FedTree: Federated Learning on Tabular Medical Data

Bhavesh Neekhra [1] Swapnanil Mukherjee [1] Vinayak Goswamy [1] Debayan Gupta [1]

[1]Ashoka University, India

Introduction

Data in the medical domain is often spread across multiple sites (labs / hospitals etc.). Owing to privacy and regulatory concerns, such data often cannot be shared or distributed [1]. The collection of such data is expensive, time-consuming, and subject to multiple situational factors (such as availability of subjects, expert technicians, and instrumentation). As such, it is infeasible to collect medical/clinical data on a large scale, which imposes a significant data-size constraint in applying ML methods to medical tasks.

A common solution to this is *Federated Learning* (FL) [2], which learns a shared model by aggregating locally-computed updates. However, we do not 'combine' individual models to create a centralized model. Instead, we take inspiration from tree-based ensemble methods and use majority voting of the local models to get a final prediction. As far as we are aware, this approach has not been used before for Federated Learning.

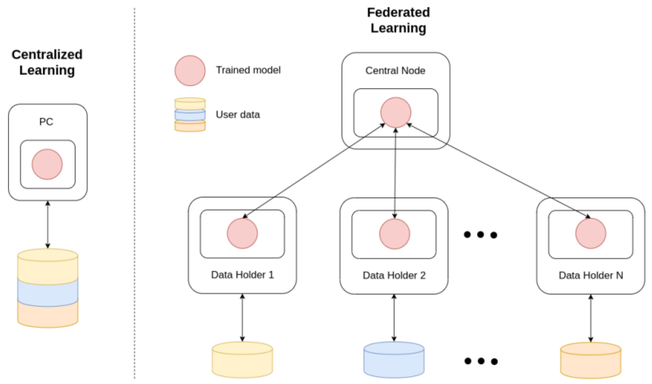

Figure 1. Conventional model training vs. federated model training

Stemformatics is an "easy-to-use and intuitive tools for biologists" [3]. In order to make the data from Stemformatics usable for ML applications, we collate, prepare and make available a unified stem-cell potency dataset. We combined 329 different stem cell datasets of heterogeneous structure hosted on Stemformatics consisting of RNASeq and Microarray gene expression data, and reduced the label set from 100+ stem cell types based on location to 4 types based on stem cell potency. After cleaning and preparation, the sample size in this data set is 3294 samples with 11980 genes.

Datasets

For the Stemformatics dataset [3], samples are divided into four stem-cell categories based on potency, namely- iPSC (896 samples), hESC (585 samples), hMSC (997 samples), and hUSC (816 samples). In the Breast Cancer Wisconsin (Diagnostic) dataset [4], the class labels are malignant (212 instances) and benign (357 instances). For the UCI Breast Cancer dataset [5] class labels are no-recurrence-events (201 instances) and recurrence-events (85 instances).

Table 1. Datasets

Dataset	# Samples	Numerical Features	Categorical Features	Classes
Stemformatics	3294	11980	0	4
Breast Cancer Wisconsin (Diagnostic)	569	30	0	2
UCI Breast Cancer	286	0	9	2

Methods

To get a measure of baseline performance, we use 5 classical ML models for the classification task for all three datasets. **Note: For the following plot, the models have access to the full data for training.**

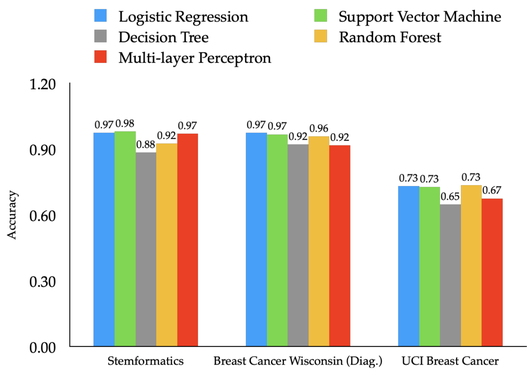

Figure 2. Performance of different models on each dataset with classical training.

Federated Learning

We experiment with classical ML models under a Federated Learning setup because tree-based methods are known to outperform Neural Networks on typical tabular data [6]. These models are also far less data-hungry than large neural network models, and are easier to train which becomes especially important in a federated setup since the local sites often may not have adequate resources to train resource-intensive models. These models are also interpretable and explainable, which is very crucial aspect in medical domain analysis. Our centralised model is an ensemble of the individual site models and to get a prediction for a sample, we take the majority vote of the predictions made by the individual models.

For the Stemformatics dataset, the classification task is: given a cell gene expression sample, classify it into one of the four stem cell classes. The Stemformatics dataset is a collection of multiple (329) smaller datasets. We separate the samples into 5 different sites containing roughly equal number of samples and the data is pre-processed locally at every site. After the initial splitting there is no interaction between the data of different sites which attempts to mimic what is usually observed in the real world.

