LOAN DEFAULT PREDICTION ANALYSIS

Parag Garg
Cristina Segreda
Abhishek Subbarayalu

EXECUTIVE SUMMARY

- Our Model predicts the Loan defaults from the bank
- Our Initial Analysis established that data set primarily
- Loan Default was segmented based on Gender and Age Group

- ❖ Number of Female defaulters are greater than Males, but the rate of defaulting is higher in male population

- ❖ Age between 25-40 tend to be the maximum defaulters

- ❖ Educated from University have higher propensity to default loans in education category

VARIABLES USED TO DEVELOP THE MODELS

Variable Name	Description
Limit_Bal	Amount of the given credit (NT dollar) Including individual consumer & family credit
Sex	Binary description of Sex
Education	Level of Education Attained
Marriage	Marital Status
Age	Age in years
Pay_(0-6)	History of Past Monthly Payments
Bill_Amt (1-6)	Amount of each bill, correlated with Pay
Pay_Amt (1-6)	Amount of each payment, correlates with Pay

OUR DATA SOURCES

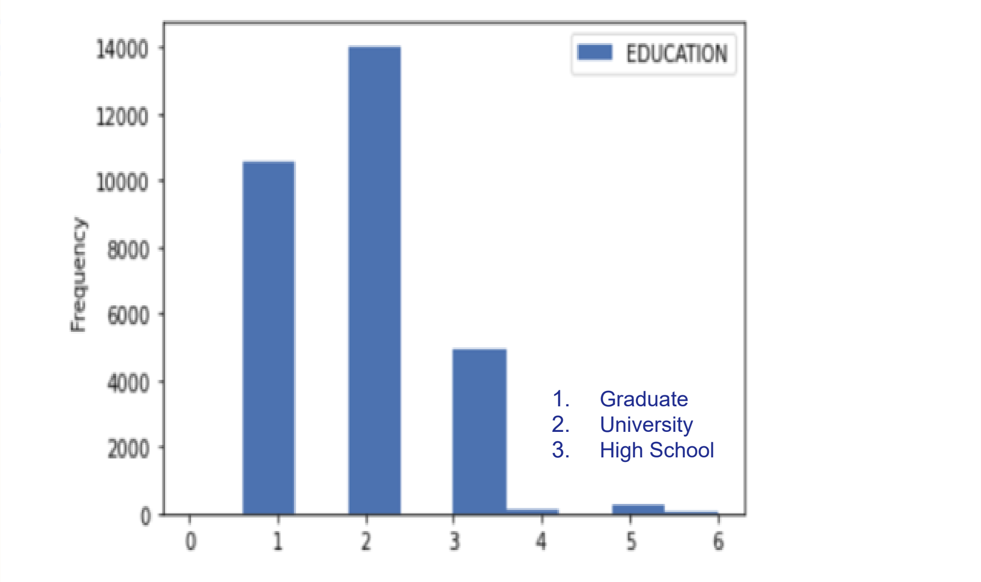
- 30,000 Customers
- Included 23 Variables
- Most Common Sex Sample is Female
- 4 Type of Marital Category

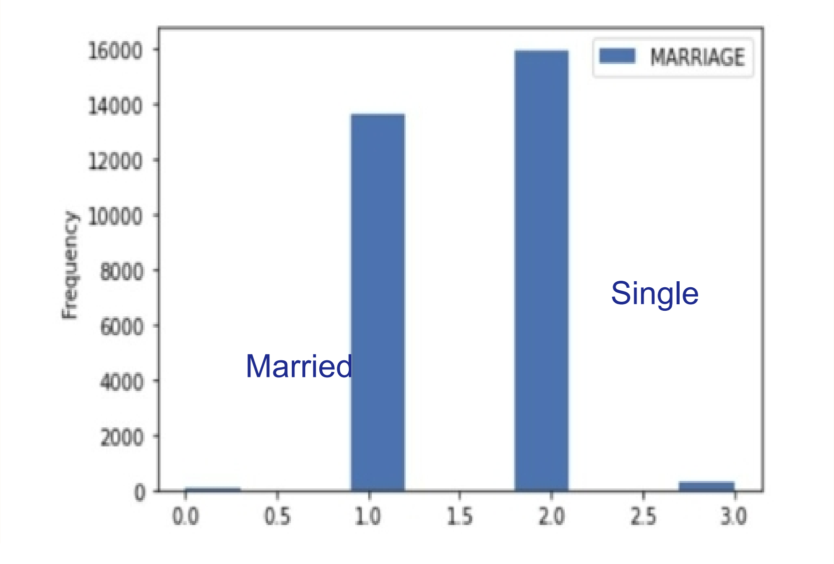
DATA SCRUBBING PROCESS

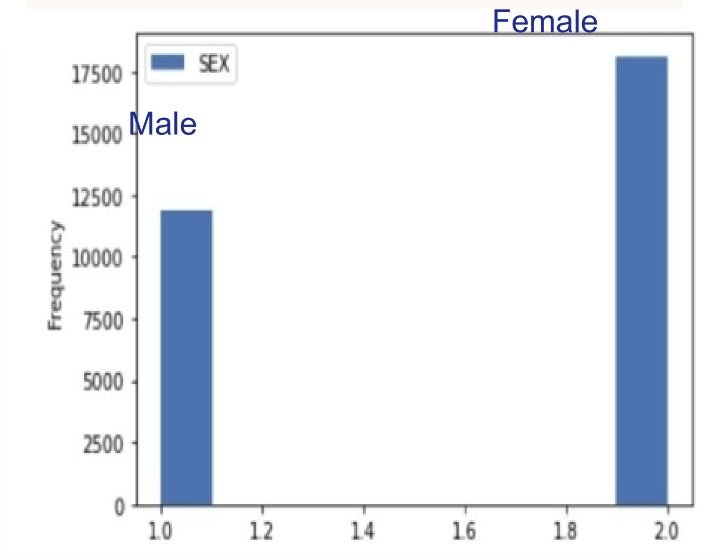
- Remove ID from Dataset
- Check the data type
- Check Missing Value in Numeric Variable or not.
- We did a mathematical Analysis of Numeric Variables
- Replace Missing Values
- Check Missing Values in Categorial Variables.

.

