



Mini-Project

Peer-to-Peer (P2P) learning in
FedML using SCAFFOLD

Under the Guidance of:

Dr. Sathya Peri

Mrs. Piduguralla Manaswini

Ms. Saheli Chakraborty

By:

Shantanu Pandey
CS20BTECH11046



Introduction

Electronic Health Records (EHR) are used to extract knowledge for medical practices. We can use ML techniques to extract knowledge from electronic health data (EHR) to improve healthcare, leading to better outcomes for patients.

NEED: large amount of datasets from several locations

FOR: training these ML models with minimal errors

Problem:

These datasets contains sensitive informations about patients and sharing them can violate the laws related to privacy and sharing of data.

Solution:

Federated Learning



Federated Learning

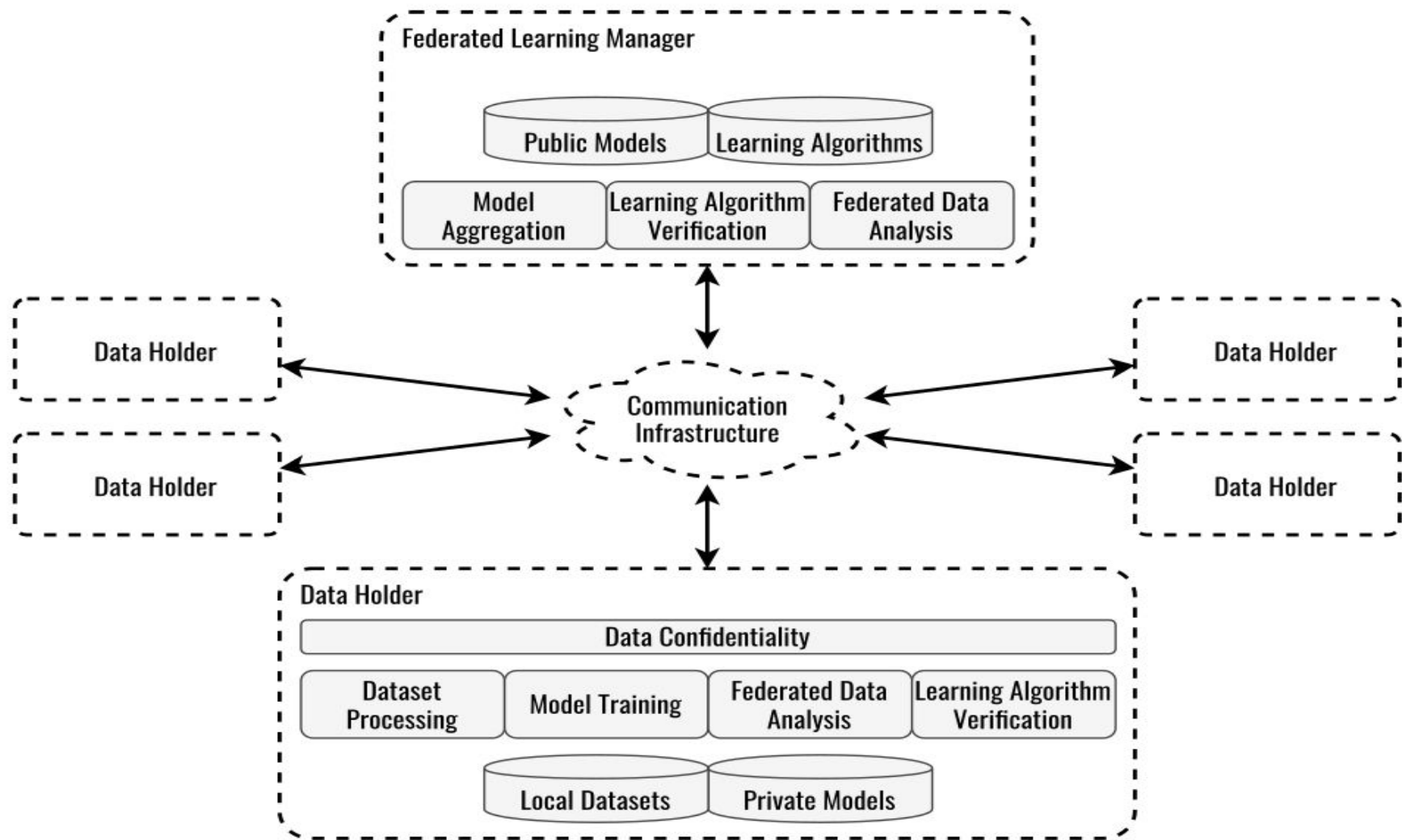
A methodology that enables the distributed training of machine learning models with remotely hosted datasets without the need to accumulate data and, therefore, compromise it or break any law.

