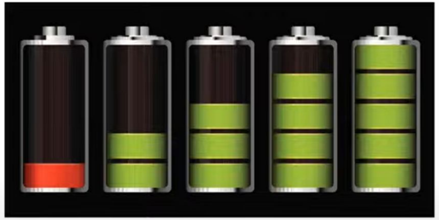
State of Charge Estimation for Li-ion Battery

Koneru Saketh
Shaik Asif Ahmad
Vishwas Chepuri

Problem Statement

- Estimating **State of Charge** using different model-based methods.
- As the SoC is an important parameter, which reflects the battery performance, so accurate estimation of SoC is an important feature in Battery Management Systems
- It is defined as the ratio of the remaining charge in the battery and the maximum charge that can be delivered by the battery.

$$SoC(t) = Q(t) / Q_n$$



$$0 < SOC < 1$$

Coulomb Counting

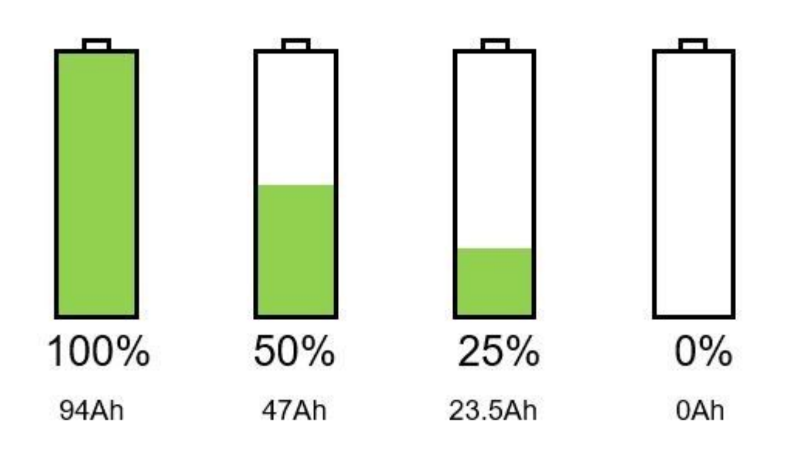
SOC estimation through Coulomb Counting

Coulomb Counting

SOC estimation through Coulomb Counting

Coulomb Counting

- SoC Estimation by Coulomb Counting is based on the measurement of the current and integration of that current over time.
- Coulomb Counting gives a relative change in SoC.
- Due to the integration, any measurement error in current accumulates over time.



$$SoC(t) = SoC(t-1) + \frac{I(t)}{Q_n}\Delta t$$

The methods that we worked on in this project are:

- Using Extended Kalman Filter
- Using Feedforward Neural Network
- Using Decision Tree Ensembles

DATA SOURCE USED

Turnigy Graphene 5000mAh 65C Li-ion Battery Data

Published: 14 August 2020 | Version 1 | DOI: 10.17632/4fx8cjprxm.1
Contributors: Phillip Kollmeyer, Michael Skells

Files

- 0 degC
- -10 degC
- 10 degC
- -20 degC
- 25 degC
- 40 degC
- Cell Data and Results for Turnigy Graphene 5000mAh 65c cell.pdf
- Readme Turnigy Graphene 5000mAh 65C cell.txt
- Turnigy Graphene 5000mAh 65C cell plots.xlsx

Graphene 5Ah Lithium Ion Battery

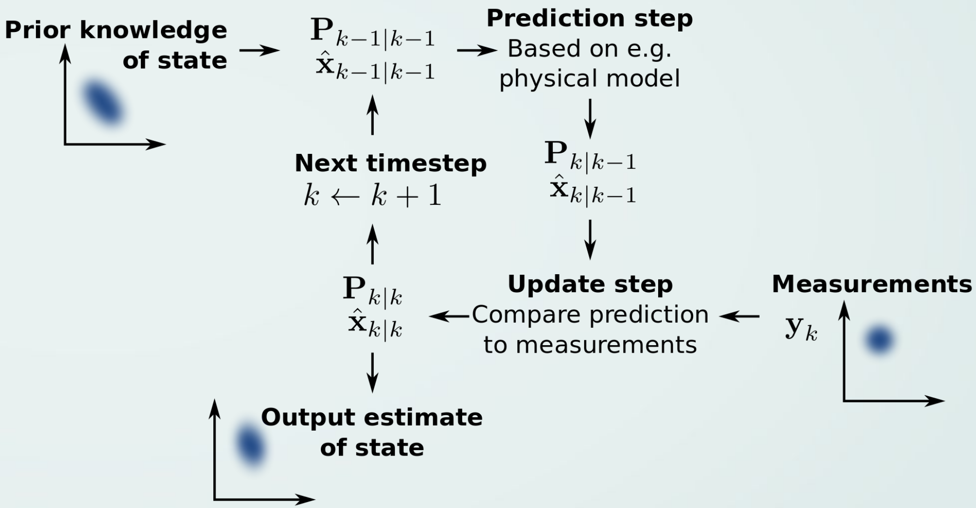
- Data pertaining to each charge or discharge consisted of Measured Temperature, Measured SOC values, Measured Terminal Voltage, Measured Current and Battery Parameters in the process of discharge.

