Quiz 7

Chaeeun Shin

4/9/2021

```
library(MASS)
library(tidyverse)
```

```
## Warning: package 'tidyr' was built under R version 4.2.3

## Warning: package 'readr' was built under R version 4.2.3

## Warning: package 'dplyr' was built under R version 4.2.3

## Warning: package 'stringr' was built under R version 4.2.3

## -- Attaching core tidyverse packages ----------------------- tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.4.4     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
## -- Conflicts ------------------------------------------- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x dplyr::select() masks MASS::select()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(caret)
```

```
## Loading required package: lattice
##
## Attaching package: 'caret'
##
## The following object is masked from 'package:purrr':
##
##     lift
```

```
library(glmnet)
```

```
## Loading required package: Matrix
##
## Attaching package: 'Matrix'
##
## The following objects are masked from 'package:tidyr':
##
##     expand, pack, unpack
##
## Loaded glmnet 4.1-8
```

```
library(caTools)
```

#1

```
AmesHousing <- read.csv("Ames_Housing_Data.csv")
AmesHousing <- na.omit(AmesHousing)
cat("There are", nrow(AmesHousing),"observations left.")
```

```
## There are 1460 observations left.
```

#2

```
set.seed(123)
training.samples <- AmesHousing$SalePrice %>%
  createDataPartition(p=0.75,list=FALSE)
train.data <- AmesHousing[training.samples,]
test.data <- AmesHousing[-training.samples,]
nrow(train.data)
```

```
## [1] 1097
```

```
nrow(test.data)
```

```
## [1] 363
```

#3

```
x <- model.matrix(SalePrice~., data=train.data)[,-1]
y <- train.data$SalePrice
cv <- cv.glmnet(x,y,alpha=0)
cat("The best lambda value: ", cv$lambda.min)
```

```
## The best lambda value:  6452.855
```

```
ridgemodel <- glmnet(x,y,alpha=0,lambda=cv$lambda.min)
coef(ridgemodel) #coefficient of the fitted model
```
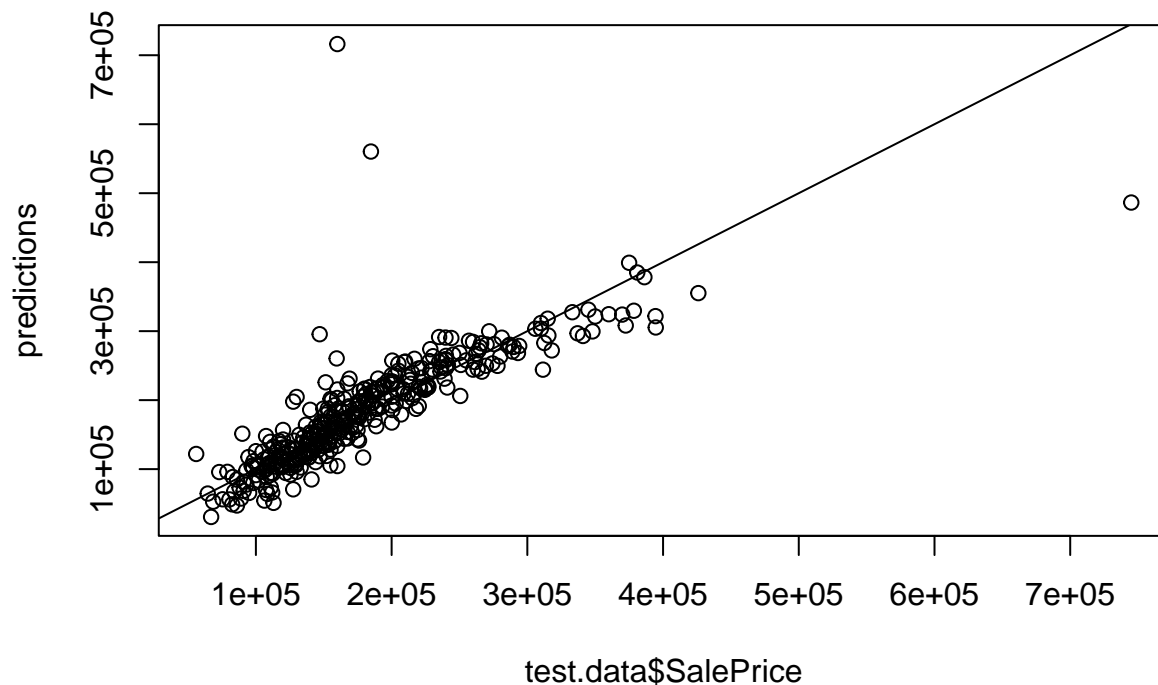
```
## 20 x 1 sparse Matrix of class "dgCMatrix"
##                        s0
## (Intercept)  -5.868377e+04
## Id           -2.400086e+00
## LotArea       6.829978e-01
## OverallQual   1.552083e+04
## OverallCond   4.734443e+03
## YearBuilt     4.367701e+02
## YearRemodAdd  2.036837e+02
## CentralAirY  -9.163648e+03
## X1stFlrSF     5.141967e+01
## X2ndFlrSF     1.721007e+01
## GrLivArea     3.865641e+01
## FullBath     -2.576679e+03
## HalfBath      3.410324e+02
## BedroomAbvGr -1.072097e+04
## KitchenAbvGr -3.025974e+04
## TotRmsAbvGrd  4.279323e+03
## Fireplaces    5.279621e+03
## GarageCars    2.424886e+03
## GarageArea    3.650612e+01
## YrSold       -6.243367e+02
```

```
x.test <- model.matrix(SalePrice~., data=test.data)[,-1]
predictions <- ridgemodel %>% predict(x.test) %>% as.vector() #making predictions
data.frame(
  RMSE = RMSE(predictions, test.data$SalePrice),
  Rsquare = R2(predictions, test.data$SalePrice)
)
```

```
##       RMSE    Rsquare
## 1 46713.63 0.6691365
```

```
plot(x=test.data$SalePrice, y=predictions)
abline(0,1)  #plot
```


