University of Tripoli

Faculty of Engineering

Computer Engineering Department

EE 569: Deep Learning

14-12-2024

Assignment 1 part b

بدر محمد شحيم 2200208256

Instructed by:

د. نوري بن بركة

Abstract

This project explores machine learning and neural network design through four key tasks. First, a non-linearly separable XOR dataset was generated, and the limitations of logistic regression in solving non-linear problems were evaluated. Next, a multi-layer perceptron (MLP) with two hidden layers was implemented using a custom Linear class, demonstrating its ability to handle non-linear datasets like XOR. The project also focused on automating tasks such as parameter initialization and graph construction to improve code maintainability. Finally, a neural network was developed to classify handwritten digits from the MNIST dataset, showcasing the practical application of neural networks to real-world problems.

Introduction

This report focuses on improving a logistic regression model by creating custom computation nodes, like the Linear node, to perform key calculations. It also introduces batching, which allows the model to process multiple data points at once for better performance. The project explores how changing the batch size affects the model's training and accuracy. All changes are made using Python libraries like NumPy, Matplotlib, and SciPy to ensure efficiency and simplicity. The goal is to better understand the logistic regression process and how batching impacts training.

Procedure and Results

A) Task 1: XOR Problem

To implement this, we utilized the sigmoid, BCE, and linear classes with their batch-enabled versions. Initially, we generated an XOR dataset with non-linearly separable classes and visualized it using a scatter plot to analyze its structure.

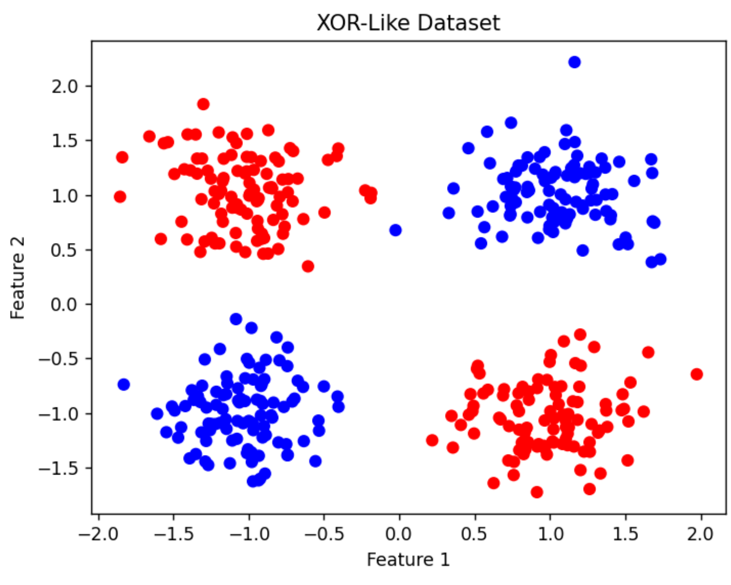

Figure 2.1: XOR Generated Data

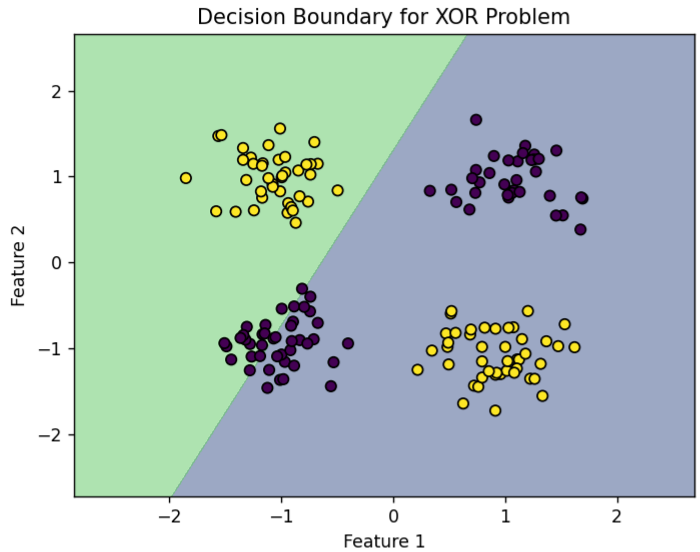

Figure 2.2: Results of Task 1

As you can see it's impossible to solve the XOR problem with Linear class only and, it gave a result of 44%

B) Task 2: Multi-Layer Perceptrons

We implemented a Multi-Layer Perceptron (MLP) using the Linear class from Assignment 1 to solve the XOR problem. The MLP had two hidden layers, each with 20 neurons, and used the sigmoid activation function.
The model was trained on the XOR dataset generated in Task 1. The forward pass computed the output through the layers, and the backward pass adjusted the model parameters using gradients.
After training, the MLP successfully learned to classify the XOR dataset, showing that adding hidden layers and non-linear activations allowed the model to solve the problem, which a simple linear model could not.

