

Laxman Dutt Degala

Vishnu Gude

Ayyappa

Akhileshwar

 | KDM – CS560 | March 31, 2014

CAAM for Smart Energy

Increment 2

**Please find below our updates for Increment 2:**

1. Recap of Increment 1
2. Picking the right data for training and testing; Deciding on the target field
3. Model Creation
4. Model Evaluation
5. Model Deployment - Exporting model to PMML (Work in Progress)
6. **Recap of Increment 1:**
7. Data has 100000 records and 1350 attributes (Columns). Decision on attributes has been made, Out of 1350 attributes we are considering 25 attributes which could decide the energy consumed by a household for a month and predicts the $ amount for the energy consumed.
8. Decision has been made to use R and Rattle (**R** **A**nalytics **T**ool **T**o **L**earn **E**asily), which is a graphical Data Mining Application.
9. Explained how RATTLE works.
10. Description about various patterns/visualizations (Boxplot, Histogram) has been provided.
11. **Picking the right data for training and testing; Deciding on the target field**

Out of 25 attributes that were finalized in increment 1, we will have to

decide on what exactly is the Identifier, Inputs and Target fields for

building a model. This is critical as we should have a high prediction rate.

We have decided to consider the DOLLAREL as the target field, which is the dollar amount that is to be paid as the electricity bill and DOEID is considered as the Identifier and remaining 22 fields are considered as inputs.

1. **Model Creation:**

In order to create a model using RATTLE with the data that we have; we will have to complete the following steps, in a sequence:

1. Data
2. Explore
3. Test
4. Transform
5. Cluster
6. Associate
7. Model

And

1. Log.
2. **Data:**

Data that we currently have is in .CSV format. We will have to upload the file and decide the Input/Target/Identifier as explained in point 2.

We can also divide the data in this phase as Training, Validation and Testing. This can be done by giving the percentage of data we want to use for creating training, validation and testing. Example: 70/15/15 will randomly divide the data as 70% for training, 15% for validation and 15% for testing.

Once this is done, we will have to execute the data, by clicking on the execute command.

1. **Explore:**

This has been explained in increment 1. Basically we have different visualizations of the data based on Min/Median/Max. We have several options like Box Plot, Histograms, Cumulative, Benford.

1. **Test:**

It’s more like a statistical test. This is applied to two different samples. Its assumed that we have 2 observations and that we are testing for a change, usually from one time period to another.

This is not required for us as we are not working on two samples.

1. **Transform:**

This step helps us in refining the data, if we feel the data is not consistent.

A Transform could be a Rescaling, Imputing, Recoding or a Cleanup.

We don’t need to worry about this step as well, as our data is clean and neat.

1. **Cluster:**

We can cluster our data by using different techniques like KMeans, Ewkm, Hierarchical, and BiCluster.

We have considered the KMeans technique to cluster the data. A KMeans algorithm will search for K Clusters (which we will have to specify). The resultant K clusters are represented by the mean or average values of each of the variables. And also, KMeans works only with numeric variables.

1. **Associate:**

This is the association rule mining technique which identifies relationships or affinities between observations between variables. These relationships are expressed as a collection of association rules. This is also referred to as a basket.

For the data that we are working on, Associate rule mining cannot be done as our input fields are not categorical.

1. **Model:**

This is the important phase where we create our model based on the data that we have. We should decide on the model which suits our data.

