# Forecasting RIT's Energy Usage Using Neural Networks

TREVOR MARTIN    ROCHESTER INSTITUTE OF TECHNOLOGY    INDUSTRIAL AND SYSTEMS ENGINEERING    ROCHESTER, NY    TMM4038@RIT.EDU

***Abstract*—RIT consumes a large amount of energy and wants to reduce the peaks of this consumption to reduce cost. Using neural networks, this consumption was modeled and forecasted to attempt to predict these peaks. Different neural network models were evaluated, and the best one was recommended for use by RIT.**

## I. INTRODUCTION

Rochester Institute of Technology is a university with over 13,000 students attending at the Henrietta Campus. A university of this size uses large amounts of electricity, and is charged for this electricity usage in two different ways. First, there is the typical consumption charge, measured in kilowatt-hours (kWh). This cost is estimated at $0.12 / hWh, and makes up about 30-50% of RIT's electricity bill. The other 50-70% of the electricity bill is due to the demand charge. The demand charge scales with the maximum consumption of electricity at any given point throughout the month. This charge is estimated at $16.53 / kW at the peak demand. RIT also produces electricity using its 2-megawatt solar field, which can help offset the demand when it is actively producing electricity.

The goal of this project is to forecast RIT's net energy demand, and to use this forecast to identify potential peak days before they happen. By accomplishing this, RIT can save thousands of dollars every month on their electricity bill by running demand response activities to reduce the energy demand on predicted peak days. However, successfully forecasting RIT's energy demand is not an easy task, as there are many different factors that can affect the net energy demand on any given day.

## II. DATASET DESCRIPTION

A dataset has been provided for this project. This dataset has hourly observations, spanning from July 1st, 2018 to February 28th, 2021. There are 45 possible inputs to our neural network. Some examples of these inputs are provided in Table 1.

*Table 1: Variables Available to be Included in the Model.*

| | |
|---|---|
| Time | COVID (pre or during), Year, Month, Day of the Week, etc. |
| Demand | Maximum Demand of the Previous Day, etc. |
| Weather | Weather (sunny, cloudy etc.), Temperature, etc. |
| RIT | Events (Spring Break, Graduation, Career Fair, etc.), Semesters (fall, spring etc.), Classes in Session, etc. |

In order to prepare these variables to be used by the neural network, they had to be classified into two categories. There are a number of these variables that are categorical, such as year, month, and whether there are classes in session. Indicator variables must be created for all of these categorical variables. For example, Year, which initially consists of four values, 2018-2021, must be split into 3 (K-1) dummy variables. The other category of variables is continuous variables. The best way to provide continuous variables to a neural network is normalizing the variables to the same scale. In this case, all continuous variables were normalized to be between zero and one.

An important part of the variable identification process is to make sure that variables included in the model are not highly correlated to each other. This process will be completed when the variables are being selected for the model.

## III. METHODOLOGY

In this project a feed forward neural network with one hidden layer was used. A simple example of this is shown in figure 1.

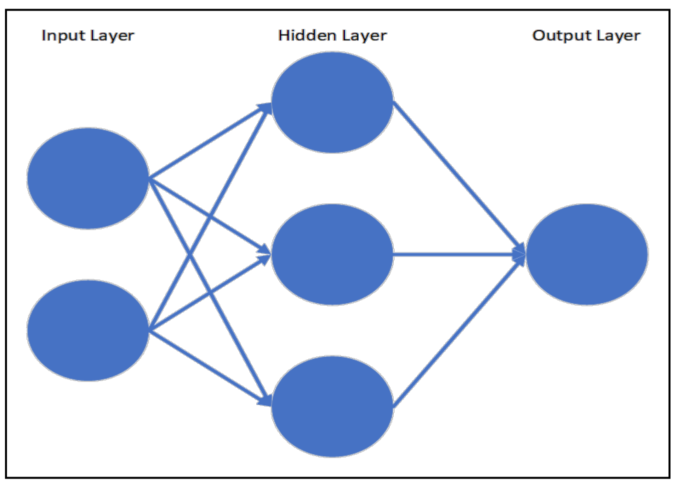


*Figure 1: Feed Forward Neural Network with 1 Hidden Layer*

The process of deciding the maximum number of iterations, number of hidden nodes, and the decay rate involves checking various values for these parameters, and plotting them against mean squared error (MSE) to analyze how model fit is affected. Both the MSE of the training data, and the MSE of the validation data are analyzed, and a value is chosen for each parameter that minimizes the error for both training and validation. This is more art than science, as the same value does not minimize both in most cases. A value that provides a relatively low MSE for both training and validation should be chosen.

