

4) Artificial Neural Networks - Multi Layer Perceptron: Write a program to use a Multi Layer Perceptron from sklearn for classifying *an outcome for any dataset*. The program should

- Contain functions to read the dataset from the file
- pre-process it for encoding categorical attributes
- split the data into train and test.
- Train the model.
- Predict the output sample test data.
- Perform Mini-Batch Training
- Plot the accuracy score for the train and test data.
- Print the confusion matrix, accuracy, precision, recall on train and test data

Data Set Used: [HR_comma_sep.csv](#)

Classification : [Will an employee stay or leave?](#)

Aim: Build an *employee churn prediction model* based on parameters in dataset, (predict if the employee will leave the organization or not) using **MLPClassifier**.

OUTPUT ANALYSIS

	satisfaction_level	last_evaluation	number_project	average_monthly_hours	time_spend_company	Work_accident	left	promotion_last_5years	sales	salary
0	0.38	0.53	2	157	3	0	1	0	sales	low
1	0.80	0.86	5	262	6	0	1	0	sales	medium
2	0.11	0.88	7	272	4	0	1	0	sales	medium
3	0.72	0.87	5	223	5	0	1	0	sales	low
4	0.37	0.52	2	159	3	0	1	0	sales	low

MLPClassifier requires numerical data. 'Sales','Salary' columns are categorical variables. They need to be encoded. It is done using sklearn **LabelEncoder** as shown below:

	Encoded Salary	Dept
0	1	7
1	2	7
2	2	7
3	1	7
4	1	7
...
14994	1	8
14995	1	8
14996	1	8
14997	1	8
14998	1	8

```
array(['sales', 'accounting', 'hr', 'technical', 'support', 'management',  
      'IT', 'product_mng', 'marketing', 'RandD', 7, 2, 3, 9, 8, 4, 0, 6,  
      5, 1], dtype=object)
```

There are 9 ‘Sales’ categories , in other words ‘Departments’

They are encoded as 0-IT, 1-Marketing, 2-Accounting’, 3-HR, 4-Management, 5-’Marketing’, 6-Product Manager, 7-Sales, 8-Support, 9-Technical

Hand-coded Mini-Batch learning

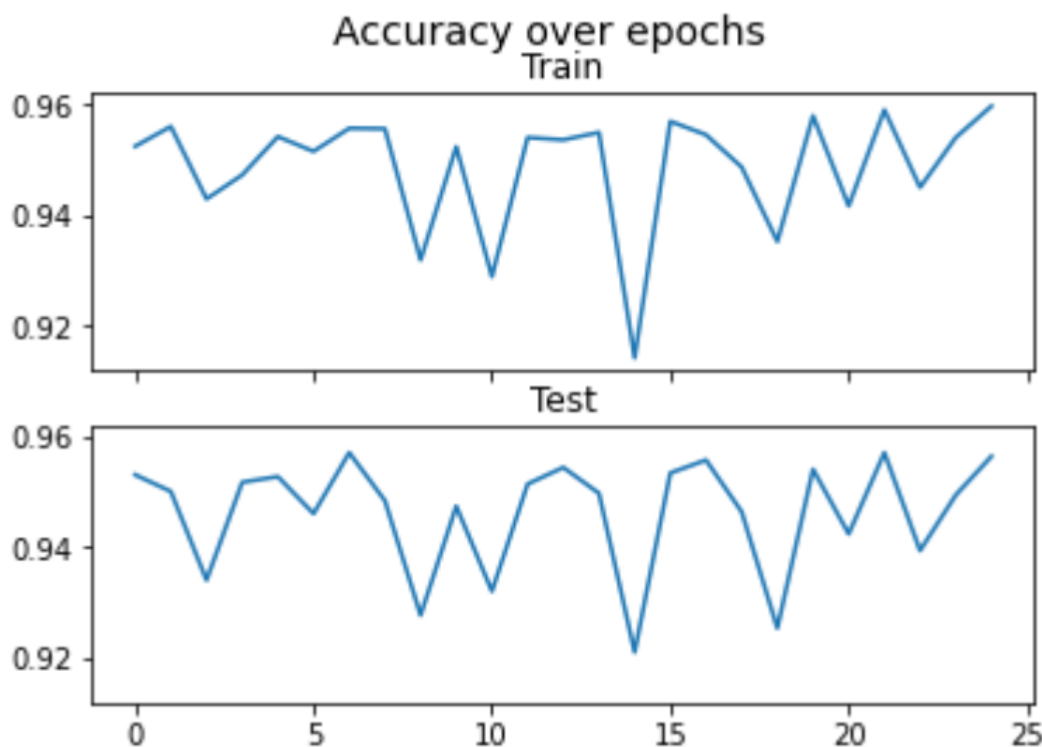
- N_TRAIN_SAMPLES will be got from the # of rows in shape[0] of the dataset
- N_EPOCHS and N_BATCH are hard-coded for this programs as 12 and 128 respectively
- N_CLASSES will be [0,1] as this is the outcome to be predicted '0' means employee will stay and '1' means employee will leave