#4

```
cv <- cv.glmnet(x,y,alpha=1)
cat("The best lambda value: ",cv$lambda.min)
```

```
## The best lambda value:  352.4729
```

```
lassomodel <- glmnet(x,y,alpha=1,lambda=cv$lambda.min)
coef(lassomodel) #coefficient of the fitted model
```

```
## 20 x 1 sparse Matrix of class "dgCMatrix"
##                       s0
## (Intercept) -6.160530e+05
## Id          -1.359380e+00
## LotArea      6.885348e-01
## OverallQual  1.673469e+04
## OverallCond  5.628004e+03
## YearBuilt    5.183887e+02
## YearRemodAdd 1.105492e+02
## CentralAirY -1.011034e+04
## X1stFlrSF    3.613856e+01
```

```
## X2ndFlrSF        .
## GrLivArea      6.455739e+01
## FullBath     -4.871532e+03
## HalfBath     -2.083079e+02
## BedroomAbvGr -1.188420e+04
## KitchenAbvGr -2.869665e+04
## TotRmsAbvGrd  3.101192e+03
## Fireplaces    3.087910e+03
## GarageCars    .
## GarageArea    3.684554e+01
## YrSold       -3.393338e+02
```

```r
predictions <- lassomodel %>% predict(x.test) %>% as.vector()
data.frame(
  RMSE = RMSE(predictions,test.data$SalePrice),
  Rsquare = R2(predictions,test.data$SalePrice)
)
```

```
##        RMSE    Rsquare
## 1 48460.31 0.6606179
```

```r
plot(x=test.data$SalePrice,y=predictions)
abline(0,1)
```


#5
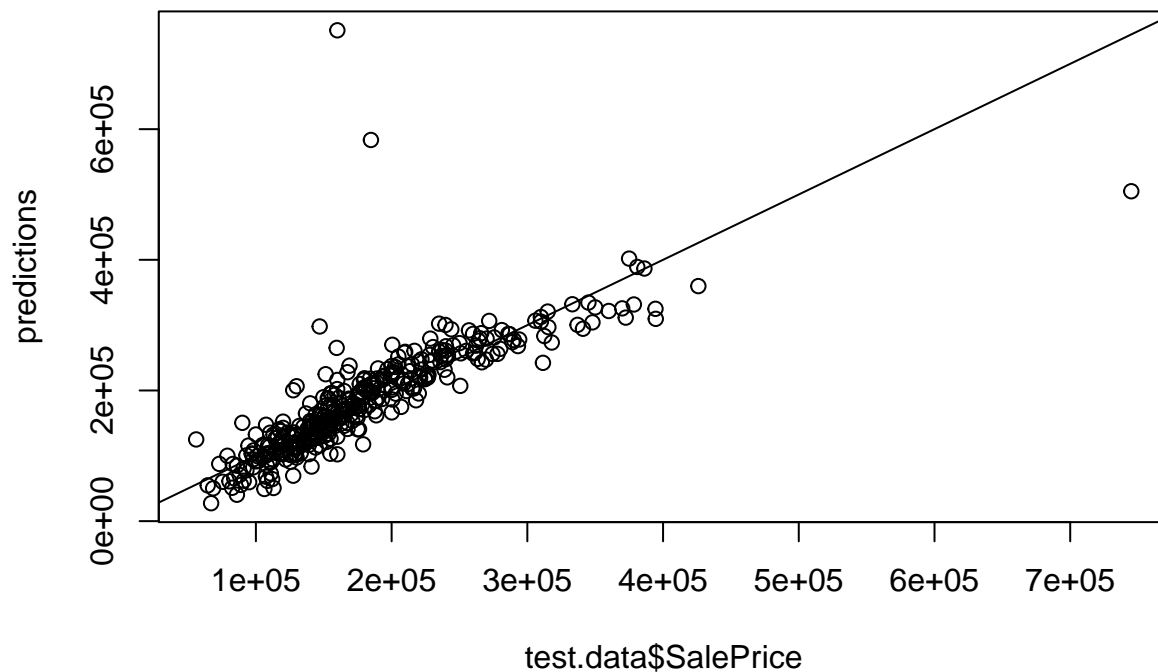
```r
elasticnetmodel <- train(SalePrice~., data=train.data, method="glmnet",trControl=trainControl("cv",numb
elasticnetmodel$bestTune
```

```
##   alpha    lambda
## 5   0.1 849.1096
```

```r
coef(elasticnetmodel$finalModel, elasticnetmodel$bestTune$lambda)
```

```
## 20 x 1 sparse Matrix of class "dgCMatrix"
##                      s1
```

```
## (Intercept)   -3.184591e+05
## Id             -2.065248e+00
## LotArea         6.999641e-01
## OverallQual     1.641517e+04
## OverallCond     5.931914e+03
## YearBuilt       5.464418e+02
## YearRemodAdd    1.242764e+02
## CentralAirY    -1.246334e+04
## X1stFlrSF       5.519033e+01
## X2ndFlrSF       2.097999e+01
## GrLivArea       4.537295e+01
## FullBath       -6.947849e+03
## HalfBath       -2.248932e+03
## BedroomAbvGr   -1.256102e+04
## KitchenAbvGr   -3.025522e+04
## TotRmsAbvGrd    3.844605e+03
## Fireplaces      3.509950e+03
## GarageCars      .
## GarageArea      3.757935e+01
## YrSold         -5.263536e+02
```