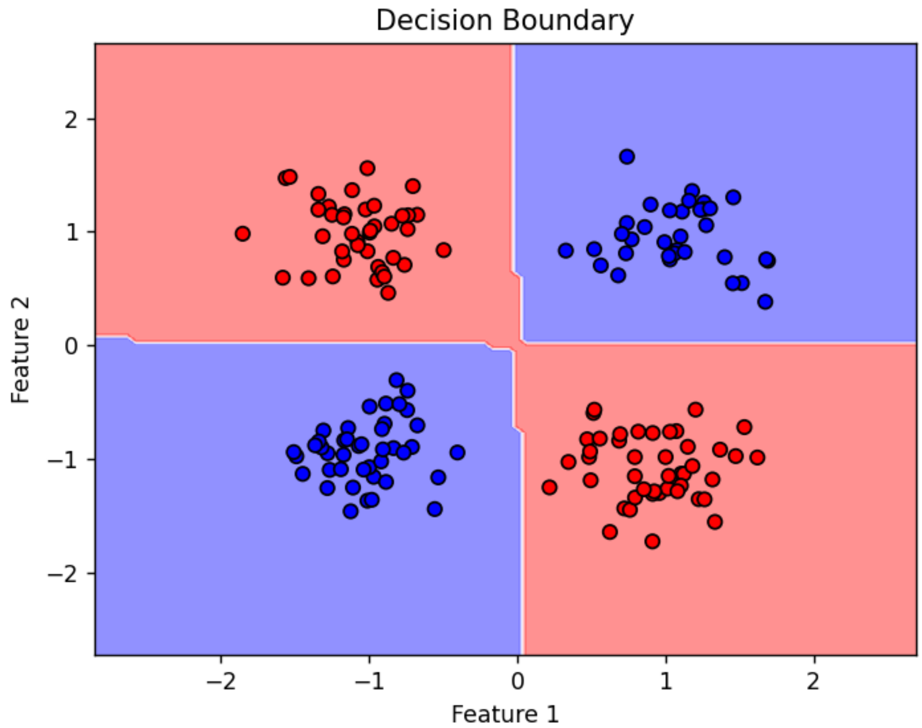

Figure 2.3: Decision Boundary of XOR data using MLP

This gave us an accuracy of 98.12%

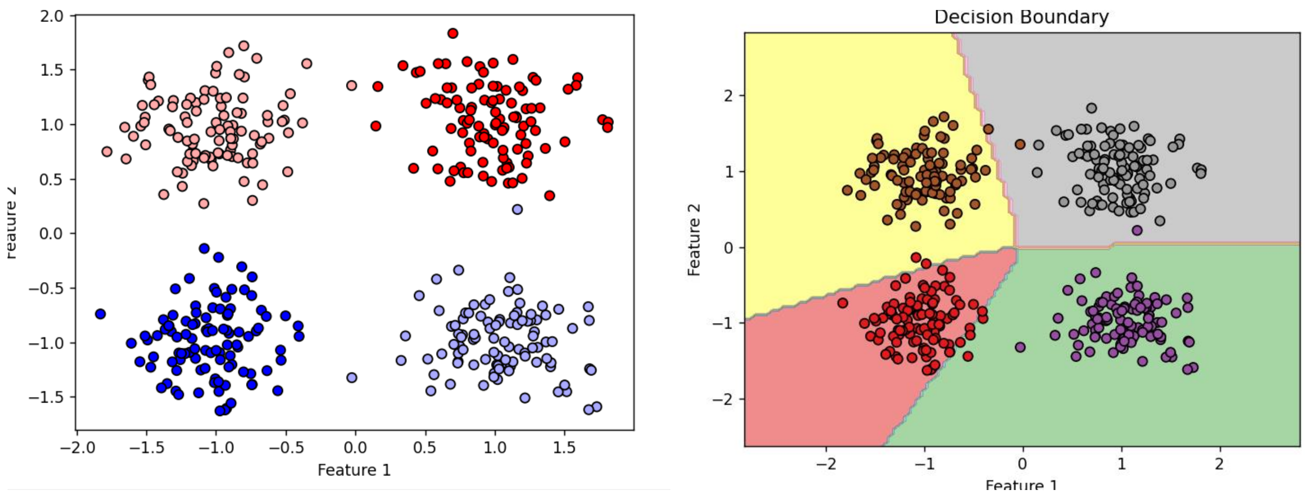

Figure 2.4: Decision Boundary of multiple clusters

C) Task 3: Code Refactoring and Automation

Training a neural network by processing the entire dataset at once can be inefficient for several reasons, including computational limitations and hardware constraints. To address this, we divide the dataset into smaller, manageable subsets called **batches**.

we modified the Linear class, Sigmoid class and BCE class to take in Batches as well.

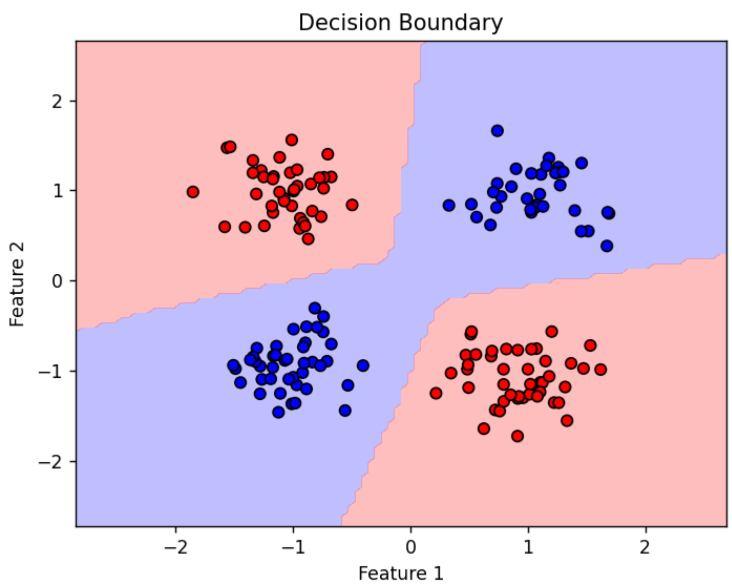

Figure 2.5: Decision Boundary of after Reducing and Enhancing Code

This Gave us an Accuracy of 100% !

D) Task 4: Handwritten Digit Classification using MNIST Dataset

In this task, a neural network model was successfully implemented to classify handwritten digits using the simplified MNIST dataset. The 64-dimensional input features (flattened 8x8 images) were used to train the model, and its performance was evaluated on unseen data. The model achieved high accuracy, demonstrating its ability to learn and generalize from the provided dataset..

