

## Assignment\_2.R

Universal Bank

Tejasvini Mavuleti

2022-02-20

```
UniversalBank <- read.csv("C:/Users/mavul/Downloads/UniversalBank.csv")
```

```
summary(UniversalBank)
```

```
##          ID          Age      Experience      Income      ZIP.Cod
## Min.      :  1  Min.      :23.00  Min.      : -3.0  Min.      :  8.00  Min.      :  9
307
## 1st Qu.:1251  1st Qu.:35.00  1st Qu.:10.0  1st Qu.: 39.00  1st Qu.:91
911
## Median :2500  Median :45.00  Median :20.0  Median : 64.00  Median :93
437
## Mean    :2500  Mean    :45.34  Mean    :20.1  Mean    : 73.77  Mean    :93
153
## 3rd Qu.:3750  3rd Qu.:55.00  3rd Qu.:30.0  3rd Qu.: 98.00  3rd Qu.:94
608
## Max.    :5000  Max.    :67.00  Max.    :43.0  Max.    :224.00  Max.    :96
651
##      Family      CCAvg      Education      Mortgage
## Min.      :1.000  Min.      : 0.000  Min.      :1.000  Min.      :  0.0
## 1st Qu.:1.000  1st Qu.: 0.700  1st Qu.:1.000  1st Qu.:  0.0
## Median :2.000  Median : 1.500  Median :2.000  Median :  0.0
## Mean    :2.396  Mean    : 1.938  Mean    :1.881  Mean    : 56.5
## 3rd Qu.:3.000  3rd Qu.: 2.500  3rd Qu.:3.000  3rd Qu.:101.0
## Max.    :4.000  Max.    :10.000  Max.    :3.000  Max.    :635.0
## Personal.Loan  Securities.Account  CD.Account      Online
## Min.      :0.000  Min.      :0.0000  Min.      :0.0000  Min.      :0.0000
## 1st Qu.:0.000  1st Qu.:0.0000  1st Qu.:0.0000  1st Qu.:0.0000
## Median :0.000  Median :0.0000  Median :0.0000  Median :1.0000
## Mean    :0.096  Mean    :0.1044  Mean    :0.0604  Mean    :0.5968
## 3rd Qu.:0.000  3rd Qu.:0.0000  3rd Qu.:0.0000  3rd Qu.:1.0000
## Max.    :1.000  Max.    :1.0000  Max.    :1.0000  Max.    :1.0000
##      CreditCard
## Min.      :0.000
## 1st Qu.:0.000
## Median :0.000
## Mean    :0.294
## 3rd Qu.:1.000
## Max.    :1.000
```

```
#Remove the Null variables
```

```
UniversalBank$ID <- NULL
```

```
UniversalBank$ZIP.Code <- NULL
```

```
summary(UniversalBank)
```

```
##      Age      Experience      Income      Family
## Min.   :23.00  Min.   :-3.0  Min.    :  8.00  Min.    :1.000
## 1st Qu.:35.00  1st Qu.:10.0  1st Qu.: 39.00  1st Qu.:1.000
## Median :45.00  Median :20.0  Median : 64.00  Median :2.000
## Mean   :45.34  Mean   :20.1  Mean   : 73.77  Mean   :2.396
## 3rd Qu.:55.00  3rd Qu.:30.0  3rd Qu.: 98.00  3rd Qu.:3.000
## Max.   :67.00  Max.   :43.0  Max.   :224.00  Max.   :4.000
##      CCAvg      Education      Mortgage      Personal.Loan
## Min.    : 0.000  Min.    :1.000  Min.    :  0.0  Min.    :0.000
## 1st Qu.: 0.700  1st Qu.:1.000  1st Qu.:  0.0  1st Qu.:0.000
## Median : 1.500  Median :2.000  Median :  0.0  Median :0.000
## Mean    : 1.938  Mean    :1.881  Mean    : 56.5  Mean    :0.096
## 3rd Qu.: 2.500  3rd Qu.:3.000  3rd Qu.:101.0  3rd Qu.:0.000
## Max.    :10.000  Max.    :3.000  Max.    :635.0  Max.    :1.000
## Securities.Account  CD.Account      Online      CreditCard
## Min.    :0.0000  Min.    :0.0000  Min.    :0.0000  Min.    :0.000
## 1st Qu.:0.0000  1st Qu.:0.0000  1st Qu.:0.0000  1st Qu.:0.000
## Median :0.0000  Median :0.0000  Median :1.0000  Median :0.000
## Mean    :0.1044  Mean    :0.0604  Mean    :0.5968  Mean    :0.294
## 3rd Qu.:0.0000  3rd Qu.:0.0000  3rd Qu.:1.0000  3rd Qu.:1.000
## Max.    :1.0000  Max.    :1.0000  Max.    :1.0000  Max.    :1.000
```

```
#Call the Libraries
```

```
library(caret)
```

```
## Loading required package: ggplot2
```

```
## Warning in register(): Can't find generic `scale_type` in package ggplot2  
to
```

```
## register S3 method.
```

```
## Loading required package: lattice
```

```
library(class)
```

```
library(ggplot2)
```

```
library(lattice)
```

```
library(FNN)
```

```
##
```

```
## Attaching package: 'FNN'
```

```
## The following objects are masked from 'package:class':
```

```
##
```

```
##      knn, knn.cv
```

```
summary(UniversalBank)
```

```
