FML_Assignment3

Hruthik M

{r setup, include=FALSE}S knitr::opts_chunk$set(echo = TRUE)

```r
#loading the libraries
library(class)
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
```

```r
library(caret)
```

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

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

```r
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1
## v readr     2.1.5
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x purrr::lift()   masks caret::lift()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```r
library(gmodels)
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
##
## The following object is masked from 'package:tidyr':
##
##     smiths
```

```r
library(e1071)
```

#loading dataset

```r
dataset_ub <-  read.csv("C:/Users/santo/OneDrive/Desktop/Fundamental of machinelearning/Assignment_2/Uni
head(dataset_ub)
```

```
##   ID Age Experience Income ZIP.Code Family CCAvg Education Mortgage
## 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
##   Personal.Loan Securities.Account CD.Account Online CreditCard
## 1             0                  1          0      0          0
## 2             0                  1          0      0          0
## 3             0                  0          0      0          0
## 4             0                  0          0      0          0
## 5             0                  0          0      0          1
## 6             0                  0          0      1          0
```

```r
#removing unwanted columns i.e ID and Zip code
dataset_ub1<-dataset_ub[,-1]
head(dataset_ub1)
```

```
##   Age Experience Income ZIP.Code Family CCAvg Education Mortgage Personal.Loan
## 1  25          1     49    91107      4   1.6         1        0             0
## 2  45         19     34    90089      3   1.5         1        0             0
## 3  39         15     11    94720      1   1.0         1        0             0
## 4  35          9    100    94112      1   2.7         2        0             0
## 5  35          8     45    91330      4   1.0         2        0             0
## 6  37         13     29    92121      4   0.4         2      155             0
##   Securities.Account CD.Account Online CreditCard
## 1                  1          0      0          0
## 2                  1          0      0          0
## 3                  0          0      0          0
## 4                  0          0      0          0
## 5                  0          0      0          1
## 6                  0          0      1          0
```

```r
dataset_ub1<-dataset_ub1[,-4]
head(dataset_ub1)
```

```
##   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
## 3  39         15     11      1   1.0         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
##   Securities.Account CD.Account Online CreditCard
## 1                  1          0      0          0
## 2                  1          0      0          0
## 3                  0          0      0          0
## 4                  0          0      0          0
## 5                  0          0      0          1
## 6                  0          0      1          0
```

```
#converting personal loan as factor
dataset_ub1$Personal.Loan=as.factor(dataset_ub1$Personal.Loan)

#running is.na to check if there are any NA values
head(is.na(dataset_ub1))
```

```
##        Age Experience Income Family CCAvg Education Mortgage Personal.Loan
## [1,] FALSE      FALSE  FALSE  FALSE FALSE     FALSE    FALSE         FALSE
## [2,] FALSE      FALSE  FALSE  FALSE FALSE     FALSE    FALSE         FALSE
## [3,] FALSE      FALSE  FALSE  FALSE FALSE     FALSE    FALSE         FALSE
## [4,] FALSE      FALSE  FALSE  FALSE FALSE     FALSE    FALSE         FALSE
## [5,] FALSE      FALSE  FALSE  FALSE FALSE     FALSE    FALSE         FALSE
## [6,] FALSE      FALSE  FALSE  FALSE FALSE     FALSE    FALSE         FALSE
##      Securities.Account CD.Account Online CreditCard
## [1,]              FALSE      FALSE  FALSE      FALSE
## [2,]              FALSE      FALSE  FALSE      FALSE
## [3,]              FALSE      FALSE  FALSE      FALSE
## [4,]              FALSE      FALSE  FALSE      FALSE
## [5,]              FALSE      FALSE  FALSE      FALSE
## [6,]              FALSE      FALSE  FALSE      FALSE
```

