Data Science with R
Getting Started with Rattle

Graham.Williams@togaware.com

9th June 2014

Visit http://onepager.togaware.com/ for more OnePageR's.

Rattle (Williams, 2014), the R Analytic Tool To Learn Easily, is a graphical data mining application built using the statistical language R (R Core Team, 2014).

Rattle runs under various operating systems, including GNU/Linux, Macintosh OS/X, and MS/Windows. R needs to be installed on your system and then

```
install.packages("rattle")
```

Rattle's user interface steps through the data mining tasks, recording the actual R code as it goes. The R code can be saved to file and used as an automatic script, loaded into R (outside of Rattle) to repeat the data mining exercise. Repeatability is important both in science and in commerce!

This laboratory provides a quick start guide to building our first models using Rattle. Record in a report the tasks you complete, including observations of the data and plots you might generate. This is to be submitted for assessment.

The required packages for this module include:

```
library(rattle)
```

As we work through this chapter, new R commands will be introduced. Be sure to review the command's documentation and understand what the command does. You can ask for help using the `?` command as in:

```
?read.csv
```

We can obtain documentation on a particular package using the *help*= option of `library()`:

```
library(help=rattle)
```

This chapter is intended to be hands on. To learn effectively, you are encouraged to have R running (e.g., RStudio) and to run all the commands as they appear here. Check that you get the same output, and you understand the output. Try some variations. Explore.

1 Starting Rattle

Rattle is started from R. There are several ways that Rattle might be configured for your particular computer. For example, some installations set up an icon on the desktop from which Rattle is automatically invoked. The most common way though is to start up Rattle from within R.

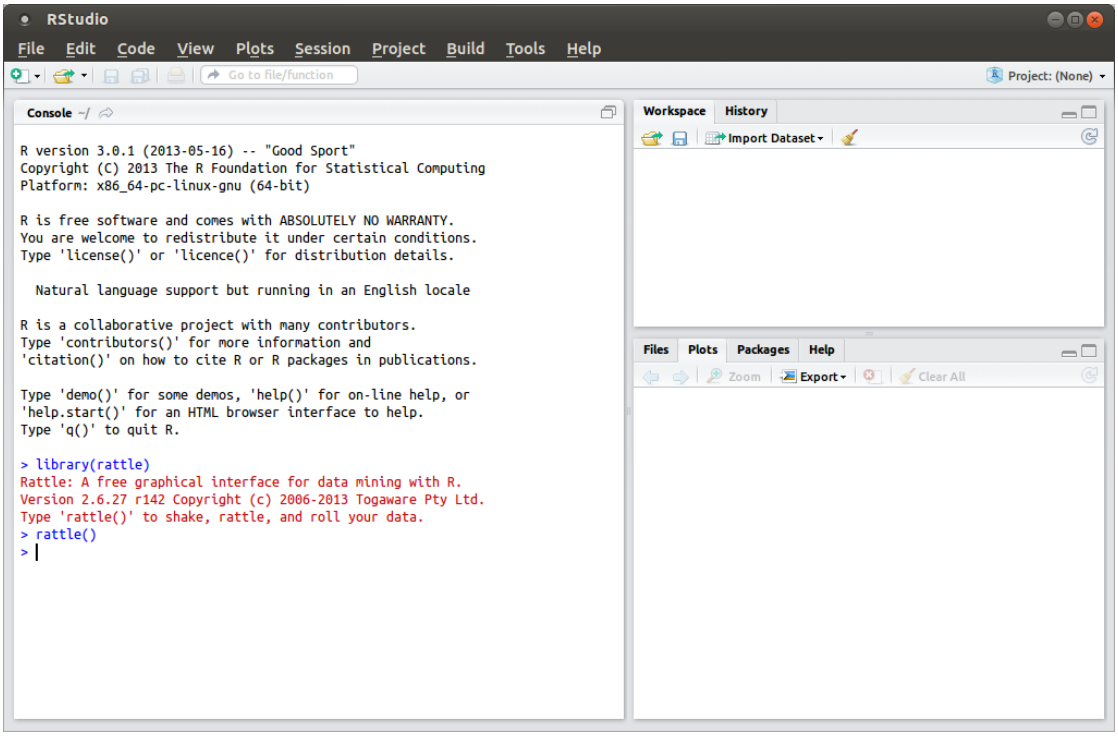
Even starting up R depends on your particular platform. Generally, it is started from a desktop icon or from the Application menu. Alternatively, on Linux it is often started up from a terminal window, like gnome-terminal or xterm. From the terminal we simply type the command R to invoke R itself.