HOW?

Training: Locally (at the location of the data)

Sharing: the resulting model, which is not reverse-engineerable, with the requesting party.

The shared models are then aggregated at a central server which results in a global model.





Problem statement:

Decentralized approach in Federated Machine learning.

Steps taken to tackle the problem statement:

1. Study the present literature corpus on “Fed-ML in Healthcare”
2. Study different aggregation methods used in “Fed-ML”
3. Develop a Fed-ML model which works on “peer to peer” architecture which is not affected by data heterogeneity or client sampling
 - a. Studying a latest method i.e. SCAFFOLD proposed for reducing effects due to data heterogeneity
 - b. Modifying the Algorithm to adapt a peer to peer learning model



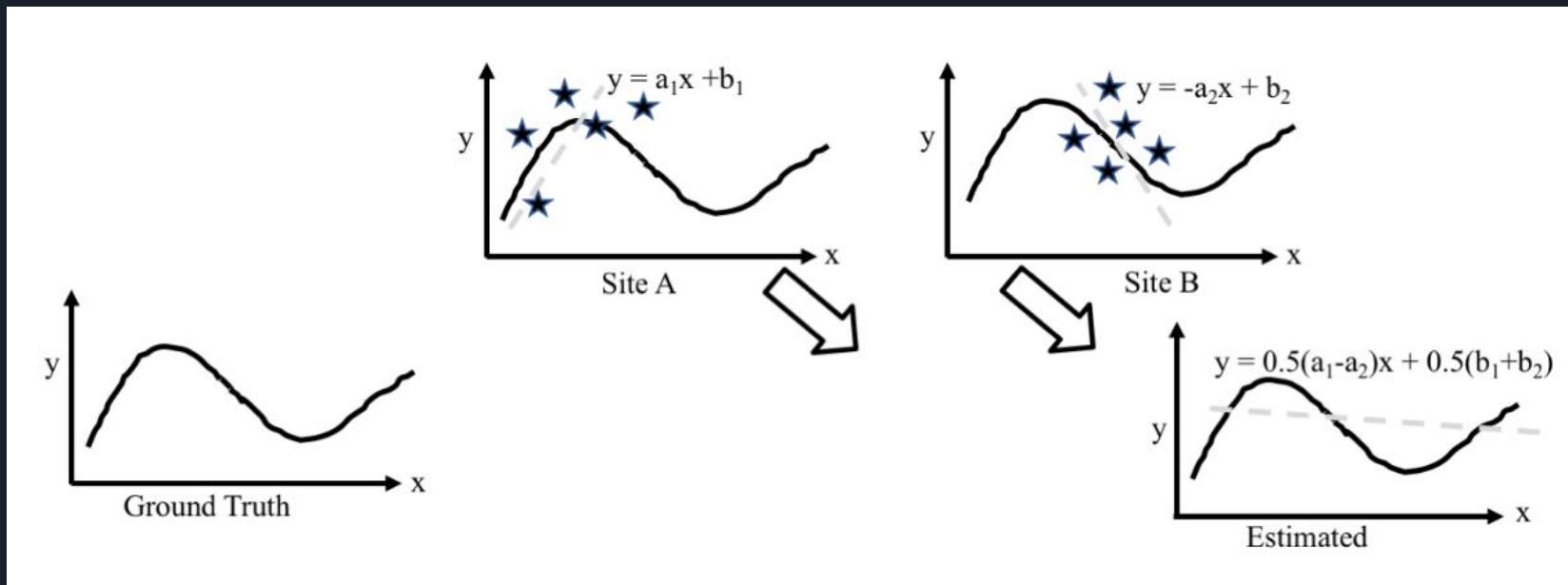
Step 1: Study on Fed-ML Literature corpus

