Assignment 1B: XOR Classification & MNIST Digit Recognition with Neural Networks
Deep learning - EE569

Talal Malek Badi - 2200208609
Supervisor Dr. Nuri Benbarka

December 24, 2024

Contents

1 Introduction **3**

2 Task 1: XOR Problem **3**
2.1 Task Description 3
2.2 Results 3
2.3 implementation 3

3 Task 2: Multi-Layer Perceptron **4**
3.1 Task Description 4
3.2 Results 4
3.3 implementation 4

4 Task 3: Code Refactoring and Automation **4**
4.1 Task Description 4
4.2 Results 4
4.3 implementation 5

5 Task 4: Handwritten Digit Classification using MNIST Dataset **5**
5.1 Task Description 5
5.2 Results 5
5.3 implementation 5

6 Conclusion **6**

List of Figures

1 XOR - using logistic regression 3
2 XOR - using MLP 4
3 Confusion matrix for MNIST handwritten digits recognition 6

1 Introduction

This Assignment includes refactoring and optimizing the previous assignment code and reuse it for the xor problem and MNIST Digit Recognition, it includes generating data for the XOR problem, implementing a Multi-Layer Perceptrons, Refactoring the code for automatic generation of the graph and the trainables, and develop a neural network model to classify handwritten digits. Each task's corresponding results are documented below and the python implementation is on git-hub repository, Each task has it's own commit in the repository for clean version control.

2 Task 1: XOR Problem

2.1 Task Description

In this task, A generation of 2 gaussian distribution classes taking the shape of an XOR problem was generated. Also, tested the precision of the linear regression for this problem.

2.2 Results

The data set was successfully generated as show in figure 1, it was also tested using the linear logistic regression and obviously as we know that the XOR can't be solved using a liner solution, this method didn't work probably, after the test the accuracy was about 40%.

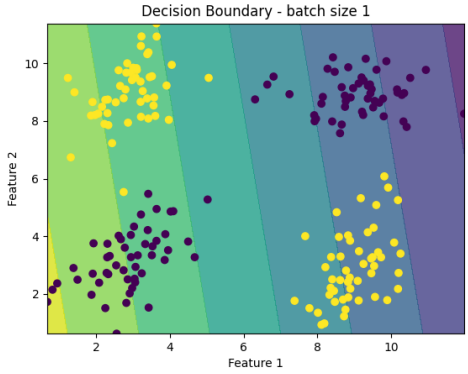

Figure 1: XOR - using logistic regression

2.3 implementation

GitHub commit for task 1

3 Task 2: Multi-Layer Perceptron

3.1 Task Description

In this task, a multi-layer perceptron (MLP) model is implemnted using the previously implemented Linear class, it includes two hidden layers (depth = 2) , with width of 20 neurons, and uses the sigmoid activation function. The model is trained and tested on the XOR dataset generated in Task 1. This Task shows the ability of MLP to handle the nonlinear decision boundaries like XOR problem, with a comparison with the logistic regression from Task 1.

3.2 Results

The multilayer perceptron (MLP) was succfully implemented also tested and got (98-100)% accuracy as shown in Figure 2. The model was tested with several learning rates and different means to see it's performance.

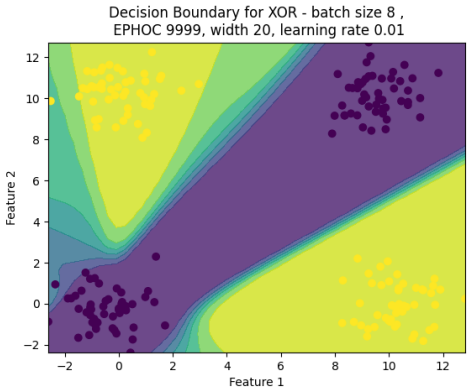

Figure 2: XOR - using MLP

3.3 implementation

GitHub commit for task 1

4 Task 3: Code Refactoring and Automation

4.1 Task Description

In this task the code was refactored for better maintainbility and to insure scalablity.

4.2 Results

The Creation of parameter nodes, Graph and trainable was successfully automated, A recursive function was implemented using the idea of topological sort. as shown in the implementation.

4.3 implementation

```
def topologicalSort(node, graph, trainable):
    for n in node.inputs:
        topologicalSort(n, graph, trainable)
    graph.append(node)
    if isinstance(node, Parameter):
        trainable.append(node)
```

GitHub commit for this task 3 b commit for this task 3&4

5 Task 4: Handwritten Digit Classification using MNIST Dataset

5.1 Task Description

In this Task MNIST simple dataset for digits was used to train and test a model, A neuron network was developed for the handwritten digits.

5.2 Results

The Model was successfully trained insuring stabilized calculations, Accuracy was more than 96% for 50 EPHOC, The Confusion matrix in Figure 3 shows detailed results for the model.

Observations: The confusion matrix shows that most samples are correctly classified, as indicated by the strong diagonal. Nearly all digits are predicted correctly with few errors,showing high accuracy and good performance for all digits.

5.3 implementation

GitHub commit for task 1

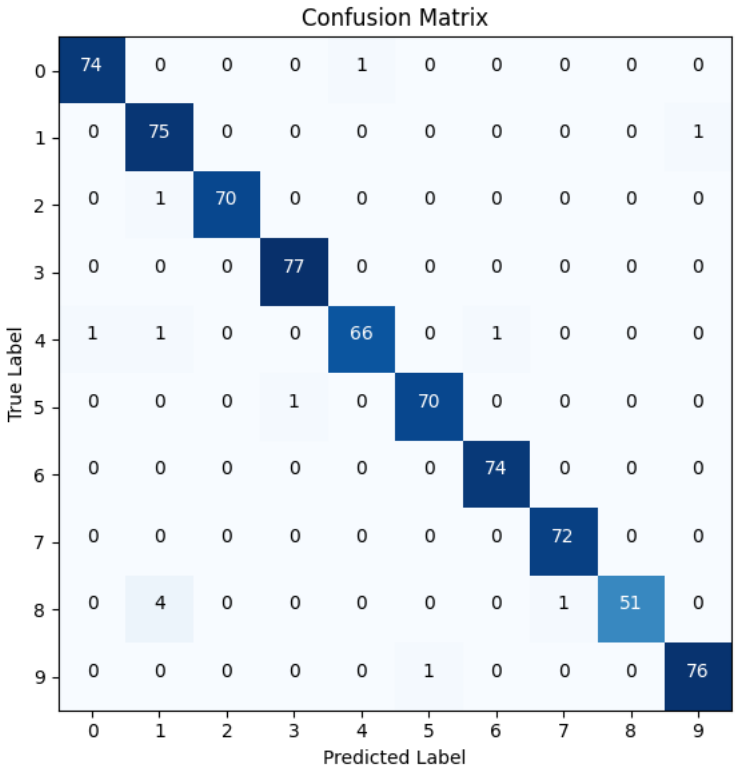

Figure 3: Confusion matrix for MNIST handwritten digits recognition

6 Conclusion

In this assignment, multiple tasks were implemented and analyzed successfully:

- The XOR problem showed the limitations of linear classifiers, as logistic regression failed to classify the data, achieving only 40% accuracy.
- A Multi-Layer Perceptron (MLP) with two hidden layers successfully solved the XOR problem, achieving an accuracy of 98-100%. This shows the power of MLPs for nonlinear decision boundaries.
- Code refactoring and automation were performed for improved scalability and maintainability, for efficient graph creation and training processes using a recursive topological sorting.
- For the MNIST handwritten digit recognition task, the neural network model outputs over 96% accuracy, showing its efficincy for nonlinear.