Dynamic Modeling of Proton Exchange Membrane Electrolyzers using regression methods

Author: Brian King

Instructor: Dr. Hwan-Sik Yoon

Contents

Background

Electrolysis is a leading contender for off-peak energy storage. Energy storage is an essential component to increasing energy production from renewable sources because the power output from these methods fluctuates with constant changes in the environment. Excess energy must be stored, and when there is a deficiency, energy can be drawn from the storage. A dynamic simulation over time is necessary to analyze how fluctuations of current from these sources such as solar panels effect the voltage response and hydrogen production of a PEM Electrolyzer.

Multiple approaches have been utilized to dynamically model a PEM Electrolyzer. These methods include an equivalent circuit approach and neural networks (such as Multilayer Perceptron Network). Long-Short Term Memory RNN's have been used with PEM Fuel Cells to predict performance on vehicle driving, but not to estimate hydrogen output from a PEM Electrolyzer. K-NN has been used with PEM electrolyzers to predict the most frequent faults in fuel cell system, but not in the estimation of hydrogen production and voltage response.

Data Source:

The source data in Figure 1 comes from the Korean Institute of Energy. This data includes the voltage response to current density in a HIAT MEAs PEM Electrolyzer stack. The porous transport layers (PTL's) used were Mott titanium layers. The test was run at 50 degrees Celsius for a period of 12 hours. 90% of the data was used as the test data, and the remaining 10% was used as the validation data. Excess noise was filtered out by removing data wherein the voltage fell outside of 2.5 standard deviations from the mean.

