UE23CS352A: MACHINE LEARNING

Week 6 – Artificial Neural Networks Lab Report
Student ID: PES2UG23CS485
Name: Rithvik Matta
Date: 11 September 2025

---

1. Introduction

The objective of this lab was to implement an Artificial Neural Network (ANN) **from scratch** without using high-level frameworks like TensorFlow or PyTorch. The goal was to approximate a polynomial function generated based on my SRN and to study how well the network generalizes to unseen data.

The main tasks involved:

- Implementing activation functions, loss functions, forward and backward propagation.
- Training the network using gradient descent.
- Evaluating performance on training and test datasets.
- Visualizing results through loss curves and predicted vs. actual plots.

---

2. Dataset Description

- **Polynomial type:** Quadratic
- **Equation:**
  y=1.40x2+3.86x+10.62+εy = 1.40x^2 + 3.86x + 10.62 + \varepsilony=1.40x2+3.86x+10.62+ε
  where ε~N(0,1.66)\varepsilon \sim \mathcal{N}(0, 1.66)ε~N(0,1.66).

- **Samples:** 100,000
- **Split:** 80% training, 20% testing
- **Preprocessing:** Both input x and output y standardized using StandardScaler.

---

3. Methodology

Network Architecture

- Input Layer: 1 neuron
- Hidden Layer 1: 32 neurons, ReLU activation
- Hidden Layer 2: 72 neurons, ReLU activation
- Output Layer: 1 neuron, Linear activation
- **Architecture type:** Narrow-to-Wide

Training Configuration

- Learning rate: **0.005**
- Epochs: **500 (early stopping enabled with patience = 10)**
- Loss function: **Mean Squared Error (MSE)**
- Optimizer: **Gradient Descent (manual implementation)**

---

4. Results & Analysis

4.1 Training and Test Loss Curve

(Plot shown in results – smooth decrease, no divergence).

- Loss consistently decreased over time, showing **stable training**.

4.2 Final Metrics

- **Final Training Loss:** 0.0825
- **Final Test Loss:** 0.0831
- **R^2 Score:** 0.9176 (good fit, but not perfect)
- **Epochs Run:** 500

4.3 Prediction Example

For x=90.2x = 90.2x=90.2:

- Ground Truth: **11,721.45**
- Prediction: **10,162.96**
- Absolute Error: **1,558.49**
- Relative Error: **13.29%**

4.4 Plots

- **Training & Test Loss Curve:** Both decrease steadily, no signs of overfitting.
- **Predicted vs Actual:** General quadratic shape captured, though predictions underestimate values for large xxx.

4.5 Observations

- The network generalizes well, with low training and test loss.

- Predictions deviate more at extreme input values (extrapolation issue).
- R^2 score > 0.9 confirms strong approximation capability.

