**Loan Risk prediction using Neural network**

**Under DBT Star College Programme**

**Project-1 Submitted in partial fulfilment for the award of**

**B.Sc. Degree in Computer Science**

**Madurai Kamaraj University, Madurai.**

**NAME : P.ANBU SELVAN**

**ROLL.NO : 22AUCS018**

**CLASS : III-B.Sc. COMPUTER SCIENCE**

**SUBMITTED ON : --/02/2025**

**PROJECT GUIDE : Dr.T.KATHIRVALAVAKUMAR**

****

**Research Center in Computer Science**

**V.H.N.SENTHIKUMARA NADAR COLLEGE(Autonomous)**

**Virudhunagar.**

**Loan Risk prediction using Neural network**

**Under DBT Star College Programme**

**Project-1 Submitted in partial fulfilment for the award of**

**B.Sc. Degree in Computer Science**

**Madurai Kamaraj University, Madurai.**

**NAME : P.ANBU SELVAN**

**ROLL.NO : 22AUCS018**

**CLASS : III-B.Sc. COMPUTER SCIENCE**

**SUBMITTED ON : --/02/2025**

**PROJECT GUIDE : Dr.T.KATHIRVALAVAKUMAR**

****

**Project Guide Signature HOD Signature**

**Research Center in Computer Science**

**V.H.N.SENTHIKUMARA NADAR COLLEGE(Autonomous)**

**Virudhunagar.**

**Loan Risk prediction using Neural network**

## **Objectives**

Objective of this project is to build a neural network-based model to predict loan approval status based on various features such as age, income, loan amount, employment length, and others. The model will predict whether an individual will default on the loan.

The project involves:

* Data preprocessing, including encoding categorical values and normalization.
* Implementing a neural network for classification.
* Evaluating the performance of the model based on accuracy.
* Allowing user input for real-time prediction.

## **Problem Description**

The goal of this project is to develop a predictive model that determines whether an individual will default on a loan based on historical loan data.

### **Dataset Overview**

* **Total Records:** 8,145
* **Target Variable:** Status (0 or 1)
* **Features Used for Prediction**

| **Feature** | **Description** |
| --- | --- |
| **Age** | The age of the individual applying for the loan. |
| **Income** | The annual income of the individual. |
| **Home Ownership** | Whether the individual owns, rents, or has a mortgage. |
| **Loan Amount** | The requested loan amount. |
| **Employment Length** | Number of years the individual has been employed. |
| **Intent** | The purpose of the loan (e.g., medical, personal, home improvement). |
| **Rate** | The interest rate associated with the loan. |
| **Percent Income** | The proportion of income allocated to loan payments. |
| **Credit Length** | The number of years the individual has had credit history. |
| **Default** | Whether the individual has previously defaulted on a loan. |

## **Procedure**

### **Data Preprocessing**

* **Data Cleaning**: Missing values are filled with the mean of each feature to prevent data loss.
* **Encoding Categorical Variables**: Categorical variables (e.g., Home, Intent, Default) are converted into numerical values using mappings.
* **Normalization**: Continuous features like Age, Income, and Amount are normalized using min-max normalization to scale the data between 0 and 1.

### **Model Architecture**

* **Neural Network Architecture**: The model consists of three layers:
  1. **Input Layer**: Accepts features with a bias term added.
  2. **Hidden Layers**: Two hidden layers with sigmoid activations.
  3. **Output Layer**: A single output node that uses a sigmoid activation to predict a value between 0 and 1, indicating the likelihood of loan default.
* **Activation Function**: Sigmoid activation function is used for both hidden layers and the output layer.
* **Backpropagation**: The error is propagated backward from the output layer to the input layer to update the weights.
* **Loss Function**: Mean Squared Error (MSE) is used to calculate the loss during training.

### **Model Training**

* **Training and Testing Split**: The dataset is split into training (80%) and testing (20%) subsets.
* **Learning Rate**: A learning rate of 0.01 is used during the training process.
* **Epochs**: The model is trained for 500 epochs to minimize the loss.

### **Prediction**

* After training, the model can predict whether an individual will default on the loan based on new input data provided by the user.

### **Libraries Used**

Pandas,NumPy,Random,Math.

