

Task 2 Documentation

Team Members:

الاسم	رقم الجلوس	رقم السكشن
سامي سامر فايز	20201700340	4
فاطمة عوض إسماعيل	20201700570	6
أيمن محمد أحمد	20201700168	2
يوسف أحمد عبداللطيف	20201700994	

Overview

This Python script implements a neural network from scratch for classification tasks using NumPy and Pandas libraries. The neural network performs classification on a dataset and evaluates its performance using accuracy metrics and a confusion matrix.

Dependencies

- NumPy: For numerical operations and handling matrices.
- Pandas: For data manipulation and preprocessing.
- Matplotlib & Seaborn: For data visualization.
- sys: For system-specific parameters and functions.

Modules and Functions

1. Data Preprocessing (preprocess)

- Reads the dataset from a CSV file, performs data cleaning by filling missing values and scaling features using Min-Max scaling.
- Splits the data into training and testing sets.

2. Activation Functions

- sigmoidFunc: Implements the sigmoid activation function.
- derivative_sigmoid: Computes the derivative of the sigmoid function.
- tanh: Implements the hyperbolic tangent (tanh) activation function.
- tanh_derivative: Computes the derivative of the tanh function.

3. Weight Initialization

- initialize_weights: Initializes weights for the neural network.

4. Forward Propagation

- activate_input: Computes neuron activation for input data.
- activate_hidden: Computes neuron activation for hidden layers.

5. Backpropagation

- backprop_out: Performs backpropagation for the output layer.
- backprop_hidden: Performs backpropagation for hidden layers.

6. Weight Update

- update_weights: Updates weights based on backpropagation and learning rate.

7. Training the Neural Network

- train: Trains the neural network using forward and backward propagation.

8. Model Evaluation

Training Accuracy

- train_acc: Calculates the training accuracy of the model and the confusion matrix.

Results:

The training accuracy = 0.9 (with bias, learning rate = 0.01, and epochs =1000)

The test accuracy =0.7

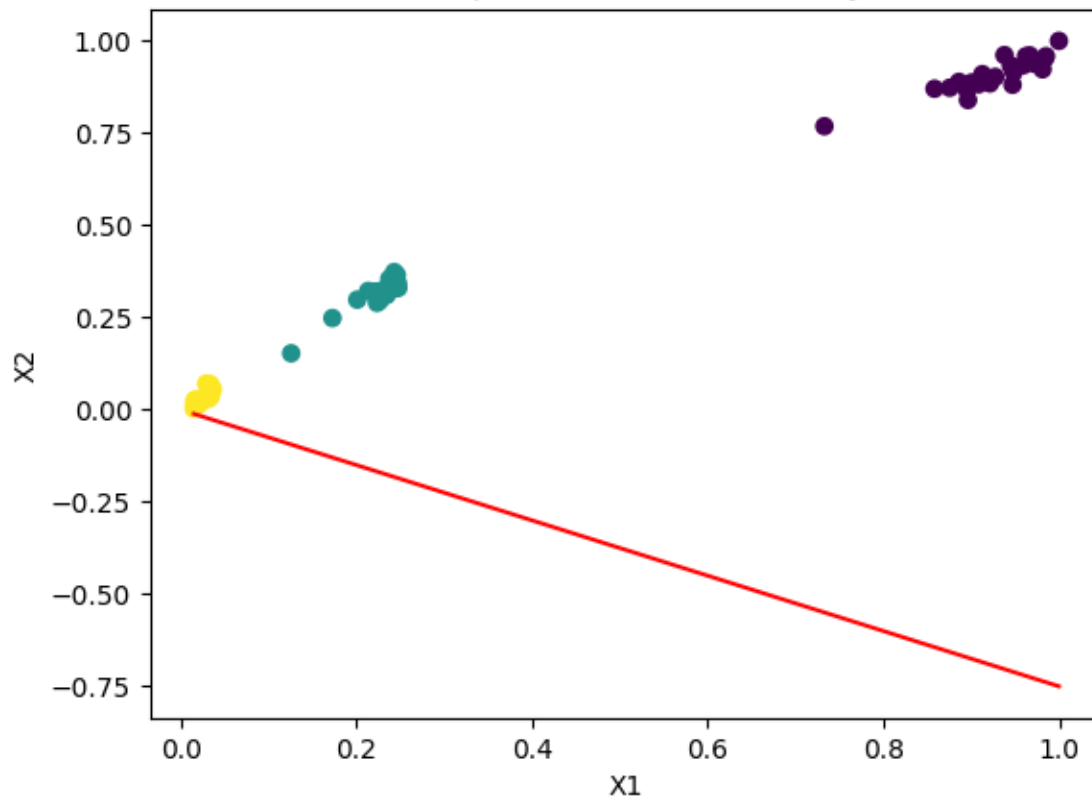
And it should be noted that the training accuracy =0.38 without adding bias

Usage

- The code contains functions for data preprocessing, neural network training, and evaluation.
- Users can adjust parameters such as learning rate, epochs, and network architecture for experimentation.
- The 'testing' function computes the testing accuracy and generates the confusion matrix.

Visualization

Perceptron Decision Boundary



```
confusion matrix=  
[[240 240 240]  
 [180 180 180]  
 [ 0  0  0]]
```