Extended Kalman Filter

SOC estimation through EKF

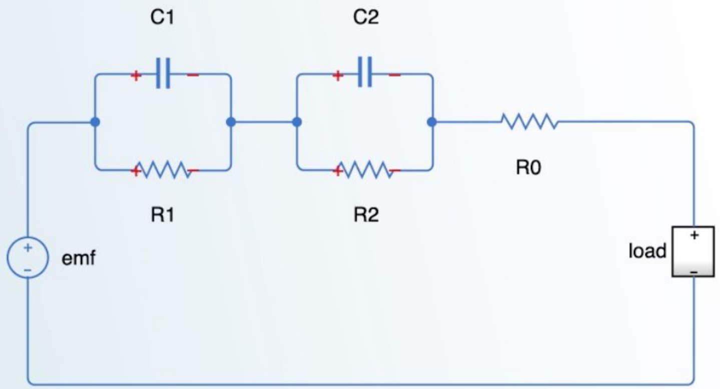
Kalman Filter

- Kalman Filtering produces estimates of unknown variables using a series of measurements observed over time, including statistical noises and other inaccuracies.
- These estimates tend to be more accurate than those based on a single measurement alone.



Extended Kalman Filter (EKF)

- An Extended Kalman Filter (EKF) is used when a system is non-linear but can be linearized about an estimate of the current mean and covariance.



Equivalent Circuit Model of Battery

- Equations for prediction step(a priori values of state variable and error covariance):

$$\hat{x}_{k+1|k} = A\hat{x}_{k|k} + Bu_k$$

$$\hat{P}_{k+1|k} = A\hat{P}_{k|k}A^T + Q_k$$

- Equations for computing Kalman gain and updating the previous measurements:

$$K_{k+1} = P_{k+1|k}C^T(CP_{k+1|k}C^T + R_{k+1})^{-}1$$

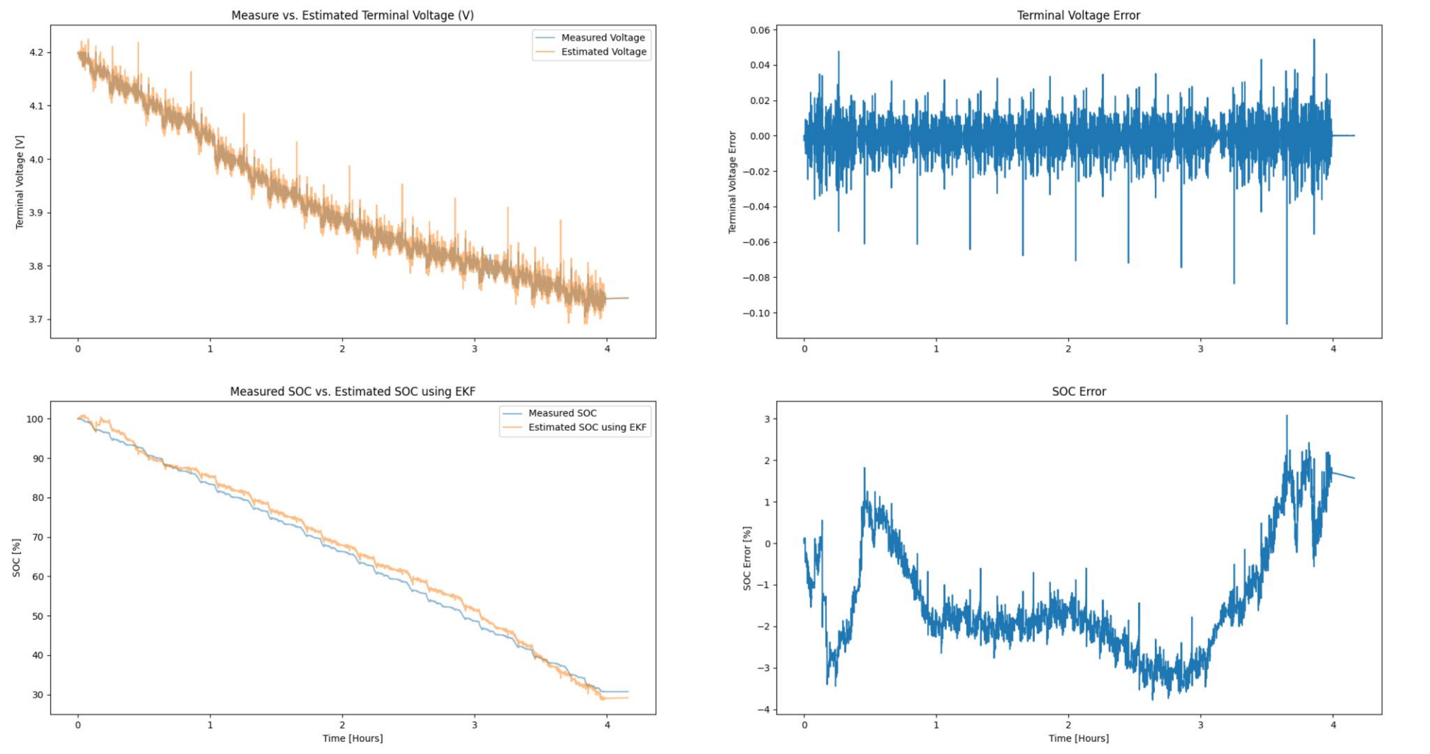
$$\hat{x}_{k+1|k+1} = \hat{x}_{k+1|k} + K_{k+1}(z_{k+1} - C\hat{x}_{k+1|k})$$

$$\hat{P}_{k+1|k+1} = (1 - K_{k+1}C)P_{k+1|k}$$

- Equation for Terminal Voltage:

$$V_t = V_{OC} - V_1 - V_2 - iR_0$$

EKF Predictions

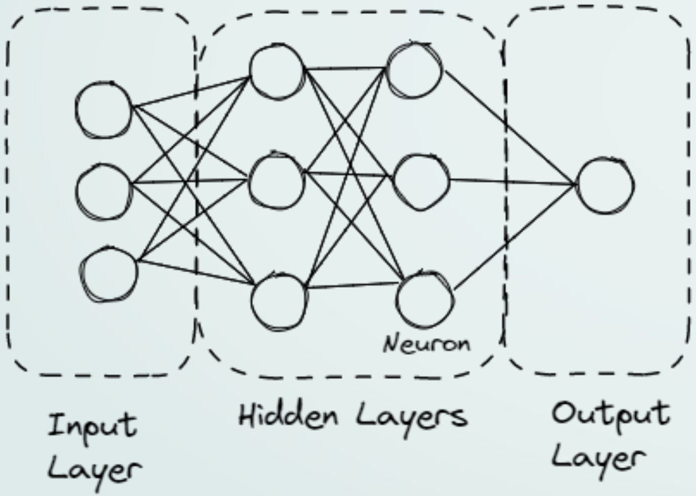


Feedforward Neural Networks

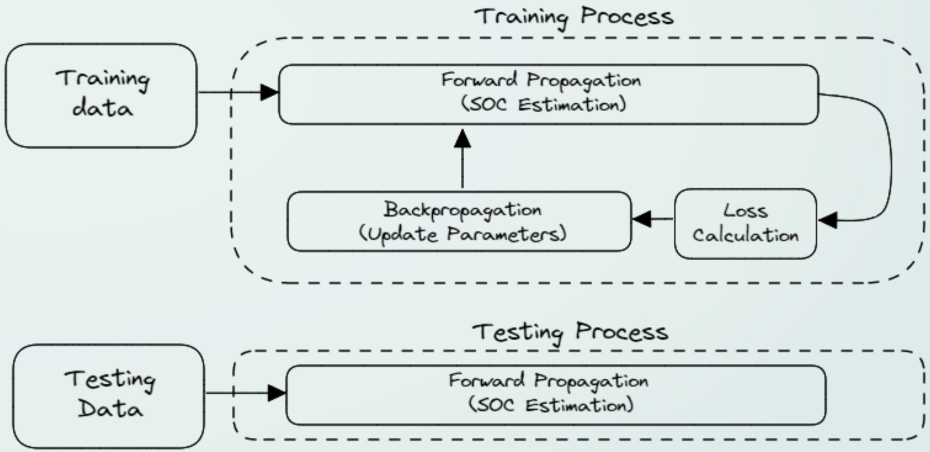
SOC estimation using FNNs

Overview of Training & Testing process

Simplified concept FNN structure for SOC Estimation



Neural Network training and testing/validation process overview



Data preparation

- We used following six measurements as inputs to the network.

