R Notebook

Code

Hide

```
library(dplyr)
library(caret)
library(ggplot2)


Bank <- read.csv("UniversalBank.csv")
##Bank <-UniversalBank
head(Bank)
```

	ID <int>	Age <int>	Experience <int>	Income <int>	ZIPCode <int>	Family <int>	CCAvg <dbl>	Education <int>	Mortgage <int>
1	1	25	1	49	91107	4	1.6	1	0
2	2	45	19	34	90089	3	1.5	1	0
3	3	39	15	11	94720	1	1.0	1	0
4	4	35	9	100	94112	1	2.7	2	0
5	5	35	8	45	91330	4	1.0	2	0
6	6	37	13	29	92121	4	0.4	2	155

6 rows | 1-10 of 14 columns

Hide

```
str(Bank)
```

/

```
'data.frame':   5000 obs. of  14 variables:
 $ ID                : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Age               : int  25 45 39 35 35 37 53 50 35 34 ...
 $ Experience        : int  1 19 15 9 8 13 27 24 10 9 ...
 $ Income            : int  49 34 11 100 45 29 72 22 81 180 ...
 $ ZIPCode           : int  91107 90089 94720 94112 91330 92121 91711 93943 90089 93023 ...
 $ Family            : int  4 3 1 1 4 4 2 1 3 1 ...
 $ CCAvg             : num  1.6 1.5 1 2.7 1 0.4 1.5 0.3 0.6 8.9 ...
 $ Education         : int  1 1 1 2 2 2 2 3 2 3 ...
 $ Mortgage          : int  0 0 0 0 0 155 0 0 104 0 ...
 $ PersonalLoan      : int  0 0 0 0 0 0 0 0 0 1 ...
 $ SecuritiesAccount: int  1 1 0 0 0 0 0 0 0 0 ...
 $ CDAccount         : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Online            : int  0 0 0 0 0 1 1 0 1 0 ...
 $ CreditCard        : int  0 0 0 0 1 0 0 1 0 0 ...
```

Hide

```
Bank1 <- Bank[,c(-1, -5)] ##excluding 2 coloumn form data "ID" and "Zip Code"

##converting Personal loan to factor
Bank1$PersonalLoan=as.factor(Bank1$PersonalLoan)
Bank1$Education = as.factor(Bank1$Education)
levels(Bank1$Education)
```

```
[1] "1" "2" "3"
```

Hide

```
dummy_model<-dummyVars(~Education,data=Bank1)
head(predict(dummy_model, Bank1))
```

```
  Education.1 Education.2 Education.3
1           1           0           0
2           1           0           0
3           1           0           0
4           0           1           0
5           0           1           0
6           0           1           0
```

/

Hide

```
Bank1<-cbind(Bank1, predict(dummy_model, Bank1))
Bank1$Education<- NULL
Bank1
```

	… <int>	Experience <int>	Income <int>	Family <int>	CCA… <dbl>	Mortgage <int>	PersonalLoan <fctr>	SecuritiesAccount <int>	CDAccount <int>
1	25	1	49	4	1.60	0	0	1	0
2	45	19	34	3	1.50	0	0	1	0
3	39	15	11	1	1.00	0	0	0	0
4	35	9	100	1	2.70	0	0	0	0
5	35	8	45	4	1.00	0	0	0	0
6	37	13	29	4	0.40	155	0	0	0
7	53	27	72	2	1.50	0	0	0	0
8	50	24	22	1	0.30	0	0	0	0
9	35	10	81	3	0.60	104	0	0	0
10	34	9	180	1	8.90	0	1	0	0

1-10 of 5,000 rows | 1-10 of 14 columns

Previous **1** 2 3 4 5 6 … 100 Next

Hide

```
View(Bank1)
```

Qs 1) Consider the following customer: Age = 40, Experience = 10, Income = 84, Family = 2, CCAvg = 2, Education_1 = 0, Education_2 = 1, Education_3 = 0, Mortgage = 0, Securities Account = 0, CD Account = 0, Online = 1, and Credit Card = 1. Perform a k-NN classification with all predictors except ID and ZIP code using k = 1. Remember to transform categorical predictors with more than two categories into dummy variables first. Specify the success class as 1 (loan acceptance), and use the default cutoff value of 0.5. How would this customer be classified?