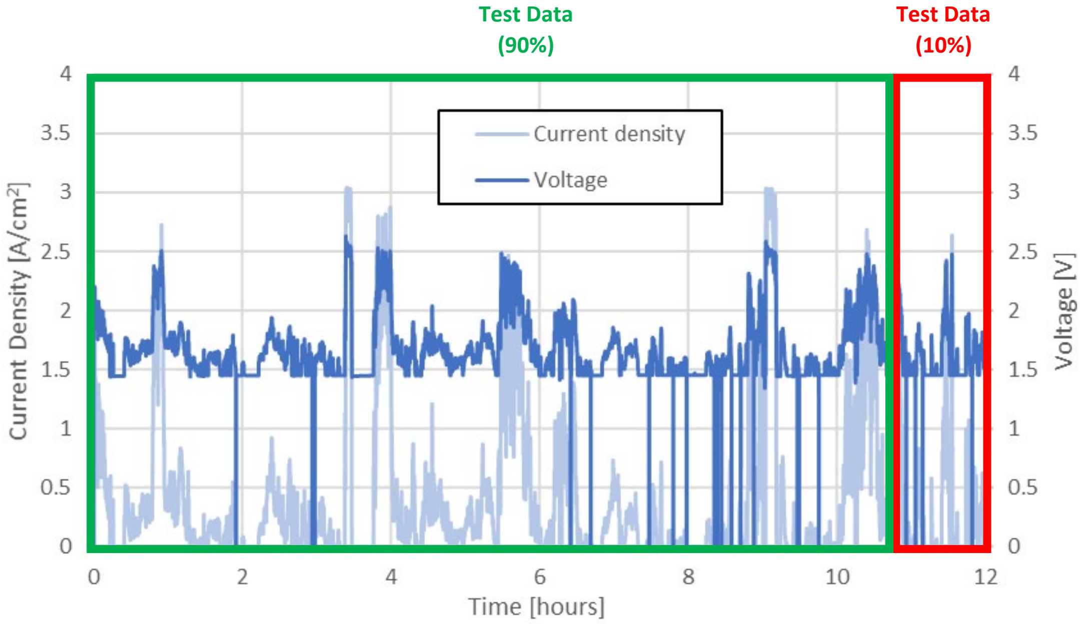

Figure 1. Time vs. Current Density and Voltage Data Split

Objective

The objective of the k-NN and Recurrent Neural Network (RNN) is to accurately predict the voltage response to changes in current density in a PEM Electrolyzer. Two dataset variations will be utilized. The first contains a single feature: the current density input to the system. The second variation includes three features: current density, the time that has passed since the last jump in current, and the size of the current jump (Figure 2). The selected change in current density to be considered a "jump" is 0.1 A/cm^2. Analyzing the data with three features is an attempt to predict the differences in voltage output when given the same input (current density). The voltage output of PEM Electrolyzers contains overshoots and capacitive responses which are dependent upon previous states. For instance, the data contains eight instances of 720 A/cm^2 as a current density input. The voltage output when given these eight current densities has a range of 21 millivolts. Adding these extra features allows past instances to be considered as factors effecting this difference in voltage output. These two dataset variations (1-feature vs 3-features) will be compared to determine how they affect the performance of the k-NN and RNN methods.

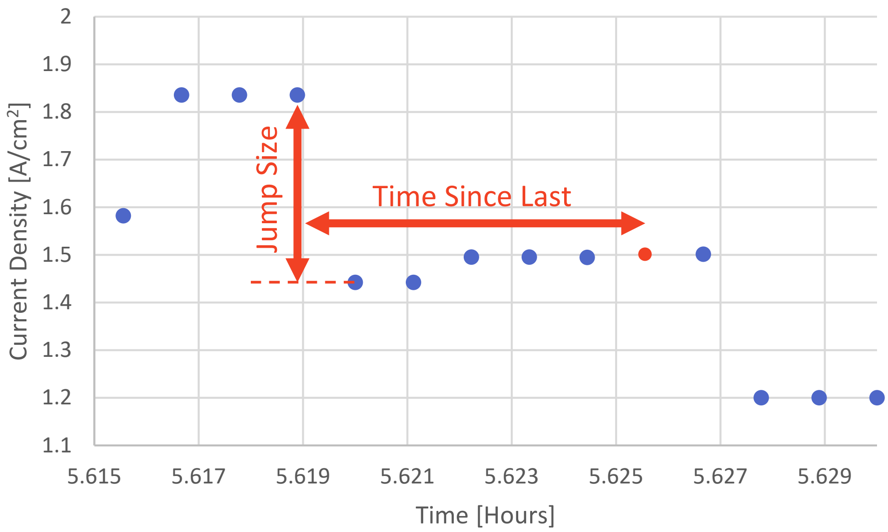

Figure 2. Features Explanation

Neural Network Results

A Recurrent Neural Network (Long Short-Term Memory) approach is ideal for this regression model, as it takes the previous outputs into consideration and stores them in memory. Therefore, the nth instance of a sample may be influenced by instances that occurred previously.

Using the time that has passed since the last jump in current, and the size of the current jump (3-Feature method) allowed for more accurate predictions from the neural network. Given these inputs, the neural network learns exceptionally faster than with current density as the only input (Figure 3).

