Assignment 3

SINGAMALLA SAIRITHIK

10-15-2023

### Install and import libraries

# Load necessary libraries  
options(warn = -1) # Suppress all warnings  
if (!require("caret")) install.packages("caret")

## Loading required package: caret

## Loading required package: ggplot2

## Loading required package: lattice

if (!require("e1071")) install.packages("e1071")

## Loading required package: e1071

if (!require("reshape2")) install.packages("reshape2")

## Loading required package: reshape2

library(caret)  
library(e1071)  
library(reshape2)

### Load the dataset

# Load the data  
data <- read.csv("UniversalBank.csv")

### partition of data into training and validation sets

set.seed(123)  
index <- createDataPartition(data$Personal.Loan, p = 0.6, list = FALSE)  
train\_data <- data[index, ]  
val\_data <- data[-index, ]  
colnames(train\_data)

## [1] "ID" "Age" "Experience"   
## [4] "Income" "ZIP.Code" "Family"   
## [7] "CCAvg" "Education" "Mortgage"   
## [10] "Personal.Loan" "Securities.Account" "CD.Account"   
## [13] "Online" "CreditCard"

### Task A

# Melt the data, specifying 'variable.name' and 'value.name'  
data\_melted <- melt(train\_data, id.vars = c("Online", "CreditCard"), measure.vars = "Personal.Loan", variable.name = "Variable", value.name = "Personal\_Loan")  
  
# Cast the melted data into a wider format for the pivot table  
pivot <- dcast(data\_melted, CreditCard + Personal\_Loan ~ Online, fun.aggregate = length)

## Using Personal\_Loan as value column: use value.var to override.

print(pivot)

## CreditCard Personal\_Loan 0 1  
## 1 0 0 785 1145  
## 2 0 1 65 122  
## 3 1 0 317 475  
## 4 1 1 34 57

### Task B

# Probability calculation for part b  
subset\_data <- subset(train\_data, CreditCard == 1 & Online == 1)  
prob\_b <- nrow(subset(subset\_data, `Personal.Loan` == 1)) / nrow(subset\_data)  
  
print(prob\_b)

## [1] 0.1071429

### Task C

# Pivot tables for part c  
pivot\_loan\_online <- table(train\_data$`Personal.Loan`, train\_data$Online)  
pivot\_loan\_cc <- table(train\_data$`Personal.Loan`, train\_data$CreditCard)  
  
print(pivot\_loan\_online)

##   
## 0 1  
## 0 1102 1620  
## 1 99 179

print(pivot\_loan\_cc)

##   
## 0 1  
## 0 1930 792  
## 1 187 91

### Task D

# Probabilities for part d  
P\_CC\_given\_loan\_1 <- pivot\_loan\_cc[2,2] / sum(pivot\_loan\_cc[2,])  
P\_online\_given\_loan\_1 <- pivot\_loan\_online[2,2] / sum(pivot\_loan\_online[2,])  
P\_loan\_1 <- sum(pivot\_loan\_cc[2,]) / sum(pivot\_loan\_cc)  
P\_CC\_given\_loan\_0 <- pivot\_loan\_cc[1,2] / sum(pivot\_loan\_cc[1,])  
P\_online\_given\_loan\_0 <- pivot\_loan\_online[1,2] / sum(pivot\_loan\_online[1,])  
P\_loan\_0 <- sum(pivot\_loan\_cc[1,]) / sum(pivot\_loan\_cc)

### Task E

# Naive Bayes probability for part e  
total <- nrow(train\_data)  
P\_CC\_1 <- sum(train\_data$CreditCard == 1) / total  
P\_online\_1 <- sum(train\_data$Online == 1) / total  
P\_loan\_1\_given\_CC\_1\_and\_online\_1 <- (P\_CC\_given\_loan\_1 \* P\_online\_given\_loan\_1 \* P\_loan\_1) / (P\_CC\_1 \* P\_online\_1)

### Task F

comparison <- data.frame(  
 Method = c("Pivot Table", "Naive Bayes"),  
 Probability = c(prob\_b, P\_loan\_1\_given\_CC\_1\_and\_online\_1)  
)  
  
print(comparison)

## Method Probability  
## 1 Pivot Table 0.1071429  
## 2 Naive Bayes 0.1106570

### Task G

# --- Task: Train a naive Bayes model and make a prediction for P(Loan = 1 | CC = 1, Online = 1) ---  
model <- naiveBayes(as.factor(`Personal.Loan`) ~ Online + CreditCard, data = train\_data)  
newdata <- data.frame(Online = 1, CreditCard = 1)  
pred <- predict(model, newdata, type = "raw")  
prob\_from\_model <- pred[1, "1"] # "1" is the factor level for loan acceptance  
print(prob\_from\_model) # This prints the probability from the naive Bayes model

## 1   
## 0.1156935