Summary of five general topics:

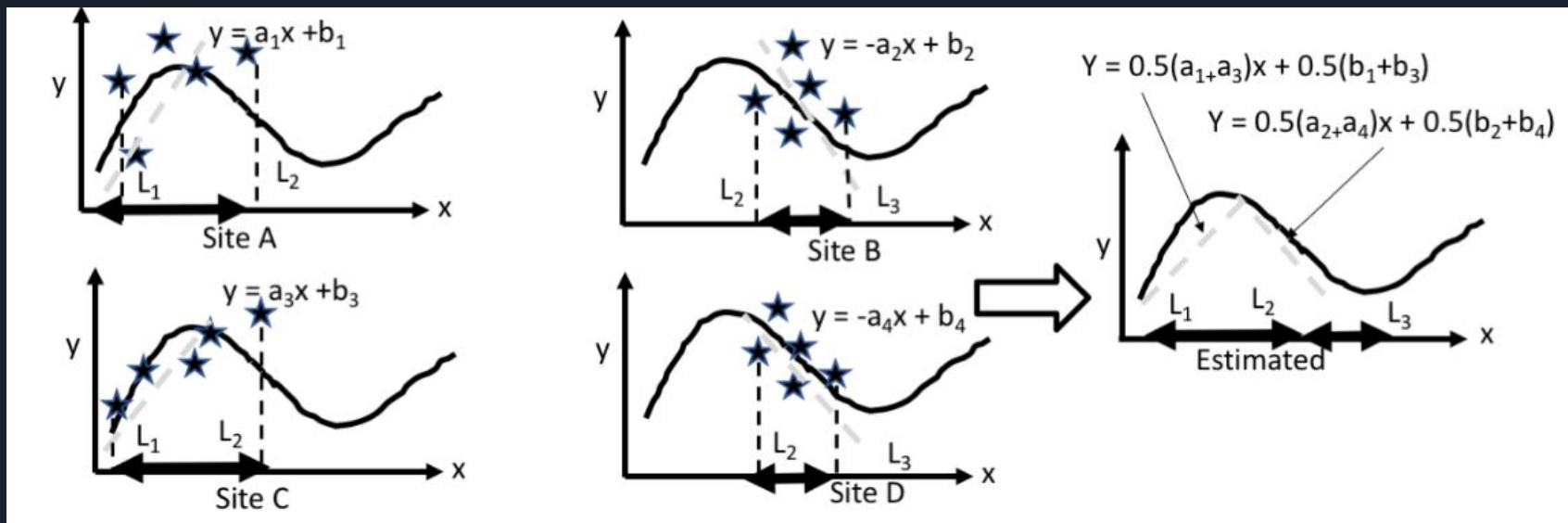
1. Case studies of FL on medical data:
 - a. Accuracy level similar to centralized C-ML (from a paper; paper link on the last slide)
 - b. Common technique: Deep Learning
2. Architectures for FL in medical applications:
 - a. Peer to Peer (P2P)
 - b. Server controlled
3. Confidentiality guarantees for training data:
 - a. Zero-trust models
 - b. Neural network models for non-interpretability
4. Aggregation methodologies for global model generation:
 - a. Characteristics of individual datasets can directly impact training results
 - b. Malicious participants
5. Federated data analysis
 - a. Generating anonymous copies of datasets
 - b. Dataset cleaning, model harmonization

