Hw4

Mingming Xu

8/2/2019

Problem 1

```
set.seed(3701)
n=100
y.vec=rnorm(n,0,1)
```

```
xcol_1=rep(1,n)
xcol_2=c(rep(1,n/4),rep(2,n/4),rep(3,n/4),rep(4,n/4))
xcol_3=c(rep(c(0.25,0.5,0.75,1),n/4))
x.mat=cbind(xcol_1,xcol_2,xcol_3)
## for the regressing model on x1 and x2
beta.hat=qr.coef(qr=qr(x.mat),y=y.vec)
beta.hat
```

```
##      xcol_1      xcol_2      xcol_3
##  0.12720338 -0.01013733 -0.11903481
```

For the regressing model $Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \xi_i$ on x1 and x2, I get the fitted model:

$\hat{Y_i} = 0.1272 - 0.0101 x_{i1} - 0.1190 x_{i2}$

```
x.mat.1=cbind(xcol_1,xcol_2)
beta.hat.1=qr.coef(qr=qr(x.mat.1),y=y.vec)
beta.hat.1
```

```
##      xcol_1      xcol_2
##  0.05578249 -0.01132768
```

for the regressing model $Y_i = \alpha + \beta x_{i1} + \epsilon_i$ on x1, I get the fitted model:

$\hat{Y_i} = 0.0558 - 0.0113 x_{i1}$

```
yi.hat.on.x1=x.mat.1 %*% beta.hat.1
ei.hat=y.vec-yi.hat.on.x1
x.mat.2=cbind(xcol_1,xcol_3)
beta.hat.ei=qr.coef(qr=qr(x.mat.2),y=ei.hat)
beta.hat.ei
```

```
##                [,1]
## xcol_1  0.07427772
## xcol_3 -0.11884435
```

for the regressing model$\hat{e_i} = \alpha_1 + \beta_1 x_{i2} + \eta_i$, I get the fitted model :

$\hat{e_i} = 0.0743 - 0.1188 x_{i2}$

We could see that the coefficents on x_{i1} and x_{i2} are so closed to the full model's.

Problem 2

```
set.seed(1729)
x.mat3=read.csv("X.mat.3.csv")
x.mat.3=as.matrix(x.mat3)
n=nrow(x.mat.3)
```

```
p=ncol(x.mat.3)
y=rnorm(n,0,1)
```

i.

```
xnew=c(1/3,1/3,1/3,-1/2,-1/2)
beta.hat=qr.coef(qr=qr(x.mat.3),y=y)
xtxin=qr.solve(crossprod(x.mat.3))
alpha=0.05
tper=qt(1-alpha/2,n-p)
sq.form=sqrt(t(xnew) %*% xtxin %*% xnew)
est.mean=sum(beta.hat*xnew)
s.q=sum((y-x.mat.3 %*% beta.hat)^2)/(n-p)
ci=c(est.mean-tper *sqrt(s.q)*sq.form,est.mean+tper*sqrt(s.q)*sq.form)
ci
```

```
## [1] -0.0454454  1.1747651
```

ii.

The test could be written as $H_0 : \frac{\beta_1+\beta_2+\beta_3}{3} - \frac{\beta_4+\beta_5}{2} = 0$

$H_a : \frac{\beta_1+\beta_2+\beta_3}{3} - \frac{\beta_4+\beta_5}{2} \neq 0$

Because the confidence interval I get in i contains 0.Hence we fail to reject the null hypothesis.

iii. We could see this test as $H_0 : C\beta = (0, 0, ..0)$

```
## the matrix c
c1=c(1/3,1/3,1/3,-1/2,-1/2)
c2=c(0,0,1,-1,0)
c.mat=rbind(c1,c2)
d=2
##
c.sq=t(c.mat) %*% qr.solve(c.mat %*% xtxin %*% t(c.mat)) %*% c.mat
rssf=sum((y-x.mat.3 %*% beta.hat)^2)
f=t(beta.hat) %*% c.sq %*% beta.hat*(n-p)/(d* rssf)
1-pf(f,d,n-p)
```

```
##           [,1]
## [1,] 0.1786628
```

Because the p value is greater than 0.05, we fail to reject the null hypothesis.

Problem 3

i.