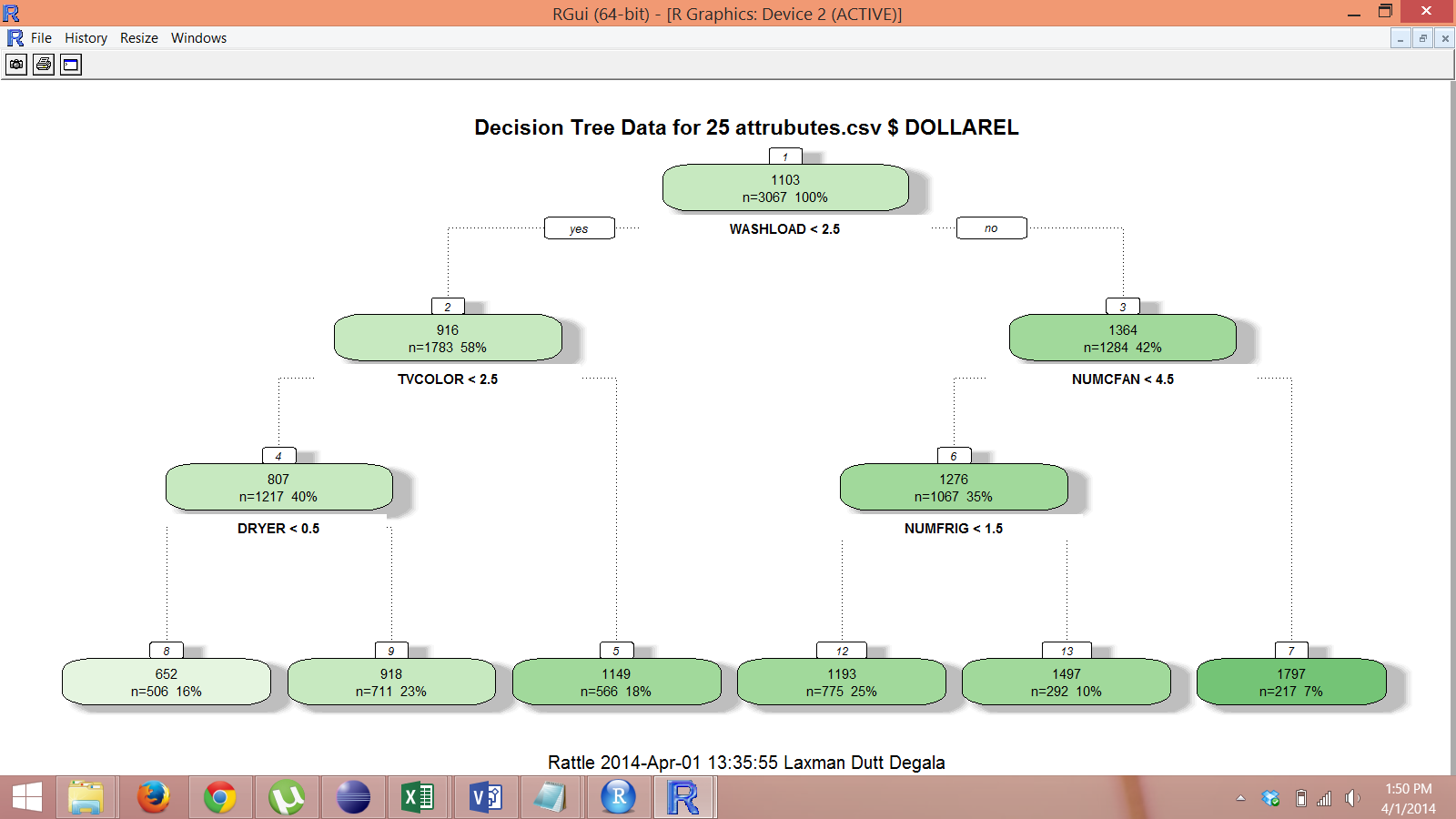
We have Decision Tree model, Random forest model, Linear and generalized linear model, Neural Network Model.

We can build all the models and see which model suits the best for our data.

Models built using our data:

**i) Decision Tree Model:**

Following is the binary tree model for classification, which is built using ‘rpart’.



Following are the rules that are created using the Binary tree model.



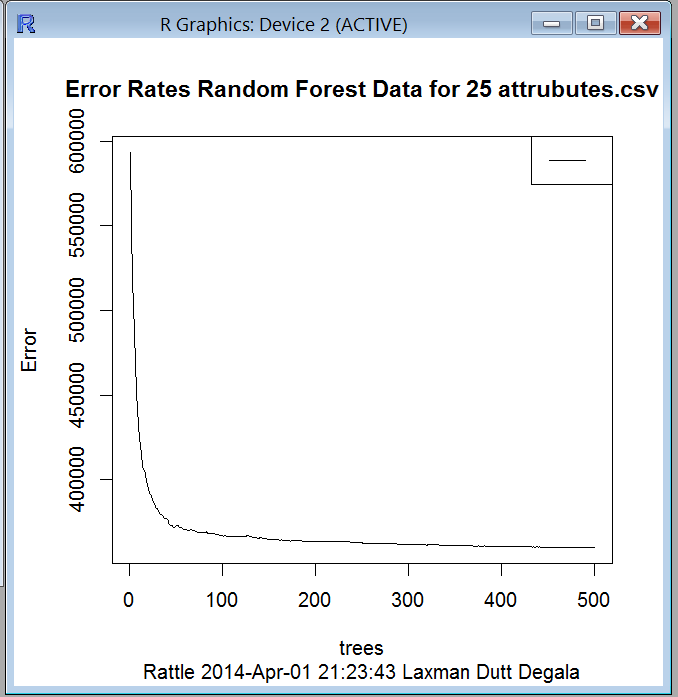
1. **Random Forest Model**:
2. Using Random Forest Model, we can get the list of variables/attributes which has maximum influence on target.



1. We can also get the set of rules obtained from a specific tree.



1. We can plot error rates, which is useful for deciding the optimal number of decision trees to be built.



1. **Neural Net Model:**

This model gives the weight for each node.

Please find below the attachment for node wise details.



1. **Log**

For all the steps performed in Rattle(explained in above steps), there is R code that’s generated in the background. This code can be seen in the **Log**.

1. **Model Evaluation:**

We will have to evaluate a model based on various scenarios like Risk, ROC. Risk Chart is used to evaluate performance of a model.

We can have *Scores* for all kinds of data: Training, Validation and Testing,

for all the models. Based on the scores, we can decide on which model to

go. Scores are generated and we are in the process of evaluating the best

suitable model.

1. **Model Deployment: (Work in Progress)**

Once the model is evaluated, it has to be deployed.

It has been decided to deploy the model by exporting to PMML – Predictive Model Markup Language.

PMML provides a standard language for representing data mining models. PMML is an XML based standard that is supported by major commercial data mining vendors and many others open source data mining tools.

Thus created PMML can be imported my applications like Zementis ADAPA tool for online execution.

The **Export** button in Rattle will deploy the model as PMML.