Homework 1

Jingyi Zhang

2/6/21

Load dataset

```
sol_test <-
  read_csv("./data/solubility_test.csv") %>%
  janitor::clean_names()

sol_train <-
  read_csv("./data/solubility_train.csv") %>%
  janitor::clean_names()

# training data
x_train <- model.matrix(solubility ~ ., sol_train)[ ,-1]
y_train <- sol_train$solubility

# test data
x_test <- model.matrix(solubility ~ ., sol_test)[ ,-1]
y_test <- sol_test$solubility
```

(a) Fit a linear model using least squares on the training data and calculate the mean squared error using the test data.

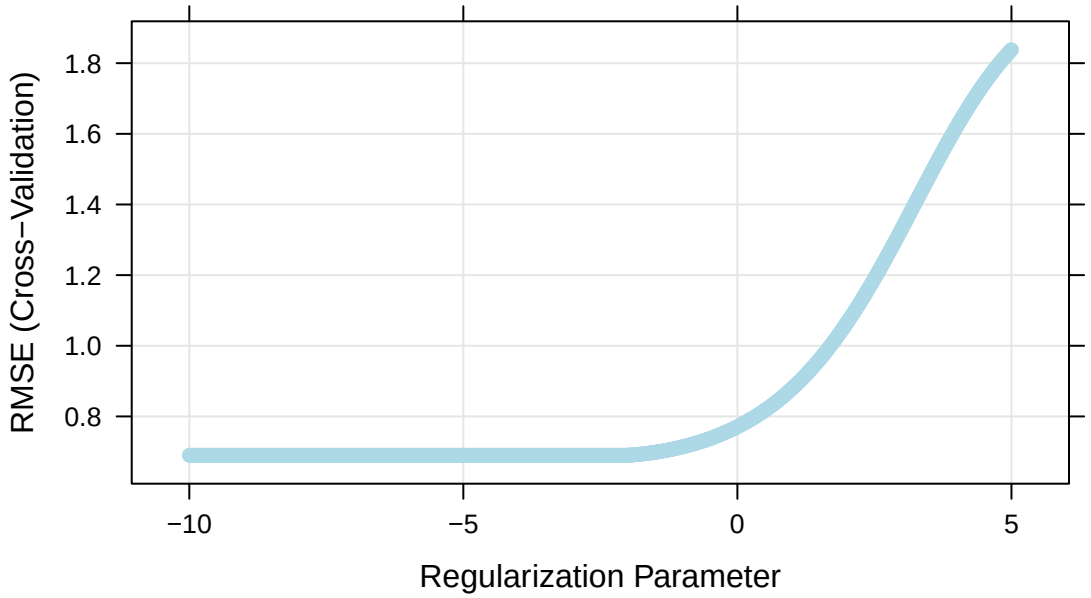
```
set.seed(7)
ctrl1 <- trainControl(method = "cv", number = 10)
lm.fit <- train(solubility~.,
                data = sol_train,
                method = "lm",
                trControl = ctrl1)
RMSE(predict(lm.fit, newdata = sol_test), sol_test$solubility)
```

```
## [1] 0.7455802
```

(b) Fit a ridge regression model on the training data, with lambda chosen by cross-validation. Report the test error.

```
set.seed(7)
ridge.fit <-
  train(solubility~.,
        data = sol_train,
```

```
        method = "glmnet",
        tuneGrid =
          expand.grid(alpha = 0,
                      lambda = exp(seq(5, -10, length = 1000))),
        trControl = ctrl1,
        preProcess = c("center", "scale"))

plot(ridge.fit, xTrans = log, pch = 1, col = "light blue")
```




```
ridge.fit$bestTune
```

```
##     alpha    lambda
## 532     0 0.1317266
```