Hide

/

```
set.seed(25)

Test_Data<-read.csv("test.csv")


Test_Data <- Test_Data[,c(-1, -5)] ##excluding 2 coloumn form data "ID" and "Zip Code"

Test_Data$Education.1 = as.factor(Test_Data$Education.1)
Test_Data$Education.2 = as.factor(Test_Data$Education.2)
Test_Data$Education.3 = as.factor(Test_Data$Education.3)
levels(Test_Data$Education)
```

```
NULL
```

Hide

```
Test_Data
```

… <int>	Experience <int>	Income <int>	Family <int>	CCA… <int>	Mortgage <int>	SecuritiesAccount <int>	CDAccount <int>	Online <int>	CreditCard <int>
40	10	84	2	2	0	0	0	1	1

1 row | 1-10 of 13 columns

Code

Hide

```
## data Partioning into training and validation data
Train_Index = createDataPartition(Bank1$PersonalLoan,p=0.6,list=FALSE)
Train_Data = Bank1[Train_Index,]
Validation_Data = Bank1[-Train_Index,]

summary(Train_Data)
```

/

```
      Age            Experience        Income
 Min.   :23.00   Min.   :-3.00   Min.   :  8.00
 1st Qu.:36.00   1st Qu.:11.00   1st Qu.: 39.00
 Median :46.00   Median :20.00   Median : 64.00
 Mean   :45.55   Mean   :20.33   Mean   : 73.79
 3rd Qu.:55.00   3rd Qu.:30.00   3rd Qu.: 99.00
 Max.   :67.00   Max.   :43.00   Max.   :224.00
     Family           CCAvg          Mortgage
 Min.   :1.000   Min.   : 0.00   Min.   :  0.00
 1st Qu.:1.000   1st Qu.: 0.70   1st Qu.:  0.00
 Median :2.000   Median : 1.50   Median :  0.00
 Mean   :2.392   Mean   : 1.94   Mean   : 55.21
 3rd Qu.:3.000   3rd Qu.: 2.60   3rd Qu.: 98.00
 Max.   :4.000   Max.   :10.00   Max.   :635.00
 PersonalLoan SecuritiesAccount   CDAccount
 0:2712       Min.   :0.0000    Min.   :0.000
 1: 288       1st Qu.:0.0000    1st Qu.:0.000
              Median :0.0000    Median :0.000
              Mean   :0.1087    Mean   :0.062
              3rd Qu.:0.0000    3rd Qu.:0.000
              Max.   :1.0000    Max.   :1.000
     Online         CreditCard      Education.1
 Min.   :0.0000   Min.   :0.0000   Min.   :0.000
 1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000
 Median :1.0000   Median :0.0000   Median :0.000
 Mean   :0.5933   Mean   :0.2963   Mean   :0.423
 3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.000
 Max.   :1.0000   Max.   :1.0000   Max.   :1.000
  Education.2      Education.3
 Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.:0.0000
 Median :0.0000   Median :0.0000
 Mean   :0.2763   Mean   :0.3007
 3rd Qu.:1.0000   3rd Qu.:1.0000
 Max.   :1.0000   Max.   :1.0000
```

Hide

```
NROW(Train_Data)
```

```
[1] 3000
```

/

Hide

```
NROW(Test_Data)
```

```
[1] 1
```

Hide

```
NROW(Validation_Data)
```

```
[1] 2000
```

Hide

```
## Normalization
# Copy the original data
train.norm.df <- Train_Data
valid.norm.df <- Validation_Data
test.norm.df <- Test_Data
norm.values <- preProcess(Train_Data[, 1:14], method=c("center", "scale"))

train.norm.df[, 1:14] <- predict(norm.values, Train_Data[, 1:14])
valid.norm.df[, 1:14] <- predict(norm.values, Validation_Data[, 1:14])
test.norm.df[, 1:13] <- predict(norm.values, Test_Data[, 1:13])
```

```
恟牶-恟牪 not meaningful for factors恟牶-恟牪 not meaningful for factors恟牶-恟牪 not meaningful for factors
```

Hide

```
norm.values <- preProcess(Train_Data[, 1:14], method=c("center", "scale"))
```