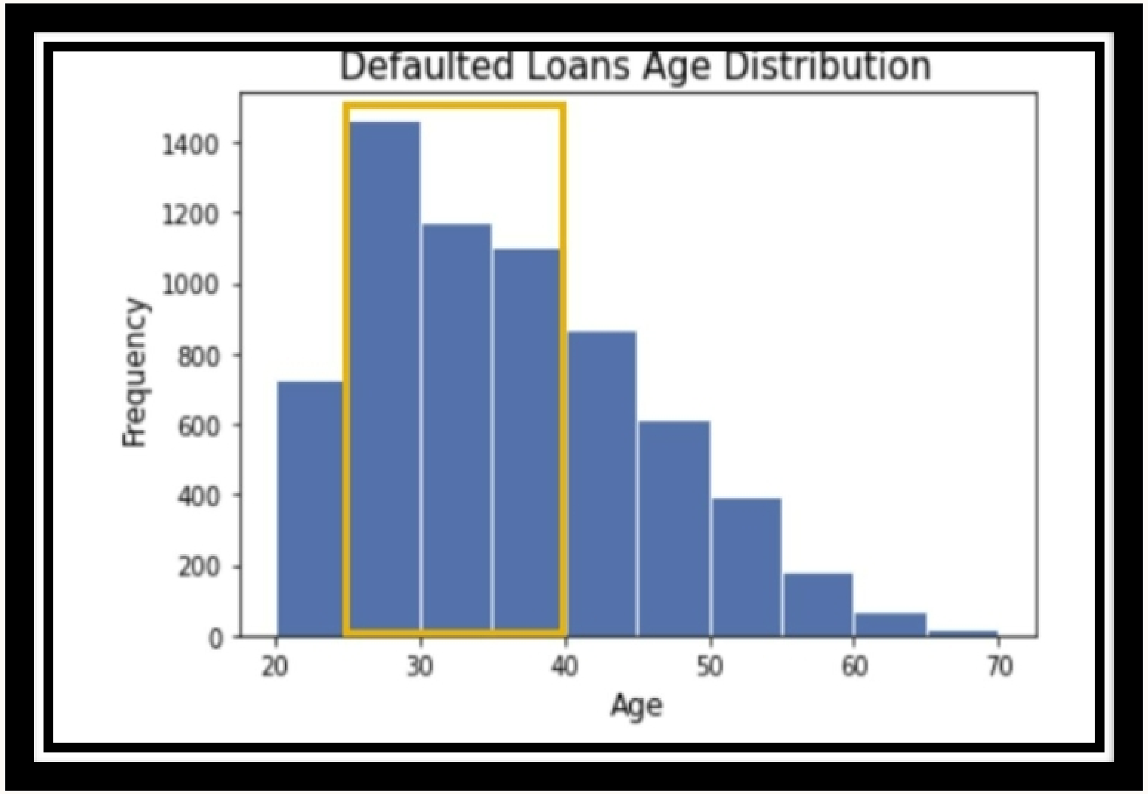
SEGMENTATION



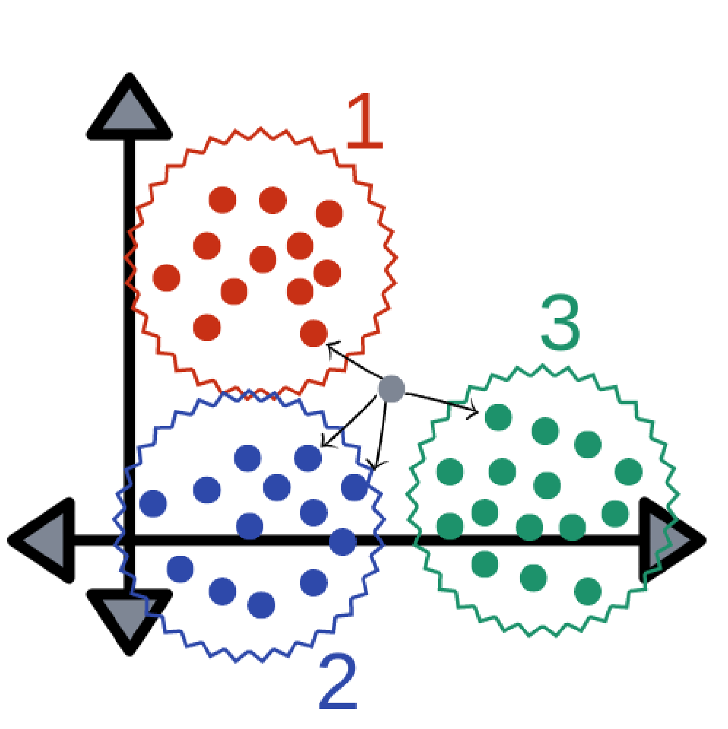




STATUS RELATED TO AGE



K NEAREST NEIGHBOR MODEL (KNN)



Assume similar things exists in close proximity

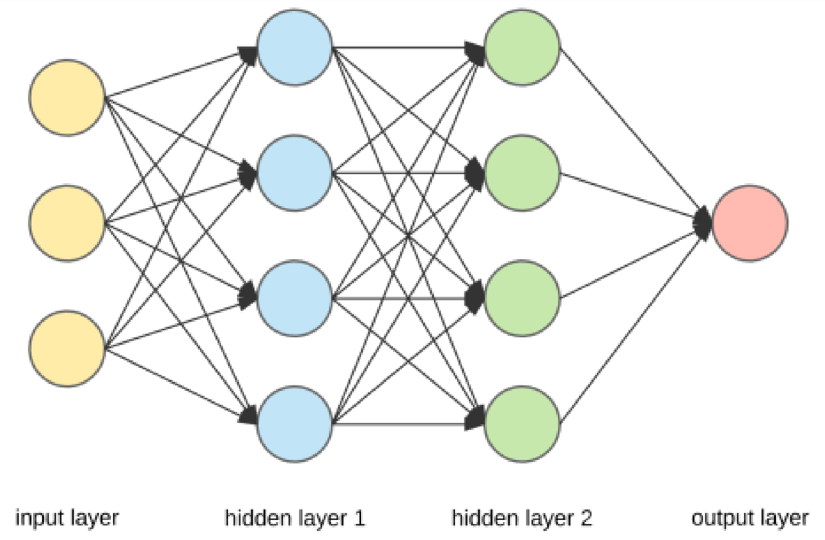
Uses a parameter 'k' that refers to the number of nearest neighbors to include

The optimal k value usually is the square root of N, where N is the total number of samples

Simple and easy to implement

The algorithm may get significantly slower as the number of predictors increase

ARTIFICIAL NEURAL NETWORK MODEL (ANN)



Based on how the human brain processes information

Learns by processing examples of inputs with their results

Composed by artificial neurons conceptually derived from biological neurons

Neurons are organized in multiple layers

MODEL COMPARISON

	kNN No Segmentation	kNN Cluster 0	kNN Cluster 1	kNN Cluster 2	kNN Cluster 3	ANN
Accuracy	77.90%	75.52%	80.68%	77.49%	75.66%	82.05%
True Positive Rate	6.95%	10.86%	3.87%	7.53%	6.84%	84.53%
False Positive Rate	1.69%	5.42%	1.36%	2.10%	2.65%	35.50%
ROC	65.96%	61.95%	64.47%	65.41%	62.87%	76.50%

Accuracy, True Positive Rate, False Positive Rate, and Specificity concludes that Neural Network is the best Model

CONCLUSION

ANN is the model that shows the best results for predicting a loan default from a bank

Further analysis to be conducted is recommended to have income levels, occupation, and loan type.

