FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY UNIVERSITI TEKNIKAL MALAYSIA MELAKA (UTeM)

SEMESTER II – SESSION 2022/2023

BITI3133 NEURAL NETWORK

MINI PROJECT
CUSTOMER CHURN PREDICTION USING ARTIFICIAL NEURAL NETWORK

PREPARED FOR:

PROFESOR TS. DR. BURHANUDDIN BIN MOHD ABOOBAIDER

PREPARED BY:

NO.	STUDENT'S NAME	MATRIC NO.	YEAR	ROLES
1	NASRUL FITRI BIN AMLEE	B032210111	3	PROGRAMMER
2	FARHAN HAFIZI BIN ISMAIL	B032210034	3	TESTER
3	ROSHINI A/P RAMES	B032110385	3	ASSISTANT MANAGER
4	HERISH CHAUDRAY A/L RAVANTHERAN	B032110409	3	PROJECT MANAGER

TABLE OF CONTENTS

1.0 INTRODUCTION

Customer churn prediction is the process of analysing and forecasting which customers are likely to discontinue their usage of a product or service, commonly referred to as churn. This predictive capability holds significant value for businesses as it enables them to proactively identify customers at risk of attrition and take preemptive measures to mitigate their departure. In essence, when a consumer has subscribed to a particular service, it becomes crucial to ascertain the probability of their discontinuation and take timely actions to address their concerns.

To illustrate, let us consider a scenario where an individual has subscribed to a premium service. Subsequently, if they contemplate cancelling or terminating their subscription, they would reach out to the company. In response, the company would endeavour to retain the customer by presenting supplementary features or benefits to dissuade them from leaving. This becomes imperative for any industry, as the loss incurred due to customer attrition is detrimental to their operations.

Customer churn prediction assumes critical importance for many businesses owing to the fact that acquiring new customers often incurs higher costs than retaining existing ones. It serves as a metric to gauge the departure of customers and comprehend the underlying reasons for their disengagement. Several methods exist for calculating customer churn, with one approach involving the division of the number of customers who discontinue their association with a business within a given time interval by the number of customers at the beginning of said period.

2.0 PROBLEM STATEMENT

Customer churn prediction is aimed to accurately identify customers who are at risk of churning or discontinuing their relationship with a business. The goal is to develop predictive models that can forecast the likelihood of churn based on historical customer data, such as demographics, transaction history, usage patterns, and customer interactions. By identifying customers at risk of churn, businesses can take proactive measures to retain them, such as implementing targeted marketing campaigns, personalised offers, or improved customer service. The objective is to reduce customer attrition and enhance customer satisfaction and loyalty, ultimately contributing to the overall success and profitability of the business.

3.0 OBJECTIVES

The primary objective is to develop an Artificial Neural Network (ANN) model that can accurately predict which customers are likely to churn. This involves training the model to learn patterns and relationships in historical customer data to make accurate predictions.

Churn prediction using ANN can help businesses optimise resource allocation by focusing their efforts and resources on customers who are most likely to churn. By prioritising retention efforts based on the predicted churn probability, businesses can allocate their resources effectively and efficiently.

Another objective is to continually refine and improve the churn prediction model. This involves monitoring the performance of the ANN model, collecting new customer data, and updating the model to adapt to changing customer behaviours and market dynamics. The objective is to ensure that the churn prediction model remains accurate and relevant over time.

4.0 SAMPLE DATA DESCRIPTION

To commence our analysis, we will utilise the Churn Modelling dataset, sourced from Kaggle, as our foundational dataset. This dataset encompasses numerous features that will serve as the basis for predicting customer churn. Prior to delving into data processing procedures, it is essential to comprehend the dataset's structure, facilitating easier manipulation and analysis.

The dataset comprises 10000 rows and 14 columns, with each row representing a distinct customer entry and each column signifying a specific customer attribute. These attributes are pivotal in forecasting the likelihood of churn for individual customers. The dimensions that we will be dealing with are as follows:-

i. RowNumber:- Represents the number of rows
ii. CustomerId:- Represents customerId
iii. Surname:- Represents surname of the customer
iv. CreditScore:- Represents credit score of the customer
v. Geography:- Represents the city to which customers belongs to
vi. Gender:- Represents Gender of the customer
vii. Age:- Represents age of the customer
viii. Tenure:- Represents tenure of the customer with a bank
ix. Balance:- Represents balance hold by the customer
x. NumOfProducts:- Represents the number of bank services used by the customer
xi. HasCrCard:- Represents if a customer has a credit card or not
xii. IsActiveMember:- Represents if a customer is an active member or not
xiii. EstimatedSalary:- Represents estimated salary of the customer
xiv. Exited:- Represents if a customer is going to exit the bank or not.

Our primary objective is to develop an Artificial Neural Network (ANN) that comprehensively incorporates all independent variables (first 13) to predict the likelihood of customer attrition, specifically whether the customer will exit the bank or not. In this context, the dependent variable "Exited" serves as the key indicator of customer departure. By leveraging the power of an ANN, we aim to accurately forecast customer churn and enable proactive measures to retain valuable clientele.

5.0 SOFTWARE AND AI TOOLS USED

To produce the Customer Churn Prediction using ANN , the programming language used for implementation is Python and it utilizes various libraries, such as NumPy and Pandas for data manipulation, TensorFlow for machine learning model development, and Matplotlib along with Seaborn for data visualization. The combination of these tools allows for effective data preprocessing, model building, and evaluation in machine learning projects.

Various tools address different facets of project planning and collaboration for the project management. Microsoft Project stands out as a comprehensive project management software developed by Microsoft, serving tasks such as project planning, scheduling, and resource management. GitHub, a web-based platform for version control using Git, is renowned for source code management and extends its functionality to include project management features, issue tracking, and collaboration capabilities. ClickUp emerges as a versatile productivity platform supporting task management, document sharing, and team collaboration. Each tool, including Microsoft Project, GitHub, and ClickUp, contributes unique strengths to project management, accommodating the diverse needs and preferences of project teams.

For specific tasks within AI project management, we leveraged Microsoft Project for generating Gantt charts, a crucial tool for visualizing project timelines and dependencies. GitHub played a pivotal role in version control, ensuring efficient tracking of code changes and collaboration among team members. Additionally, tools like ClickUp were instrumental in cost management and task coordination, facilitating seamless communication and document sharing within the AI project team. This strategic use of tools streamlined project workflows and enhanced overall efficiency in different aspects of AI project development.

6.0 PROJECT MANAGEMENT PROCESS

1) Initiating Process
a) Business Case of Customer Churn Prediction System

Executive Summary	The Customer attrition Prediction System is a strategic project that uses artificial neural networks (ANN) to **anticipate customer attrition** by **analysing historical customer data and predicting churn**. This proactive approach empowers businesses to optimize resource allocation, enhance customer retention strategies, and ultimately improve overall profitability.
Problem statements	1. Developing an accurate ANN model faces challenges in addressing complex data patterns, feature identification, and parameter fine-tuning. 2. Inefficient resource allocation arises due to the inability to effectively identify and prioritize customers most likely to churn. 3. The lack of objective solution where there is an importance for an automated system that can provide objective solution to improvise the business
Benefits	**Increased Customer Retention:** Accurate predictions enable targeted retention strategies, reducing churn rates and preserving customer relationships. **Resource Optimization: I**mproved allocation of resources, leading to cost savings by focusing efforts on customers with a higher likelihood of churn. **Competitive Edge:** Enhances business competitiveness through data-driven decision-making and proactive customer management.

