#— #title: “Assignment3” #output:html_document

#—

setting up the working Directory

#Importing Data set #changing to factors

```
unbank_main <- read.csv("UniversalBank (1).csv")

unbank_main$Personal.loan<-as.factor(unbank_main$Personal.Loan)
unbank_main$Creditcard<-as.factor(unbank_main$CreditCard)
unbank_main$Online<-as.factor(unbank_main$Online)

library(caret)
```

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

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

```
library(ggplot2)
library(lattice)
library(e1071)
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
```

```
library(tidyr)
library(ISLR)
library(FNN)

# splitting the data

set.seed(20)
Index<-createDataPartition(unbank_main$Income,p=0.6,list=FALSE)
train_data<-unbank_main[Index,]
dim(train_data)
```

```
## [1] 3002   16
```

```
valid_data<-unbank_main [-Index ,]
dim(valid_data)
```

```
## [1] 1998   16
```

```
summary(train_data)
```

```
##        ID            Age          Experience       Income         ZIP.Code
##  Min.   :   3   Min.   :23.00   Min.   :-3.00   Min.   :  8.0   Min.   : 9307
##  1st Qu.:1246   1st Qu.:35.00   1st Qu.:10.00   1st Qu.: 39.0   1st Qu.:91911
##  Median :2498   Median :45.00   Median :20.00   Median : 64.0   Median :93437
##  Mean   :2498   Mean   :45.21   Mean   :19.99   Mean   : 74.2   Mean   :93144
##  3rd Qu.:3738   3rd Qu.:55.00   3rd Qu.:30.00   3rd Qu.: 98.0   3rd Qu.:94608
##  Max.   :4999   Max.   :67.00   Max.   :43.00   Max.   :218.0   Max.   :96651
##      Family          CCAvg          Education        Mortgage
##  Min.   :1.000   Min.   : 0.000   Min.   :1.000   Min.   :  0.00
##  1st Qu.:1.000   1st Qu.: 0.700   1st Qu.:1.000   1st Qu.:  0.00
##  Median :2.000   Median : 1.500   Median :2.000   Median :  0.00
##  Mean   :2.402   Mean   : 1.945   Mean   :1.892   Mean   : 58.55
##  3rd Qu.:3.000   3rd Qu.: 2.500   3rd Qu.:3.000   3rd Qu.:103.00
##  Max.   :4.000   Max.   :10.000   Max.   :3.000   Max.   :635.00
##  Personal.Loan     Securities.Account   CD.Account      Online
##  Min.   :0.00000   Min.   :0.0000     Min.   :0.00000   0:1219
##  1st Qu.:0.00000   1st Qu.:0.0000     1st Qu.:0.00000   1:1783
##  Median :0.00000   Median :0.0000     Median :0.00000
##  Mean   :0.09793   Mean   :0.1069     Mean   :0.06296
##  3rd Qu.:0.00000   3rd Qu.:0.0000     3rd Qu.:0.00000
##  Max.   :1.00000   Max.   :1.0000     Max.   :1.00000
##    CreditCard     Personal.loan Creditcard
##  Min.   :0.0000   0:2708        0:2129
##  1st Qu.:0.0000   1: 294        1: 873
##  Median :0.0000
##  Mean   :0.2908
##  3rd Qu.:1.0000
##  Max.   :1.0000
```

```
summary(valid_data)
```

```
##        ID            Age          Experience        Income      
##  Min.   :   1   Min.   :23.00   Min.   :-3.00   Min.   :  8.00  
##  1st Qu.:1263   1st Qu.:36.00   1st Qu.:11.00   1st Qu.: 39.00  
##  Median :2506   Median :46.00   Median :20.00   Median : 63.50  
##  Mean   :2504   Mean   :45.53   Mean   :20.27   Mean   : 73.13  
##  3rd Qu.:3768   3rd Qu.:56.00   3rd Qu.:30.00   3rd Qu.: 98.00  
##  Max.   :5000   Max.   :67.00   Max.   :43.00   Max.   :224.00  
##     ZIP.Code         Family          CCAvg           Education    
##  Min.   :90005   Min.   :1.000   Min.   : 0.000   Min.   :1.000  
##  1st Qu.:91911   1st Qu.:1.000   1st Qu.: 0.700   1st Qu.:1.000  
##  Median :93422   Median :2.000   Median : 1.600   Median :2.000  
##  Mean   :93166   Mean   :2.388   Mean   : 1.928   Mean   :1.865  
##  3rd Qu.:94608   3rd Qu.:3.000   3rd Qu.: 2.600   3rd Qu.:3.000  
##  Max.   :96651   Max.   :4.000   Max.   :10.000   Max.   :3.000  
##     Mortgage      Personal.Loan     Securities.Account   CD.Account     
##  Min.   :  0.00   Min.   :0.00000   Min.   :0.0000     Min.   :0.00000  
##  1st Qu.:  0.00   1st Qu.:0.00000   1st Qu.:0.0000     1st Qu.:0.00000  
##  Median :  0.00   Median :0.00000   Median :0.0000     Median :0.00000  
##  Mean   : 53.42   Mean   :0.09309   Mean   :0.1006     Mean   :0.05656  
##  3rd Qu.: 97.00   3rd Qu.:0.00000   3rd Qu.:0.0000     3rd Qu.:0.00000  
##  Max.   :601.00   Max.   :1.00000   Max.   :1.0000     Max.   :1.00000  
##  Online     CreditCard     Personal.loan Creditcard
##  0: 797   Min.   :0.0000   0:1812        0:1401    
##  1:1201   1st Qu.:0.0000   1: 186        1: 597    
##           Median :0.0000                           
##           Mean   :0.2988                           
##           3rd Qu.:1.0000                           
##           Max.   :1.0000                           
```

