

Federated Learning

Using FL on NASA Battery Dataset

20260125-20260203

Battery State of Health (SOH) Prediction

Battery ID	Temp Index	Dataset Source	processed Training Cycles	Condition Description
B48 (Client 1)	High Temp	B48_discharge_soh.csv	38	Low Data Density: Smaller number of recorded cycles, representing a battery with limited historical telemetry.
B33 (Client 2)	Low (4°)	B33_discharge_soh.csv	133	High Load: Differing discharge currents and load profiles compared to standard aging tests.
B05 (Client 3)	Room Temp (24°C)	B05_discharge_soh.csv	125	Baseline Aging: Standard cycle life test under controlled environmental conditions.

Executive Summary

- This project aimed to transition a battery SOH prediction system from “Independent Local Learning” (isolated models) to “Federated Learning (FL)”
- The goal was to train a robust “Global Model” across multiple battery datasets (`B48`, `B33`, `B05`) without centralizing the raw data, thereby preserving privacy and efficiency.

Key achievements:

- ❑ Implemented “Federated Averaging (FedAvg)”.
- ❑ Achieved model convergence across 3 heterogeneous clients.
- ❑ Implemented “Personalization (Fine-Tuning)” to resolve performance issues for clients with smaller datasets

Methodology & Architecture

Federated Averaging (FedAvg)

Used a centralized server-client architecture:

1. Server: Maintains the Global Model.
2. Clients:
 - Download global weights.
 - Train locally on their private battery data (5 epochs).
 - Upload updated weights to the server.
3. Aggregation: Server averages client weights (weighted by dataset size) to update the Global Model.
4. **Model Architecture:** A **Gated Recurrent Unit (GRU)** network was used for time-series prediction, consisting of **two GRU layers (60 units each)** with dropout (0.2) and a final dense layer to output the SOH value.
5. **Training Parameters:** The global training consisted of 20 communication rounds.

Each local client trained for 5 epochs per round with a learning rate of 0.001 using the Adam optimizer.

Addressing Data Imbalance (Personalization)

Identified significant **data imbalance** (Client 1 ~23k rows vs. others ~50k).
This biased the Global Model towards the larger clients.

Solution: Added a Personalization Step where, after global training,
Client 1 fine-tunes the global model on its local data for 10 additional epochs.

```
# Fine-tuning for Client 1 (B48)

ft_model = create_model(input_shape)

ft_model.set_weights(server.global_weights)

# Train locally for 10 additional epochs

history_ft = ft_model.fit(client1.X_train, client1.y_train, epochs=10)
```

