Case 2 Report

Objective:

The objective of the problem statement is to build a forecasting model to predict the energy usage (Kwh) for the city of Boston using Tree model and neural network classification. To achieve the above, this report contains the following implementations:

1. Algorithm Implementation
    A. Dara Wrangling, Cleansing and Multiple Linear Regression
    B. Prediction
    C. Forecasting
2. Classification

Part 1) Algorithm Implementation

1) Dara Wrangling, Cleansing and Multiple Linear Regression

Following are the steps for removing zero entries:

1) Converting to the required output file from the input file
    a) Take the data variable and change it to the required format using posixlt function.
    b) Calculate the variables day, month, year, weekday and day of week
    c) Calculate whether it is a peak hour or not
    d) Construct the dataset with peak hour and temperature
2) Retrieve the temperature data from wunderground and combine it accordingly
3) Column bind the temperature data with the existing dataset
4) Creating the structure of the dataset and populating the values
5) Neglecting the NA's in Kwh
6) Converting 0's to NA
7) Neglecting the NA's in Kwh

Output(Metrics) of the filled data:

	ME	RMSE	MAE	MPE	MAPE
Test set	-0.24492	93.43287	66.1188	-7186.19	7217.552

Following are the steps to replace the zero's with model generated values:

1) Split the data into training and testing
2) Construct a logistic regression model using the variables
3) Measure the predictive accuracy
4) Convert the predicted value to a data frame
5) Column bind the predicted value with the existing dataset
6) Replacing the 0s with the predicted value

7) Neglecting the newly generated column and write into CSV the filled data

Output(Metrics) of the filled data:

	ME	RMSE	MAE	MPE	MAPE
Test set	3.856837	73.89283	41.35294	-6294.1	6317.858

Following are the steps to implement Zoo Package for replacing NA's:

1) Repeat the steps you used for removing zero entries
2) Neglect the NAs data in the kWh column
3) Neglect 0s in Kwh by Changing the 0s to NA in Kwh and not considering it
4) Replace NAs using the Zoo package function "na.locf"

Output(Metrics) of the filled data:

	ME	RMSE	MAE	MPE	MAPE
Test set	4.236474	85.86203	58.84939	-17596.6	21310.96

Following are the steps to compute the values of in-sample MAPE, RMS and MAE when applied this to the above 3 models

1) Read the CSV File
2) Take 75% of the sample size and set the seed to make your partition reproducible
3) Split the data into training and testing
4) Construct a logistic regression model using the variables
5) Measure the predictive accuracy

Using the non-zero dataset to create the model will ensure that there are no abnormal values used to design the prediction model. Once we generate the values with the non-zero dataset and use it to replace the zeros in the original dataset, we can be assured of a better prediction model since we have predicted the values and used them to replace the outliers.

2) Prediction

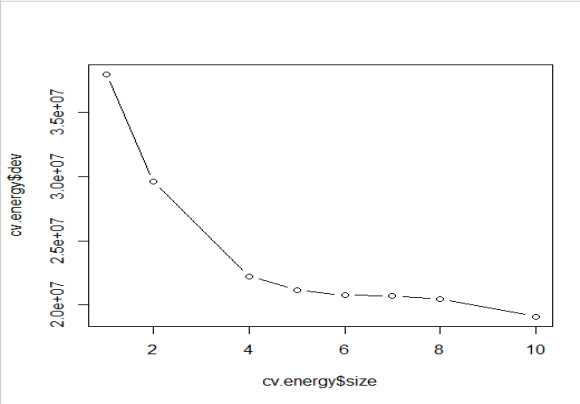
Following are the steps and techniques to build models for prediction of KWH using Regression tree:

1) Reading the CSV and assigning the CSV file to data frame
2) Getting the weather data for a year in intervals and combining them as a data frame
3) Performing the left join so that 2 data frames are merged
4) Considering the data set with temperature as 0 and NA and neglect NA in kWh
5) Set the seed to make your partition reproducible and split the data into training and testing

6) Fit in a regression tree

```
Regression tree:
tree(formula = kWh ~ Temperature + Weekday + DayOfWeek + peakhour,
    data = tree_data, subset = train)
Variables actually used in tree construction:
[1] "Temperature" "peakhour"    "Weekday"
Number of terminal nodes:  10
Residual mean deviance:  4603 = 18390000 / 3996
Distribution of residuals:
    Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
-263.100  -29.410   -7.603    0.000   23.990  302.700
```