problem 1-Create a pivot table for the training data with Online as a column variable, CC as a row variable, 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 panda dataframe methods melt()and pivot().

```r
library(reshape2)
```

```
##
## Attaching package: 'reshape2'
```

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

```r
T_melt<-melt(train_data ,id=c("CreditCard","Personal.Loan"), measure.variable="Online
")
```

```
## Warning: attributes are not identical across measure variables; they will be
## dropped
```

```r
T_cast<-dcast(T_melt,CreditCard+ Personal.Loan ~ variable)
```

```
## Aggregation function missing: defaulting to length
```

```r
T_cast[,c(1:2,14)]
```

```
##   CreditCard Personal.Loan Online
## 1          0             0   1919
## 2          0             1    210
## 3          1             0    789
## 4          1             1     84
```

problem 2 -Consider the task of classifying a customer who owns a bank credit card and is actively using online banking services. Looking at the pivot table, what is the probability that this customer will accept the loan offer? [This is the

probability of loan acceptance (Loan = 1) conditional on having a bank credit card (CC = 1) and being an active user of online banking services (Online = 1)].

```
a <- table(train_data[,c(10,13,14)])
b <- as.data.frame(a)
b
```

```
##   Personal.Loan Online CreditCard Freq
## 1             0      0          0  788
## 2             1      0          0   82
## 3             0      1          0 1131
## 4             1      1          0  128
## 5             0      0          1  316
## 6             1      0          1   33
## 7             0      1          1  473
## 8             1      1          1   51
```

#Answer=82/(82+788)=0.094

0.094 is the probability of a customer who has a bank CC and actively uses online banking services, as per the pivot table created in the steps above.

#problem 3 -Create two separate pivot tables for the training data. Onewill have Loan (rows) as a function of Online (columns) and the other will have Loan (rows) as a function of CC.

```
library(reshape2)
library(ggplot2)
T_melt1<-melt(train_data,id=c("Personal.Loan"),variable ="Online")
```

```
## Warning: attributes are not identical across measure variables; they will be
## dropped
```

```
T_melt2<-melt(train_data,id=c("CreditCard"), variable = "Online")
```

```
## Warning: attributes are not identical across measure variables; they will be
## dropped
```

```
T_cast1<-dcast(T_melt1,Personal.Loan~Online)
```

```
## Aggregation function missing: defaulting to length
```

```
T_cast2<-dcast(T_melt2, CreditCard~Online)
```

```
## Aggregation function missing: defaulting to length
```

```
LOnline <- T_cast1[,c(1,13)]
LCC <- T_cast2[,c(1,14)]
LOnline
```

```
##   Personal.Loan Online
## 1             0   2708
## 2             1    294
```

```
LCC
```

```
##   CreditCard Online
## 1          0   2129
## 2          1    873
```

problem 4- 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)

```
table(train_data[,c(14,10)])
```

```
##           Personal.Loan
## CreditCard    0    1
##          0 1919  210
##          1  789   84
```

Answer= 84/(84+210)=0.2857

#2. #p(online=1| Loan = 1)

```
table(train_data[,c(13,10)])
```

```
##       Personal.Loan
## Online    0    1
##      0 1104  115
##      1 1604  179
```

Answer = 179/(179+115)=0.6088

#3 p(Loan=1) (The proportion of loan acceptors)

```
table(train_data[,c(10)])
```

```
##
##    0    1
## 2708  294
```

Answer= 294/(2708+294)= 0.097

#4 #P(CC=1 | Loan = 0)

```
table(train_data[c(10,14)])
```

```
##                CreditCard
## Personal.Loan    0    1
##             0 1919  789
##             1  210   84
```

Answer = 789/(1919+789)=0.2913

5

#P(Online = 1 | Loan =0)

```
table(train_data[c(10,13)])
```

```
##                Online
## Personal.Loan    0    1
##             0 1104 1604
##             1  115  179
```

Answer = 1604/(1604+1104)=0.5923

#6 p(Loan=0)

```
table(train_data[,10])
```

```
##
##    0    1
## 2708  294
```

Answer = 2708/(2708+294)=0.902

#problem 5 -Use the quantities computed above to compute the naive Bayes probability P(Loan = 1 | CC = 1, Online = 1).

