Learning and Forgetting in Neural Networks

Bhargav Krishnamurthy[1], Yutong Luo[2], and Anjali Singh[3], Marcin Abram[4, 5]

[1]Computer Science Program, Viterbi, USC, [2]Physics and Linguistics Programs, Dornsife, USC, [3]Data Science and Cognitive Science Programs, Dornsife, USC
[4]Department of Physics and Astronomy, Dornsife, USC, [5]Information Sciences Institute, Viterbi, USC.

Abstract

Federated Learning is a way of learning that involves multiple client models and a community model. The clients are trained with their own data, and the community weights are updated with client weights. In this project, we investigate the effectiveness of federated learning when the data sets for each client are not independent from each other (non-IID regime). We aim to understand the mechanism of forgetting about weights in this process.

Non-IID Data Distribution

Each client has a local dataset. Those datasets can come from non-IID distribution. We model the difference between distributions by varying access to different labels. For each client, we sample data from **(a)** general distribution of all labels and **(b)** specific distribution of some labels. The non-independence factor is the ratio of the amount of general data to the amount of specific data of a client. In these graphs, Client 1 is fed more data with labels $\{0, 1\}$, Client 2 with $\{2, 3\}$, etc. A non-independence factor of 0 would mean that Client 1 only has data with labels $\{0, 1\}$, Client 2 with $\{2, 3\}$, etc. (this limit corresponds to the fully non-IID case).

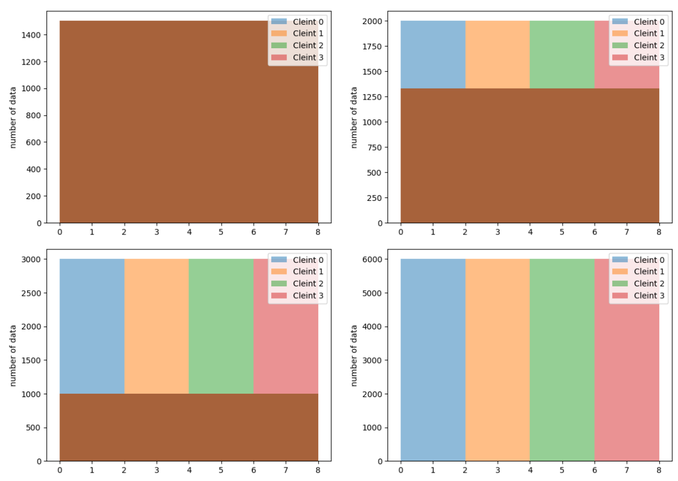


Figure 1. Data sets used for each client according to different non-independence factors. The figures have non-independence factors 1, 0.67, 0.33, 0, respectively from upper left to lower right.

Up to this point, the data that we use comes from the fashion-mnist data. It is a data set containing of grayscale images showing ten different types of clothings.

Federated Learning

- **Step 1: Initialization:** Initialize a global model, including parameters for client networks and a cloud network
- **Step 2: Local Training:** On each decentralized device, a copy of the global model is created and the local model is trained on the data available on that device.
- **Step 3: Model Updates:** After local training, the local models send only the model updates to the central server, not the raw data. These model updates represent the changes needed to improve the global model's accuracy.
- **Step 4: Aggregation:** The central server aggregates the received model updates from all participating devices or servers. Common aggregation methods include averaging or more complex methods.
- **Step 5: Global Model Update:** The central server updates the global model using the aggregated model updates. The global model now reflects the collective knowledge of all participating devices without ever having direct access to their data.

Key advantages of this algorithm include improved privacy, reduced communication overhead, and the ability to leverage decentralized data sources. However, handling non-IID data and synchronization issues can be challenging. It is also computationally very costly.

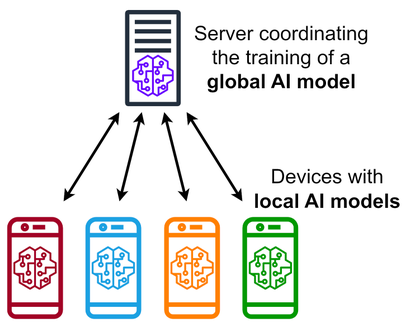


Figure 2. A schematic depiction of a federated learning workflow [1].

Learning with IID Data Distributions

As a comparison, we show the result of federated learning when clients have access to IID data distributions.

