HW10

Giang Vu

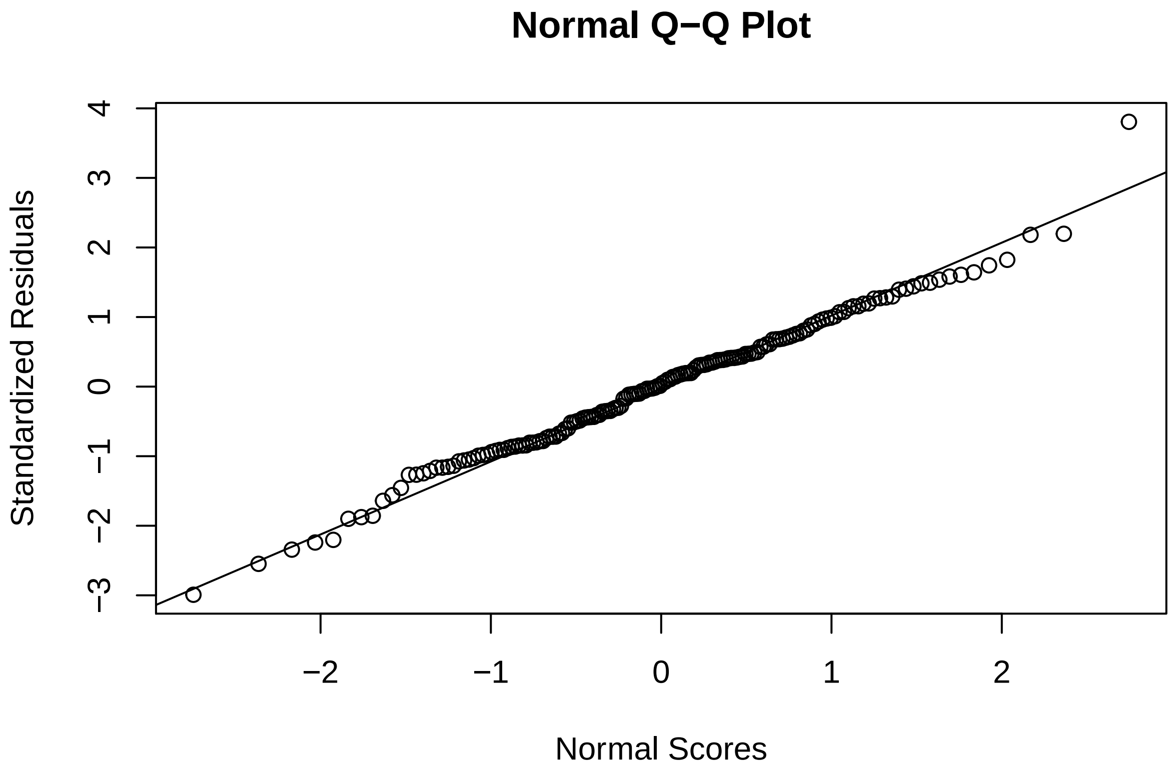
11/15/2020

Problem 1

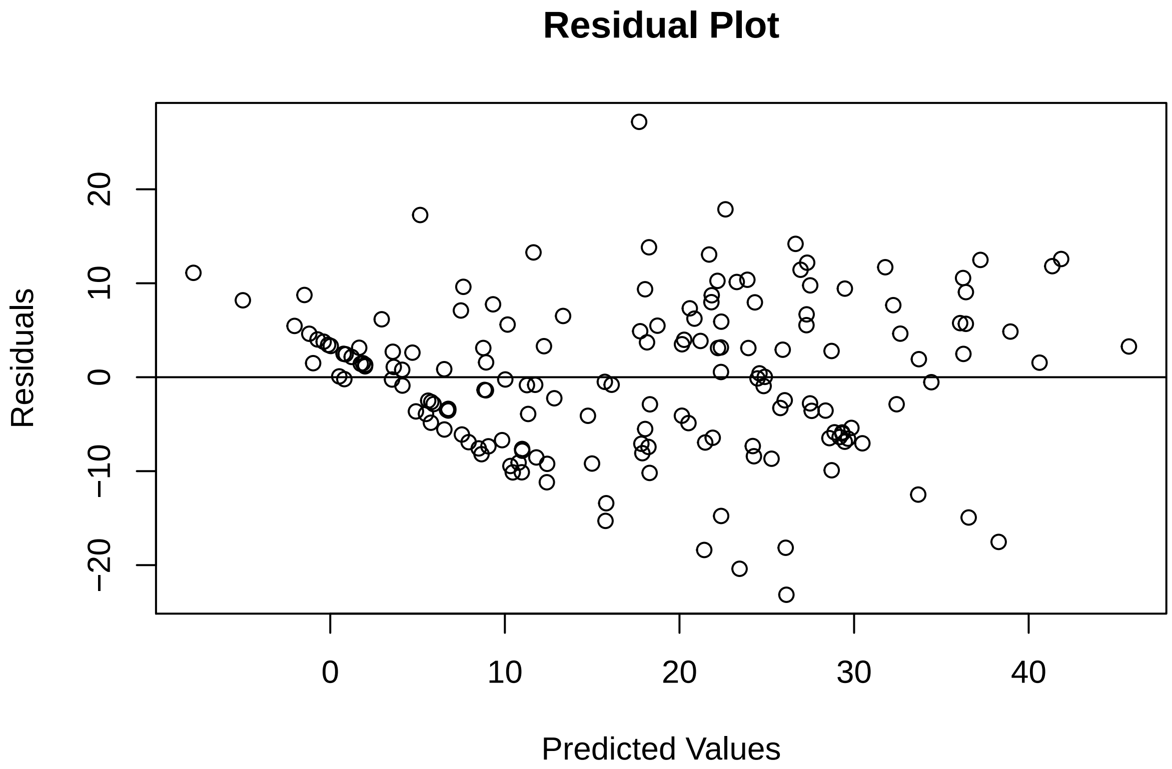
a)