$f(y) = \lambda exp(-\lambda y)$

$l(\lambda; y_1, y_2...y_n) = \prod_{i=1}^{n} f(y_i) = \prod_{i=1}^{n} \lambda exp(-\lambda y_i)$

$log(l(\lambda; y_1, y_2...y_n)) = log(\prod_{i=1}^{n} \lambda exp(-\lambda y_i)) = \sum_{i=1}^{n} log(\lambda exp(-\lambda y_i)) = nlog\lambda - \lambda \sum_{i=1}^{n} y_i = nlog\lambda - n\lambda\bar{y}$

$\frac{dlogl}{d\lambda} = \frac{n}{\lambda} - n\bar{y} = 0$,so $\hat{\lambda} = 1/\bar{y}$

$\frac{d^2logl}{d\lambda^2} = \frac{-n}{\lambda^2}$, it is less than 0 so that $\hat{\lambda}$ is a global maximiner.

Hence, the MLE is $\hat{\lambda} = 1/\bar{y}$.

ii.

```r
reps=5e4
n=40
lambda=3
est.matrix=matrix(NA,nrow = reps,ncol = 3)

for(r in 1:reps){
  ##generate y
  y.list=rexp(n,lambda)
  ##compute the lambda hat
  ybar=mean(y.list)
  est.matrix[r,1]=1/ybar
  ##compute the lambda s
  s.sq=(1/(n-1))*sum((y.list-ybar)^2)
  est.matrix[r,2]=1/(sqrt(s.sq))
  ##compute lambda m
  est.matrix[r,3]=log(2)/median(y.list)
}

apply(est.matrix,2,quantile,probs=c(0.025,0.975))
```

```
##           [,1]     [,2]     [,3]
## 2.5%  2.249398 2.067968 1.985789
## 97.5% 4.209494 4.792442 4.787730
```

iii.

For these estimators, I wanna to compare their MSE to choose which estimator is best:

```r
mse.lambda.hat=mean((est.matrix[,1]-lambda)^2)
mse.lambda.s=mean((est.matrix[,2]-lambda)^2)
mse.lambda.m=mean((est.matrix[,3]-lambda)^2)
mse.lambda.hat
```

```
## [1] 0.2567693
```

```r
mse.lambda.s
```

```
## [1] 0.5310726
```

```r
mse.lambda.m
```

```
## [1] 0.5337215
```

Because the MSE of $\hat{\lambda}$ is the samllest, it is the best estimator.

Problem 4

$g(x_i) = pf_1(x_i) + (1-p)f_2(x_i)$

$f_1(x_i) = \frac{1}{\sqrt{8\pi}} exp(\frac{-(x-\mu)^2}{8\pi})$

$f_2(x_i) = \frac{1}{\sqrt{10\pi}} exp(\frac{-(x-\mu)^2}{10\pi})$

The likelihood function:

$l(\mu, p, x_1, x_2, ...x_n) = \prod_{i=1}^{n} g(x_i)$

$log(l(\mu, p, x_1, x_2...x_n)) = log(\prod_{i=1}^{n} g(x_i)) = \sum_{i=1}^{n} log(g(x_i))$

$= \sum_{i=1}^{n} log[p\frac{1}{8\pi} exp(\frac{-(x-\mu)^2}{8\pi}) + (1-p)\frac{1}{\sqrt{10\pi}} exp(\frac{-(x-\mu)^2}{10\pi})]$

$\frac{dlogl}{d\mu} = \sum_{i=1}^{n} \frac{1}{g(x_i)}[\frac{p}{\sqrt(8\pi)}e^{\frac{-(x_i-\mu)^2}{8}}\frac{(x_i-\mu)}{4} + (1-p)\frac{1}{\sqrt(10\pi)}e^{\frac{-(x_i-\mu)^2}{10}}\frac{(x_i-\mu)}{5}]$

$= \sum_{i=1}^{n} \frac{(x_i-\mu)}{g(x_i)}[\frac{p}{4\sqrt(8\pi)}e^{\frac{-(x_i-\mu)^2}{8}} + (1-p)\frac{1}{5\sqrt(10\pi)}e^{\frac{-(x_i-\mu)^2}{10}}]$

Then ,let the $\frac{dlogl}{d\mu}$ equals 0 we could get the MLE of μ.