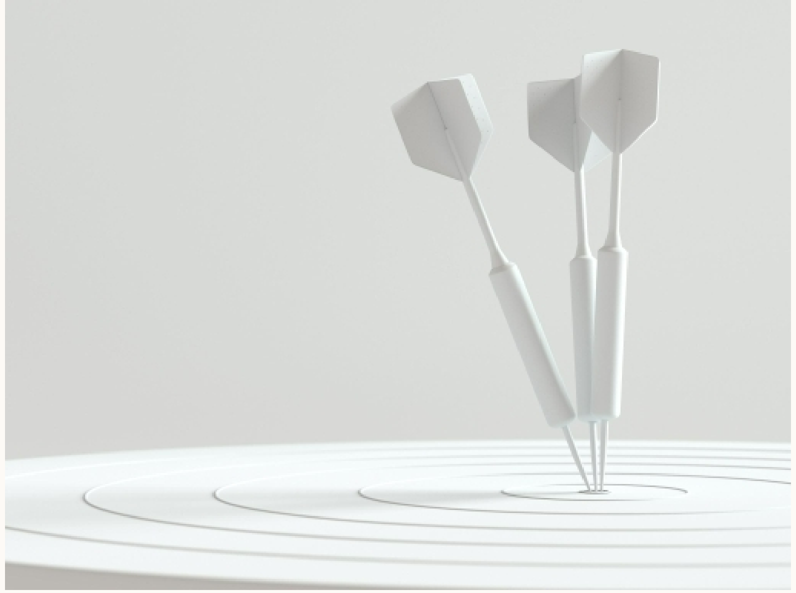
RECOMMENDATION

TARGET AUDIENCE

- Women between 25 and 40 years old with university degree

NON- TARGET AUDIENCE

- People over 60 years old that have only high school education



THANK YOU

APPENDIX

Q1: SLICE AND

Q1.1 How many customers are in the sample?

```
In [11]:  bank.shape

Out[11]:  (30000, 24)
```

There are 30,000 customers in the sample.

Q1.2 What is the most common sex in the sample?

```
In [12]:  bank["SEX"].value_counts()

Out[12]:  2    18112
          1    11888
          Name: SEX, dtype: int64
```

So we conclude that Male = 11888 and Female = 18112

```
In [13]:  male = 11888
          female = 18112
          common_sex = female - male
          Total_sex = female + male
          print (common_sex)

          6224
```

The most common sex in this sample is females as there are 6,224 more females versus males.

```
In [14]:  percentage_male = round((male/Total_sex)*100)
          percentage_female = round((female/Total_sex)*100)
          print("Percentage of Male " , percentage_male,"%")
          print("Percentage of Female " , percentage_female,"%")

          Percentage of Male  40 %
          Percentage of Female  60 %
```

Sample is 60 % female and 40 % male.

Q1: SLICE AND DICE

Q1.3 Which sex has the most defaults?

```
In [15]:  bank_male = bank[bank["SEX"] == 1]
          bank_female = bank[bank["SEX"] == 2]
```

```
In [16]:  #male count - 0 = No Default and 1 = Default
          bank_male["default payment next month"].value_counts()

Out[16]:  0    9015
          1    2873
          Name: default payment next month, dtype: int64
```

```
In [17]:  male_default = 2873
          male_no_default = 9015
          Total = male_default + male_no_default
          Percentage_default = (male_default/Total)*100
          print("From a percentage prospective, male default rate :",Percentage_default,"%" )

          From a percentage prospective, male default rate : 24.16722745625841 %
```

```
In [18]:  #Female count - 0 = No Default and 1 = Default
          bank_female["default payment next month"].value_counts()

Out[18]:  0    14349
          1     3763
          Name: default payment next month, dtype: int64
```

```
In [19]:  female_default = 3763
          female_no_default = 14349
          total = female_default + female_no_default
          Percentage_default_female = (female_default/total)*100
          print("From a percentage prospective, female default rate :",Percentage_default_female,"%" )

          From a percentage prospective, female default rate : 20.776280918727917 %
```

From a percentage perspective, males have a higher rate of defaults (24.17%) compared to females (20.78%)

Q1: SLICE AND DICE

Q1.4 How many distinct values does marriage take on?

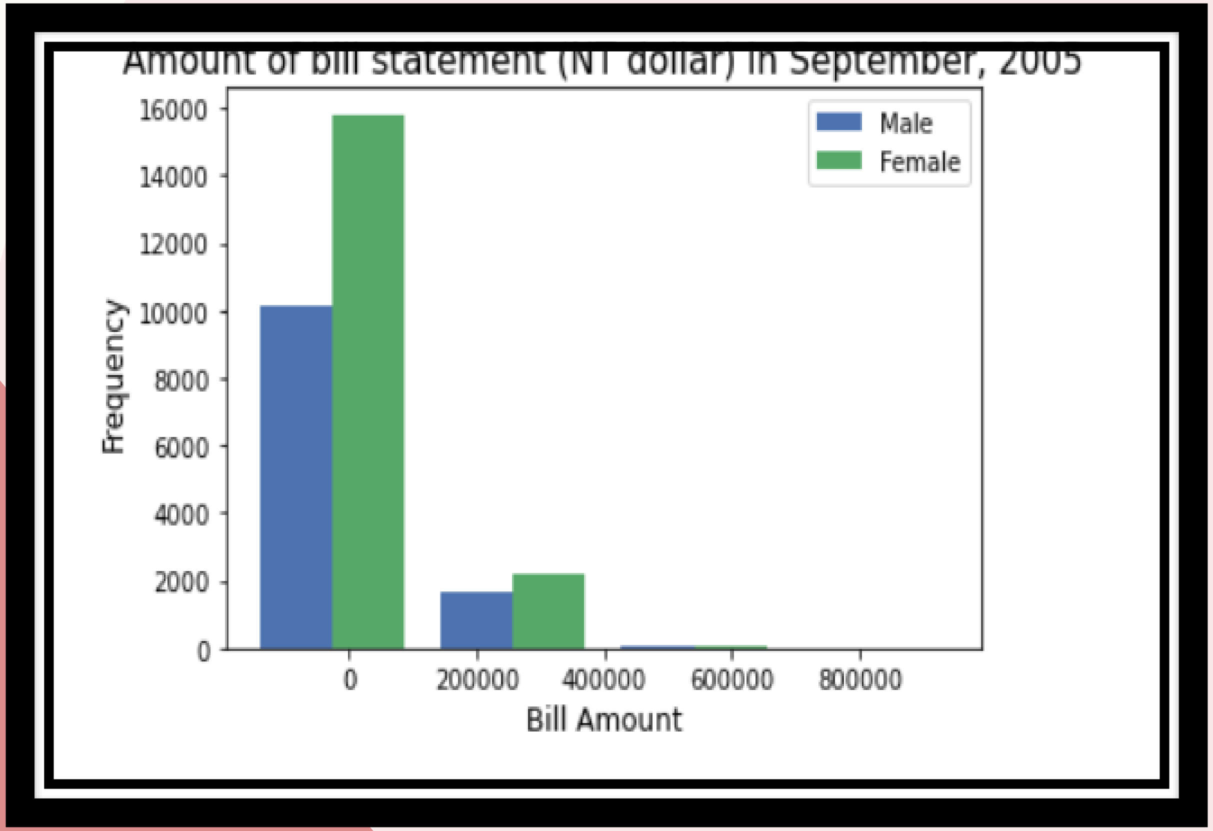
In [20]:
```
bank["MARRIAGE"].value_counts()
```