Hide

```
summary(train.norm.df)
```

/

```
      Age              Experience           Income      
 Min.   :-1.96767   Min.   :-2.03572   Min.   :-1.4249  
 1st Qu.:-0.83333   1st Qu.:-0.81427   1st Qu.:-0.7535  
 Median : 0.03924   Median :-0.02905   Median :-0.2120  
 Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.0000  
 3rd Qu.: 0.82455   3rd Qu.: 0.84341   3rd Qu.: 0.5461  
 Max.   : 1.87163   Max.   : 1.97761   Max.   : 3.2536  
     Family            CCAvg            Mortgage      
 Min.   :-1.2104   Min.   :-1.1088   Min.   :-0.5457  
 1st Qu.:-1.2104   1st Qu.:-0.7087   1st Qu.:-0.5457  
 Median :-0.3406   Median :-0.2515   Median :-0.5457  
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.5291   3rd Qu.: 0.3771   3rd Qu.: 0.4230  
 Max.   : 1.3988   Max.   : 4.6062   Max.   : 5.7306  
 PersonalLoan SecuritiesAccount   CDAccount        
 0:2712       Min.   :-0.3491   Min.   :-0.2571  
 1: 288       1st Qu.:-0.3491   1st Qu.:-0.2571  
              Median :-0.3491   Median :-0.2571  
              Mean   : 0.0000   Mean   : 0.0000  
              3rd Qu.:-0.3491   3rd Qu.:-0.2571  
              Max.   : 2.8635   Max.   : 3.8890  
     Online          CreditCard       Education.1      
 Min.   :-1.2077   Min.   :-0.6488   Min.   :-0.8561  
 1st Qu.:-1.2077   1st Qu.:-0.6488   1st Qu.:-0.8561  
 Median : 0.8277   Median :-0.6488   Median :-0.8561  
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.8277   3rd Qu.: 1.5407   3rd Qu.: 1.1677  
 Max.   : 0.8277   Max.   : 1.5407   Max.   : 1.1677  
  Education.2       Education.3      
 Min.   :-0.6178   Min.   :-0.6556  
 1st Qu.:-0.6178   1st Qu.:-0.6556  
 Median :-0.6178   Median :-0.6556  
 Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 1.6180   3rd Qu.: 1.5248  
 Max.   : 1.6180   Max.   : 1.5248  
```

Hide

```
summary(valid.norm.df)
```

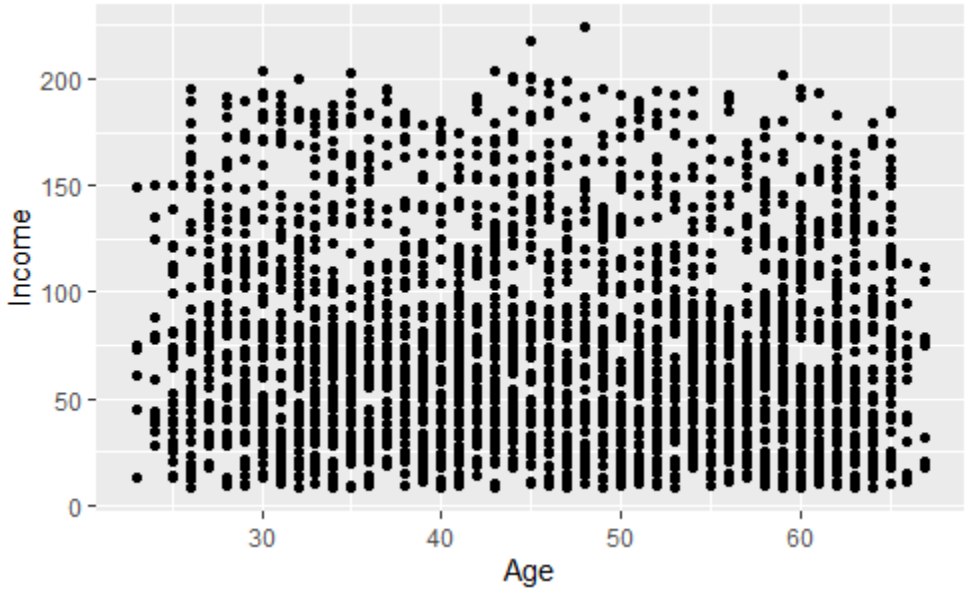
/

```
      Age              Experience
 Min.   :-1.96767   Min.   :-2.03572
 1st Qu.:-0.92059   1st Qu.:-0.90152
 Median :-0.04802   Median :-0.02905
 Mean   :-0.04623   Mean   :-0.04982
 3rd Qu.: 0.82455   3rd Qu.: 0.77798
 Max.   : 1.87163   Max.   : 1.97761
     Income                Family
 Min.   :-1.4249365   Min.   :-1.21036
 1st Qu.:-0.7534778   1st Qu.:-1.21036
 Median :-0.2336388   Median :-0.34064
 Mean   :-0.0006751   Mean   : 0.01029
 3rd Qu.: 0.5244598   3rd Qu.: 0.52908
 Max.   : 2.8420755   Max.   : 1.39880
     CCAvg               Mortgage       PersonalLoan
 Min.   :-1.108753   Min.   :-0.5457   0:1808
 1st Qu.:-0.708710   1st Qu.:-0.5457   1: 192
 Median :-0.251517   Median :-0.5457
 Mean   :-0.003098   Mean   : 0.0319
 3rd Qu.: 0.319974   3rd Qu.: 0.4946
 Max.   : 4.606156   Max.   : 5.5033
 SecuritiesAccount    CDAccount            Online
 Min.   :-0.34910   Min.   :-0.25705   Min.   :-1.20770
 1st Qu.:-0.34910   1st Qu.:-0.25705   1st Qu.:-1.20770
 Median :-0.34910   Median :-0.25705   Median : 0.82775
 Mean   :-0.03427   Mean   :-0.01658   Mean   : 0.01764
 3rd Qu.:-0.34910   3rd Qu.:-0.25705   3rd Qu.: 0.82775
 Max.   : 2.86352   Max.   : 3.88896   Max.   : 0.82775
   CreditCard         Education.1        Education.2
 Min.   :-0.64884   Min.   :-0.85607   Min.   :-0.61784
 1st Qu.:-0.64884   1st Qu.:-0.85607   1st Qu.:-0.61784
 Median :-0.64884   Median :-0.85607   Median :-0.61784
 Mean   :-0.01277   Mean   :-0.01923   Mean   : 0.02385
 3rd Qu.: 1.54071   3rd Qu.: 1.16774   3rd Qu.: 1.61801
 Max.   : 1.54071   Max.   : 1.16774   Max.   : 1.61801
  Education.3
 Min.   :-0.655584
 1st Qu.:-0.655584
 Median :-0.655584
 Mean   :-0.002544
 3rd Qu.: 1.524850
 Max.   : 1.524850
```

