```
rm(list=ls())
# Load necessary libraries
library(MASS)
library(robNB)
library(COUNT)
data(nuts)
y=nuts[,1]
x=nuts[,-c(1,6,7)]
d=data.frame(x,y)
library(corrplot)
M = cor(d)
M
p <- ncol(x)

# Create the design matrix 'designX' using the predictor variables
designX <- model.matrix(~.- 1, data = d)

# Fit negative binomial (NB) regression model using glm.nb
ml <- glm.nb(y ~.-1, data = d, control = glm.control(epsilon = 1e-10))

# Obtain weights from the NB model fit and calculate the weighted design matrix
W <- diag(ml$weights)
x_weighted <- t(x)%*%W

# Calculate the eigenvalues and eigenvectors of the weighted design matrix
s <- x_weighted%*%as.matrix(x)
e <- eigen(s)$values
q <- eigen(s)$vectors
```

```
# Perform some calculations based on the eigenvalues and eigenvectors
mle <- ml$coefficients
a <- t(q) %*% mle

# Fit a robust NB regression model using nb.glm.rob and extract the coefficients
MM <- nb.glm.rob(y, designX, offset = rep(0, length(y)), c.tukey.beta = 10, c.tukey.sigma = 10,
            weights.on.x = "none", quantile.used = floor(length(y) * 0.8),
            minsig = 0.001, maxsig = 50, minmu = 1e-10, maxmu = 1e+20, maxit = 1000,
            tol = 1e-07, maxit.sig = 30, tol.sig = 1e-06, warn = FALSE)$coef

MM <- MM[-c(1, 2)]



#k=p/sum(a^2)
k=1/max(a^2)
 #km=1/sum(MM^2)
km=(1/max(MM^2))
 k1=(1/((2*a^2)+(1/e)))
 k2=(p/sum((2*a^2)+(1/e)))
 k3=min(e/((2*e*a^2)+1))^.5
 k4=min(e/((2*e*a^2)+1))
 d=max(0,(((a^2)-1)/((1/e)+(a^2))))
 d1=min(1/(a^2))
 d1m=min(1/(MM^2))
 I=diag(1,p)
 re=solve(s+diag(as.numeric(k),p))%*%s%*%mle
 rem=solve(s+diag(as.numeric(km),p))%*%s%*%MM
 kl1=solve(s+k*I)%*%(s-k*I)%*%mle
 kl1m=solve(s+km*I)%*%(s-km*I)%*%MM
```

```
msemle=sum(1/e)
msere=sum((e+(k^2)*(a^2))/(e+k)^2)
mserre=sum((e+(km^2)*(MM^2))/(e+km)^2)
msekl=sum(((e-k)^2)/(e*(k+e)^2))+sum(((2*k)^2)*((a^2)/(e+k)^2))
mserkl=sum(((e-km)^2)/(e*(km+e)^2))+sum(((2*km)^2)*((MM^2)/(e+km)^2))
mse=c(msemle,msere,mserre,msekl,mserkl)
coeff=cbind(mle,re,rem,kl1,kl1m)
dat=rbind(coeff,mse)
colnames(dat)=c('MLE','RIDGE','M-RIDGE', 'KL','M-KL')
rownames(dat)=c('X1','X2','X3','X4','X5','MSE')
dat
par(mfrow = c(2, 2))
plot(ml)
```