An increasingly popular approach is to use RStudio. RStudio includes an R console. We can see the RStudio application below, with the commands to start up Rattle. Do note that this only works with the Desktop version of RStudio and not the server version of RStudio. The server version runs the interface in a browser on your desktop and communicates to a remote server running R itself. RStudio handles all of the graphical interface. Because Rattle has its own graphical interface, RStudio is unable to capture that interface from the server and display it on your desktop.

We can access the desktop version of RStudio from a server by running an XWindows server, such as xming, on our desktop.

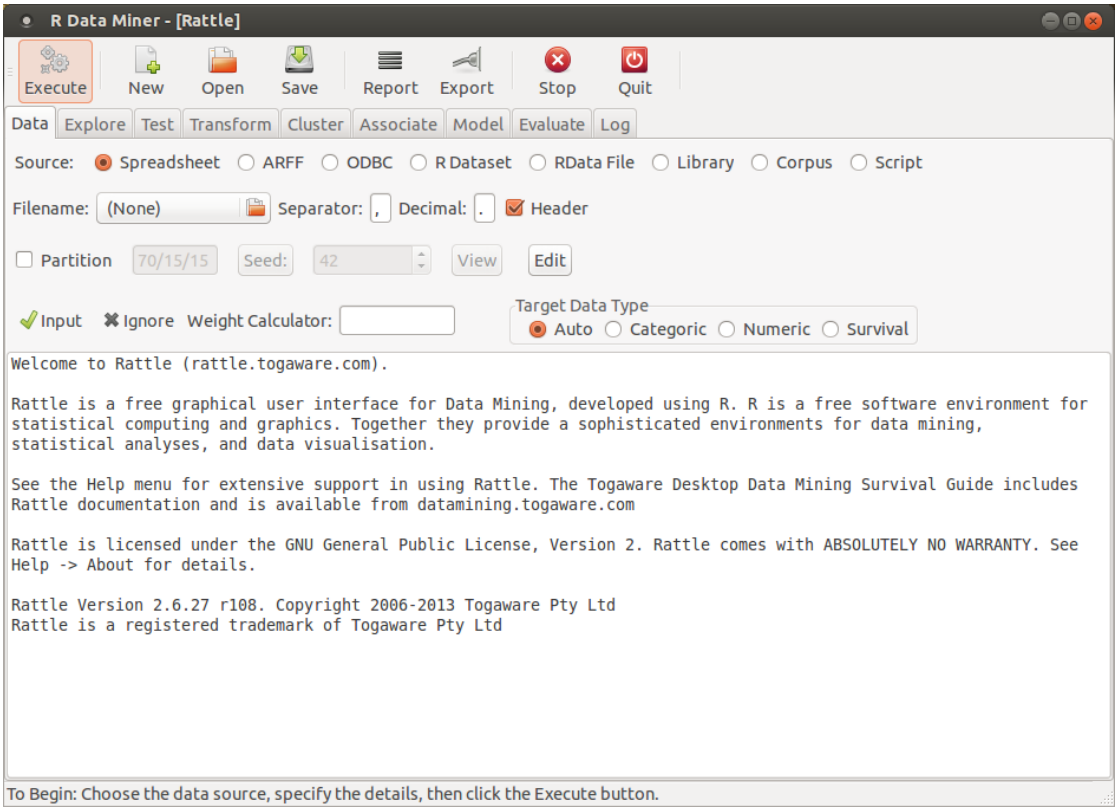
Whichever way we start R, we initiate Rattle with:

```
library(rattle)
rattle()
```



2 Getting Familiar With Rattle

The Rattle interface is based on a set of tabs through which we proceed, left to right. For any tab, once we have set up the required information, we **must** click the Execute button (or F2) to perform the actions. Take a moment to explore the interface a little by clicking through the various tabs. Notice the Help menu and find that the help layout mimics the tab layout.



To Quit from Rattle we simply click on the Quit button in the main Rattle window.

To Quit from RStudio we choose Quit from the File menu.