```
any(is.na(dataset_ub1))
```

```
## [1] FALSE
```

```
# Converting categorical variable into i.e education into dummy variables

#converting education into character
education<-as.character(dataset_ub1$Education)

dataset_ub2<-cbind(dataset_ub1[,-6],education)
head(dataset_ub2)
```

```
##   Age Experience Income Family CCAvg Mortgage Personal.Loan Securities.Account
## 1  25          1     49      4   1.6        0             0                  1
## 2  45         19     34      3   1.5        0             0                  1
## 3  39         15     11      1   1.0        0             0                  0
```

```
## 4  35          9    100      1   2.7        0              0                  0
## 5  35          8     45      4   1.0        0              0                  0
## 6  37         13     29      4   0.4      155              0                  0
##   CD.Account Online CreditCard education
## 1          0      0          0         1
## 2          0      0          0         1
## 3          0      0          0         1
## 4          0      0          0         2
## 5          0      0          1         2
## 6          0      1          0         2
```

```
dummymodel<-dummyVars("~education",data = dataset_ub2)
educationdummy<-data.frame(predict(dummymodel,dataset_ub2))
head(educationdummy)
```

```
##   education1 education2 education3
## 1          1          0          0
## 2          1          0          0
## 3          1          0          0
## 4          0          1          0
## 5          0          1          0
## 6          0          1          0
```

```
dataset_ub_dummy<-cbind(dataset_ub2[,-12],educationdummy)
head(dataset_ub_dummy)
```

```
##   Age Experience Income Family CCAvg Mortgage Personal.Loan Securities.Account
## 1  25          1     49      4   1.6        0              0                  1
## 2  45         19     34      3   1.5        0              0                  1
## 3  39         15     11      1   1.0        0              0                  0
## 4  35          9    100      1   2.7        0              0                  0
## 5  35          8     45      4   1.0        0              0                  0
## 6  37         13     29      4   0.4      155              0                  0
##   CD.Account Online CreditCard education1 education2 education3
## 1          0      0          0          1          0          0
## 2          0      0          0          1          0          0
## 3          0      0          0          1          0          0
## 4          0      0          0          0          1          0
## 5          0      0          1          0          1          0
## 6          0      1          0          0          1          0
```

```
#dividing data into training and testing set
set.seed(555)
train<-createDataPartition(dataset_ub_dummy$Personal.Loan,p=0.60,list = FALSE)
trainset<-dataset_ub_dummy[train,]
nrow(trainset)
```

```
## [1] 3000
```

```
validationset<-dataset_ub_dummy[-train,]
nrow(validationset)
```

```
## [1] 2000
```

```r
testset<-data.frame(Age = 40, Experience = 10, Income = 84, Family = 2, CCAvg = 2,  Mortgage = 0, Secur
     CreditCard = 1,education1 = 0, education2 = 1, education3 = 0)


summary(trainset)
```

```