Costs and Expected Return of Investment :	Budget of **RM 1,087,531** only Expected ROI: To **gain with a 20%** of return in the third year which is approximately **RM 216,306**

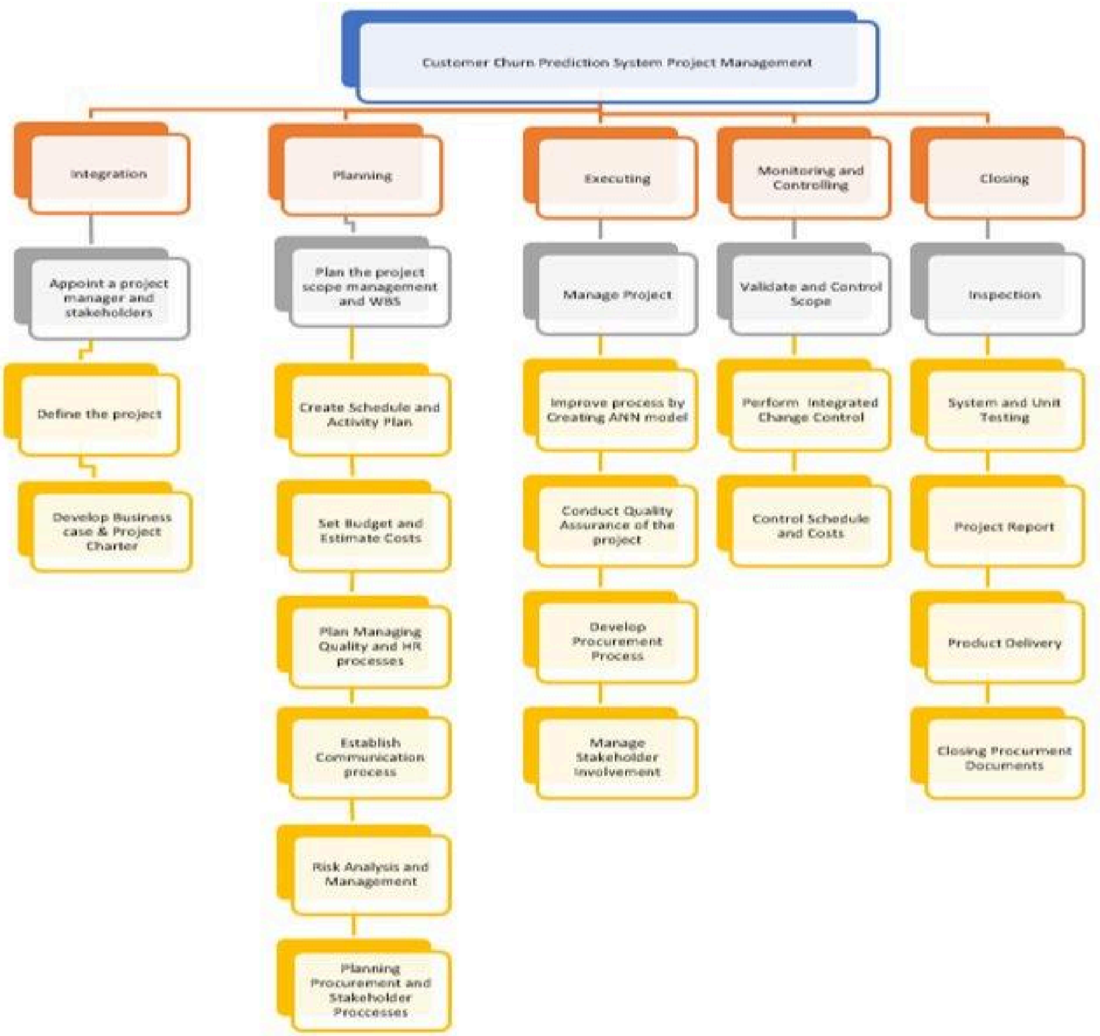
b)Project Charter

Project Name: Customer Churn Prediction System		
Project Description:	This project aims to build an intelligent system capable of analysing historical customer data and predicting the likelihood of churn. By employing ANN, the model aims to capture intricate patterns and relationships within the data, providing more accurate predictions compared to traditional models. This system analyses the historical customer data, predicts churn, and empowers businesses to proactively retain customers. The project involves data preprocessing, feature engineering, ANN model training, evaluation, integration with a simple and intuitive user interface and system testing to validate the system's accuracy performance.	
Project Objectives:	1. To develop an ANN model that accurately predict which customers are likely to churn. 2. To help businesses optimise resource allocation by focusing their efforts and resources on customers who are most likely to churn. 3. To continually refine the churn prediction model by monitoring the performance of an ANN model and increase the accuracy of the system	
Success Criteria:	1. High Prediction Accuracy: The customer churn prediction system should achieve a hight overall accuracy in predicting the customer churn. 2. Resource Allocation Improvement: The system should be able to demonstrate a measurable improvement in resource allocation efficiency with retaining customers who are less likely to churn. 3. Continuous Model Refinement: The system establishes updates to the model, incorporating new customer data and insights to accuracy over time and should be able to adapt to changes in customer behavior.	
Project Participants:	Role	Name

	Project Manager: Assistant Manager: Team Member 1: Team Member 2:	Herish Chaudray A/L Ravantheran Roshini A/P Rames Nasrul Fitri Bin Amlee Farhan Hafizi Bin Ismail
Budget:	Budget of RM 1,087,531 only	
Deliverables:	i. User Manual and Training Materials ii. System Documentation iii. Testing and Quality Assurance Reports iv. Technical Support	
Milestones:	i. System Design and development completed by end of Week 7 ii. User acceptance testing completed by end of Week 10 iii. System Implementation and Rollout completed by end of Week 13	
Potential Risks:	1. i. Data Quality and Availability: Incomplete or inaccurate customer data could compromise the accuracy of the churn prediction model. 2. ii. Model Complexity and Overfitting: The ANN model may become overly complex, leading to overfitting on the training data and poor generalization to new data. 3. iii. iii. Lack of Stakeholder Engagement: Inadequate involvement and feedback from stakeholders on project progress may result in misaligned expectations.	
Approval:	Investor:SRT.Inc Herish Project Manager: Herish Chaudray	Date:28/12/2023 Date:28/12/2023

2) Planning Process

a) Work Breakdown Strcuture(WBS)



b) Project Scope Management

Project Scope Management Process	
Project Title	Customer Churn Prediction System
Project Manager	Herish Chaudray A/L Ravantheran
Project Objectives	a. To develop an ANN model that accurately predict which customers are likely to churn. b. To help businesses optimise resource allocation by focusing their efforts and resources on customers who are most likely to churn. c. To continually refine the churn prediction model by monitoring the performance of an ANN model and increase the accuracy of the system
Project Deliverables	a. User Manual and Training Materials b. System Documentation c. Testing and Quality Assurance Reports d. Technical Support
Milestones	a. System Design and development completed by end of Week 7 b. User acceptance testing completed by end of Week 10 c. System Implementation and Rollout completed by end of Week 13
Project Constraints	a. Budget of RM b. Completion date of the project is December 28,2022

Assumptions	a. Availability of qualified development and testing resources b. Timely Approval of Project deliverables c. Availability of necessary hardware and software resources d. No major changes to project requirements or objectives

Cost Estimation

Cost estimation for the system is as follows:-

	Unit/Hrs	**Cost/unit/hr (RM)**	**Total (RM)**
Project Management			
Project manager	570	150	85,500
Project team members	1140	100	114,000
Contractors			68,000
Hardware			
Graphics card	2	1800	3600
Servers	1	2000	2000
Networking structures	1	3700	3700
Software			
Integrated Development Enviroments	1	1155.73	1155.73
Azure AI	2000	1.16	2320
Sotware Development			400,000
Testing			42,000
Training and Support			
Trainee cost	100	300	30,000
Travel cost	8	500	40,000
Project team members	1140	100	114,000
Reserves			181,255

TOTAL			**1,087,531**

Role	# Units/Hrs.	Cost/Unit/Hr.	Subtotals	Calculations
Data Scientist labor estimate	760	$100	$76,000	760 * 100
Machine Learning Engineer estimate	760	$100	$76,000	760 * 100
Project Manager estimate	380	$120	$45,600	380 * 120
Total labor estimate	**1900**		**$197,600**	Sum above two values

Task	Quantity	Conversion Factor	Function Points	Calculations
Data preprocessing	20	3	60	20 * 3
Model training	15	5	75	15 * 5
Model evaluation	10	4	40	10 * 4
Deployment	5	6	30	5 * 6
Monitoring & updates	5	2	10	5 * 2
Total function points			**215**	Sum above values