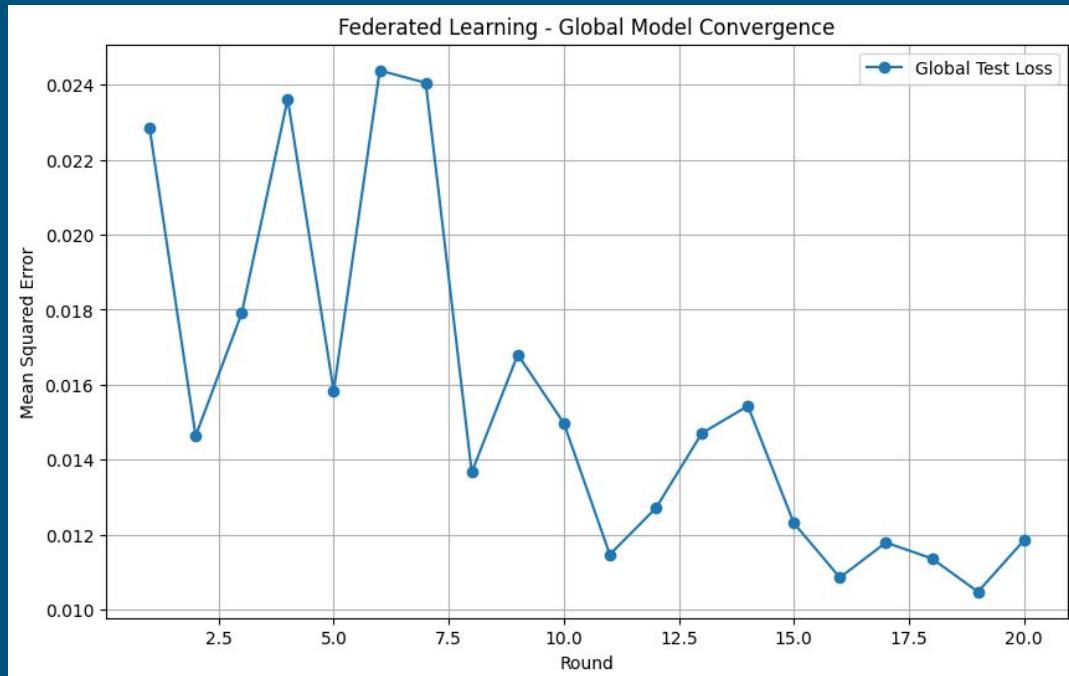
Results & Visualizations

Global Model Convergence

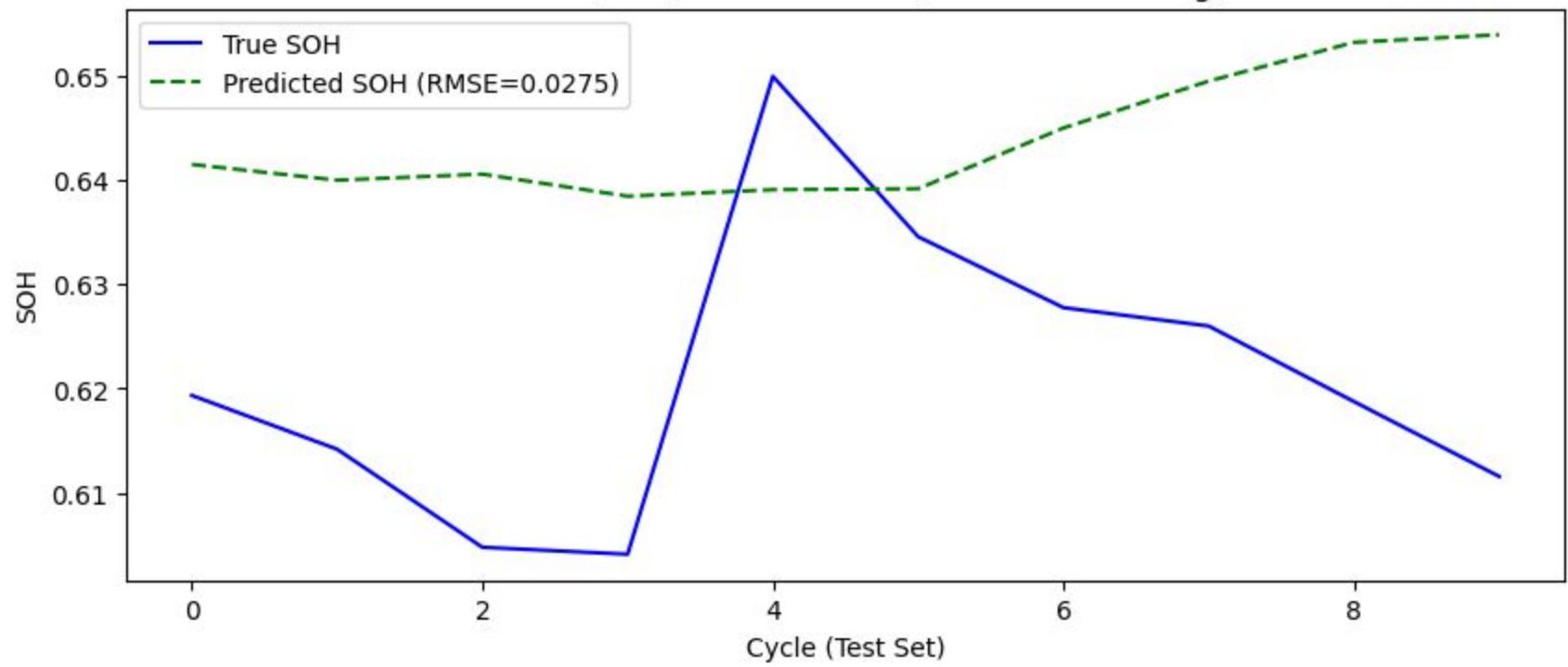
The Global Test Loss (MSE) demonstrates a clear downward trend over 20 rounds, indicating that the model successfully learned generalizable features from all batteries.