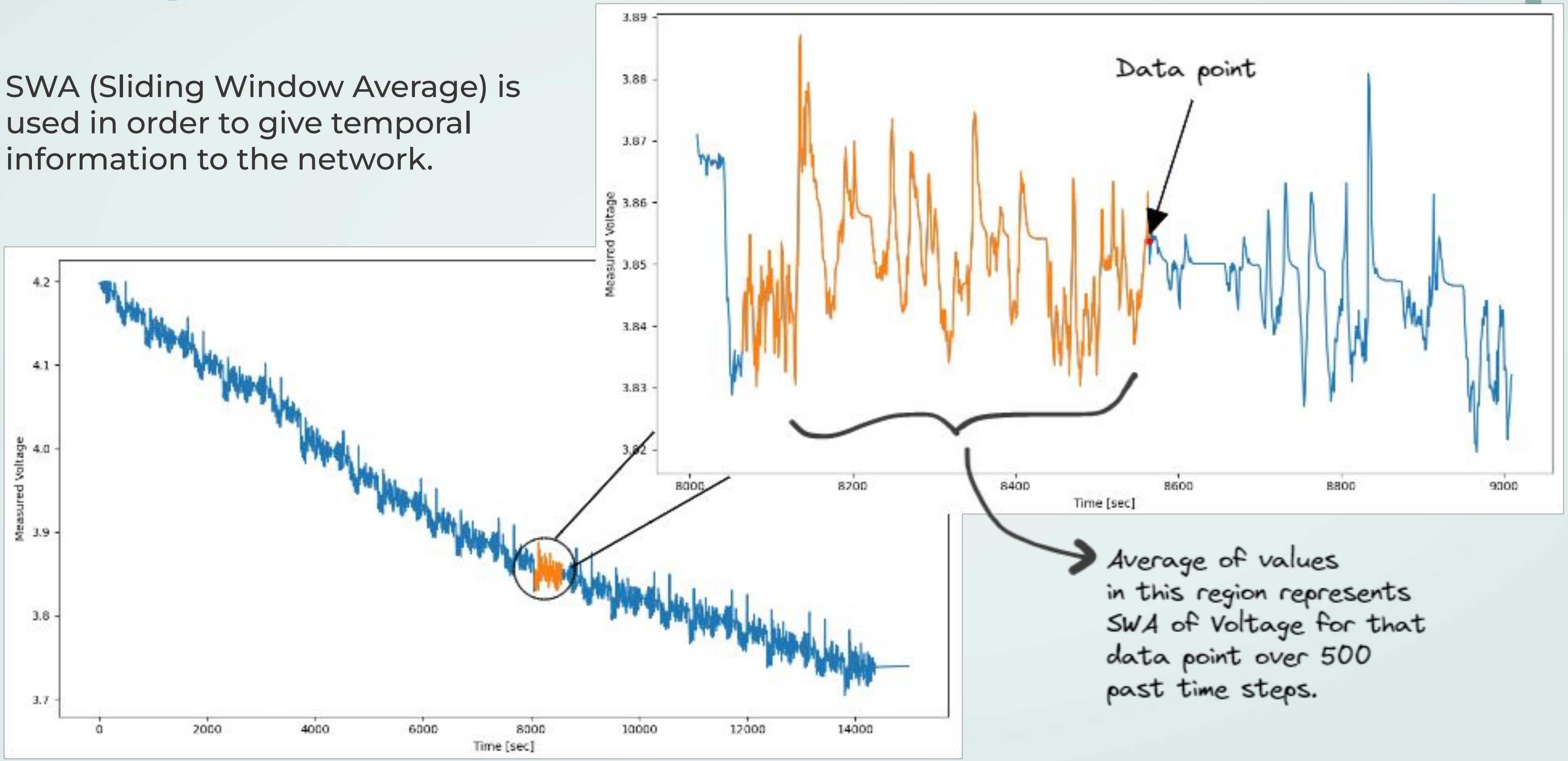
  1) Measured Voltage
  2) Measured Current
  3) Measured Temperature
  4) SWA of Voltage
  5) SWA of Current
  6) SWA of Temperature

  Here SWA means sliding window average which was taken with a window size of 500sec.