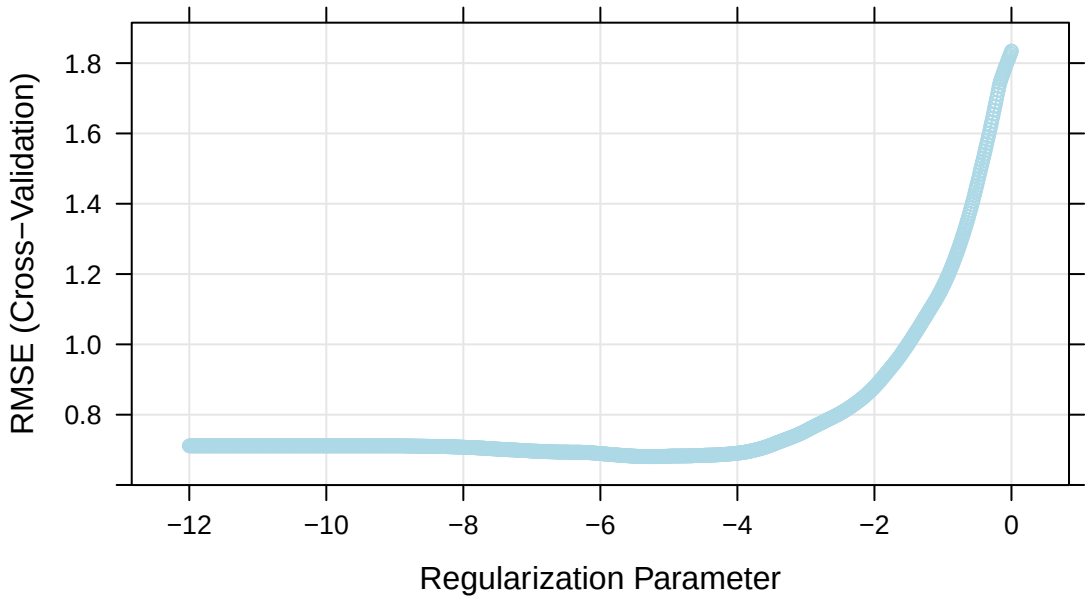
```
RMSE(predict(ridge.fit, s = "lambda.min", newx = sol_test), sol_test$solubility)
```

```
## [1] 2.929737
```

(c) Fit a lasso model on the training data, with lambda chosen by cross-validation. Report the test error and the number of non-zero coefficient estimates in your model.

```
set.seed(7)
lasso.fit <-
  train(solubility~.,
        data = sol_train,
        method = "glmnet",
        tuneGrid =
          expand.grid(alpha = 1,
                      lambda = exp(seq(0, -12, length = 1000))),
        trControl = ctrl1,
        preProcess = c("center", "scale"))
```

```
plot(lasso.fit, xTrans = log, col = "light blue", pch = 1)
```




```
lasso.fit$bestTune
```

```
##     alpha      lambda
## 565     1 0.005379148
```

```
RMSE(predict(lasso.fit, s = "lambda.min", newx = sol_test), sol_test$solubility)
```

```
## [1] 2.945769
```

```
sum(coef(lasso.fit$finalModel, lasso.fit$bestTune$lambda) != 0)
```

