

In this homework, you will implement algorithms to fine tune hyperparameters for the image classification task using a subset of the Federated EMNIST dataset[1] with samples of digits only (10 classes). This is the **same** dataset as Assignment 3, which you can find uploaded to Canvas.

1. You can use any neural network architecture for the image classification task.
2. You should use stochastic gradient descent (SGD) as the optimizer with mini-batches to train the network.
3. In the dataset, use `train_X.npy`, `train_y.npy` for training and split them into 80% of training and 20% of validation set. Use `test_X.npy`, `test_y.npy` for testing. The test data should not be used in any part of the training and the hyperparameter tuning.
4. Use the macro-averaged F1 score as the evaluation metric.

You will implement the following algorithms to fine tune the mini-batch size  $B$  and the activation function for the hidden layer to maximize the validation F1 score. The activation function should be one of {RELU, sigmoid, tanh}. The mini-batch size should be in range of [16, 1024].

## 1 Genetic Algorithm

Implement the Genetic Algorithm using the roulette rule, one-point crossover, and age based selection. You should implement the algorithm without using any packages specialized for genetic algorithms.

- Plot the average and the highest fitness score of the population versus the generation of the population.
- Select the vector with the highest fitness score in the last generation of the population. Report the hyperparameters.
- Train the model on the training and validation data using the selected hyperparameters. Evaluate the model on the test data. Plot the training F1 score versus the training epochs. Report the test F1 score.

## 2 Bayesian Optimization

Use the package [here](#) to apply Bayesian Optimization to fine tune the mini-batch size  $B$  and the activation function. The black-box function should be defined as the validation F1 score at the convergence of the model trained by the specified  $B$  and the activation function.

- Report the progress output and the final values of the hyperparameters found by the algorithm.
-

- Train the model on the training and validation data using the selected hyperparameters. Evaluate the model on the test data. Plot the training F1 score versus the training epochs. Report the test F1 score.
- Compare the hyperparameters chosen by the Genetic Algorithm and the Bayesian Optimization method. Briefly discuss the pros and cons of the two methods.

**Submission Guidelines:** Please submit your plots and answers to the questions in a pdf report on Canvas. Be sure to include a link to your repository or your GitHub username in your report.

## References

- [1] Sebastian Caldas, Sai Meher Karthik Duddu, Peter Wu, Tian Li, Jakub Konečný, H Brendan McMahhan, Virginia Smith, and Ameet Talwalkar. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.