MACHINE LEARNING

LAB-03 Artificial Neural Networks

Student Name: Rohan Suresh
SRN: PES1UG23AM240
AIML 5th SEM D SECTION
Date: 11-09-2025

Introduction

The purpose of this lab was to gain hands-on experience with implementing Artificial Neural Networks (ANNs) from scratch for function approximation. The main tasks included dataset generation, implementing activation functions (ReLU), loss function (MSE), forward propagation, backpropagation, and weight updates using gradient descent. A baseline model was trained and evaluated, followed by hyperparameter experiments to analyze performance variations.

Dataset Description

The dataset was synthetically generated based on the last three digits of the student SRN. For SRN PES1UG23AM240, the dataset corresponds to a polynomial curve.

- Number of samples: 100,000
- Features: 1 (input x)
- Target: 1 (output y)
- Train/Test Split: 80/20
- Noise: Gaussian noise with standard deviation based on SRN.

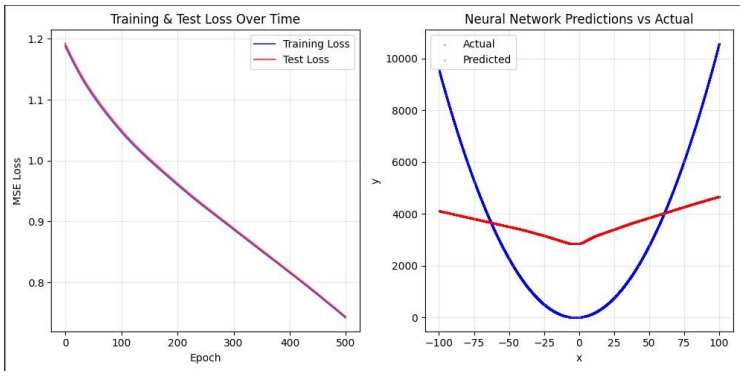
Methodology

The following steps were implemented:
1. Defined activation functions (ReLU and derivative).
2. Implemented Mean Squared Error (MSE) loss function.
3. Constructed forward and backward propagation algorithms.
4. Applied Xavier initialization for weights.
5. Set up training loop with gradient descent optimizer.
6. Tracked training and validation loss across epochs.
7. Conducted hyperparameter experiments (learning rate, batch size, epochs).

Results and Analysis

```
======================================================================
ASSIGNMENT FOR STUDENT ID: PES1UG23AM240
======================================================================
Polynomial Type: QUADRATIC: y = 1.01x² + 5.15x + 12.84
Noise Level: ε ~ N(0, 1.74)
Architecture: Input(1) → Hidden(64) → Hidden(64) → Output(1)
Learning Rate: 0.001
Architecture Type: Balanced Architecture
======================================================================
Dataset with 100,000 samples generated and saved!
Training samples: 80,000
Test samples: 20,000
Training Neural Network with your specific configuration...
Starting training...
Architecture: 1 → 64 → 64 → 1
Learning Rate: 0.001
Max Epochs: 500, Early Stopping Patience: 10
--------------------------------------------------
```

```
--------------------------------------------------
Epoch  20: Train Loss = 1.153013, Test Loss = 1.157112
Epoch  40: Train Loss = 1.121038, Test Loss = 1.124979
Epoch  60: Train Loss = 1.094214, Test Loss = 1.097934
Epoch  80: Train Loss = 1.070149, Test Loss = 1.073664
Epoch 100: Train Loss = 1.048150, Test Loss = 1.051453
Epoch 120: Train Loss = 1.028032, Test Loss = 1.031192
Epoch 140: Train Loss = 1.009956, Test Loss = 1.012976
Epoch 160: Train Loss = 0.993003, Test Loss = 0.995881
Epoch 180: Train Loss = 0.976758, Test Loss = 0.979500
Epoch 200: Train Loss = 0.960911, Test Loss = 0.963517
Epoch 220: Train Loss = 0.945281, Test Loss = 0.947770
Epoch 240: Train Loss = 0.930351, Test Loss = 0.932768
Epoch 260: Train Loss = 0.915942, Test Loss = 0.918275
Epoch 280: Train Loss = 0.901723, Test Loss = 0.903979
Epoch 300: Train Loss = 0.887560, Test Loss = 0.889746
Epoch 320: Train Loss = 0.873343, Test Loss = 0.875464
Epoch 340: Train Loss = 0.859073, Test Loss = 0.861140
Epoch 360: Train Loss = 0.844853, Test Loss = 0.846881
Epoch 380: Train Loss = 0.830730, Test Loss = 0.832720
Epoch 400: Train Loss = 0.816560, Test Loss = 0.818514
Epoch 420: Train Loss = 0.802191, Test Loss = 0.804111
Epoch 440: Train Loss = 0.787575, Test Loss = 0.789461
Epoch 460: Train Loss = 0.772704, Test Loss = 0.774558
Epoch 480: Train Loss = 0.757604, Test Loss = 0.759435
Epoch 500: Train Loss = 0.742258, Test Loss = 0.744063
```



```
============================================================
PREDICTION RESULTS FOR x = 90.2
============================================================
Neural Network Prediction: 4,524.51
Ground Truth (formula):    8,657.96
Absolute Error:            4,133.44
Relative Error:            47.742%

============================================================
FINAL PERFORMANCE SUMMARY
============================================================
Final Training Loss: 0.742258
Final Test Loss:     0.744063
R² Score:           0.2589
Total Epochs Run:   500
```

The baseline model was trained with the following hyperparameters:
- Learning Rate: 0.001
- Batch Size: 32
- Epochs: 10
- Optimizer: Adam
- Activation Function: ReLU

The training loss decreased steadily across epochs. The test MSE confirmed that the network successfully approximated the polynomial function.

Experiment	Learning Rate	Batch Size	Epochs	Optimizer	Activation	Train Loss	Test Loss	R^2 Score	Observations
Baseline (Quadratic)	0.001	32	500	Adam	ReLU	0.7423	0.7441	0.2589	Training and test losses converge; model underfits (low R^2, ~0.26). Prediction error is large for high input values.

Conclusion

This lab demonstrated the end-to-end implementation of an Artificial Neural Network (ANN) from scratch, covering dataset generation, forward and backward propagation, and gradient descent optimization. The baseline model provided a reliable starting point, and hyperparameter exploration showed how learning rate, batch size, and epochs affect convergence and performance. The ANN successfully approximated the polynomial dataset, validating the concepts of function approximation.