/

Hide

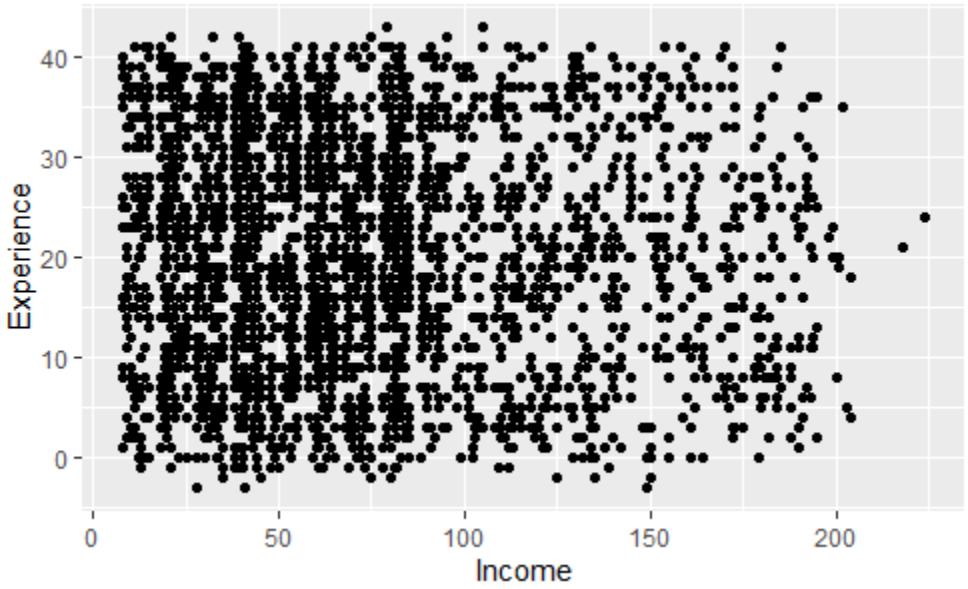
```
## Data Plotting

library(ggplot2)
ggplot(Train_Data, aes(x=Age,y=Income, Color="PersonalLoan")) +
  geom_point()
```



Hide

```
ggplot(Train_Data, aes(x=Income,y=Experience, Color="Mortagage")) +
  geom_point()
```

/



Qs 2) What is a choice of k that balances between overfitting and ignoring the predictor information?

