Lab2_solution

Ladan Tazik

2022-12-08

Q1

a)

```
X <- matrix(seq(1, 6), nrow=3)
H = X%*%(solve(t(X)%*%X))%*%t(X)
H
```

```
##            [,1]      [,2]       [,3]
## [1,]  0.8333333 0.3333333 -0.1666667
## [2,]  0.3333333 0.3333333  0.3333333
## [3,] -0.1666667 0.3333333  0.8333333
```

b)

```
all.equal(H, H%*%H)
```

```
## [1] TRUE
```

c) Calculate the eigenvalues and eigenvectors of H.

```
eigen_ <- eigen(H, only.values = FALSE) #this will leturn both eigen values and eigenvectors
eigen_
```

```
## eigen() decomposition
## $values
## [1]  1.000000e+00  1.000000e+00 -6.661338e-16
##
## $vectors
##            [,1]      [,2]       [,3]
## [1,]  0.8831434 0.2310651  0.4082483
## [2,]  0.2400591 0.5250762 -0.8164966
## [3,] -0.4030253 0.8190873  0.4082483
```

d) Calculate the trace of the matrix H, and compare with the sum of the eigenvalues

```
trace_H <- sum(diag(H))
all.equal(trace_H, round(sum(eigen_$values),3))
```

```
## [1] TRUE
```

e) Calculate the determinant of the matrix H, and compare with the product of the eigen- values.

```
det_H <- det(H)

all.equal(det_H, round(prod(eigen_$values),3))
```

```
## [1] TRUE
```

f) Using the definition of eigenvector, verify that the columns of X are eigenvectors of H.

```
col1 <- X[,1]
col2 <- X[,2]

col1%*%H
```

```
##      [,1] [,2] [,3]
## [1,]    1    2    3
```

```
col1*eigen_$values[1] # are the same
```

```
## [1] 1 2 3
```

```
#do the same for the second column
col2%*%H
```

```
##      [,1] [,2] [,3]
## [1,]    4    5    6
```

```
col2*eigen_$values[2] # are the same
```

```
## [1] 4 5 6
```

Therefore, the columns of X are eigenvectors for H.

Q2

b) yes, they are converging to a specific distribution for each row.

```
P <- matrix(c(.5, .1, .1, .1, .2, .1, .2, .3, .1, .1, .1, .1, .2, .7, .6, .5), nrow=4)
P2 <- P%*%P
P3 <- P2%*%P
P5 <- P2%*%P3
P5
```

```
##         [,1]    [,2] [,3]    [,4]
## [1,] 0.17520 0.22640  0.1 0.49840
## [2,] 0.16496 0.22784  0.1 0.50720
## [3,] 0.16496 0.22800  0.1 0.50704
## [4,] 0.16496 0.22816  0.1 0.50688
```

```
P10 <- P5%*%P5
P10
```

```
##           [,1]      [,2] [,3]      [,4]
## [1,] 0.1667540 0.2277632  0.1 0.5054828
## [2,] 0.1666492 0.2277808  0.1 0.5055700
## [3,] 0.1666492 0.2277807  0.1 0.5055701
## [4,] 0.1666492 0.2277807  0.1 0.5055702
```

Q3

we changed this question to exercise 5 from chapter 4.

(a) Write down the design matrix X

```
x <- c (0,1,2,3,4,5)
intercept_column <- rep(1,6)
X <- cbind(intercept_column, x)
```

```
colnames(X) <- c("", "")
X
```

```
##
## [1,] 1 0
## [2,] 1 1
## [3,] 1 2
## [4,] 1 3
## [5,] 1 4
## [6,] 1 5
```

b) Determine the QR decomposition for X

```
X_QR <- qr(X)
```

c) calculate U^-1.

```
U <- qr.R(X_QR, complete=FALSE)
U_inv <- solve(U)
U_inv
```

```
##           [,1]       [,2]
##   -0.4082483 -0.5976143
##    0.0000000  0.2390457
```

d) Determine the slope and intercept estimates, using the QR decomposition

```
Q <- qr.Q(X_QR, complete=TRUE)
Q1 <- Q[, 1:2]
y <- c(1,3,5,6,5,7)
Q1y <- t(Q1)%*%y
betahat <- solve(U, Q1y)
betahat
```

```
##        [,1]
##   1.857143
##   1.057143
```

e) Determine the residual sum of squares, using the QR decomposition.

```
Q2 <- Q[, -(1:2)]
Q2y <- t(Q2)%*%y
SSE <- t(Q2y)%*%Q2y
SSE
```

```
##            [,1]
## [1,] 3.942857
```

f) Estimate the error variance.

```
MSE <- SSE/4
MSE
```

```
##             [,1]
## [1,] 0.9857143
```

g, h) Calculate the test statistic used to determine whether the slope is 0 or not.