**Experiment Results**

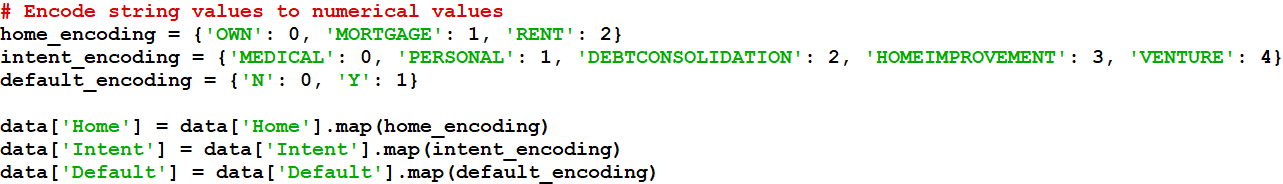
| **Case1:**  Number of hidden layer:1  Learning rate:0.1  Number of Cycles (Epochs):500  Total Neurons:10+10+1  Accuracy:**69.7%** | **Case2:**  Number of hidden layer:1  Learning rate:0.01  Number of Cycles (Epochs):500  Total Neurons:10+10+1  Accuracy:**72.5%** |
| --- | --- |
| **Case3:**  Number of hidden layer:1  Learning rate:0.01  Number of Cycles (Epochs):1000  Total Neurons:10+10+1  Accuracy:**73.9%** | **Case4:**  Number of hidden layer:2  Learning rate:0.01  Number of Cycles (Epochs):500  Total Neurons:10+20+1  Accuracy:**74.2%** |
| **Case5:**  Number of hidden layer:2  Learning rate:0.1  Number of Cycles (Epochs):1000  Total Neurons:10+20+1  Accuracy:**70.7%** | **Case6:**  Number of hidden layer:2  Learning rate:0.001  Number of Cycles (Epochs):500  Total Neurons:10+20+1  Accuracy:**65.4%** |
| **Case7:**  Number of hidden layer:2  Learning rate:0.01  Number of Cycles (Epochs):500  Total Neurons:10+20+1  Accuracy:**77.8%** | **Case8:**  Number of hidden layer:2  Learning rate:0.01  Number of Cycles (Epochs):500  Total Neurons:10+30+1  Accuracy:**84.94%** |

### **Observations & Insights:**

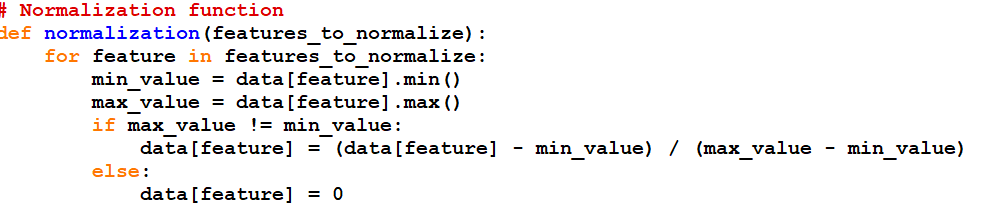
* Increasing the number of neurons and hidden layers generally improved accuracy.
* Lower learning rates (0.01) performed better in most cases than higher learning rates (0.1 or 0.001).
* The best performance was observed in Case 8 with 84.94% accuracy using 2 hidden layers and 10+30+1 neurons.
* Higher cycles (e.g., 1000) did not always guarantee better performance.