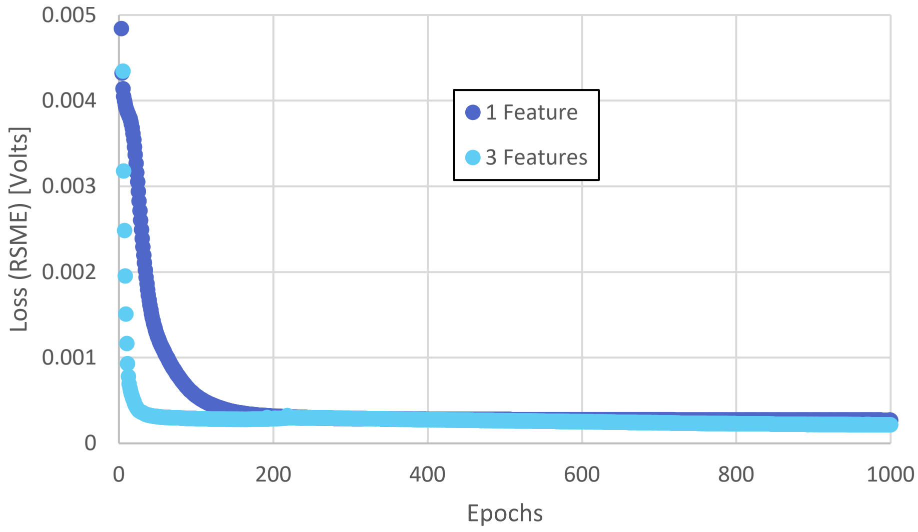

Figure 3. Losses vs Epochs for 1 and 3 Feature Datasets

Expanding the data to observe only losses below 0.4 millivolts, the 1-Feature dataset (Figure 4) begins to hit its asymptote before the 3-Feature dataset (Figure 5). By 1800 epochs, the losses for the 1-Feature validation set begin increasing, while the 3-Feature set continues to decrease in error. The "elbow" where the losses for the 3-Feature dataset begin to increase is around 2775 epochs.