After randomly selecting 10 variables and running regression with OLS, I created QQ plot and residual plot like below and saw no strange pattern in the plots that warrant a transformation of our data. However that could be caused by the randomness in the way I chose the 10 variables, and since we could see that our Y (glucose concentration) is skewed (see histogram), I still suggest we transform Y, for example taking a square root of Y, to mitigate the skewness. As we can see from the second histogram below, the distribution of sqrt(Y) looks more normal.

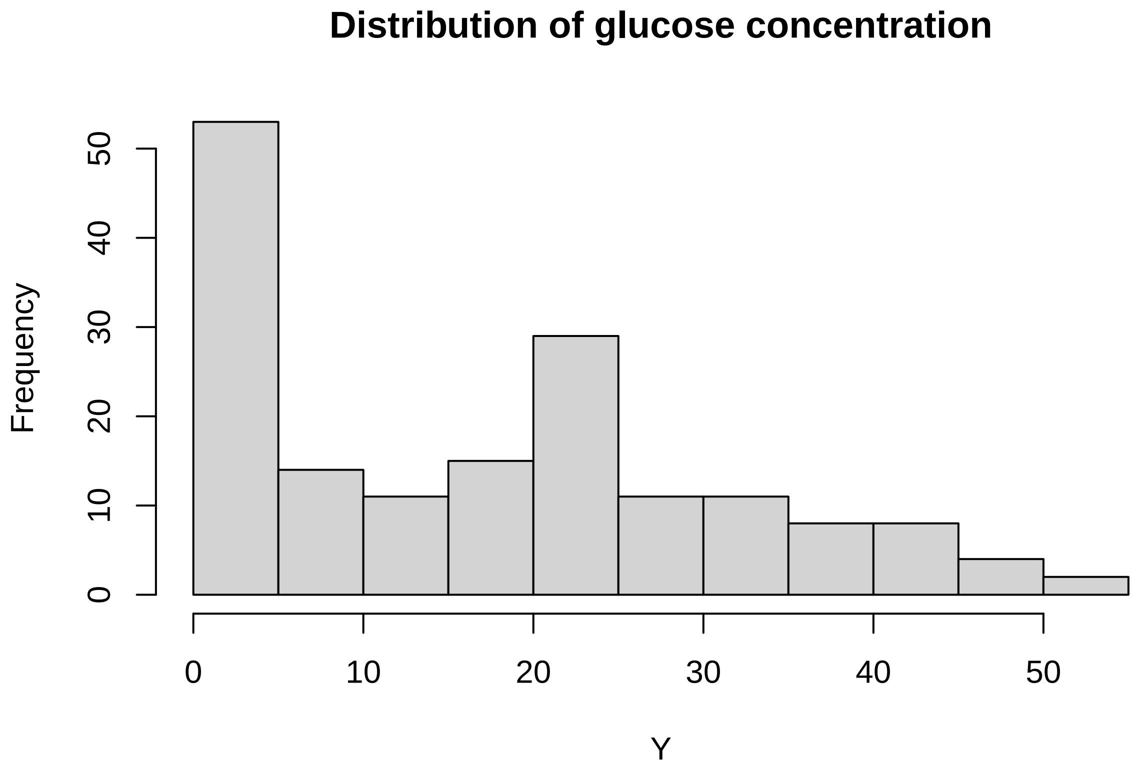
```
#read data
data(NIR)

#randomly choose 10 predictors
set.seed(101)
sp10a <- sample(NIR$xNIR,size = 10)
sp10a$Y <- NIR$yGlcEtOH[,1]

#Regress y with sample
fit10a <- lm(Y~.,data = sp10a)
sum10a <- summary(fit10a)

qqnorm(rstandard(fit10a),
       ylab="Standardized Residuals",
       xlab="Normal Scores")
qqline(rstandard(fit10a))
```



