Chapter 5

Rattle

Rattle (the R Analytical Tool To Learn Easily) is a graphical data mining application based on the statistical language R. (The R language is described in more detail in the following chapter, but an understanding of R is not required in order to use Rattle.) It is a low overhead, rapid development, data mining and modelling tool.

Rattle uses the Gnome graphical user interface and runs under GNU/Linux, Macintosh OS/X, and MS/Windows. Rattle provides an intuitive interface that takes you through the basic steps of data mining, as well as illustrating the R code that is used to achieve this.

Whilst the tool itself may be sufficient for all of a user's needs, it also provides a stepping stone to more sophisticated processing and modelling in R itself, for sophisticated and unconstrained data mining.

The user interface for Rattle is designed to flow through the data mining process. This is achieved through the use of a Tab interface, working from left to right: first load some Data, select Variables for exploring and mining, Sample the data into training and test datasets, Explore the data, identify Clusters in the data, build your Models and Evaluate them. We describe the data mining process through this paradigm in the following sections.

But first, we step through the simple process of installing Rattle.

5.1 Installation

Rattle has been packaged as an R package and is available from CRAN, the Comprehensive R Archive Network. The latest version is also available as an R package from Togaware. The source is available from Google Code.

The most raised issue is ensuring that you have the GTK+ libraries installed for your operating system. This is independent of R itself and is emphasised as a preliminary step:

1. *Install GTK+ libraries:* For GNU/Linux these are already installed if you are running GNOME. If you are not running gnome you may need to install the GTK+ libraries in your distribution.

   For **MS/Windows**, install the package from Glade for Windows:

   ```
   MS/Windows: run gtk-win32-devel-2.8.10-rc1.exe
   ```

If you are new to R, then to get up and running with Rattle there are a few steps:

1. *Install R:* R is included in many GNU/Linux distributions, such as Debian:

   ```
   Debian: $ wajig install r-recommended
   ```

   Versions for **MS/Windows** can be obtained from the R Project.

   ```
   MS/Windows: run R-2.3.1-win32.exe
   ```

   Versions for Mac/OSX are also available.

   To check if you have R installed, start up a Terminal and enter the command `R` (that's just the capital letter R). If the response is that the command is not found, then you probably need to install the R application.

   ```
       $ R
       R : Copyright 2006, The R Foundation for Statistical Computing
       Version 2.3.1 (2006-06-01)
       ISBN 3-900051-07-0

       R is free software and comes with ABSOLUTELY NO WARRANTY.
       You are welcome to redistribute it under certain conditions.
       Type 'license()' or 'licence()' for distribution details.

         Natural language support but running in an English locale

       R is a collaborative project with many contributors.
       Type 'contributors()' for more information and
       'citation()' on how to cite R or R packages in publications.
   ```

```
    Type 'demo()' for some demos, 'help()' for on-line help, or
    'help.start()' for an HTML browser interface to help.
    Type 'q()' to quit R.

    [Previously saved workspace restored]

    >
```

2. *Install RGtk2:* This package is available on CRAN and from the RGtk2 web site. From R, use:

```
R: > install.packages("RGtk2")
```

On Debian you can simply install the Deb package:

```
Debain: $ wajig install r-cran-gtk2
```

To test whether you have RGtk2 installed enter the R command

```
R: > library(RGtk2)
```

3. The following additional R packages are suggested if you want the full functionality of Rattle. Type `?install.packages` at the R prompt for further help on installing packages.

```
R: > install.packages(c("bitops", "cba", "combinat", "ellipse",
                        "fBasics", "fpc", "gbm", "gregmisc",
                         "kernlab", "maptree", "randomForest",
                         "RODBC", "ROCR", "rpart", "XML"))
```

4. *Install Rattle:* From within R, either install rattle directly from CRAN with:

```
R: > install.packages("rattle")
```

or else download the rattle package for GNU/Linux from `http://rattle.togaware.com/src/contrib/rattle_2.1.28.tar.gz` or for MS/Windows from `http://rattle.togaware.com/src/contrib/rattle_2.1.28.zip`. This might be done using the right mouse button menu on the above links, to Save link as.... Save the files to your local disk, and then install with, for example:

```
    > install.packages("rattle_2.1.28.zip", repos=NULL)
```

5. Now, after starting R ask it to load the rattle package into its library:

```
R: > library(rattle)
```

This loads the Rattle functionality (which is also available without running the Rattle GUI). To start the Rattle GUI simply run the command:

```
R: > rattle()
```

5.2 Introduction

We present the functionality of Rattle through the use of a simple data set, the *audit* data set, which is supplied as part of the Rattle package (it is also available for download as a CSV file from http://rattle.togaware.com/audit.csv). This is an artificial data set consisting of 2000 fictional clients who have been audited, perhaps for compliance with regard the amount of a tax refund that is being claimed. For each case an outcome is recorded (whether the taxpayer's claims had to be adjusted or not) and any amount of adjustment that resulted is also recorded.

The dataset is only 2,000 entities in order to ensure model building is relatively quick, for illustrative purposes. It contains 13 columns, with the first being a unique client identifier.

We proceed through the typical steps of a data mining project, beginning with a data load and selection, then an exploration of the data, and finally, modelling and evaluation.

The data mining process steps through each tab, left to right, performing the corresponding actions. For any tab, the modus operandi is to configure the options available and to then click the Execute button (or F5) to perform the appropriate tasks. It is important to note that the tasks are **not** performed until the Execute button (or F5 or the Execute menu item under Tools) is clicked.