We will now run our model and test on the validation set

Hide

```
##Modelling knn
library(FNN)
nn <- knn(train = train.norm.df[, 1:6], test = test.norm.df[, 1:6],
          cl = train.norm.df[, 7], k = 1, prob=TRUE)
row.names(Train_Data)[attr(nn, "nn.index")]
```

```
[1] "701"
```

knn of test set is 1 that is loan would be accept

Hide

```
library(caret)
accuracy.df <- data.frame(k = seq(1, 14, 1), accuracy = rep(0, 14))

# compute knn for different k on validation.
for(i in 2:14) {
  knn.pred <- knn(train.norm.df[,-7], valid.norm.df[,-7],
                  cl = train.norm.df[,7], k = i)
  accuracy.df[i, 2] <- confusionMatrix(knn.pred, valid.norm.df[,7])$overall[1]
}
accuracy.df
```

k <dbl>	accuracy <dbl>
1	0.0000
2	0.9530
3	0.9595
4	0.9525
5	0.9575
6	0.9535
7	0.9585
8	0.9530
9	0.9550
10	0.9495

1-10 of 14 rows Previous **1** 2 Next

Qs 3) Show the confusion matrix for the validation data that results from using the best k.

Hide

```
knn.pred <- knn(train.norm.df[,-7], valid.norm.df[,-7],
                cl = train.norm.df[,7], k = 3)
accuracy.df <- confusionMatrix(knn.pred, valid.norm.df[,7])$overall[1]
accuracy.df
```

/

```
Accuracy
  0.9595
```

Hide

```
 CrossTable(x=valid.norm.df[,7],y=knn.pred)
```

/

```
   Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  2000


                   | knn.pred
valid.norm.df[, 7] |         0 |         1 | Row Total |
-------------------|-----------|-----------|-----------|
                 0 |      1795 |        13 |      1808 |
                   |     7.296 |    99.213 |           |
                   |     0.993 |     0.007 |     0.904 |
                   |     0.963 |     0.095 |           |
                   |     0.897 |     0.006 |           |
-------------------|-----------|-----------|-----------|
                 1 |        68 |       124 |       192 |
                   |    68.702 |   934.252 |           |
                   |     0.354 |     0.646 |     0.096 |
                   |     0.037 |     0.905 |           |
                   |     0.034 |     0.062 |           |
-------------------|-----------|-----------|-----------|
      Column Total |      1863 |       137 |      2000 |
                   |     0.931 |     0.068 |           |
-------------------|-----------|-----------|-----------|
```

Qs 4) Consider the following customer: Age = 40, Experience = 10, Income = 84, Family = 2, CCAvg = 2, Education_1 = 0, Education_2 = 1, Education_3 = 0, Mortgage = 0, Securities Account = 0, CD Account = 0, Online = 1 and Credit Card = 1. Classify the customer using the best k.

Hide

/

```
library(FNN)
nn <- knn(train = train.norm.df[, 1:6], test = test.norm.df[, 1:6],
          cl = train.norm.df[, 7], k = 3, prob=TRUE)
row.names(train.norm.df)[attr(nn, "nn.index")]
```

```
[1] "701"  "4927" "3416"
```

Qs 5) Repartition the data, this time into training, validation, and test sets (50% : 30% : 20%). Apply the k-NN method with the k chosen above. Compare the confusion matrix of the test set with that of the training and validation sets. Comment on the differences and their reason.

Hide

```
## repartion the data  into training, validation, and test sets (50% : 30% : 20%)
Bank2 <- read.csv("UniversalBank.csv")
Bank2 <- Bank[,c(-1, -5)] ##excluding 2 coloumn form data "ID" and "Zip Code"
```

Hide

```
##converting Personal loan to factor
Bank2$PersonalLoan=as.factor(Bank2$PersonalLoan)
Bank2$Education = as.factor(Bank2$Education)
levels(Bank2$Education)
```

```
[1] "1" "2" "3"
```

Hide

```
dummy_model<-dummyVars(~Education,data=Bank2)
head(predict(dummy_model, Bank2))
```

```
  Education.1 Education.2 Education.3
1           1           0           0
2           1           0           0
3           1           0           0
4           0           1           0
5           0           1           0
6           0           1           0
```

/

Hide

```
Bank2<-cbind(Bank2, predict(dummy_model, Bank2))
Bank2$Education<- NULL
Bank2
```

	… <int>	Experience <int>	Income <int>	Family <int>	CCA… <dbl>	Mortgage <int>	PersonalLoan <fctr>	SecuritiesAccount <int>	CDAccount <int>
1	25	1	49	4	1.60	0	0	1	0
2	45	19	34	3	1.50	0	0	1	0
3	39	15	11	1	1.00	0	0	0	0
4	35	9	100	1	2.70	0	0	0	0
5	35	8	45	4	1.00	0	0	0	0
6	37	13	29	4	0.40	155	0	0	0
7	53	27	72	2	1.50	0	0	0	0
8	50	24	22	1	0.30	0	0	0	0
9	35	10	81	3	0.60	104	0	0	0
10	34	9	180	1	8.90	0	1	0	0