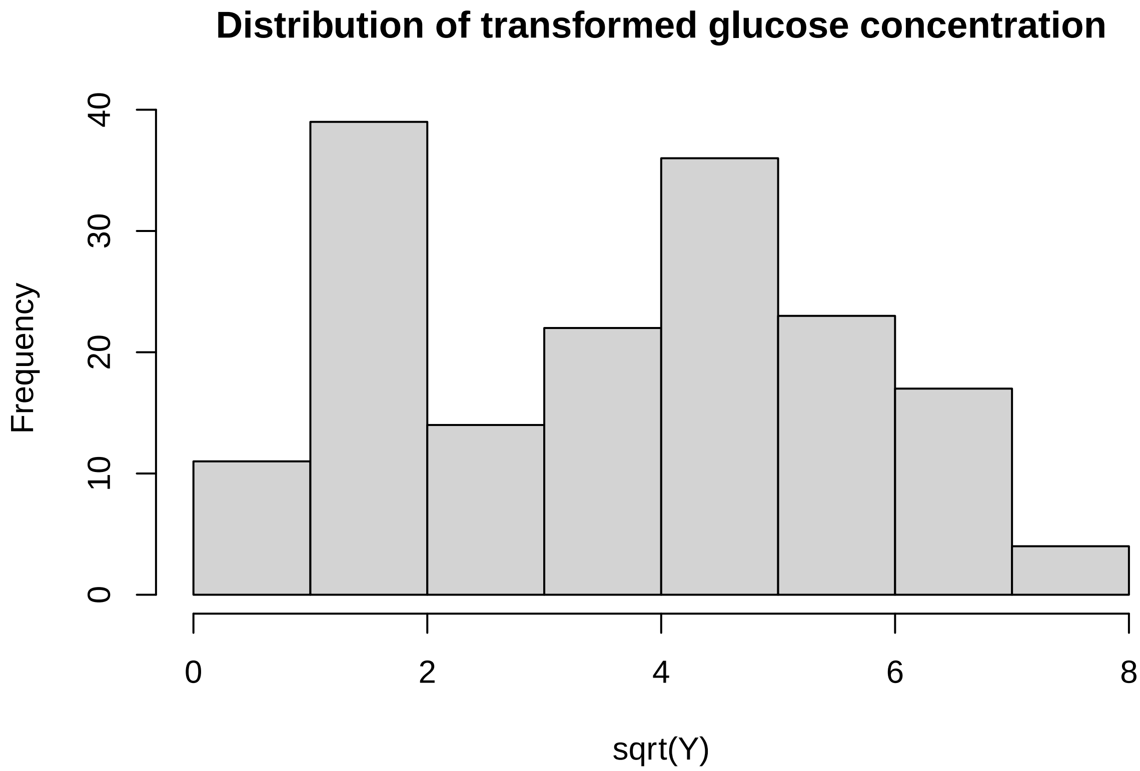
```r
plot(predict(fit10a), resid(fit10a),
     ylab  = "Residuals", xlab = "Predicted Values", main ="Residual Plot")
abline(0,0)
```



```
##### should we make transformations?
hist(NIR$yGlcEtOH[,1], xlab = "Y", main = "Distribution of glucose concentration")
```



```
hist(sqrt(NIR$yGlcEtOH[,1]),xlab = "sqrt(Y)", main = "Distribution of transformed glucose concentration
```



