1. Hyperparameter tuning on subset of the larger dataset(0.1% training data)

There are 39763 rows in the smaller training set, and 10083 in the smaller test set. (Required for observation)

a) **Random Forest**

- Grid choices for Random Forest model-

maxDepth- [1, 5, 10])

maxBins- [10, 20, 50])

subsamplingRate- [0.1, 0.5, 0.9]

Accuracy for best Random Forest model after hyperparameter tuning on small train data = 0.504

- Best Random Forest model parameters-

"maxBins": 10,

"maxDepth": 10,

"subsamplingRate": 0.9

b) **Logistic Regression**

- Grid choices for Logistic Regression model-

elasticNetParam- [0.0, 0.3, 0.5, 0.7, 1.0]

regParam- [0.001, 0.01, 0.1]

maxIter- [25, 50, 100]

Accuracy for best Logistic Regression model after hyperparameter tuning on small train data = 0.503

- Best Logistic Regression model parameters-

"elasticNetParam": 0.3,

"maxIter": 25,

"regParam": 0.01

c) **Multilayer Perceptron Classifier**

- Grid choices for Multilayer Perceptron Classifier model-

stepSize- [0.1, 0.01, 0.001]

blockSize- [32,64,128]

maxIter- [10, 25, 100]

Accuracy for best Multilayer Perceptron Classifier model after hyperparameter tuning on small train data = 0.512

- Best Multilayer Perceptron Classifier model parameters-

"blockSize": 128,

"maxIter": 25,

"stepSize": 0.1

2. **Testing on Larger Dataset after hyperparameter tuning on smaller dataset**

There are 5000000 rows in the full training set, and 1000000 in the full test set

i) **10 cores**

P.S.- **AUROC** stands for Area under Receiver Operating Characteristic Curve and

AUPRC stands for Area under Precision Recall Curve.

Model name	Training time	Testing time	Accuracy	AUROC	AUPRC
Random Forest	134.58 sec	4.85 sec	0.500543	0.501	0.500
Logistic Regression	46.18 sec	4.11 sec	0.499462	0.500	0.499
Multilayer Perceptron Classifier	176.26 sec	3.22 sec	0.499811	0.500	0.499

ii) **5 cores**

Model name	Training time	Testing time	Accuracy	AUROC	AUPRC
Random Forest	208.14 sec	8.03 sec	0.500543	0.501	0.500
Logistic Regression	74.82 sec	4.92 sec	0.499462	0.500	0.499
Multilayer Perceptron Classifier	290.41 sec	6.04 sec	0.499811	0.500	0.499

3. **Interesting Observations**

i) Data split using randomSplit-

- Our raw training data has 5000000 rows of data and selecting 1% of data from 5M rows of data should give us 50k rows of data, but looking at the split from this smaller data and adding the counts from training(39763) and test data(10083) doesn't equal 50000(it equals 49846).
- This is a normal functionality of randomSplit(), and we can't get the same number of data points, only a close number. It is due to the sampling method used to create each split[1].

ii) Accuracy on trainset-

- Even though the models were trained on the training set, the accuracy computed on them after training is still roundabout in 50s.
- Accuracy for best Random Forest model on trainset was = 0.53157
  Accuracy for best Logistic Regression model on trainset was = 0.50017
  Accuracy for best Multilayer Perceptron Classifier model on trainset was = 0.500969

iii) 10 cores faster 37% -

- The training time for all models using 10 cores is less than the time it takes for these models to train using 5 cores and is around 37% faster on 10 cores than 5 cores.
- This is because the number of tasks running in parallel for 10 cores config is twice more than the number of tasks running in parallel for 5 cores config.

iv) Label -1 is invalid with model classifiers-

- When the models were trained on the data initially, the error- "Classifier was given dataset with invalid label -1.0. Labels must be integers in range [0, 2)" was thrown up.
- This is because our data has labels -1 and 1, but the Classifiers require labels as integers. Thus our labels which were '-1' were changed to value "0", and the issue was resolved.

References/Scouces:

[1] https://medium.com/udemy-engineering/pyspark-under-the-hood-randomsplit-and-sample-inconsistencies-examined-7c6ec62644bc