2) Executing Process for Customer Churn Prediction System

Perform Quality Assurance

Quality assurance (QA) in the context of a customer churn prediction system involves a meticulous and comprehensive process to ensure the system's reliability, accuracy, and compliance with predefined standards. The foundation of QA lies in defining clear requirements and objectives for the system, establishing the criteria against which its performance will be measured. Data quality assurance is paramount, involving thorough cleansing and preprocessing of training data to eliminate anomalies and maintain representativeness. The evaluation of AI models, using metrics such as accuracy, precision, recall, and cross-validation, ensures robustness and generalization to new data. Special attention is given to bias and fairness assessment, guarding against discrimination and ensuring equitable predictions across diverse user groups. Robustness testing involves subjecting the model to various scenarios and monitoring its stability against perturbations. Transparent documentation of model architecture and the QA process provides a foundation for accountability and future improvements. Continuous deployment monitoring, user feedback collection, and iterative improvements enable the system to adapt to changing conditions and evolving requirements. Additionally, compliance with regulations and ethical considerations is paramount to address privacy concerns and uphold ethical standards in AI development and deployment. In summary, a holistic QA approach is crucial for the success of a customer churn prediction system, ensuring it not only meets but exceeds performance expectations while aligning with ethical and regulatory guidelines.

Human Resource Management

In the execution of human resource management for customer churn prediction system, the project team focuses on optimizing the utilization of human resources. This involves defining roles and responsibilities, creating a staffing plan, and recruiting qualified individuals with the necessary skills for AI development and project management. Once the team is assembled, effective leadership and communication are essential for fostering a collaborative and productive work environment. Human resource management also includes performance monitoring, addressing team dynamics, and providing ongoing training and development opportunities to enhance skill sets and adapt to evolving project requirements. Regular feedback mechanisms and performance assessments contribute to the continuous improvement of the team's effectiveness in executing the customer churn prediction system.

Managing Communication

Managing communications for a customer churn prediction system involves a systematic approach to ensure effective and transparent information flow among project stakeholders. First, it is crucial to identify and categorize all relevant stakeholders, including team members, executives, and end-users. Subsequently, the development of a comprehensive communication plan is essential, outlining the frequency, methods, and content of communications throughout the project lifecycle. This plan should include regular status meetings, selected communication channels and tools, and a clear definition of roles and responsibilities within the team. Documentation plays a key role in maintaining transparency and continuity, involving detailed records of project plans, requirements, and design decisions. Regular status meetings provide a platform for team members to share updates, discuss challenges, and plan for upcoming milestones. Additionally, a well-defined protocol for communicating risks and issues, along with an escalation process, ensures prompt and effective resolution. User training and support communication plans should be established, along with mechanisms for gathering feedback from stakeholders. Adaptability and flexibility are critical, allowing communication strategies to evolve based on project needs and stakeholder feedback. Finally, celebrating achievements contributes to a positive working atmosphere and team morale, reinforcing successful collaboration throughout the project.

Conducting Procurements

Conducting procurements for a customer churn prediction system involves a systematic approach to acquiring necessary goods and services. Initially, the project team identifies the required resources, such as AI software, hardware, or external expertise. Subsequently, a procurement plan is developed, outlining the procurement strategy, selection criteria, and evaluation process. This may involve issuing requests for proposals (RFPs) or quotes (RFQs) and conducting a thorough vendor selection process based on factors like cost, quality, and expertise. Once vendors are selected, contracts are negotiated and finalized, and ongoing supplier performance is monitored to ensure adherence to the agreed-upon terms.

Managing Stakeholder Engagements

Managing stakeholder engagements is crucial for the success of the customer churn prediction system. This process begins with the identification of all stakeholders, including internal team members, executives, end-users, and external partners. Stakeholder engagement plans are then developed, outlining communication strategies, engagement levels, and methods for addressing concerns or feedback. Regular communication, feedback sessions, and stakeholder meetings help maintain alignment with project goals and expectations. Stakeholder engagement is an iterative process, requiring ongoing adaptation to evolving project needs and feedback.

3) Monitoring and Controlling Process for Customer Churn Prediction System

Monitoring and controlling processes are critical components of project management for a customer churn prediction system, ensuring that the project stays on track, objectives are met, and deviations are promptly addressed. The following sequential steps outline how these processes can be effectively implemented:

1) **Establish Key Performance Indicators (KPIs):** Begin by defining key performance indicators that align with the project objectives. In the context of a customer churn prediction system, KPIs may include model accuracy, system performance, and adherence to project timelines.

2) **Baseline Assessment:** Conduct a baseline assessment to establish a starting point for comparison. This involves evaluating the initial state of the project, including the status of deliverables, resource allocation, and overall project performance.
3) **Regular Progress Monitoring:** Implement a regular monitoring schedule to track progress against the established KPIs. This may involve weekly or bi-weekly status updates, performance reviews, and milestone evaluations.
4) **Deviation Analysis:** Analyze any deviations from the project plan, comparing actual progress to the initial projections. Identify the root causes of deviations, whether they relate to resource constraints, unforeseen challenges, or changes in project scope.
5) **Risk Management:** Continuously assess and update the project's risk management plan. Identify new risks that may impact progress and implement mitigation strategies to proactively address potential issues.
6) **Communication and Reporting:** Maintain open lines of communication with stakeholders. Regularly report project status, highlighting achievements, challenges, and any necessary adjustments to the plan. Transparent communication ensures that all stakeholders are informed and aligned.
7) **Corrective Action:** If deviations are identified, take corrective action promptly. This may involve reallocating resources, adjusting timelines, or revisiting project priorities to realign with the original plan.
8) **Change Control Process:** Implement a change control process to manage any changes in project scope or objectives. Evaluate the impact of changes on the project plan, budget, and timeline before approving and incorporating modifications.
9) **Performance Metrics Review:** Periodically review and reassess the chosen performance metrics to ensure they remain relevant and aligned with project objectives. Adjust KPIs if necessary to reflect changes in project priorities or stakeholder expectations.
10) **Continuous Improvement:** Emphasize a culture of continuous improvement. Encourage feedback from team members and stakeholders to identify opportunities for enhancing processes, increasing efficiency, and optimizing project outcomes.
11) **Documentation and Lessons Learned:** Maintain comprehensive documentation of all monitoring and controlling activities. Capture lessons learned and best practices to inform future projects and improve the overall project management process.

4) Closing Process for Customer Churn Prediction System

<u>Closing Project</u>

The closing process for Customer Churn Prediction System ends with the final report that completely explains about the system along with the project management processes and the final product that accurately predicts the customer who likely to churn using ANN.

<u>Closing Procurements</u>

Closing procurements for the customer churn prediction system involves a series of well-defined steps to conclude the procurement phase successfully.The initial phase includes the verification of deliverables, where project teams ensure that all procured goods and services meet the specified requirements outlined in the contracts.Following this, a thorough inspection and acceptance process is conducted, involving collaboration with technical experts and end-users to confirm alignment with system requirements. The documentation review is paramount, encompassing a comprehensive assessment of all relevant procurement paperwork, such as contracts, invoices, and acceptance records, to ensure completeness and compliance with legal and organizational standards.

Financial settlements must be finalized, addressing any outstanding matters such as payments and discrepancies in invoices to meet the financial obligations outlined in the procurement contracts. Subsequently, the formal closure of procurement contracts is executed, involving obtaining necessary approvals and signatures. A holistic evaluation of supplier performance follows, encompassing aspects such as adherence to contractual terms, deliverable quality, timeliness, and overall cooperation. The release of retainage, if applicable, is managed upon successful completion and acceptance of deliverables.

As part of the closure process, careful planning for the transition of responsibilities and handover of assets or knowledge from the procurement phase to the ongoing project or operational teams is essential. A lessons-learned session is conducted to capture insights from the procurement process, highlighting successes and areas for improvement that inform future procurement activities and projects. All procurement-related documents and records are meticulously archived for future reference or audits.

