hw1 DS2

```
setwd("/Users/siyanchen/Desktop/data")
train = read_csv("solubility_train.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double()
## )
```

```
## See spec(...) for full column specifications.
```

```
test = read_csv("solubility_test.csv")
```

```
## Parsed with column specification:
## cols(
##   .default = col_double()
## )
## See spec(...) for full column specifications.
```

```
x_train = model.matrix(Solubility ~ ., train)[,-1]
y_train = train$Solubility
x_test = model.matrix(Solubility ~ ., test)[,-1]
y_test = test$Solubility
```

a)

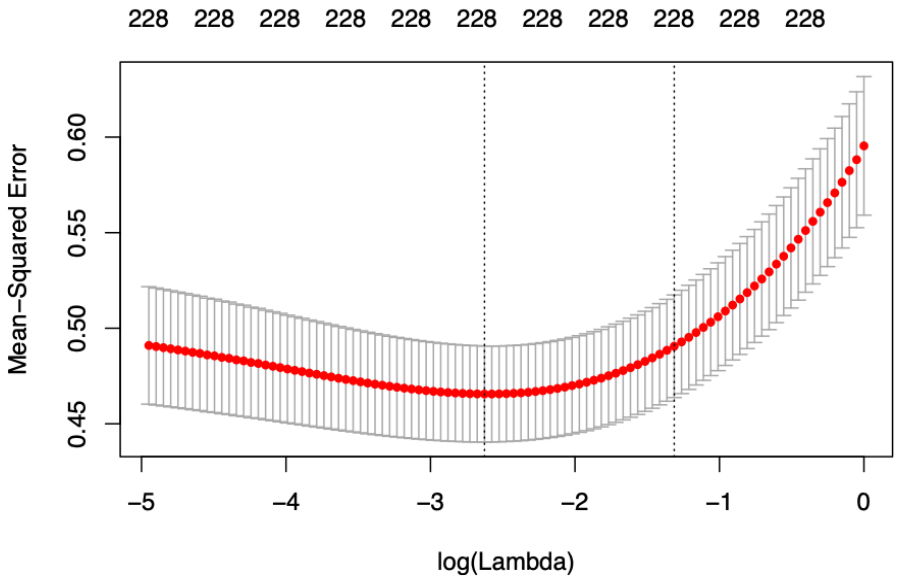
```
ls_model = lm(Solubility ~ ., data = train)
train_mse = mean(ls_model$residuals^2)
test_mse = mean((test$Solubility-predict(ls_model,test))^2)
test_mse
```

```
## [1] 0.5558898
```

MSE is 0.5558898 if fit model using least squares.

b)

```
set.seed(2)
ridge_mod = cv.glmnet(x_train, y_train,
                      alpha = 0,
                      lambda = exp(seq(-5, 0, length = 100)),
                      type.measure = "mse")
plot(ridge_mod)
```




```
ridge_mod$lambda.min
```

```
## [1] 0.07234835
```

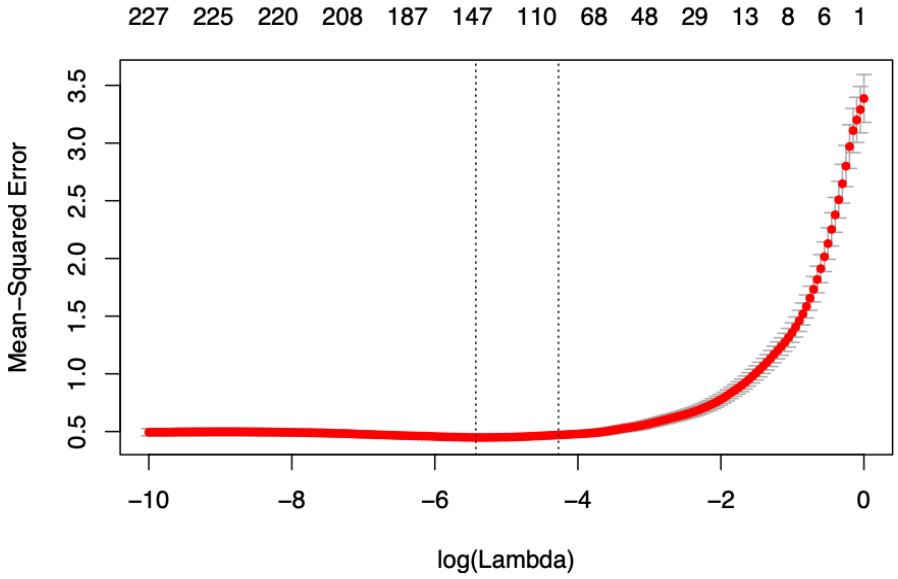
```
c_predict = predict(ridge_mod, s  = ridge_mod$lambda.min, type = "coefficients")

#mse
ridge_predict = predict(ridge_mod, s = ridge_mod$lambda.min, newx = x_test )
mean((ridge_predict - y_test)^2)
```

```
## [1] 0.5118012
```