Out[20]:
```
2    15964
1    13659
3      323
0       54
Name: MARRIAGE, dtype: int64
```

There are 4 Distinct Value for Marriage

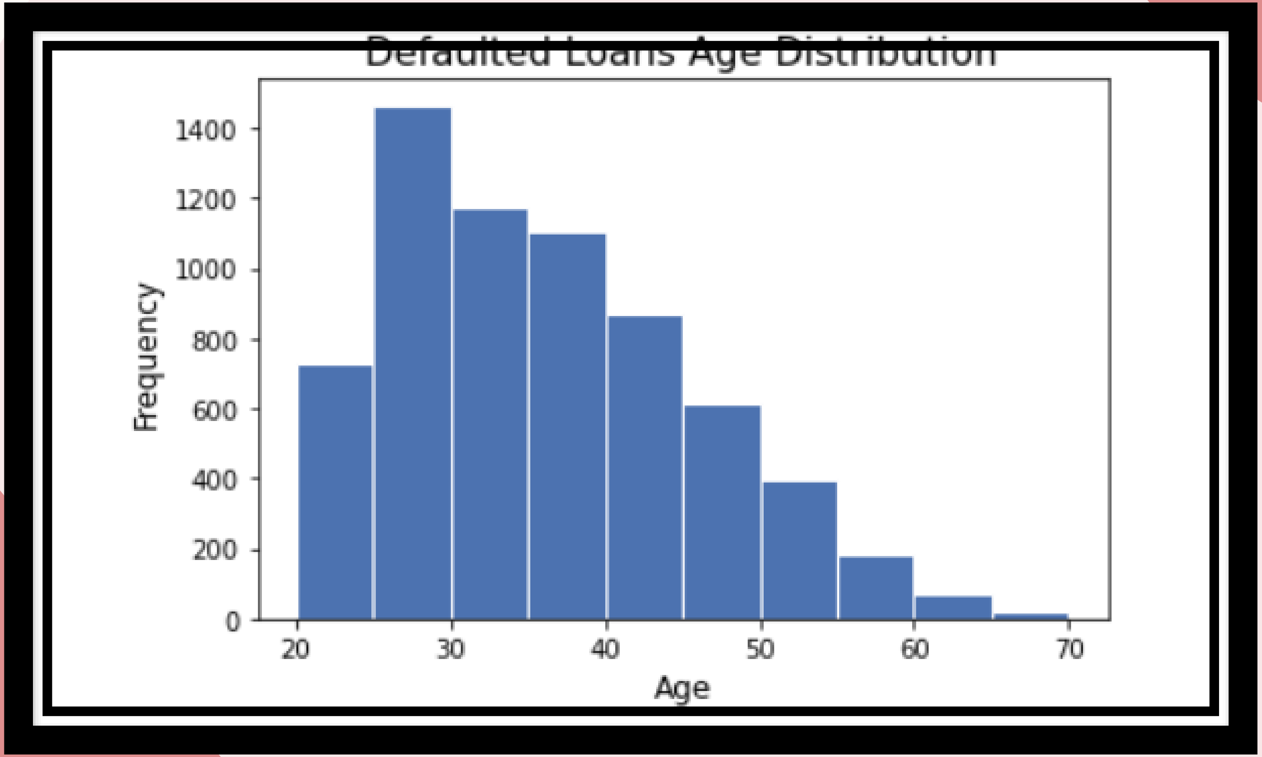
Q2: HISTOGRAMS

Q2.1 HOW IS BILL_AMT1 DISTRIBUTED BY SEX?



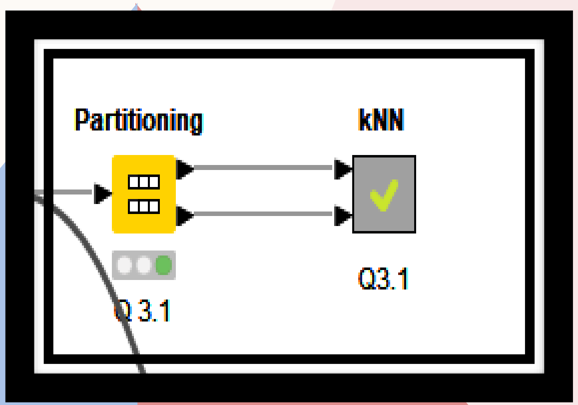
Q2: HISTOGRAMS

Q2.2 DOES THERE APPEAR TO BE ANY RELATIONSHIP BETWEEN DEFAULT AND AGE?

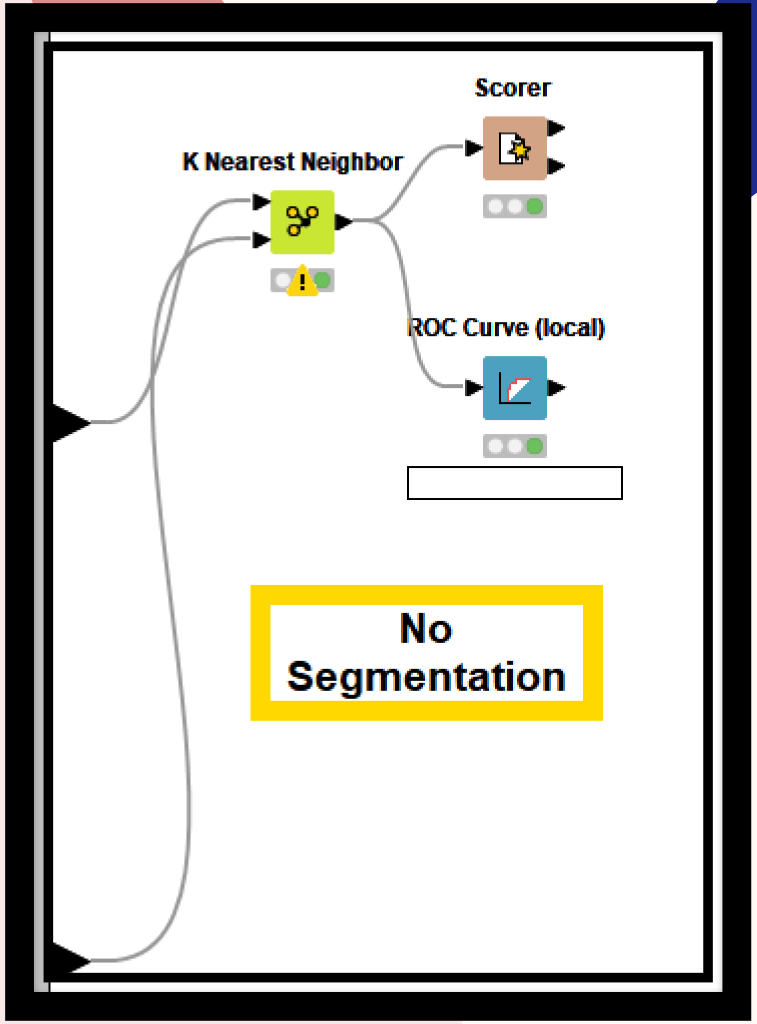


Q3: KNN MODEL

Q3.1 Build a model of default using kNN. Randomly partition the data into a training set (70%) and a validation set (30%). What value of k did you decide to use and why?