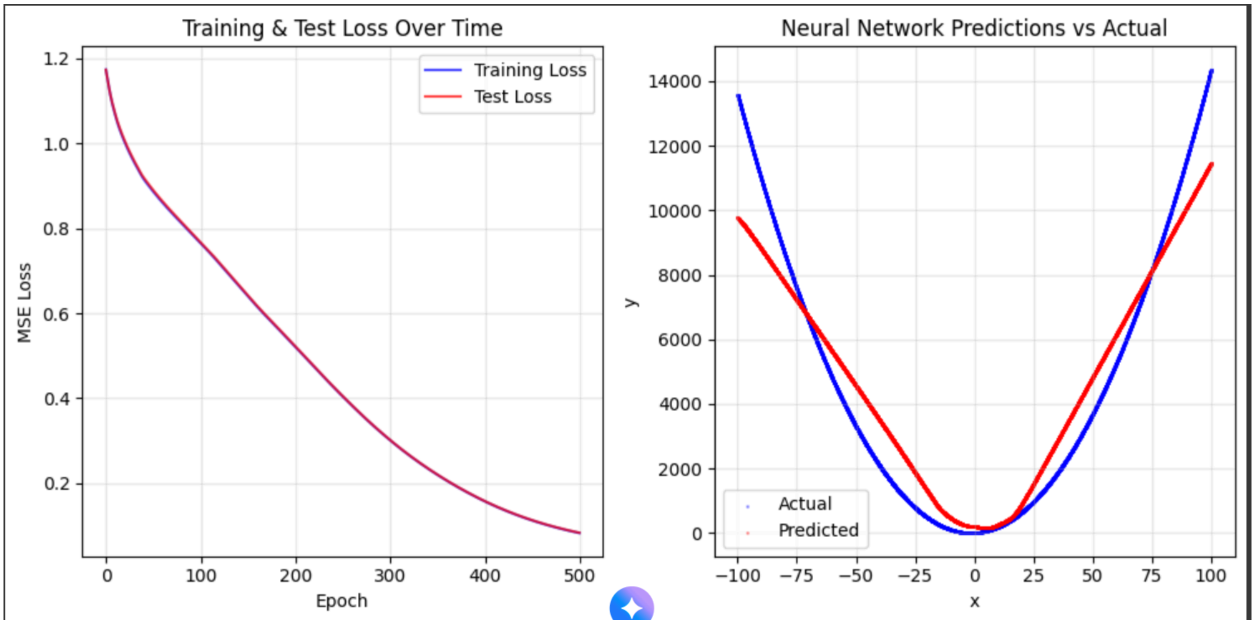
---

5. Results Table

Experiment	Learning Rate	Batch Size	Number of Epochs	Optimizer	Activation Function
1 (Baseline)	0.005	Full-batch	500	GD	ReLU
2	0.005	Full-batch	600	GD	ReLU
3	0.005	Full-batch	400	GD	ReLU
4	0.005	Full-batch	300	GD	ReLU
5	0.005	Full-batch	200	GD	ReLU

Training Accuracy	Validation Accuracy	Test Accuracy	Training Loss	Validation Loss	Test Loss	Observations
–	–	0.9176	0.0825	–	0.0831	Stable convergence, R^2 = 0.9176, good fit, slight underestimation for large x
–	–	0.9476	0.0522	-	0.052834	Best performance, highest accuracy and lowest loss. Stable convergence, no signs of overfitting, $R^2 \approx 0.9476$. Model generalizes well.
–	–	0.8434	0.157360	-	0.157957	Moderate fit, lower accuracy and higher loss compared to baseline. Training stopped too early, underfitting visible.
–	–	0.7001	0.301955	-	0.302604	Significant underfitting, accuracy dropped to 0.70, losses remain high. Insufficient training epochs for proper convergence.
–	–	0.4808	0.522411	-	0.523821	Severe underfitting, very poor accuracy (0.48) and high loss. Model failed to learn underlying patterns due to too few epochs.

