Assignment 1

2023-03-02

A1. What is the main purpose of regularization when training predictive models?

Ans. The goal of building a model is to make predictions on the unseen by learning from the existing data set. This is called generalization. But there are various limitations where a model can only partially generalize. They are 1. Overfitting 2. Non-representative training Data 3. Data profile variations 4—concept drift. Regularization aims to improve the model performance by simplifying the model. In most cases, every model tends to overfit the training set, reducing the model's generalization. The concept of regularization helps minimize the model's overfitting of the training data by adding penalizing terms, controlling the model, and keeping it as simple as possible.

A2. What is the role of a loss function in a predictive model? And name two common loss functions for regression models and two common loss functions for classification models.

Ans. In machine learning, the model is trained using input data and the corresponding target variable. When the trained machine learning model is used to predict the output, there is a difference between the actual and predicted values, i.e., y(actual) and y^(predicted); such a function is called a loss function. The goal of any machine learning model is to reduce this difference in the loss function.

Different types of loss functions for regression models are:

1. Root Mean Square Error (RMSE)

2. Mean Absolute Error (MAE)

Different types of loss functions for classification models are:

1. Binary Cross entropy Loss

2. Hinge Loss

A3.Consider the following scenario. You are building a classification model with many hyper-parameters on a relatively small dataset. You will see that the training error is extremely small. Can you fully trust this model? Discuss the reason.

Ans. No, we cannot fully trust such a model. The reason is that a classification model is built with many hyper-parameters on a small dataset; it is very easy for the model to follow the data points and understand the data, leading to the overfitting of the model. Thus, creating an impression that the model's performance is good. However, the model's training error is low, but when the same model is used on the unseen dataset, the model will not perform help as the model has overfitted on the training data.

A4. What is the role of the lambda parameter in regularized linear models such as Lasso or Ridge regression models?

Ans. Lambda is a hyper-parameter in regularized linear models that plays an important role in maintaining a balance between reducing the loss on the training data and minimizing the amplitude of the model's coefficients.
In the Lasso model, the penalty term, i.e., Lambda, is an L1 penalty that tries to minimize the sum of absolute values of coefficients. In the Ridge regression model, the L2 penalty minimizes the sum square of the coefficients.

#Loading required libraries for the current dataset.

```
library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ISLR)
library(glmnet)

## Loading required package: Matrix

## Loaded glmnet 4.1-6
```

#Loading and selecting required columns from ISLR carseats dataset.

```
carseats <- Carseats %>% select("Sales", "Price",
"Advertising","Population","Age","Income","Education")
```

#Scaling of the the carseats dataset using preProcess function from Caret package

```
carseats_scaled <- preProcess(carseats, method = c("scale", "center"))
carseats_predict<- predict(carseats_scaled, carseats)
summary(carseats_predict)
```

```
##      Sales              Price            Advertising        Population
##  Min.   :-2.65440   Min.   :-3.87702   Min.   :-0.9977   Min.   :-1.72918
##  1st Qu.:-0.74584   1st Qu.:-0.66711   1st Qu.:-0.9977   1st Qu.:-0.85387
##  Median :-0.00224   Median : 0.05089   Median :-0.2459   Median : 0.04858
##  Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.00000
##  3rd Qu.: 0.64575   3rd Qu.: 0.64219   3rd Qu.: 0.8067   3rd Qu.: 0.90693
##  Max.   : 3.10670   Max.   : 3.17633   Max.   : 3.3630   Max.   : 1.65671
##       Age               Income           Education
##  Min.   :-1.74827   Min.   :-1.70290   Min.   :-1.48825
##  1st Qu.:-0.83779   1st Qu.:-0.92573   1st Qu.:-0.72504
##  Median : 0.07268   Median : 0.01224   Median : 0.03816
##  Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.00000
##  3rd Qu.: 0.78255   3rd Qu.: 0.79834   3rd Qu.: 0.80137
##  Max.   : 1.64673   Max.   : 1.83458   Max.   : 1.56457
```

