FLOW OF DESIGN

1)Choosing the Data set from 1)a,1)b 1)c

We have chosen 1)C as our final filled data to proceed further for classification and prediction process

The reason we have chosen 1)c is that our performance metrics are much better when we have replaced the zeros with na.fill function

We have installed zoo package. We have gone for na.fill option instead of na.approx and na.locf because if there are any NA values at the end or beginning of a vector then na.approx and na.locf doesn't replace those NA's instead they delete those observations

Hence keeping this point in mind we have choosen na.fill over the other two

in.csv format which have the performance metrics and regression outputs of raw data with zeros in 1)a 1)b and 1)c approaches and they are same for all the three approaches

So in the compiling part first you have to run the file" 1)C + 2) PREDICTION WITHOUT NEURAL NETWORKS" for this it will ask an input of NewData

And by the end of this compiling we will getting performance metrics of 1)c approach filled data and also performance metrics of regression trees

In the 1)b we were getting 48 observations of kWh which were negative and initially we thought of deleting those observations and going ahead with the rest of the data but our performance metrics for these observations without zeros(after deleting negative value observations) wasn't good compared to 1)c. Hence we have decided to go ahead with

7) We have noticed that the order of compiling the files plays an important role in the outcomes of the values and models.

2. PREDICTION

In building Regression trees we have considered all the parameters except for Account and Year.

Later on we have taken the model built for this regression trees in predicting the kWh values for each hour.

The difference in classification was we were predicting categorical values like “Above normal’ and “Optimal” where as in Prediction of kWh we were predicting quantitative values like kWh

We were facing challenges in forecasting KWH_Class since it is of factor format and while it is given out form the predict function it would be in the form list data type. So handling the data type while forecasting Categorical values was a good challenge and we have successfully overcome it by makng changes to the data format before giving the in the classification tree model.

PART 2

We have calculated the KWH_Class column according to the requirements and used it in builing the logeistics regression,classification trees and Neural network models

Classification trees model

Account Year and kWh parameter were ignored in building the tree

Tree model which is not pruined was good in predicitng the accuracy of values of "Above_Normal" in KWH_Class.

the pruined tree model was good in predicting the values of "Optimal" in KWH_Class

But the overall error rate hasn't improved by pruining the tree

The error rate were 11.02 when the tree is not pruined and 11.48 when the tree is pruined

The residual mean deviance were 0.52 when the tree is not pruined and 0.5667 when the tree is pruined

Going by the above statistics we have decided that our tree model which we have built by ignoring Account, Year and kWh parameters for predicting the KWH_Class is good.

Neural Networks

1. Splitting the data into train and test data set for linear model

2. Fitting t he linear model with the sampleformat dataset using glm()

3. Predicted data from lm.fit

4. NeuralNet Fitting : Scaling the data using the Min and Max values

5. Splitting the data into train and test using indexed value

6. Performing Neural Network training to plot the Neural Network graph

7. Prediction is done and the covariate value should be same as nn's covariate number

8. Results from NN are normalized

9. Cross Validations using the Linear and Neural Net

Observation:

1. The smaller the threshold longer it takes to process and takes more number of steps to predict.

2. While computing the pr.nn (prediction of neural networks) the covariate value should be same as nn's covariate number