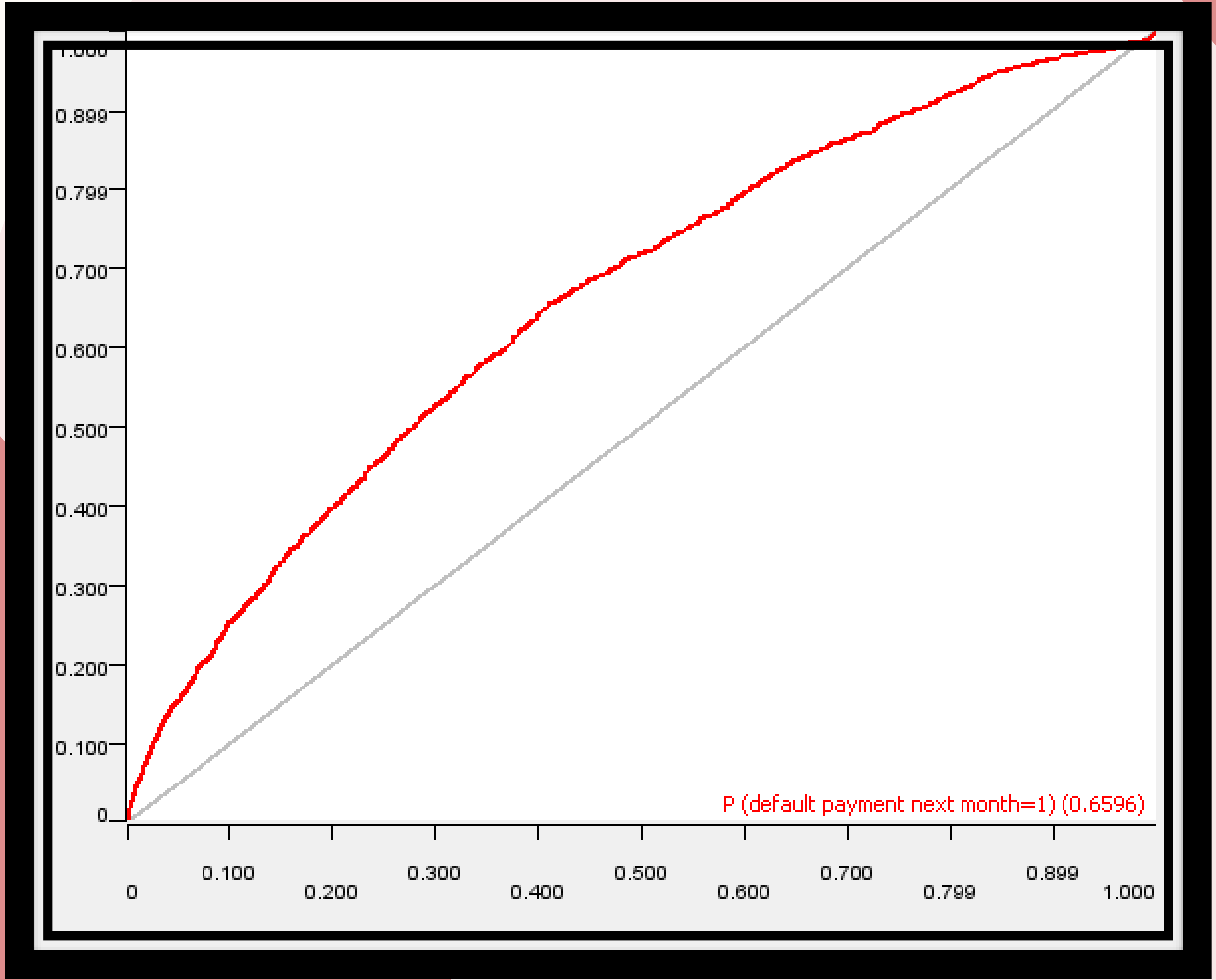


We used k = 95, that is the root square of n



Q3: KNN MODEL

Q3.2 Score the validation data (predict) using the model. Produce a confusion table and an ROC for the scored validation data.



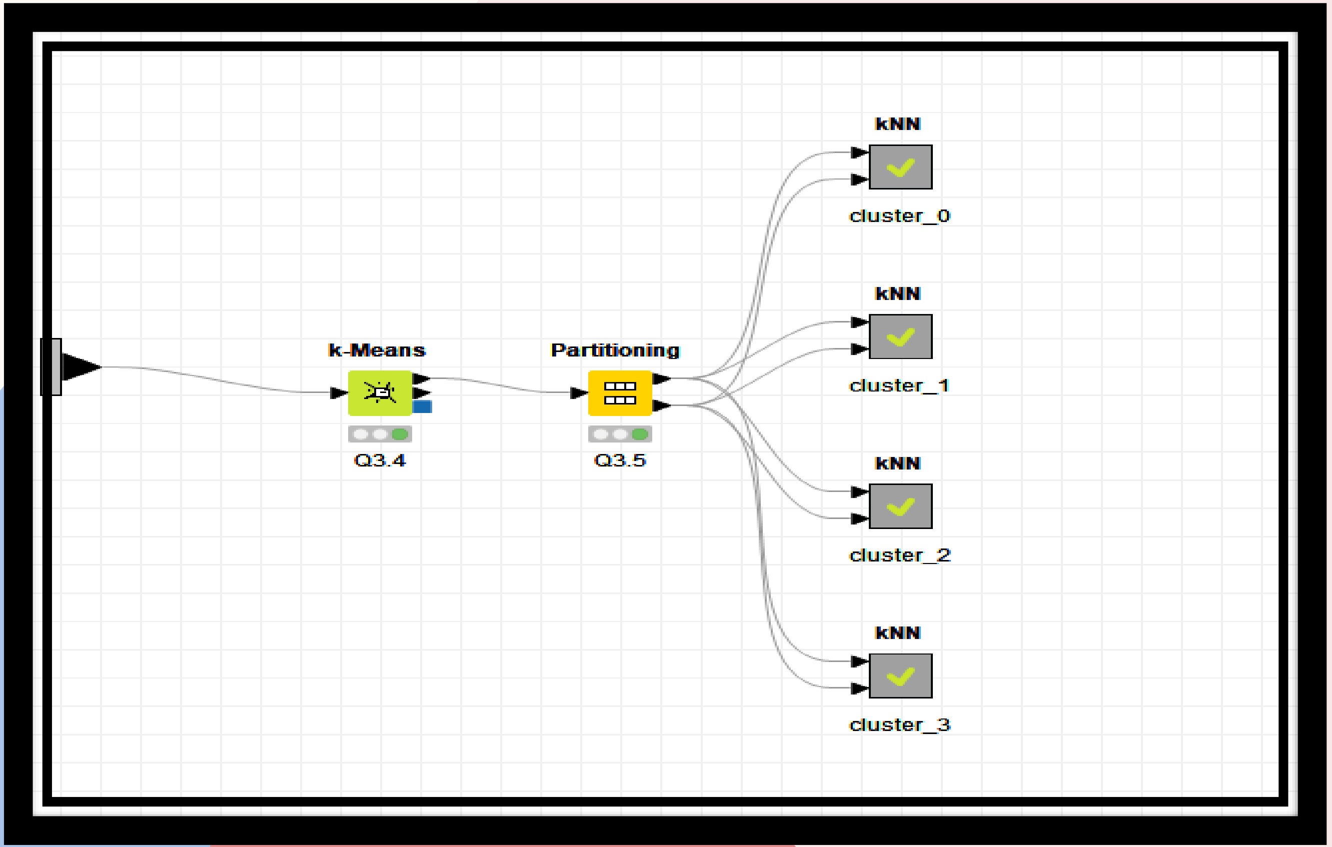
Q3: KNN MODEL

Q3.3 From the confusion table calculate the following metrics: accuracy, misclassification rate, true positive rate, false positive rate, specificity, precision, and prevalence ?