Effective communication is integral to the closure process, involving the dissemination of information regarding the successful closure of procurements to relevant stakeholders, including project team members, executives, and suppliers.This communication includes a summary of outcomes, lessons learned, and any future implications. If the procurement closure aligns with the overall project closeout, it ensures a comprehensive conclusion to all aspects of the project, incorporating final reporting, stakeholder communications, and the archiving of project materials.By following these sequential steps, project managers can systematically conclude procurements for the customer churn prediction system, ensuring a smooth transition and valuable insights for future endeavors.

7.0 FLOWCHART & NEURAL LEARNING PROCESS

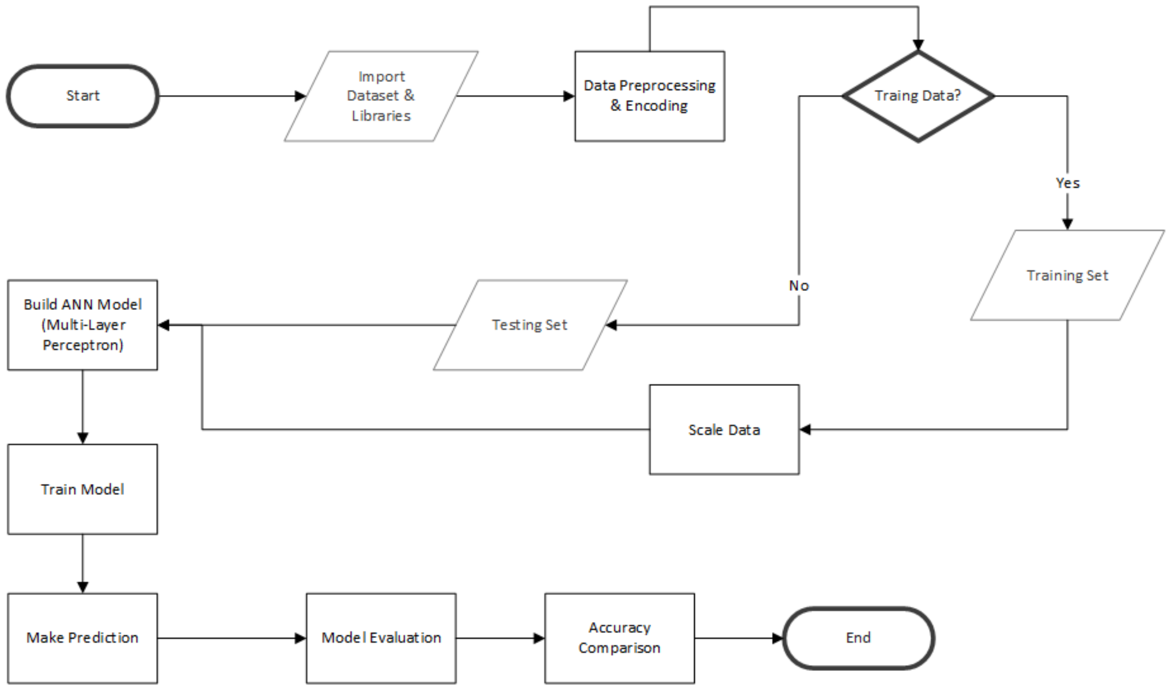


Figure 5.1: Flowchart of Churn Prediction implementation

The code implementation begins by importing the necessary libraries, including \`numpy\`, \`pandas\`, \`tensorflow\`, \`matplotlib\`, \`seaborn\`, and modules from \`sklearn\` (scikit-learn). These libraries provide various functions and tools for data manipulation, model building, and evaluation.

Next, the dataset is loaded using the \`pd.read_csv\` function from pandas. The input features are stored in the variable \`X\`, and the target variable (labels) is stored in the variable \`Y\`. The dataset is assumed to be in a CSV file named "Churn_Modelling.csv".

Data preprocessing steps are performed on the loaded data. This includes label encoding and one-hot encoding to convert categorical data into numerical format. The \`ColumnTransformer\` is used to apply transformations specifically on selected columns, such as one-hot encoding. Additionally, the input features are standardised using the \`StandardScaler\` to ensure they have zero mean and unit variance.

The Artificial Neural Network (ANN) model is built using the \`build_ann_model\` function. This function creates a sequential model using \`tf.keras.models.Sequential\` and defines three layers: two hidden layers with a specified number of neurons and an output layer with a single neuron. The hidden layers use the ReLU activation function, while the output layer uses the sigmoid activation function. The model is compiled with the Adam optimizer, binary cross-entropy loss function, and accuracy metric.

The code utilises a multi-layer perceptron (MLP) or feedforward neural network as a type of perceptron. In an MLP, neurons are arranged in multiple layers, where each neuron in a layer is connected to all neurons in the subsequent layer. This sequential connection allows the network to capture complex relationships between the input data and the target variable.

To construct the MLP model, the code employs the 'Sequential' class from the 'tf.keras.models' module. This class enables the creation of a neural network by stacking layers in a sequential manner. The model architecture is built by adding layers to the model using the 'add' method.

The MLP model in the code consists of three layers. The first layer is the hidden layer, created using the 'Dense' class from 'tf.keras.layers'. The number of neurons in this layer is determined by the 'units' parameter. The activation function applied to this layer is the Rectified Linear Unit (ReLU), which introduces non-linearity into the model, enabling it to learn complex patterns and relationships in the data. Similarly, the second hidden layer is added to the model, also employing the 'Dense' class with the ReLU activation function. These hidden layers contribute to the model's ability to extract meaningful features from the input data.

The final layer added to the model is the output layer. It utilises the 'Dense' class with a single neuron, suitable for binary classification problems like customer churn prediction. The activation function used in this layer is the sigmoid function. By applying the sigmoid function, the output is squashed into the range of [0, 1], representing the probability of the target variable being 1, which in this case denotes the likelihood of customer churn.

The model is then trained and evaluated. The code iterates over different values of `num_neurons` (5 and 10 in this case) and performs the following steps for each value: building the ANN model, training it on the training data (`X_train` and `Y_train`), making predictions on the test set (`X_test`), converting the predictions to binary values based on a threshold of 0.5, calculating the accuracy by comparing the predicted values with the actual values (`Y_test`), and displaying the predictions, accuracy, confusion matrix, and classification report. The accuracy values for each model are stored in the `accuracies` list for later comparison. Finally, the code prints the accuracies for each value of `num_neurons` to compare their performance.

In summary, the code performs data preprocessing and builds an Artificial Neural Network (ANN) model using the Multi-Layer Perceptron (MLP) architecture. It then trains and evaluates the model using different numbers of neurons in the hidden layers, comparing their accuracies. The MLP architecture enables the model to capture complex relationships between the input data and the target variable. The ReLU activation functions in the hidden layers introduce non-linearity, while the sigmoid activation function in the output layer represents the probability of customer churn. By leveraging this architecture and activation functions, the model learns and processes the necessary information for accurate customer churn prediction. The code utilises appropriate libraries and modules for data manipulation, model building, and evaluation tasks, facilitating efficient implementation.

8.0 STEP BY STEP LEARNING & ANALYSIS

We mainly use 2 methods which are Sigmoid Activation Function & ReLU (Rectified Linear Unit) Activation Function.

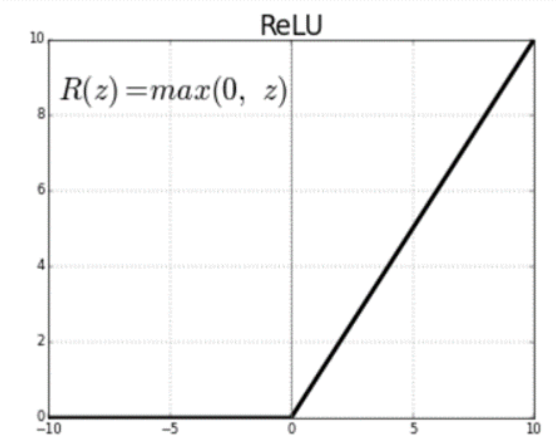


Figure 6.1: ReLU Activation Function

The ReLU activation function is defined as f(x) = max(0, x), where x is the input to the function. It is a simple and computationally efficient activation function that introduces non-linearity to the neural network. When the input to the ReLU function is positive (x > 0), the function returns the input value itself. In other words, the output of the ReLU function is equal to the input value. This means that positive values are preserved and passed through unchanged. On the other hand, when the input to the ReLU function is negative (x <= 0), the function returns 0. In this case, the output of the ReLU function is set to 0, effectively "turning off" the neuron. This non-linear behaviour of the ReLU function allows the neural network to learn complex patterns and make non-linear transformations of the input data.