Naive Bayes Probability =

#P (Loan =1 | CC =1 , Online =1) = P (CC=1 | Loan = 1) * P (Loan =1)/ [(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))] # = 0.2857*0.6088*0.097/(0.2857*0.6088*0.097)+(0.2913*0.5923*0.902) # =0.09743

problem 6-Compare this value with the one obtained from the pivot table in (B). Which is a more accurate estimate?

Answer= The value from the pivot table is 0.094 and the value computed from Naive Bayes probability is 0.097 we can see here the different is significant. The difference is beacuse of the assumption of conditional Independene in the Naive Bayes formula.For a smaller dataset, the exact values are easy to be calculated.But for bigger chunks of data Naive Bayes probability will be preferred based on the insignificant differnce in the probabilities from the pivot and Naive Bayes formula.

problem 7 -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
library(e1071)

Naivebayesmodel<-naiveBayes(Personal.loan~.,train_data)
Naivebayesmodel
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##          0          1
## 0.90206529 0.09793471
##
## Conditional probabilities:
##    ID
## Y       [,1]     [,2]
##   0 2512.766 1450.491
##   1 2364.071 1424.162
##
##    Age
## Y       [,1]     [,2]
##   0 45.27585 11.42992
##   1 44.63265 11.69060
##
##    Experience
## Y       [,1]     [,2]
##   0 20.05355 11.45498
##   1 19.41837 11.69807
##
##    Income
## Y        [,1]     [,2]
##   0  66.25443 40.36059
##   1 147.39796 32.97110
##
##    ZIP.Code
## Y       [,1]     [,2]
##   0 93150.36 2383.999
##   1 93082.15 1722.333
##
##    Family
## Y       [,1]     [,2]
##   0 2.375554 1.145341
##   1 2.646259 1.113386
```

```
##
##    CCAvg
## Y       [,1]     [,2]
##   0 1.719730 1.579905
##   1 4.016973 2.124193
##
##    Education
## Y       [,1]      [,2]
##   0 1.854505 0.8396956
##   1 2.234694 0.7635156
##
##    Mortgage
## Y        [,1]      [,2]
##   0  53.29136  93.96906
##   1 106.93878 165.26445
##
##    Personal.Loan
## Y   [,1] [,2]
##   0    0    0
##   1    1    0
##
##    Securities.Account
## Y        [,1]      [,2]
##   0 0.1045052 0.3059713
##   1 0.1292517 0.3360503
##
##    CD.Account
## Y        [,1]      [,2]
##   0 0.0372969 0.1895234
##   1 0.2993197 0.4587409
##
##    Online
## Y           0         1
##   0 0.4076809 0.5923191
##   1 0.3911565 0.6088435
##
##    CreditCard
## Y        [,1]      [,2]
##   0 0.2913589 0.4544724
##   1 0.2857143 0.4525242
##
##    Creditcard
## Y           0         1
##   0 0.7086411 0.2913589
##   1 0.7142857 0.2857143
```

```r
pred_Test<-predict(Naivebayesmodel,valid_data)

library(gmodels)
# Confusion Matrix of the Naive bayes Model

CrossTable(valid_data$Personal.Loan,pred_Test,prop.chisq=FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  1998
##
##
##                          | pred_Test
## valid_data$Personal.Loan |         0 |         1 | Row Total |
## -------------------------|-----------|-----------|-----------|
##                        0 |      1810 |         2 |      1812 |
##                          |     0.999 |     0.001 |     0.907 |
##                          |     1.000 |     0.011 |           |
##                          |     0.906 |     0.001 |           |
## -------------------------|-----------|-----------|-----------|
##                        1 |         0 |       186 |       186 |
##                          |     0.000 |     1.000 |     0.093 |
##                          |     0.000 |     0.989 |           |
##                          |     0.000 |     0.093 |           |
## -------------------------|-----------|-----------|-----------|
##             Column Total |      1810 |       188 |      1998 |
##                          |     0.906 |     0.094 |           |
## -------------------------|-----------|-----------|-----------|
##
##
```