	kNN No Segmentation
Accuracy	77.90%
Missclassification Rate	22.12%
True Positive Rate	6.95%
False Positive Rate	1.69%
Specificity	98.31%
Precision	54.26%
Prevalence	2.87%
ROC	65.96%

Q3: KNN MODEL

Q3.4 Use k-means clustering to segment the customers on AGE. What value of k did you decide to use and why?
Q3.5 Build a model of default using kNN for each segment. Randomly partition the data into a training set (70%) and a validation set (30%) for each segment. What value of k did you decide to use and why?
Q3.6 Score the validation data (predict) using the models. Produce a confusion table for the scored validation data for each segment. How do they compare?



For k-means clustering we used k = 4, assuming that would be a good split by generation age (silent, boomers, generation X, millennials)

For each cluster w selected k = 44, 52, 52 and 40 respectively, using the same logic of the square root of n

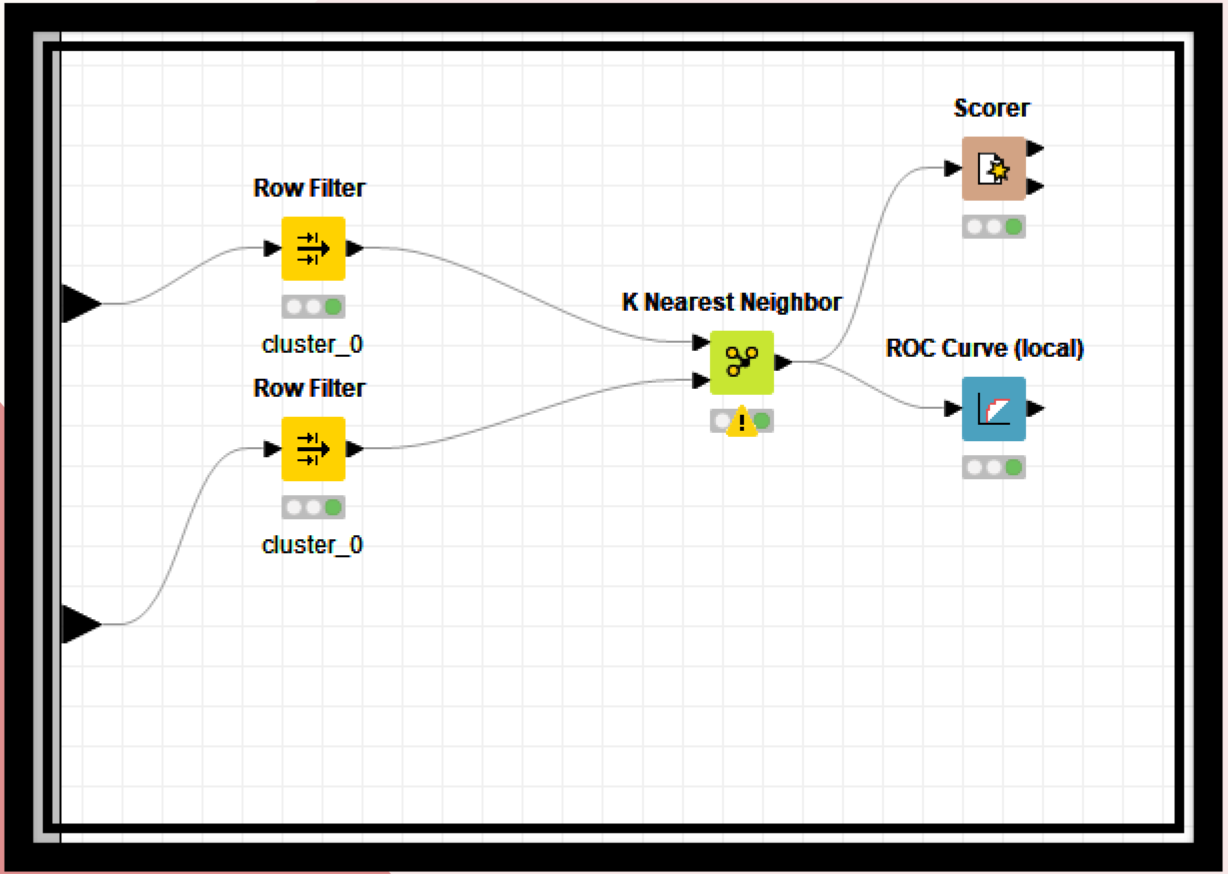
Q3: KNN MODEL

Q3.7 From the confusion tables for each segment calculate the following metrics: accuracy, misclassification rate, true positive rate, false positive rate, specificity, precision, and prevalence. How do they compare?

	kNN Cluster 0	kNN Cluster 1	kNN Cluster 2	kNN Cluster 3
Accuracy	75.52%	80.68%	77.49%	75.66%
Missclassification Rate	24.48%	19.32%	22.51%	24.34%
True Positive Rate	10.86%	3.87%	7.53%	6.84%
False Positive Rate	5.42%	1.36%	2.10%	2.65%
Specificity	94.58%	98.64%	97.90%	97.35%
Precision	37.12%	40.00%	51.11%	44.83%
Prevalence	6.66%	1.83%	3.33%	3.66%
ROC	61.05%	64.47%	65.41%	62.87%