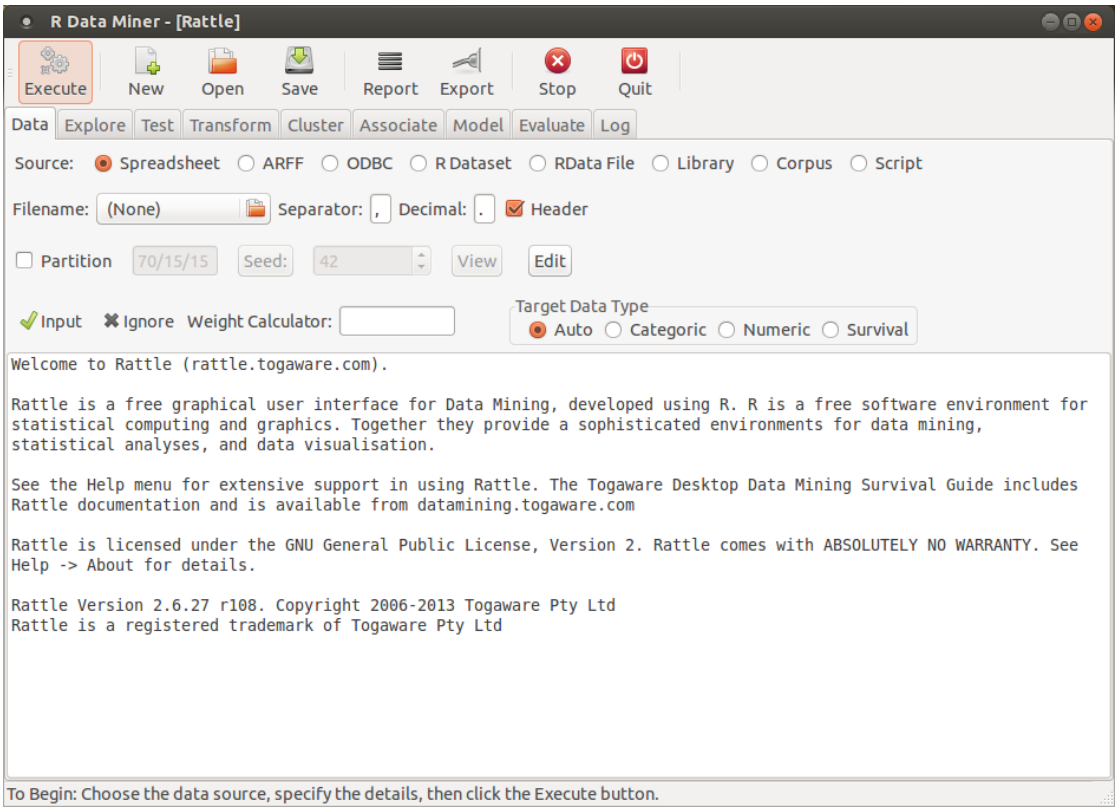
If we are using a terminal to run R then we can press 'Ctrl-D' (i.e. press the 'Control' key and then the 'D' key together).

In most cases we are asked whether to save our workspace. For now (and indeed for most users) we **do not** save the workspace.

3 The Initial Interface

The process that we implement in Rattle and that is reflected in the tabs that we see in the Rattle interface is:

1. Load a **Data**set;
2. Select **Variables** for exploring and mining;
3. **Sample** the data into training and test datasets;
4. **Explore** the distributions of the data;
5. Perhaps **Test** some of the distributions;
6. Optionally **Transform** our data;
7. Build **Cluster**s or **Association Rules** from the data;
8. Build predictive **Model**s;
9. **Evaluate** the models;
10. Record the steps in building your model as listed in the **Log**.