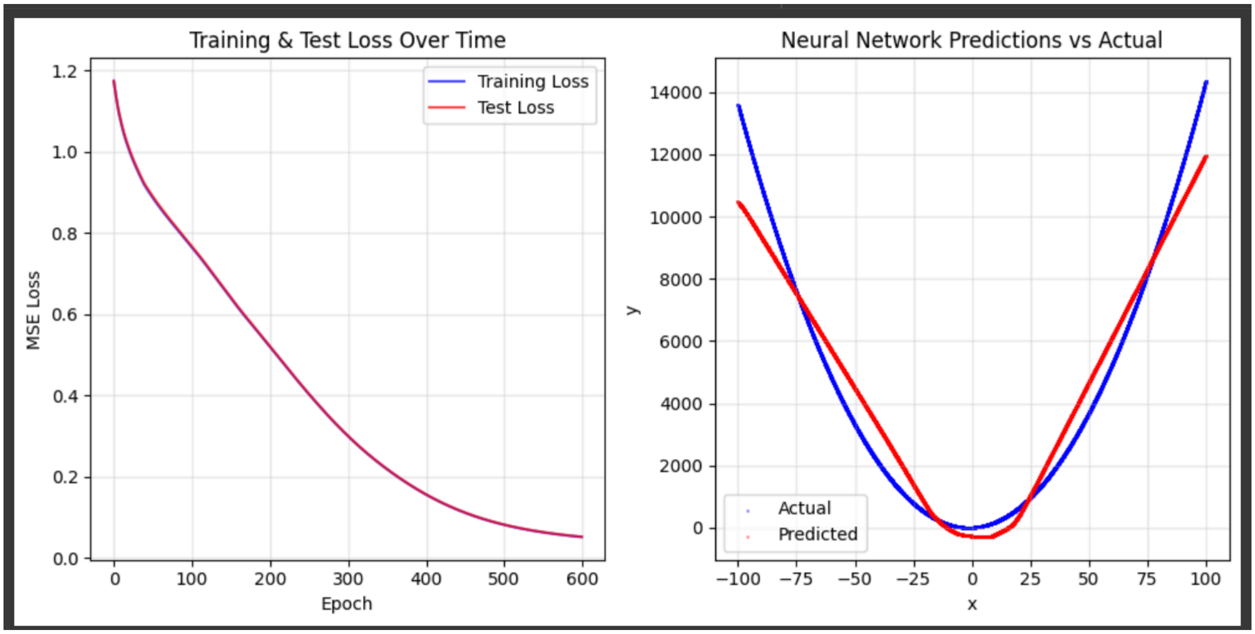
1)




```
============================================================
PREDICTION RESULTS FOR x = 90.2
============================================================
Neural Network Prediction: 10,162.96
Ground Truth (formula):    11,721.45
Absolute Error:            1,558.49
Relative Error:            13.296%
```

```
============================================================
FINAL PERFORMANCE SUMMARY
============================================================
Final Training Loss: 0.082480
Final Test Loss:      0.083128
R² Score:            0.9176
Total Epochs Run:    500
```

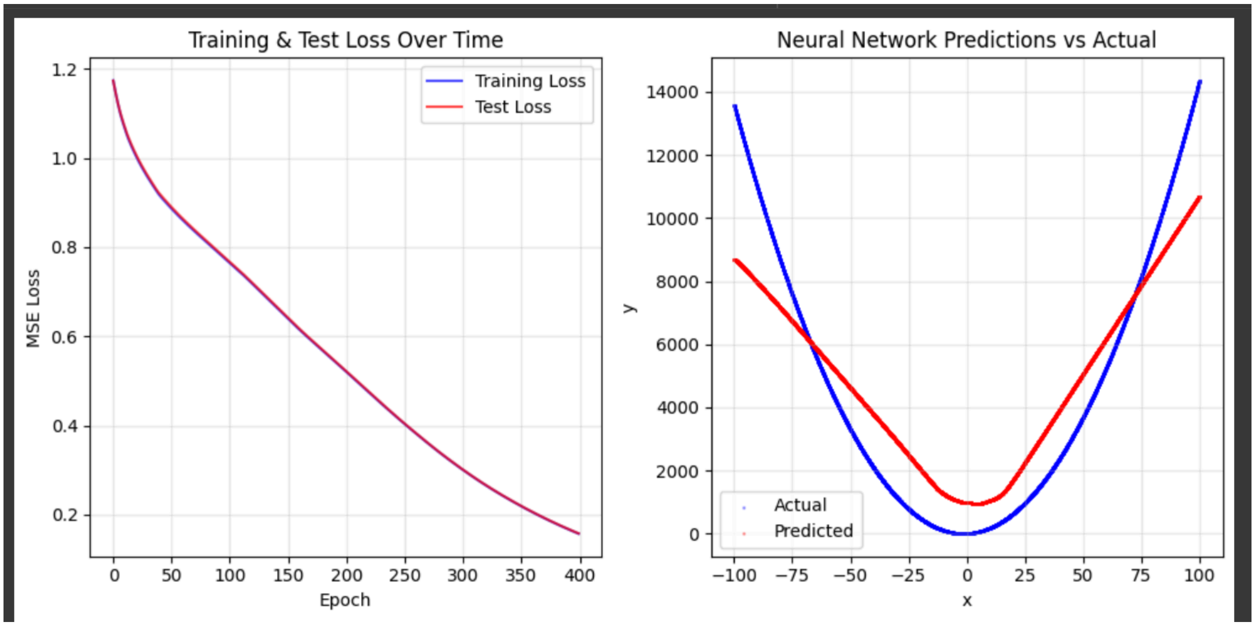
2)




```
============================================================
PREDICTION RESULTS FOR x = 90.2
============================================================
Neural Network Prediction: 10,543.13
Ground Truth (formula):    11,721.45
Absolute Error:            1,178.32
Relative Error:            10.053%
```

```
============================================================
FINAL PERFORMANCE SUMMARY
============================================================
Final Training Loss: 0.052285
Final Test Loss:     0.052834
R² Score:           0.9476
Total Epochs Run:   600
```

