 5 Part A

Plot the density of the simulated $-2\log\Lambda_n$ values against the density of the corresponding chi-squared distribution

```r
numberofSimulations <- 1000
simulatedTestStat <- rep(0, numberofSimulations)
# computes MLE under the null hypothesis with the actual response variables
# this MLE will be needed to generate simulated response variables
thetahat0 <- maximize.ell(ell, score, X_rest, Y, rep(0, totalParameters-parametersOmittedNum))
thetahat0Thetas <- thetahat0$theta
for(i in 1:numberofSimulations) {
  # simulates response variable under the MLE for theta0 found in Q4
  simulatedY <- generate.ys(X_rest, thetahat0Thetas)
  # computes the simulated test statistic using function written in Q4
  simulatedTestStat[i] <- testStatisticIs(simulatedY)
}
# plotting the densities
xOfDensityChi <- dchisq(seq(0, 20, by=0.05), degreesOfFreedom)
plot(seq(0, 20, by=0.05), xOfDensityChi, col="blue", type="l", ylab ="Density")
lines(density(simulatedTestStat), col="red")
legend(10, y=0.10, legend=c("Simulated Test Statistic", "Density of chi squared df 4"), col=c("red", "blue"), lty=1:1)
```



Question 5 Part B

Perform the Pearson’s goodness-of-fit test. [Note that the intervals chosen to quantize the observed data should meet the criterion of each interval having expected counts ≥ 5.]

Before we can perform the Pearson’s goodness of fit test, we must quantize our observed data into intervals which we do in R as follows to ensure each interval has expected counts ≥ 5

```
degreeOfFreedomH0 <- 4
# creating the intervals
intervals <- c(-Inf, seq(0.5, 11, by=0.5), Inf)
# quantizing observed data into intervals
countInInterval <- table(cut(simulatedTestStat, intervals))
# observed values in each interval stored in an array
observed <- as.vector(countInInterval)
degreeOfFreedomTestStat <- length(observed) - 1
numberOfObserved <- sum(observed)
# find the probability of observing each interval under H0
probObservingInterval <- rep(0, length(intervals)-1)
# probability of observing the first interval is just being less than 0.5 w.r.t H0
probObservingInterval[1] <- pchisq(intervals[2], degreeOfFreedomH0)
# probability of observing the last interval is just being larger than 11 w.r.t H0
probObservingInterval[length(intervals)-1] <- 1-pchisq(intervals[length(intervals)-1
], degreeOfFreedomH0)
# probability of observing the rest of the intervals w.r.t H0
for(i in 2:length(intervals)-2) {
  probObservingInterval[i] <- pchisq(intervals[i+1], degreeOfFreedomH0) - pchisq(inte
rvals[i], degreeOfFreedomH0)
}
# finds ej = the expected number of observations in category j w.r.t H0
expectedCount <- numberOfObserved*probObservingInterval
print(expectedCount)
```

```
##  [1] 26.499021 63.704989 83.154522 90.882585 91.123089 86.810393 79.947056
##  [8] 71.872495 63.458370 55.249985 47.568016 40.581206 34.357890 28.902158
## [15] 24.178933 20.131098 16.690966 13.787747 11.352234  9.319565  7.630692
## [22]  6.232976 26.564014
```

We see the expected count of each interval is ≥ 5 thus we can now perform Pearson's goodness of fit test as follows :

$$H_0 : -2\log\Lambda_n \sim \chi^2(4)$$
$$H_1 : -2\log\Lambda_n \sim \chi^2(4)\ \textit{does not hold}$$

Our test statistic is as follows :

$T(X) = \sum_{i=1}^{23} \frac{(O_j - E_j)^2}{E_j} \sim \chi^2(23-1) = \chi^2(22)$ (since we have 23 intervals thus we sum from 1 to 23)

Our observed test statistic is as follows :

$T(x) = \sum_{i=1}^{23} \frac{(o_j - e_j)^2}{e_j}$ where $o_j = n_j$ (the number of observations in category j) and $e_j = np_{0,j}$ (the expected number of observations in category j under H_0)

Our p-value is as follows :

p := $\mathbb{P}(T(X) > T(x))$ under H_0

We compute this in R as follows :

```
# computes the test statistic
testStatisticObtained <- sum((observed-expectedCount)^2/expectedCount)
# p-value
pValueObtained <- 1-pchisq(testStatisticObtained, degreeOfFreedomTestStat)
paste("The p-value obtained is " , pValueObtained)
```

```
## [1] "The p-value obtained is  0.728508792916306"
```

This p-value is not significantly small (it is larger than any sensible level α test) thus suggesting we have insufficient evidence to reject H_0 suggesting $-2\log\Lambda_n \sim \chi^2(4)$.