##       Age          Experience        Income           Family     
##  Min.   :23.00   Min.   :-3.00   Min.   :  8.00   Min.   :1.000  
##  1st Qu.:35.00   1st Qu.:10.00   1st Qu.: 40.00   1st Qu.:1.000  
##  Median :45.00   Median :20.00   Median : 65.00   Median :2.000  
##  Mean   :45.31   Mean   :20.08   Mean   : 74.81   Mean   :2.382  
##  3rd Qu.:55.00   3rd Qu.:30.00   3rd Qu.:100.00   3rd Qu.:3.000  
##  Max.   :67.00   Max.   :43.00   Max.   :224.00   Max.   :4.000  
##      CCAvg           Mortgage      Personal.Loan Securities.Account
##  Min.   : 0.000   Min.   :  0.00   0:2712        Min.   :0.0000    
##  1st Qu.: 0.700   1st Qu.:  0.00   1: 288        1st Qu.:0.0000    
##  Median : 1.500   Median :  0.00                 Median :0.0000    
##  Mean   : 1.946   Mean   : 56.32                 Mean   :0.1067    
##  3rd Qu.: 2.600   3rd Qu.:101.00                 3rd Qu.:0.0000    
##  Max.   :10.000   Max.   :635.00                 Max.   :1.0000    
##    CD.Account          Online         CreditCard      education1    
##  Min.   :0.00000   Min.   :0.0000   Min.   :0.000   Min.   :0.0000  
##  1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.000   1st Qu.:0.0000  
##  Median :0.00000   Median :1.0000   Median :0.000   Median :0.0000  
##  Mean   :0.06167   Mean   :0.5963   Mean   :0.297   Mean   :0.4267  
##  3rd Qu.:0.00000   3rd Qu.:1.0000   3rd Qu.:1.000   3rd Qu.:1.0000  
##  Max.   :1.00000   Max.   :1.0000   Max.   :1.000   Max.   :1.0000  
##    education2     education3    
##  Min.   :0.00   Min.   :0.0000  
##  1st Qu.:0.00   1st Qu.:0.0000  
##  Median :0.00   Median :0.0000  
##  Mean   :0.28   Mean   :0.2933  
##  3rd Qu.:1.00   3rd Qu.:1.0000  
##  Max.   :1.00   Max.   :1.0000
```

```r
summary(validationset)
```

```
##       Age          Experience        Income           Family     
##  Min.   :23.00   Min.   :-3.00   Min.   :  8.00   Min.   :1.000  
##  1st Qu.:35.00   1st Qu.:10.00   1st Qu.: 38.00   1st Qu.:1.000  
##  Median :45.50   Median :20.00   Median : 62.00   Median :2.000  
##  Mean   :45.38   Mean   :20.14   Mean   : 72.22   Mean   :2.418  
##  3rd Qu.:55.00   3rd Qu.:30.00   3rd Qu.: 94.00   3rd Qu.:4.000  
##  Max.   :67.00   Max.   :43.00   Max.   :205.00   Max.   :4.000  
##      CCAvg          Mortgage      Personal.Loan Securities.Account
##  Min.   :0.000   Min.   :  0.00   0:1808        Min.   :0.000     
##  1st Qu.:0.700   1st Qu.:  0.00   1: 192        1st Qu.:0.000     
##  Median :1.500   Median :  0.00                 Median :0.000     
##  Mean   :1.925   Mean   : 56.77                 Mean   :0.101     
##  3rd Qu.:2.500   3rd Qu.:101.00                 3rd Qu.:0.000     
##  Max.   :9.300   Max.   :617.00                 Max.   :1.000
```

```
##    CD.Account          Online          CreditCard        education1
##  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 :1.0000   Median :0.0000   Median :0.000
##  Mean   :0.0585   Mean   :0.5975   Mean   :0.2895   Mean   :0.408
##  3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.000
##    education2        education3
##  Min.   :0.0000   Min.   :0.0000
##  1st Qu.:0.0000   1st Qu.:0.0000
##  Median :0.0000   Median :0.0000
##  Mean   :0.2815   Mean   :0.3105
##  3rd Qu.:1.0000   3rd Qu.:1.0000
##  Max.   :1.0000   Max.   :1.0000
```

```r
summary(testset)
```

```
##       Age       Experience      Income        Family       CCAvg      Mortgage
##  Min.   :40   Min.   :10   Min.   :84   Min.   :2   Min.   :2   Min.   :0
##  1st Qu.:40   1st Qu.:10   1st Qu.:84   1st Qu.:2   1st Qu.:2   1st Qu.:0
##  Median :40   Median :10   Median :84   Median :2   Median :2   Median :0
##  Mean   :40   Mean   :10   Mean   :84   Mean   :2   Mean   :2   Mean   :0
##  3rd Qu.:40   3rd Qu.:10   3rd Qu.:84   3rd Qu.:2   3rd Qu.:2   3rd Qu.:0
##  Max.   :40   Max.   :10   Max.   :84   Max.   :2   Max.   :2   Max.   :0
##  Securities.Account   CD.Account     Online     CreditCard   education1
##  Min.   :0          Min.   :0    Min.   :1    Min.   :1    Min.   :0
##  1st Qu.:0          1st Qu.:0    1st Qu.:1    1st Qu.:1    1st Qu.:0
##  Median :0          Median :0    Median :1    Median :1    Median :0
##  Mean   :0          Mean   :0    Mean   :1    Mean   :1    Mean   :0
##  3rd Qu.:0          3rd Qu.:0    3rd Qu.:1    3rd Qu.:1    3rd Qu.:0
##  Max.   :0          Max.   :0    Max.   :1    Max.   :1    Max.   :0
##    education2   education3
##  Min.   :1    Min.   :0
##  1st Qu.:1    1st Qu.:0
##  Median :1    Median :0
##  Mean   :1    Mean   :0
##  3rd Qu.:1    3rd Qu.:0
##  Max.   :1    Max.   :0
```

