**A Project On**

**“bank customer prediction”**

**Prepared by**

Rathod Ankit (16IT104)

**Under the supervision of**

Prof. Jalpesh Vasa

A Report Submitted to

Charotar University of Science and Technology

for Partial Fulfillment of the Requirements for the

Degree of Bachelor of Technology

in Information Technology

IT322 Software Group Project-IV (5th sem)

**Submitted at**



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**Chandubhai S. Patel Institute of Technology**

**At: Changa, Dist: Anand – 388421**

**July-November 2018**



**CERTIFICATE**

This is to certify that the report entitled “**Bank Customer Prediction**” is a bonafied work carried out by **Mr. Ankit Rathod(16IT104)** under the guidance and supervision of **Prof. Jalpesh Vasa** for the subject **Software Group Project-IV(IT322)** of **5th** Semester of Bachelor of Technology in **Information Technology** at Faculty of Technology & Engineering – CHARUSAT, Gujarat.

To the best of my knowledge and belief, this work embodies the work of candidate **himself**, has duly been completed, and fulfills the requirement of the ordinance relating to the B.Tech. Degree of the University and is up to the standard in respect of content, presentation and language for being referred to the examiner.

|  |
| --- |
| Under supervision of,  Prof. Jalpesh Vasa  Assistant Professor  Dept. of Information Technology  CSPIT, Changa, Gujarat. |
| Prof. Parth Shah  Head & Associate Professor  Department of Information Technology  CSPIT, Changa, Gujarat. |

**Chandubhai S Patel Institute of Technology**

At: Changa, Ta. Petlad, Dist. Anand, PIN: 388 421. Gujarat

**TABLE OF CONTENTS**

**Acknowledgement 2**

**Abstract 3**

**Chapter 1 Introduction 5**

1.1 Project Summary 5

1.1.1 Purpose….……………………………………………………………………...5

1.2 Scope 5

1.3 Objective 5

**Chapter 2 System Requirements Study 6**

2.1 Tools & Technology Used 6

**Chapter 3 System Design 8**

3.1 Project Flow 8

**Chapter 4 Implementation Planning 9**

4.1 Code 9

4.2 Snapshots of project 11

**Chapter 5 Limitations and Future Enhancement 13**

**Chapter 6 Conclusion 14**

**References 15**

**ACKNOWLEDGEMENT**

We have found this rare opportunity to evince a word of thanks to all those who played a key role in the successful completion of our project. We are thankful to our Head of Department **Prof. Parth Shah** Sir for giving the chance as well as support for all the time being. And his able guidance and continuous encouragement made us work in all the challenges during project development.

We express deep gratitude to **Mr. Jalpesh Vasa**, assistant professor and internal project guide from Faculty of Engineering, CHARUSAT for their valuable suggestions, help and moral support. Finally, most of all, we thank our family members for their unconditional love, encouragement and support to complete our project work.

We also thank to all those who could not find a separate name but have helped directly and indirectly.

**Abstract**

Artificial neural network which predicts that whether customer will leave bank or not using different machine learning models ,this model is made from 10,000 realtime data of a bank in which 8000 are training data and 2000 are testing data. Machine Learning is a great way to predict things and study datasets. The work has been implemented using Python (3.5.2), Pandas, keras and NumPy.

**LIST OF FIGURES**

* **Figure 1 datase in Excel form …………………………...13**
* **Figure 2 output start……………………………………...13**
* **Figure 3 output end…………..…………………………...14**
* **Figure 4 confusion matrix…….…………………………...14**

**Chapter 1: Introduction**

* 1. **Project Summary**

**●** In this application, This model will tell us if the customer is going or not to exit from the bank. This will be done by trained model which will we implement using machine learning algorithm.

**1.1.1 Purpose**

* Our purpose is to predicts that whether customer will leave bank or not.
  1. **Scope**

● In this application we have focussed on smaller datasets. We can make this more accurate by taking large datasets.

* 1. **Objective**
* Our objective is to learn about machine learning, python and artificial neural network.