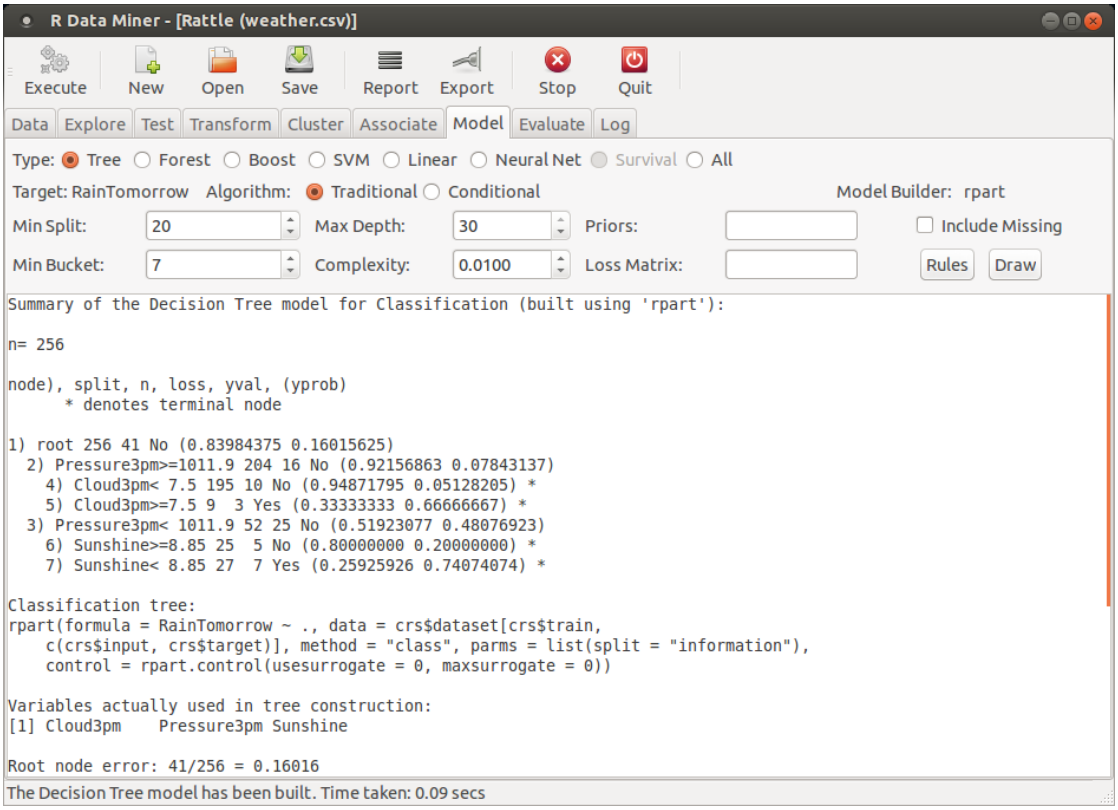


4 Load Data, Build Model

Our first familiarisation task is to load the sample weather dataset supplied with Rattle and build a simple model.

1. Start up Rattle.
2. Click the Execute button.
3. Answer Yes to load the example weather dataset.
4. Click the Model tab.
5. Click the Execute button.
6. Click the Draw button.

This is our very first model. It is a decision tree model and can be used to predict the probability that it will rain in Canberra (Australia) tomorrow, given today's conditions in Canberra.