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

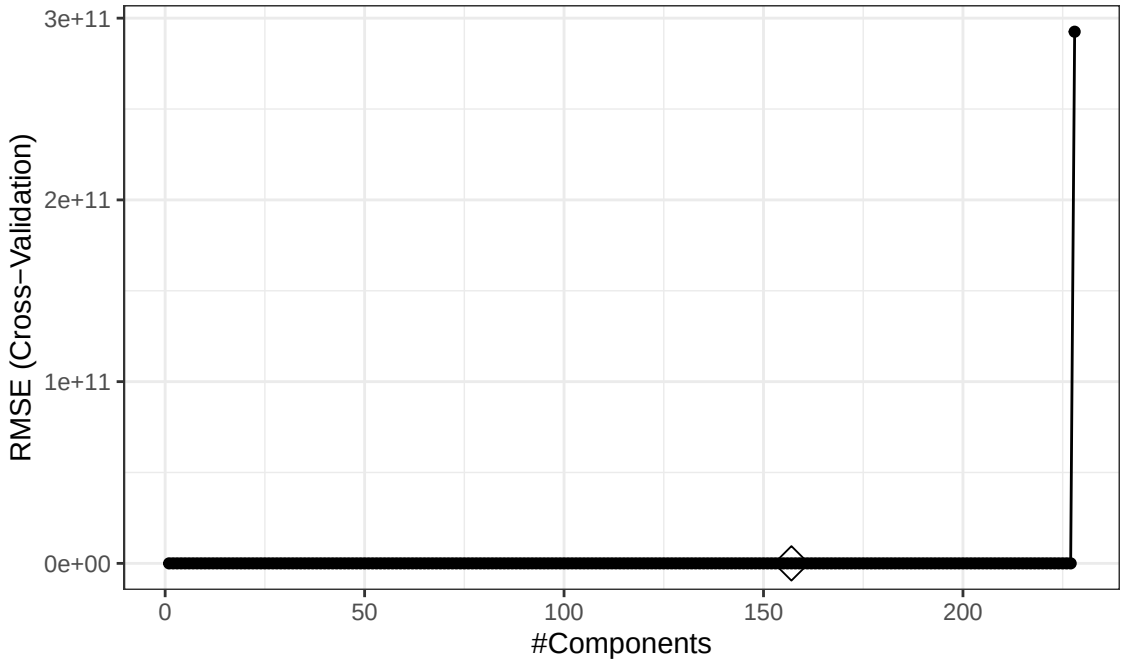
(d) Fit a principle component regression model on the training data, with M chosen by cross-validation. Report the test error and the value of M selected by cross-validation.

```
ctrl2 <- trainControl(method = "cv", selectionFunction = "best")

set.seed(7)
pcr.fit <-
  train(solubility~.,
        data = sol_train,
        method = "pcr",
        tuneGrid =
          data.frame(ncomp = seq(1, ncol(x_train))),
        trControl = ctrl2,
        preProcess = c("center", "scale"))

pcr.fit$bestTune # value of selected M
```

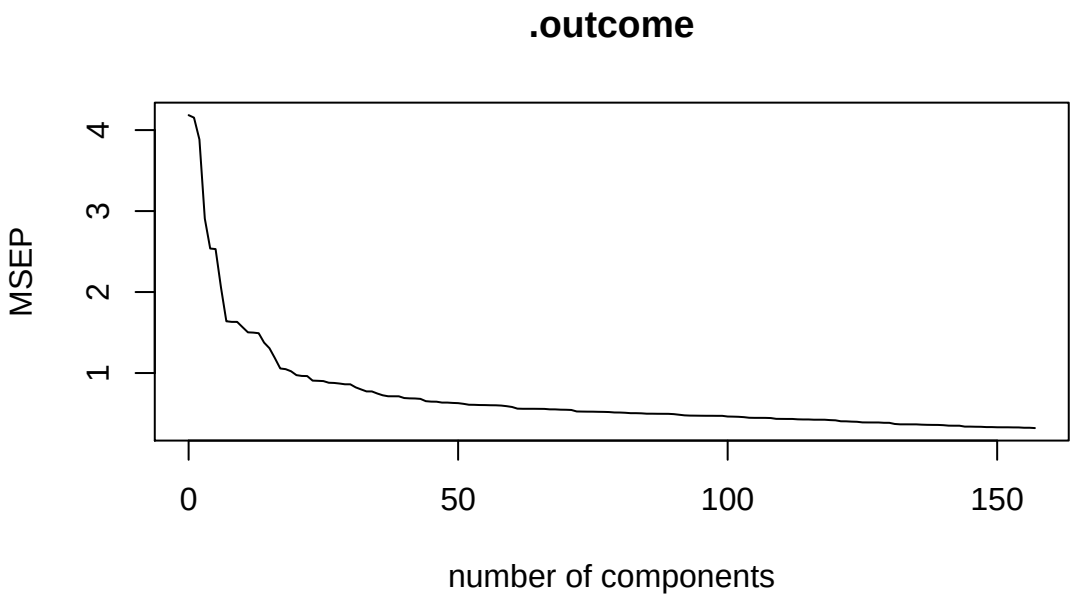
```
##     ncomp
```

```
## 157   157
```

```
ggplot(pcr.fit, highlight = TRUE) + theme_bw() # hard to interpret
```




```
validationplot(pcr.fit$finalModel, val.type = "MSEP")
```




```
predy2.pcr2 <- predict(pcr.fit, newdata = x_test)
mean((y_test - predy2.pcr2)^2) # test MSE
```

```
## [1] 0.549917
```

```
RMSE(predict(pcr.fit, x_test), y_test) # RMSE
```