The training and validation data consists of all available data except the weeks being tested. The test weeks include the last week of February 2021, and the easy and hard forecasting weeks identified in part 2 of this project. With the test weeks removed, the remaining data is split into training and validation sets randomly, with 80% of the data for training and 20% for validation. In order to ensure that the random split was always the same, the seed was set in R. This is important because all models that are evaluated should be evaluated on the same training and validation data to make comparing two models valid.

In order to train the neural network, the inputs must be identified. These will be a subset of the variables that were discussed in the dataset description section. In order to identify which inputs are helpful to include, the correlation for each input with the response variable net demand was calculated. These inputs were sorted from highest to lowest correlation with net demand, and any inputs with a correlation coefficient with an absolute value greater than 0.1 were taken into consideration. As previously mentioned, these inputs cannot all be included in the model due to possible correlation with one another. In order to ensure that correlated inputs were not included in the model, a correlation plot was created. A reduced version of this with fewer parameters is included for the sake of visibility (Figure 2).

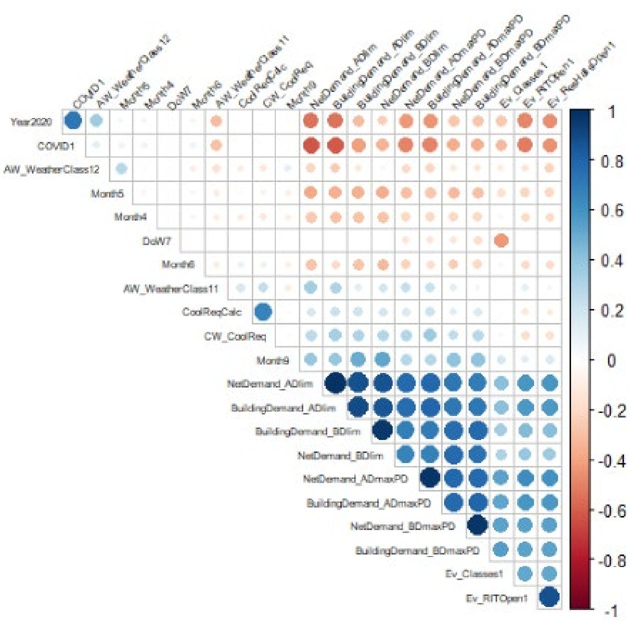


*Figure 2: Reduced Version of Correlation Plot*

If two variables were correlated, the variable with the higher correlation with net demand was included in the variable selection process.

## IV. Results of Model Fitting

Models with different inputs were tested, and the summary of this testing is shown in Table 2. Because each model has different inputs, the parameters for each model were reoptimized to provide the best possible fit for those inputs. The model with the lowest root mean squared error (RMSE) will be the chosen model going forward.

*Table 2: Summary of Model Testing*

| Model # | # of Inputs | Max # of Iterations (i) | # of Hidden Nodes (h) | Decay Rate (d) | RMSE |
|---|---|---|---|---|---|
| 1 | 18 | 500 | 6 | .01 | 546.69 |
| 2 | 17 | 400 | 6 | .001 | 551.17 |
| 3 | 19 | 500 | 8 | .02 | 529.16 |
| 4 | 50 | 300 | 6 | .001 | 378.24 |
| 5 | 49 | 300 | 6 | .01 | 375.34 |
| 6 | 51 | 300 | 7 | .001 | 366.42 |

Model 6 has the lowest RMSE, so this model was chosen going forward. Model 6 consists of the following parameters:

- All variables for month, day of week, and hour of day
- Previous day maximum net demand
- Classes in session, RIT open, event increasing demand
- Various types of weather (clear, cloudy, rain)
- Building heating and cooling required
- Pre-COVID or Post-COVID

A walkthrough of the process to train and validate the models will be completed using the chosen model. This follows the steps discussed in the methodology section. The first step was to plot the maximum number of iterations vs MSE (Figures 3 & 4).