Step 2: Dealing with Data Skew Problem in Fed-ML

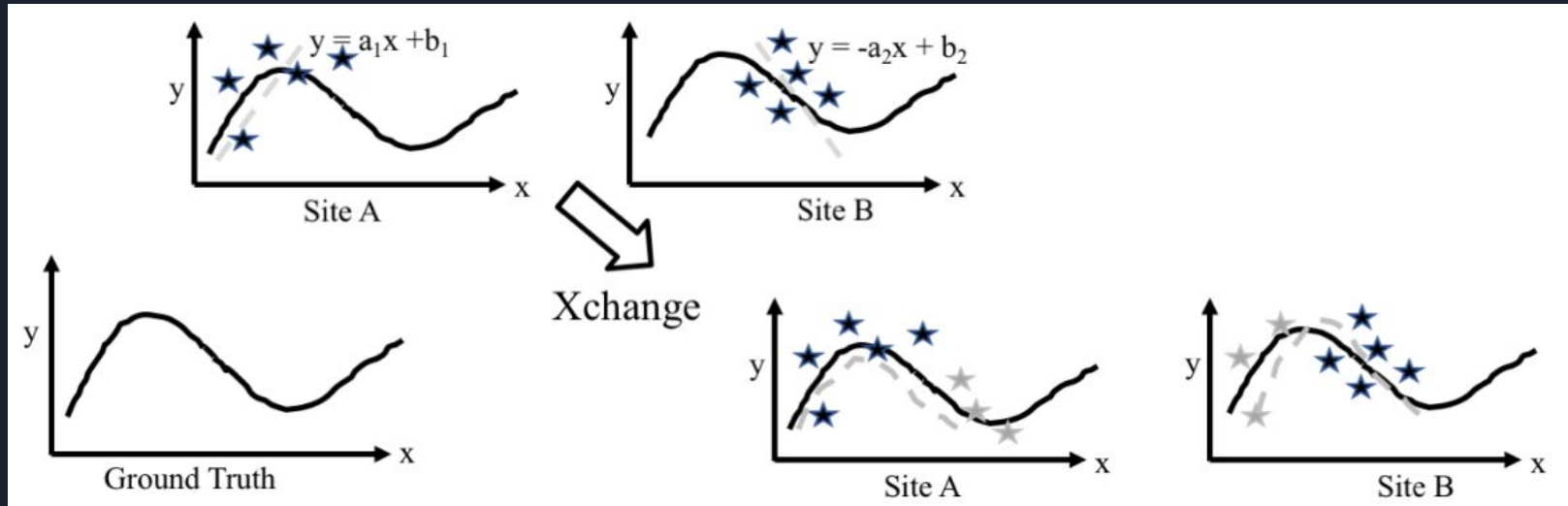
Data skewness can bring a large amount of error in the trained model accuracy.




1. Bounds-Aware Fusion



2. Bounds Expanding Data Exchange





3. Data Reconstruction for Bounds Expanding Data Exchange

1. Estimation:
 - a. Simple average (mean/median)
 - b. Creating a model of a feature using the other features available (logistic regression is likely to be sufficient)
2. Oversampling:
 - a. Independently at each site with the synthetic samples being swapped among partners.
 - b. Server the selecting of samples and swapping of averages over the selected data set.



Results:

Approach	Accuracy Range	Mean Accuracy	Standard Deviation
Site 1 alone	37% - 37%	37%	0%
Site 2 alone	37% - 37%	37%	0%
Site 3 alone	63% - 63%	63%	0%
Site 4 alone	37% - 37%	37%	0%
Naive Fusion	63% - 97%	71%	11%
Data Exchange	63% - 94%	84%	13%
Bounds Aware Method	91% - 91%	91%	0%



Step 3 (A): Studying SCAFFOLD

The earlier explained methods of dealing with data-skewness are not efficient for Dataset which has wide range of categories for values.

Root cause of PROBLEM:

Large heterogeneity (non-iid-ness) in the data present on the different clients

RESULTING IN:

drift in the updates of each client resulting in slow and unstable convergence.

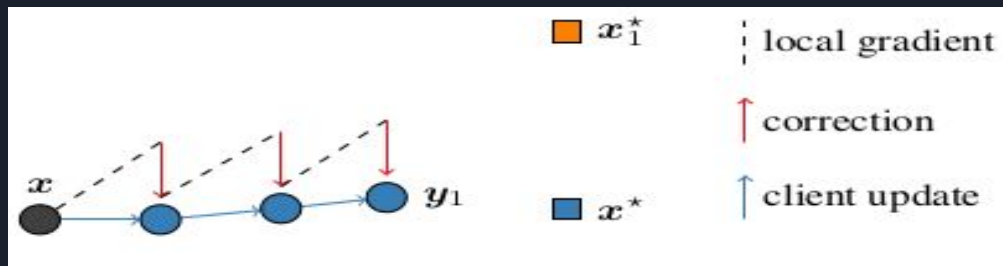
Solution:

Stochastic Controlled Averaging algorithm (SCAFFOLD): corrects the client-drift by estimating the update direction for the server model (c) and the update direction for each client $c(i)$. The difference $[c - c(i)]$ is then an estimate of the client-drift which is used to correct the local update.

