ADM Assignment-1

Akshaya Mamidipalli

2024-03-11

**PART-A**

**QA1)**

The main purpose of regularization when training predictive models is to prevent overfitting. In machine learning, regularization involves moving the coefficients closer to zero. With the generalization we can prevent overfitting, improve generalization.

**QA2)**

A Loss function is used to analyze the model's performance over the training dataset. The discrepancy between model predictions and real-world problem scenarios is expressed by loss functions.   
For regression models, the two most used loss functions are Mean Absolute Error and Mean Squared Errors.   
Log loss and Hinge loss are the two popular loss functions used in classification models.

**QA3)**

Even though we have extremely small error in the training dataset we cannot fully trust the model because there are many hyperparameters and it doesn’t perform well on the test dataset.

**QA4)**

The regularized linear model's penalty level is set by the hyperparameter lambda , tuning the lambda parameter is crucial to obtaining optimal performance and generalization. By serving as a regularization hyperparameter, the lambda parameter enables the model to strike a compromise between creating a simple, less susceptible to overfitting model and one that fits the training data well.

**PART-B**

library(ISLR)  
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(glmnet)

## Warning: package 'glmnet' was built under R version 4.3.3

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.3.3

## Loaded glmnet 4.1-8

library(caret)

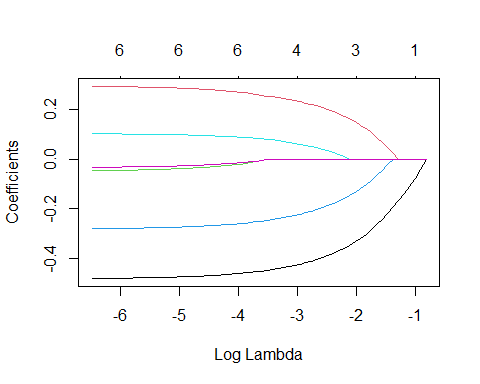
## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.3.2

## Loading required package: lattice

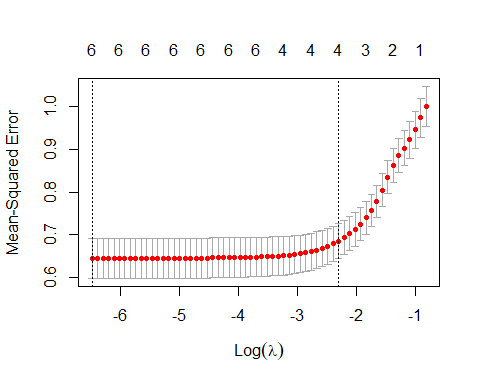
Carseats\_Filtered <- Carseats %>% select("Sales", "Price","Advertising","Population","Age","Income","Education")

Carseats\_Filtered\_Normalized <- scale(Carseats\_Filtered)  
X <- as.matrix(Carseats\_Filtered\_Normalized[, c('Price','Advertising','Population','Age','Income','Education')])  
Y <- Carseats\_Filtered\_Normalized[,'Sales']  
fit.lasso <- glmnet(X,Y,alpha = 1)  
plot(fit.lasso,xvar = "lambda")



#QB1)

lasso\_model <- cv.glmnet(X,Y, alpha = 1)  
plot(lasso\_model)



best\_value <- lasso\_model$lambda.min  
best\_value

## [1] 0.001524481

The best value of the lambda for the lasso- regression model is 0.0015

#QB2)

coef(fit.lasso, s=best\_value)

## 7 x 1 sparse Matrix of class "dgCMatrix"  
## s1  
## (Intercept) 9.866665e-17  
## 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 (normalized) attribute in the best model is -4.79 \* 10^-1

#QB3)

coef(fit.lasso, s=0.01)

## 7 x 1 sparse Matrix of class "dgCMatrix"  
## s1  
## (Intercept) 9.798009e-17  
## Price -4.696889e-01  
## Advertising 2.815718e-01  
## Population -3.323443e-02  
## Age -2.693300e-01  
## Income 9.585212e-02  
## Education -2.330455e-02

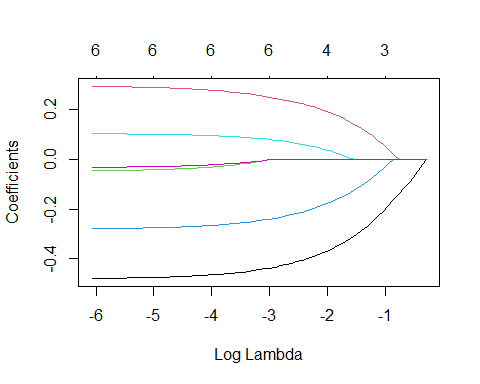
coef(fit.lasso, s=0.1)

## 7 x 1 sparse Matrix of class "dgCMatrix"  
## s1  
## (Intercept) 9.803050e-17  
## Price -3.691394e-01  
## Advertising 1.839178e-01  
## Population .   
## Age -1.684796e-01  
## Income 1.925921e-02  
## Education .

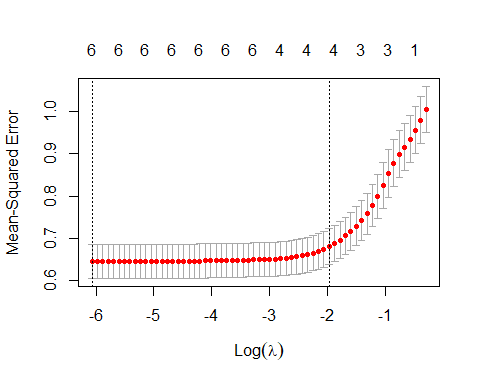
When the lambda value is set to 0.01, all the attributes remaining in the model with non-zero coefficients. Nevertheless, the number of attributes with non-zero coefficients drops to four if the lambda value is raised to 0.1. ‘Population’ and ‘Education’ attributes are eliminated from the model because their coefficients are practically zero.

#QB4)

fit.elastic <- glmnet(X,Y,alpha = 0.6)  
plot(fit.elastic,xvar = "lambda")



elastic\_model <- cv.glmnet(X,Y, alpha = 0.6)  
plot(elastic\_model)



best\_value\_elastic <- elastic\_model$lambda.min  
best\_value\_elastic

## [1] 0.002315083

The best value of the lambda for an elastic-net model is 0.0023