**Implementation guid for HeteroSAg**

* For the results of HeteroSAg under heterogeneous quantization using CIFAR10 dataset

python3 HeteroSAg\_for\_quantization\_non\_IID\_CIFAR10.py --trainTotalRounds 250 --Batch\_size 0.2 --Epochs\_num 5 --Quantization\_type 'Homo' --Distribution non\_iid --lr 0.02 --K\_0 2 --K\_1 2 --K\_2 2 --K\_3 2 --K\_4 2 --nClients 100 --alpha 0.7

python3 HeteroSAg\_for\_quantization\_non\_IID\_CIFAR10.py --trainTotalRounds 250 --Batch\_size 0.2 --Epochs\_num 5 --Quantization\_type 'Hetro' --Distribution non\_iid --lr 0.02 --K\_0 2 --K\_1 6 --K\_2 8 --K\_3 10 --K\_4 12 --nClients 100 --alpha 0.7

python3 HeteroSAg\_for\_quantization\_non\_IID\_CIFAR10.py --trainTotalRounds 250 --Batch\_size 0.2 --Epochs\_num 5 --Quantization\_type 'no' --Distribution non\_iid --lr 0.02 --K\_0 2 --K\_1 6 --K\_2 8 --K\_3 10 --K\_4 12 --nClients 100 --alpha 0.7

The above commands will execute HeteroSAg for CIFAR10 with 100 clients. Furthermore, the distribution is non-IID across the clients, and follows Dirichlet distribution with parameterα.

* For the results of HeteroSAg under heterogeneous quantization using MNIST dataset under non-IID data distribution

Python3 HeteroSAg\_for\_quantization\_non\_IID\_MNIST.py --Quantization\_type 'no' --K\_0 2 --K\_1 6 --K\_2 8 --K\_3 10 --K\_4 12 --nClients 100

The above commands will execute HeteroSAg for MNIST. Furthermore, the distribution is non-IID across the clients. We set the number of epochs to be 5, use a batch size of 240, and constant learning rate 0.03.

* For the results of HeteroSAg for Byzantine robustness using CIFAR10 with non-IID dataset

python3 HeteroSAg\_for\_Byzantine\_CIFAR10\_non\_IID.py --alpha 0.7 --case 2 --case1 2

The above commands will execute HeteroSAg for CIFAR10 under non IID distribution with parameter alpha for N =300 clients. The code is set for four epochs and learning rate 0.03. The possible list for cases is [ non-attack, Gaussian, sign\_flip] where for example case = 2 gaussian. Case1 is “mean” while case2 is the “defense with median”.

* For the results of HeteroSAg for Byzantine robustness using CIFAR10 with IID dataset

python3 HeteroSAg\_for\_Byzantine\_CIFAR10\_IID.py --case 2 --case1 2

The above commands will execute HeteroSAg for CIFAR10 under IID for N =300 clients. The code is set for four epochs and learning rate 0.03. The possible list for cases is [non-attack, Gaussian, sign flip, label flip] where for example case = 2 gaussian. Case1 is “mean” while case2 is the “defense with median”.

* For the results of HeteroSAg for Byzantine robustness using MNIST with IID dataset

python3 HeteroSAg\_for\_Byzantine\_MNIST.py --case 2 --case1 2

The above commands will execute HeteroSAg for MNIST under IID for N =300 clients. The code is set for 1 epoch using learning rate 0.06. The possible list for cases is [non-attack, Gaussian, sign flip, label flip] where for example case = 2 gaussian. Case1 is “mean” while case2 is the “defense with median”.