Assignment 03

Jovan Zivak

2025-10-09

```r
# Import the (really big) UniversalBank dataset
library(readxl)
UniversalBank <- read_excel("C:/Users/jovan/Downloads/UniversalBank.xlsx")
#View(UniversalBank)            COMMENTED OUT FOR CLARITY
```

```r
# Filter down to the three columns being used:
# Online, CreditCard, PersonalLoan
MyData <- UniversalBank[, c(10, 13, 14)]
set.seed(123)

# Partition the data
Index_Train<-createDataPartition(MyData$`Personal Loan`, p=0.6, list=FALSE)
Train<-MyData[Index_Train, ]
Test<-MyData[-Index_Train, ]
```

```r
#PART A
# Build a Pivot Table using Online as a column, Credit Card as a row,
# Personal Loan as a secondary row
Pivot_A <- table(Train$CreditCard, Train$`Personal Loan`, Train$Online)

# "Melt" data to flatten
melted_A <- melt(Pivot_A, value.name = "Count")
names(melted_A) <- c("CreditCard", "Personal Loan", "Online", "Count")

# "Cast" data: rows = CreditCard + Loan, columns = Online
Pivot_A <- dcast(melted_A, CreditCard + `Personal Loan` ~ Online,
                                                value.var = "Count")

# Rename columns so they make actual sense
names(Pivot_A)[3:4] <- c("Online_0", "Online_1")

# Print result
Pivot_A
```

```
##   CreditCard Personal Loan Online_0 Online_1
## 1          0             0      785     1145
## 2          0             1       65      122
## 3          1             0      317      475
## 4          1             1       34       57
```

```r
#PART B
# A customer who is both using a credit card and online accepting a loan
cat("The likelihood of this customer accepting a loan is 34/91,\n",
       "or approximately 37.3%")
```

```
## The likelihood of this customer accepting a loan is 34/91,
##  or approximately 37.3%
```

```r
#PART C (Table B Overall)
# Build Pivot Table B using Personal Loan as a column and Online as a row
Pivot_B <- table(Train$`Personal Loan`, Train$Online)

# "Melt" data to flatten
melted_B <- melt(Pivot_B, value.name = "Count")
names(melted_B) <- c("Personal Loan", "Online", "Count")

# "Cast" data: rows = Online, columns = Personal Loan
Pivot_B <- dcast(melted_B, Online ~ `Personal Loan`, value.var = "Count")

# Rename columns so they make actual sense
names(Pivot_B)[2:3] <- c("Decline Loan", "Accept Loan")

# Print result
Pivot_B
```

```
##   Online Decline Loan Accept Loan
## 1      0         1102          99
## 2      1         1620         179
```

```r
#PART C (Pivot Table C Overall)
# Build Pivot Table B using Personal Loan as a row and Credit Card as a column
Pivot_C <- table(Train$CreditCard, Train$`Personal Loan`)

# "Melt" data to flatten
melted_C <- melt(Pivot_C, value.name = "Count")
names(melted_C) <- c("Credit Card", "Personal Loan", "Count")

# "Cast" data: rows = Personal Loan, columns = Credit Card
Pivot_C <- dcast(melted_C, `Personal Loan` ~ `Credit Card`,
                 value.var = "Count")

# Rename columns so they make actual sense
names(Pivot_C)[2:3] <- c("No Card", "With Card")

# Print result
Pivot_C
```

```
##   Personal Loan No Card With Card
## 1             0    1930       792
## 2             1     187        91
```

```
## Part D: Echo disabled for ease of reading.
```

```
## i.   The proportion of cardholders who accepted a loan is 91/883,
##  or 10.3% of loan accepters.

## ii.  The proportion of online users who accepted a loan is 179/1799,
##  or 9.9% of loan accepters.

## iii. The proportion of loan accepters is 278/3000,
##  or 9.3% of bank customers.

## iv.  The proportion of cardholders who did not accept a loan is 792/883,
##  or 89.7% of loan decliners.

## v.   The proportion of online users who declined a loan is 1620/1799,
##  or 89.1% of loan decliners.

## vi.  The proportion of loan decliners is 2722/3000,
##  or 90.7% of bank customers.
```

```r
#PART E
# Using the six proportions from Part D:
# i:   91/883
# ii:  179/1799
# iii: 278/3000
# iv:  792/883
# v:   1620/1799
# vi:  2722/3000

# Math out Naive-Bayes formulation
numerator <- (91/883) * (179/278) * (278/3000)
denominator <- numerator + (792/883) * (1620/2722) * (2722/3000)
Naive_Bayes <- numerator / denominator



# Print outcome and convert to percentage
Naive_Bayes * 100
```

```
## [1] 1.253647
```

```r
# Print it properly for viewing
cat(
  sprintf(
    "The naive Bayes probability P(Loan = 1 | CC = 1, Online = 1)\n
is approximately %.2f%%\n",
    Naive_Bayes * 100
  )
)
```

```
## The naive Bayes probability P(Loan = 1 | CC = 1, Online = 1)
##
## is approximately 1.25%
```

```r
#PART F

# Analysis
cat("In Part B, the likelihood that a user simultaneously accepts a loan,\n",
       "is an online user, and is a cardholder is roughly 57/3000 or 1.9%,\n",
       "so given the Pivot Table's use of actual data, it is likely a more\n",
       "accurate estimate.\n")
```

```
## In Part B, the likelihood that a user simultaneously accepts a loan,
##  is an online user, and is a cardholder is roughly 57/3000 or 1.9%,
##  so given the Pivot Table's use of actual data, it is likely a more
##  accurate estimate.
```

```r
# PART G
# Which entries are needed?
cat("Entries needed: P(CC=1 | Loan=1), P(Online=1 | Loan=1),\n",
       "P(Loan=1), P(CC=1 | Loan=0), P(Online=1 | Loan=0), P(Loan=0).\n")
```

```
## Entries needed: P(CC=1 | Loan=1), P(Online=1 | Loan=1),
##  P(Loan=1), P(CC=1 | Loan=0), P(Online=1 | Loan=0), P(Loan=0).
```

```r
# 1) Train Naive Bayes on TRAIN
# build a naive Bayes classifier (TRAIN)
nb_model <- naiveBayes(`Personal Loan` ~ Online + CreditCard, data = Train)

# predict the loan status on TEST (class labels)
Predicted_Test_Labels <- predict(nb_model, Test)

# confusion matrix (TEST)
CrossTable(
  x = Test$`Personal Loan`,
  y = Predicted_Test_Labels,
  prop.chisq = FALSE
)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  2000
##
##
##                      | Predicted_Test_Labels
## Test$`Personal Loan` |         0 | Row Total |
## ---------------------|-----------|-----------|
##                    0 |      1798 |      1798 |
##                      |     0.899 |           |
```

```
## ---------------------|-----------|-----------|
##                    1 |       202 |       202 |
##                      |     0.101 |           |
## ---------------------|-----------|-----------|
##         Column Total |      2000 |      2000 |
## ---------------------|-----------|-----------|
##
##
```

```r
# Sanity check to confirm some weird numbers...
table(Predicted_Test_Labels)
```

```
## Predicted_Test_Labels
##    0    1
## 2000    0
```

```r
# Print conclusion
cat("This result is very close to the result from Part E (1.25% vs 1.01%).")
```

```
## This result is very close to the result from Part E (1.25% vs 1.01%).
```