1-10 of 5,000 rows | 1-10 of 14 columns

Previous **1** 2 3 4 5 6 … 100 Next

Hide

```
##Repartition the data, tnto training, validation, and test sets (50% : 30% : 20%).
set.seed(20)
Train_Index = createDataPartition(Bank2$PersonalLoan,p=0.5, list=FALSE)
Train_Data = Bank2[Train_Index,]
ValTest_data = Bank2[-Train_Index,]

Validation_Index <- createDataPartition(ValTest_data$PersonalLoan,p=0.6, list=FALSE)
Validation_Data <- ValTest_data[Validation_Index,]
Test_Data <- ValTest_data[-Validation_Index,]
summary(Train_Data)
```

/

```
      Age            Experience       Income
 Min.   :23.00   Min.   :-3.0   Min.   :  8.00
 1st Qu.:36.00   1st Qu.:11.0   1st Qu.: 39.00
 Median :46.00   Median :21.0   Median : 64.00
 Mean   :45.62   Mean   :20.4   Mean   : 74.15
 3rd Qu.:56.00   3rd Qu.:30.0   3rd Qu.: 99.00
 Max.   :67.00   Max.   :43.0   Max.   :205.00
     Family          CCAvg           Mortgage
 Min.   :1.000   Min.   : 0.000   Min.   :  0.0
 1st Qu.:1.000   1st Qu.: 0.700   1st Qu.:  0.0
 Median :2.000   Median : 1.500   Median :  0.0
 Mean   :2.409   Mean   : 1.915   Mean   : 57.3
 3rd Qu.:4.000   3rd Qu.: 2.600   3rd Qu.:101.2
 Max.   :4.000   Max.   :10.000   Max.   :635.0
 PersonalLoan SecuritiesAccount   CDAccount
 0:2260       Min.   :0.0000    Min.   :0.0000
 1: 240       1st Qu.:0.0000    1st Qu.:0.0000
              Median :0.0000    Median :0.0000
              Mean   :0.1032    Mean   :0.0572
              3rd Qu.:0.0000    3rd Qu.:0.0000
              Max.   :1.0000    Max.   :1.0000
     Online        CreditCard      Education.1
 Min.   :0.000   Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.000   1st Qu.:0.0000   1st Qu.:0.0000
 Median :1.000   Median :0.0000   Median :0.0000
 Mean   :0.592   Mean   :0.2856   Mean   :0.4236
 3rd Qu.:1.000   3rd Qu.:1.0000   3rd Qu.:1.0000
 Max.   :1.000   Max.   :1.0000   Max.   :1.0000
  Education.2      Education.3
 Min.   :0.0000   Min.   :0.0000
 1st Qu.:0.0000   1st Qu.:0.0000
 Median :0.0000   Median :0.0000
 Mean   :0.2848   Mean   :0.2916
 3rd Qu.:1.0000   3rd Qu.:1.0000
 Max.   :1.0000   Max.   :1.0000
```

Hide

```
summary(Validation_Data)
```

/

```
     Age          Experience        Income
Min.   :23.00   Min.   :-3.00   Min.   :  8.00
1st Qu.:35.00   1st Qu.:10.00   1st Qu.: 39.00
Median :45.00   Median :20.00   Median : 63.00
Mean   :44.93   Mean   :19.67   Mean   : 73.59
3rd Qu.:55.00   3rd Qu.:29.00   3rd Qu.: 98.00
Max.   :67.00   Max.   :42.00   Max.   :218.00
    Family          CCAvg           Mortgage
Min.   :1.000   Min.   : 0.000   Min.   :  0.00
1st Qu.:1.000   1st Qu.: 0.700   1st Qu.:  0.00
Median :2.000   Median : 1.500   Median :  0.00
Mean   :2.381   Mean   : 1.941   Mean   : 58.85
3rd Qu.:3.000   3rd Qu.: 2.500   3rd Qu.:104.00
Max.   :4.000   Max.   :10.000   Max.   :590.00
PersonalLoan SecuritiesAccount   CDAccount
0:1356       Min.   :0.0000    Min.   :0.00000
1: 144       1st Qu.:0.0000    1st Qu.:0.00000
             Median :0.0000    Median :0.00000
             Mean   :0.1127    Mean   :0.06133
             3rd Qu.:0.0000    3rd Qu.:0.00000
             Max.   :1.0000    Max.   :1.00000
    Online         CreditCard      Education.1
Min.   :0.0000   Min.   :0.0000   Min.   :0.0000
1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000
Median :1.0000   Median :0.0000   Median :0.0000
Mean   :0.6027   Mean   :0.2973   Mean   :0.4113
3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000
Max.   :1.0000   Max.   :1.0000   Max.   :1.0000
 Education.2      Education.3
Min.   :0.0000   Min.   :0.0000
1st Qu.:0.0000   1st Qu.:0.0000
Median :0.0000   Median :0.0000
Mean   :0.2813   Mean   :0.3073
3rd Qu.:1.0000   3rd Qu.:1.0000
Max.   :1.0000   Max.   :1.0000
```