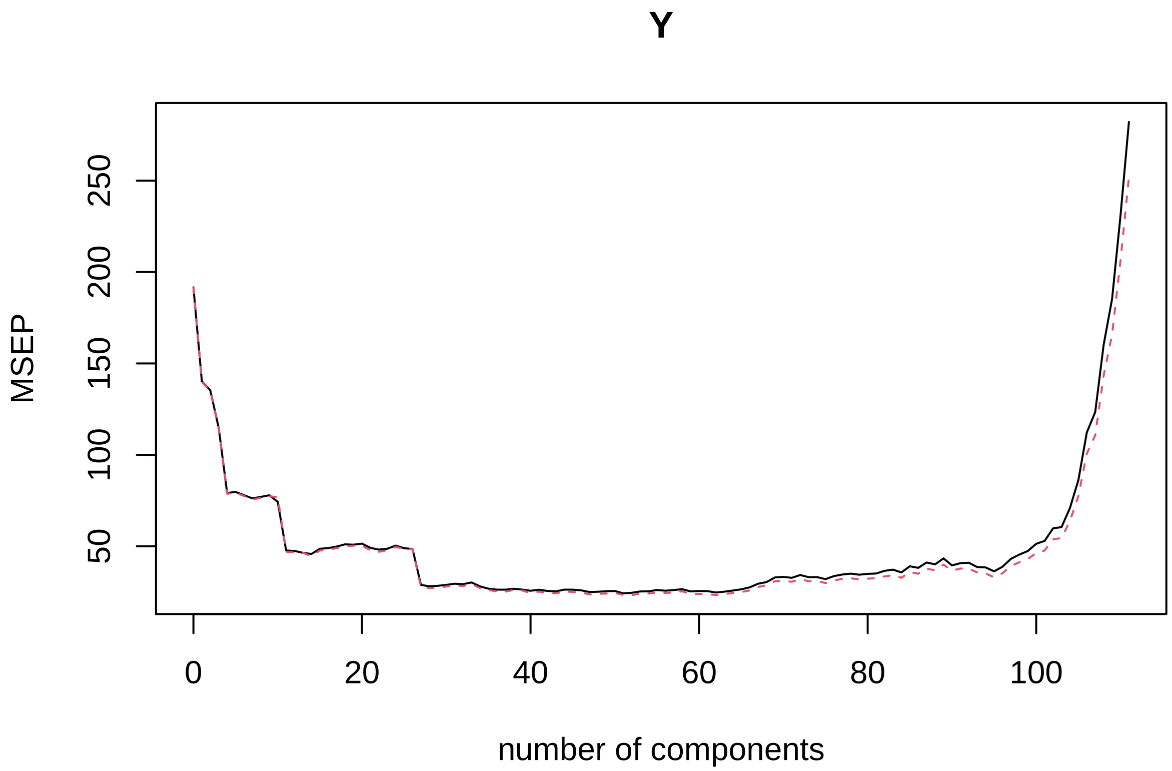
b)
With Y being skewed, using the same reasoning with previous part, we can't simply use OLS to estimate the parameters for the model specified. I suggest 4 other methods, they are PCR, PLS, Lasso and Ridge.

c)
Principal component regression with 9 fold cross validation.
(i)
The K that minimizes CV MSE is 50, minimized CV MSE is 34.80.

```
#permute
set.seed(1968)
dat10full <- NIR$xNIR
dat10full$Y <- NIR$yGlcEtOH[,1]
perm10 <- dat10full[sample(nrow(dat10full)),]

#training and test sets
train10 <- perm10[1:126,]
test10 <- anti_join(perm10,train10)
```

```
## Joining, by = c("X1115.0", "X1120.0", "X1125.0", "X1130.0", "X1135.0", "X1140.0", "X1145.0", "X1150.
```

```
#PCR with 9-fold CV
set.seed(102)
pcr10c1 <- pcr(Y~., data = train10, scale = TRUE, validation = "CV",segments=9)

#i)
validationplot(pcr10c1, val.type = "MSEP")
```



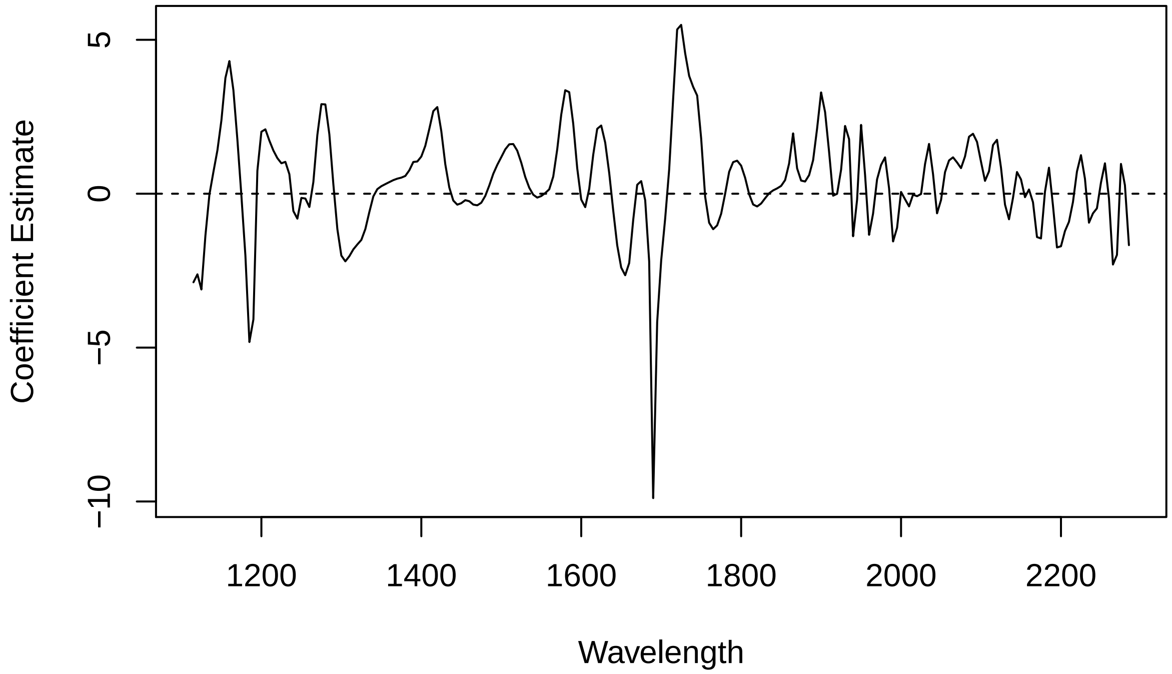
```
pcr10c1_mse <-MSEP(pcr10c1)
which.min(pcr10c1_mse$val[1,1,]) #K hat is 50
```

```
## 51 comps
##       52
```

(ii)

Plot estimates for beta as function of wavelength lambda with K chosen to be 50 from previous part. We can see how the coefficient estimates fluctuate with increasing wavelength values.

