Handwritten Digit Recognition using Neural Networks

Authors: Anjali Pathak, Mohitha Mahesh Gowda
Date: December 11, 2023

Executive Summary

This report documents the design, implementation, and analysis of a custom-built neural network for handwritten digit recognition. The project commenced with a basic neural network consisting of a single hidden layer with 10 neurons and progressively evolved to incorporate two hidden layers with 128 and 64 neurons, respectively. This report outlines the methodologies, outcomes, and insights gained through this iterative process.

Introduction

Background Information

Handwritten digit recognition is a fundamental task in the field of machine learning and computer vision. This project focuses on creating a neural network from scratch without relying on pre-built libraries like Keras or TensorFlow. The objective is to gain a comprehensive understanding of neural network architectures and their applications in image classification tasks.

Objectives and Goals

1. Construct a neural network using fundamental concepts and algorithms to accurately recognize handwritten digits.
2. Investigate the influence of modifying network architecture, including hidden layers and neurons, on recognition accuracy.
3. Analyse the impact of different parameters on the network's learning and performance.

Scope of the Project

The project's primary scope involves implementing a custom neural network using Python and core mathematical concepts to recognize handwritten digits based on the widely-used MNIST dataset.

Methodology

Methods and Approaches Used

The project leveraged Python programming language, NumPy for numerical computations, and Matplotlib for data visualisation. The neural network was designed with multiple layers and adjustable neurons. Backpropagation, a fundamental technique in neural network training, was employed for learning from the dataset.

Tools and Technologies

Python: For programming and implementation
NumPy: For numerical computations and array manipulation

Matplotlib: For visualising data and model performance
Data Sources: The MNIST dataset, comprising 60,000 training examples and 10,000 test examples of handwritten digits, was utilised for both training and evaluating the neural network model. It was acquired from Kaggle.

Project Member Work Distribution

The team was organised with specific roles and responsibilities:

TEAM MEMBER	ROLES AND RESPONSIBILITIES
MOHITHA	**Worked on Initialise function** (This was used to randomly initialise the weights in the network) **Conducted Data Preprocessing** (This included collecting the data, Loading the data, normalising the data, extracting labels, and splitting data into training and test sets.)
ANJALI	**Developed Prediction Function and Display Function** (This involved tasks such as computing activations for hidden and output layers, predicting classes using the highest hypothesis, determining precision, training set accuracy, and testing set accuracy and finally displaying the image with its actual and predicted labels.)
ANJALI AND MOHITHA	**Implemented Core Neural Network Algorithm** (This included functionalities like Forward propagation, Backward propagation, Calculating Gradient and, Estimating Cost Function)