7) Plotting the tree

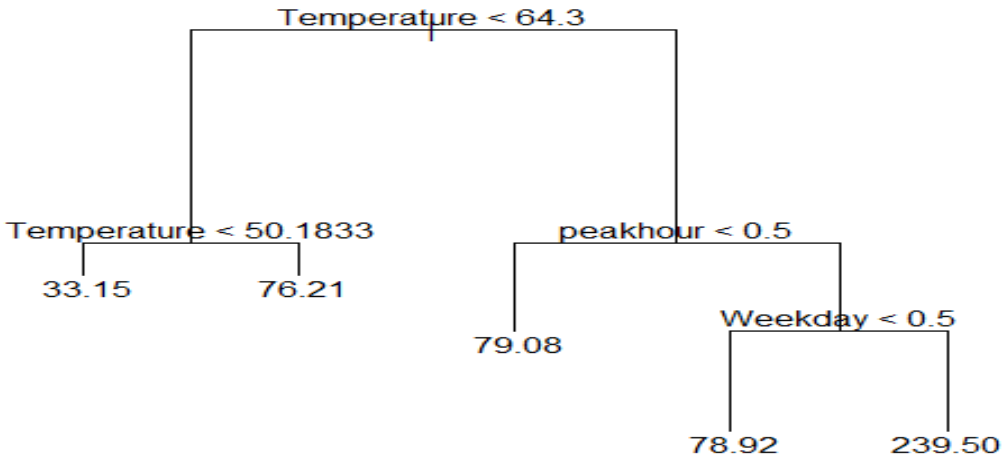


8) Pruning a tree

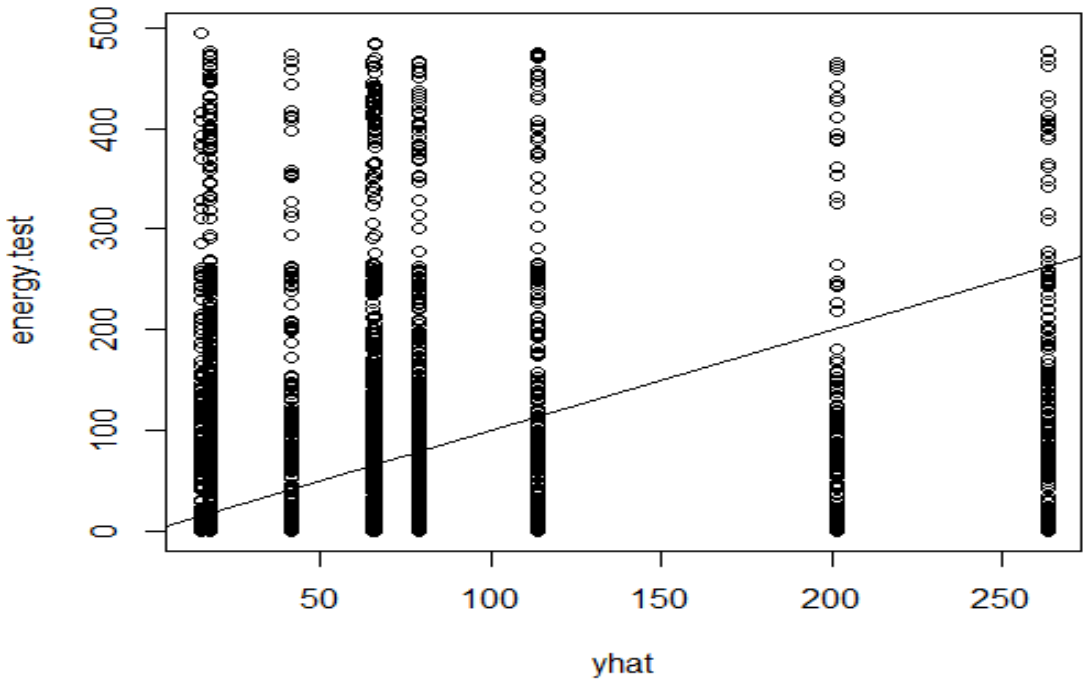
prune.energy =prune.tree(tree.energy, best = 5)

plot(prune.energy)

text(prune.energy, pretty = 0)