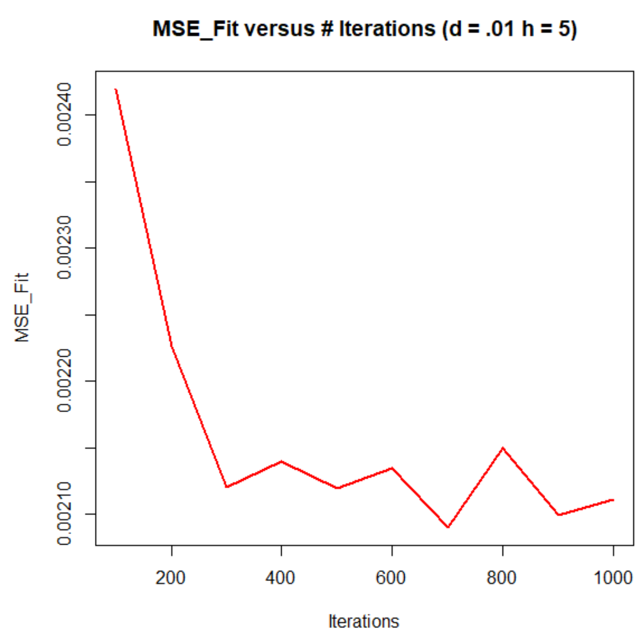


*Figure 3: # of Iterations vs Training MSE*

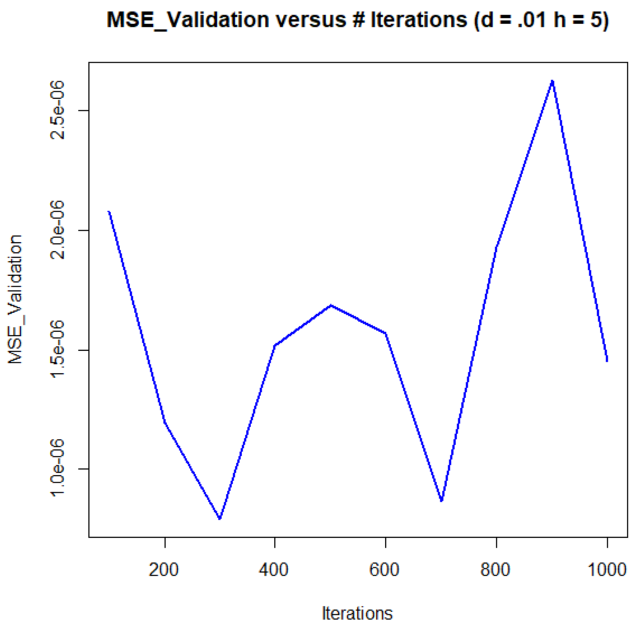


*Figure 4: # of Iterations vs Validation MSE*

The results of this indicated that 300 iterations was a good number to choose when training the model. The next step was to plot the number of hidden nodes vs MSE (Figures 5 & 6).

