# Customer Churn Prediction using Artificial Neural Network (ANN)

This project aims to predict the likelihood of customer churn using an Artificial Neural Network (ANN). Customer churn refers to the phenomenon where customers stop using a company's products or services. By predicting churn, businesses can take proactive measures to retain customers and reduce revenue loss.

## How to Use

1. Clone the repository.

2. Install the required dependencies (see below).

3. Run the Streamlit app using the command:

```bash

streamlit run app.py

```

4. Input the customer details in the web interface and get the churn prediction.

## Project Overview

The project involves building and training an ANN model to predict customer churn based on various customer attributes. The model is trained on a dataset containing customer information and their churn status. The trained model is then deployed using Streamlit to create a web application where users can input customer details and get churn predictions.

## Dataset

The dataset used in this project is `Churn\_Modelling.csv`, which contains the following features:

### Independent Features:

- `RowNumber`,`CustomerId`,`Surname`, `CreditScore`, `Geography`, `Gender`, `Age`, `Tenure`, `Balance`, `NumOfProducts`, `HasCrCard`, `IsActiveMember`, `EstimatedSalary`

### Dependent Feature:

- `Exited`

## Data Preprocessing

1. \*\*Dropping Irrelevant Columns\*\*: Columns like `RowNumber`, `CustomerId`, and `Surname` are dropped as they do not contribute to the prediction.

2. \*\*Encoding Categorical Variables\*\*:

- `Gender` is encoded using `LabelEncoder`.

- `Geography` is one-hot encoded using `OneHotEncoder`.

3. \*\*Splitting the Dataset\*\*: The dataset is split into training and testing sets using an 80-20 split.

4. \*\*Scaling the Data\*\*: The features are scaled using `StandardScaler` to normalize the data.

## Model Architecture

The ANN model is built using TensorFlow and Keras with the following architecture:

- \*\*Input Layer\*\*: 64 neurons with ReLU activation.

- \*\*Hidden Layer 1\*\*: 32 neurons with ReLU activation.

- \*\*Output Layer\*\*: 1 neuron with Sigmoid activation (for binary classification).

The model is compiled using the Adam optimizer with a learning rate of 0.01 and binary cross-entropy loss.

## Training the Model

The model is trained for 100 epochs with early stopping to prevent overfitting. TensorBoard is used for logging and visualizing the training process.

## Hyperparameter Tuning

Grid search is performed to find the best hyperparameters for the model. The parameters tuned include the number of neurons, layers, and epochs.

## Prediction

The trained model is used to predict the churn probability for new customer data. The prediction is made by preprocessing the input data (encoding and scaling) and then passing it through the model.

## Deployment

The model is deployed using Streamlit to create a web application. Users can input customer details, and the application will predict the likelihood of churn.

## Dependencies

- Python 3.x, TensorFlow, Keras, Scikit-learn, Pandas, NumPy, Streamlit, Pickle

## Conclusion

This project demonstrates the use of an Artificial Neural Network to predict customer churn. By leveraging machine learning, businesses can identify at-risk customers and take proactive measures to retain them, ultimately reducing churn and increasing revenue. The deployment of the model using Streamlit makes it accessible and easy to use for end-users.