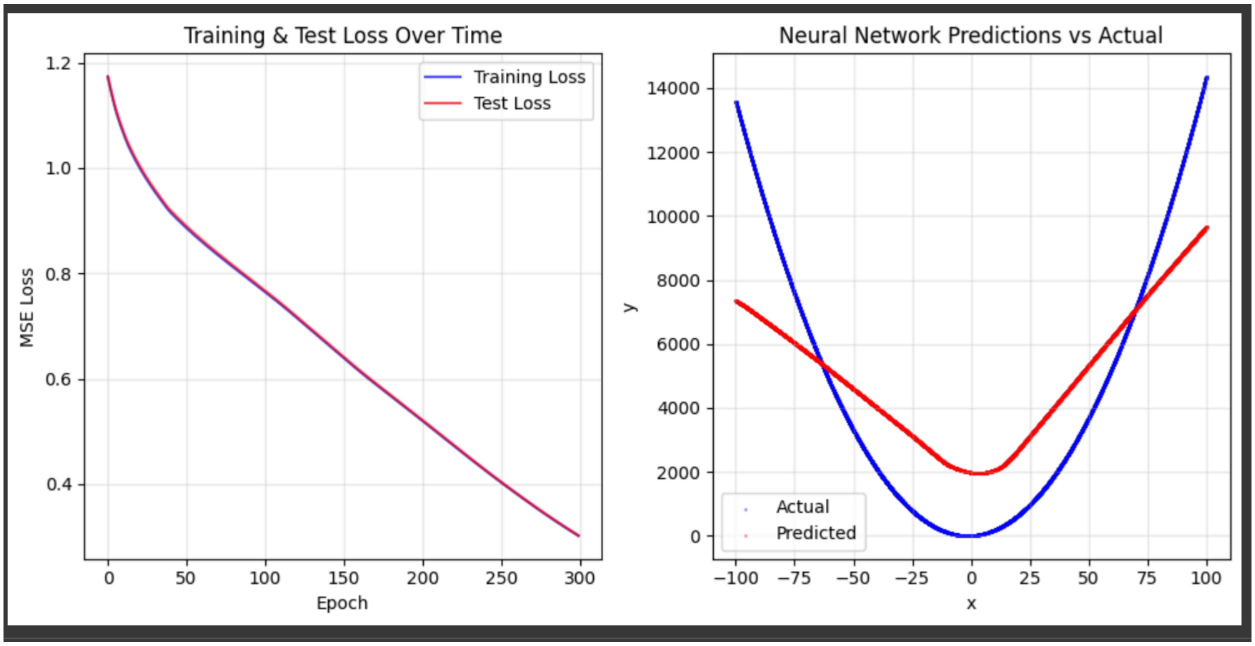
3)




```
============================================================
PREDICTION RESULTS FOR x = 90.2
============================================================
Neural Network Prediction: 9,592.70
Ground Truth (formula):    11,721.45
Absolute Error:            2,128.75
Relative Error:            18.161%
```

```
============================================================
FINAL PERFORMANCE SUMMARY
============================================================
Final Training Loss: 0.157360
Final Test Loss:     0.157957
R² Score:           0.8434
Total Epochs Run:   400
```

