Intrusion Detection Using Feedforward Neural Networks with MATLAB

Babu Pallam (P2849288)
MSc. Artificial Intelligence

Executive Summary

This project focused on creating and testing a Neural Network-Based Intrusion Detection System (IDS) to identify normal and suspicious network activity. The KDD Cup 1999 dataset, which includes 41 features covering 22 types of attacks and normal connections. This was used to build an effective feedforward neural network by testing and experimentations.

Objective and Approach

The goal was to develop an IDS that could accurately classify network activity as normal or intrusive. The approach included:

- **Data Analysis** The Supervised KDD dataset was studied to understand its structure, types of features, and class distribution.
- **Data Preparation** Given the size and complexity of the KDD dataset, thorough data preprocessing was essential. Key steps included encoding categorical variables, handling any missing data, and normalizing continuous features to ensure the model could effectively process and learn from the data. The data was then split into training and testing sets, and cross-validation was carried out to make the model more reliable.
- **Model Design and Tuning** A feedforward neural network was built and its hyperparameters were tuned using Bayesian Optimization. Techniques like early stopping were used along with 5-fold cross-validation to prevent overfitting.

Results and Insights

The model achieved high accuracy, with the best configuration reaching 98.85%, though the best observed solution achieved an accuracy of 99.88%, albeit with higher training time and complexity. The results showed that the model could handle both seen and unseen data effectively.

As part of further experimentation, several steps were taken. First, the entire dataset "kddcup.data.gz" was used to test the best final models (2 models received) on new, unseen data, and the models performed impressively. Next, the best final model was adapted to handle multi-class classification, and its performance was observed, confirming some existing challenges. Finally, Bayesian Optimization was conducted without cross-validation, and the results were compared with previous optimization efforts, leading to important observations.

Conclusion and Future Work

The project showed that neural networks could effectively detect network intrusions with good accuracy and reliability. Future improvements will focus on using more advanced models, like Convolutional or Recurrent Neural Networks, using more advanced techniques for handling class imbalance, and experimenting with other tuning methods, such as Grid Search or Evolutionary Algorithms, to further enhance performance and address the current challenges.