#normalizing

```r
normvar<-c('Age',"Experience","Income","Family","CCAvg","Mortgage","Securities.Account","CD.Account","On
normalization_values<-preProcess(trainset[,normvar],method = c('center','scale'))

trainset.norm<-predict(normalization_values,trainset)
summary(trainset.norm)
```

```
##       Age               Experience          Income             Family
##  Min.   :-1.95104   Min.   :-2.0186   Min.   :-1.4431   Min.   :-1.2107
##  1st Qu.:-0.90159   1st Qu.:-0.8817   1st Qu.:-0.7519   1st Qu.:-1.2107
##  Median :-0.02705   Median :-0.0072   Median :-0.2119   Median :-0.3344
##  Mean   : 0.00000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
```

```
##  3rd Qu.: 0.84749   3rd Qu.: 0.8673   3rd Qu.: 0.5441   3rd Qu.: 0.5418
##  Max.   : 1.89694   Max.   : 2.0042   Max.   : 3.2226   Max.   : 1.4180
##      CCAvg           Mortgage       Personal.Loan Securities.Account
##  Min.   :-1.0976   Min.   :-0.5527   0:2712        Min.   :-0.3455
##  1st Qu.:-0.7028   1st Qu.:-0.5527   1: 288        1st Qu.:-0.3455
##  Median :-0.2517   Median :-0.5527                 Median :-0.3455
##  Mean   : 0.0000   Mean   : 0.0000                 Mean   : 0.0000
##  3rd Qu.: 0.3687   3rd Qu.: 0.4385                 3rd Qu.:-0.3455
##  Max.   : 4.5418   Max.   : 5.6790                 Max.   : 2.8935
##    CD.Account          Online          CreditCard        education1
##  Min.   :-0.2563   Min.   :-1.2152   Min.   :-0.6499   Min.   :-0.8625
##  1st Qu.:-0.2563   1st Qu.:-1.2152   1st Qu.:-0.6499   1st Qu.:-0.8625
##  Median :-0.2563   Median : 0.8226   Median :-0.6499   Median :-0.8625
##  Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.:-0.2563   3rd Qu.: 0.8226   3rd Qu.: 1.5383   3rd Qu.: 1.1590
##  Max.   : 3.9001   Max.   : 0.8226   Max.   : 1.5383   Max.   : 1.1590
##    education2        education3
##  Min.   :-0.6235   Min.   :-0.6442
##  1st Qu.:-0.6235   1st Qu.:-0.6442
##  Median :-0.6235   Median :-0.6442
##  Mean   : 0.0000   Mean   : 0.0000
##  3rd Qu.: 1.6033   3rd Qu.: 1.5519
##  Max.   : 1.6033   Max.   : 1.5519
```

```
validationset.norm<-predict(normalization_values,validationset)
summary(validationset.norm)
```

```
##       Age              Experience              Income              Family
##  Min.   :-1.951044   Min.   :-2.018590   Min.   :-1.44310   Min.   :-1.21067
##  1st Qu.:-0.901594   1st Qu.:-0.881718   1st Qu.:-0.79509   1st Qu.:-1.21067
##  Median : 0.016675   Median :-0.007200   Median :-0.27668   Median :-0.33443
##  Mean   : 0.006355   Mean   : 0.004868   Mean   :-0.05588   Mean   : 0.03227
##  3rd Qu.: 0.847489   3rd Qu.: 0.867317   3rd Qu.: 0.41453   3rd Qu.: 1.41805
##  Max.   : 1.896939   Max.   : 2.004190   Max.   : 2.81218   Max.   : 1.41805
##      CCAvg              Mortgage          Personal.Loan Securities.Account
##  Min.   :-1.09759   Min.   :-0.552664   0:1808        Min.   :-0.34549
##  1st Qu.:-0.70283   1st Qu.:-0.552664   1: 192        1st Qu.:-0.34549
##  Median :-0.25168   Median :-0.552664                 Median :-0.34549
##  Mean   :-0.01177   Mean   : 0.004477                 Mean   :-0.01835
##  3rd Qu.: 0.31226   3rd Qu.: 0.438506                 3rd Qu.:-0.34549
##  Max.   : 4.14705   Max.   : 5.502307                 Max.   : 2.89348
##    CD.Account           Online             CreditCard         education1
##  Min.   :-0.25632   Min.   :-1.215236   Min.   :-0.64987   Min.   :-0.86252
##  1st Qu.:-0.25632   1st Qu.:-1.215236   1st Qu.:-0.64987   1st Qu.:-0.86252
##  Median :-0.25632   Median : 0.822611   Median :-0.64987   Median :-0.86252
##  Mean   :-0.01316   Mean   : 0.002377   Mean   :-0.01641   Mean   :-0.03774
##  3rd Qu.:-0.25632   3rd Qu.: 0.822611   3rd Qu.: 1.53825   3rd Qu.: 1.15901
##  Max.   : 3.90015   Max.   : 0.822611   Max.   : 1.53825   Max.   : 1.15901
##    education2          education3
##  Min.   :-0.62351   Min.   :-0.6442
##  1st Qu.:-0.62351   1st Qu.:-0.6442
##  Median :-0.62351   Median :-0.6442
##  Mean   : 0.00334   Mean   : 0.0377
##  3rd Qu.: 1.60330   3rd Qu.: 1.5519
```

```
## Max.   : 1.60330   Max.   : 1.5519
```

```r
testset.norm<-predict(normalization_values,testset)
summary(testset.norm)
```

```
##       Age           Experience          Income           Family
##  Min.   :-0.4643   Min.   :-0.8817   Min.   :0.1985   Min.   :-0.3344
##  1st Qu.:-0.4643   1st Qu.:-0.8817   1st Qu.:0.1985   1st Qu.:-0.3344
##  Median :-0.4643   Median :-0.8817   Median :0.1985   Median :-0.3344
##  Mean   :-0.4643   Mean   :-0.8817   Mean   :0.1985   Mean   :-0.3344
##  3rd Qu.:-0.4643   3rd Qu.:-0.8817   3rd Qu.:0.1985   3rd Qu.:-0.3344
##  Max.   :-0.4643   Max.   :-0.8817   Max.   :0.1985   Max.   :-0.3344
##      CCAvg            Mortgage       Securities.Account   CD.Account
##  Min.   :0.03029   Min.   :-0.5527   Min.   :-0.3455    Min.   :-0.2563
##  1st Qu.:0.03029   1st Qu.:-0.5527   1st Qu.:-0.3455    1st Qu.:-0.2563
##  Median :0.03029   Median :-0.5527   Median :-0.3455    Median :-0.2563
##  Mean   :0.03029   Mean   :-0.5527   Mean   :-0.3455    Mean   :-0.2563
##  3rd Qu.:0.03029   3rd Qu.:-0.5527   3rd Qu.:-0.3455    3rd Qu.:-0.2563
##  Max.   :0.03029   Max.   :-0.5527   Max.   :-0.3455    Max.   :-0.2563
##      Online          CreditCard      education1         education2
##  Min.   :0.8226   Min.   :1.538   Min.   :-0.8625   Min.   :1.603
##  1st Qu.:0.8226   1st Qu.:1.538   1st Qu.:-0.8625   1st Qu.:1.603
##  Median :0.8226   Median :1.538   Median :-0.8625   Median :1.603
##  Mean   :0.8226   Mean   :1.538   Mean   :-0.8625   Mean   :1.603
##  3rd Qu.:0.8226   3rd Qu.:1.538   3rd Qu.:-0.8625   3rd Qu.:1.603
##  Max.   :0.8226   Max.   :1.538   Max.   :-0.8625   Max.   :1.603
##    education3
##  Min.   :-0.6442
##  1st Qu.:-0.6442
##  Median :-0.6442
##  Mean   :-0.6442
##  3rd Qu.:-0.6442
##  Max.   :-0.6442
```