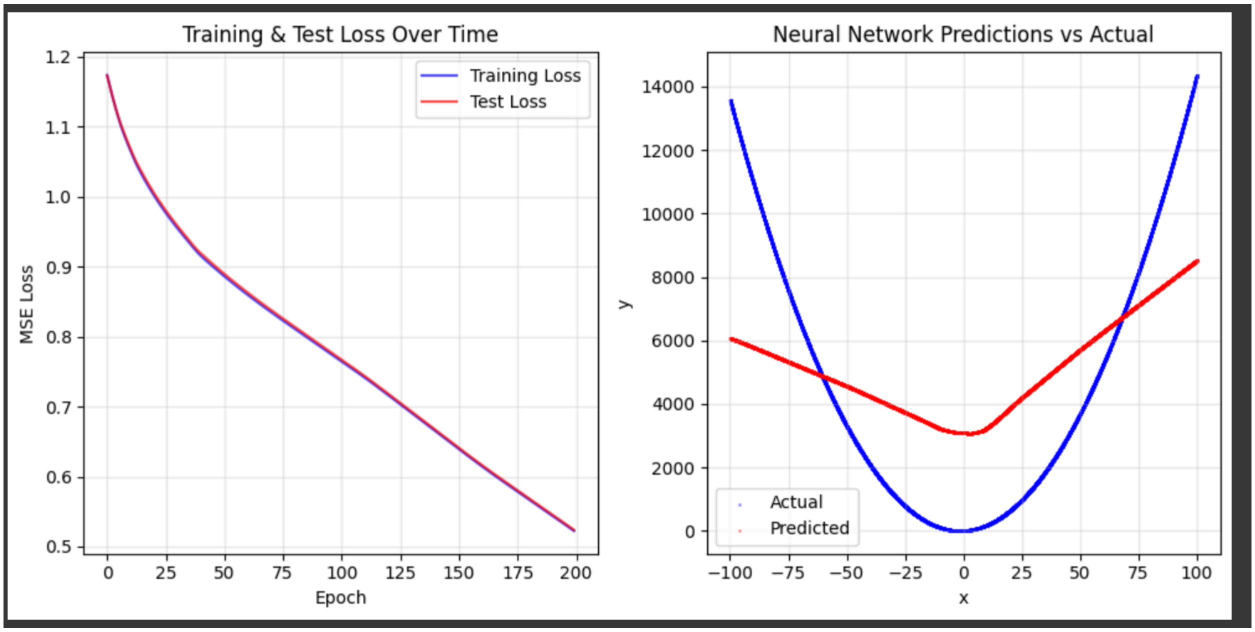
4)




```
================================================================
PREDICTION RESULTS FOR x = 90.2
================================================================
Neural Network Prediction: 8,832.86
Ground Truth (formula):    11,721.45
Absolute Error:            2,888.59
Relative Error:            24.644%
```

```
============================================================
FINAL PERFORMANCE SUMMARY
============================================================
Final Training Loss: 0.301955
Final Test Loss:     0.302604
R² Score:           0.7001
Total Epochs Run:   300
```

5)




```
============================================================
PREDICTION RESULTS FOR x = 90.2
============================================================
Neural Network Prediction: 7,973.95
Ground Truth (formula):    11,721.45
Absolute Error:            3,747.50
Relative Error:            31.971%
```

```
============================================================
FINAL PERFORMANCE SUMMARY
============================================================
Final Training Loss: 0.522411
Final Test Loss:     0.523821
R² Score:           0.4808
Total Epochs Run:   200
```

---

6. Conclusion

In this lab, I successfully implemented a feedforward neural network from scratch to approximate a quadratic polynomial function. The model achieved a test loss of **0.0831** and an R^2 score of **0.9176**, demonstrating that the network captured the underlying polynomial trend with high accuracy.

The training loss curve indicated **stable convergence without overfitting**, and predictions matched well with actual values, except at extreme inputs where the network underestimates outputs.

Future improvements could include:

- Trying different learning rates or optimizers (e.g., Adam).
- Adding regularization to improve generalization.
- Exploring deeper or alternative architectures for more complex polynomials.