The ReLU activation function offers several advantages in artificial neural networks. One of the key advantages is that it helps to avoid the vanishing gradient problem that can occur with other activation functions, such as the sigmoid function. The vanishing gradient problem arises when the gradients become extremely small during backpropagation, making it difficult for the network to learn. However, the ReLU function does not saturate for positive inputs, meaning that the gradients remain large and allow for efficient gradient-based optimization during training. This property of the ReLU function helps to address the vanishing gradient problem and facilitates better learning in deep neural networks.

Another advantage of the ReLU activation function is its computational efficiency. The ReLU function is a simple thresholding operation that involves comparing the input value to zero and returning the input value if it is positive, or zero if it is negative. This simplicity makes the ReLU function computationally efficient to compute, especially when dealing with large-scale neural networks with a high number of neurons and layers.

Additionally, the ReLU activation function provides sparsity in activation. By setting negative values to zero, the ReLU function introduces sparsity in the network's activation patterns. This means that a significant portion of the neurons in the network will be inactive (outputting zero) for certain inputs, resulting in a more sparse representation. This sparsity can have benefits in terms of memory usage and computational efficiency, as it reduces the number of computations required during forward and backward passes.

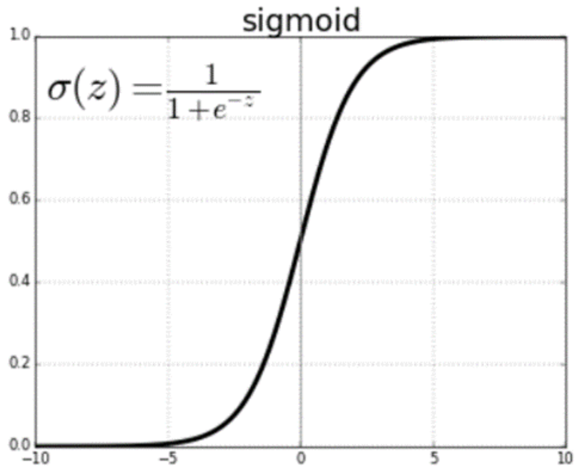


Figure 6.2: Sigmoid Activation Function

The sigmoid activation function is defined as f(x) = 1 / (1 + exp(-x)), where x is the input to the function. It is a smooth, S-shaped function that maps any input value to a value between 0 and 1. This makes it useful for binary classification tasks, where the output of the neural network needs to be interpreted as a probability of belonging to a certain class. When the input to the sigmoid function is positive (x > 0), the function returns a value greater than 0.5, which can be interpreted as a positive prediction or a probability of belonging to the positive class.

On the other hand, when the input to the sigmoid function is negative (x < 0), the function returns a value less than 0.5, which can be interpreted as a negative prediction or a probability of belonging to the negative class. The sigmoid function is differentiable, which allows for gradient-based optimization during training. This means that the weights and biases of the neural network can be adjusted using backpropagation to minimise the error between the predicted output and the actual output.

The sigmoid activation function is a commonly used activation function in artificial neural networks. One of the advantages of the sigmoid function is that it squashes the output to the range [0, 1], which can be interpreted as a probability. This makes it particularly useful for binary classification tasks, where the output of the neural network needs to be interpreted as a probability of belonging to a certain class. Additionally, the sigmoid function is differentiable, which allows for gradient-based optimization during training. This means that the weights and biases of the neural network can be adjusted using backpropagation to minimise the error between the predicted output and the actual output.

However, the sigmoid function also has some limitations. One of the main limitations is that it can suffer from the vanishing gradient problem, which can impact the training process in deep neural networks. The vanishing gradient problem occurs when the gradient of the sigmoid function becomes very small for large input values, making it difficult for the neural network to learn. Another limitation of the sigmoid function is that it is prone to saturation for large input values. When the sigmoid function saturates, the gradient becomes very small, leading to slow learning. These limitations can make it challenging to use the sigmoid function in deep neural networks, where the input values can be very large and the network can have many layers.

9.0 HOW ANALYSIS IS CONDUCTED & ITS INTERPRETATION

Firstly, the necessary libraries such as numpy, pandas, tensorflow, matplotlib, seaborn, and scikit-learn are imported. These libraries provide essential functions and tools for data manipulation, model building, and evaluation.

```
import numpy as np
import pandas as pd
import tensorflow as tf
```

Figure 7.1: Import Libraries

The dataset is then loaded from the CSV file named "Churn_Modelling.csv" using the function from the pandas library. The next line extracts the input features from the dataset. It selects all rows and columns starting from the 3rd column up to the second-to-last column . The selected features are then converted to a NumPy array using the **values** attribute.

```
data = pd.read_csv("Churn_Modelling.csv")
X = data.iloc[:, 3:-1].values  # credit score -> estimated salary
```

Figure 7.2: Loading Dataset

In this part of the code, there are two data preprocessing steps being performed to prepare the input features (**x**) for the customer churn prediction model. The first step is label encoding. Label encoding is applied to convert categorical data into numerical format.

The second step is one-hot encoding. One-hot encoding is used to represent categorical data as binary vectors. It helps capture the categorical nature of the data without introducing any ordinal relationship between the categories.

```
LE1 = LabelEncoder()
X[:, 2] = np.array(LE1.fit_transform(X[:, 2]))

ct = ColumnTransformer(transformers=[('encoder', OneHotEncoder(), [1])], remainder="passthrough")
X = np.array(ct.fit_transform(X))
```

Figure 7.3: Preprocessing Dataset

After that**,** the input features (**X**) and target variable (**Y**) are split into training and testing sets using the **train_test_split** function from **scikit-learn**. The purpose of splitting the data is to have separate subsets for training and evaluating the model's performance.

Next, the **StandardScaler** object is created to perform standardisation on the input features. Standardisation is a common preprocessing technique that transforms the data to have zero mean and unit variance. It helps to bring the features to a similar scale and prevents certain features from dominating the model's learning process.

```
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=0)

sc = StandardScaler()
X_train = sc.fit_transform(X_train)
X_test = sc.transform(X_test)
```

Figure 7.4: Train-Test Split Dataset

Then, we implemented both ReLU & Sigmoid Activation Function into Google Colab to build and compile the ANN model. In our coding, the ReLU activation function acts as the 1st & 2nd layer is applied to the hidden layers of the ANN model using the **tf.keras.layers.Dense** class with **activation="relu"**.

```
ann.add(tf.keras.layers.Dense(units=units, activation="relu"))  # Layer 1
ann.add(tf.keras.layers.Dense(units=units, activation="relu"))  # Layer 2
```

Figure 7.5: ReLU Activation

Then, the sigmoid activation function acts as the 3rd layer of neuron is applied to the output layer of the ANN model using the **tf.keras.layers.Dense** class with **activation="sigmoid"**.

```
ann.add(tf.keras.layers.Dense(units=1, activation="sigmoid"))  # Layer 3
```

Figure 7.6: Sigmoid Activation

During the training of the ANN model, the specified activation functions are used to introduce non-linearity and capture complex relationships between the input features and the target variable. The model is trained using the Adam optimizer and the binary cross-entropy loss function, as specified in the **ann.compile** function.