for this, we need to use t distribution so t test is $t = \frac{\beta_1}{SE_{\beta_1}} = \frac{betahat[1]}{SE_{\beta_1}}$, in order to get SE_{β_1}, we do:

```
Cii <- sqrt(diag(solve(t(U)%*%U)))
SEii <- Cii*as.numeric(sqrt(MSE))
SEii[2]
```

```
##
## 0.2373321
```

so t test would be $\frac{1.05}{0.23} = 4.56$ on 4 degress of freedom. P value coresspond to this is:

```
pt(q=-4.56, df=4, lower.tail=TRUE)
```

```
## [1] 0.00516942
```

which is very small. so we reject the null hypotheses and slope is not 0.

Q5

a) identify the logit of the probability of home ownership as a linear function of family income.

```
library(MPV)
```

```
## Loading required package: lattice
```

```
## Loading required package: KernSmooth
```

```
## KernSmooth 2.23 loaded
## Copyright M. P. Wand 1997-2009
```

```
my_glm = glm(y~.,
              family = binomial,
              data = p13.2)
coef(my_glm)
```

```
##   (Intercept)             x
## -8.7395139021  0.0002009056
```

$logit(y) = -8.74 + 0.0002 \times x$

b) determine if the logistic model is reasonable

```
summary(my_glm)
```

```
##
## Call:
## glm(formula = y ~ ., family = binomial, data = p13.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0232  -0.8766   0.5072   0.7980   1.6046
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.7395139  4.4394326  -1.969   0.0490 *
## x            0.0002009  0.0001006   1.998   0.0458 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 27.526  on 19  degrees of freedom
## Residual deviance: 22.435  on 18  degrees of freedom
## AIC: 26.435
##
## Number of Fisher Scoring iterations: 4
```

c) estimate the probability that a family with an income of $40000 owns their home.

```
newdata = data.frame(x = 40000)

predict(my_glm,newdata,type = 'response')
```

```
##         1
## 0.3310835
```

Since the probability is less than 0.5, we assign y to be 0.

Q6

a) Fit a Poisson regression with glm.

```
library(MASS)
```

```
##
## Attaching package: 'MASS'

## The following object is masked from 'package:MPV':
##
##     cement
```

```
glm_2 <- glm(y ~ trt + age, family = poisson, data = epil)
```

b) Are the coefficients significant? *yes*

```
summary(glm_2)
```

```
##
## Call:
## glm(formula = y ~ trt + age, family = poisson, data = epil)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -4.3628  -2.4087  -1.3791   0.0006  17.8489
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)   2.533031   0.110638  22.895  < 2e-16 ***
## trtprogabide -0.092514   0.045596  -2.029 0.042460 *
## age          -0.013331   0.003708  -3.596 0.000324 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 2517.8  on 235  degrees of freedom
## Residual deviance: 2502.0  on 233  degrees of freedom
## AIC: 3273.9
##
## Number of Fisher Scoring iterations: 6
```

c) What is the 95% confidence interval for the estimates of the coefficients.

```
beta <- glm_2$coefficients
SE <- c(0.11, 0.04, 0.003)
#intercept
beta[1] +SE[1]*qt(c(.025, .975), df = 233) #236-3 degrees of freedom
```

```
## [1] 2.316310 2.749753
```

```
#beta_1
beta[2] +SE[2]*qt(c(.025, .975), df = 233)
```

```
## [1] -0.17132166 -0.01370585
```

```
#beta_2
beta[3] +SE[3]*qt(c(.025, .975), df = 233)
```

```
## [1] -0.019242066 -0.007420881
```

d) Does the treatment reduce the frequency of the seizures? *Yes, the coefficient for the treatment (progabide) is negative*

e) According to this model, what would be the number of seizures for 20 years old patient with progabide treatment? *9*

```
predict(glm_2,data.frame(age = 20,trt = 'progabide'),type = 'response')
```

```
##        1
## 8.792404
```