Advanced DM Assignment Three

Khutso Ledwaba

2023-04-14

```
library('ISLR')
```

```
## Warning: package 'ISLR' was built under R version 4.2.2
```

```
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 4.2.2

## Loading required package: Matrix

## Warning: package 'Matrix' was built under R version 4.2.2

## Loaded glmnet 4.1-6
```

```
library('caret')
```

```
## Loading required package: ggplot2

## Warning: package 'ggplot2' was built under R version 4.2.2

## Loading required package: lattice
```

```
library('tree')
```

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

```
library('class')
library('kernlab')
```

```
## Warning: package 'kernlab' was built under R version 4.2.2
```

```
##
## Attaching package: 'kernlab'
```

```
## The following object is masked from 'package:ggplot2':
##
##     alpha
```

```
Carseats <- Carseats
Carseats_Filtered <- Carseats %>% select("Sales", "Price", "Advertising","Population","Age","Income","Ed

##Question 1

#Normalizing the data
Normalizaton_data <- preProcess(Carseats_Filtered[,-1], method=c("center","scale"))
DATA <- predict(Normalizaton_data, Carseats_Filtered)
head(DATA)
```

```
##   Sales      Price Advertising  Population        Age     Income   Education
## 1  9.50  0.1776010   0.6563550  0.07572445 -0.6989069  0.1551667  1.18296763
## 2 11.22 -1.3851191   1.4081936 -0.03284107  0.7208201 -0.7381360 -1.48824960
## 3 10.06 -1.5118261   0.5059873  0.02822704  0.3504565 -1.2026533 -0.72504468
## 4  7.40 -0.7938196  -0.3962189  1.36494005  0.1035475  1.1199335  0.03816025
## 5  4.15  0.5154865  -0.5465866  0.50998655 -0.9458160 -0.1664223 -0.34344221
## 6 10.81 -1.8497116   0.9570904  1.60242713  1.5232746  1.5844509  0.80136517
```

```
set.seed(123)
#Building the linear regression model
model <- train(Sales~., data=DATA, method="svmLinear")
model
```

```
## Support Vector Machines with Linear Kernel
##
## 400 samples
##   6 predictor
##
## No pre-processing
## Resampling: Bootstrapped (25 reps)
## Summary of sample sizes: 400, 400, 400, 400, 400, 400, ...
## Resampling results:
##
##   RMSE      Rsquared   MAE
##   2.312705  0.3462025  1.858189
##
## Tuning parameter 'C' was held constant at a value of 1
```

```r
#Question 2

# set up the training control for cross-validation
set.seed(123)
ctrl<- trainControl(method="repeatedcv", number=5, repeats=2)
# define the search grid
grid <- expand.grid(C= c(0.1, 0.5, 1, 10))
# train the model using the customized search grid
searchmodel <- train(Sales~., data=DATA,method="svmLinear", trControl=ctrl,tuneGrid=grid)
# print the results
searchmodel
```

```
## Support Vector Machines with Linear Kernel
##
## 400 samples
##   6 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold, repeated 2 times)
## Summary of sample sizes: 320, 321, 319, 320, 320, 319, ...
## Resampling results across tuning parameters:
##
##   C     RMSE      Rsquared   MAE
##    0.1  2.299265  0.3462103  1.840099
##    0.5  2.298622  0.3466721  1.840422
##    1.0  2.300082  0.3457111  1.841175
##   10.0  2.298811  0.3464293  1.840017
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was C = 0.5.
```

```r
#Question 3

#Neural network model
set.seed(123)
M2 <- train(Sales~., data=DATA, method="nnet", linout=TRUE, verbose=FALSE, trace=FALSE)
M2
```

```
## 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  2.405588  0.2988709  1.911134
##   1     1e-04  2.342795  0.3270677  1.866081
##   1     1e-01  2.305827  0.3469510  1.840931
##   3     0e+00  2.624813  0.2338646  2.056537
```

```
##   3     1e-04  2.578680  0.2415518  2.041563
##   3     1e-01  2.461853  0.2773936  1.969713
##   5     0e+00  2.702451  0.2137557  2.165259
##   5     1e-04  2.682123  0.2216052  2.141906
##   5     1e-01  2.635539  0.2296870  2.120646
##
## 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.
```

```
#Question 4
#predictedSales
newDATA <- data.frame(Price=6.54, Population=124, Advertising=0, Age=76, Income= 110, Education=10)
newDATA
```

```
##   Price Population Advertising Age Income Education
## 1  6.54        124           0  76    110        10
```

```
predictedSales <- predict(M2, newDATA)
predictedSales
```

```
##        1
## 4.976256
```