LASSO Regression and Ridge Regression

Contents

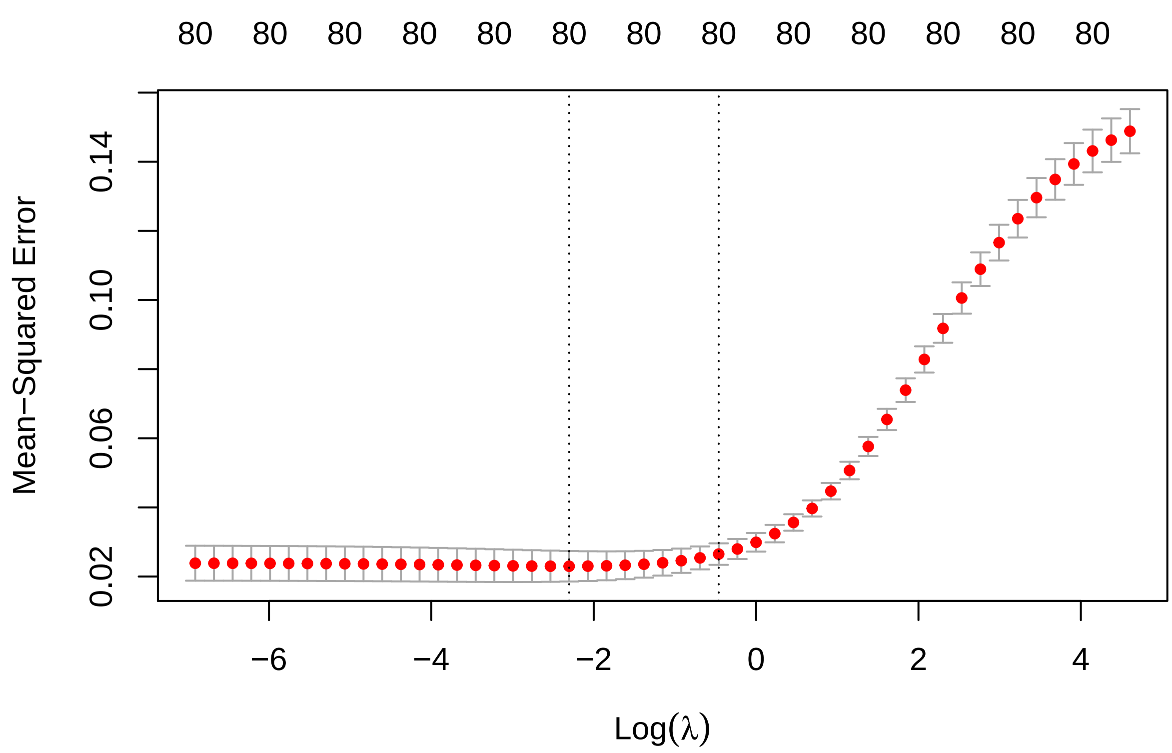
Data Source: House Prices https://www.kaggle.com/c/house-prices-advanced-regression-techniques/data

```
train <- read.csv("train_cleaned.csv", header = TRUE)
test <- read.csv("test_cleaned.csv", header = TRUE)
```