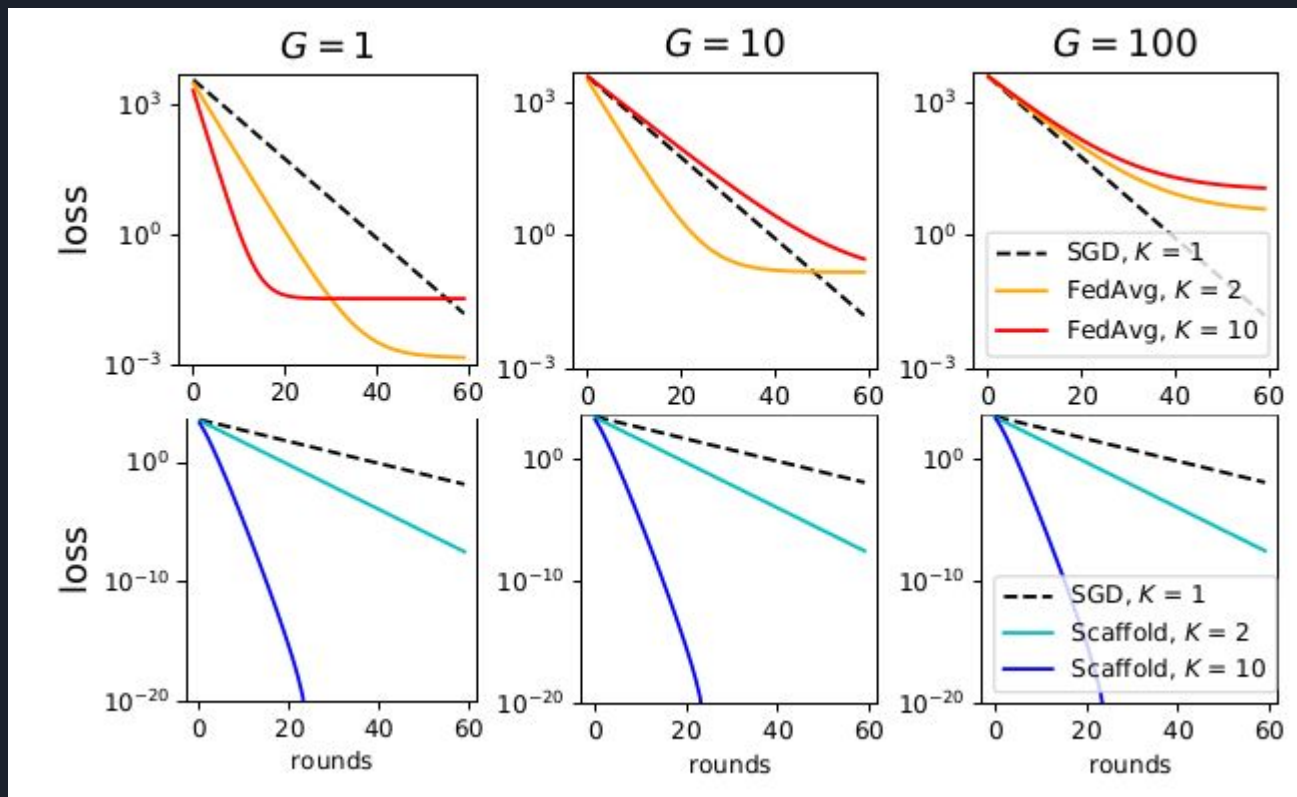
Visualization Of How SCAFFOLD prevents drift

SCAFFOLD estimates the update direction for the server model (c) and the update direction for each client c_i . The difference ($c - c_i$) is then an estimate of the client-drift which is used to correct the local update.

Update steps of SCAFFOLD on a single client:
The local gradient (dashed black) points to x_1^* (orange square/not true optimum), but the correction term ($c - c_i$) (in red) ensures the update moves towards the true optimum x^* (black square).