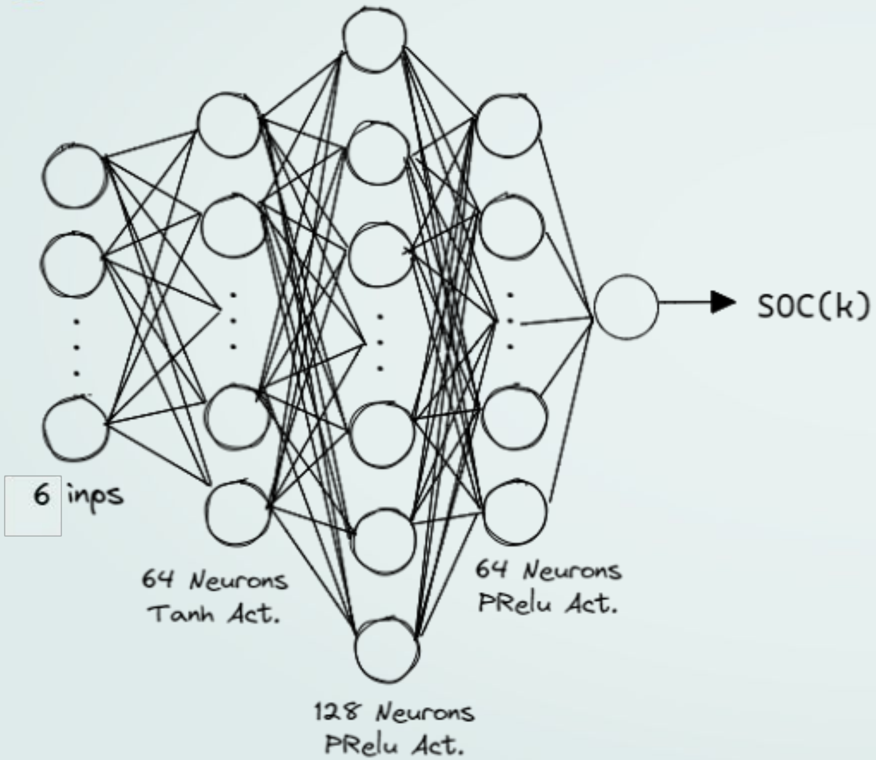
- We normalize the SOC values before training so that values are between 0 and 1.

Data preparation

SWA (Sliding Window Average) is used in order to give temporal information to the network.