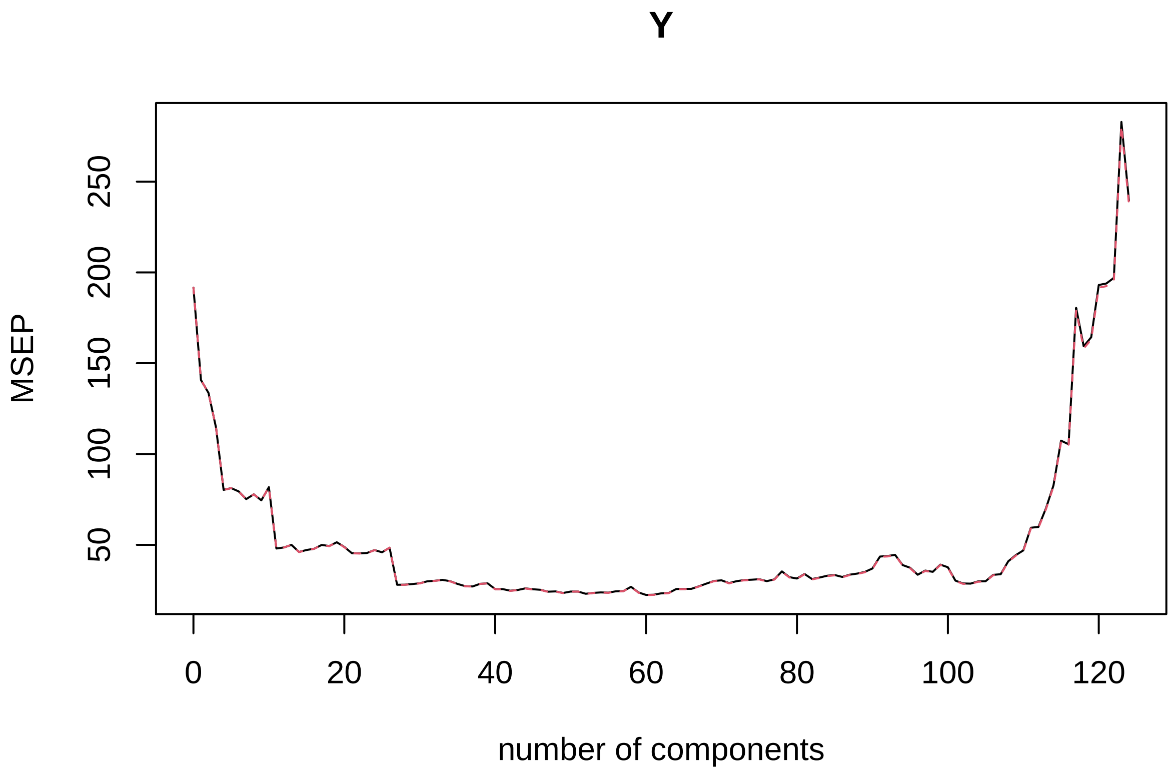
```
#with K hat = 50, get the coefficient estimates
coef10c2 <- data.frame(betaH=pcr10c1$coefficients[, , 50])
coef10c2$X <- as.numeric(substring(row.names(coef10c2),2,5))
plot(coef10c2$X, coef10c2$betaH,type="l", xlab = "Wavelength",ylab = "Coefficient Estimate")
abline(h = 0,lty=2)
```



(iii)

Repeat part (i) but with LOO-CV. Now K hat is 49, CV MSE is 30.98, which is smaller than 9-fold CV.

```
#PCR with 9-fold CV
set.seed(103)
pcr10c3 <- pcr(Y~., data = train10, scale = TRUE, validation = "LOO")

#iii)
validationplot(pcr10c3, val.type = "MSEP")
```