##Question A ##Create a pivot table for the training data with Online as a column variable, CC as a rowvariable, and Loan as a secondary row variable. The values inside the table should convey the count. In R use functions melt() and cast(), or function table(). In Python, use pandadataframe methods melt() and pivot().

```r
trainset$Personal.Loan <- as.factor(trainset$Personal.Loan)

pivot_table <- table(trainset$CC, trainset$Personal.Loan, trainset$Online)

# Display the pivot table
head(pivot_table)
```

```
## , ,  = 0
##
##
##         0   1
##   0    29   0
##   0.1  46   1
```

```
##   0.2 54  2
##   0.3 57  0
##   0.4 44  1
##   0.5 32  3
##
## , ,  = 1
##
##
##        0  1
##   0   43  0
##   0.1 66  1
##   0.2 68  4
##   0.3 85  3
##   0.4 60  0
##   0.5 67  2
```

```
#If you want to visualize the pivot table in a more tabular format use data frame function
pivot_df <- as.data.frame.table(pivot_table)
head(pivot_df)
```

```
##   Var1 Var2 Var3 Freq
## 1    0    0    0   29
## 2  0.1    0    0   46
## 3  0.2    0    0   54
## 4  0.3    0    0   57
## 5  0.4    0    0   44
## 6  0.5    0    0   32
```