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

Figure 7.7: Adam & Binary Cross-Entropy

After training the model, predictions are obtained using the trained model on the test dataset. The predictions are thresholded at 0.5 using **(predictions > 0.5)** to convert them into binary values (0 or 1) based on the sigmoid activation. The accuracy of the predictions is computed using the **accuracy_score** function from scikit-learn.

```
predictions = ann.predict(X_test)
predictions = (predictions > 0.5)
print(f"Predictions (Neurons={num_neurons}):\n{predictions}")
accuracy = accuracy_score(Y_test, predictions)
accuracies.append(accuracy)
print(f"Accuracy (Neurons={num_neurons}): {accuracy}")
```

Figure 7.8: Prediction & accuracy_score

Finally, a confusion matrix is created using the confusion_matrix function, and it is visualised using **matplotlib** and **seaborn** libraries to analyse the performance of the model.

```
import matplotlib.pyplot as plt
import seaborn as sns
```

Figure 7.9: External libraries for visualisation

The code then repeats the process for different numbers of neurons in the hidden layers (5 and 10), and the accuracies are stored in the **accuracies** list. The accuracies are printed and compared at the end of the code.

```
neurons = [5, 10]
accuracies = []
```

Figure 7.10: Hidden layer neurons

10.0 RESULT FINDINGS

The figure below shows the prediction of the 5 Neurons in Hidden Layers. All the predictions are outputted as false after 100 epochs.

```
Predictions (Neurons=5):
[[False]
 [False]
 [False]
 ...
 [False]
 [False]
 [False]]
```

Figure 8.1: Prediction (Neurons = 5)

The Classification Report for 5 Neurons in Hidden Layers is shown in Figure 8.2. Regarding Class 0, With an accuracy of 0.89, 89% of the examples that were predicted to be non-churning (class 0) were in fact right. With a recall of 0.93, it can be inferred that 93% of the occurrences that truly fall into the "not churn" class were properly recognised by the model. An accurate measurement that is balanced is given by the F1-score of 0.91, which is the harmonic mean of precision and recall. The test set contains 1595 instances that have been classified as not likely to churn, according to the support of 1595.Regarding Class 1, the accuracy of 0.69 indicates that 69% of the cases classified as Class 1's most likely to churn were accurate. With a recall of 0.57, the model successfully recognised 57% of the cases that truly fall into the churn class. The accuracy and recall for class 1 are balanced by the F1-score of 0.62. The support of 405 indicates that there are 405 occurrences in the test set that have been classified as likely to churn.The unweighted mean for each classes is taken into account when computing the macro average of accuracy, recall, and F1-score. In this instance, the macro average accuracy, recall, and F1-score are 0.79, 0.75, and 0.77 respectively. The weighted average of accuracy, recall, and F1-score represents the average while accounting for the variety of classes. In this instance, the macro average accuracy, recall, and F1-score are 0.79, 0.75, and 0.77 respectively. The weighted average of accuracy, recall, and F1-score represents the average while accounting for the variety of classes. In this scenario, the weighted average accuracy, recall, and F1-score are 0.85, showing a balanced performance across both classes.The overall accuracy of the model is 0.86, suggesting that the model successfully identified 86% of the cases in the test set.

```
Accuracy (Neurons=5): 0.86
Classification Report (Neurons=5):
              precision    recall  f1-score   support

           0       0.89      0.93      0.91      1595
           1       0.69      0.57      0.62       405

    accuracy                           0.86      2000
   macro avg       0.79      0.75      0.77      2000
weighted avg       0.85      0.86      0.85      2000
```
Figure 8.2: Classification Report (Neurons = 5)

The figure below shows the prediction of the 10 layer neurons. All the predictions are outputted as false after 100 epochs.

```
Predictions (Neurons=10):
[[False]
 [False]
 [False]
 ...
 [False]
 [False]
 [False]]
```

Figure 8.3: Prediction (Neurons = 10)

The Classification Report for 10 Neurons in Hidden Layers is shown in Figure 8.4. Regarding the Class 0, With an accuracy of 0.89, 89% of the examples that were predicted to be non-churning (class 0) were in fact right. With a recall of 0.94, the model successfully recognised 94% of the occurrences that truly fall into the "not churn" class. For class 0, the F1-score of 0.91 is a balanced indicator of accuracy. The number of test set instances classified as not likely to churn is represented by the support of 1595.Regarding Class 1, the accuracy of 0.70 indicates that 70% of the cases classified as Class 1 were correctly expected to churn.With a recall of 0.54, the model successfully recognised 54% of the cases that truly fall into the churn class. For class 1, the F1-score of 0.61 reflects a compromise between recall and accuracy. The support of 405 indicates that there are 405 occurrences in the test set that have been classified as likely to churn.The macro average accuracy, recall, and F1-score, which is the unweighted mean across both classes, are 0.80, 0.74, and 0.76, respectively.Given the amount of occurrences in each class, the weighted average accuracy, recall, and F1-score are all 0.85, showing a balanced performance.The model's overall accuracy is 0.86, meaning that 86% of the test set's occurrences were properly categorised by the model.

```
Accuracy (Neurons=10): 0.8605
Classification Report (Neurons=10):
              precision    recall  f1-score   support

           0       0.89      0.94      0.91      1595
           1       0.70      0.54      0.61       405

    accuracy                           0.86      2000
   macro avg       0.80      0.74      0.76      2000
weighted avg       0.85      0.86      0.85      2000
```

Figure 8.4: Classification Report (Neurons = 10)

For the final accuracy comparison, we can see the 10 Neurons in Hidden Layer has a bit higher of accuracy score approximately about 0.0005 than 5 Neurons in Hidden Layer which is 0.86 while the latter is 0.8605.In conclusion, the model exhibits respectable performance in predicting class 0 (not churn), with high accuracy, recall, and F1-score. This is shown for the classification report of 5 neurons (Figure 8.2). However, as seen by reduced accuracy, recall, and F1-score when compared to class 0, the model finds it difficult to forecast class 1 (churn). The weighted average F1-score, with an accuracy of 86%, points to a balanced overall performance of the model.On the other hand, the model performs better than 5 neurons in predicting both class 0 and class 1 for the classification report of 10 neurons (Figure 8.4). Although they have improved much from class 0, class 1's accuracy, recall, and F1-score remain below those of class 0. The weighted average F1-score indicates an accurate 86% overall performance, comparable to the model with 5 neurons. As a result, when compared to the model with 5 neurons, the model with 10 neurons performs somewhat better overall, especially when predicting class 1 (churn).

```
Comparison
Accuracy (Neurons=5): 0.86
Accuracy (Neurons=10): 0.8605
```

Figure 8.5: Final Accuracy Comparison (n=5 vs n=10)

For the confusion matrices below, we can see that the values of True Positive, False Positive, False Negative & True Negative are different from both [5,5,1] & [10,10,1] layers. The [5,5,1] layer has more values in TP & TN than [10,10,1] layer. However, it has less values for FP & FN than the latter. It means that the [5,5,1] layer has a very good prediction but is very inaccurate as the value of accuracy does not come from prediction alone. The

[10,10,1] layer otherwise might be less predicted but the accuracy is much better due to better recall & F1-score. Hence, [5,5,1] layer is better than [10,10,1] layer based on their confusion matrix.

