INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR

Dependable & Secure AI-ML

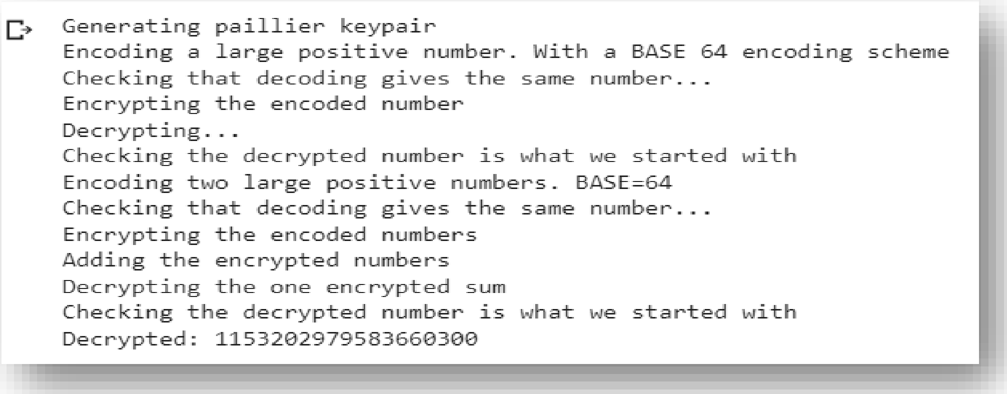
Assignment – 2

Name: GAJULA SAI CHAITANYA **ROLL NO**: 18CS30018

PART – 1:

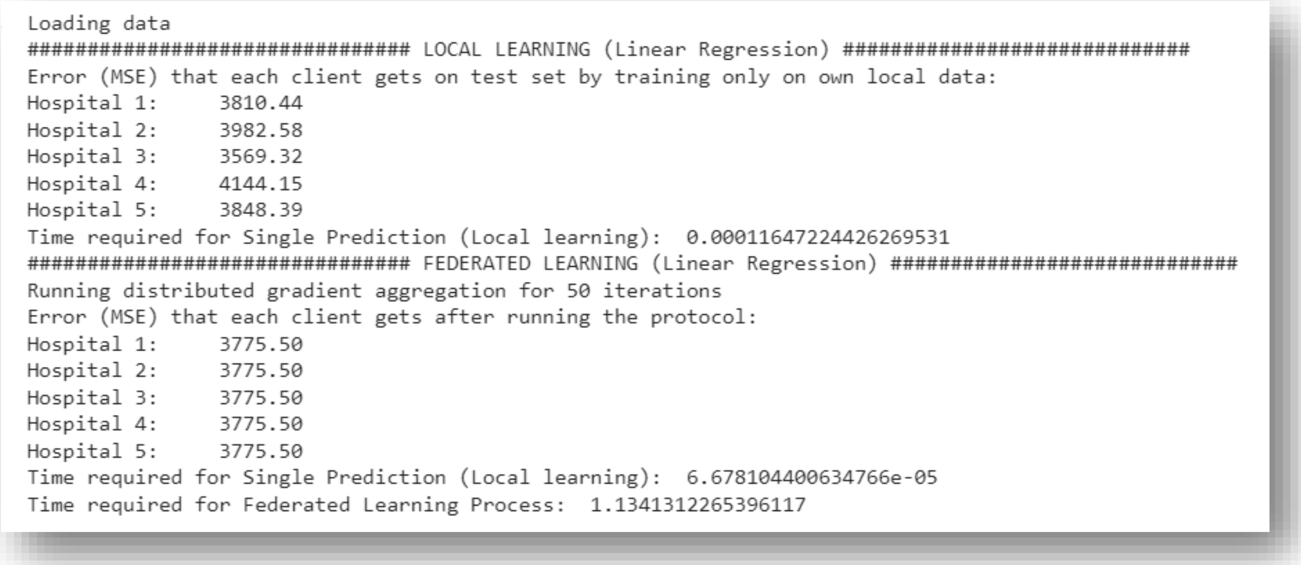
Running the python code and recording the timing required to compute the Federated Learning process.