##      Age      Experience      Income      Family
## Min.   :23.00   Min.    :-3.0   Min.    : 8.00   Min.    :1.000
## 1st Qu.:35.00   1st Qu.:10.0   1st Qu.: 39.00   1st Qu.:1.000
## Median :45.00   Median :20.0   Median : 64.00   Median :2.000
## Mean   :45.34   Mean    :20.1   Mean    : 73.77   Mean    :2.396
## 3rd Qu.:55.00   3rd Qu.:30.0   3rd Qu.: 98.00   3rd Qu.:3.000
## Max.   :67.00   Max.    :43.0   Max.    :224.00   Max.    :4.000
##      CCAvg      Education      Mortgage      Personal.Loan
## Min.    : 0.000   Min.    :1.000   Min.    : 0.0   Min.    :0.000
## 1st Qu.: 0.700   1st Qu.:1.000   1st Qu.: 0.0   1st Qu.:0.000
## Median : 1.500   Median :2.000   Median : 0.0   Median :0.000
## Mean    : 1.938   Mean    :1.881   Mean    : 56.5   Mean    :0.096
## 3rd Qu.: 2.500   3rd Qu.:3.000   3rd Qu.:101.0   3rd Qu.:0.000
## Max.    :10.000   Max.    :3.000   Max.    :635.0   Max.    :1.000
## Securities.Account  CD.Account      Online      CreditCard
## Min.    :0.0000   Min.    :0.0000   Min.    :0.0000   Min.    :0.000
## 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
## Median :0.0000   Median :0.0000   Median :1.0000   Median :0.000
## Mean    :0.1044   Mean    :0.0604   Mean    :0.5968   Mean    :0.294
## 3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.000
## Max.    :1.0000   Max.    :1.0000   Max.    :1.0000   Max.    :1.000
```

```
UniversalBank$Personal.Loan=as.factor(UniversalBank$Personal.Loan)
summary(UniversalBank)
```

```
##      Age      Experience      Income      Family
## Min.   :23.00   Min.    :-3.0   Min.    : 8.00   Min.    :1.000
## 1st Qu.:35.00   1st Qu.:10.0   1st Qu.: 39.00   1st Qu.:1.000
## Median :45.00   Median :20.0   Median : 64.00   Median :2.000
## Mean   :45.34   Mean    :20.1   Mean    : 73.77   Mean    :2.396
## 3rd Qu.:55.00   3rd Qu.:30.0   3rd Qu.: 98.00   3rd Qu.:3.000
## Max.   :67.00   Max.    :43.0   Max.    :224.00   Max.    :4.000
##      CCAvg      Education      Mortgage      Personal.Loan
## Min.    : 0.000   Min.    :1.000   Min.    : 0.0   0:4520
## 1st Qu.: 0.700   1st Qu.:1.000   1st Qu.: 0.0   1: 480
## Median : 1.500   Median :2.000   Median : 0.0
## Mean    : 1.938   Mean    :1.881   Mean    : 56.5
## 3rd Qu.: 2.500   3rd Qu.:3.000   3rd Qu.:101.0
## Max.    :10.000   Max.    :3.000   Max.    :635.0
## Securities.Account  CD.Account      Online      CreditCard
## Min.    :0.0000   Min.    :0.0000   Min.    :0.0000   Min.    :0.000
## 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
## Median :0.0000   Median :0.0000   Median :1.0000   Median :0.000
## Mean    :0.1044   Mean    :0.0604   Mean    :0.5968   Mean    :0.294
## 3rd Qu.:0.0000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.000
## Max.    :1.0000   Max.    :1.0000   Max.    :1.0000   Max.    :1.000
```

```
#Question 1 -----
#Normalize the data
```

```

Bank_Norm <- UniversalBank
Norm_model<-preProcess(UniversalBank[, -7], method = c("center", "scale"))
Bank_Norm[, -7] = predict(Norm_model, UniversalBank[, -7])
summary(Bank_Norm)

##      Age      Experience      Income      Family
## Min.   :-1.94871  Min.    :-2.014710  Min.    :-1.4288  Min.    :-1.2167
## 1st Qu.: -0.90188  1st Qu.: -0.881116  1st Qu.: -0.7554  1st Qu.: -1.2167
## Median :-0.02952  Median :-0.009121  Median :-0.2123  Median :-0.3454
## Mean   : 0.00000  Mean    : 0.000000  Mean    : 0.0000  Mean    : 0.0000
## 3rd Qu.: 0.84284  3rd Qu.: 0.862874  3rd Qu.: 0.5263  3rd Qu.: 0.5259
## Max.    : 1.88967  Max.    : 1.996468  Max.    : 3.2634  Max.    : 1.3973
##      CCAvg      Education      Mortgage      Personal.Loan
## Min.   :-1.1089  Min.    :-1.0490  Min.    : 0.0  0:4520
## 1st Qu.: -0.7083  1st Qu.: -1.0490  1st Qu.: 0.0  1: 480
## Median :-0.2506  Median : 0.1417  Median : 0.0
## Mean   : 0.0000  Mean    : 0.0000  Mean    : 56.5
## 3rd Qu.: 0.3216  3rd Qu.: 1.3324  3rd Qu.:101.0
## Max.    : 4.6131  Max.    : 1.3324  Max.    :635.0
## Securities.Account  CD.Account      Online      CreditCard
## Min.   :-0.3414  Min.    :-0.2535  Min.    :-1.2165  Min.    :-0.6452
## 1st Qu.: -0.3414  1st Qu.: -0.2535  1st Qu.: -1.2165  1st Qu.: -0.6452
## Median :-0.3414  Median :-0.2535  Median : 0.8219  Median :-0.6452
## Mean   : 0.0000  Mean    : 0.0000  Mean    : 0.0000  Mean    : 0.0000
## 3rd Qu.: -0.3414  3rd Qu.: -0.2535  3rd Qu.: 0.8219  3rd Qu.: 1.5495
## Max.    : 2.9286  Max.    : 3.9438  Max.    : 0.8219  Max.    : 1.5495

#Train the data

Train_Index=createDataPartition(UniversalBank$Personal.Loan,p=0.6, list = FALSE)
Train.df=Bank_Norm[Train_Index,]
Validation.df=Bank_Norm[-Train_Index,]

#Predicting the data frame

To_Predict=data.frame(Age=40, Experience=10, Income=84, Family=2, CCAvg=2,
                      Education=1, Mortgage=0, Securities.Account=0, CD.Account=0,
                      Online=1, CreditCard=1)

print(To_Predict)

##   Age Experience Income Family CCAvg Education Mortgage Securities.Account
## 1  40         10     84      2      2          1          0              0
##   CD.Account Online CreditCard
## 1          0       1          1

