

# Homework 1

Yiying Wu (yw3996)

## R packages

```
library(tidyverse)
library(caret)
library(tidymodels)
```

## Input dataset

```
housing_train<-read_csv("./data/housing_training.csv")
housing_train <- na.omit(housing_train)
housing_test<-read_csv("./data/housing_test.csv")
housing_test <- na.omit(housing_test)
```

Response: Sale price

(a) Fit a lasso model on the training data. Report the selected tuning parameter and the test error. When the 1SE rule is applied, how many predictors are included in the model?

```
ctrl1 <- trainControl(method = "cv", number = 10,
                      selectionFunction = "oneSE")

# Lasso
set.seed(8106)
lasso.fit <- train(Sale_Price ~ .,
                   data = housing_train,
                   method = "glmnet",
                   tuneGrid = expand.grid(alpha = 1,
                                          lambda = exp(seq(10, 0, length = 200))),
                   trControl = ctrl1)
# plot(lasso.fit, xTrans = log)
```

Here's the selected tuning parameter when 1SE rule is applied

```
lasso.fit$bestTune
```

```
##      alpha  lambda
## 142      1 1194.433
```

The best tuning parameter is 1194.433

And the test error is

```
lasso.pred <- predict(lasso.fit, newdata = housing_test)
# test error
mean((lasso.pred - housing_test$Sale_Price)^2)
```

```
## [1] 428517009
```

MSE= $4.2851701 \times 10^8$

coefficients in the final model are

```
# coefficients in the final model
coef(lasso.fit$finalModel, lasso.fit$bestTune$lambda)
```

```
## 40 x 1 sparse Matrix of class "dgCMatrix"
##                               s1
## (Intercept)                -6.308774e+05
## Gr_Liv_Area                  5.365061e+01
## First_Flr_SF                 1.318967e+00
## Second_Flr_SF                .
## Total_Bsmt_SF               3.670034e+01
## Low_Qual_Fin_SF             -1.914919e+01
## Wood_Deck_SF                7.582525e+00
## Open_Porch_SF               5.507913e+00
## Bsmt_Unf_SF                 -1.799497e+01
## Mas_Vnr_Area                 1.460328e+01
## Garage_Cars                  3.118493e+03
## Garage_Area                  1.220022e+01
## Year_Built                   3.237812e+02
## TotRms_AbvGrd               -9.424521e+01
## Full_Bath                    .
## Overall_QualAverage          -2.551515e+03
## Overall_QualBelow_Average    -7.804630e+03
## Overall_QualExcellent         8.736996e+04
## Overall_QualFair             -3.907651e+03
## Overall_QualGood              8.564734e+03
## Overall_QualVery_Excellent   1.557633e+05
## Overall_QualVery_Good        3.441309e+04
## Kitchen_QualFair             -3.154024e+03
## Kitchen_QualGood              .
## Kitchen_QualTypical          -9.951405e+03
## Fireplaces                   7.024303e+03
## Fireplace_QuFair              .
## Fireplace_QuGood              3.983928e+03
## Fireplace_QuNo_Fireplace      .
## Fireplace_QuPoor              .
## Fireplace_QuTypical           .
## Exter_QualFair               -1.270689e+04
## Exter_QualGood                .
## Exter_QualTypical            -5.671510e+03
## Lot_Frontage                 5.488407e+01
```

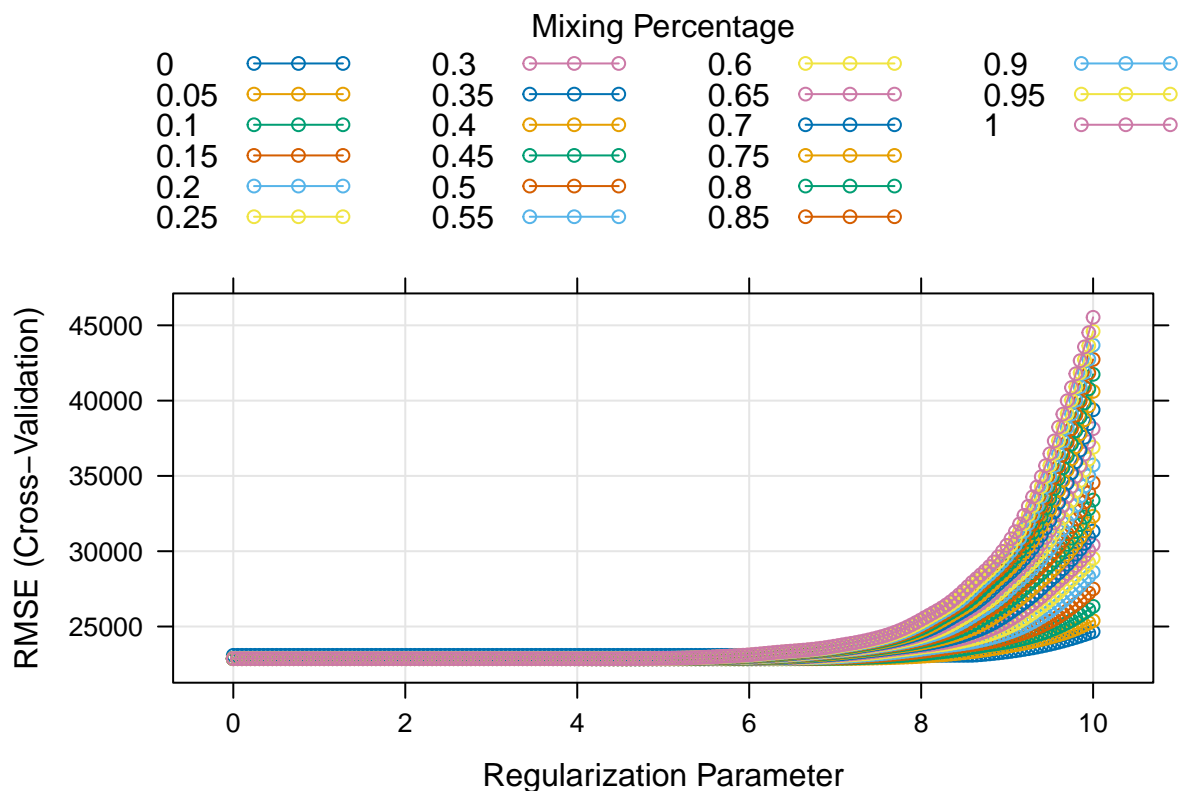
```
## Lot_Area          5.192928e-01
## Longitude        -2.114214e+02
## Latitude         3.862327e+02
## Misc_Val         .
## Year_Sold         .
```