### **Major function used**

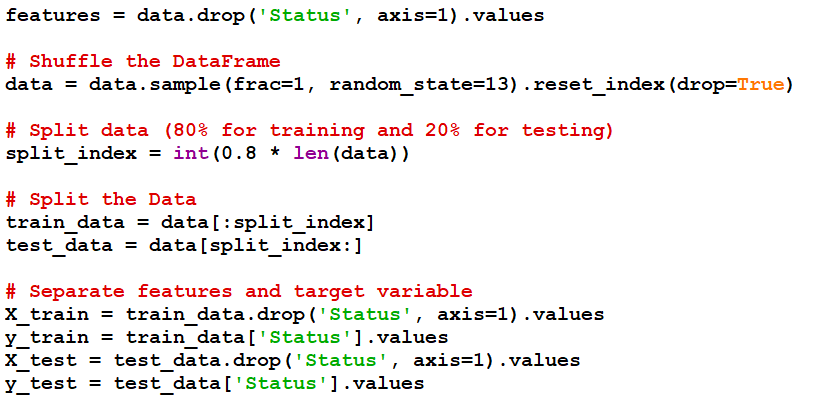
Encoding



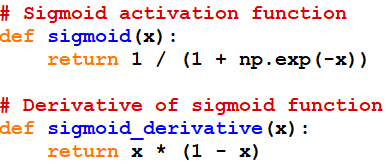
### Normalization



### Splitting the data

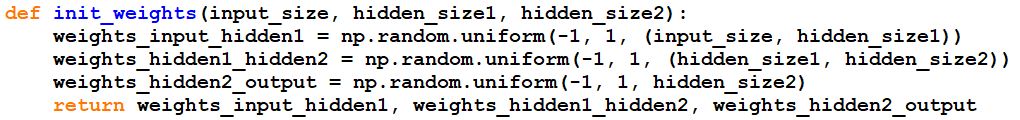


### Activation function

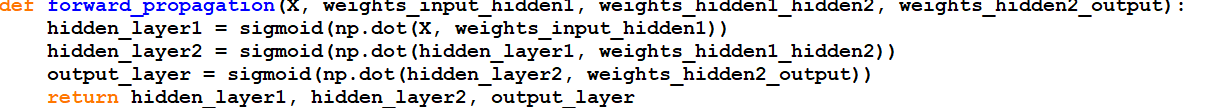


### 

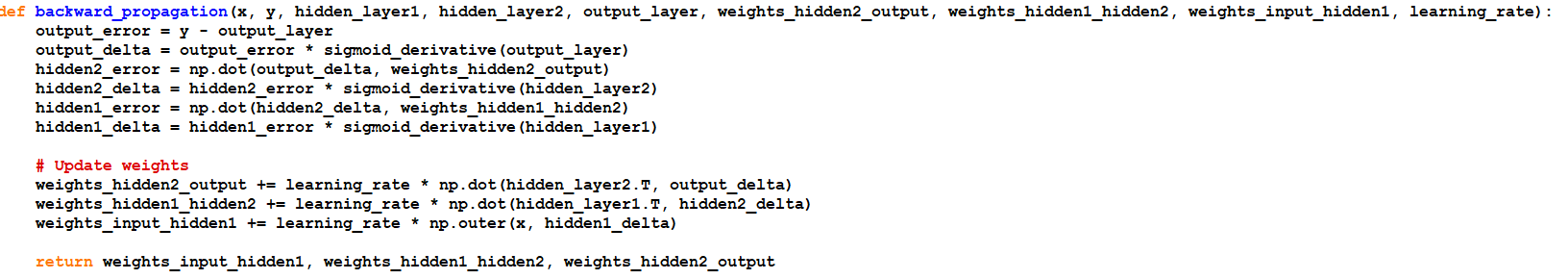
### Initialize weights



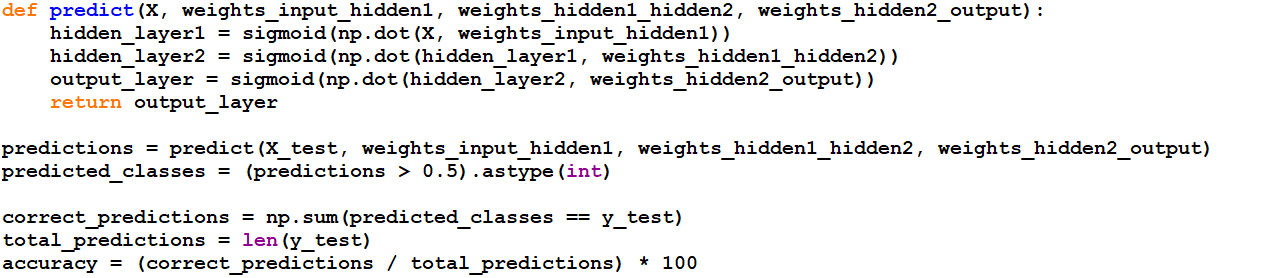
### Forward propagation



### Backward propagation



### Prediction



### **Conclusion**

This project successfully implemented a neural network-based model to predict loan default using key financial and demographic features. Through multiple experiments, we analyzed the impact of different hyperparameters, including the number of hidden layers, learning rate, number of neurons, and training cycles, on model performance.