Comparison: (with SGD & FedAvg)



Changed SCAFFOLD Algorithm for P2P

server input: initial x and c , and global step-size η_g
client i's input: c_i , and local step-size η_l
for each round $r = 1, \dots, R$ **do**
 sample clients $S \subseteq \{1, \dots, N\}$
 communicate (x, c) to all clients $i \in S$

 on client $i \in S$ **in parallel do**
 initialize local model $y_i \leftarrow x$
 for $k = 1, \dots, K$ **do**
 compute mini-batch gradient $g_i(y_i)$
 $y_i \leftarrow y_i - \eta_l (g_i(y_i) - c_i + c)$
 end for
 $c_i^+ \leftarrow g_i(x)$, or $c_i - c + \frac{1}{K\eta_l}(x - y_i)$

 communicate $(\Delta y_i, \Delta c_i) \leftarrow (y_i - x, c_i^+ - c_i)$
 $c_i \leftarrow c_i^+$
 end on client
 $(\Delta x, \Delta c) \leftarrow \frac{1}{|S|} \sum_{i \in S} (\Delta y_i, \Delta c_i)$

 $x \leftarrow x + \eta_g \Delta x$ and $c \leftarrow c + \frac{\Delta c}{N}$
end for

Centralized Algorithm

In this approach I included file system to access parameters like x, c
main:
for each round $r = 1, \dots, R$ **do**
 sample clients $S \subseteq \{1, \dots, N\}$

 on client $i \in S$ **do**
 $(\Delta x, \Delta c) \leftarrow \frac{1}{|S|} \sum_{i \in S} (\Delta y_i, \Delta c_i)$
 $x \leftarrow x + \eta_g \Delta x$ and $c \leftarrow c + \frac{\Delta c}{N}$

 initialize local model $y_i \leftarrow x$
 for $k = 1, \dots, K$ **do**
 compute mini-batch gradient $g_i(y_i)$
 $y_i \leftarrow y_i - \eta_l (g_i(y_i) - c_i + c)$
 end for
 $c_i^+ \leftarrow g_i(x)$, or $c_i - c + \frac{1}{K\eta_l}(x - y_i)$

 $(\Delta y_i, \Delta c_i, c_i) \leftarrow (y_i - x, c_i^+ - c_i, c_i^+)$
 end on client
end for

Decentralized Algorithm



Each of the parameters mean the following:

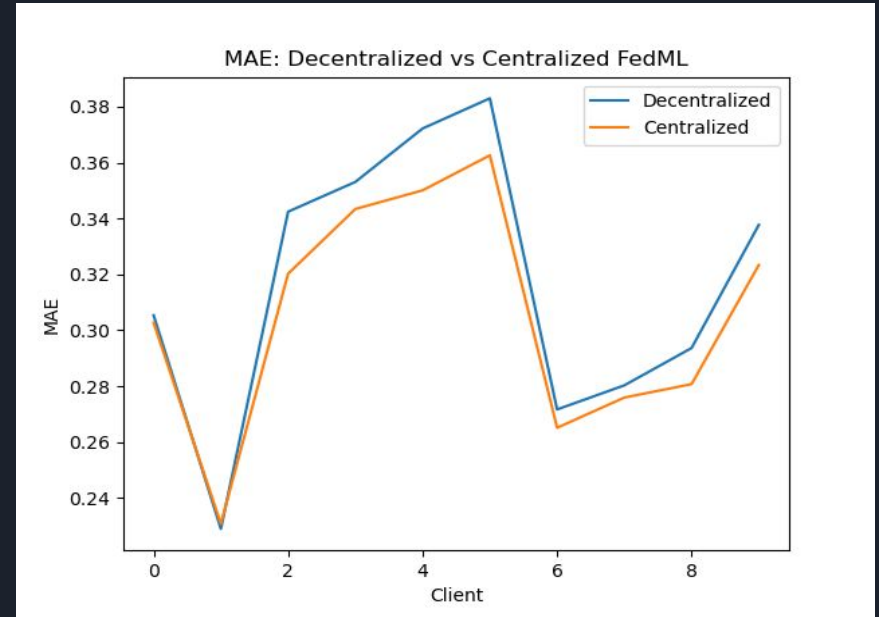
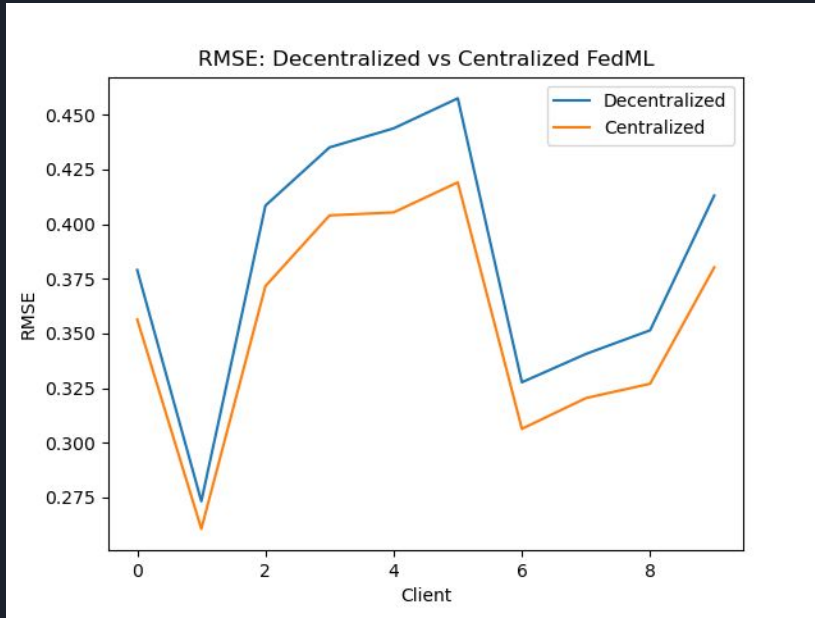
N: No.of clients