Therefore, there are 29 predictors included in the model.

(b) Fit an elastic net model on the training data. Report the selected tuning parameters and the test error. Is it possible to apply the 1SE rule to select the tuning parameters for elastic net? If the 1SE rule is applicable, implement it to select the tuning parameters. If not, explain why.

```
set.seed(8106)
enet.fit <- train(Sale_Price ~ .,
  data = housing_train,
  method = "glmnet",
  tuneGrid = expand.grid(alpha = seq(0, 1, length = 21),
    lambda = exp(seq(10, 0, length = 200))),
  trControl = ctrl1)

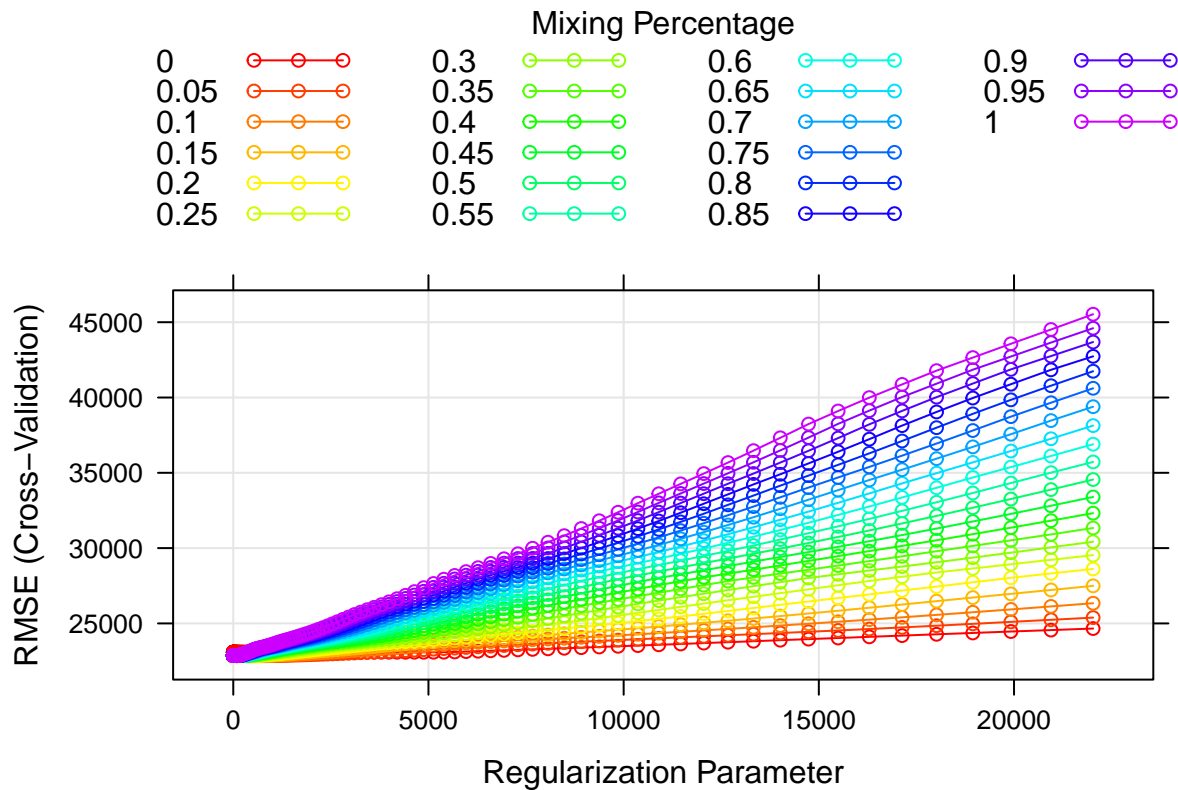
plot(enet.fit, xTrans = log)
```



```
enet.fit$bestTune
```

```
##      alpha  lambda
## 190      0 13326.2

myCol <- rainbow(25)
myPar <- list(superpose.symbol = list(col = myCol),
              superpose.line = list(col = myCol))
plot(enet.fit, par.settings = myPar)
```



- (c) Fit a partial least squares model on the training data and report the test error. How many components are included in your model?
- (d) Choose the best model for predicting the response and explain your choice.
- (e) If “caret” was used for the elastic net in (b), retrain this model with “tidy-models”, and vice versa. Compare the selected tuning parameters between the two software approaches. Should there be discrepancies in the chosen parameters, discuss potential reasons for these differences.