BA810 individual assignement

Chiebuka Onwuzurike (U15815811)

2/3/2021

Set Up

Adding Necessary libraries and theme setting

```
library(data.table)
library(ggplot2)
library(ggthemes)
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1
```

```
theme_set(theme_bw())
```

Importing DataTable

```
dd <-fread("C:/Users/chieb/Documents/Coding/School_Work/BA 810/housing.csv")

total_bedrooms_median <- median(dd$total_bedrooms, na.rm = TRUE)
dd[is.na(total_bedrooms), total_bedrooms := total_bedrooms_median]
```

Splitting the dataset 1-5000 for training and 15001-18000

```
train_offsets <- seq(5000)
test_offsets <- 15000 + seq(3000)
```

Modeling the dataset

```
x_data <- model.matrix( ~ -1 + total_rooms + total_bedrooms +
households + housing_median_age +
population + median_income + ocean_proximity, dd)
```

Setting up variables

```
y_data <- dd$median_house_value / 1e6

x_train <- x_data[train_offsets, ]
y_train <- y_data[train_offsets]

x_test <- x_data[test_offsets, ]
y_test <- y_data[test_offsets]

est <- glmnet(x_train, y_train, alpha = 1 , nlambda = 100)
```

Predict

```
y_train_hat <- predict(est, x_train)
y_test_hat <- predict(est, x_test)
```

Compute MSE train by for looping different lasso predict

```
mse_train <- c()

for (val in seq(ncol(y_train_hat))) {
  mse_train[val] <- mean((y_train - y_train_hat[,val])^2)
}
```

Compute MSE test by for looping different lasso predict

```
mse_test <- c()

for (val in seq(ncol(y_test_hat))) {
  mse_test[val] <- mean((y_test - y_test_hat[,val])^2)
}
```

Selecting lowest MSE for train test

```
lambda_min_mse_train <- c(est$lambda[which.min(mse_train)],mse_train[which.min(mse_train)])
lambda_min_mse_test <- c(est$lambda[which.min(mse_test)],mse_test[which.min(mse_test)])
```

Create a data.table of train MSEs and lambdas

```
dd_mse <- data.table(
lambda = est$lambda,
mse = mse_train,
dataset = "Train"
)
```

Use the rbind command to combine dd_mse_train and dd_mse_test into a single data table

```
dd_mse <- rbind(dd_mse, data.table(
lambda = est$lambda,
mse = mse_test,
dataset = "Test"
))
```

Lambda with the lowest MSE and the associated coefficients

An extremely high lambda lasso’s coefficient variables to 0, while when lambda approaches zero all variables have a coefficients not equal to 0. The lambda associated with the lowest MSE_test balances the trade of bias and under-fitting on the left (too few variables)and variance and over-filling on the right (too many variables).

```
print(lambda_min_mse_test[2])
```