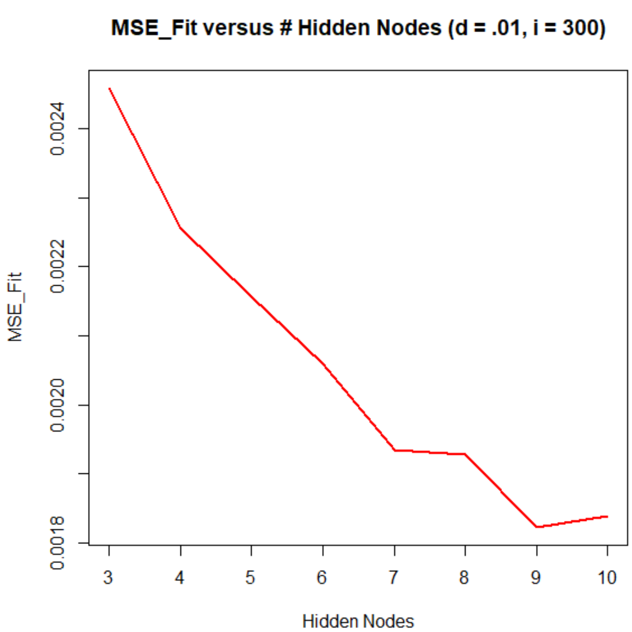


*Figure 5: # of Hidden Nodes vs Training MSE*

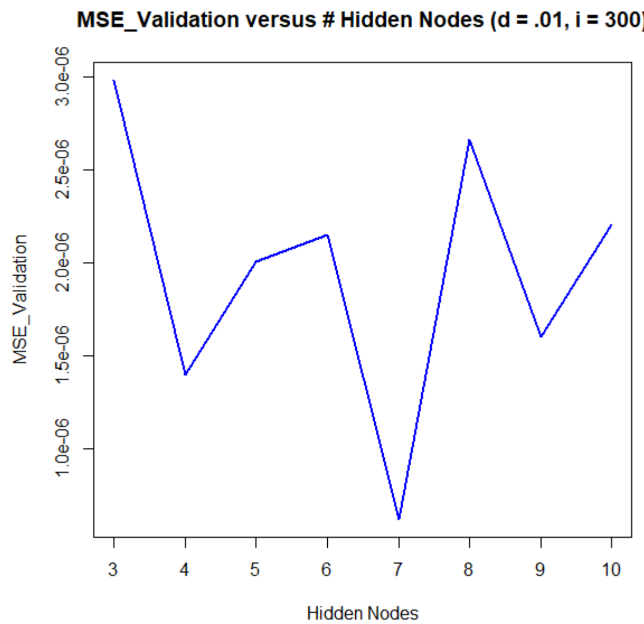


*Figure 6: # of Hidden Nodes vs Validation MSE*

The results of this indicated that 7 was a good number of hidden nodes to choose when training this model. The final step was to plot the decay rate vs MSE (Figures 7 & 8).