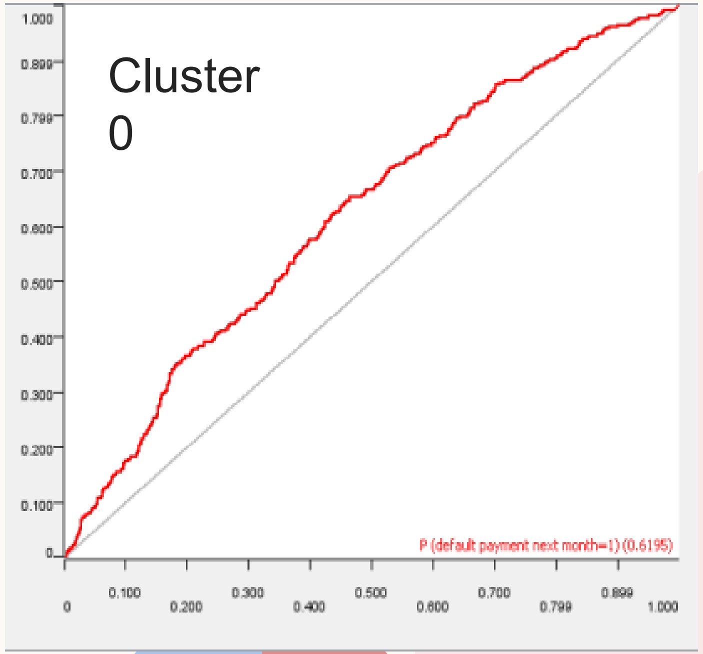
Q3: KNN MODEL

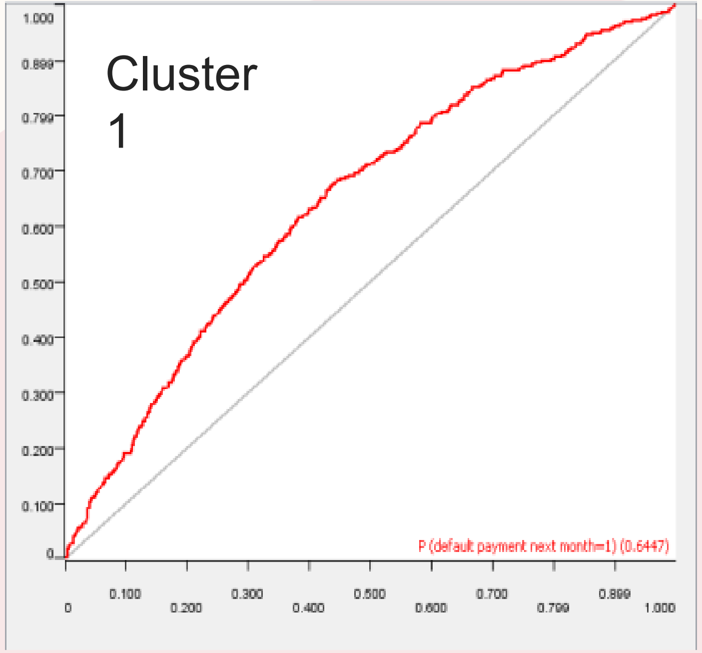
Q3.8 Produce an ROC curve for each AGE segment and report the AUCs.

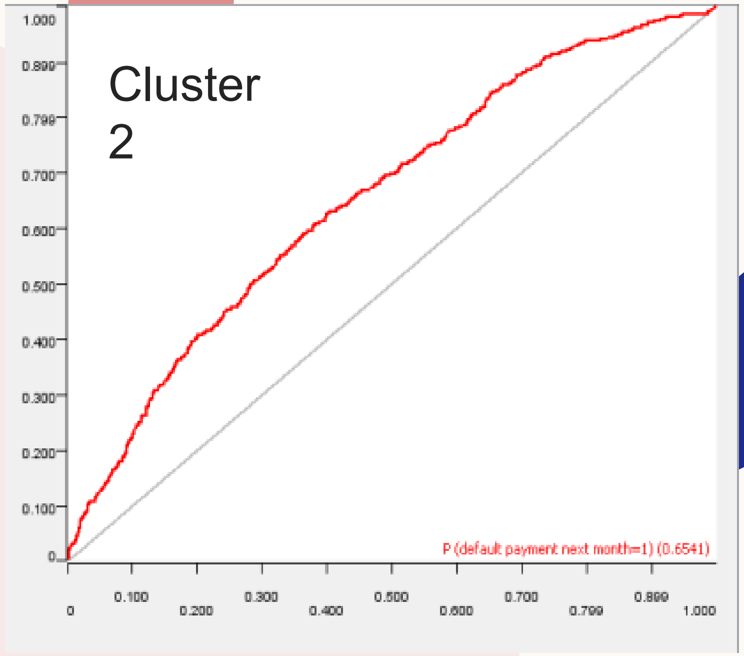


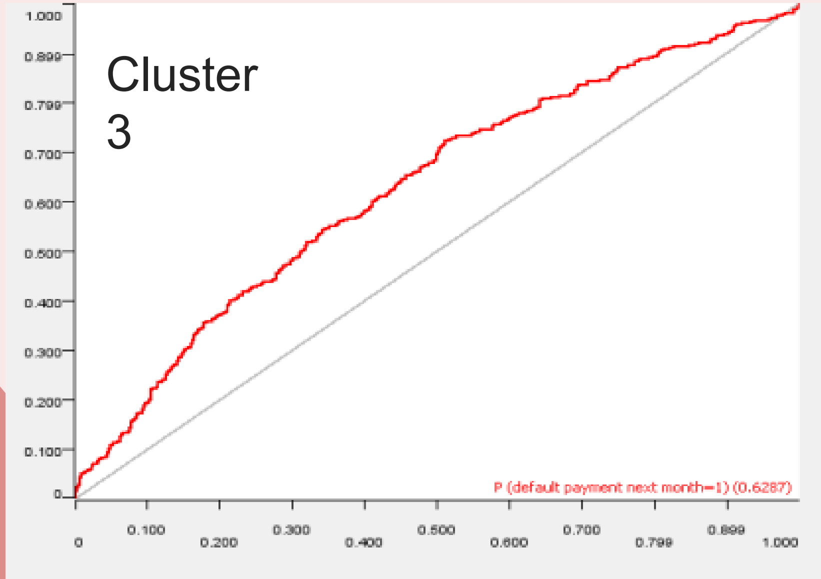
This is the same structure for all the 4 clusters

Q3: KNN MODEL


Cluster 0
P (default payment next month=1) (0.6195)


Cluster 1
P (default payment next month=1) (0.6447)


Cluster 2
P (default payment next month=1) (0.6541)


Cluster 3
P (default payment next month=1) (0.6287)

Q4: NEURAL NETWORK MODEL

Q4.1 Build a model of default using ANN. Randomly partition the data into a training set (70%) and a validation set (30%).

Q4.1 Build a model of default using ANN. Randomly partition the data into a training set (70%) and a validation set (30%).