##Explanation ##This code is prints the data stored in trainset, specifically examining relationships between the variables CC, Personal.Loan, and Online through the creation of a pivot table and subsequent conversion to a dataframe for further analysis or visualization.

##Question B ##Consider the task of classifying a customer who owns a bank credit card and is actively usingonline banking services. Looking at the pivot table, what is the probability that this customerwill accept the loan offer? [This is the probability of loan acceptance (Loan = 1) conditional onhaving a bank credit card (CC = 1) and being an active user of online banking services (Online = 1)]

```
pivot_table_loan_cc_online <- table(trainset.norm$Personal.Loan, trainset.norm$CreditCard, trainset.norm

count_loan_1_cc_1_online_1 <- pivot_table_loan_cc_online[2, 2, 2]

count_cc_1_online_1 <- sum(pivot_table_loan_cc_online[, 2, 2])

probability_loan_acceptance <- count_loan_1_cc_1_online_1 / count_cc_1_online_1

# Output the result
cat("Probability of loan acceptance given CC = 1 and Online = 1:", probability_loan_acceptance, "\n")
```

```
## Probability of loan acceptance given CC = 1 and Online = 1: 0.1005587
```

##Explanation ##The provided code calculates the probability of loan acceptance for customers who have an average credit card spending of 1 and are actively using online banking services.

##Question C ##Create two separate pivot tables for the training data. One will have Loan (rows) as a function of Online (columns) and the other will have Loan (rows) as a function of CC

```r
# Pivot table for Loan as a function of Online
table_loan_online <- table(trainset.norm$Personal.Loan, trainset.norm$Online)
print(table_loan_online)
```

```
##
##     -1.21523621742208 0.822610988987221
##   0              1098              1614
##   1               113               175
```

```r
# Pivot table for Loan as a function of CC
table_loan_cc <- table(trainset.norm$Personal.Loan, trainset.norm$CreditCard)
print(table_loan_cc)
```

```
##
##     -0.649872511843724 1.53825042365703
##   0               1919              793
##   1                190               98
```

##Explanation ##These pivot tables provide valuable insights into the relationships between loan acceptance and the variables Online and CreditCard, respectively. They help in understanding the distribution of loan acceptance based on these factors.