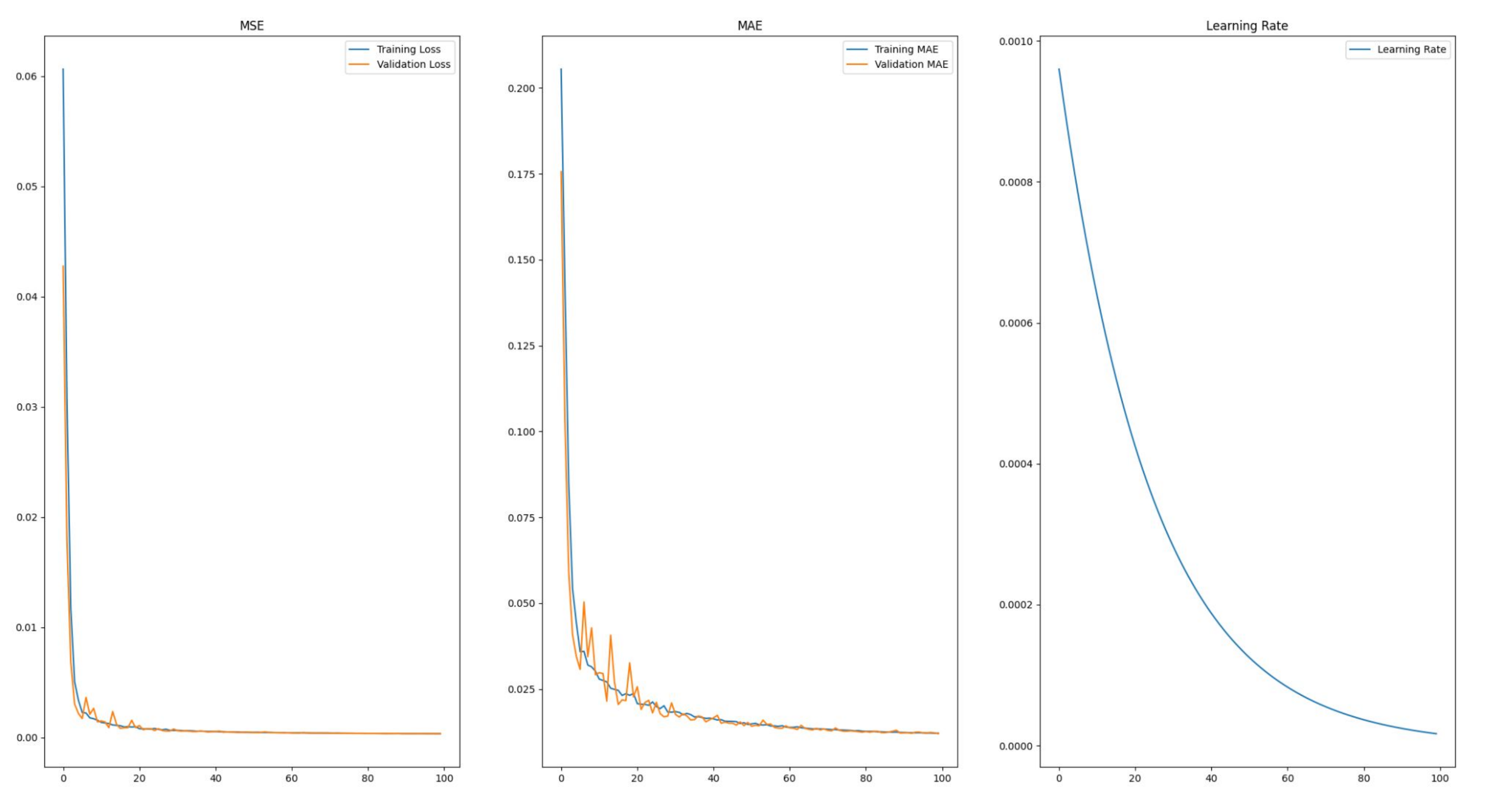


Network and training parameters

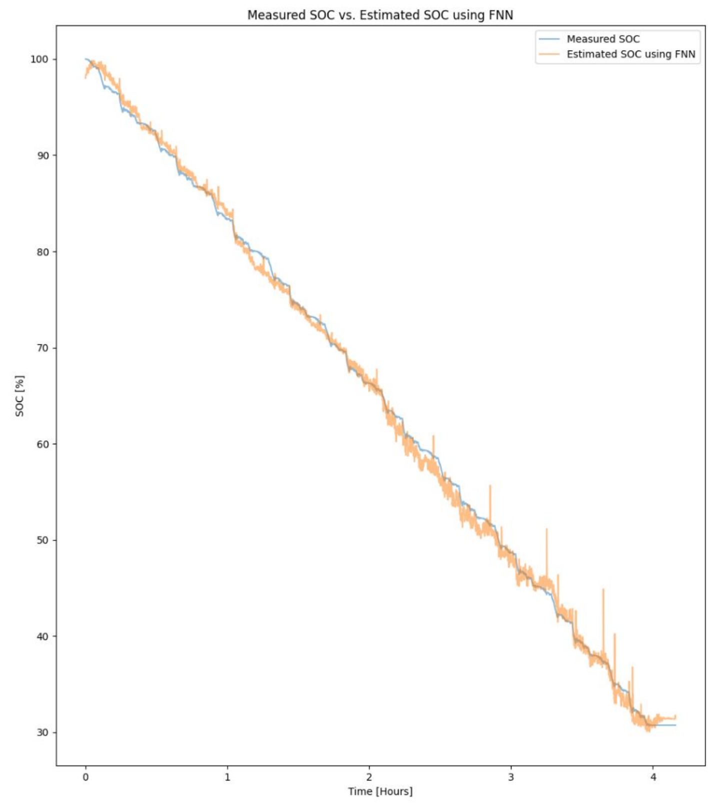


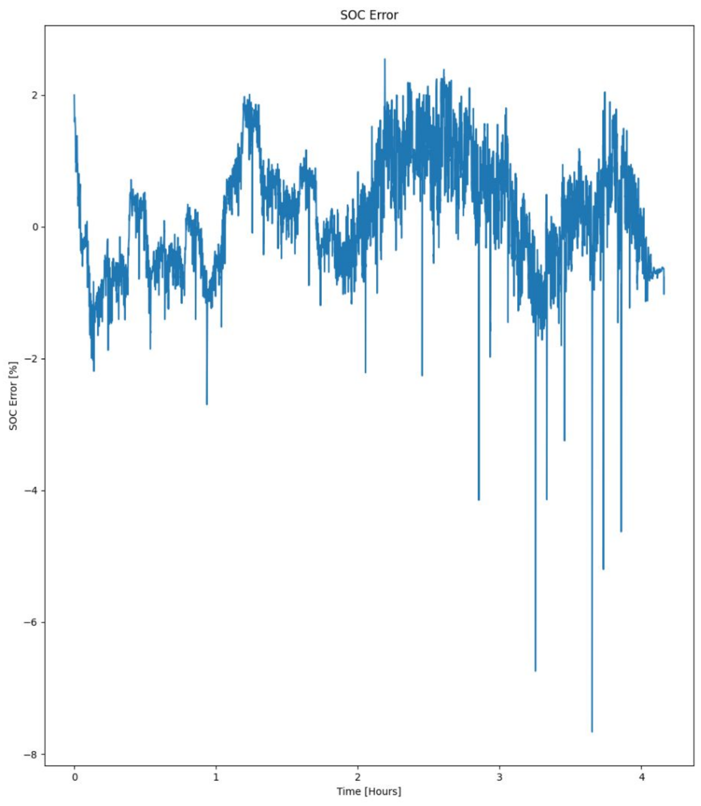
Number of hidden layers	3
Number of Neurons in each layer	[64, 128, 64]
Number of epochs	100
Initial learning rate	0.001
Learning rate drop factor	0.96
