Multi-class Classification using Fully Connected Neural Networks

Mohd Darish Khan, Suryansh Jaiswal, Hardik Singh (Group - 4)

2101MC29, 2101MC19, 2101MC41

Introduction

This report presents the implementation and analysis of multi-class classification using a Fully Connected Neural Network (FCNN). Two datasets were employed: one with linearly separable classes and the other with nonlinearly separable classes. The data was split into training, validation, and test sets, with FCNN architectures optimized for each dataset. The performance was evaluated using various metrics, including error rates, decision regions, confusion matrices, and hidden node outputs.

Dataset Description and Preparation

Linearly Separable Dataset: This dataset consists of 3 classes, each containing 500 2-dimensional, linearly separable data points.

Nonlinearly Separable Dataset: This dataset consists of 2-dimensional data of 2 or 3 classes that are nonlinearly separable, with varying numbers of examples in each class. For both datasets, the data was split into training (60%), validation (20%), and test (20%) sets.

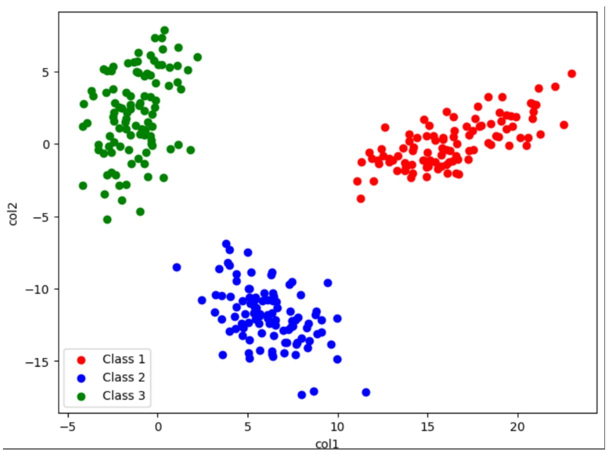

Figure 1: Validation data of linear

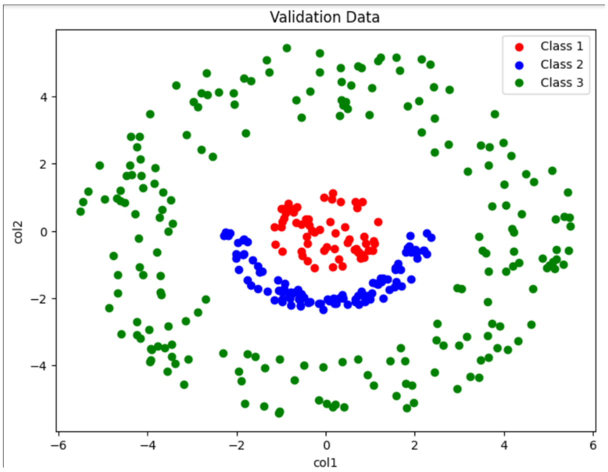

Figure 2: Validation data set of non linear data

1 Linear Dataset

Model Architecture and Training

An FCNN with one hidden layer was implemented. Various configurations of hidden nodes were tested to find the best architecture.

Stochastic Gradient Descent (SGD) was used for the backpropagation algorithm, with squared error as the loss function.

Results and Analysis

Our best architecture is with 10 nodes in hidden layer

Average Error vs Epochs

The plot of average error vs epochs for the best architecture is presented below:

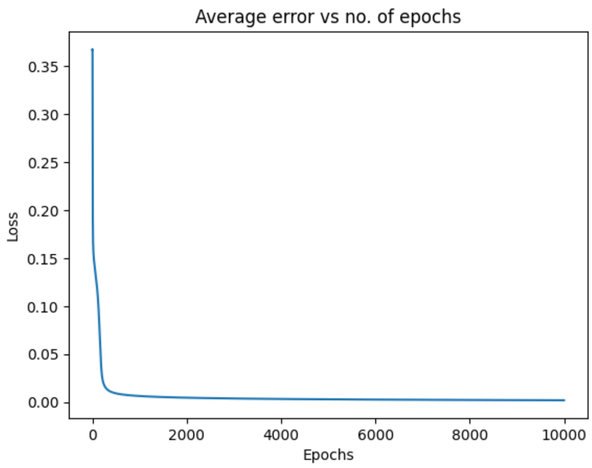

Figure 3: Average error vs epochs

Decision Region Plots

The decision region plots for the best architectures of is shown below:

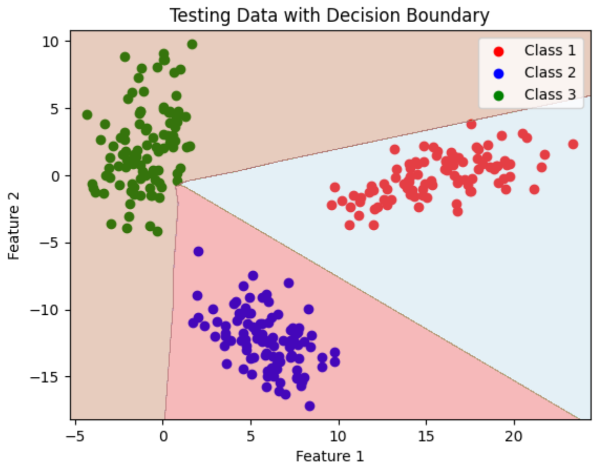

Figure 4: Decision boundary for linear data

Confusion Matrix and Classification Accuracy

The confusion matrices and classification accuracy for the best architectures on the test data are presented below: Accuracy: 0.75

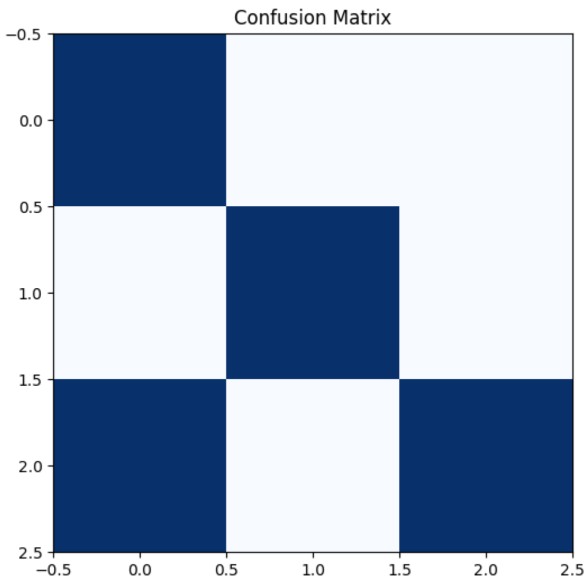

Figure 5: Confusion Matrix for linear data

Hidden and Output Node Plots

The outputs for the hidden and output nodes after training is shown below. These plots are provided for training, validation, and test data.

