```
# Daren Purnell
# Predict 422, Project 1

library(ISLR)
library(lars)
library(leaps)
library(glmnet)

# Predict Function for Regsubsets
predict.regsubsets=function(object, newdata,id,...){
  form=as.formula(object$call[[2]])
  mat=model.matrix(form,newdata)
  coefi=coef(object,id=id)
  xvars=names(coefi)
  mat[,xvars]%*%coefi
}

# Load diabetes data from lars library
data(diabetes)
data.all <- data.frame(cbind(diabetes$x, y=diabetes$y))

# Partition into training (75%) and test (25%) set
n <- dim(data.all)[1] # Sample size 442
set.seed(1306)
test <- sample(n, round(n/4))
data.train <- data.all[-test,]
data.test <- data.all[test,]

x <- model.matrix(y ~., data=data.all)[,-1]
x.train <- x[-test,]
x.test <- x[test,]
y <- data.all$y
y.train <- y[-test]
y.test <- y[test]
n.train <- dim(data.train)[1] #training sample size = 332
n.test <- dim(data.test)[1] #test sample size = 110

# EDA
head(data.all)
summary(data.all)
sum(is.na(data.all$y))
round(cor(data.all),2)
par(mfrow=c(3,3))
attach(data.all)
```

```
plot(age,y)
plot(sex,y)
plot(bmi,y)
plot(map,y)
plot(ldl,y)
plot(hdl,y)
plot(tch,y)
plot(ltg,y)
plot(glu,y)

# Q.1 Least squares regression using all ten predictors
lm.reg <- lm(y ~ ., data=data.train)
summary(lm.reg)
coef(lm.reg)
pred.lm.reg <- predict(lm.reg, data.test, interval="prediction")
lm.reg.stde <- sd((pred.lm.reg[,1]-y.test)^2)/sqrt(n.test)
lm.reg.stde # Least Squares Standard Error 361.1
lm.reg.mse <- mean((pred.lm.reg[,1]-y.test)^2)
lm.reg.mse #Least Squares Regression MSE 3111.26

# Q.2 Apply best subset selection using BIC to select the number of predictors
regfit.full <- regsubsets(y ~., data=data.train, nvmax=10)
reg.summary <-summary(regfit.full)
par(mfrow=c(1,1))
plot(reg.summary$bic, xlab="Number of variables", ylab="BIC", main= "No.of Subset Variables
vs BIC",
    type="l") # 6 variables model has lowest BIC
which.min(reg.summary$bic) # 6 variables has lowest BIC
points(6, reg.summary$bic[6], col="red", cex=2, pch=20)
coef(regfit.full, 6)
pred.subset <- predict.regsubsets(regfit.full, data.test,6)
pred.subset.se <- sd((pred.subset-y.test)^2)/sqrt(n.test)
pred.subset.se # Best Subsets Standard Error 369.75
pred.subset.mse <- mean((pred.subset-y.test)^2)
pred.subset.mse # 6 Variable Best Subset MSE 3095.48

# Q.3 Apply best subset selection using 10-fold CV to select the number of predictors
set.seed(1306)
k = 10
folds <- sample(1:k, nrow(data.train), replace=TRUE)
cv.errors=matrix(NA,k,10, dimnames=list(NULL, paste(1:10)))
for(j in 1:k){
  best.fit=regsubsets(y~., data=data.train[folds !=j,], nvmax=10)
  for(i in 1:10){
```

```
   pred=predict(best.fit, data.train[folds==j,], id=i)
   cv.errors[j,i]=mean( (data.train$y[folds==j]-pred)^2)
  }
}
mean.cv.errors=apply(cv.errors,2,mean)
mean.cv.errors
par(mfrow=c(1,1))
plot(mean.cv.errors, xlab= "No. of Variables", main="Mean CV Error vs. No. of Variables",
   type='b') # 6 variables has lowest CV.MSE 2978.91
points(6, mean.cv.errors[6], col="red", cex=2, pch=20)
coef(best.fit, 6)
pred.k10.subset <- predict.regsubsets(best.fit, data.test,6)
pred.K10.subset.se <- sd((pred.k10.subset-y.test)^2)/sqrt(n.test)
pred.K10.subset.se # Best 10-fold CV Subsets Standard Error 372.17
pred.K10.subset.mse <- mean((pred.k10.subset-y.test)^2)
pred.K10.subset.mse # 6 Variable Best 10-fold CV Subset MSE 3136.85

# Q.4 Ridge Regression using 10-fold CV to select the largest value of lambda
# that the CV error is within 1 SE of the minimum
set.seed(1306)
grid=10^seq(10,-2,length=100)
ridge.mod <- glmnet(x,y,alpha=0,lambda=grid)
cv.out <- cv.glmnet(x.train, y.train, alpha=0) # Default is 10 fold CV
ridge.lambda <- cv.out$lambda.1se
ridge.lambda #41.67209
ridge.coef <- predict(ridge.mod, s=ridge.lambda, type="coefficients")[1:11,]
ridge.coef
ridge.pred <- predict(ridge.mod, s=ridge.lambda, newx=x.test)
pred.ridge.mse <- mean((ridge.pred-y.test)^2)
pred.ridge.mse # Ridge Regression MSE 2947.51
pred.ridge.se <- sd((ridge.pred-y.test)^2)/sqrt(n.test)
pred.ridge.se # Ridge Regression SE 339.57


# Q.5 Lasso Model using 10-fold CV to select thre largest value of lambda
# that the CV error is within 1 SE of the minimum
set.seed(1306)
grid=10^seq(10,-2,length=100)
lasso.mod <- glmnet(x.train,y.train,alpha=1,lambda=grid)
cv.out <- cv.glmnet(x.train, y.train, alpha=1) # Default is 10 fold CV
lasso.lambda <- cv.out$lambda.1se
lasso.lambda #4.79
lasso.coef <- predict(lasso.mod, s=lasso.lambda, type="coefficients")[1:11,]
lasso.coef
```

```
lasso.pred <- predict(lasso.mod, s=lasso.lambda, newx=x.test)
pred.lasso.mse <- mean((lasso.pred-y.test)^2)
pred.lasso.mse # Ridge Regression MSE 2920.08
pred.lasso.se <- sd((lasso.pred-y.test)^2)/sqrt(n.test)
pred.lasso.se # Ridge Regression SE 346.228
```