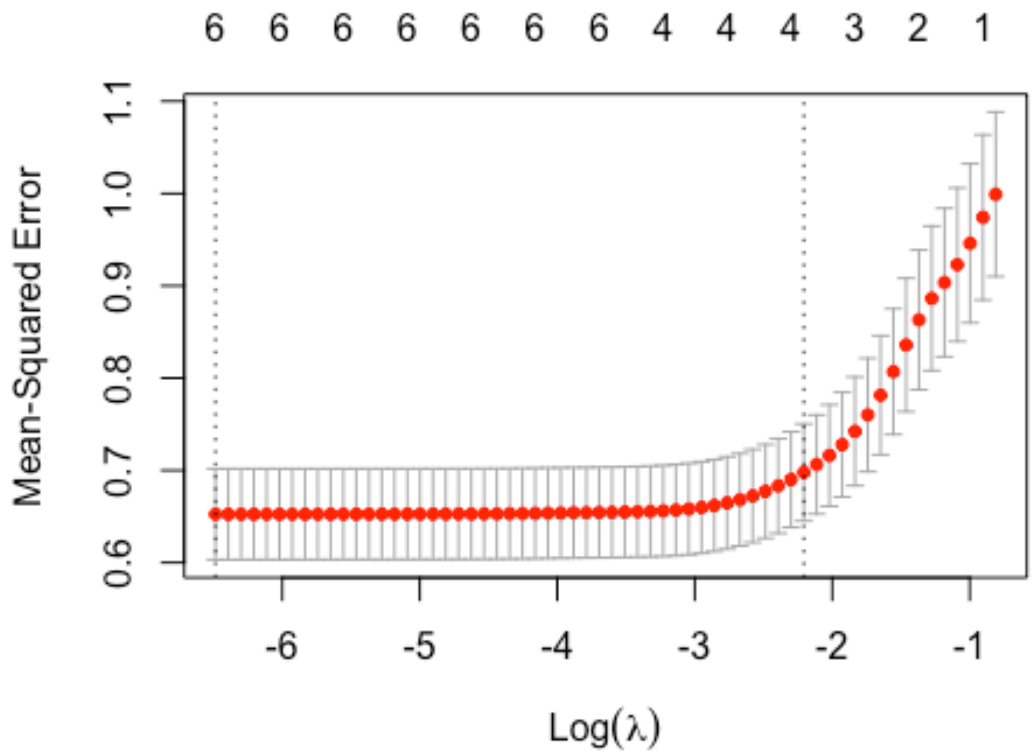
#Creating a matrix for glmnet library for current dataset.

```
y <- carseats_predict$Sales

x<- data.matrix(carseats_predict[,c("Price",
"Advertising","Population","Age","Income","Education")])
```

#QB1. Build a Lasso regression model to predict Sales based on all other attributes.

```
lasso_model<- cv.glmnet(x, y, alpha = 1)
plot(lasso_model)
```




```
best_lambda <- lasso_model$lambda.min
best_lambda

## [1] 0.001524481

#The best value of lambda is 0.001524481
```

#QB2.The coefficient for the price (normalized) attribute in the best model

```
price_coef<- coef(lasso_model, s= "lambda.min")
price_coef

## 7 x 1 sparse Matrix of class "dgCMatrix"
##                          s1
## (Intercept)  3.723500e-16
## Price       -4.793834e-01
## Advertising  2.932098e-01
## Population  -4.624934e-02
## Age         -2.792202e-01
## Income       1.024459e-01
## Education   -3.223128e-02

#The coefficient for the price is -4.793834e-01.
```

#QB3.Changing Lambda value to 0.01 and 0.1

```r
#Changing Lambda value to 0.01
lasso_model1<- cv.glmnet(x, y, alpha = 0.01)


best_lambda1 <- lasso_model1$lambda.min
best_lambda1

## [1] 0.00587463

coef<- coef(lasso_model1, s = "lambda.min")
coef

## 7 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)  3.712074e-16
## Price       -4.779918e-01
## Advertising  2.931612e-01
## Population  -4.753870e-02
## Age         -2.789635e-01
## Income       1.033067e-01
## Education   -3.361099e-02

#Changing Lambda value to 0.1
lasso_model2<- cv.glmnet(x, y, alpha = 0.1)


best_lambda2 <- lasso_model2$lambda.min
best_lambda2

## [1] 0.005478699

coef1<- coef(lasso_model2, s = "lambda.min")
coef1

## 7 x 1 sparse Matrix of class "dgCMatrix"
##                         s1
## (Intercept)  3.712642e-16
## Price       -4.778993e-01
## Advertising  2.928057e-01
## Population  -4.693700e-02
## Age         -2.786935e-01
## Income       1.029705e-01
## Education   -3.312333e-02

#The comparison between changing the lambda value shows that all the
attributes remain in the model, even after changing the lambda.
```

#QB4. Build an elastic-net model with alpha set to 0.6.

```r
elastic_model<- cv.glmnet(x, y, alpha = 0.6)

best_lambda3<- elastic_model$lambda.min
best_lambda3
```

```
## [1] 0.002315083
```

```
coef2<- coef(lasso_model2, s = "lambda.min")
coef2
```

```
## 7 x 1 sparse Matrix of class "dgCMatrix"
##                        s1
## (Intercept)  3.712642e-16
## Price       -4.778993e-01
## Advertising  2.928057e-01
## Population  -4.693700e-02
## Age         -2.786935e-01
## Income       1.029705e-01
## Education   -3.312333e-02
```

```
#The best value of lambda for current model is 0.002315083
```