```

```

To_Predict_norm<-predict(Norm_model,To_Predict)
print(To_Predict_norm)

##           Age Experience      Income      Family      CCAvg Education Mortgage
## 1 -0.4657003 -0.8811162 0.2221371 -0.3453975 0.0355115 -1.048973          0
##   Securities.Account CD.Account      Online CreditCard
## 1           -0.3413892 -0.2535149 0.8218687    1.549477

Prediction <- knn(train = Train.df[,1:7], test = To_Predict_norm[,1:7],
                  cl=Train.df$Personal.Loan, k=1)
print(Prediction)

## [1] 0
## attr(,"nn.index")
##      [,1]
## [1,] 809
## attr(,"nn.dist")
##      [,1]
## [1,] 0.4207974
## Levels: 0

# --- This shows that the 5 nearest neighbors will be classified as 0 and the
# customer will also be 0.

#Question 2 -----
set.seed(123)
fitControl <- trainControl(method = "repeatedcv", number = 3, repeats = 2)
searchGrid=expand.grid(k=1:10)

knn.model=train(Personal.Loan~., data = Train.df, method = 'knn', tuneGrid =
searchGrid, trControl = fitControl)

knn.model

## k-Nearest Neighbors
##
## 3000 samples
## 11 predictor
## 2 classes: '0', '1'
##
## No pre-processing
## Resampling: Cross-Validated (3 fold, repeated 2 times)
## Summary of sample sizes: 2000, 2000, 2000, 2000, 2000, ...
## Resampling results across tuning parameters:
##
##  k  Accuracy  Kappa
##  1  0.9361667  0.5878252
##  2  0.9248333  0.5165691
##  3  0.9348333  0.5263843
##  4  0.9328333  0.5071197