S: Sample of clients for each round

K: No.of update steps in the clients

R: No.of communication rounds


Plotting of RMSE and MAE for both the approaches



Results: for traditional SCAFFOLD

```
data processing...
100%|
mae: 0.30277757435602726 rmse: 0.3554986405554239
data processing...
100%|
mae: 0.23113199457081773 rmse: 0.260441166669378
data processing...
100%|
mae: 0.3191603229981416 rmse: 0.36982442009766664
data processing...
100%|
mae: 0.3430669133320218 rmse: 0.40247805435977313
data processing...
100%|
mae: 0.3489718121641295 rmse: 0.40341037108269634
data processing...
100%|
mae: 0.36158317769424986 rmse: 0.4171395265596049
data processing...
100%|
mae: 0.265024872009515 rmse: 0.30559534110985864
data processing...
100%|
mae: 0.275942384695946 rmse: 0.3197040350016284
data processing...
100%|
mae: 0.28023266491013876 rmse: 0.32597396191859235
data processing...
100%|
mae: 0.3228046436644102 rmse: 0.37861169586846216
```

Results: for modified SCAFFOLD



100%	
mae:	0.3030574096118815 rmse: 0.3550650323382735
100%	
mae:	0.23158394502857013 rmse: 0.2605255680962878
100%	
mae:	0.3185679338005474 rmse: 0.3687056620401559
100%	
mae:	0.3430703674350912 rmse: 0.4016357890939206
100%	
mae:	0.34839616461811923 rmse: 0.40220859355469385
100%	
mae:	0.3610384951088169 rmse: 0.41593488325819056
100%	
mae:	0.26512258506468817 rmse: 0.3052334024265274
100%	
mae:	0.27612361896818055 rmse: 0.31936225562878706
100%	
mae:	0.2800410250388529 rmse: 0.3254465229321971
100%	
mae:	0.3225926961059538 rmse: 0.37765278794410695



Further Scope:

1. Integrate an image dataset.
2. Integrate medical images
3. Integrate Blockchain

References:

1. <https://dl.acm.org/doi/full/10.1145/3501813#d1e3670>
2. <https://www.spiedigitallibrary.org/conference-proceedings-of-spie/11006/110061I/Approaches-to-address-the-data-skew-problem-in-federated-learning/10.1117/12.2519621.short?SSO=1>
3. <https://cs.nyu.edu/~mohri/pub/scaffold.pdf>