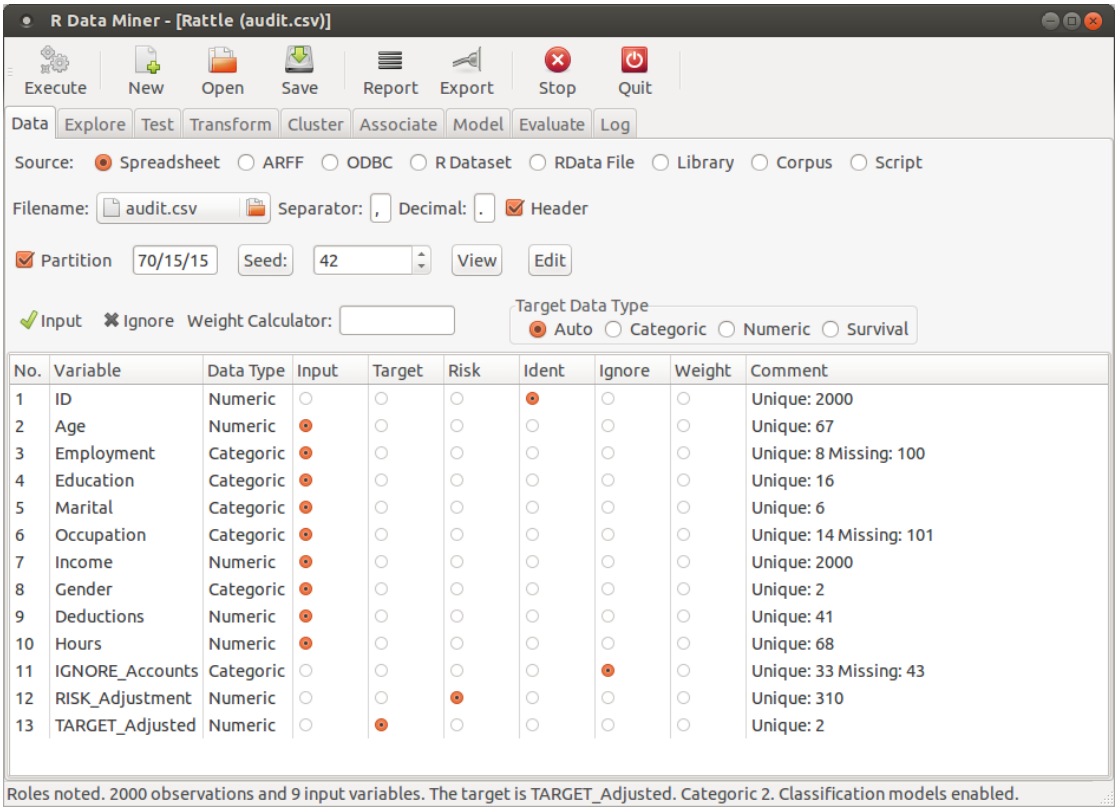


5 Audit: Load Dataset

We now switch to the sample Audit dataset provided with rattle (Williams, 2014).

1. Click the Data tab.
2. Click the Filename: button where weather.csv is currently listed.
3. Choose the audit.csv file to load
4. Load the file into Rattle.

Be sure to investigate what the audit dataset is about, and the meaning of each of the variables. You should document this.



6 Audit: Explore

Switching to the Explore tab investigate for any interesting patterns in the data. In particular, consider at least the following options.

1. Various summaries, noting any skewness or high values of kurtosis.
2. Anything interesting about missing values?
3. What does the cross tabulation suggest, if anything?
4. Various distribution plots including Benford's Law.
5. Any correlation between variables?

7 Audit: Test

The Test tab provides the opportunity to test out statistical hypotheses.

8 Audit: Transform

9 Audit: Cluster

10 Audit: Associate

11 Audit: Predictive Model

Exercise: Draw a tree and plot the evaluation.

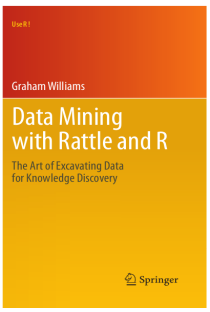
12 Audit: Evaluate

13 Audit: Review the Log

14 Assessment Activity

Now that you are familiar with interacting with a dataset in Rattle, load one of your own datasets, or else a public dataset from the Internet, and repeat the steps above using this dataset. Produce a report of your activities and discoveries. Submit the report as a PDF for assessment.

15 Further Reading



The Rattle Book, published by Springer, provides a comprehensive introduction to data mining and analytics using Rattle and R. It is available from Amazon. Other documentation on a broader selection of R topics of relevance to the data scientist is freely available from http://datamining.togaware.com, including the Datamining Desktop Survival Guide.

This module is one of many OnePageR modules available from http://onepager.togaware.com. In particular follow the links on the website with a * which indicates the generally more developed OnePageR modules.

Other resources include:

- http://rattle.togaware.com
- http://datamining.togaware.com
- http://datamining.togaware.com/survivor/index.html

16 References

R Core Team (2014). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.

Williams GJ (2009). "Rattle: A Data Mining GUI for R." *The R Journal*, **1**(2), 45–55. URL http://journal.r-project.org/archive/2009-2/RJournal_2009-2_Williams.pdf.

Williams GJ (2011). *Data Mining with Rattle and R: The art of excavating data for knowledge discovery*. Use R! Springer, New York. URL http://www.amazon.com/gp/product/1441998896/ref=as_li_qf_sp_asin_tl?ie=UTF8&tag=togaware-20&linkCode=as2&camp=217145&creative=399373&creativeASIN=1441998896.

Williams GJ (2014). *rattle: Graphical user interface for data mining in R*. R package version 3.0.4, URL http://rattle.togaware.com/.

This document, sourced from StartO.Rnw revision 419, was processed by KnitR version 1.6 of 2014-05-24 and took 1 seconds to process. It was generated by gjw on nyx running Ubuntu 14.04 LTS with Intel(R) Xeon(R) CPU W3520 @ 2.67GHz having 4 cores and 12.3GB of RAM. It completed the processing 2014-06-09 10:37:46.