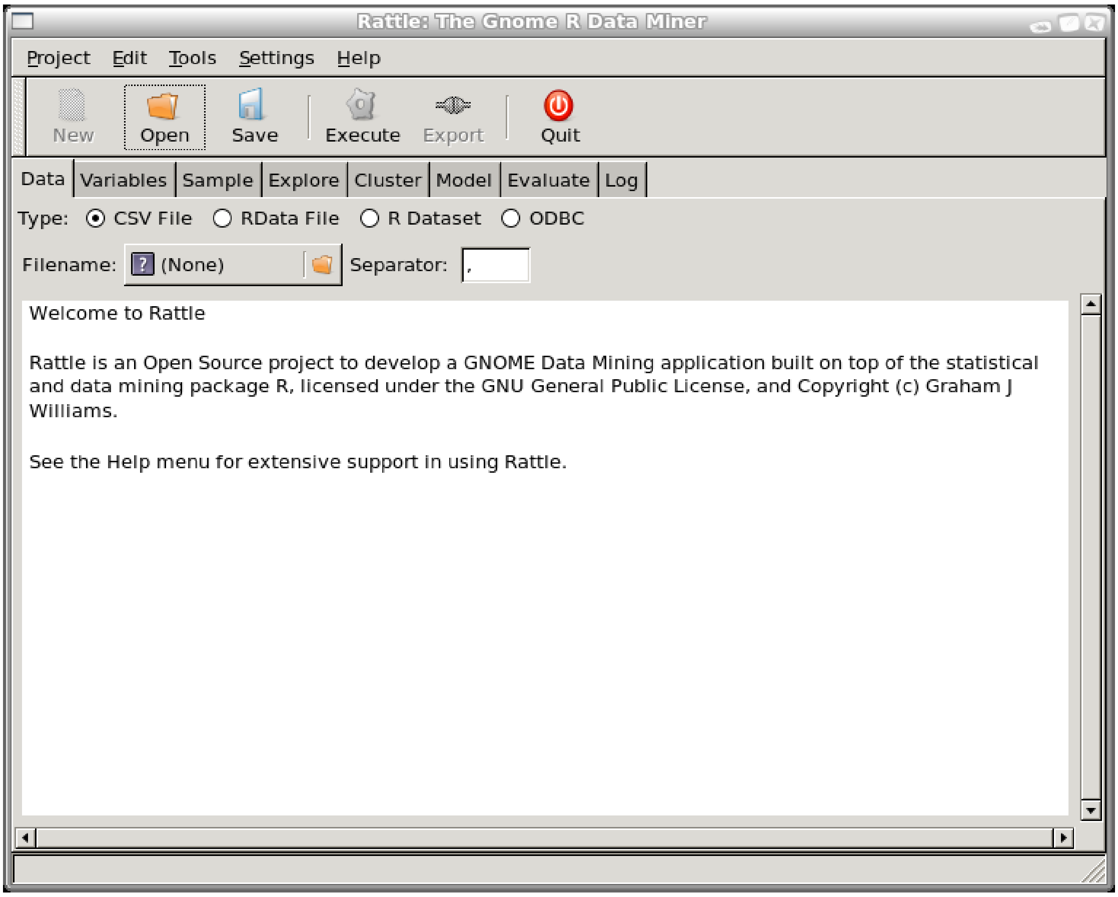
The Status Bar will indicate when the action is completed. Messages from R (e.g., error messages, although many R error messages are captured by Rattle and displayed in a popup) will appear in the R console from where Rattle was started.

The R Code that is executed underneath will appear in the Log tab. This allows for a review of the R commands that perform the corresponding data mining tasks. The R code snippets can be copied as text from the Log tab and pasted into the R Console from which Rattle is running, to be directly executed. This allows a user to deploy Rattle for basic tasks, yet allow the full power of R to be deployed as needed, perhaps through using more command options than exposed through the Rattle interface.

Rattle is being extensively tested on binary classification problems (with 0/1 or a two level variable as the outcomes for the Target variable). It is less well tested on general classification and regression tasks, but this will follow. Also in the pipeline is support for text mining.

5.3 Startup Rattle

```
$ R
> library(rattle)
> rattle()
```


Rattle: The Gnome R Data Miner
Project Edit Tools Settings Help
New
Open
Save
Execute
Export
Quit
Data
Variables
Sample
Explore
Cluster
Model
Evaluate
Log
Type: ● CSV File ○ RData File ○ R Dataset ○ ODBC
Filename: (None)
Separator: ,

Welcome to Rattle

Rattle is an Open Source project to develop a GNOME Data Mining application built on top of the statistical and data mining package R, licensed under the GNU General Public License, and Copyright (c) Graham J Williams.

See the Help menu for extensive support in using Rattle.

The main Rattle window will pop up. You will see a welcome message and a hint about using Rattle. Essentially, you will proceed through the tabs in this interface from left to right. Once you have set up the required information on any one of the tabs, you need to click the execute button to perform the actions. Take a moment to explore the interface a little. Notice the Help menu and find that the help layout mimics the tab layout.

5.4 Menus and Buttons

5.4.1 Projects

A project is a packaging of a dataset, variable selections, explorations, clusters and models built from the data. Rattle allows projects to be saved for later resumption of the work or for sharing the data mining project with other users.

A project is typically saved to a file with the `.rattle` extension (although in reality it is just a standard `.Rdata` file.

At a later time you can load a project into rattle to restore the data, models, and other displayed information relating to the project, and resume your data mining from that point. You can also share these project files with other Rattle users, which is quite useful for data mining teams.