In [42]:

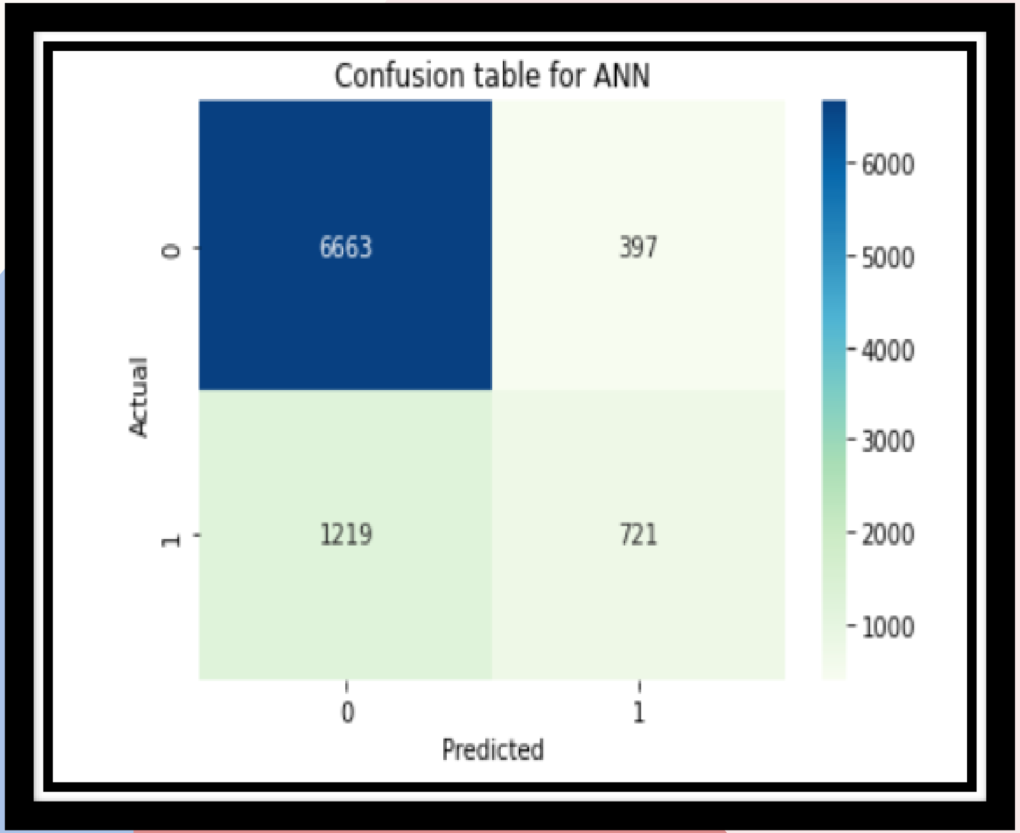
```
#Neural Network setup
newX = bank.drop(columns=['default payment next month'])
y = bank["default payment next month"]
x_train, x_test, y_train, y_test = train_test_split(newX, y, test_size=0.30, random_state=0)

scaler = StandardScaler().fit(x_train)
x_train = scaler.transform(x_train)
x_test = scaler.transform(x_test)
```

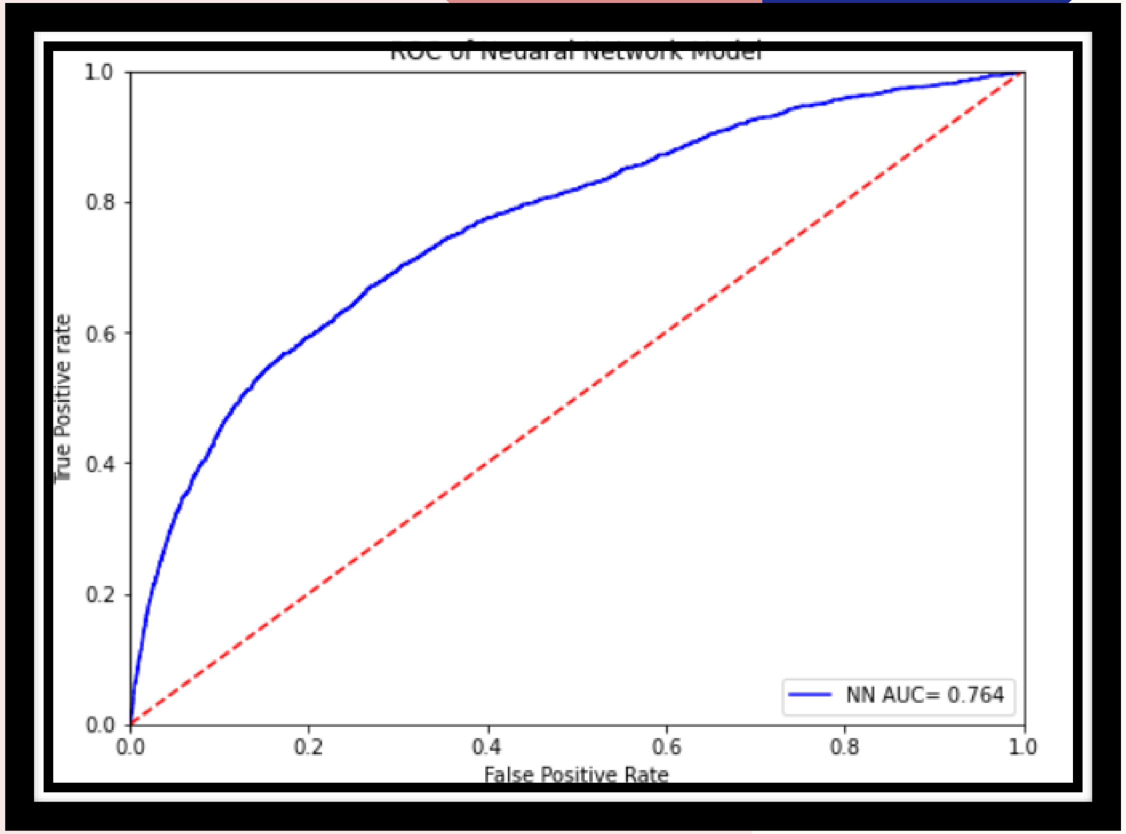
In [43]:

```
#Define ANN model
ANNmodel = Sequential()

ANNmodel.add(Dense(10, activation='relu', input_shape=(len(newX.columns),)))
ANNmodel.add(Dense(6, activation='relu'))
ANNmodel.add(Dense(1, activation='sigmoid'))

ANNmodel.compile(loss='binary_crossentropy',
          optimizer='adam',
          metrics=['accuracy'])
```

Q4: NEURAL NETWORK MODEL

Q4.2 Score the validation data (predict) using the model. Produce a confusion table and an ROC for the scored validation data.





Q4: NEURAL NETWORK MODEL

Q4.3 From the confusion table calculate the following metrics: accuracy, misclassification rate, true positive rate, false positive rate, specificity, precision, and prevalence

	ANN
Accuracy	82.05%
Missclassification Rate	17.90%
True Positive Rate	84.53%
False Positive Rate	35.50%
Specificity	64.49%
Precision	94.37%
Prevalence	78.45%
ROC	76.50%

Q5: COMPARE MODELS

	kNN No Segmentation	kNN Cluster 0	kNN Cluster 1	kNN Cluster 2	kNN Cluster 3	ANN
Accuracy	77.90%	75.52%	80.68%	77.49%	75.66%	82.05%
Missclassification Rate	22.12%	24.48%	19.32%	22.51%	24.34%	17.90%
True Positive Rate	6.95%	10.86%	3.87%	7.53%	6.84%	84.53%
False Positive Rate	1.69%	5.42%	1.36%	2.10%	2.65%	35.50%
Specificity	98.31%	94.58%	98.64%	97.90%	97.35%	64.49%
Precision	54.26%	37.12%	40.00%	51.11%	44.83%	94.37%
Prevalence	2.87%	6.66%	1.83%	3.33%	3.66%	78.45%
ROC	65.96%	61.95%	64.47%	65.41%	62.87%	76.50%