**Chapter 2: System Requirements Study**

**2.2 Tools & Technology Used**

* **Tools:**
* Anaconda 3
* Spyder
* **Technology**
* machine learning
* artificial neural network machine learning
* **Library**
* NumPy
* Pandas
* Keras
* matplotlib.pyplot

**Chapter 3: System Design**

* 1. **Project Flow**

1. Randomly initialise the weights to small numbers close to 0, but not 0
2. Input the first observation of your dataset, in the input layer, each feature in one input node
3. Forward-Propagation: from left to right, the neurons are activated in a way the impact of each neuron’s activation is limited by the weights and it runs until getting the y.
4. Compare the predicted result to the actual result. Measure the error generated.
5. Back-propagation: from right to left. The error is back propagated. Update the weights according to how much they are responsible for the error. The learning rate decides how much we update the weights.
6. Repeat steps 1 to 5 and update the weights after each observation (**Reinforcement Learning**). Repeat steps 1 to 5 but update the weights only after a batch of observation (**Batch Learning**)
7. When the whole training set passed through the ANN, that makes an epoch. Redo more epochs.

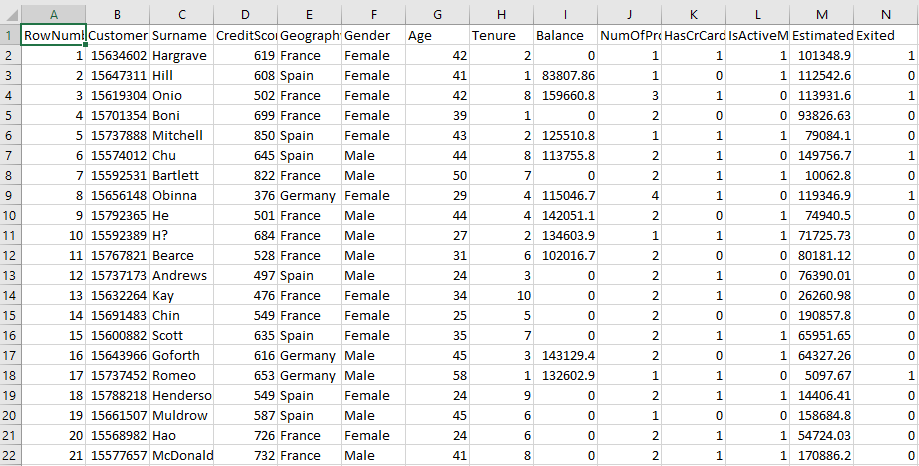
**Chapter 4 Implementation Planning**

**4.1 Code**

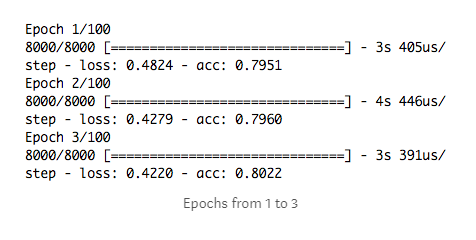
|  |
| --- |
| **temp.py** |
| # Installing Theano | | |
| # pip install --upgrade --no-deps git+git://github.com/Theano/Theano.git | | |
|  | | |
| # Installing Tensorflow | | |
| # pip install tensorflow | | |
|  | | |
| # Installing Keras | | |
| # pip install --upgrade keras | | |
|  | | |
| # Part 1 - Data Preprocessing | | |
|  | | |
| # Importing the libraries | | |
| import numpy as np | | |
| import matplotlib.pyplot as plt | | |
| import pandas as pd | | |
|  | | |
|  | | |
| import keras | | |
|  | | |
| # Importing the dataset | | |
| dataset = pd.read\_csv('Churn\_Modelling.csv') | | |
| X = dataset.iloc[:, 3:13].values | | |
| y = dataset.iloc[:, 13].values | | |
|  | | |
| # Encoding categorical data | | |
| from sklearn.preprocessing import LabelEncoder, OneHotEncoder | | |
| labelencoder\_X\_1 = LabelEncoder() | | |
| X[:, 1] = labelencoder\_X\_1.fit\_transform(X[:, 1]) | | |
| labelencoder\_X\_2 = LabelEncoder() | | |
| X[:, 2] = labelencoder\_X\_2.fit\_transform(X[:, 2]) | | |
| onehotencoder = OneHotEncoder(categorical\_features = [1]) | | |
| X = onehotencoder.fit\_transform(X).toarray() | | |
| X = X[:, 1:] | | |
|  | | |
| # Splitting the dataset into the Training set and Test set | | |
| from sklearn.cross\_validation import train\_test\_split | | |
| X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size = 0.2, random\_state = 0) | | |
|  | | |
| # Feature Scaling | | |
| from sklearn.preprocessing import StandardScaler | | |
| sc = StandardScaler() | | |
| X\_train = sc.fit\_transform(X\_train) | | |
| X\_test = sc.transform(X\_test) | | |
|  | | |
| # Part 2 - Now let's make the ANN! | | |
|  | | |
| # Importing the Keras libraries and packages | | |
| import keras | | |
| from keras.models import Sequential | | |
| from keras.layers import Dense | | |
|  | | |
| # Initialising the ANN | | |
| classifier = Sequential() | | |
|  | | |
| # Adding the input layer and the first hidden layer | | |
| classifier.add(Dense(units = 6, kernel\_initializer = 'uniform', activation = 'relu', input\_dim = 11)) | | |
|  | | |
| # Adding the second hidden layer | | |
| classifier.add(Dense(units = 6, kernel\_initializer = 'uniform', activation = 'relu')) | | |
|  | | |
| # Adding the output layer | | |
| classifier.add(Dense(units = 1, kernel\_initializer = 'uniform', activation = 'sigmoid')) | | |
|  | | |
| # Compiling the ANN | | |
| classifier.compile(optimizer = 'adam', loss = 'binary\_crossentropy', metrics = ['accuracy']) | | |
|  | | |
| # Fitting the ANN to the Training set | | |
| classifier.fit(X\_train, y\_train, batch\_size = 10, epochs = 100) | | |
|  | | |
| # Part 3 - Making predictions and evaluating the model | | |
|  | | |
| # Predicting the Test set results | | |
| y\_pred = classifier.predict(X\_test) | | |
| y\_pred = (y\_pred > 0.5) | | |
|  | | |
| # Making the Confusion Matrix | | |
| from sklearn.metrics import confusion\_matrix | | |
| cm = confusion\_matrix(y\_test, y\_pred) | | |