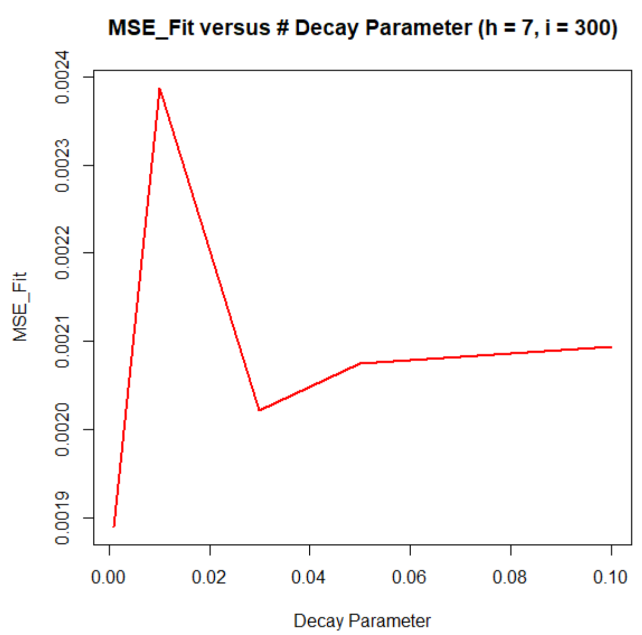


*Figure 7: Decay Parameter vs Training MSE*

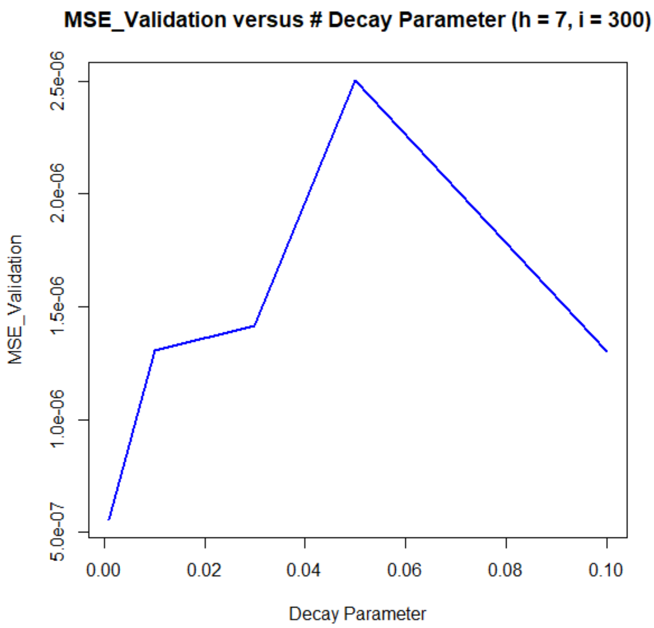


*Figure 8: Decay Parameter vs Validation MSE*

The results of this indicated that a small value for decay rate provides the best fit when training this model. 0.001 was chosen for this parameter. With the parameters chosen, the residuals of the model were evaluated to validate that the fit passed the residual assumptions (Figure 9).

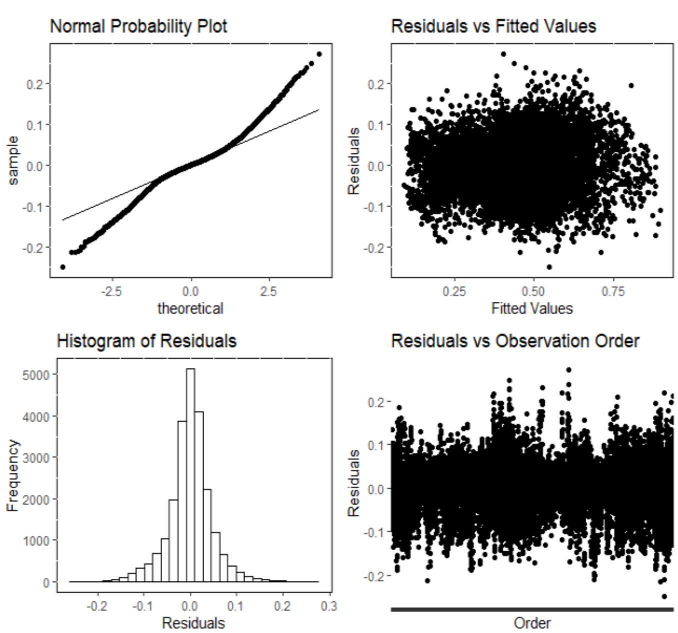


*Figure 9: 4 in 1 Plot of Residuals*

These residual plots show that the model passes the assumptions of normality, equal variance, autocorrelation, and linearity.

## V. Results of Model Forecasts

With the optimal model inputs and parameters chosen, the neural network was retrained on all available training data, which was a combination of the training and validation sets. Forecasts were then created for the easy week, hard week, and last week of February 2021. The easy week forecast will be discussed first (Figure 10).

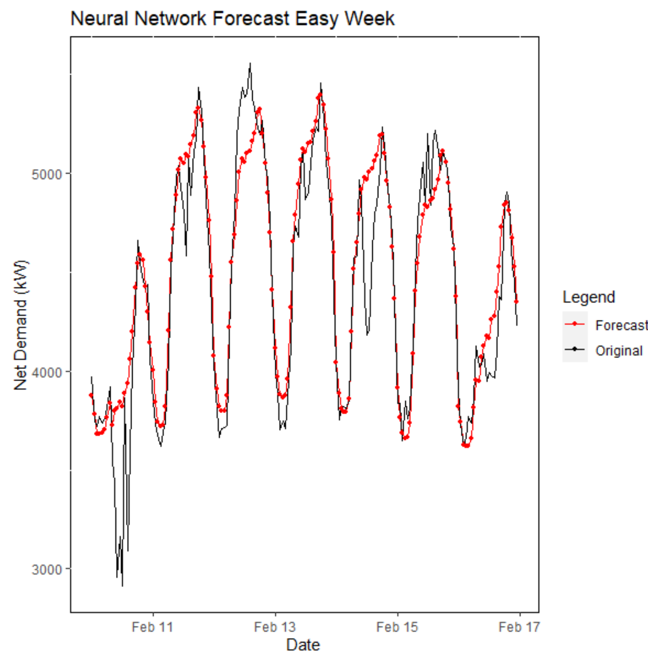


*Figure 10: Easy Week Forecast*

This forecast looks good and has a mean absolute percent error (MAPE) of 3.56. It struggles to capture the drop on the first day, but missing a dip is not necessarily bad. Missing a peak would be much more costly. The residuals of this forecast were evaluated to ensure it passed assumptions (Figure 11).