#Question D ## Compute the following quantities [P(A | B) means "the probability ofA given B"]: #i. P(CC = 1 | Loan = 1) (the proportion of credit card holders among the loan acceptors) #ii. P(Online = 1 | Loan = 1) #iii. P(Loan = 1) (the proportion of loan acceptors) #iv. P(CC = 1 | Loan = 0) #v. P(Online = 1 | Loan = 0) #vi. P(Loan = 0)

```r
# Compute the probabilities
# i. P(CC = 1 | Loan = 1)
p_cc_given_loan_1 <- trainset.norm %>%
filter(Personal.Loan == 1) %>%
summarize(p_cc_given_loan_1 = mean(CreditCard == 1))
cat("i. P(CC = 1 | Loan = 1):", p_cc_given_loan_1$p_cc_given_loan_1, "\n")
```

```
## i. P(CC = 1 | Loan = 1): 0
```

```r
# ii. P(Online = 1 | Loan = 1)
p_online_given_loan_1 <- trainset.norm %>%
filter(Personal.Loan == 1) %>%
summarize(p_online_given_loan_1 = mean(Online == 1))
cat("ii. P(Online = 1 | Loan = 1):", p_online_given_loan_1$p_online_given_loan_1, "\n")
```

```
## ii. P(Online = 1 | Loan = 1): 0
```

```r
# iii. P(Loan = 1)
p_loan_1 <- mean(trainset.norm$Personal.Loan == 1)
cat("iii. P(Loan = 1):", p_loan_1, "\n")
```

```
## iii. P(Loan = 1): 0.096
```

```r
# iv. P(CC = 1 | Loan = 0)
p_cc_given_loan_0 <- trainset.norm %>%
filter(Personal.Loan == 0) %>%
summarize(p_cc_given_loan_0 = mean(CreditCard == 1))
cat("iv. P(CC = 1 | Loan = 0):", p_cc_given_loan_0$p_cc_given_loan_0, "\n")
```

```
## iv. P(CC = 1 | Loan = 0): 0
```

```r
# v. P(Online = 1 | Loan = 0)
p_online_given_loan_0 <- trainset.norm %>%
filter(Personal.Loan == 0) %>%
summarize(p_online_given_loan_0 = mean(Online == 1))
cat("v. P(Online = 1 | Loan = 0):", p_online_given_loan_0$p_online_given_loan_0, "\n")
```

```
## v. P(Online = 1 | Loan = 0): 0
```

```r
# vi. P(Loan = 0)
p_loan_0 <- mean(trainset.norm$Personal.Loan == 0)
cat("vi. P(Loan = 0):", p_loan_0, "\n")
```

```
## vi. P(Loan = 0): 0.904
```

##Explanation #i. P(CC = 1 | Loan = 1) (the proportion of credit card holders among the loan acceptors)= 0 #ii. P(Online = 1 | Loan = 1)= 0 #iii. P(Loan = 1) (the proportion of loan acceptors)= 0.096 #iv. P(CC = 1 | Loan = 0)= 0 #v. P(Online = 1 | Loan = 0)= 0 #vi. P(Loan = 0)= 0.904

##Question E ## Use the quantities computed above to compute the naive Bayes probability P(Loan = 1 | CC= 1, Online = 1)

```r
p_loan_1_given_cc_online <- function(cc_1, online_1) {
p_cc_1 <- p_cc_given_loan_1$p_cc_given_loan_1
p_online_1 <- p_online_given_loan_1$p_online_given_loan_1
p_cc_online <- p_cc_1 * p_online_1  # P(CC = 1) * P(Online = 1)

# Using Bayes' theorem
numerator <- p_cc_1 * p_online_1 * p_loan_1
denominator <- p_cc_online
p_loan_1_given_cc_online <- numerator / denominator

return(p_loan_1_given_cc_online)
}

p_loan_1_given_cc_online <- p_loan_1_given_cc_online(cc_1 = 1, online_1 = 1)
```

```r
p_loan_1_given_cc_online <- function(cc_1, online_1) {
p_cc_1 <- p_cc_given_loan_1$p_cc_given_loan_1
p_online_1 <- p_online_given_loan_1$p_online_given_loan_1
p_cc_online <- p_cc_1 * p_online_1  # P(CC = 1) * P(Online = 1)

# Using Bayes' theorem
numerator <- p_cc_1 * p_online_1 * p_loan_1
```

```r
denominator <- p_cc_online
p_loan_1_given_cc_online <- numerator / denominator

return(p_loan_1_given_cc_online)
}

p_loan_1_given_cc_online <- p_loan_1_given_cc_online(cc_1 = 1, online_1 = 1)

# Output the result
cat("P(Loan = 1 | CC = 1, Online = 1):", p_loan_1_given_cc_online, "\n")
```

```
## P(Loan = 1 | CC = 1, Online = 1): NaN
```