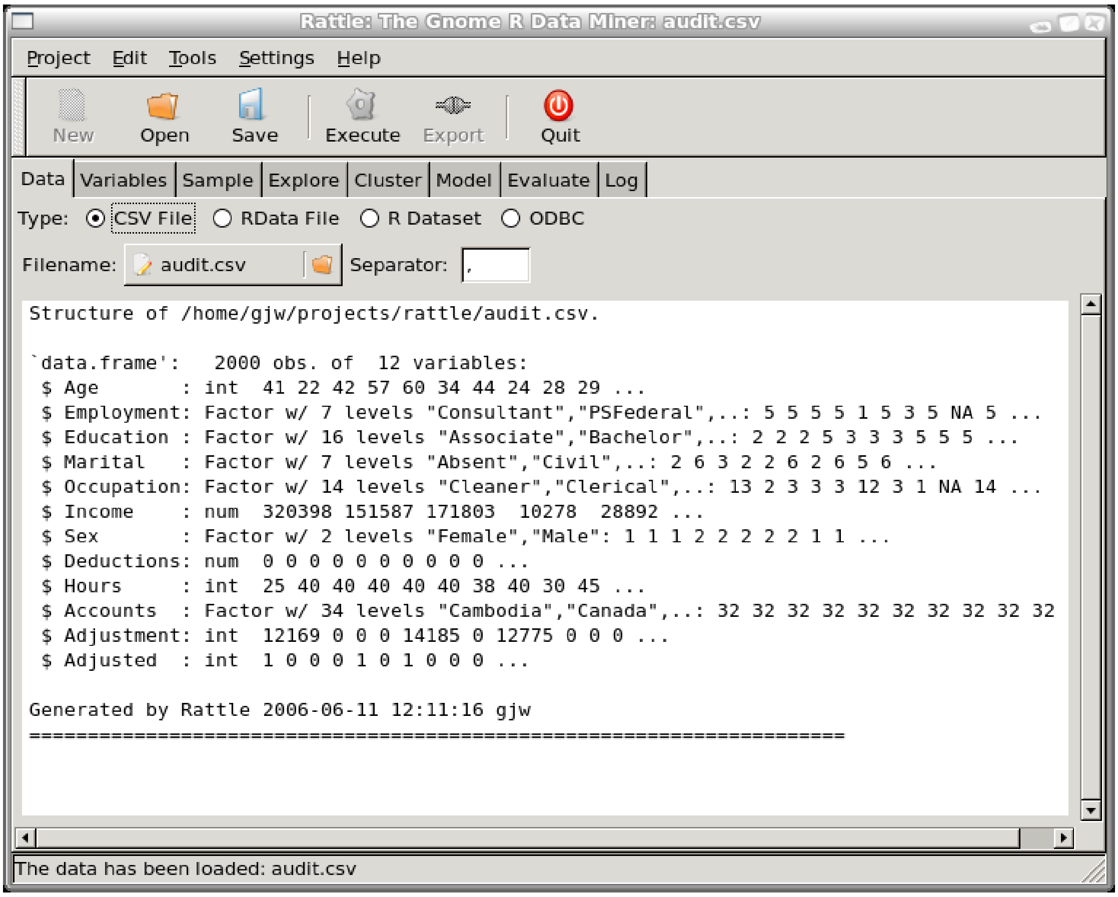
You can rename the files, keeping the `.rattle` extension, without impacting the project file itself — that is, the file name has no formal bearing on the contents, so use it to be descriptive — but best to avoid vacant spaces and unusual characters!

5.5 Data Tab

5.5.1 CSV File Option

Rattle can load data from a comma separated value (CSV) file, as might be generated by spreadsheets and databases, including Excel, Gnumeric, SAS/EM, QueryMan, and many other applications. This is a good option for importing your data into Rattle.

From the Data tab click the Filename button and choose `audit.csv`. Now click the Execute button to load the data set. This will load the data set from the `audit.csv` file. The contents of the window changes to give a brief summary of the data set. Notice that we have loaded 2,000 entities, each described by 12 variables. The data type and the first few values for each entity are also displayed. We can start getting an idea of the shape of the data, noting that Adjusted, for example looks like it might be a categorical variable!


Rattle: The Gnome R Data Miner: audit.csv
Project Edit Tools Settings Help
New Open Save Execute Export Quit
Data Variables Sample Explore Cluster Model Evaluate Log

Type: ● CSV File ○ RData File ○ R Dataset ○ ODBC

Filename: audit.csv Separator: ,

```
Structure of /home/gjw/projects/rattle/audit.csv.

`data.frame':   2000 obs. of  12 variables:
 $ Age       : int  41 22 42 57 60 34 44 24 28 29 ...
 $ Employment: Factor w/ 7 levels "Consultant","PSFederal",..: 5 5 5 5 1 5 3 5 NA 5 ...
 $ Education : Factor w/ 16 levels "Associate","Bachelor",..: 2 2 2 5 3 3 3 5 5 5 ...
 $ Marital   : Factor w/ 7 levels "Absent","Civil",..: 2 6 3 2 2 6 2 6 5 6 ...
 $ Occupation: Factor w/ 14 levels "Cleaner","Clerical",..: 13 2 3 3 3 12 3 1 NA 14 ...
 $ Income    : num  320398 151587 171803  10278  28892 ...
 $ Sex       : Factor w/ 2 levels "Female","Male": 1 1 1 2 2 2 2 2 1 1 ...
 $ Deductions: num  0 0 0 0 0 0 0 0 0 0 ...
 $ Hours     : int  25 40 40 40 40 40 38 40 30 45 ...
 $ Accounts  : Factor w/ 34 levels "Cambodia","Canada",..: 32 32 32 32 32 32 32 32 32 32
 $ Adjustment: int  12169 0 0 0 14185 0 12775 0 0 0 ...
 $ Adjusted  : int  1 0 0 0 1 0 1 0 0 0 ...

Generated by Rattle 2006-06-11 12:11:16 gjw
======================================================================
```

The data has been loaded: audit.csv

The CSV file is assumed to begin with a header row, listing the names of the variables. The remainder of the file is expected to consist of rows of data that record information about the entities, with fields generally separated by commas recording the values of the variables for this entity.

You can choose the field delimiter through the Separator entry. A comma is the default. To load a `.txt` file which uses a tab as the field separator enter `\\t` as the separator. You can also leave the separator empty and any white space will be used as the separator.

Any data with missing values, or having the value "NA" or else ".", is treated as a missing value, which is represented in R as the string `NA`. Support for the "." convention allows the importation of CSV data generated by SAS.

Underneath, the corresponding R code uses the *read.csv* function to load the data.

5.5.2 ODBC Option

5.5.3 RData File Option

Using the *RData File* option data can be loaded directly from a native R data file (usually with the `.Rdata` or `.RData` extension. Such files may contain multiple datasets (compressed) and you will be given an option to choose just one of the available data sets.