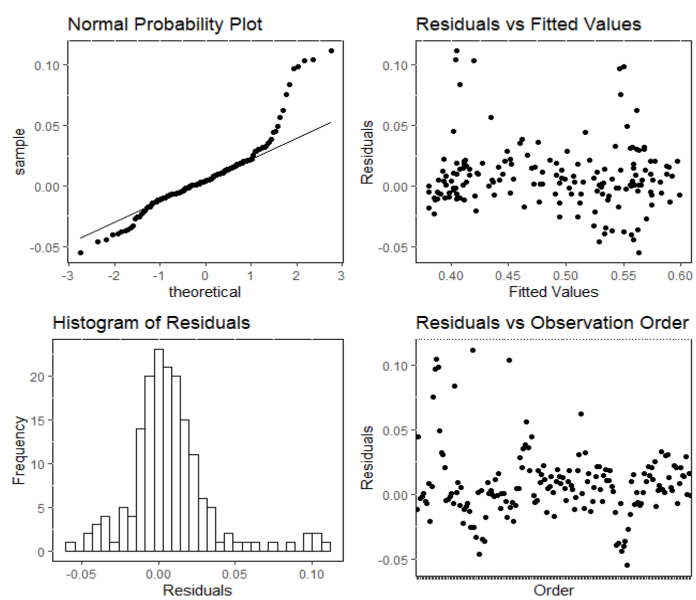


*Figure 11: Easy Forecast Residuals*

These residuals look good and pass assumptions. To test whether the model can predict peaks, the hard week forecast was created (Figure 12).

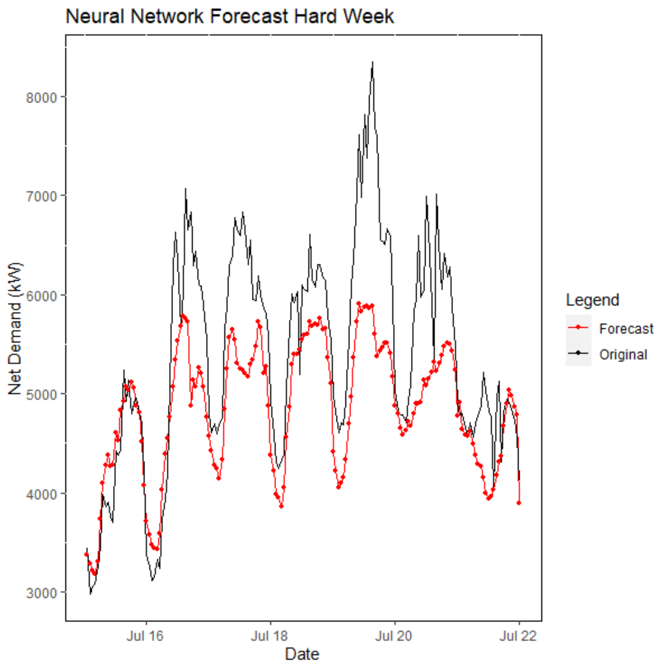


*Figure 12: Hard Week Forecast*

This forecast does not look as good and has a MAPE of 10.95. It struggles to predict the peak days that are present during this week. This could be due to this set being the highest peak in the entire data set, so the model has not seen any similar data. The residuals of this model were evaluated (Figure 13).

