# Neural Network Model for Classification

## Technique

Implementing a neural network for classification using a dataset is the task at hand. Preprocessing of the dataset involved standardizing numerical features and encoding categorical ones. Training (70%), validation (15%), and test (15%) sets of data were separated. The neural network had a softmax output layer for multi-class classification and a single hidden layer with 32 neurons (ReLU activation). The sparse categorical cross entropy loss function and learning rate of 0.01 were employed in the Adam optimizer. Training was abruptly stopped after five epochs in which the validation loss did not improve.

## Results

The model achieved the following metrics on the test dataset:

Test Loss: 0.1148

Test Accuracy: 96.0%

Precision:

Class 0: 0.97

Class 1: 0.88

Recall:

Class 0: 0.99

Class 1: 0.63

F1-Score:

Class 0: 0.98

Class 1: 0.73

Macro Average:

Precision: 0.92, Recall: 0.81, F1-Score: 0.85

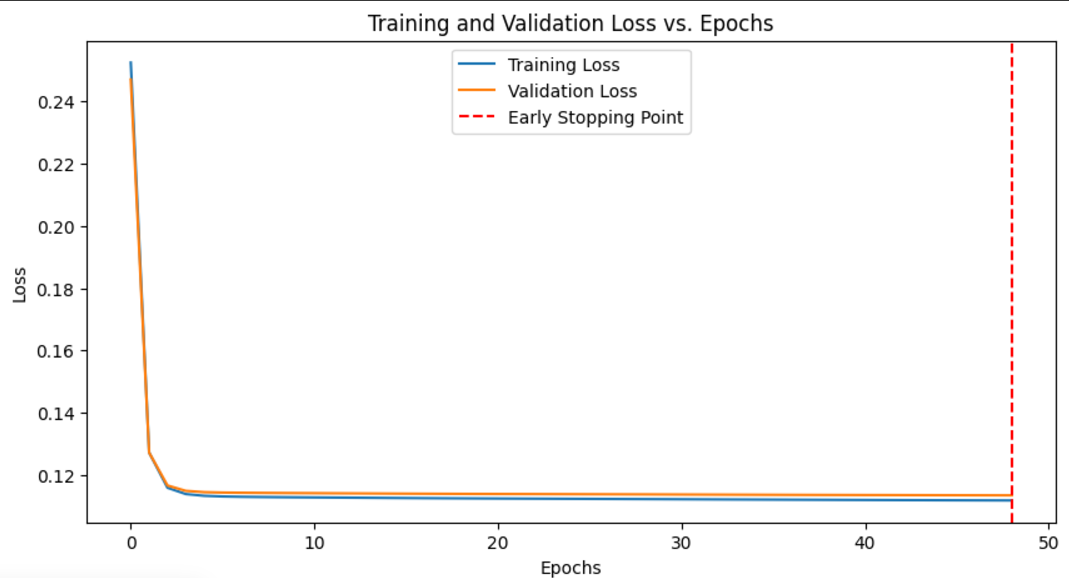
Weighted Average:

Precision: 0.96, Recall: 0.96, F1-Score: 0.96

The training and validation loss curves showed that the model stopped improving after epoch 49, where early stopping was triggered.

## Discussion

The classification test was successfully completed by the neural network, which showed excellent accuracy and precision. Consistent validation loss trends were ensured by early halting, which avoided overfitting. Additional hidden layers might be explored, hyperparameters like learning rate could be changed, and regularization strategies could be used. All things considered, the task demonstrated the value of neural networks and the significance of early halting in reaching generalization.

****