9) Predicting the kWh value by plotting the yhat with energy.test

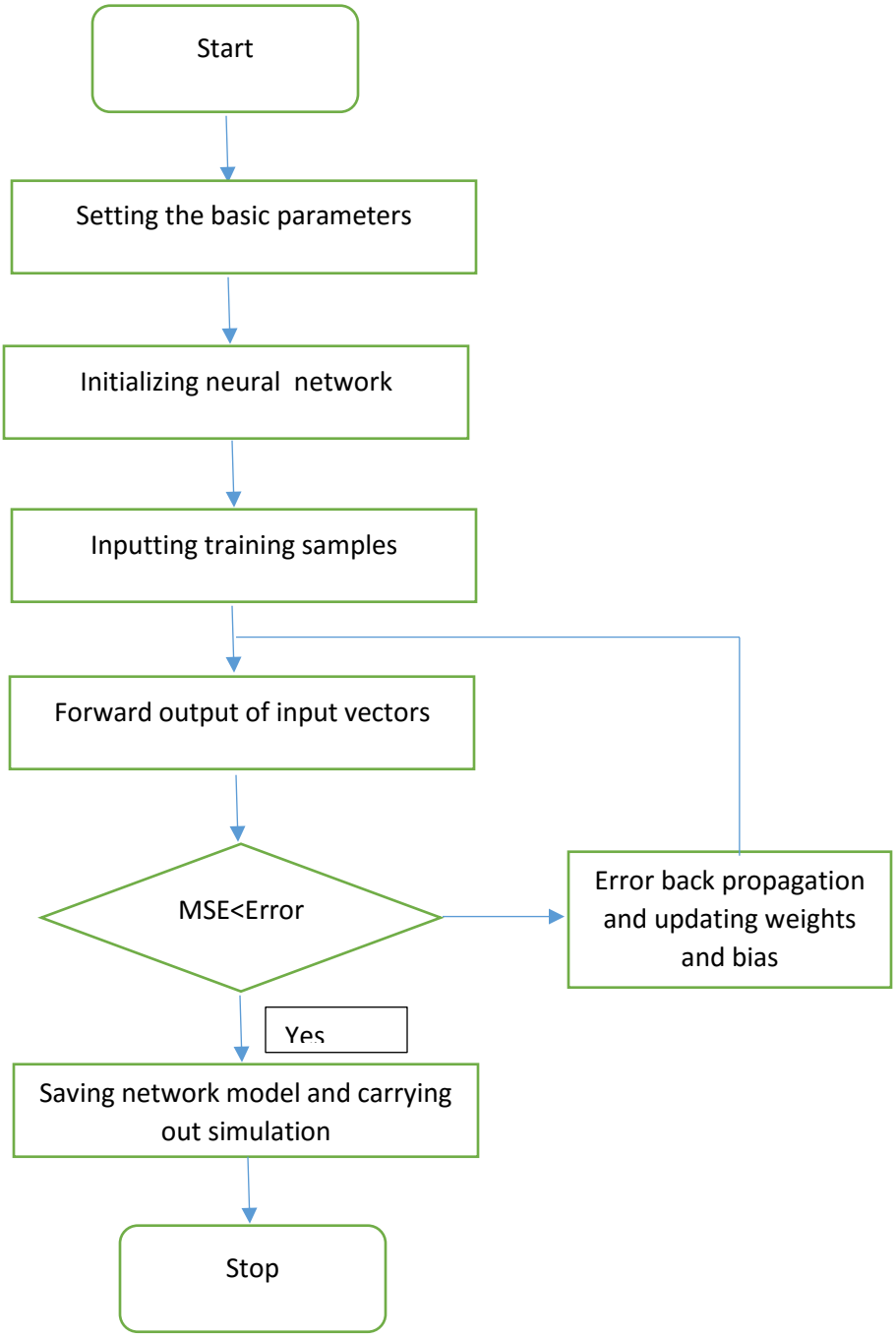



```
yhat=predict (tree.energy, newdata4 = tree_data[-train,])
  energy.test=tree_data [-train, "kWh"]
  plot(yhat,energy.test)
```