1 Ridge Regression

```
library(glmnet)
set.seed(123)
train.x <- train[,names(train) != "SalePrice"]
x <- data.matrix(train.x)
y <- log(train$SalePrice)
lambdas <- 10^seq(2, -3, by = -.1)

## Fit a ridge regression model
fit.ridge <- glmnet(x,y,alpha = 0,family = 'gaussian',lambda = lambdas)
## plot the CV error versus regularization parameters lambdas
cv.ridge <- cv.glmnet(x, y, alpha = 0, nfold = 20, lambda = lambdas)
plot(cv.ridge)
```




```
cv.ridge
```

```
##
## Call:  cv.glmnet(x = x, y = y, lambda = lambdas, nfolds = 20, alpha = 0)
##
```

```
## Measure: Mean-Squared Error
##
##     Lambda Measure       SE Nonzero
## min  0.100 0.02297 0.004416      80
## 1se  0.631 0.02650 0.003110      80
```

```
# Print the coefficients best model
coef(fit.ridge,s=cv.ridge$lambda.min)
```

```
## 81 x 1 sparse Matrix of class "dgCMatrix"
##                           1
## (Intercept)    1.889057e+01
## Id            -8.312168e-06
## MSSubClass    -1.803978e-04
## MSZoning      -1.086268e-02
## LotFrontage   -1.339313e-04
## LotArea        1.427546e-06
## Street         1.749524e-01
## Alley          3.508729e-02
## LotShape      -6.095832e-03
## LandContour    5.118581e-03
## Utilities     -1.381975e-01
## LotConfig     -1.761800e-03
## LandSlope      2.306539e-02
## Neighborhood   1.030218e-03
## Condition1     2.369163e-03
## Condition2    -3.221325e-02
## BldgType      -8.288130e-03
## HouseStyle    -2.248492e-03
## OverallQual    4.769723e-02
## OverallCond    2.946694e-02
## YearBuilt      6.579386e-04
## YearRemodAdd   9.484482e-04
## RoofStyle      8.591871e-03
## RoofMatl       1.364299e-02
## Exterior1st   -1.344502e-03
## Exterior2nd    1.818642e-03
## MasVnrType     9.292828e-03
## MasVnrArea     3.578290e-05
## ExterQual     -1.675454e-02
## ExterCond      1.009753e-02
## Foundation     1.268211e-02
## BsmtQual      -2.169110e-02
## BsmtCond       6.129723e-03
## BsmtExposure  -1.001784e-02
## BsmtFinType1  -6.413727e-03
## BsmtFinSF1     2.531737e-05
## BsmtFinType2   3.636600e-03
## BsmtFinSF2     4.265805e-05
## BsmtUnfSF      1.340722e-05
## TotalBsmtSF    4.713178e-05
## Heating       -3.456183e-03
## HeatingQC     -8.415564e-03
## CentralAir     8.069780e-02
## Electrical     2.098013e-03
```

```
## X1stFlrSF      7.421698e-05
## X2ndFlrSF      4.845286e-05
## LowQualFinSF  -1.140249e-05
## GrLivArea      7.325138e-05
## BsmtFullBath   3.639285e-02
## BsmtHalfBath   1.000644e-02
## FullBath       3.990982e-02
## HalfBath       2.430737e-02
## BedroomAbvGr   1.084468e-02
## KitchenAbvGr  -3.933072e-02
## KitchenQual   -2.291328e-02
## TotRmsAbvGrd   1.554198e-02
## Functional     1.537032e-02
## Fireplaces     4.049177e-02
## FireplaceQu   -4.732122e-03
## GarageType    -5.251381e-03
## GarageYrBlt    2.965267e-05
## GarageFinish  -1.354080e-02
## GarageCars     4.272835e-02
## GarageArea     8.798922e-05
## GarageQual     5.932895e-04
## GarageCond     1.064561e-02
## PavedDrive     2.892451e-02
## WoodDeckSF     9.631123e-05
## OpenPorchSF    2.407044e-05
## EnclosedPorch  7.690061e-05
## X3SsnPorch     1.445236e-04
## ScreenPorch    2.721250e-04
## PoolArea      -2.224724e-04
## PoolQC        -2.614018e-02
## Fence         -9.516909e-03
## MiscFeature   -4.495988e-03
## MiscVal       -1.895242e-06
## MoSold         4.883528e-04
## YrSold        -5.750895e-03
## SaleType      -7.495861e-04
## SaleCondition  1.839330e-02
```

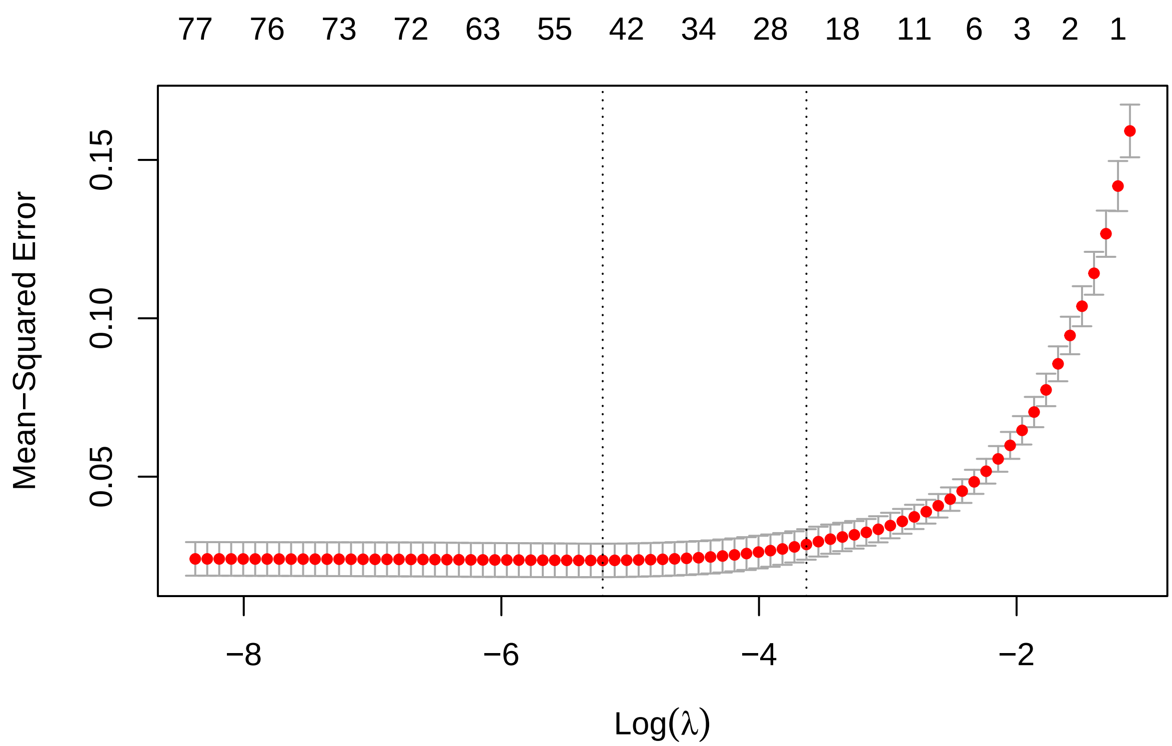
```r
## Make prediction
pred.ridge <- as.vector(predict(
fit.ridge,
s = cv.ridge$lambda.min,
newx = data.matrix(test)
))
```

Comment: Through the cross-validation, we select the optimal λ for the ridge regression. The minimal MSE of 20-fold cross-validation is 02297, and the corresponding λ is 0.1. Therefore, the best λ is 0.1.