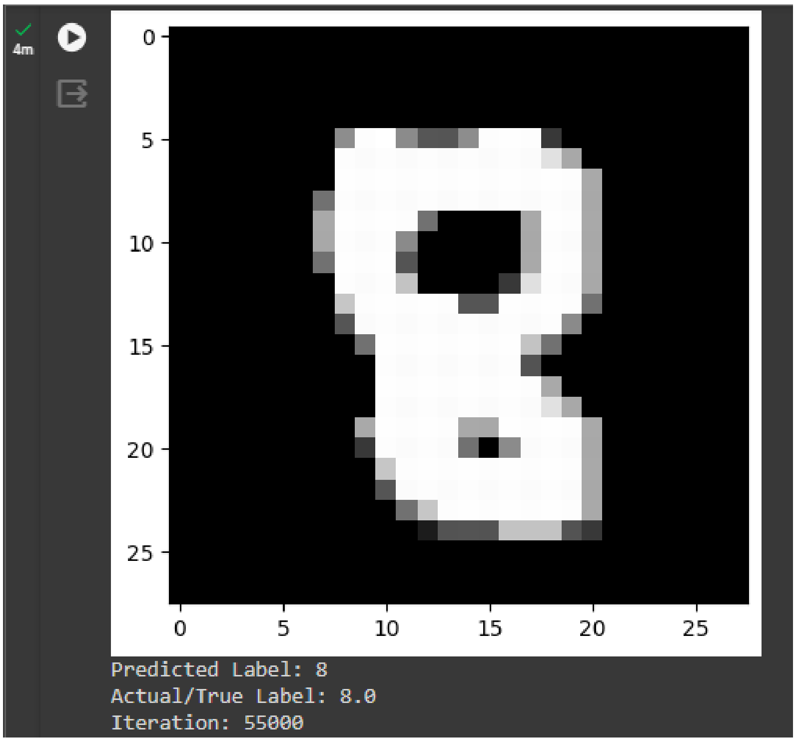
Project Progress/Development

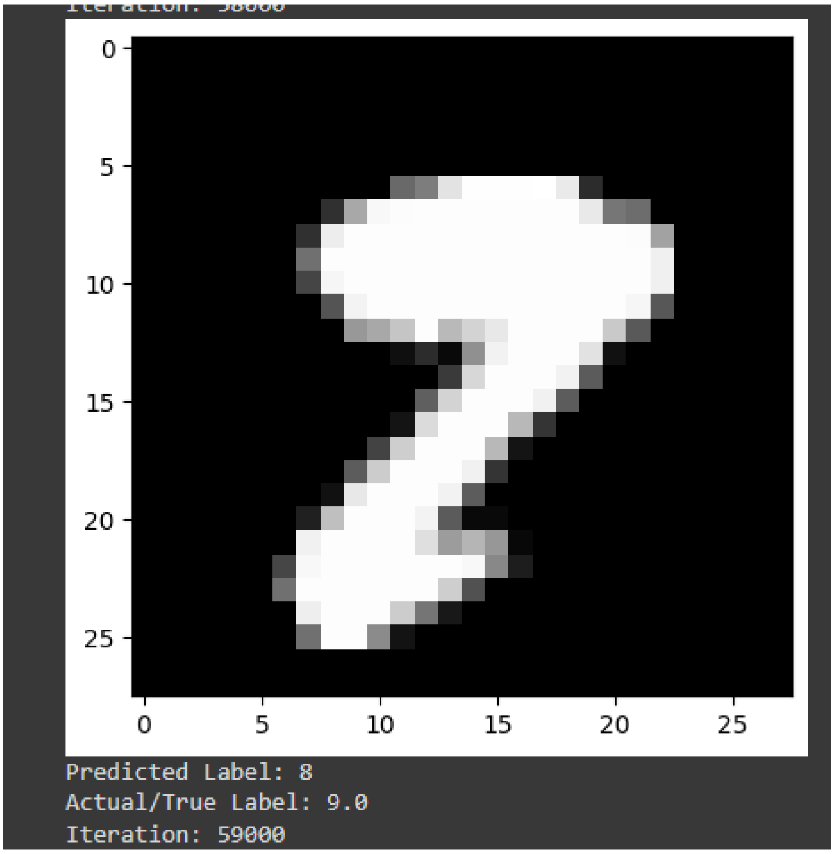
The project evolved through various stages:

Stage 1: Initial Implementation: A basic neural network was constructed with a single hidden layer and 10 neurons.
Stage 2: Architecture Enhancement: The network architecture was iteratively modified to include two hidden layers with 128 and 64 neurons, respectively.
Stage 3: Rigorous Evaluation: Extensive testing, evaluation, and analysis of the model's accuracy and performance were conducted using the MNIST dataset.

Results and Findings

The custom-built neural network demonstrated noteworthy improvements in accuracy as the number of hidden layers and neurons increased. Visual representations showcased the enhanced ability of the model in recognizing handwritten digits across various classes.





```
Actual/True Label: 9.0
Test Set Accuracy: 97.190000
Iteration: 0
```

```
Training Set Accuracy: 98.908333
Precision =  0.9890833333333333
```

Discussion

The findings were analysed and discussed in relation to the initial objectives of the project. The direct correlation between increased network complexity and improved accuracy was highlighted.

Conclusions and Recommendations

Conclusions

The custom-built neural network exhibited enhanced accuracy with the inclusion of additional hidden layers and neurons.
Increased network complexity positively impacted the model's performance in recognizing handwritten digits.

Recommendations

Further exploration into diverse architectures, activation functions, and regularisation techniques to enhance accuracy and generalisation.
Experimentation with different optimization algorithms to improve the learning efficiency of the custom neural network.

References

[1] https://www.youtube.com/watch?v=0idoEomDc9E

[2] https://towardsdatascience.com/the-maths-behind-back-propagation-cf6714736abf

[3] https://www.geeksforgeeks.org/handwritten-digit-recognition-using-neural-network/

[4] https://www.kaggle.com/avnishnish/mnist-original/download

[5] https://towardsdatascience.com/the-maths-behind-back-propagation-cf6714736abf