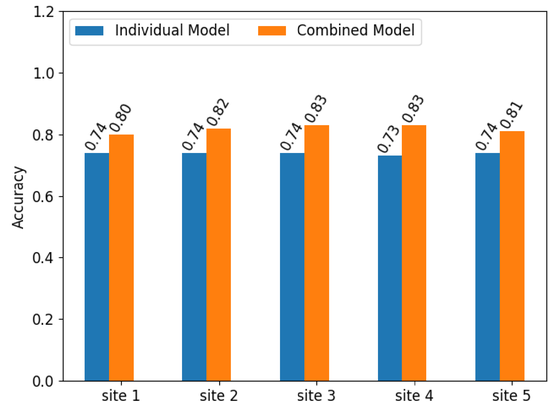

Figure 3. Performance comparison of local site models with ensemble model: Stemformatics

For the breast cancer datasets, we take a hypothetical setting of 3 different sites, containing roughly equal samples and the data is not shared across sites. We train a Decision Tree (DT) classifier at each site with its own data and measure the performance of the model at that particular site. We use only 3 sites instead of 5 because the number of samples in both these datasets is around 10X less than that in the Stemformatics, and we wanted every site to contain at least 100 samples.

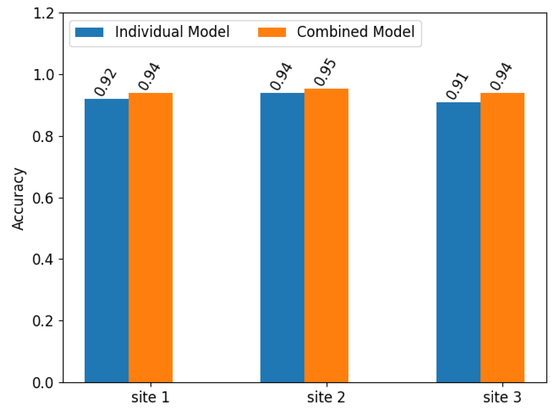

Figure 4. Performance comparison of local site models with ensemble model: Breast Cancer Wisconsin

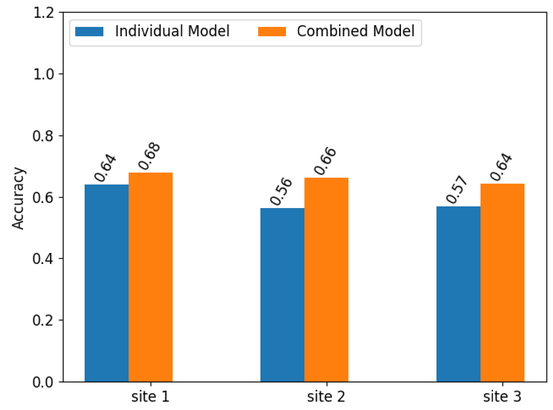

Figure 5. Performance comparison of local site models with ensemble model: Breast Cancer Dataset

Discussion

From Figures 3, 4, 5, it can be observed that Federated Learning on Decision Tree with majority voting improves the performance at each site when compared to the local model. Table 2 compares the FL ensemble model's performance with that of a model trained on the entire training dataset. From the results we can see that sample size and the type of features affect the models performance. UCI Breast Cancer dataset has only 286 samples with categorical features, and all models perform poorly on this dataset. For the other two datasets, all models' accuracy is more than 90%. For Stemformatics dataset, the FL ensemble model's accuracy is lower than that of a centralised model, which can be attributed to the loss in information due to the data-split across 5 sites.

Dataset	Centralised training with Decision Tree	Ensemble in Federated Learning
Stemformatics	88%	80-82%
Breast Cancer Wisconsin (Diagnostic)	92%	94-95%
UCI Breast Cancer	65%	64-68%

Table 2. Comparison of models' performance in centralised vs federated setting

Conclusion and Future Work

A unified gene expression dataset from Stemformatics is made available in a format suitable for training ML models. We also compare the FL-based ensemble model with conventional models. Then, We show that FL-based ensemble of Decision Trees improves the performance compared to a local model at individual sites. In this work, we have considered samples distributed across multiple sites, i.e., data is horizontally partitioned. For future work, we would like to experiment where each site has different set of features, i.e., data is vertically partitioned. Since taking the majority vote of predictions by local models is a simplistic approach, it can be beneficial to explore better ways of combining the local models into a centralized model.

Acknowledgements

This work was funded by the Mphasis F1 Foundation. The authors thank Shreyansh Priyadarshi, Debojyoti Chowdhury and Dr. Shubhasis Haldar for the helpful discussions.

References

[1] Hao Jin, Yan Luo, Peilong Li, and Jomol Mathew. A review of secure and privacy-preserving medical data sharing. *IEEE Access*, 7:61656–61669, 2019.

[2] Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-Efficient Learning of Deep Networks from Decentralized Data. In Aarti Singh and Jerry Zhu, editors, *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pages 1273–1282. PMLR, 20–22 Apr 2017.

[3] Jarny Choi, Chris M Pacheco, Rowland Mosbergen, Othmar Korn, Tyrone Chen, Isha Nagpal, Steve Englart, Paul W Angel, and Christine A Wells. Stemformatics: visualize and download curated stem cell data. *Nucleic Acids Research*, 2018.

[4] Matjaz Zwitter and Milan Soklic. Breast Cancer. UCI Machine Learning Repository, 1988. DOI: https://doi.org/10.24432/C51P4M.

[5] William Wolberg, Olvi Mangasarian, Nick Street, and W. Street. Breast Cancer Wisconsin (Diagnostic). UCI Machine Learning Repository, 1995. DOI: https://doi.org/10.24432/C5DW2B.

[6] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in Neural Information Processing Systems*, 35:507–520, 2022.

Scan for GitHub repository

bhavesh.neekhra_phd18@ashoka.edu.in