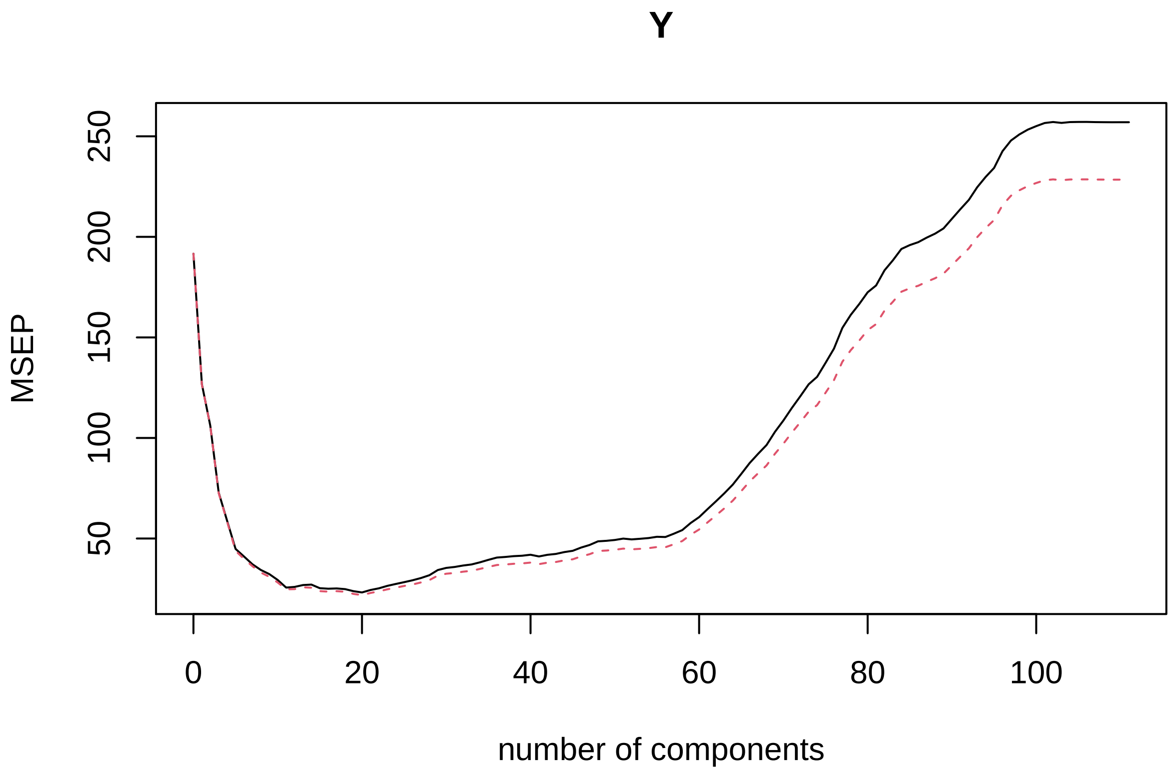


```
pcr10c3_mse <-MSEP(pcr10c3)
which.min(pcr10c3_mse$val[1,1,]) #K hat is 50
```

```
## 60 comps
##       61
```

d)
Partial Least Squares with 9 fold cross validation.
(i)
The K that minimizes CV MSE is 11, minimized CV MSE is 32.36.

```
#PLS with 9-fold CV
set.seed(104)
pcr10d1 <- plsr(Y~., data = train10, scale = TRUE, validation = "CV",segments=9)

#i)
validationplot(pcr10d1, val.type = "MSEP")
```



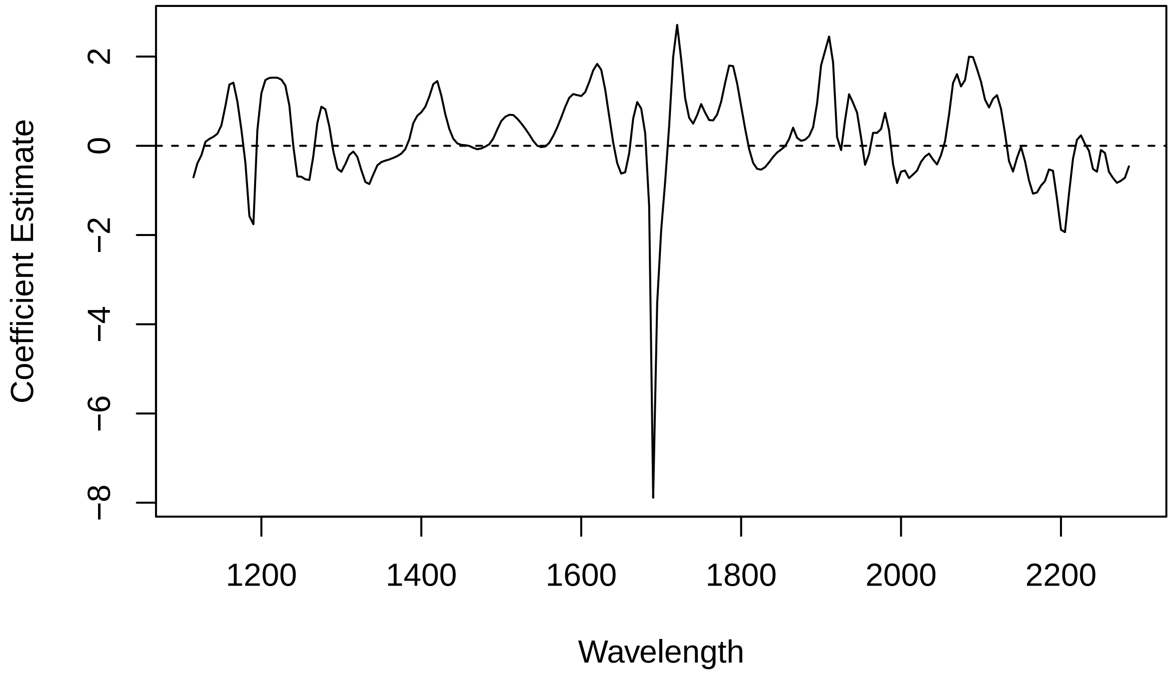
```
pcr10d1_mse <-MSEP(pcr10d1)
which.min(pcr10d1_mse$val[1,1,]) #K hat is 11
```

```
## 20 comps
##       21
```