##Explanation #The probability P(Loan = 1 | CC= 1, Online = 1) using naive Bayes is NaN.

##Question F ##Compare this value with the one obtained from the pivot table in (B). Which is a more accurate estimate?

```r
pivot_table_loan_cc_online <- table(trainset.norm$Personal.Loan, trainset.norm$CreditCard, trainset.norm

count_loan_1_cc_1_online_1 <- pivot_table_loan_cc_online[2, 2, 2]

count_cc_1_online_1 <- sum(pivot_table_loan_cc_online[, 2, 2])

p_loan_1_given_cc_online_pivot <- count_loan_1_cc_1_online_1 / count_cc_1_online_1

# Output the result
cat("P(Loan = 1 | CC = 1, Online = 1) from pivot table:", p_loan_1_given_cc_online_pivot, "\n")
```

```
## P(Loan = 1 | CC = 1, Online = 1) from pivot table: 0.1005587
```

##Explanation ##In summary, this code computes the probability of loan acceptance given that a customer has a bank credit card and is actively using online banking services, based on counts from a pivot table. ##The probability using the pivot table is 0.1005587 and the probability using naive bayes is NaN the probability using pivot table is more accurate.

##Question G ## Which of the entries in this table are needed for computing P(Loan = 1 | CC = 1, Online = 1)? Run naive Bayes on the data. Examine the model output on training data, and find the entry that corresponds to P(Loan = 1 | CC = 1, Online = 1). Compare this to the number you obtained in (E).

```r
count_loan_1_cc_1_online_1 <- pivot_table_loan_cc_online[2, 2, 2]
count_cc_1_online_1 <- sum(pivot_table_loan_cc_online[, 2, 2])

p_loan_1_given_cc_online_pivot <- count_loan_1_cc_1_online_1 / count_cc_1_online_1

# Output the result
cat("P(Loan = 1 | CC = 1, Online = 1) from pivot table:", p_loan_1_given_cc_online_pivot, "\n")
```

```
## P(Loan = 1 | CC = 1, Online = 1) from pivot table: 0.1005587
```

```r
#Fit naive Bayes model
naive_bayes_model <- naiveBayes(Personal.Loan ~ CreditCard + Online, data = trainset.norm)

# print the model output
print(naive_bayes_model)
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##     0     1
## 0.904 0.096
##
## Conditional probabilities:
##    CreditCard
## Y           [,1]      [,2]
##   0 -0.01005633 0.9954884
##   1  0.09469710 1.0385427
##
##    Online
## Y            [,1]      [,2]
##   0 -0.002446619 1.0004951
##   1  0.023038995 0.9967648
```

```r
#Find the probability P(Loan = 1 | CC = 1, Online = 1) from the model output
p_loan_1_given_cc_online_model <- naive_bayes_model$table$Personal.Loan[2, "Yes"]

# Output the result
cat("P(Loan = 1 | CC = 1, Online = 1) from naive Bayes model:", p_loan_1_given_cc_online_model, "\n")
```

```
## P(Loan = 1 | CC = 1, Online = 1) from naive Bayes model:
```

##Explanation #This code calculates the probability of loan acceptance given that a customer has a bank credit card and is actively using online banking services using both a pivot table approach and a Naive Bayes model, providing the results from both methods. ##When we compare the solution with value obtained in "Question E".The solution obtained in Question G is more accurate than value obtained in the Question E.