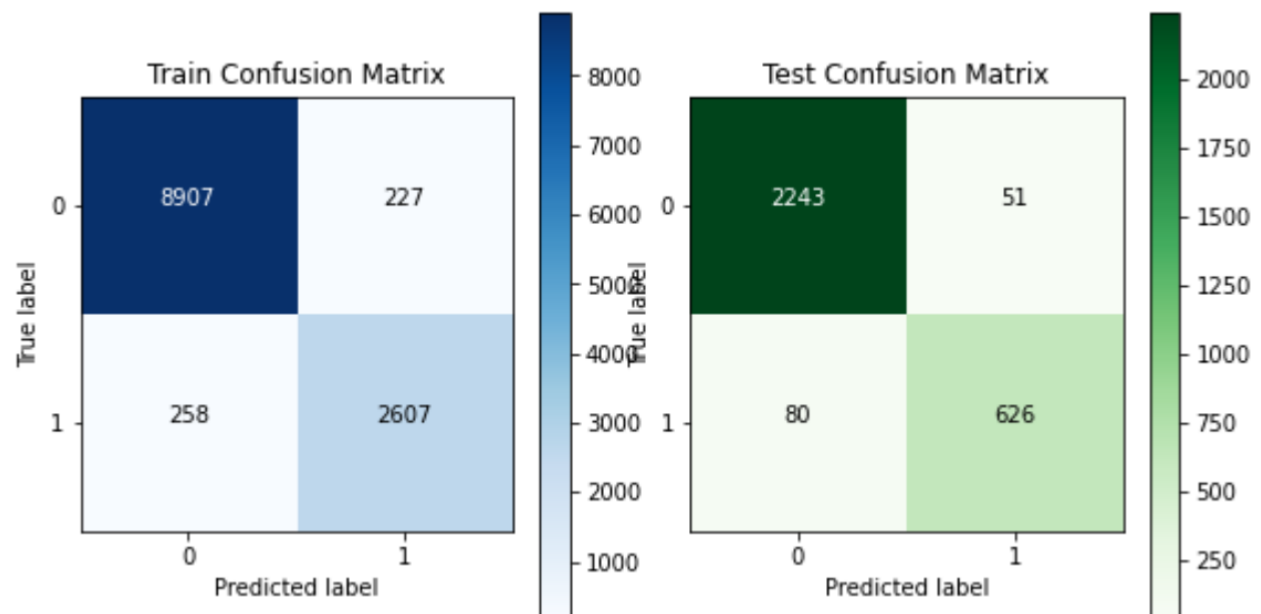
Mini Batch Training is done

- To ensure random # of samples for mini-batch training **random.permutation** function is used
- The training is done for N_EPOCHS. Here it is 25 epochs.

Plots

- The Accuracy score for the train and test dataset is plotted





<p>For Train Data</p> <p>TH 8907</p> <p>TP 2607</p> <p>FH 227</p> <p>FP 258</p>	<p>For Test Data</p> <p>TH 2243</p> <p>TP 626</p> <p>FH 51</p> <p>FP 80</p>
<p>Train Recall is 0.9199011997177134</p> <p>Train Precision is 0.9199011997177134</p> <p>Accuracy Score for Train Data is 0.9595799649970831</p>	<p>Test Recall is 0.886685552407932</p> <p>Test Precision is 0.9246676514032496</p> <p>Accuracy Score for Test Data is 0.9563333333333334</p>

```
testpred is : [1]
Employee will leave
```