Hide

/

```
## Normalization
# Copy the original data
train.norm.df1 <- Train_Data
valid.norm.df1 <- ValTest_data
test.norm.df1 <- Test_Data
norm.values <- preProcess(Train_Data[, 1:14], method=c("center", "scale"))

train.norm.df1[, 1:14] <- predict(norm.values, Train_Data[, 1:14])
valid.norm.df1[, 1:14] <- predict(norm.values, ValTest_data[, 1:14])
test.norm.df1[, 1:14] <- predict(norm.values, Test_Data[, 1:14])
norm.values <- preProcess(Train_Data[, 1:14], method=c("center", "scale"))
```

Hide

```
summary(train.norm.df1)
```

/

```
      Age              Experience           Income      
 Min.   :-1.97910   Min.   :-2.04693   Min.   :-1.4397  
 1st Qu.:-0.84189   1st Qu.:-0.82237   1st Qu.:-0.7650  
 Median : 0.03289   Median : 0.05231   Median :-0.2209  
 Mean   : 0.00000   Mean   : 0.00000   Mean   : 0.0000  
 3rd Qu.: 0.90767   3rd Qu.: 0.83952   3rd Qu.: 0.5409  
 Max.   : 1.86992   Max.   : 1.97660   Max.   : 2.8480  
     Family            CCAvg            Mortgage      
 Min.   :-1.2170   Min.   :-1.1213   Min.   :-0.5660  
 1st Qu.:-1.2170   1st Qu.:-0.7113   1st Qu.:-0.5660  
 Median :-0.3534   Median :-0.2428   Median :-0.5660  
 Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 1.3739   3rd Qu.: 0.4015   3rd Qu.: 0.4342  
 Max.   : 1.3739   Max.   : 4.7356   Max.   : 5.7066  
 PersonalLoan SecuritiesAccount   CDAccount        
 0:2260       Min.   :-0.3392   Min.   :-0.2463  
 1: 240       1st Qu.:-0.3392   1st Qu.:-0.2463  
              Median :-0.3392   Median :-0.2463  
              Mean   : 0.0000   Mean   : 0.0000  
              3rd Qu.:-0.3392   3rd Qu.:-0.2463  
              Max.   : 2.9473   Max.   : 4.0591  
     Online         CreditCard       Education.1      
 Min.   :-1.204   Min.   :-0.6322   Min.   :-0.8571  
 1st Qu.:-1.204   1st Qu.:-0.6322   1st Qu.:-0.8571  
 Median : 0.830   Median :-0.6322   Median :-0.8571  
 Mean   : 0.000   Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 0.830   3rd Qu.: 1.5813   3rd Qu.: 1.1663  
 Max.   : 0.830   Max.   : 1.5813   Max.   : 1.1663  
  Education.2       Education.3      
 Min.   :-0.6309   Min.   :-0.6415  
 1st Qu.:-0.6309   1st Qu.:-0.6415  
 Median :-0.6309   Median :-0.6415  
 Mean   : 0.0000   Mean   : 0.0000  
 3rd Qu.: 1.5844   3rd Qu.: 1.5583  
 Max.   : 1.5844   Max.   : 1.5583  
```