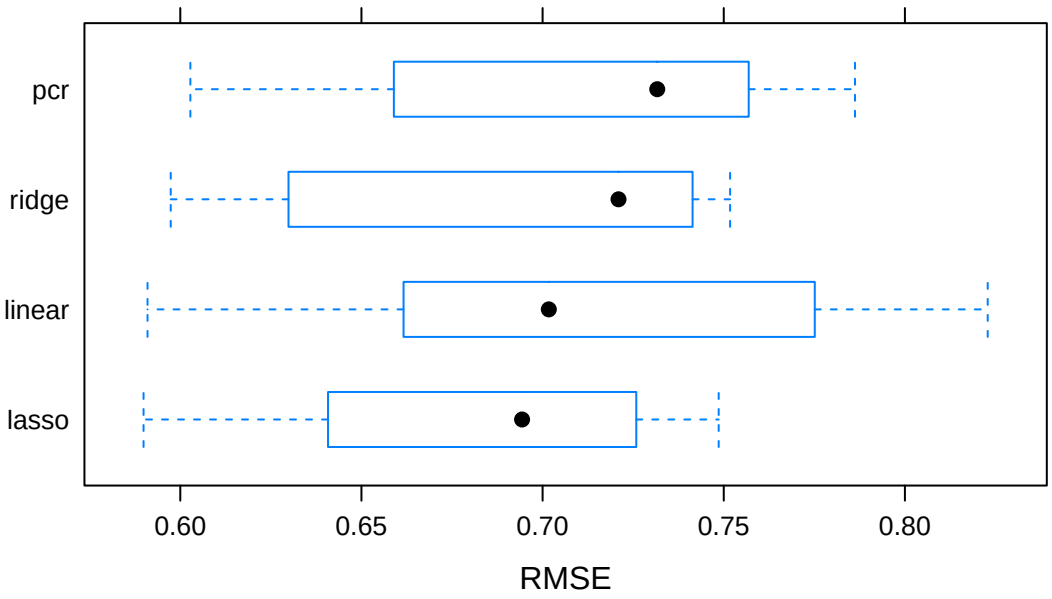
```
## [1] 0.7415639
```

(e) Which model will you choose for predicting solubility?

```
resamp <- resamples(list(
                        linear = lm.fit,
                        ridge = ridge.fit,
                        lasso = lasso.fit,
                        pcr = pcr.fit))
summary(resamp)
```

```
##
## Call:
## summary.resamples(object = resamp)
##
## Models: linear, ridge, lasso, pcr
## Number of resamples: 10
##
## MAE
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## linear 0.4593506 0.5065095 0.5411760 0.5344250 0.5621513 0.5859706    0
## ridge  0.4668550 0.4842955 0.5407178 0.5266765 0.5569412 0.5790663    0
## lasso  0.4525769 0.5013758 0.5336942 0.5218328 0.5459192 0.5795652    0
## pcr    0.4656072 0.5114016 0.5661166 0.5496161 0.5754840 0.6400421    0
##
## RMSE
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## linear 0.5909568 0.6670690 0.7017143 0.7105437 0.7616208 0.8228557    0
## ridge  0.5973602 0.6364183 0.7209517 0.6899735 0.7373330 0.7517499    0
## lasso  0.5898579 0.6472640 0.6943397 0.6805959 0.7227745 0.7486055    0
## pcr    0.6027874 0.6742055 0.7316448 0.7123739 0.7559649 0.7862302    0
##
## Rsquared
##             Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## linear 0.8217876 0.8718251 0.8843464 0.8798403 0.9035411 0.9143856    0
## ridge  0.8469126 0.8680781 0.8851333 0.8862387 0.9084998 0.9200666    0
## lasso  0.8505948 0.8737344 0.8867085 0.8893925 0.9120246 0.9230088    0
## pcr    0.8412169 0.8662138 0.8763487 0.8790284 0.8972487 0.9220060    0
```

```
bwplot(resamp, metric = "RMSE")
```



Based on the numeric and graph outputs, lasso model has the smallest mean RMSE among all four models. Lasso will be chosen for predicting solubility.