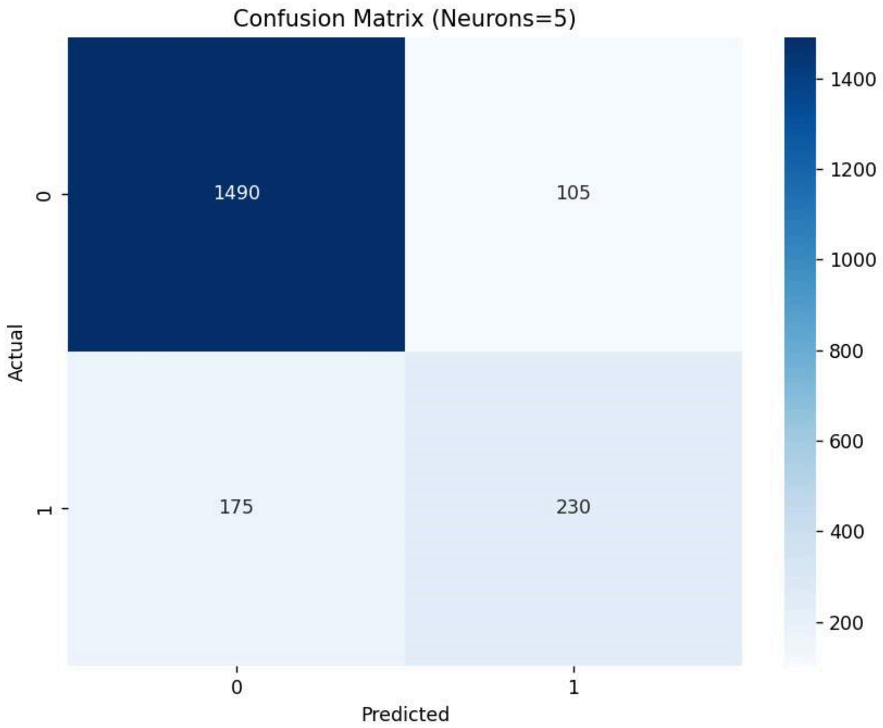


Figure 8.6: Confusion Matrix [5,5,1]

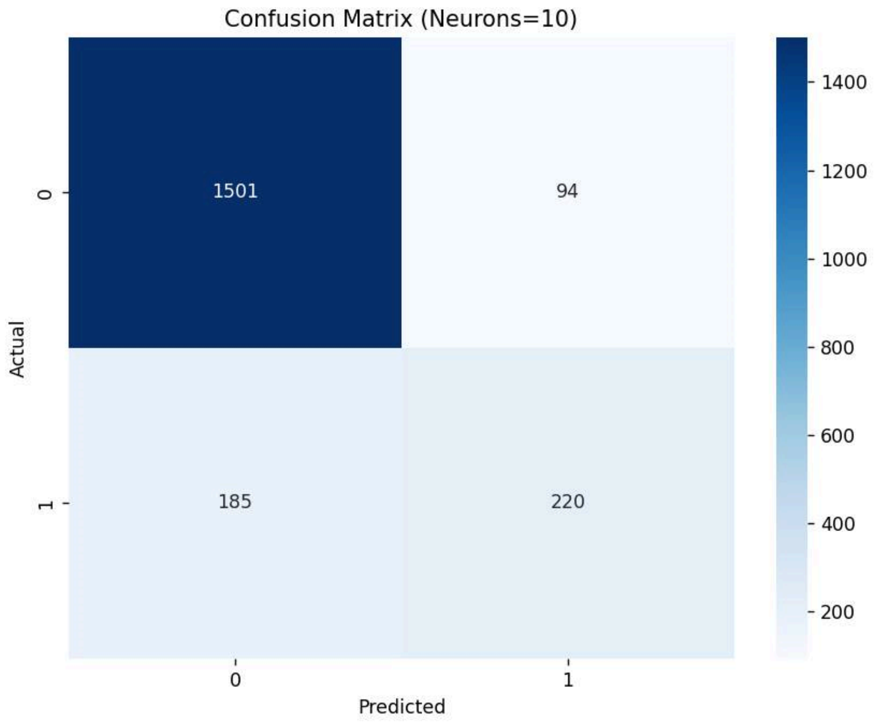


Figure 8.7: Confusion Matrix [10,10,1]

11.0 BUSINESS/INTELLIGENCE MODEL & FUTURE STUDY

Predictive analytics plays a crucial role in customer churn prediction, enabling businesses to forecast and predict which customers are likely to churn in the future. By leveraging historical customer data, including demographics, transaction history, usage patterns, and interactions, churn prediction models provide valuable insights to businesses. Armed with this information, businesses can take proactive measures to retain customers and prevent churn.

One of the key benefits of churn prediction models is their ability to support proactive maintenance planning. In industries where long-term contracts or subscription-based services are prevalent, such as telecommunications or utilities, these models help identify customers at high risk of churn. By proactively addressing customer issues, providing support, or offering incentives, businesses can mitigate the risk of churn and enhance customer satisfaction.

Furthermore, churn prediction models contribute to cost reduction and efficiency by optimising resource allocation. Instead of allocating equal resources to all customers, businesses can focus their retention efforts on high-risk customers. By targeting these customers with personalised offers or improved customer service, businesses can allocate resources effectively and reduce costs associated with customer acquisition.

Optimal resource allocation is another advantage offered by churn prediction models. These models help businesses identify the factors that contribute most to churn, enabling them to prioritise their efforts. By allocating sales and marketing resources to customers who are more likely to churn, optimising retention campaigns, and tailoring retention strategies based on predicted churn probabilities, businesses can allocate their resources efficiently and maximise their impact.

Churn prediction models also contribute to data-driven business intelligence. By analysing customer behaviour, demographics, and interactions, businesses can gain valuable insights into their customer base. These insights enable businesses to understand patterns, identify trends, and make informed decisions to reduce churn and improve customer retention. By leveraging the power of data, businesses can enhance their decision-making processes and drive long-term success.

Looking to the future, there are several potential extensions and advancements that can enhance customer churn prediction. One avenue is the incorporation of additional data sources beyond the existing features in the dataset. By integrating customer support interactions, social media activity, or survey responses, businesses can obtain a more comprehensive view of customer behaviour and improve the accuracy of churn prediction models.

Other feature engineering and selection techniques offer another promising direction for improvement. By exploring new features or deriving meaningful insights from existing ones, businesses can capture more relevant information to predict churn accurately. Techniques such as creating interaction terms, aggregating variables, or applying dimensionality reduction methods can enhance the representation of the data and uncover valuable patterns.

Ensemble methods present an opportunity to leverage the power of multiple models. By combining predictions from various churn prediction models through bagging, boosting, or stacking, businesses can benefit from diverse perspectives and improve the overall accuracy and robustness of the predictions.

Incorporating advanced modelling techniques, including Convolutional Neural Networks (CNNs) or Generative Adversarial Networks (GANs), holds promise for churn prediction. These deep learning models excel at capturing complex patterns and relationships in data, enabling more accurate churn predictions and informing targeted retention strategies.

Real-time prediction and proactive interventions represent an important frontier for churn prediction. Integrating churn prediction models into real-time systems allows businesses to identify customers at risk of churn in near real-time. With timely interventions such as personalised offers, targeted retention strategies, or improved customer service, businesses can proactively prevent churn and enhance customer satisfaction.

12.0 CONCLUSION

In conclusion, customer churn prediction using Artificial Neural Networks (ANN) has proven to be a powerful tool for businesses in forecasting and mitigating customer attrition. By harnessing the capabilities of ANN models, companies can analyse vast amounts of historical data, identify relevant patterns, and make accurate predictions regarding customer churn.

Through the deployment of ANN models, businesses can proactively identify customers who are at risk of churning, enabling them to take appropriate measures to retain these valuable customers. By leveraging customer insights, personalised offers, improved customer service, and tailored retention strategies can be implemented to enhance customer satisfaction and loyalty. This approach not only reduces customer attrition but also contributes to the overall success and profitability of the business.

Moreover, the use of ANN models allows for continuous learning and refinement of predictions. As new data becomes available, the model can be updated and recalibrated to adapt to changing customer behaviours and market dynamics. This iterative process ensures the accuracy and relevance of churn predictions, further empowering businesses to make informed decisions and allocate resources effectively.