i) Screenshot of running 'alternative_base.py'



```
Generating paillier keypair
Encoding a large positive number. With a BASE 64 encoding scheme
Checking that decoding gives the same number...
Encrypting the encoded number
Decrypting...
Checking the decrypted number is what we started with
Encoding two large positive numbers. BASE=64
Checking that decoding gives the same number...
Encrypting the encoded numbers
Adding the encrypted numbers
Decrypting the one encrypted sum
Checking the decrypted number is what we started with
Decrypted: 1153202979583660300
```

ii) Screenshot of running and recording time for federated learning process, used time.time() to get the time before and after the federated learning execution.



```
Loading data
################################ LOCAL LEARNING (Linear Regression) #############################
Error (MSE) that each client gets on test set by training only on own local data:
Hospital 1:     3810.44
Hospital 2:     3982.58
Hospital 3:     3569.32
Hospital 4:     4144.15
Hospital 5:     3848.39
Time required for Single Prediction (Local learning):  0.00011647224426269531
################################ FEDERATED LEARNING (Linear Regression) #############################
Running distributed gradient aggregation for 50 iterations
Error (MSE) that each client gets after running the protocol:
Hospital 1:     3775.50
Hospital 2:     3775.50
Hospital 3:     3775.50
Hospital 4:     3775.50
Hospital 5:     3775.50
Time required for Single Prediction (Local learning):  6.678104400634766e-05
Time required for Federated Learning Process:  1.1341312265396117
```

PART – 2:

Implementing privacy-preserving SVM assuming public model private data scenario (data in encrypted but model parameters are unencrypted):

- Used similar code structure as given in the repository, by incorporating SVM trained hinge loss as the loss function and accuracy as metric.

- Taken a spam classification dataset from online (The dataset is downloaded within the code itself, please refer to the code).

- From the screenshots, we can observe that, the federated learning achieves better performance (in terms of Hinge loss) and the time required for single prediction is lesser than that required for local learning. (In the below outputs, Hospital – n refers to nth client in the federated setup)

- Total time required for federated learning process: 0.2457757

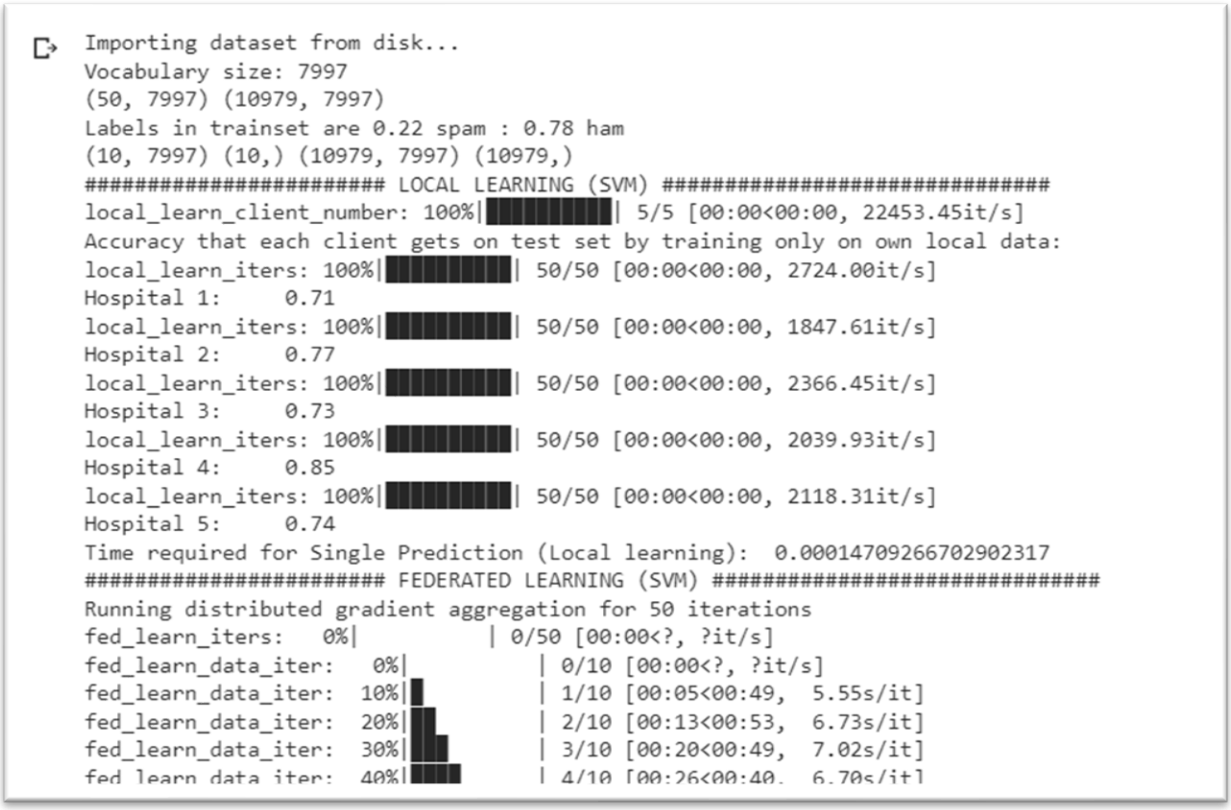
- Time required for single prediction in local learning: 0.000147
  Time required for single prediction in federated learning: 0.000106

- Accuracy values for each of the clients for Local and Federated setups.