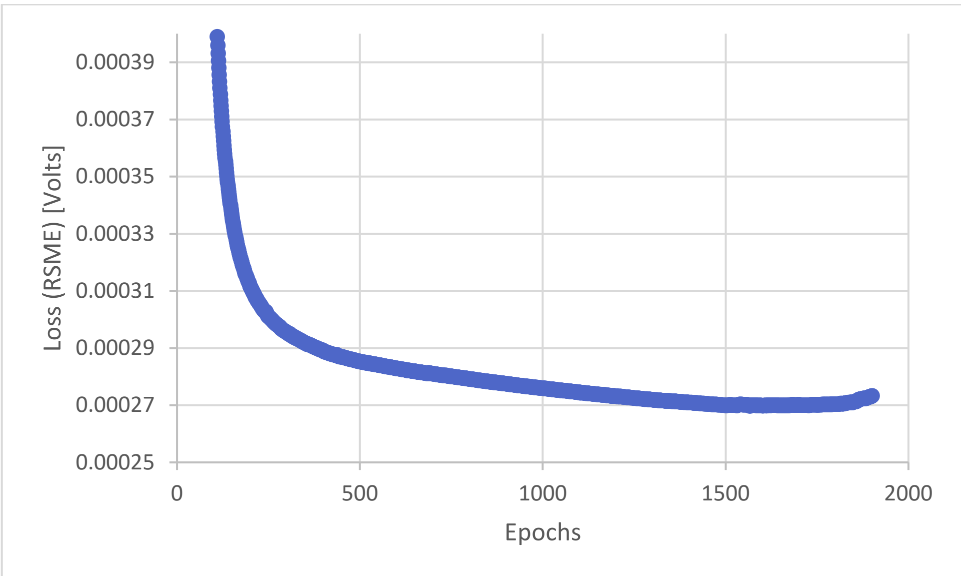

Figure 4. Losses vs Epochs for 1-Feature Dataset between 0.25 and 0.39 mV

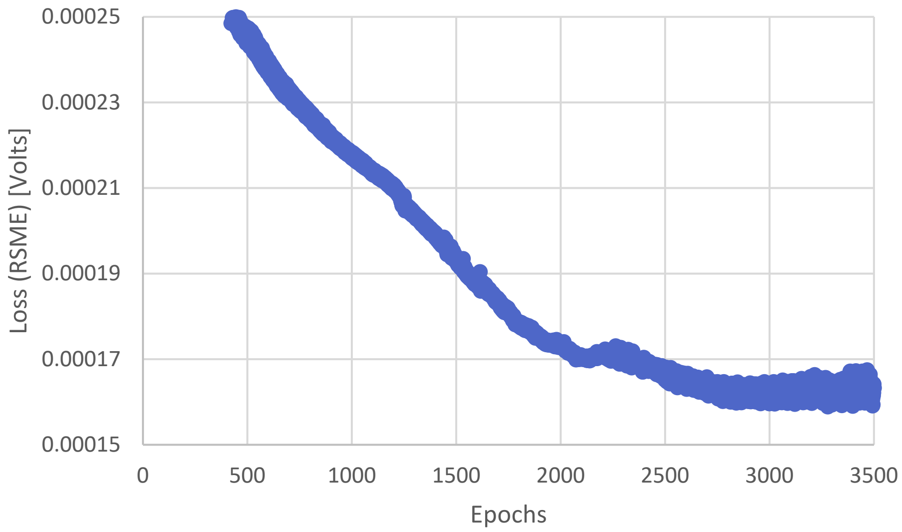

Figure 5. Losses vs Epochs for 3-Feature Dataset between 0.15 and 0.25 mV

In Figure 6, the individual data points are shown for the experimental data compared to the predicted values from the neural network. In this instance, the 3-Feature dataset is utilized at 1900 epochs. The largest errors occur with high voltages in response to high currents that lasted for short periods of time. The neural network makes the best predictions when the current density is held steady or the changes in voltage response are minute.

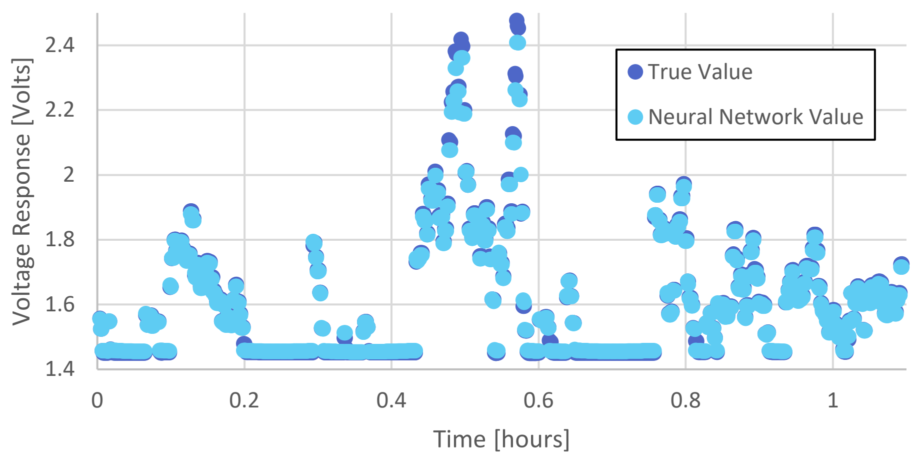

Figure 6. True vs. Predicted Voltages

k-Nearest Neighbors Results:

K-Nearest Neighbor is a nonparametric approach where the values of the "K" most similar inputs are used to determine the output. The Euclidean distance is used to determine the "K" nearest values (Equation 1).

$$d = \sqrt{\sum_{i=1}^{k}(x_i - y_i)^2} \quad \text{Eq. 1}$$

The downside to this method is the large computational time. Due to the nonparametric characteristics of the method, each new data must be compared to all the values in the test data set. Increasing the size of the test data increases the test accuracy, but exponentially increases the run-time of the algorithm.

When the time that has passed since the last jump in current, and the size of the current jump is used as data inputs (3-Feature Method), the error is higher for most K values. The kNN method works best with only the current density as the input feature (Figure 7).

