Project Report

By: Hussein Hammoud

Since this project is a Binary classification problem I decided to experiment with many methods and then on testing decide which method is best the methods I choose to use are:

1. Decision Tree
2. Random Forest
3. Neural Network
4. Histogram Gradient Booster Classifier

First of all I extracted the Data from the .CVS file then I cleaned the data using

df = df.apply(pd.to\_numeric, errors='coerce')

df = df.dropna()  
Which will Remove all data examples that are not numeric.  
then we split the data into input and output petameters and perform feature scaling using: {  
y = df.iloc[:,0]  
scaler = StandardScaler()  
X = scaler.fit\_transform(X) }

Decision Tree:

I evaluated the decision tree model at different depths and we can clearly see form the data that we reach the DT max accuracy and adding depth does not get us any better results.

Random Forest:

Testing Random Forest using multiple Number of Trees.

Histogram Testing:

I assessed Histogram Gradient using a variation of max iteration, depth, and min leaf nodes as well as different steps.

Here we can as we tweek the hyperparameters of the histogram Gradient the performance of the algorithm barely changes which means we reached the max accuracy we can from the given data set.

Then we go on to Neural Networks:

I tested neural network by comparing different configurations:

Test 1: 3 Layers 2 of size 64 and the last of size 1 with sigmoid function.

Test 2: 3 Layers the first of size 64 with relu function the second of size 32 with sigmoid function and the last of size 1 also sigmoid.

Test 3: 3 Layers same as the second but increasing the sizes of the layers to 128 and 64, respectively.

Test 4: 5 Layers the first 2 of size 128 with relu and the second 2 of size 64 with sigmoid with last layer of size 1.

Test 5: 5 Layers same as test 4 except for the first layer being of size 256 with all the rest increased to 128.

As we can see from the data Test 5 gave the best results and is the Model chosen.