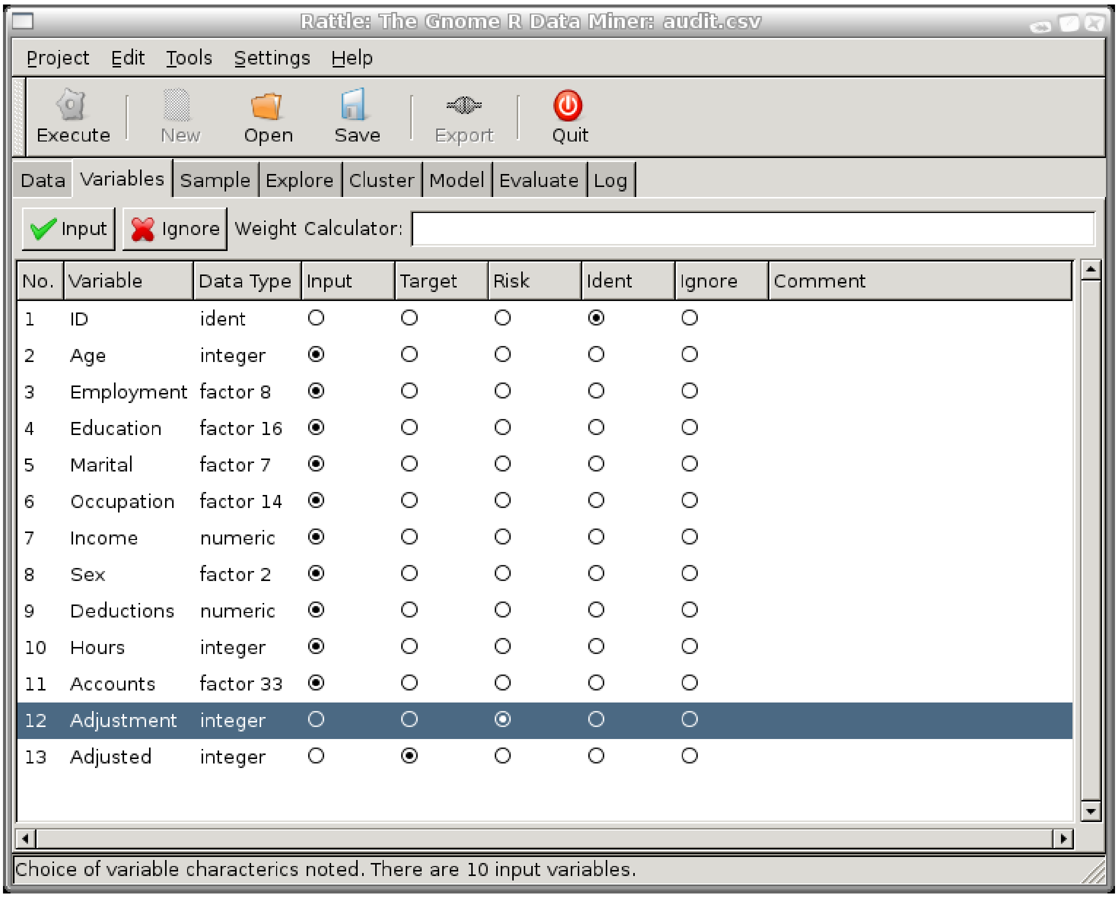
5.5.4 R Dataset Option

Rattle can use a dataset that is already loaded into R (although it will take a copy of it, with memory implications). Only data frames are currently supported, and Rattle will list for you the names of all of the available data frames.

The data frames need to be constructed in the same R session that is running Rattle (i.e., the same R Console in which you lo the Rattle package). This provides much more flexibility in loading data into Rattle, than is provided directly through the actual Rattle interface. For example, you may want to load data from an SQLite database directly, and have this available in Rattle.

5.6 Variables Tab

5.6.1 Roles



No.	Variable	Data Type	Input	Target	Risk	Ident	Ignore	Comment
1	ID	ident	○	○	○	●	○	
2	Age	integer	●	○	○	○	○	
3	Employment	factor 8	●	○	○	○	○	
4	Education	factor 16	●	○	○	○	○	
5	Marital	factor 7	●	○	○	○	○	
6	Occupation	factor 14	●	○	○	○	○	
7	Income	numeric	●	○	○	○	○	
8	Sex	factor 2	●	○	○	○	○	
9	Deductions	numeric	●	○	○	○	○	
10	Hours	integer	●	○	○	○	○	
11	Accounts	factor 33	●	○	○	○	○	
12	Adjustment	integer	○	○	●	○	○	
13	Adjusted	integer	○	●	○	○	○	

Choice of variable characterics noted. There are 10 input variables.

The Variables tab is used to identify the role played by each of the variables in the data set. The default role for most variables is that of an Input variable. generally, these are the variables that will be used to predict the value of a Target variable.

Rattle uses simple heuristics to guess at a Target role for one of the variables. Here we see that Adjusted has been selected as the target variable. In this instance it is correct. The heuristic involves examining the number of distinct values that a variable has, and if it has less than 5, then it is considered as a candidate. The candidate list is ordered starting with the last variable (often the last variable is the target), and then proceeding from the first onwards to find the first variable that meets the conditions of looking like a target.