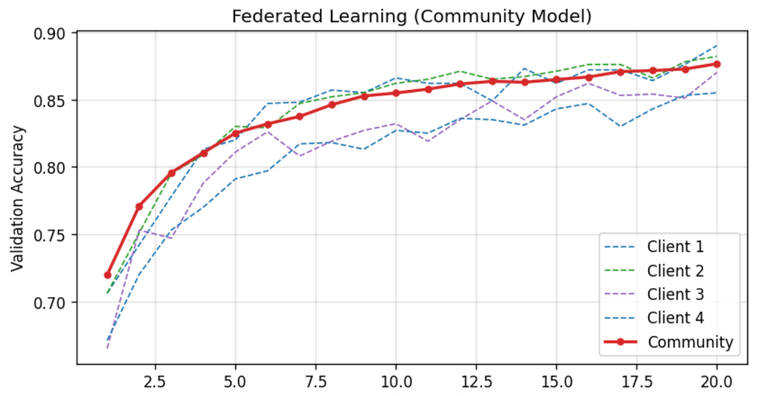


Figure 3. Accuracy of community model. Each client is trained on IID data distributions.

Learning with non-IID Data Distributions

To stabilize learning in non-IID regime, we vary the momentum parameter.

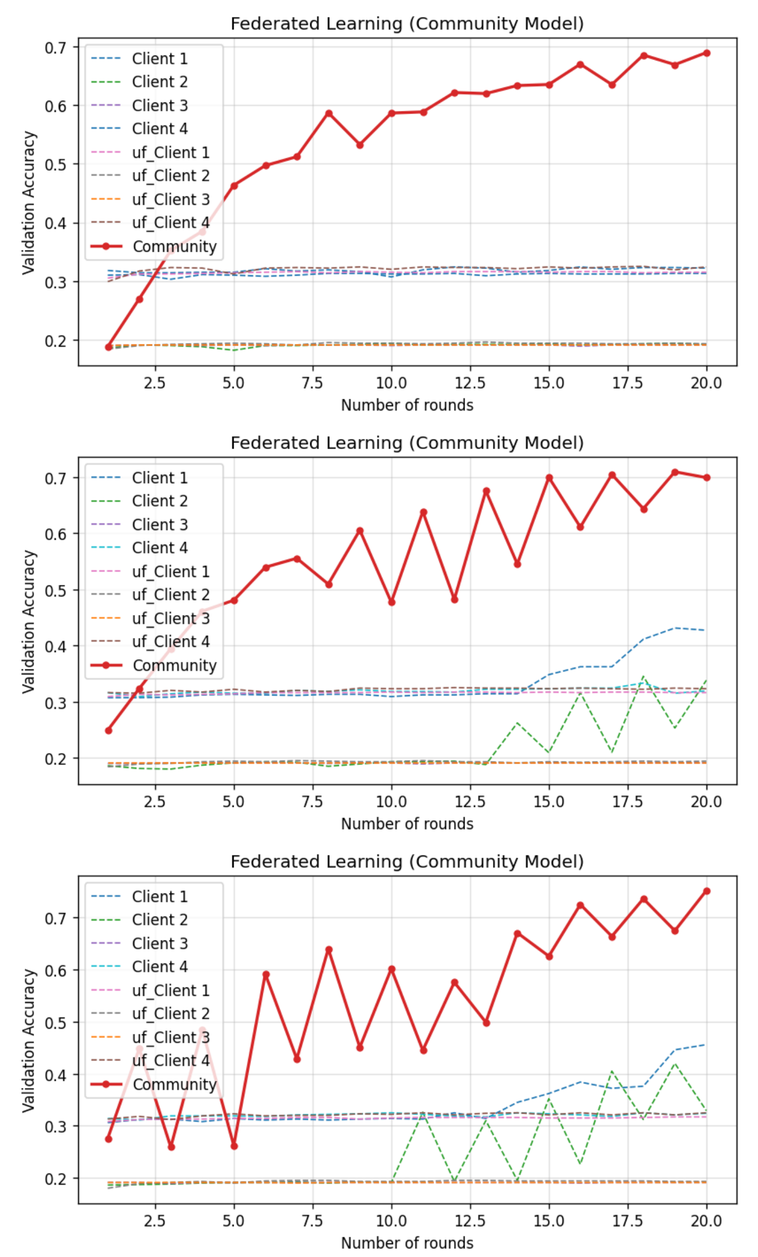


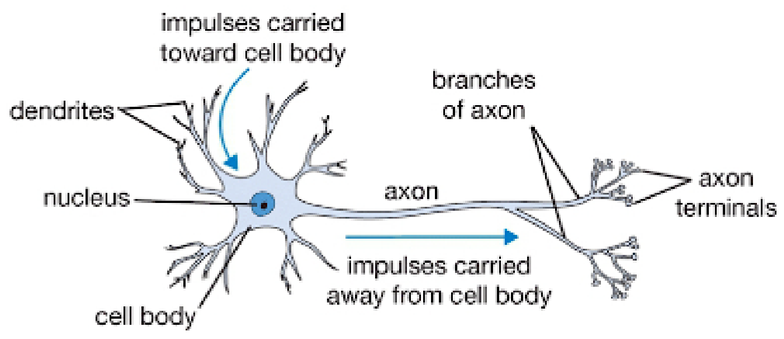
Figure 4. The uf clients are clients that do not update their weights with the weights of the community model. Clients 1 and 4 are given training data sets for 3 labels while the rest of them have 2. Respectively, in each update, the ratio of the original community model weights to the client weights average is 6:4, 4:6, 2:8. Some clients are given data on three labels while others two, hence the difference in validation accuracy.

Conclusions: We observe that using non-IID data for training does have a huge impact on the effect of the training process: At round 20, Federated Learning with IID data can reach a validation accuracy of 90%, while that with non-IID data can only reach 70%. We also observe that the clients that do not update their weights with the weights from the community model have about the same performance in each validation as the ones that do update their weights. This shows that the clients just forget about what they learn from the community model in each training round. Another trend that is shown in these graphs is that how aggressively we update the weights of the community model has a significant impact on the effect of training: as we give more weight to client weights over original community weights, we observe a large fluctuation in the validation accuracy for the community model, and we see changes in validation accuracy for the client models.

Neuron Activations

Neuron Activation Breakdown