```

```

##      5  0.9351667  0.5137446
##      6  0.9320000  0.4932105
##      7  0.9315000  0.4745260
##      8  0.9310000  0.4686273
##      9  0.9296667  0.4545287
##     10  0.9301667  0.4536660
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was k = 1.

# --- We can see that the best choice which balances the model from over fitting is k=3

#Question 3 -----
predictions <- predict(knn.model,Validation.df)
confusionMatrix(predictions,Validation.df$Personal.Loan)

## Confusion Matrix and Statistics
##
##              Reference
## Prediction      0      1
##              0 1762    82
##              1   46   110
##
##              Accuracy : 0.936
##              95% CI : (0.9244, 0.9463)
##              No Information Rate : 0.904
##              P-Value [Acc > NIR] : 1.808e-07
##
##              Kappa : 0.5975
##
##  Mcnemar's Test P-Value : 0.001977
##
##              Sensitivity : 0.9746
##              Specificity : 0.5729
##              Pos Pred Value : 0.9555
##              Neg Pred Value : 0.7051
##              Prevalence : 0.9040
##              Detection Rate : 0.8810
##              Detection Prevalence : 0.9220
##              Balanced Accuracy : 0.7737
##
##              'Positive' Class : 0
##

# --- This is the confusion matrix.

#Question 4 -----
To_Predict=data.frame(Age=40, Experience=10, Income=84, Family=2, CCAvg=2,
                      Education=1, Mortgage=0, Securities.Account=0, CD.Accou