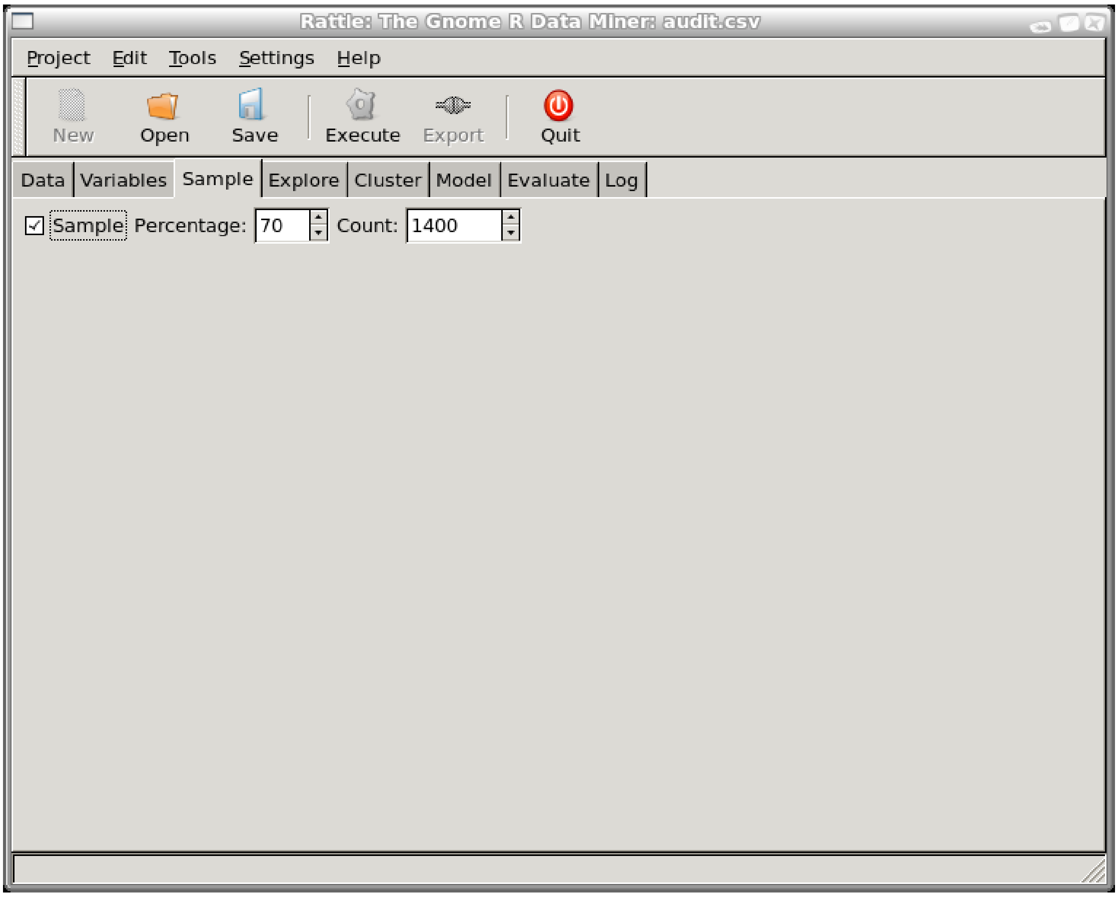
Any numeric variables that have a unique value for each record is automatically identified as an Ident. Any number of variables can be tagged as being an Ident. All Ident variables are ignored when modelling, but are used after scoring a data set, being written to the resulting score file so that the cases that are scored can be identified.

Sometimes not all variables in your data set should be used or may not be appropriate for a particular modelling task. For example, the random forest model builder does not handle categorical variables with more than 32 levels, so you may choose to Ignore Accounts. You can change the role of any variable to suit your needs, although you can only have one Target and one Risk.

For any changes you make to the Variables tab to take effect click the Execute button.

For an example of the use of the Risk variable, see Section 5.11.2.

5.7 Sample Tab

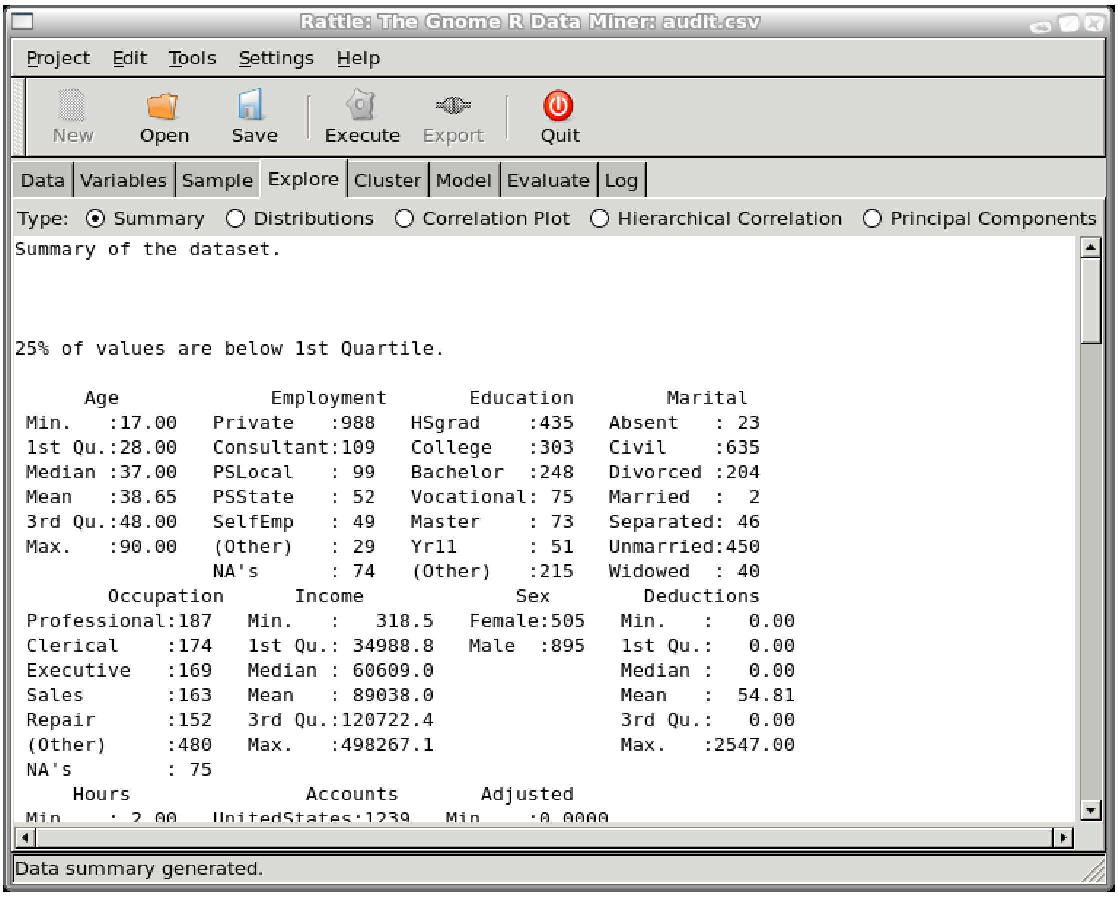

Rattle: The Gnome R Data Miner: audit.csv
Project Edit Tools Settings Help
New Open Save Execute Export Quit
Data Variables Sample Explore Cluster Model Evaluate Log