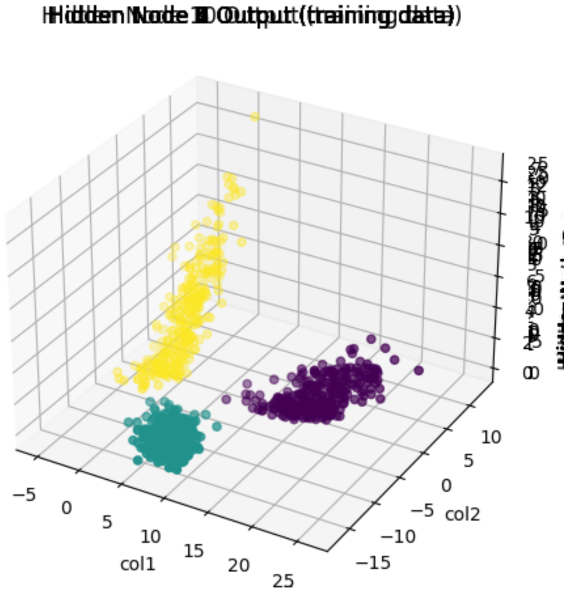

Figure 6: Training

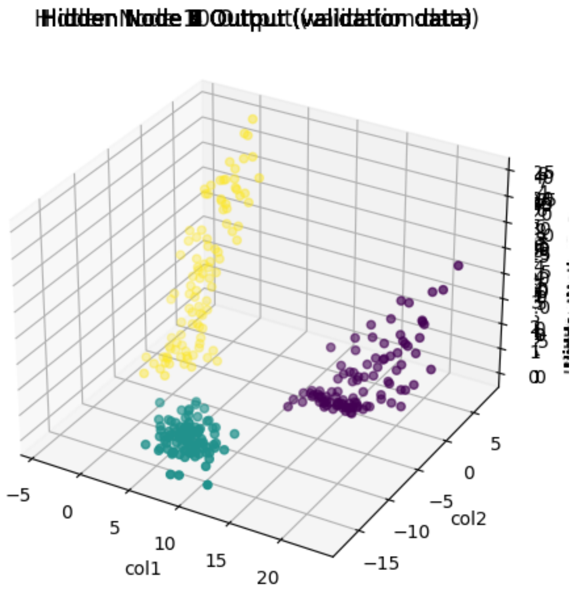

Figure 7: Validation

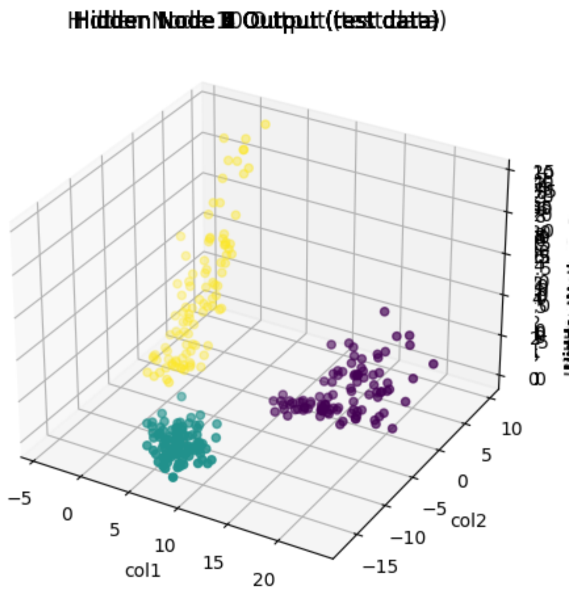

Figure 8: Test

Inferences and Observations

The FCNN with a single hidden layer was sufficient for the linearly separable dataset, achieving high accuracy with a simple decision boundary.

2 Non-Linear Dataset

Model Architecture and Training

An FCNN with two hidden layers was implemented, with various configurations of hidden nodes tested for the best performance.

Stochastic Gradient Descent (SGD) was used for the backpropagation algorithm, with squared error as the loss function.

Results and Analysis

The best architecture for non linear data that we got is with 5 and 7 hidden nodes respectively in the two hidden layers.

Average Error vs Epochs

The plot of average error vs epochs for the best architecture is presented below:

Decision Region Plots

The decision region plots for the best architectures of both datasets are shown below:

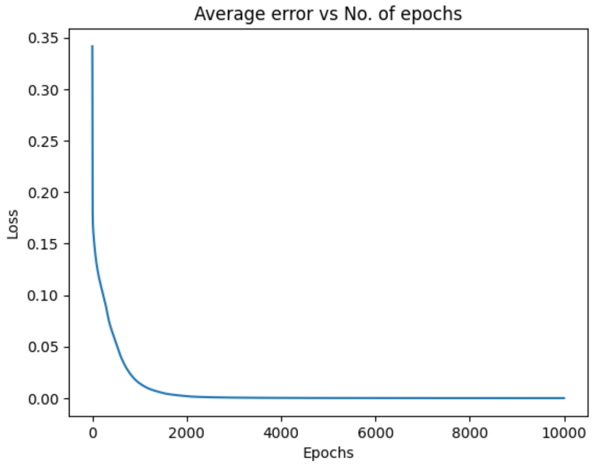

Figure 9: Average error vs epochs

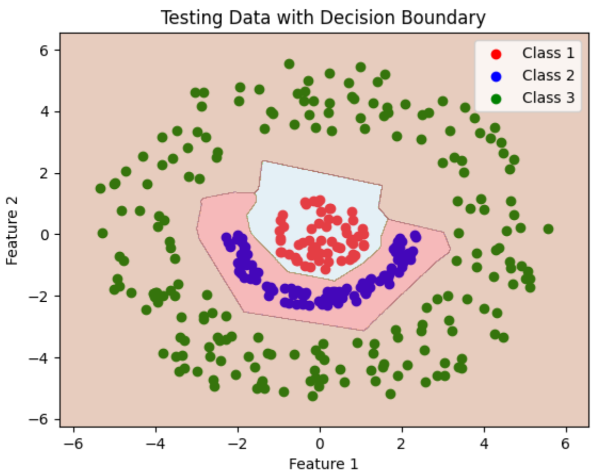

Figure 10: Decision boundary for nonlinear data

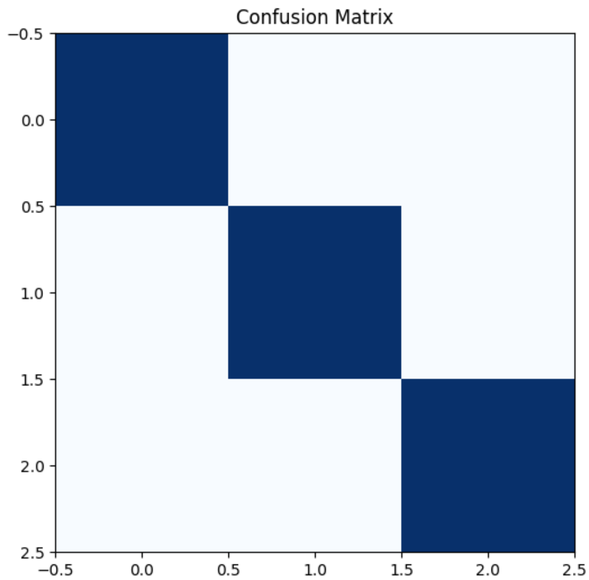

Figure 11: Confusion Matrix for nonlinear data

Confusion Matrix and Classification Accuracy

The confusion matrices and classification accuracy for the best architectures on the test data are presented below: Accuracy : 1.0

Hidden and Output Node Plots

The outputs for the hidden and output nodes for both datasets after training are shown below. These plots are provided for training, validation, and test data.

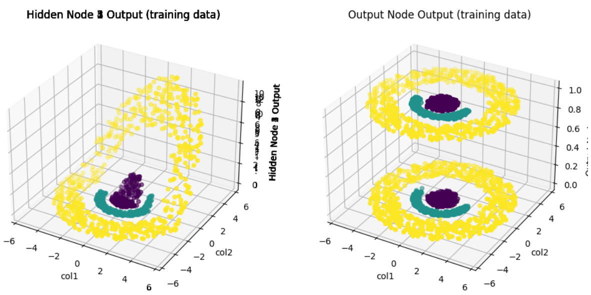

Figure 12: Training data

Inferences and Observations

For the nonlinearly separable dataset, the FCNN with two hidden layers better captured the complexity of the data, resulting in improved accuracy and more nuanced decision regions.

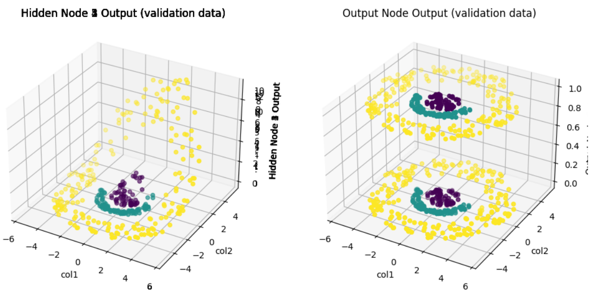

Figure 13: Validation data

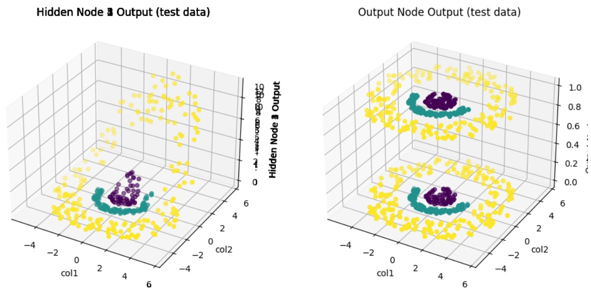

Figure 14: Test data

Conclusion

This report demonstrates the effectiveness of FCNNs in handling both linearly and nonlinearly separable datasets. The results highlight the importance of selecting an appropriate architecture based on the nature of the data. Future work could explore more complex architectures, such as deeper networks or convolutional layers, to further improve performance on complex datasets.