**4.3 Snapshots of project**

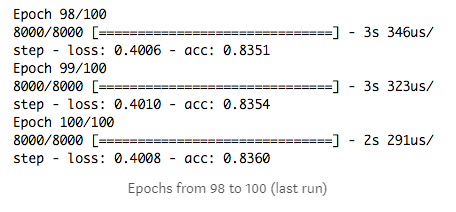
**Dataset in Excel form:**



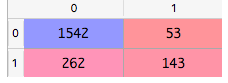
**Figure 5 datase in Excel form**



**Figure 6 output start**



**Figure 7 output end**



**Figure 8 confusion matrix**

Now we can calculate the accuracy with the correct prediction divided by the total number of predictions = (1542+143)/(1542+53+262+143) →84,2%

This is a good result and we have our model with 84% of accuracy.

**Chapter 5: Limitations and Future Enhancement**

**5.1 Limitations**

* Prediction can be true atlest 84%.
* required a lot of data in particular bank.
  1. **Future Enhancement**
* New future can be implemented like predicate customers monthly transaction cost and acoredling that suggest difrent polis

**Chapter 6 :Conclusion**

Artificial neural network which predicts that whether customer will leave bank or not using different machine learning models ,this model is made from 10,000 realtime data of a bank in which 8000 are training data and 2000 are testing data. Machine Learning is a great way to predict things and study datasets. This is a good result and we have our model with 84% of accuracy.

**References:-**

1. [**https://medium.com/@luigi.bungaro/artificial-neural-network-for-customers-churn-prediction-python-code-part-2-2-68366e864966**](https://www.coursera.org/learn/machine-learning)
2. [**https://becominghuman.ai/artificial-neural-network-for-customers-churn-prediction-python-code-part-1-27797a110a91**](%20https://becominghuman.ai/artificial-neural-network-for-customers-churn-prediction-python-code-part-1-27797a110a91)
3. [**https://machinelearningmastery.com/introduction-python-deep-learning-library-keras/**](%20https://machinelearningmastery.com/introduction-python-deep-learning-library-keras/)
4. [**https://www.glassdoor.co.in/Reviews/Sigmoid-Analytics-Reviews-E785952.htm**](https://www.glassdoor.co.in/Reviews/Sigmoid-Analytics-Reviews-E785952.htm)
5. [**http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks**](https://www.investopedia.com/university/stocks/)