Optimizer	Adam
Batch size	1024
Loss	MSE

Training results



Predictions



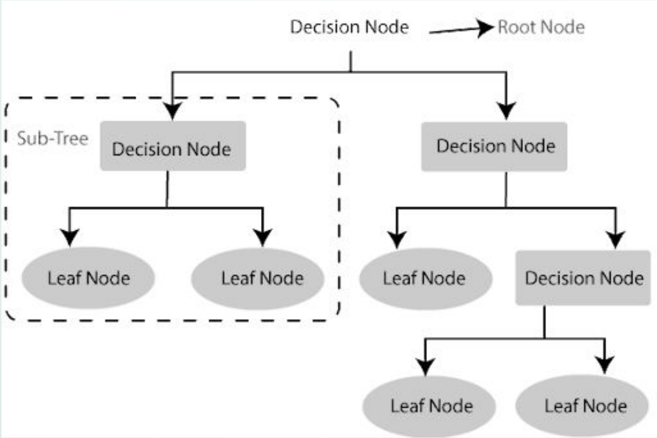


Decision Tree Ensembles

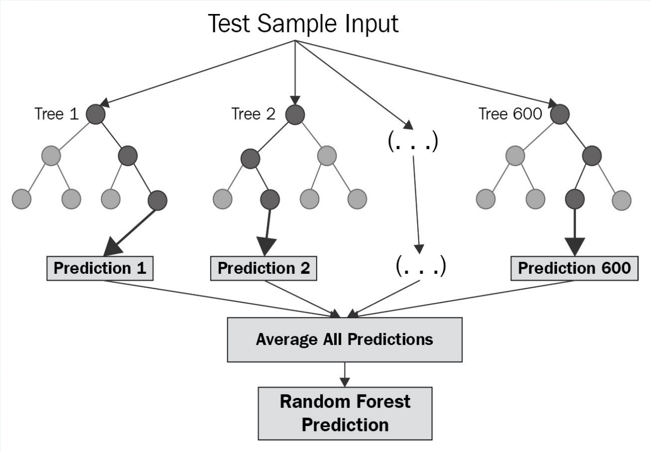
SOC estimation through XGBoost

Decision Tree vs Tree Ensemble

- Decision Trees:



- Tree Ensembles:



Data preparation

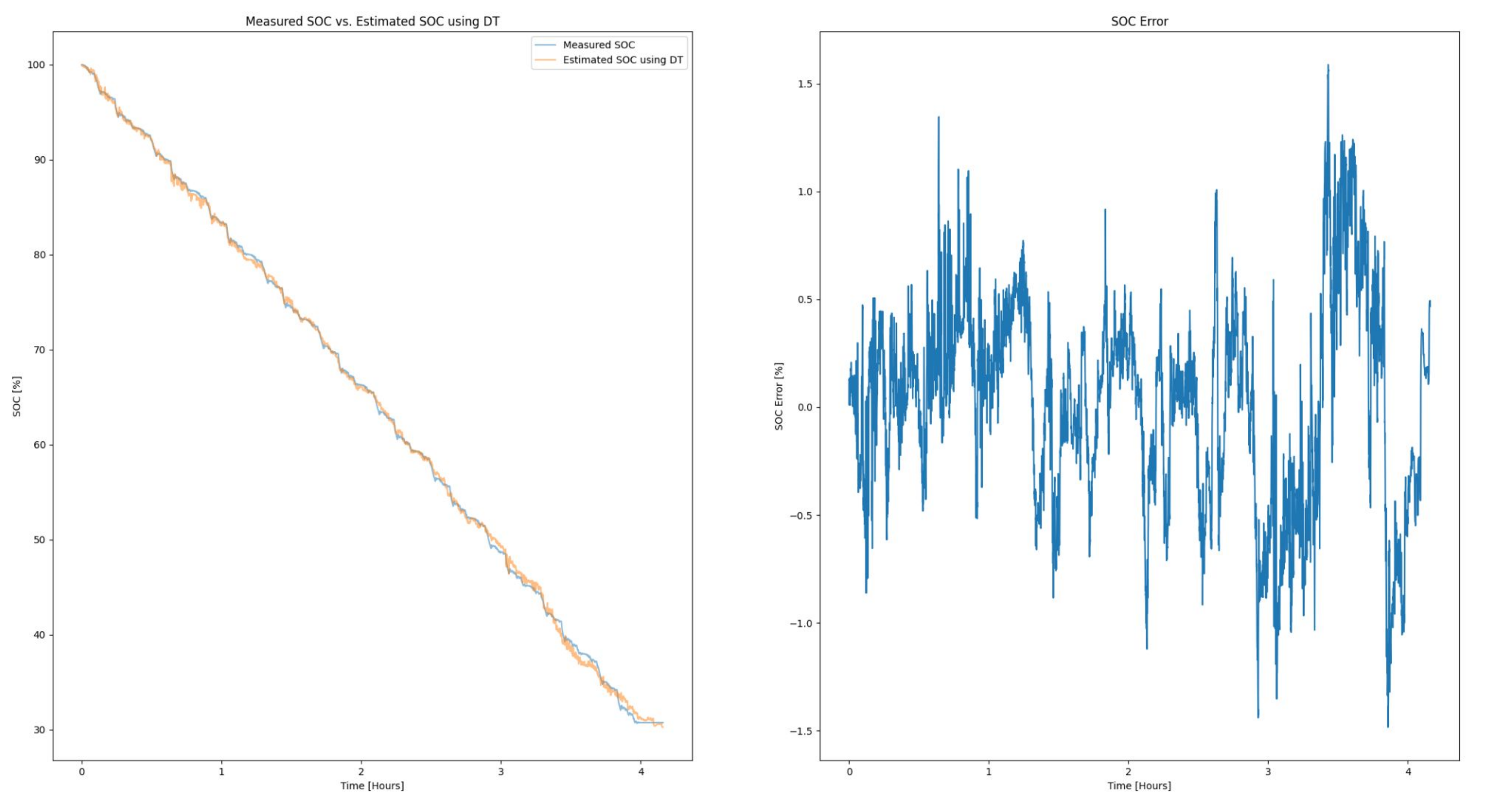
- We used following six measurements as inputs to the network.

  1) Measured Voltage
  2) Measured Current
  3) Measured Temperature
  4) SWA of Voltage
  5) SWA of Current
  6) SWA of Temperature

  Here SWA means sliding window average which was taken with a window size of 500sec.

- We normalize the SOC values before training so that values are between 0 and 1.

Predictions



Observations and Results

- Errors of the three methods:
    - Mean Percentage Error of EKF = 3.03%
    - Mean Percentage Error of FNN = 1.20%
    - Mean Percentage Error of DTE = 0.75%

- Decision Tree Ensembles works better than the other two methods in predicting the value of SOC.

THANK YOU

References

- How to Estimate Battery State of Charge using Deep Learning
- Overview of batteries State of Charge estimation methods
- https://data.mendeley.com/datasets/4fx8cjprxm/
- State of Charge Estimation Using Extended Kalman Filters for Battery Management System
- State of Charge Estimation based on Kalman Filter