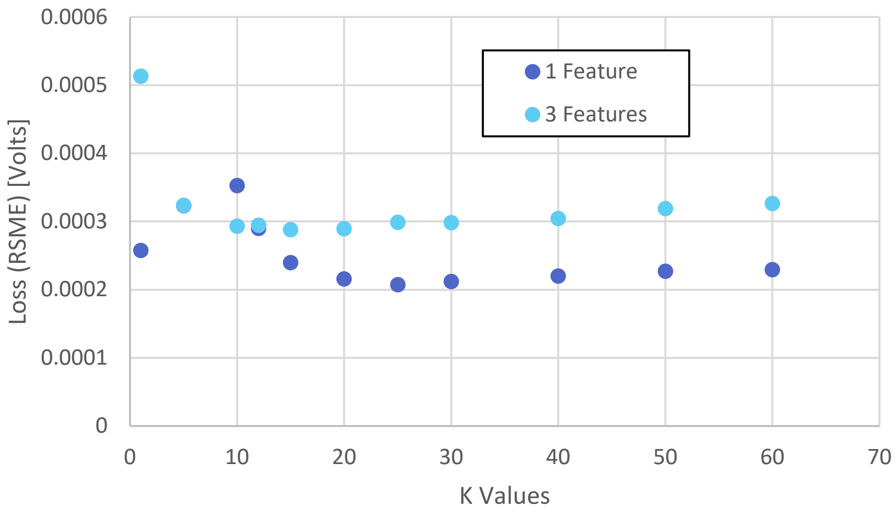

Figure 7. Losses for 1 & 3-Feature Datasets given various K-Values

The loss (root mean square deviation) reaches an optimal value with a K value of approximately 25 nearest-neighbors with the 1-Feature dataset (Figure 8).

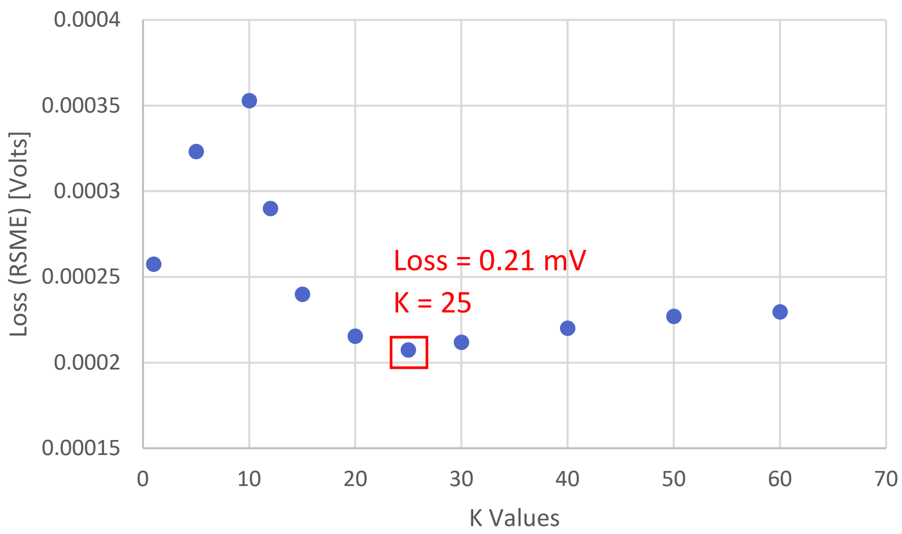

Figure 8. RSME for various K values

k-NN vs. Neural Network

The data from Figure 1 was used to compare the k-NN method to the Neural Network. For the kNN method, an optimal K values of 25 was used. For the neural network, 1900 epochs were used for the comparison. All four methods performed similarly, with a range in error of 0.14 millivolts between the top-performing and worst-performing method. The 3-feature dataset obtained the best results with the neural network (RNN), with a 41% decrease in losses when using the 3-feature datasets over the 1-feature method. The opposite effect was seen with the k-NN method. There was a 31% decrease in error when the 1-feature dataset was employed instead of the 3-feature dataset.