(ii)

Plot estimates for beta as function of wavelength lambda with K chosen to be 50 from previous part. We can see how the coefficient estimates fluctuate with increasing wavelength values, overall same general trend compared to when we used PCR in part (c).

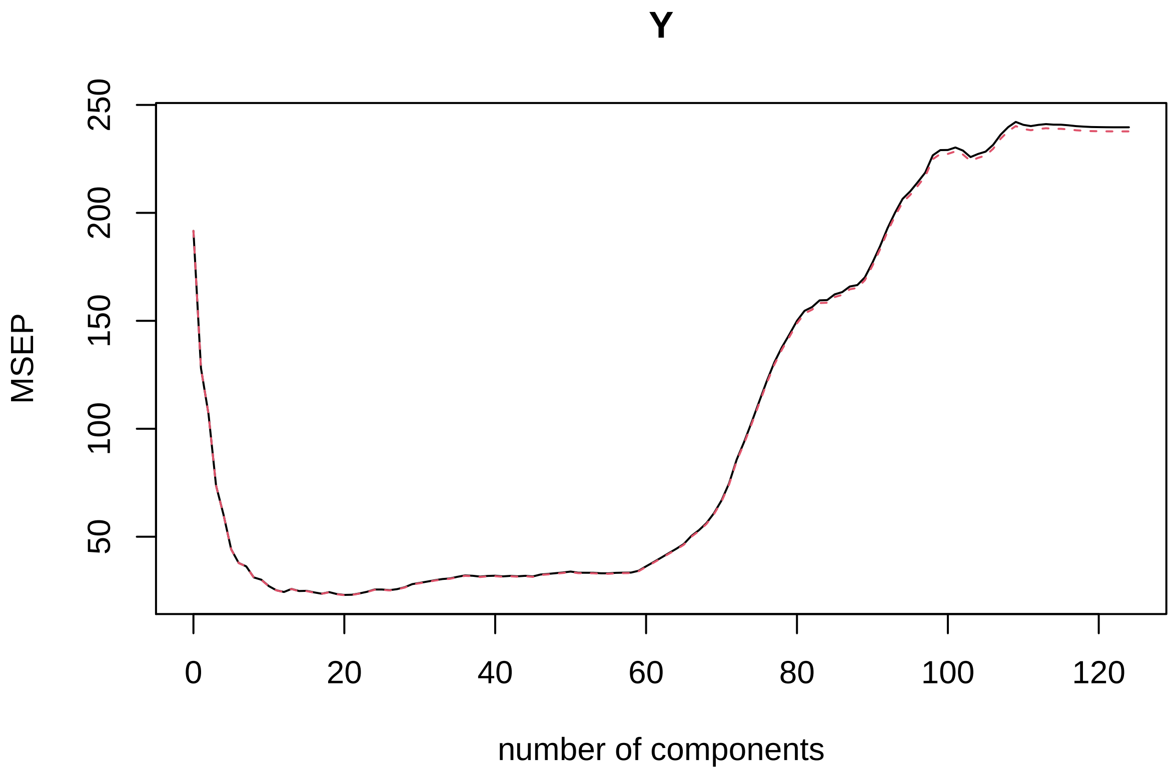
```
#with K hat = 11, get the coefficient estimates
coef10d2 <- data.frame(betaH=pcr10d1$coefficients[, , 11])
coef10d2$X <- as.numeric(substring(row.names(coef10d2),2,5))
plot(coef10d2$X, coef10d2$betaH,type="l", xlab = "Wavelength",ylab = "Coefficient Estimate")
abline(h = 0,lty=2)
```



(iii)

Repeat part (i) but with LOO-CV. Now K hat is 18, CV MSE is 33.09, which is larger than 9-fold CV, also involves more components than 9-fold CV as well.

```
#PLS with 9-fold CV
set.seed(105)
pcr10d3 <- plsr(Y~., data = train10, scale = TRUE, validation = "LOO")

#iii)
validationplot(pcr10d3, val.type = "MSEP")
```