☑ Sample Percentage: 70 Count: 1400

Here we specify how we might partition the dataset for exploratory and modelling purposes. The default for Rattle is to build two subsets of the dataset: one is a training set from which to build models, while the other is used for testing the performance of the model. The default for Rattle is to use a 70% training and a 30% testing split, but you are welcome to turn sampling off, or choose other samplings. A very small sampling may be required to perform some explorations of the smaller dataset, or to build models using the more computationally expensive algorithms (like support vector machines).

5.8 Explore Tab

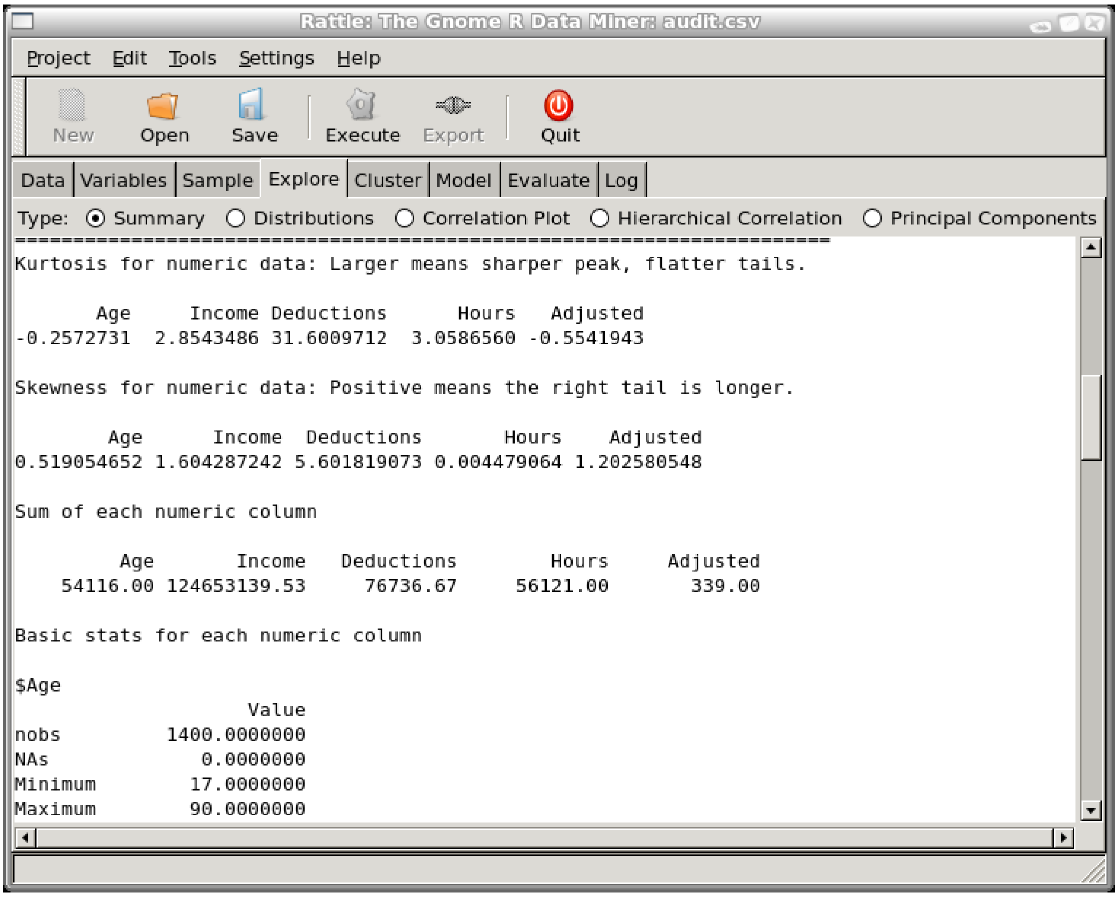
5.8.1 Summary Option


Rattle: The Gnome R Data Miner: audit.csv
Project Edit Tools Settings Help
New Open Save Execute Export Quit
Data Variables Sample Explore Cluster Model Evaluate Log
Type: ● Summary ○ Distributions ○ Correlation Plot ○ Hierarchical Correlation ○ Principal Components

```
Summary of the dataset.



25% of values are below 1st Quartile.

      Age                Employment         Education          Marital
 Min.   :17.00   Private     :988   HSgrad    :435   Absent   : 23
 1st Qu.:28.00   Consultant:109   College   :303   Civil    :635
 Median :37.00   PSLocal     : 99   Bachelor  :248   Divorced :204
 Mean   :38.65   PSState     : 52   Vocational: 75   Married  :  2
 3rd Qu.:48.00   SelfEmp     : 49   Master    : 73   Separated: 46
 Max.   :90.00   (Other)     : 29   Yr11      : 51   Unmarried:450
                 NA's        : 74   (Other)   :215   Widowed  : 40
        Occupation       Income              Sex          Deductions
 Professional:187   Min.   :   318.5   Female:505   Min.   :   0.00
 Clerical    :174   1st Qu.: 34988.8   Male  :895   1st Qu.:   0.00
 Executive   :169   Median : 60609.0                Median :   0.00
 Sales       :163   Mean   : 89038.0                Mean   :  54.81
 Repair      :152   3rd Qu.:120722.4                3rd Qu.:   0.00
 (Other)     :480   Max.   :498267.1                Max.   :2547.00
 NA's        : 75
     Hours              Accounts          Adjusted
 Min.   : 2.00   UnitedStates:1239   Min.   :0.0000
```

Data summary generated.