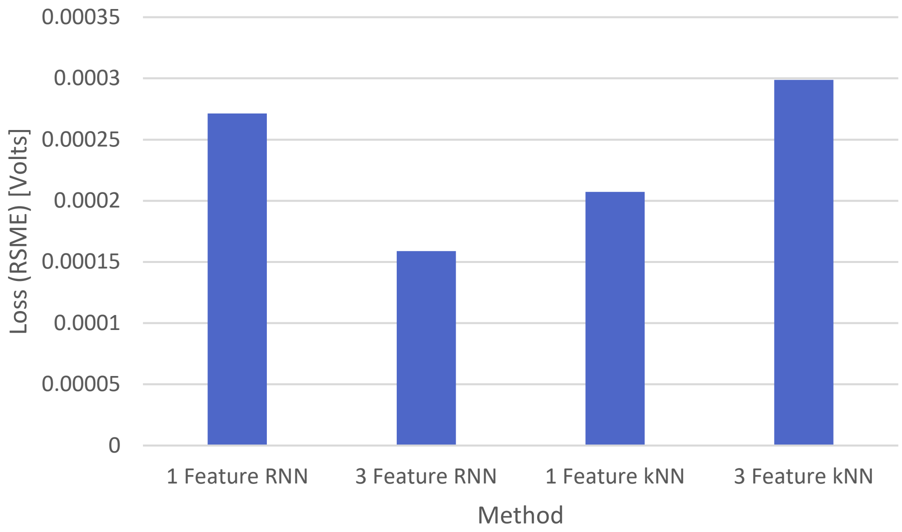

Figure 9. Comparison of Losses for each prediction method

Conclusion

The recurrent neural network (long short-term memory) method using the 3-feature dataset outperformed all other methods. This implies that previous jumps in current effect the voltage response. However, adding previous states marginally decreased the accuracy with the k-NN method. The lack of effect from previous states can be attributed to the type of data used in this report. The current density was adjusted at a rapid rate to obtain the data above. When slower changes in current are used, the voltage response is affected more drastically by the time since the previous jump in current density (Figure 10). Once more slow-changing data is obtained, the model can be run on this data to determine if the accuracy increases when using the kNN method with previous state changes as inputs (3-feature method) compared to the kNN method with 1 feature.

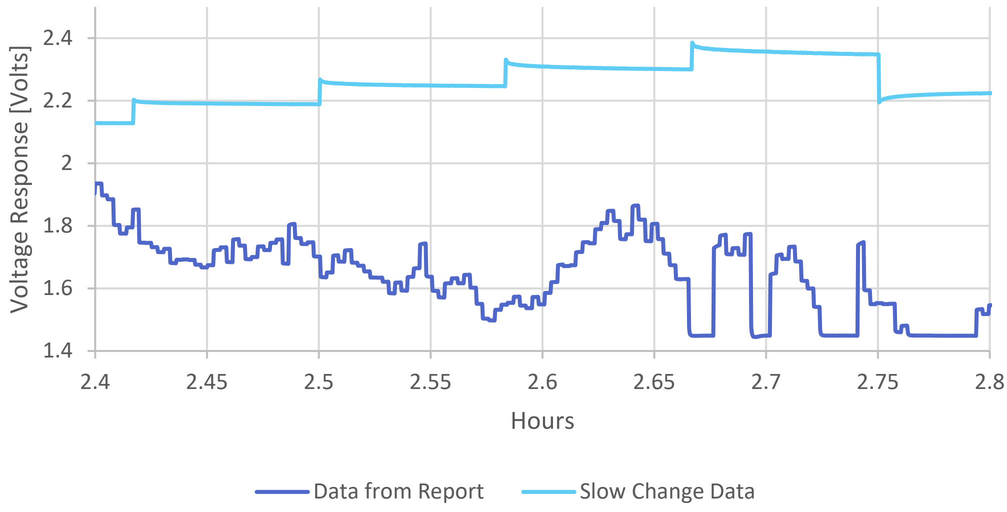

Figure 10. Time snippet with multiple data types

The recurrent neural network is more accurate while taking less computational time since, unlike the kNN method, new data does not need to access all the training data to make a prediction. The neural network was also the most accurate and therefore is the optimal method out of the two from this report to predict the voltage response in a PEM Electrolyzer when given the current density.