Neurons receive input signals from other neurons and process information. Once the cumulative input exceeds a certain threshold, the neuron becomes activated. Activation results in an electrical signal known as action potential. This signal travels through a neuron's axon which then releases chemical messengers called neurotransmitters. Neurotransmitters bind to receptors on the next neuron which transmits a signal across the synapse. Within artificial neural networks, neuron activation is simulated through mathematical functions. Neurons receive weighted input signals, apply an activation function, and produce an output signal.



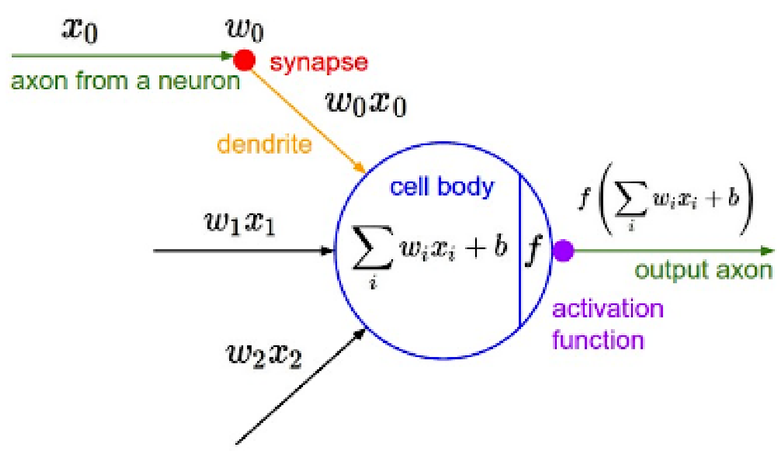


Figure 5. A schematic depiction of a neuron activation (adopted from [2]).

How does this tie into our Project?

Within our project, we calculated the SR distance [3] which is a measure of how different neurons in the client-side neural network respond to data from the local characteristic distribution. This helps identify which neurons are sensitive to the data. We used the SR-distance analysis to address significant differences in network activation areas in each client of federated learning.

Discussion and conclusions

Our problem statement deals with how Non-IID distributions can create weak diagnostics due to weight shifts within neural networks. Through our analysis so far, we have noticed that federated units start out with the accuracy of the community model but flatten and unlearn the material after local training. We have noticed that weight manipulation is a key factor because we need to measure a numerical metric that averages out the values more efficiently in order to avoid the problem. In the future we hope to find a more defined metric to compute competition among models to skew the distribution. Furthermore, certain computations, like the L2 distance, can be computationally intensive in the current form.

References

[1] Wikipedia, "Federated learning." Last accessed October 19, 2023.

[2] Towards Data Science. Last accessed October 19, 2023.

[3] S. Wang and Y. Zhang, "FedSiM: a similarity metric federal learning mechanism based on stimulus response method with non-IID data," *Measurement Science and Technology*, vol. 34, p. 125045, Sept. 2023.