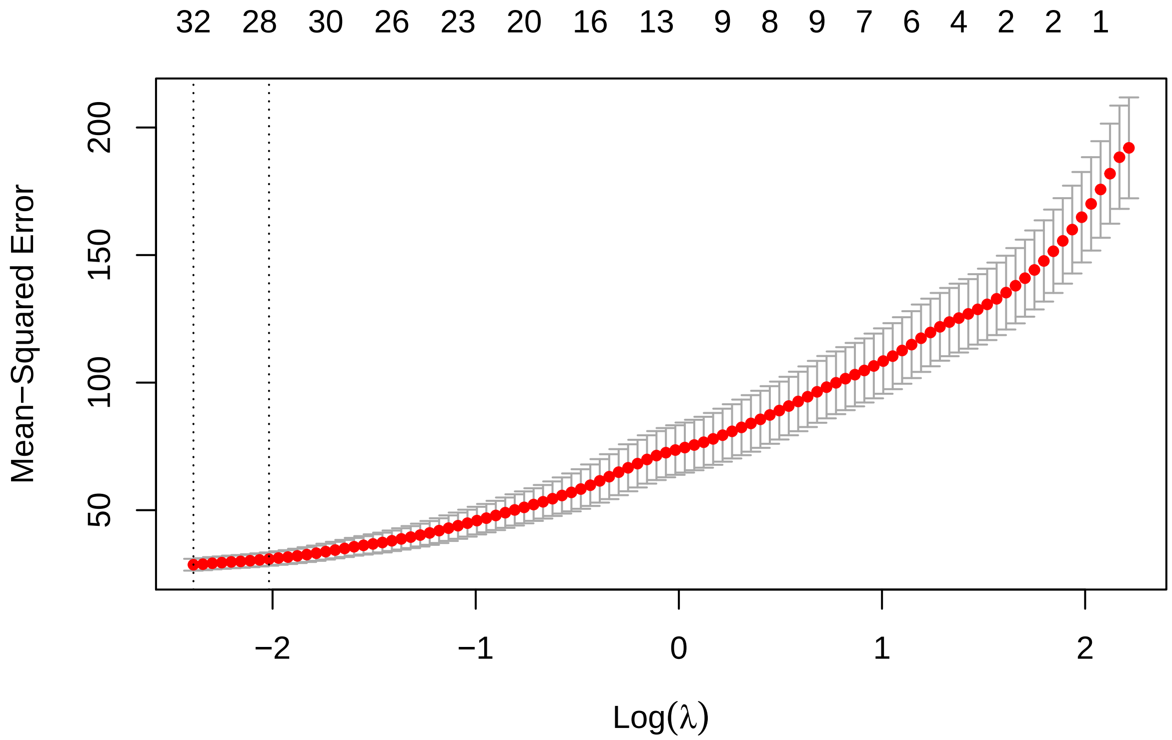
```
pcr10d3_mse <-MSEP(pcr10d3)
which.min(pcr10d3_mse$val[1,1,]) #K hat is 50
```

```
## 20 comps
##       21
```

e)

Lasso with 9 fold cross validation. Optimal lambda chosen is 0.08298233. Minimized MSE is 38.77573.

```
#Lasso with 9-fold CV
#glmnet??
train10X <- as.matrix(train10[,-236])
train10Y <- as.matrix(train10$Y)
lasso10e=glmnet(train10X,train10Y,alpha=1,lambda=seq(0, 0.1, 0.0001))
set.seed(106)
cv.out10e=cv.glmnet(train10X,train10Y,alpha=1,nfolds = 9)
plot(cv.out10e)
```



```
bestlam10e=cv.out10e$lambda.min
bestlam10e
```

```
## [1] 0.09166438
```

```
cv.out10e$cvm[which.min(cv.out10e$cvm)]
```

```
## [1] 28.61636
```

f)
Using test set to evaluate predictive performance, PCR has MSE of 27.4641, PLS has MSE of 34.05982, and Lasso has MSE of 30.895. Loss function overestimates for PCR and Lasso, but underestimates for PLS.

```
#PCR with comp = 49
pcr10c_pred = predict(pcr10c3, test10[,-236], ncomp=49)
mean((pcr10c_pred-test10$Y)^2)
```

```
## [1] 48.19426
```

```
#PLS with comp = 11
pcr10d_pred = predict(pcr10d1, test10[,-236], ncomp=11)
mean((pcr10d_pred-test10$Y)^2)
```

```
## [1] 51.45569
```

```
#Lasso
lasso10e.pred=predict(lasso10e,s=bestlam10e,newx=as.matrix(test10[,-236]))
mean((lasso10e.pred-test10$Y)^2)
```

```
## [1] 60.5662
```