REFERENCES

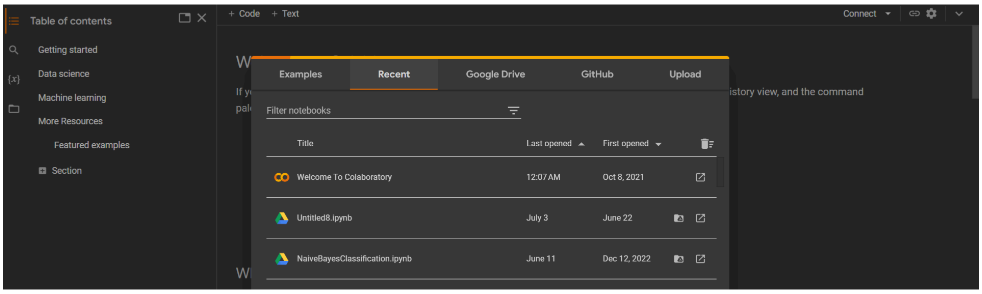
Mohadarkar, S. (2023). Implementing Artificial Neural Network(classification) in python from scratch. Analytics Vidhya. Retrieved from https://www.analyticsvidhya.com/blog/2021/10/implementing-artificial-neural-networkclassification-in-python-from-scratch/

Badole, M. (2023). *Customer churn prediction using Artificial Neural Network*. Analytics Vidhya. Retrieved from https://www.analyticsvidhya.com/blog/2021/10/customer-churn-prediction-using-artificial-neural-network/

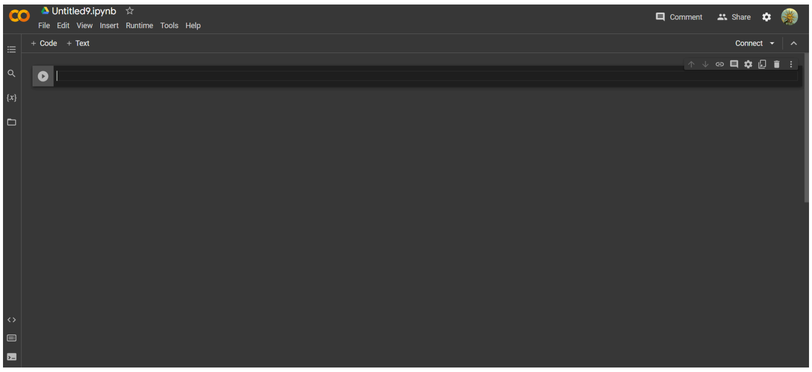
APPENDIX

a) User Manual - detail step by step how to run the program

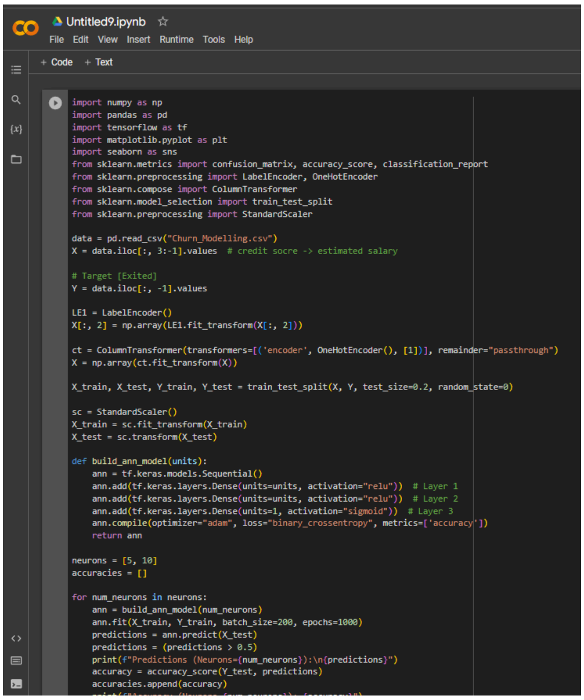
i. Open Google Colab



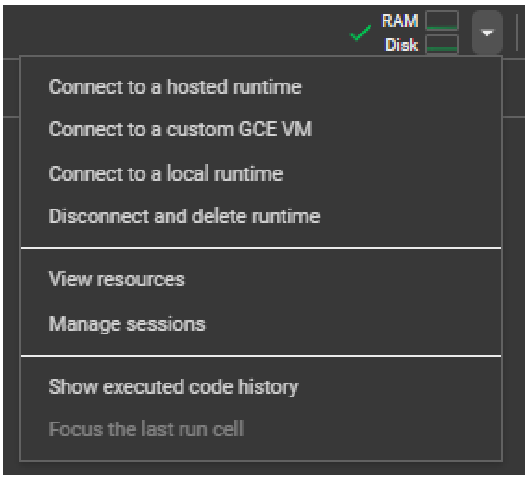
ii. Create new notebook



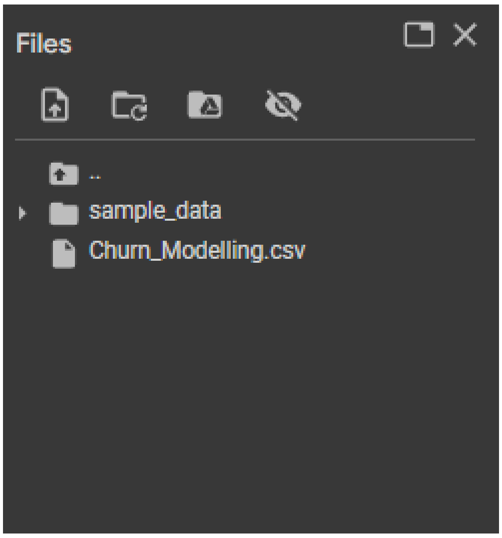
iii. Copy paste the coding from main.py file



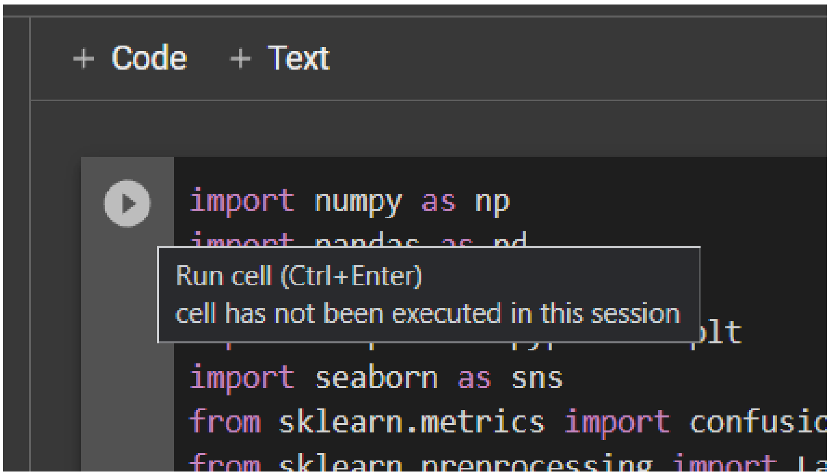
iv. Connect to the server runtime



v. Upload the dataset into Google Colab repository



vi. Run the coding



b) Other materials i.e. journal articles

c) Sample of raw data and dictionaries eg. few records

A	B	C	D	E	F	G	H	I	J	K	L	M	N
RowNumb	Customer	Surname	CreditScore	Geograph	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	Estimated	Exited
1	15634602	Hargrave	619	France	Female	42	2	0	1	1	1	101348.9	1
2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.6	0
3	15619304	Onio	502	France	Female	42	8	159660.8	3	1	0	113931.6	1
4	15701354	Boni	699	France	Female	39	1	0	2	0	0	93826.63	0
5	15737888	Mitchell	850	Spain	Female	43	2	125510.8	1	1	1	79084.1	0
6	15574012	Chu	645	Spain	Male	44	8	113755.8	2	1	0	149756.7	1
7	15592531	Bartlett	822	France	Male	50	7	0	2	1	1	10062.8	0
8	15656148	Obinna	376	Germany	Female	29	4	115046.7	4	1	0	119346.9	1
9	15792365	He	501	France	Male	44	4	142051.1	2	0	1	74940.5	0
10	15592389	H?	684	France	Male	27	2	134603.9	1	1	1	71725.73	0
11	15767821	Bearce	528	France	Male	31	6	102016.7	2	0	0	80181.12	0
12	15737173	Andrews	497	Spain	Male	24	3	0	2	1	0	76390.01	0
13	15632264	Kay	476	France	Female	34	10	0	2	1	0	26260.98	0
14	15691483	Chin	549	France	Female	25	5	0	2	0	0	190857.8	0
15	15600882	Scott	635	Spain	Female	35	7	0	2	1	1	65951.65	0
16	15643966	Goforth	616	Germany	Male	45	3	143129.4	2	0	1	64327.26	0

Figure Appendix Sample of Customer Churn Prediction Data