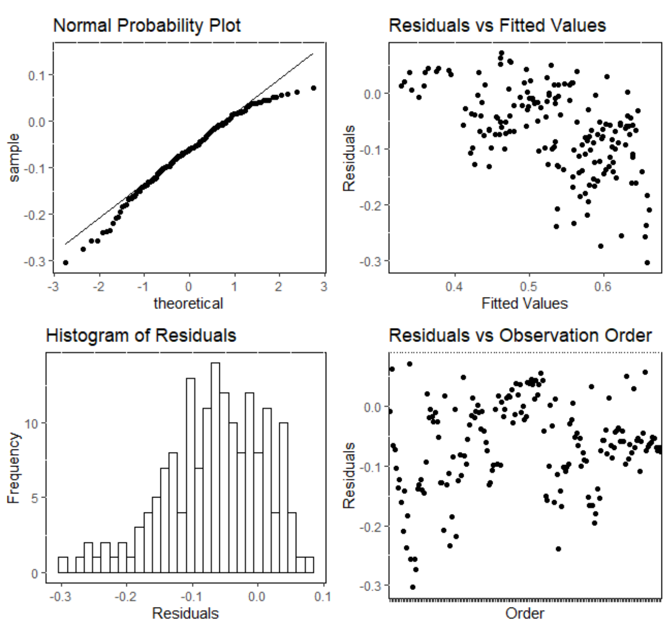


*Figure 13: Hard Forecast Residuals*

These residuals do not look as good, and do not pass assumptions. This is due to consistently underpredicting the data in this set. The last test week to forecast is the last week of February 2021 (Figure 14).

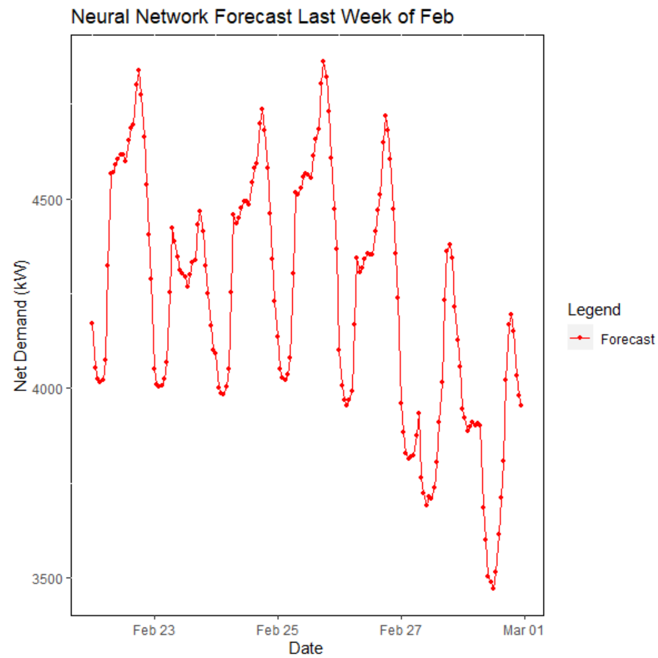


*Figure 14: Last Week of February 2021 Forecast*

This forecast looks fairly normal, with the weekend being lower than the weekdays. Another good thing to note is that on Tuesday Feb 23rd there were no classes, and the model predicts a lower net demand on this day compared to the other weekdays. This looks like a promising forecast.

## VI. Conclusions and Recommendations

As expected, this neural network model significantly outperformed all of the other models that have been tested in previous parts of this project. The summary of all models is shown in Table 3.

*Table 3: Summary of Models Tested*

| Model | Easy MAPE | Hard MAPE |
|---|---|---|
| Holt Winters | 4.701 | 16.616 |
| Seasonal ARIMA | 3.256 | 22.374 |
| Dynamic Regression | 5.917 | 21.643 |
| Fourier Transforms | 4.713 | 29.097 |
| Neural Network | 3.561 | 10.949 |

If RIT were to use one of these models, the neural network model would be highly recommended due to its forecasting accuracy and ability to account for many factors at once. If a new factor is identified to be important, it is easy to retrain the model to include this new factor. The main concern with this model is that it did not fully predict the peaks in the hard test week. However, if this week is included in the training data, the model may be more likely to predict similar peak days in the future. For a typical week, a MAPE between 3-5 could be expected. For a difficult week, a MAPE of <10 could be expected. RIT would benefit from using this neural network model to forecast their net energy demand, and help reduce their electricity costs.