2 LASSO Regression

```r
## Fit a Lasso model
fit.lasso <- cv.glmnet(x, y, alpha = 1)
cv.lasso <- cv.glmnet(x, y, alpha = 1, nfolds = 20)
```

```r
## Visualize model
plot(cv.lasso)
```




```r
cv.lasso
```

```
##
## Call:  cv.glmnet(x = x, y = y, nfolds = 20, alpha = 1)
##
## Measure: Mean-Squared Error
##
##       Lambda Measure      SE Nonzero
## min 0.005443 0.02355 0.00528      45
## 1se 0.026468 0.02860 0.00482      26
```

```r
# the coefficients of the bes model with the lowest CV error
coef(fit.lasso, s= cv.lasso$lambda.min)
```

```
## 81 x 1 sparse Matrix of class "dgCMatrix"
##                             1
## (Intercept)     1.097903e+01
## Id              .
## MSSubClass     -3.101908e-04
## MSZoning       -1.082849e-02
## LotFrontage     .
## LotArea         1.453514e-06
## Street          8.149080e-02
## Alley           2.801580e-02
```

```
## LotShape      -4.791910e-03
## LandContour    .
## Utilities      .
## LotConfig      .
## LandSlope      8.869038e-03
## Neighborhood   .
## Condition1     .
## Condition2    -1.742432e-02
## BldgType      -7.946254e-03
## HouseStyle     .
## OverallQual    7.880504e-02
## OverallCond    3.208484e-02
## YearBuilt      1.243674e-03
## YearRemodAdd   7.114636e-04
## RoofStyle      .
## RoofMatl       1.807869e-03
## Exterior1st    .
## Exterior2nd    .
## MasVnrType     .
## MasVnrArea     .
## ExterQual     -1.014574e-03
## ExterCond      3.948548e-03
## Foundation     5.051611e-03
## BsmtQual      -1.715084e-02
## BsmtCond       .
## BsmtExposure  -4.929917e-03
## BsmtFinType1  -8.375068e-03
## BsmtFinSF1     7.490262e-07
## BsmtFinType2   .
## BsmtFinSF2     .
## BsmtUnfSF      .
## TotalBsmtSF    4.211355e-05
## Heating        .
## HeatingQC     -6.772524e-03
## CentralAir     7.985145e-02
## Electrical     .
## X1stFlrSF      3.045978e-05
## X2ndFlrSF      .
## LowQualFinSF   .
## GrLivArea      1.788611e-04
## BsmtFullBath   4.255452e-02
## BsmtHalfBath   .
## FullBath       1.836959e-02
## HalfBath       2.659648e-03
## BedroomAbvGr   .
## KitchenAbvGr  -2.544272e-03
## KitchenQual   -2.046860e-02
## TotRmsAbvGrd   8.873080e-03
## Functional     1.151218e-02
## Fireplaces     3.408877e-02
## FireplaceQu    .
## GarageType    -3.621548e-03
## GarageYrBlt    .
## GarageFinish  -7.681020e-03
```

```
## GarageCars     6.598091e-02
## GarageArea     1.483880e-06
## GarageQual     .
## GarageCond     .
## PavedDrive     2.166898e-02
## WoodDeckSF     8.361824e-05
## OpenPorchSF    .
## EnclosedPorch  .
## X3SsnPorch     .
## ScreenPorch    2.384525e-04
## PoolArea      -1.589361e-04
## PoolQC        -1.393367e-02
## Fence          .
## MiscFeature    .
## MiscVal        .
## MoSold         .
## YrSold        -2.157765e-03
## SaleType       .
## SaleCondition  1.897004e-02
```

```
## Make prediction
pred.lasso <- as.vector(predict(
fit.lasso,
s = cv.lasso$lambda.min,
newx = data.matrix(test)
))
```

Comment: Through the cross-validation, we select the optimal λ for the Lasso regression. The minimal MSE of 20-fold cross-validation is 0.02365, and the corresponding λ is 0.005443. Therefore, the best λ is 0.02355. We find the coefficients of many predictor variables are shinkaged to zero, which achieves a subset selection effect.

3 References

Gareth James, Daniela Witten, Trevor Hastie Robert Tibshirani (2013), An Introduction to Statistical Learning with Applications in R.

https://www.pluralsight.com/guides/linear-lasso-and-ridge-regression-with-r