```

```

nt=0,
                                Online=1,CreditCard=1)

To_Predict_norm=predict(Norm_model,To_Predict)

predict(knn.model,To_Predict_norm)

## [1] 0
## Levels: 0 1

# --- The customer is classified as 1.
#Question 5 -----
set.seed(123)
Test_Index_1 = createDataPartition(UniversalBank$Age, p= 0.2 , list=FALSE)
Test_Data_1 = UniversalBank[Test_Index_1,]
Rem_DATA = UniversalBank[-Test_Index_1,]
head(Rem_DATA)

##   Age Experience Income Family CCAvg Education Mortgage Personal.Loan
## 1  25           1     49      4  1.6           1           0           0
## 2  45          19     34      3  1.5           1           0           0
## 4  35           9    100      1  2.7           2           0           0
## 5  35           8     45      4  1.0           2           0           0
## 6  37          13     29      4  0.4           2        155           0
## 7  53          27     72      2  1.5           2           0           0
##   Securities.Account CD.Account Online CreditCard
## 1                   1           0           0           0
## 2                   1           0           0           0
## 4                   0           0           0           0
## 5                   0           0           0           1
## 6                   0           0           1           0
## 7                   0           0           1           0

Train_Index_1 = createDataPartition(Rem_DATA$Age, p= 0.5 , list=FALSE)
Train_Data_1 = Rem_DATA[Train_Index_1,]
Validation_Data_1 = Rem_DATA[-Train_Index_1,]
head(Validation_Data_1)

##   Age Experience Income Family CCAvg Education Mortgage Personal.Loan
## 1  25           1     49      4  1.6           1           0           0
## 4  35           9    100      1  2.7           2           0           0
## 5  35           8     45      4  1.0           2           0           0
## 12 29           5     45      3  0.1           2           0           0
## 13 48          23    114      2  3.8           3           0           0
## 17 38          14    130      4  4.7           3        134           1
##   Securities.Account CD.Account Online CreditCard
## 1                   1           0           0           0
## 4                   0           0           0           0
## 5                   0           0           0           1
## 12                  0           0           1           0

```

```

## 13          1          0          0          0
## 17          0          0          0          0

train.norm.df_1 <- Train_Data_1
valid.norm.df_1 <- Validation_Data_1
test.norm.df_1 <- Test_Data_1
rem_data.norm.df_1 <- Rem_DATA

norm.values_1 <- preProcess(Train_Data_1[-7], method=c("center", "scale"))
train.norm.df_1[-7] <- predict(norm.values_1, Train_Data_1[-7])
valid.norm.df_1[-7] <- predict(norm.values_1, Validation_Data_1[-7])
test.norm.df_1[-7] <- predict(norm.values_1, test.norm.df_1[-7])

rem_data.norm.df_1[-7] <- predict(norm.values_1, Rem_DATA[-7])
head(test.norm.df_1)

##          Age  Experience      Income      Family      CCAvg Education Mortg
age
## 3  -0.5520988 -0.44264562 -1.3508132 -1.2488911 -0.5364945 -1.054404
0
## 8   0.4074662  0.34195519 -1.1108626 -1.2488911 -0.9329309  1.331125
0
## 11  1.7159639  1.64962320  0.6996744  1.3693292  0.2563782  1.331125
0
## 16  1.2797980  0.86502239 -1.1108626 -1.2488911 -0.2533257  1.331125
0
## 24 -0.1159329 -0.18111202 -0.6527749 -0.3761510 -0.7063958 -1.054404
163
## 26 -0.2031661 -0.09393415 -0.9581667  0.4965891 -0.8196633 -1.054404
97
##  Personal.Loan Securities.Account CD.Account      Online CreditCard
## 3              0          -0.3442513 -0.2492064 -1.2411790 -0.6498183
## 8              0          -0.3442513 -0.2492064 -1.2411790  1.5381222
## 11             0          -0.3442513 -0.2492064 -1.2411790 -0.6498183
## 16             0          -0.3442513 -0.2492064  0.8052827  1.5381222
## 24             0          2.9034024 -0.2492064 -1.2411790 -0.6498183
## 26             0          -0.3442513 -0.2492064  0.8052827 -0.6498183

```

*#--- Therefore, the model is best fit on the training data and is most accurate on the training data and the least on the testing data.*