Assignment - 3

NANDINI NAIR

2023-12-06

PART- 2

```
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(kernlab)
library(caret)

## Loading required package: ggplot2

##
## Attaching package: 'ggplot2'

## The following object is masked from 'package:kernlab':
##
##     alpha

## Loading required package: lattice

library(glmnet)

## Loading required package: Matrix

## Loaded glmnet 4.1-8

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

head(data)

##   Sales Price Advertising Population Age Income Education
## 1  9.50   120          11        276  42     73        17
```

```
## 2 11.22    83          16        260  65     48        10
## 3 10.06    80          10        269  59     35        12
## 4  7.40    97           4        466  55    100        14
## 5  4.15   128           3        340  38     64        13
## 6 10.81    72          13        501  78    113        16
```

QB1

```
#Building SVM model

set.seed(1800)

Model1<- train(Sales ~ .,
               data = data,
               method = "svmLinear",
               preProcess = c("center", "scale"))
Model1

## Support Vector Machines with Linear Kernel
##
## 400 samples
##   6 predictor
##
## Pre-processing: centered (6), scaled (6)
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 400, 400, 400, 400, 400, 400, ...
## Resampling results:
##
##   RMSE     Rsquared   MAE
##   2.29201  0.3516323  1.834006
##
## Tuning parameter 'C' was held constant at a value of 1
```

R squared is 35.16

The value for coefficient to determinization for SVMmodel= 35.16

QB2

```
#Training SVM model using repeat cross- validation &cost parameter of
0.1,0.5,1, 10 using search g

set.seed(1800)
grid = expand.grid(C= c(0.1,0.5,1,10))
trctrl <- trainControl(method = "repeatedcv", number = 5, repeats = 2)
Model2 <- train(Sales ~., data = data, method =
"svmLinear",trControl=trctrl,preProcess = c("center", "scale"),tuneGrid =
grid,tuneLength = 10)
Model2
```

```
## Support Vector Machines with Linear Kernel
##
## 400 samples
##   6 predictor
##
## Pre-processing: centered (6), scaled (6)
## Resampling: Cross-Validated (5 fold, repeated 2 times)
## Summary of sample sizes: 321, 319, 320, 320, 320, 320, ...
## Resampling results across tuning parameters:
##
##   C     RMSE      Rsquared   MAE
##    0.1  2.269187  0.3589452  1.819715
##    0.5  2.268261  0.3598035  1.818847
##    1.0  2.268602  0.3595973  1.819189
##   10.0  2.268419  0.3597283  1.819037
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was C = 0.5.
```

#R-squared = 35.97

Best R-squared value of 35.60 is obtained and RMSE value is least when we use C= 0.5

QB3

```
set.seed(1800)

#Scaling data using pre-process and predict function

scale <- preProcess(data[,2:7],method = c("center","scale"))
scale_data<-predict(scale,data)

#Building a model using neural network

Model3<- train(Sales~.,data=scale_data,method="nnet",linout=TRUE, trace =
FALSE)

## Warning in nominalTrainWorkflow(x = x, y = y, wts = weights, info =
trainInfo,
## : There were missing values in resampled performance measures.

Model3

## Neural Network
##
## 400 samples
##   6 predictor
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 400, 400, 400, 400, 400, 400, ...
```

```
## Resampling results across tuning parameters:
##
##   size  decay  RMSE      Rsquared   MAE
##   1     0e+00  2.324489  0.3428203  1.846006
##   1     1e-04  2.365103  0.3027520  1.889336
##   1     1e-01  2.296196  0.3471990  1.822618
##   3     0e+00  2.612228  0.2331132  2.034811
##   3     1e-04  2.935296  0.2276099  2.113722
##   3     1e-01  2.485492  0.2648533  1.984189
##   5     0e+00  2.716672  0.2010844  2.168360
##   5     1e-04  2.760406  0.1988235  2.180913
##   5     1e-01  2.614667  0.2256446  2.096217
##
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were size = 1 and decay = 0.1.
```

R- Squared = 34.71 and size = 1

Q4

```
#Predicting Sales for following data when Price = 6.54, Population = 124,
Advertising = 0, Age = 76, Income = 11 ,0, Education = 10
df <- data.frame(Price = 6.54, Population = 124, Advertising = 0, Age = 76,
Income = 110, Education = 10)
Pred_sale<- predict(Model3,df )
Pred_sale

##       1
## 4.97624
```

Predicted value for sales = 4.97