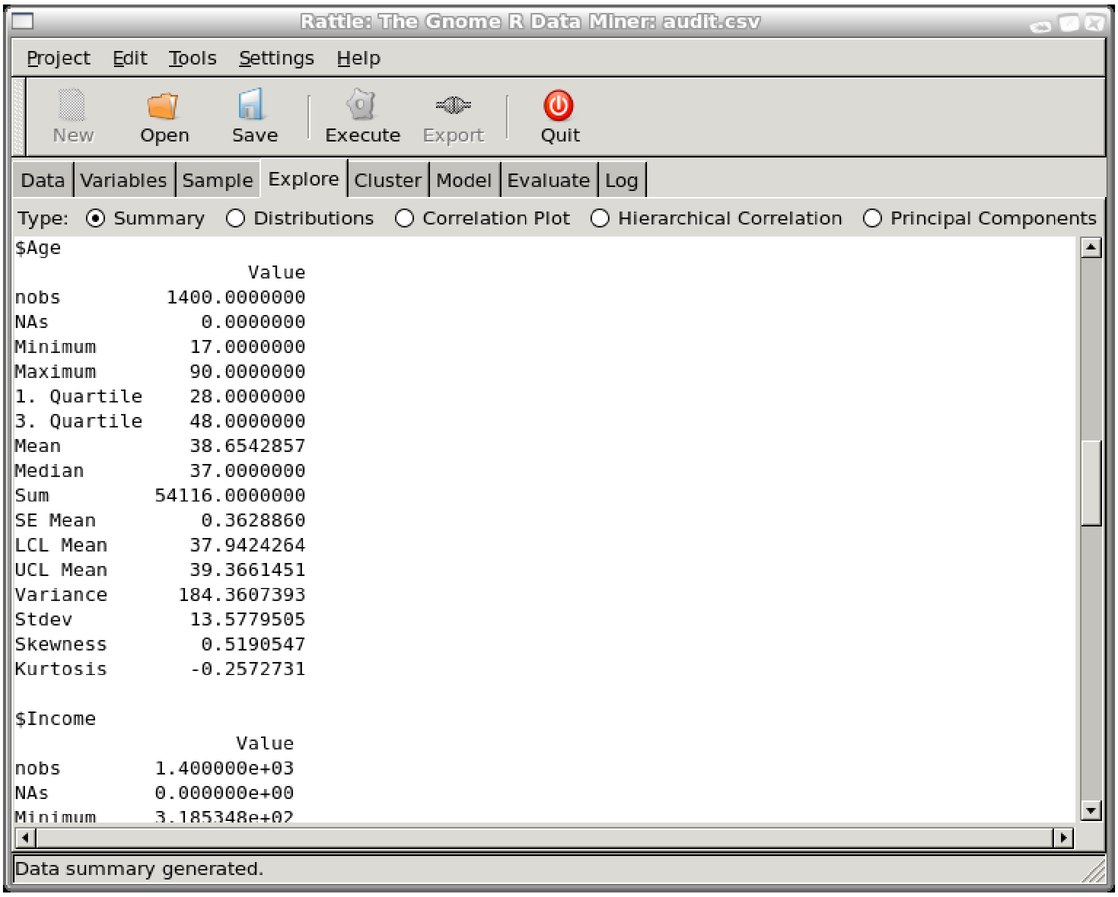
The Explore tab provides an opportunity to understand the data in various ways. The Summary option provides numerous measures for each variable, including, the in the first instance, the minimum, maximum, median, mean, and the first and third quartiles. Generally, if the mean and median are significantly different then we would think that there are some entities with very large values in the data pulling the mean in one direction. It does not seem to be the case for Age but is for Income.


Rattle: The Gnome R Data Miner: audit.csv
Project Edit Tools Settings Help
New Open Save Execute Export Quit
Data Variables Sample Explore Cluster Model Evaluate Log

Type: ● Summary ○ Distributions ○ Correlation Plot ○ Hierarchical Correlation ○ Principal Components

```
======================================================================
Kurtosis for numeric data: Larger means sharper peak, flatter tails.

       Age     Income Deductions      Hours   Adjusted
-0.2572731  2.8543486 31.6009712  3.0586560 -0.5541943

Skewness for numeric data: Positive means the right tail is longer.

        Age      Income  Deductions       Hours    Adjusted
0.519054652 1.604287242 5.601819073 0.004479064 1.202580548

Sum of each numeric column

         Age        Income   Deductions        Hours     Adjusted
    54116.00 124653139.53     76736.67     56121.00       339.00

Basic stats for each numeric column

$Age
                    Value
nobs         1400.0000000
NAs             0.0000000
Minimum        17.0000000
Maximum        90.0000000
```

Scroll the Explore tab window to see a summary of some particular measures of the spread of the distribution of numeric data. Kurtosis is a measure of the nature of the peaks in the distribution of the data. A high kurtosis indicates a sharper peak and fatter tails while a lower kurtosis indicates a more rounded peak with wider shoulders. Skewness indicates the assymetry of the distribution. A positive skew indicates that the tail to the right is longer, and a negative skew that the tail to the left is longer.


Rattle: The Gnome R Data Miner: audit.csv
Project Edit Tools Settings Help
New Open Save Execute Export Quit
Data Variables Sample Explore Cluster Model Evaluate Log

Type: ● Summary ○ Distributions ○ Correlation Plot ○ Hierarchical Correlation ○ Principal Components

```
$Age
                    Value
nobs         1400.0000000
NAs             0.0000000
Minimum        17.0000000
Maximum        90.0000000
1. Quartile    28.0000000
3. Quartile    48.0000000
Mean           38.6542857
Median         37.0000000
Sum         54116.0000000
SE Mean         0.3628860
LCL Mean       37.9424264
UCL Mean       39.3661451
Variance      184.3607393
Stdev          13.5779505
Skewness        0.5190547
Kurtosis       -0.2572731

$Income
                   Value
nobs        1.400000e+03
NAs         0.000000e+00
Minimum     3.185348e+02
```

Data summary generated.