Hide

```
summary(valid.norm.df1)
```

/

```
     Age              Experience           Income
Min.   :-1.97910   Min.   :-2.04693   Min.   :-1.43974
1st Qu.:-0.92936   1st Qu.:-0.90984   1st Qu.:-0.76502
Median :-0.05459   Median :-0.03516   Median :-0.24266
Mean   :-0.04997   Mean   :-0.05203   Mean   :-0.01631
3rd Qu.: 0.82019   3rd Qu.: 0.75205   3rd Qu.: 0.51912
Max.   : 1.86992   Max.   : 1.88913   Max.   : 3.26153
    Family             CCAvg             Mortgage
Min.   :-1.21705   Min.   :-1.12133   Min.   :-0.56598
1st Qu.:-1.21705   1st Qu.:-0.71135   1st Qu.:-0.56598
Median :-0.35340   Median :-0.24279   Median :-0.56598
Mean   :-0.02211   Mean   : 0.02742   Mean   :-0.01576
3rd Qu.: 0.51024   3rd Qu.: 0.34291   3rd Qu.: 0.41442
Max.   : 1.37389   Max.   : 4.73564   Max.   : 5.47939
PersonalLoan SecuritiesAccount     CDAccount
0:2260       Min.   :-0.339160   Min.   :-0.24626
1: 240       1st Qu.:-0.339160   1st Qu.:-0.24626
             Median :-0.339160   Median :-0.24626
             Mean   : 0.007888   Mean   : 0.02755
             3rd Qu.:-0.339160   3rd Qu.:-0.24626
             Max.   : 2.947278   Max.   : 4.05905
    Online           CreditCard        Education.1
Min.   :-1.20433   Min.   :-0.63215   Min.   :-0.85710
1st Qu.:-1.20433   1st Qu.:-0.63215   1st Qu.:-0.85710
Median : 0.83001   Median :-0.63215   Median :-0.85710
Mean   : 0.01953   Mean   : 0.03719   Mean   :-0.01781
3rd Qu.: 0.83001   3rd Qu.: 1.58127   3rd Qu.: 1.16626
Max.   : 0.83001   Max.   : 1.58127   Max.   : 1.16626
 Education.2        Education.3
Min.   :-0.63091   Min.   :-0.64146
1st Qu.:-0.63091   1st Qu.:-0.64146
Median :-0.63091   Median :-0.64146
Mean   :-0.01861   Mean   : 0.03784
3rd Qu.: 1.58437   3rd Qu.: 1.55833
Max.   : 1.58437   Max.   : 1.55833
```

Hide

```
##Let’s now run knn on the training set, and compare the confusion matrices on the validation and test sets
##Apply the k-NN method with the k = 3


 knn.pred <- knn(train.norm.df1[,-7], valid.norm.df1[,-7],
                 cl = train.norm.df1[,7], k = 3)
accuracy.df <- confusionMatrix(knn.pred, valid.norm.df1[,7])$overall[1]
accuracy.df
```

```
Accuracy
  0.9676
```

Hide

Hide

```
CrossTable(x=valid.norm.df1[,7],y=knn.pred)
```

```
   Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  2500


                     | knn.pred
valid.norm.df1[, 7] |         0 |         1 | Row Total |
--------------------|-----------|-----------|-----------|
                  0 |      2256 |         4 |      2260 |
                    |    10.241 |   143.074 |           |
                    |     0.998 |     0.002 |     0.904 |
                    |     0.967 |     0.024 |           |
                    |     0.902 |     0.002 |           |
--------------------|-----------|-----------|-----------|
                  1 |        77 |       163 |       240 |
                    |    96.441 |  1347.280 |           |
                    |     0.321 |     0.679 |     0.096 |
                    |     0.033 |     0.976 |           |
                    |     0.031 |     0.065 |           |
--------------------|-----------|-----------|-----------|
       Column Total |      2333 |       167 |      2500 |
                    |     0.933 |     0.067 |           |
--------------------|-----------|-----------|-----------|
```

Hide

```
# Test Data
knn.pred <- knn(train.norm.df1[,-7], test.norm.df1[,-7],
                 cl = train.norm.df1[,7], k = 3)
accuracy.df <- confusionMatrix(knn.pred, test.norm.df1[,7])$overall[1]
accuracy.df
```

```
Accuracy
   0.971
```

Hide

```
CrossTable(x=test.norm.df1[,7], y=knn.pred)
```

```
   Cell Contents
|-------------------------|
|                       N |
| Chi-square contribution |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  1000


                   | knn.pred
test.norm.df1[, 7] |         0 |         1 | Row Total |
-------------------|-----------|-----------|-----------|
                 0 |       901 |         3 |       904 |
                   |     4.735 |    60.128 |           |
                   |     0.997 |     0.003 |     0.904 |
                   |     0.972 |     0.041 |           |
                   |     0.901 |     0.003 |           |
-------------------|-----------|-----------|-----------|
                 1 |        26 |        70 |        96 |
                   |    44.588 |   566.209 |           |
                   |     0.271 |     0.729 |     0.096 |
                   |     0.028 |     0.959 |           |
                   |     0.026 |     0.070 |           |
-------------------|-----------|-----------|-----------|
      Column Total |       927 |        73 |      1000 |
                   |     0.927 |     0.073 |           |
-------------------|-----------|-----------|-----------|
```