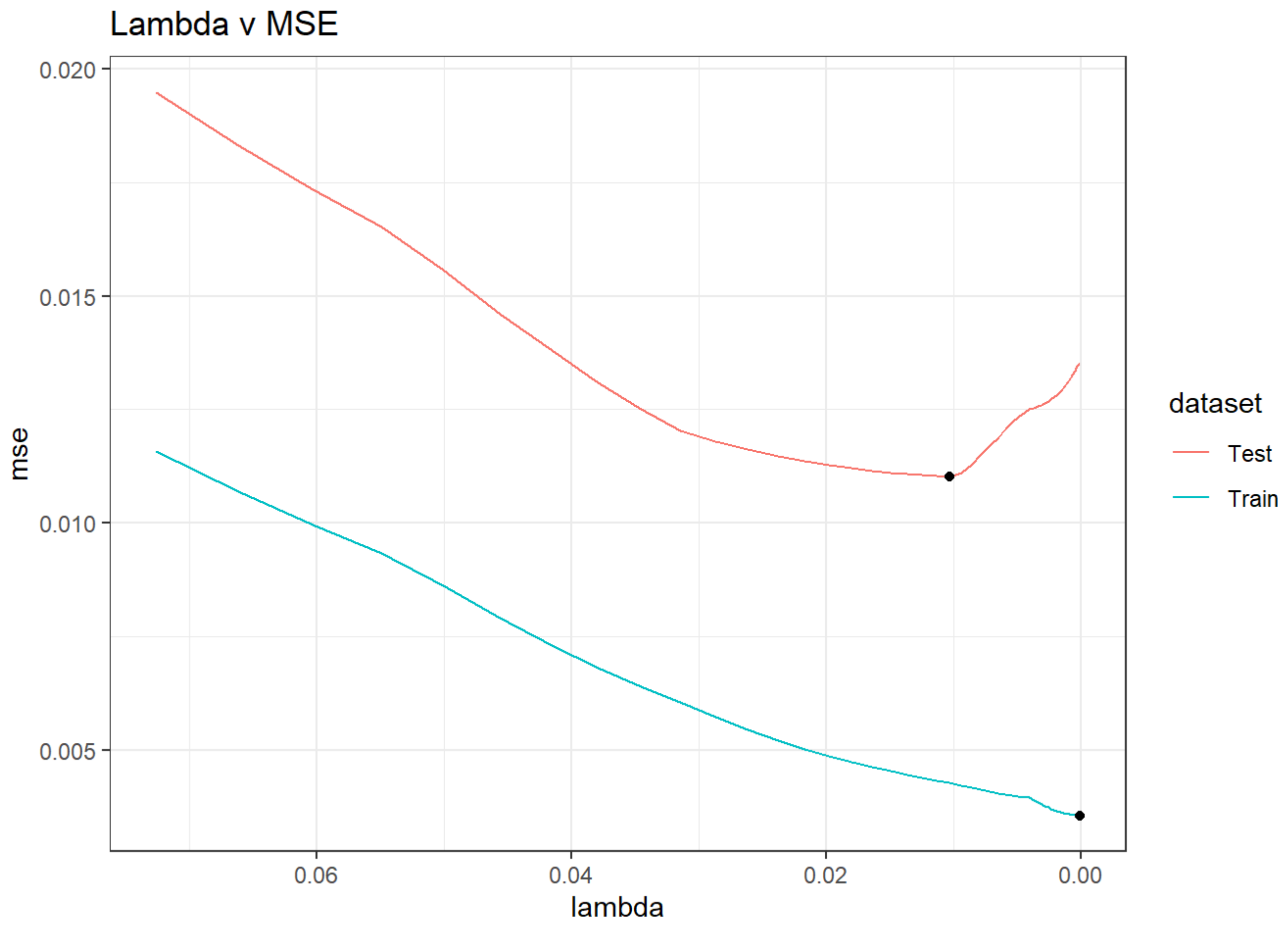
```
## [1] 0.01101823
```

```
coef(est,s=lambda_min_mse_test[2])
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                       1
## (Intercept)                8.243411e-02
## total_rooms                .           
## total_bedrooms             1.614134e-06
## households                 .           
## housing_median_age         .           
## population                 .           
## median_income              3.007328e-02
## ocean_proximity<1H OCEAN   2.843570e-02
## ocean_proximityINLAND     -5.720641e-02
## ocean_proximityISLAND      .           
## ocean_proximityNEAR BAY    .           
## ocean_proximityNEAR OCEAN  .           
```

Plot data frame containing λ’s and MSEs

```
ggplot() +
  geom_line(data = dd_mse, aes(x = lambda, y = mse, color= dataset, group = dataset))+
  geom_point(data = dd_mse, aes(x = lambda_min_mse_train[1] , y = lambda_min_mse_train[2]))+
  geom_point(data = dd_mse, aes(x = lambda_min_mse_test[1] , y = lambda_min_mse_test[2]) ) +
  scale_x_reverse()+
  ggtitle("Lambda v MSE")
```



Collabration

Leah Fowlkes helped me with using ggplot. Specifically with the “group= dataset” and general geom_point. Also helped me put my thoughts together for what the significant of the lambda associated with lowest MSE_test.