Visualization:

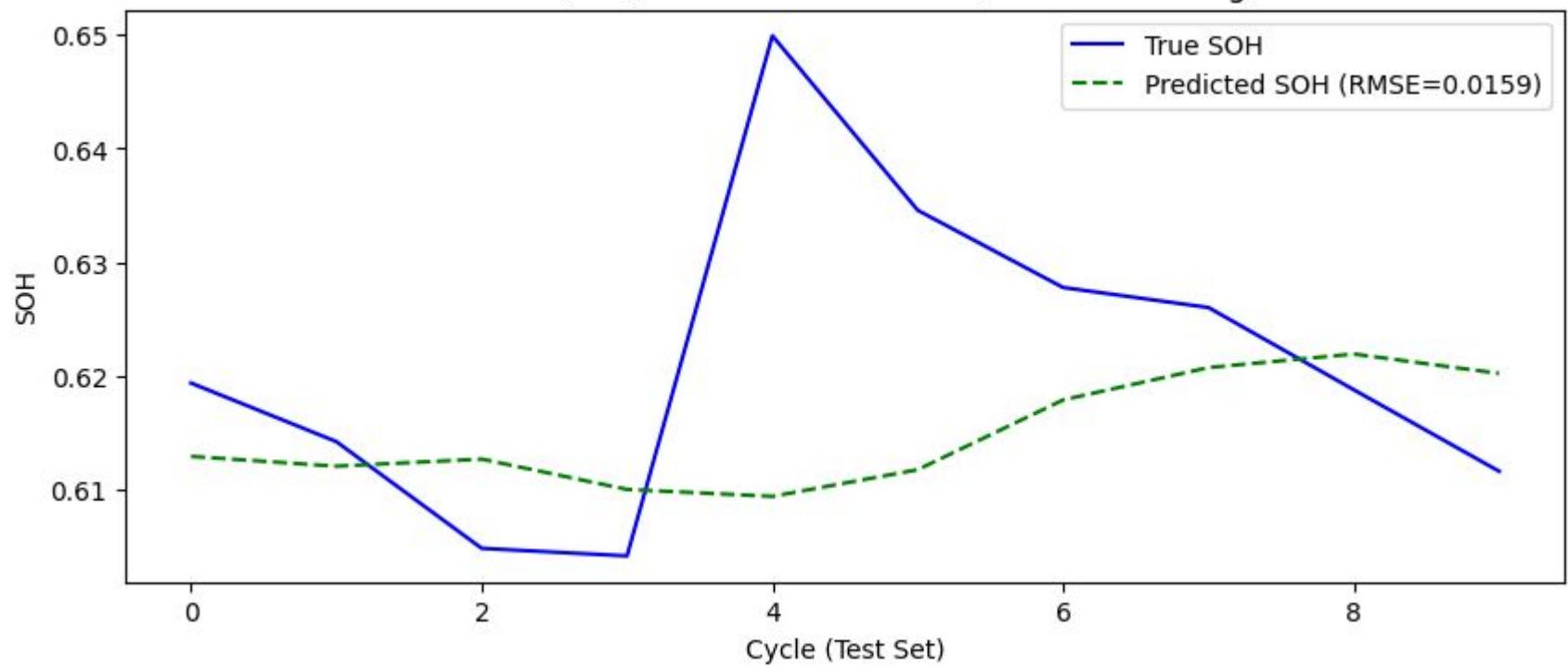
- **X-Axis:** Communication Rounds (1-20)
- **Y-Axis:** Global Test Loss (MSE)
- **Trend:** Exponential decay, stabilizing around Round 15.



