Introduction

In this experiment, we are given a real dataset as well as labels and are going to develop a model that predicts probabilities for six different labels:

1. Training set size: 108,904 data instances
2. Test set size: 105,560 data instances

Methodology

To model this problem, a multilayered perceptron is used with 4 intermediate layers and ReLU as the activation function at each node. Finally, the sigmoid activation function is used in order to get a probability between 0 and 1. Since there was some overfitting, one of the layers had a dropout.

Libraries Used

- Matplotlib
- Numpy
- SKLearn
- CSV
- Pandas
- Pytorch

Analysis

Unfortunately, the data is very noisy and there is lots of information that is missing. In the MLP model, all inputs are required to be numerical so the data needed a lot of preprocessing. In order to process the data, the following was done:

1. **Differentiating between continuous and categorical variables**: Some of the features had a very small range of unique values. Thus, they seemed to represent categories more so than continuous variables. Therefore, for all features, if they had less than 40 unique values and more than 2, the feature is one-hot encoded.
2. **Dropping non numerical values**: Unfortunately, the dataset has quite an amount of non-numerical datasets. After trying multiple approaches to deal with them, dropping them was what gave the best results.
3. **Filling NA values with interpolated values: T**he missing values were replaced with the average of the column they are in.
4. **Normalizing the data:** Since the data has quite a large range, the features need to be normalized. To normalize all the data, we subtract the minimum of each column and divide by the maximum minus the minimum of each column.
5. **Polishing the data:** Finally, there is a very few columns at the end that are strings that have a lot of unique values. These strings cannot of that much use to the MLP model, so they were also dropped.

The Model

The training model that is used is a multilayer neural network consisting of 5 hidden layers as follows:

- Layer 1 – 256 nodes
- Layer 2 – 128 nodes, with a dropout of 0.3 (to counteract overfitting)
- Layer 3 – 64 nodes
- Layer 4 – 32 nodes

All the 5 layers have ReLU as their activation function. For the output layer, sigmoid is the activation function. Finally, the hyperparameters were chosen as follows after intensive testing:

- Number of epochs: 6 – This seemed to be the optimal value after investigation
- Batch size: 64 – Lower and higher values gave slightly worse results
- Loss function – BCELoss, or Binary Cross Entropy as we have a binary classification problem
- Optimizer – Adam
- Learning rate – 0.0001

Results

On the validation set, the AUROC was measured to be the following:

- Problem 1: 0.73
- Problem 2: 0.76
- Problem 3: 0.78
- Problem 4: 0.82
- Problem 5: 0.72
- Problem 6: 0.76

Conclusion

The MLP proved to be a good model since the AUROC is significantly higher than 0.5. To improve these results, one could further try to analyze each feature and perhaps get rid of more features that prove to be uncorrelated or of very low correlation in order to reduce noise.