10) Finding the mean value

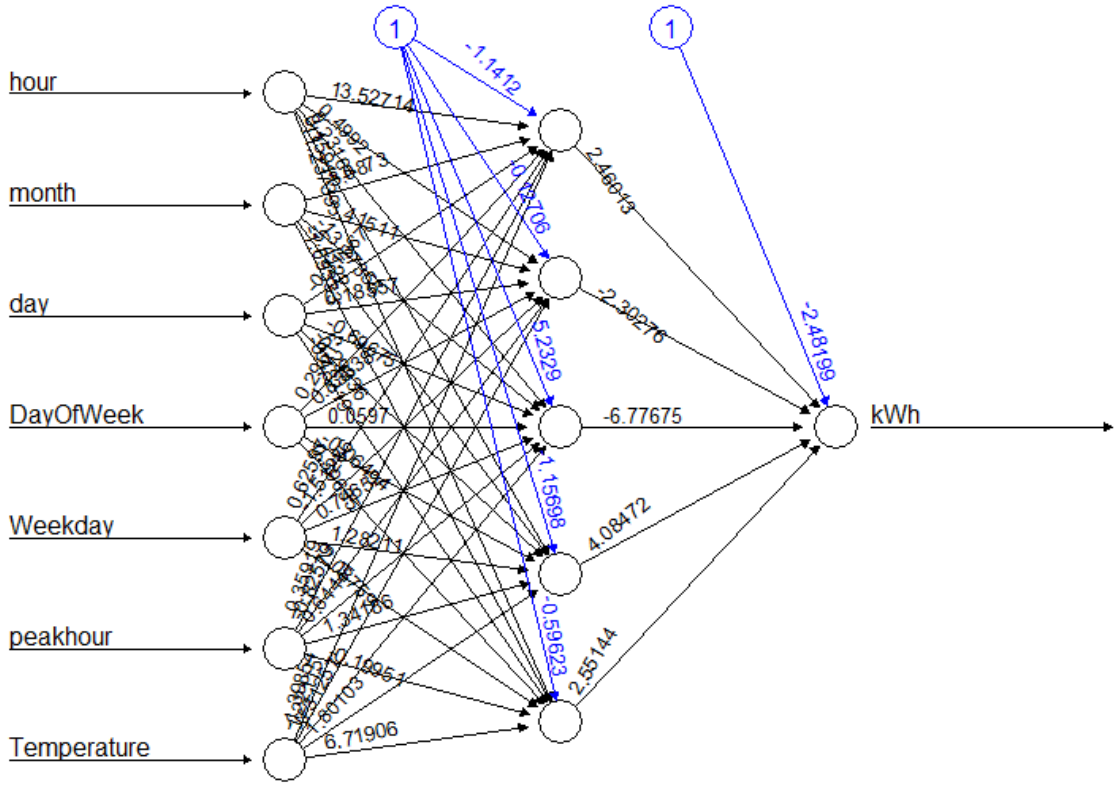
```
mean((yhat -energy.test)^2)
[1] 15192.69
```

Neural Networks Flowchart:



Following are the steps and techniques to build models for prediction of KWH using Neural Networks:

1) check that there are no columns with 0s
2) Setting the max and min values
3) Converting the columns to numeric
4) Scale the data and split them into training and testing
5) Fit in neural networks model and plot



6) Compute the values of MAP MAPE and RMSE for Prediction performance

3) Forecast

Following are the steps to predict the power usage in kWh for Forecasting using Regression tree and neural networks:

1) Set the seed to make your partition reproducible
2) Fit in the tree model and find the summary of the tree
3) Neglect the unwanted columns
4) Setting the max and min values
5) Converting the columns to numeric
6) Scale the data and split the data into training and testing
7) Fit in neural networks model

8) Compute the values using regression tree as well as neural network

Part 2) Classification

1) Logistic Regression

Steps to predict the KWH_Class variable using Logistic Regression:
1) Set the seed to make your partition reproductible
2) Split the data into training and testing
3) Fit into a linear regression model
4) Take the fit summary and measure of predictive accuracy

2) Classification Tree

Build the tree based on the training set

```
classification3.train = tree(kWh_Class ~ kWh, classification3, subset = train)
```

Evaluate its performance on the test data

```
tree.pred = predict(classification3.train, classification3.test, type ="class")
```

```
table(tree.pred, classification3)
```

3) Neural Network

```
library(nnet)
```

```
ideal <- class.ind(seeds$kWh)
```

```
train<- sample(1:nrow(seeds),5359)
```

```
test<-setdiff(1:nrow(seeds),train)
```

```
seedsANN =
nnet(class.ind(kWh_Class)~Temperature+hour+month+day+year+peakhour+Weekday,
seeds[train,], size=10,  softmax=TRUE,maxit=10)
```

```
predict(seedsANN,seeds[train,-12], type="class")
```

```
output<-table(predict(seedsANN, seeds[test,-12], type="class"),seeds[test,]$KWh_Class)
```