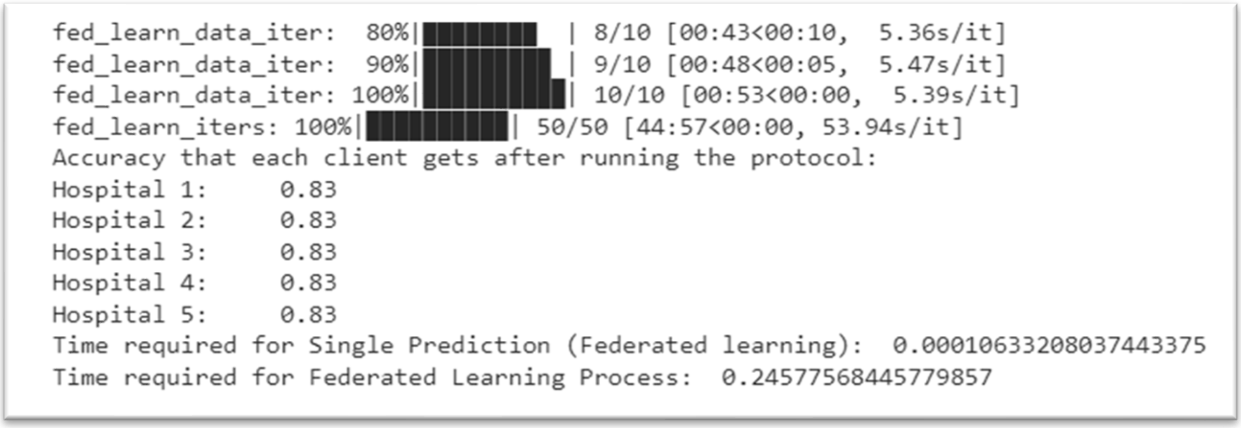
Client	**Local**	**Federated**
Hospital – 1	0.71	0.83
Hospital – 2	0.77	0.83
Hospital – 3	0.73	0.83
Hospital – 4	0.85	0.83
Hospital – 5	0.74	0.83
Avg. Accuracy	0.76	0.83

(**Note:** All the clients have same global model at the end of federated learning, that is the reason behind same accuracy values for all the clients)

- Screenshots for PART - 2 are attached in the next page



```
Importing dataset from disk...
Vocabulary size: 7997
(50, 7997) (10979, 7997)
Labels in trainset are 0.22 spam : 0.78 ham
(10, 7997) (10,) (10979, 7997) (10979,)
######################## LOCAL LEARNING (SVM) ###############################
local_learn_client_number: 100%|██████████| 5/5 [00:00<00:00, 22453.45it/s]
Accuracy that each client gets on test set by training only on own local data:
local_learn_iters: 100%|██████████| 50/50 [00:00<00:00, 2724.00it/s]
Hospital 1:     0.71
local_learn_iters: 100%|██████████| 50/50 [00:00<00:00, 1847.61it/s]
Hospital 2:     0.77
local_learn_iters: 100%|██████████| 50/50 [00:00<00:00, 2366.45it/s]
Hospital 3:     0.73
local_learn_iters: 100%|██████████| 50/50 [00:00<00:00, 2039.93it/s]
Hospital 4:     0.85
local_learn_iters: 100%|██████████| 50/50 [00:00<00:00, 2118.31it/s]
Hospital 5:     0.74
Time required for Single Prediction (Local learning):  0.00014709266702902317
######################## FEDERATED LEARNING (SVM) ###############################
Running distributed gradient aggregation for 50 iterations
fed_learn_iters:   0%|          | 0/50 [00:00<?, ?it/s]
fed_learn_data_iter:   0%|          | 0/10 [00:00<?, ?it/s]
fed_learn_data_iter:  10%|█         | 1/10 [00:05<00:49,  5.55s/it]
fed_learn_data_iter:  20%|██        | 2/10 [00:13<00:53,  6.73s/it]
fed_learn_data_iter:  30%|███       | 3/10 [00:20<00:49,  7.02s/it]
fed_learn_data_iter:  40%|████      | 4/10 [00:26<00:40,  6.70s/it]
```



```
fed_learn_data_iter:  80%|████████  | 8/10 [00:43<00:10,  5.36s/it]
fed_learn_data_iter:  90%|█████████ | 9/10 [00:48<00:05,  5.47s/it]
fed_learn_data_iter: 100%|██████████| 10/10 [00:53<00:00,  5.39s/it]
fed_learn_iters: 100%|██████████| 50/50 [44:57<00:00, 53.94s/it]
Accuracy that each client gets after running the protocol:
Hospital 1:     0.83
Hospital 2:     0.83
Hospital 3:     0.83
Hospital 4:     0.83
Hospital 5:     0.83
Time required for Single Prediction (Federated learning):  0.00010633208037443375
Time required for Federated Learning Process:  0.24577568445779857
```