Figure 2.6: Predicted Outcome with the Actual Number

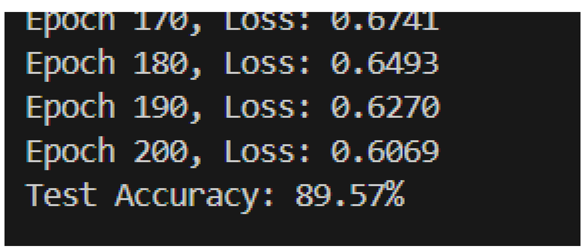

Figure 2.7: Predicted Numbers Accuracy

Conclusion & Discussion

Task 1 (XOR Problem): Logistic regression performed poorly on the XOR dataset as it is linearly inseparable, highlighting the need for non-linear models.

Task 2 (Multi-Layer Perceptrons): The MLP with two hidden layers and sigmoid activations successfully solved the XOR problem, achieving near-perfect accuracy by capturing its non-linear nature.

Task 3 (Code Refactoring and Automation): Automating parameter creation and graph generation using topological sorting improved code maintainability and reduced manual effort.

Task 4 (Handwritten Digit Classification): The neural network effectively classified handwritten digits from the MNIST dataset, demonstrating its capability to handle high-dimensional inputs with strong performance.

References

1. Numpy Library:
   https://numpy.org/

2. Matplotlib
   https://matplotlib.org/

3. Scipy
   https://docs.scipy.org/doc/scipy/

4. SkeLearn
   https://scikit-learn.org/