Client 1 (B48) - Global Model (Before Fine-Tuning)



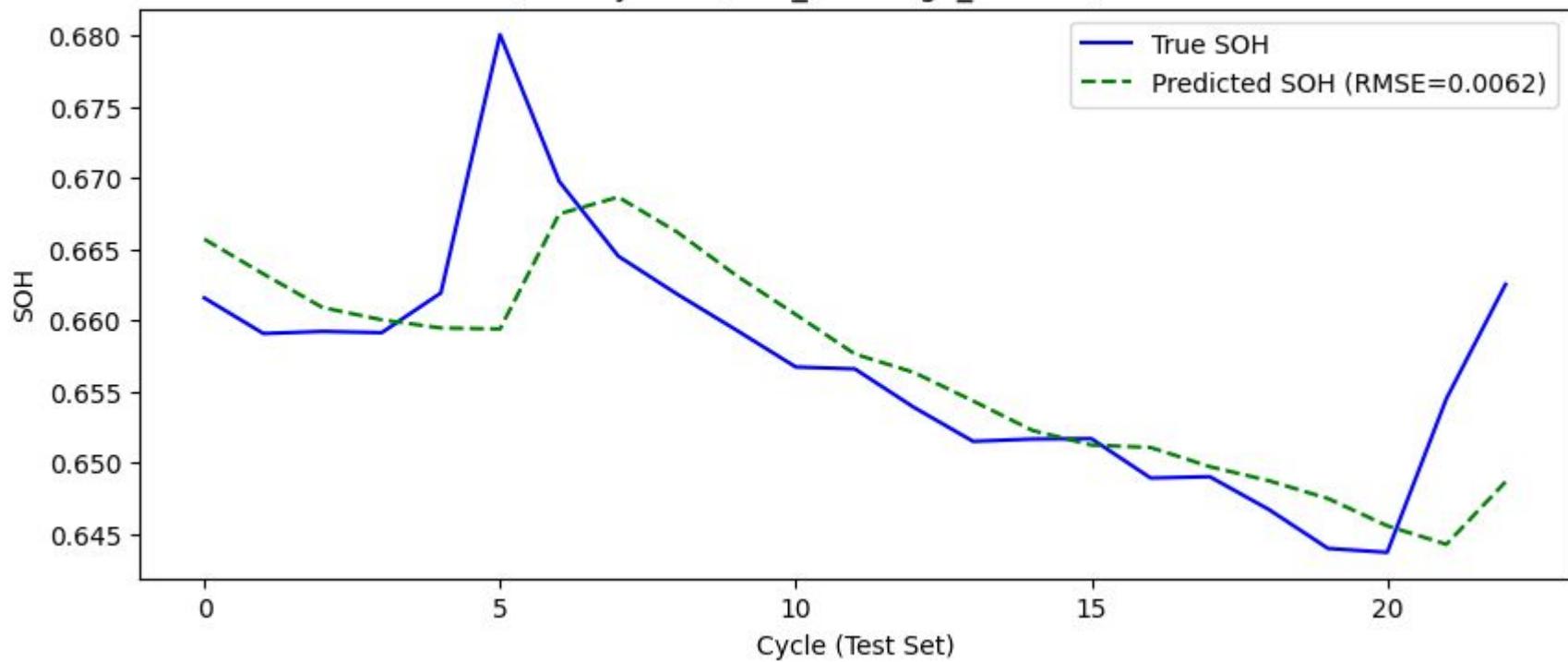
Client 1 (B48) - Personalized Model (After Fine-Tuning)



Client 2 (Battery DATA/B33_discharge_soh.csv) - Global Model



Client 3 (Battery DATA/B05_discharge_soh.csv) - Global Model



Client / Battery ID	Model Version	Mean Squared Error (MSE)	Root Mean Squared Error (RMSE)
Client 1 (B48)	Global (Before Fine-Tuning)	0.000759	0.027542
Client 1 (B48)	Personalized (After Fine-Tuning)	0.000252	0.015883
Client 2 (B33)	Global Model	0.000191	0.013811
Client 3 (B05)	Global Model	0.000038	0.006176