Big Data Analytics Individual Assignment 2

After completing a variety of analyses on the dataset provided, it is concluded that whether a customer is an active user of online banking services (“Online”), or if the customer has a credit card issued by the bank (“Credit Card”) are not correlated to a customer’s response to the personal loan campaign (“PersonalLoan”).

With the goal of this to test the validity of using these variables to predict what would be a customer’s response to a personal loan campaign, I wanted to use a variety of tools to help analyze the correlation. When looking at the dataset provided, I first partitioned the data into training and validation data (60% and 40% respectively). I then ran the Naïve Bayes algorithm on the two variables against the PersonalLoan response variable (Exhibit 1).

Exhibit 1: Naïve Bayes

```
nb.out <- naiveBayes(PersonalLoan ~ Online + CreditCard, data = traindf)
nb.out
```

```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = X, y = Y, laplace = laplace)

A-priori probabilities:
Y
         0          1 
0.90633333 0.09366667 

Conditional probabilities:
   Online
Y        [,1]      [,2]
  0 0.5847738 0.4928516
  1 0.6120996 0.4881410

   CreditCard
Y        [,1]      [,2]
  0 0.2872380 0.4525568
  1 0.3096085 0.4631571
```

I then completed confusion matrices where, based on the data provided, we see a high but equal accuracy vs. no information rate, meaning that the model is not better at predicting PersonalLoan customer response (Exhibit 2). As well, for the training data, we see a sensitivity of 1 and a specificity of 0. This means that all actual cases were correctly predicted as positive, and all actual non-cases were incorrectly predicted as positive. Therefore, the matrix shows that everything was predicted to be positive, which means that the model is actually predicting nothing. This is the exact opposite case for the other matrix.

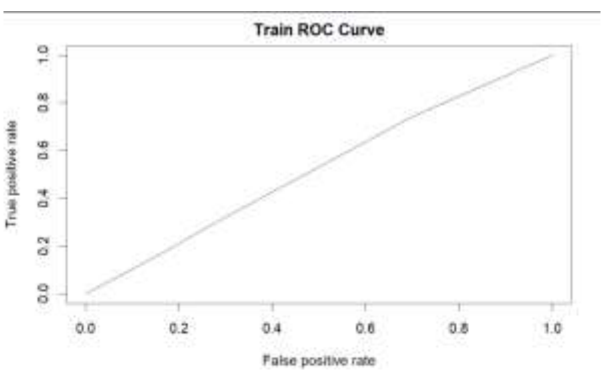
Exhibits 2 and 3: Confusion Matrices

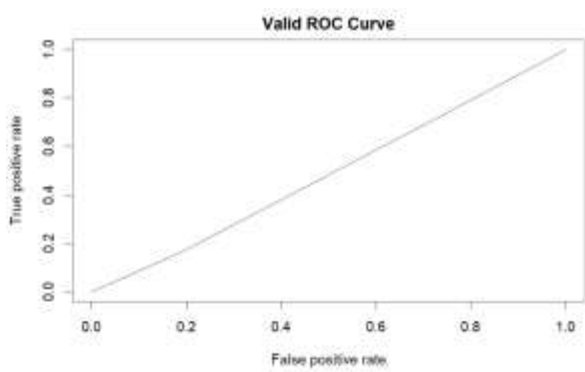
```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 1801  199
         1    0    0

               Accuracy : 0.9005
                 95% CI : (0.8865, 0.9133)
    No Information Rate : 0.9005
    P-Value [Acc > NIR] : 0.5189

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 1.0000
            Specificity : 0.0000
         Pos Pred Value : 0.9005
         Neg Pred Value :    NaN
             Prevalence : 0.9005
         Detection Rate : 0.9005
   Detection Prevalence : 1.0000
      Balanced Accuracy : 0.5000

       'Positive' Class : 0
```

```
Confusion Matrix and Statistics

          Reference
Prediction    0    1
         0 2719  281
         1    0    0

               Accuracy : 0.9063
                 95% CI : (0.8953, 0.9165)
    No Information Rate : 0.9063
    P-Value [Acc > NIR] : 0.5159

                  Kappa : 0

 Mcnemar's Test P-Value : <2e-16

            Sensitivity : 0.00000
            Specificity : 1.00000
         Pos Pred Value :     NaN
         Neg Pred Value : 0.90633
             Prevalence : 0.09367
         Detection Rate : 0.00000
   Detection Prevalence : 0.00000
      Balanced Accuracy : 0.50000

       'Positive' Class : 1
```

I then wanted to analyze the ROC curves for the train and validation data and they confirmed my findings about the sensitivity and specificity scores from the confusion matrices, where it is a direct linear relationship (Exhibits 4 and 5).

Exhibits 4 and 5: ROC curve.





In conclusion, based on the data provided, neither of these variables are indicative of the response variable. To further test this, I would want to analyze more instances where PersonalLoan = 1, as in this dataset, that was only approx. 10% of the data provided, which could be a small enough sample size that when looking at a larger sample size the results could be altered.

Code:

````
```{r}
bankdata<-read.csv("Assignment2Data.csv", header=TRUE)
bankdata

install.packages(ROCR)
library(ROCR)
library(caret)
```
```{r}
trainindex<-sample(c(1:dim(bankdata)[1]), dim(bankdata)[1]*0.6)
traindf<-bankdata[trainindex, ]
validdf<-bankdata [-trainindex, ]
```

```{r}
install.packages("caret")
install.packages("neuralnet")
install.packages("forecast")
install.packages("gains")
library(caret)
library(neuralnet)
library(forecast)
library(gains)
library(e1071)

nb.out<-naiveBayes(PersonalLoan ~ Online + CreditCard, family = "binomial", data = traindf)
pred.prob<-predict(nb.out,traindf,type="raw")
pred.class<-predict(nb.out,traindf)
cbind(traindf,pred.class,pred.prob)
nb.out


nb.out <- naiveBayes(PersonalLoan ~ Online + CreditCard, data = traindf)
nb.out
```

```{r}
fitprob <-predict(traindata, type = "response")
predictions.train1<-ifelse(fitprob>0.3,1,0)
predicted<-factor(predictions.train1)
actual<-factor(traindf$PersonalLoan)
confusionm <-confusionMatrix(data=predicted, reference=actual, positive='1')
confusionm
```

```{r}
fitprob1 <- factor (predict(nb.out, validdf[ , names(validdf) != "PersonalLoan"]))
validdf$PersonalLoan <- factor (validdf$PersonalLoan)
confusionm1 <- confusionMatrix(nbpred, validdf$PersonalLoan)
confusionm1
```
````


````
```{r}
fitprob1 <- factor (predict(nb.out, validdf[ , names(validdf) != "PersonalLoan"]))
validdf$PersonalLoan <- factor (validdf$PersonalLoan)
confusionm1 <- confusionMatrix(nbpred, validdf$PersonalLoan)
confusionm1
```
````


````
```{r}
library(ROCR)
library(caret)
library(e1071)
library(gains)

trainp <- predict(nb.out, traindf, type = "raw")[,2]
validp <- predict(nb.out, validdf, type = "raw")[,2]
bpredicto <- prediction(trainp, traindf$PersonalLoan)
rocb <- performance(bpredicto, measure = "tpr", x.measure = "fpr")
plot(rocb, main="Train ROC Curve")
bvalido <- prediction(validp, validdf$PersonalLoan)
rocv <- performance(bvalido, measure = "tpr", x.measure = "fpr")
plot(rocv, main = "Valid ROC Curve")
```
````