ASA_Assignment_1

Shreyas Namjoshi

23/03/2022

R Markdown

```
Loading the data and dividing it in Train and Test
```

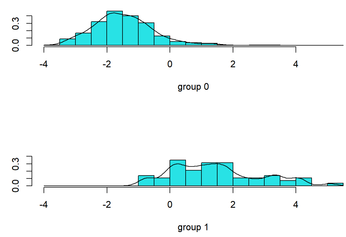
```
data<-read_excel("FinalDataSet_ASA.xlsx")
## 70% of the data for training
train_size <- floor(0.70 * nrow(data))

## set the seed to make your partition reproducible
set.seed(213)
train_ind <- sample(seq_len(nrow(data)),
                    size = train_size)
train <- data[train_ind,]
test <- data[-train_ind,]
```

Q1. Build a Discriminant Analysis Model to predict whether the person is likely to accept the bank's offer for a personal loan. If necessary, create new variables to improve the model performance.

```
lda.fit<-lda(data$`Personal Loan`~.,data=data,subset = train_ind)
lda.fit
```

```
## Call:
## lda(data$`Personal Loan` ~ ., data = data, subset = train_ind)
##
## Prior probabilities of groups:
##         0         1
## 0.91857143 0.08142857
##
## Group means:
##       Age Experience    Income `ZIP Code`    Family     CCAvg Education
## 0 45.28305   20.00467  68.32193   93054.84 2.343701 1.657092  1.819596
## 1 45.73684   20.70175 145.42105   93008.44 2.578947 3.834211  2.157895
##   Mortgage `Securities Account` `CD Account`   Online CreditCard
## 0 59.88554           0.09642382   0.03421462 0.5972086 0.3119420
## 1 110.63158          0.15789474   0.38596491 0.6842105 0.2807018
##
## Coefficients of linear discriminants:
##                              LD1
## Age                 -3.414638e-02
## Experience           3.676351e-02
## Income               1.707491e-02
## `ZIP Code`          -3.353735e-05
## Family               1.885591e-01
## CCAvg                1.550846e-01
## Education            5.604256e-01
## Mortgage             7.169486e-04
## `Securities Account` -3.338199e-01
## `CD Account`         2.931902e+00
## Online              -1.589666e-01
## CreditCard          -3.469410e-01
```

```
plot(lda.fit, type='b')
```




```
  Here from the plot we can see that there is clear difference between 2 classes (availed and not availed) with v
ery less overlap or misclassification between the two classes.
```

Q2. Carry out significance tests using Wilk's Lambda.

```
  Assuming alpha=0.1
```

```
dependent1=train$`Personal Loan`
formulaAll= dependent1 ~ train$Age+train$Experience+train$Income+train$`ZIP Code`+train$Family+train$CCAvg+train
$Education+train$Mortgage+train$`Securities Account`+train$`CD Account`+train$Online+train$CreditCard
print(formulaAll)
```

```
## dependent1 ~ train$Age + train$Experience + train$Income + train$`ZIP Code` +
##     train$Family + train$CCAvg + train$Education + train$Mortgage +
##     train$`Securities Account` + train$`CD Account` + train$Online +
##     train$CreditCard
```

```
greedy.wilks(formulaAll,data=train,niveau = 0.1)
```

```
## Formula containing included variables:
##
## dependent1 ~ train$Income + train$`CD Account` + train$Education +
##     train$Family + train$CCAvg + train$CreditCard
## <environment: 0x000000002455ad98>
##
##
## Values calculated in each step of the selection procedure:
##
##                vars Wilks.lambda F.statistics.overall p.value.overall
## 1       train$Income    0.7788474             198.19605    8.360639e-40
## 2 train$`CD Account`    0.6852210             160.09501    6.123437e-58
## 3    train$Education    0.6374653             131.94138    1.135936e-67
## 4       train$Family    0.6271354             103.30339    4.999329e-69
## 5        train$CCAvg    0.6177953              85.86989    3.081335e-70
## 6   train$CreditCard    0.6119585              73.23830    1.156063e-70
##   F.statistics.diff p.value.diff
## 1         198.19604 8.360639e-40
## 2          95.23584 0.000000e+00
## 3          52.14088 1.355693e-12
## 4          11.44768 7.559380e-04
## 5          10.49258 1.255816e-03
## 6           6.60978 1.034924e-02
```

Q3. Comment on the variables that are significant

```
 Comment:From the above table, the variables that are significant with cut off= 0.1 are Income,CD Account,Educati
on,Family,CCAvg and Credit Card.SO even if we drop rest of the variables from our model, our prediction is not go
ing to change much.
```

4. Create the confusion matrix and comment on the prediction accuracy.

```
 Here we get the prediction accuracy for test set(300 Records)
```

```
lda.pred <- predict(lda.fit, test)
results <- confusionMatrix(data=lda.pred$class, reference=as.factor(test$`Personal Loan`))
print(results)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0 261  19
##          1   5  15
##
##                Accuracy : 0.92
##                  95% CI : (0.8833, 0.9481)
##     No Information Rate : 0.8867
##     P-Value [Acc > NIR] : 0.037094
##
##                   Kappa : 0.5148
##
##  Mcnemar's Test P-Value : 0.007963
##
##             Sensitivity : 0.9812
##             Specificity : 0.4412
##          Pos Pred Value : 0.9321
##          Neg Pred Value : 0.7500
##              Prevalence : 0.8867
##          Detection Rate : 0.8700
##    Detection Prevalence : 0.9333
##       Balanced Accuracy : 0.7112
##
##        'Positive' Class : 0
##
```

```
 From the above output, it is seen that Prediction Accuracy is 92% with sensitivity as 98% and specificity as 4
4%.
```

5. The bank would like to address the top 30 persons with an offer for personal loan based on the probability (propensity). Create a table displaying all the details of the "top" 30 persons who are most likely to accept the bank's offer. Make sure to include the probability of accepting the offer along with all the other details.

```
  Here we are including only test set to find the top 30 person who are most likely to accept bank offer based on
probability
```

```
probdf<-lda.pred$posterior
test['Prob of Availing']<-probdf[,"1"]
test<-test[order(test$`Prob of Availing`,decreasing = TRUE),]
head(test,n=30)
```

```
## # A tibble: 30 x 14
##      Age Experience Income `ZIP Code` Family CCAvg Education Mortgage
##    <dbl>      <dbl>  <dbl>      <dbl>  <dbl> <dbl>     <dbl>    <dbl>
##  1    50         26    192      90245      2   1.8         3      301
##  2    41         15    159      90057      1   5.5         3        0
##  3    53         28    175      95060      3   3.6         3        0
##  4    43         19    174      92028      3   1.7         3      231
##  5    57         31    164      94607      2   3.8         3      422
##  6    47         23    170      90254      2   6.5         2        0
##  7    64         38    143      95039      2   6.4         3        0
##  8    56         32    173      94022      1   4.6         2       88
##  9    52         26    110      94501      2   5.4         3      204
## 10    45         15    202      91380      3  10           3        0
## # ... with 20 more rows, and 6 more variables: Personal Loan <dbl>,
## #   Securities Account <dbl>, CD Account <dbl>, Online <dbl>, CreditCard <dbl>,
## #   Prob of Availing <dbl>
```