For ridge regression model, lambda is set to be 0.07234835 by cross-validation. Test error is 0.5118012.

c)

```
set.seed(2)
cv.lasso = cv.glmnet(x_train, y_train,
                     alpha = 1,
                     lambda = exp(seq(-10, 0, length = 200)))
plot(cv.lasso)
```




```
cv.lasso$lambda.min
```

```
## [1] 0.004395668
```

```
coef_preidict = predict(cv.lasso, s = cv.lasso$lambda.min, type = "coefficients")
# mse
lasso_predict = predict(cv.lasso, s = cv.lasso$lambda.min, newx = x_test)
mean((lasso_predict - y_test)^2)
```

```
## [1] 0.5003869
```

```
# number of nonzero coefficients
coefs = coef_preidict[which(coef_preidict != 0 ) ]
features = coef_preidict@Dimnames[[1]][which(coef_preidict != 0 )]
results = data.frame(
  features, #intercept included
  coefs)
nrow(results)
```

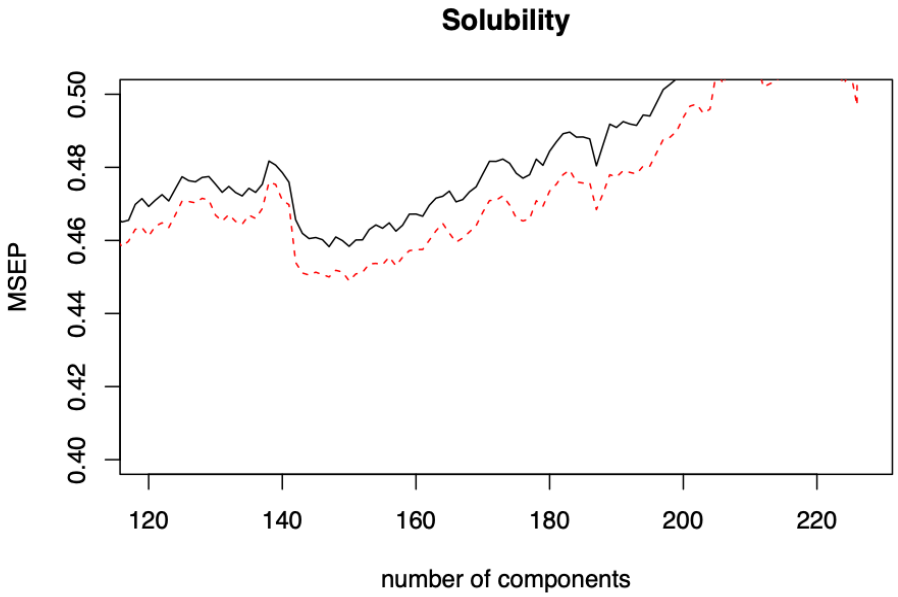
```
## [1] 144
```

For lasso model on the training data, minimum lambda is 0.004395668. Test error is 0.5003869. The number of non-zero coefficient estimates is 142 (including intercept)

d)

```
set.seed(4)
pcr_mod = pcr(Solubility~.,
              data = train,
              sclae = TRUE,
              validation = "CV")
```

```
validationplot(pcr_mod, val.type = "MSEP", xlim = c(120, 227), ylim = c(0.4,0.5))
```




```
pred_pcr = predict(pcr_mod, newdata = x_test, ncomp = 141)
mean((y_test - pred_pcr)^2)
```

```
## [1] 0.5159756
```

According to the plot, we choose the number of principal component to be 141. Test error is 0.5159756.

e

Based on the test error of different models, lasso model has smallest MSE and least square model has greatest MSE.Simple linear model is simplest model therefore fitting data not well compared to other models. Lasso model fits data well and there are mutiple coefficients shrinks to 0.