Scroll the Explore tab window even further to see a more detailed summary of each variable. Here we see a summary of Age with some of the same data as before, but more! If there were any missing values they would be counted here under the NAs. The sum of values is included, as well as the standard deviation.

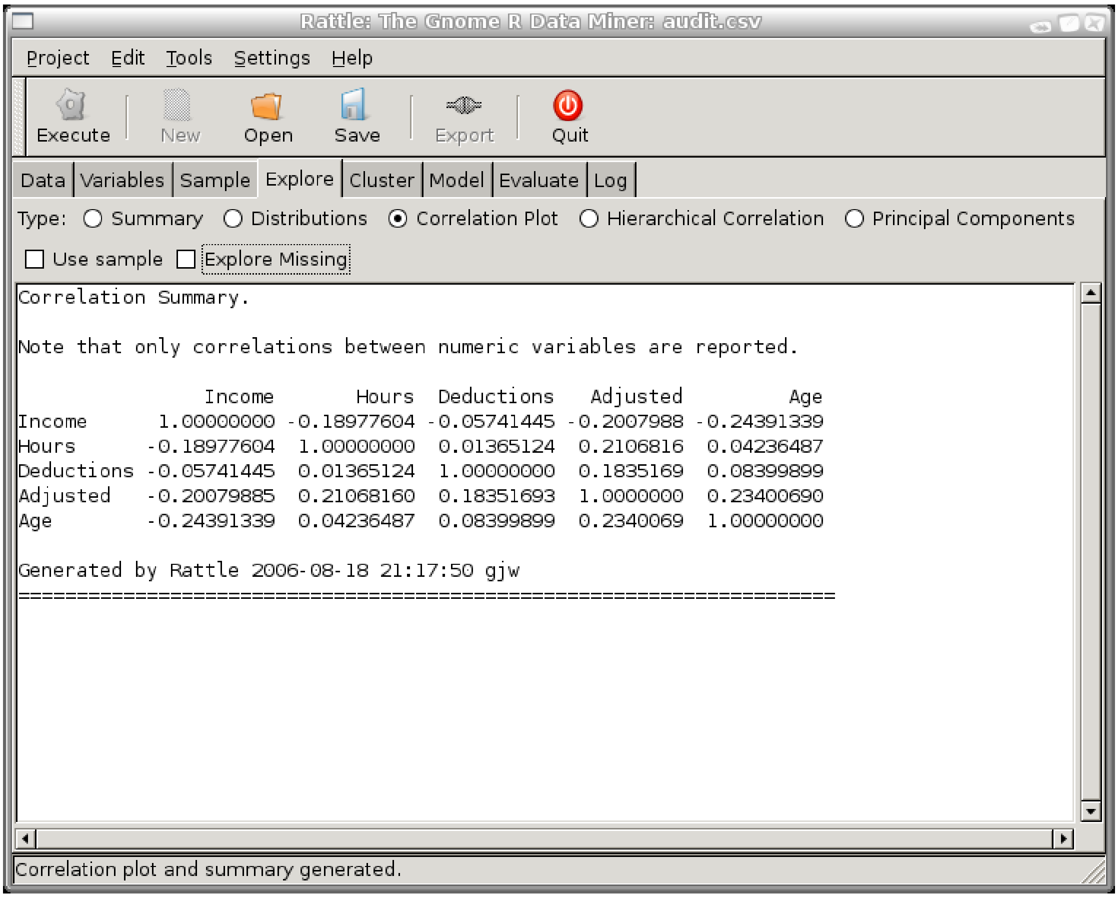
5.8.2 Distributions Option

It is usually a good idea to review the distributions of the values of each of the variables in your dataset. The Distributions option allows you to visually explore the distributions for specific variables.

Graphical presentations are more effective for most people, and Rattle provides a graphical summary of the distribution of the data with the Distribution option of the Explore tab.

The default is to not show the distribution for any variables. The intention is that you should select collections of specific variables. Selecting many will lead to many plots being displayed.

5.8.3 Correlation Plot Option


Rattle: The Gnome R Data Miner: audit.csv
Project Edit Tools Settings Help
Execute
New
Open
Save
Export
Quit
Data Variables Sample Explore Cluster Model Evaluate Log

Type: ○ Summary ○ Distributions ● Correlation Plot ○ Hierarchical Correlation ○ Principal Components

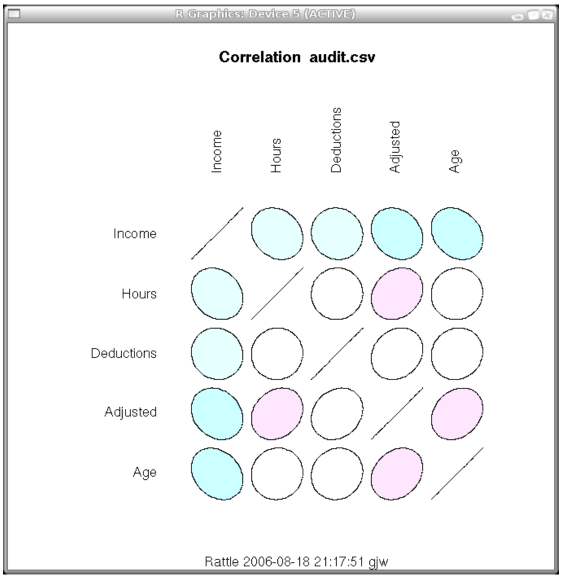
□ Use sample □ Explore Missing

```
Correlation Summary.

Note that only correlations between numeric variables are reported.

                Income       Hours  Deductions   Adjusted         Age
Income      1.00000000 -0.18977604 -0.05741445 -0.2007988 -0.24391339
Hours      -0.18977604  1.00000000  0.01365124  0.2106816  0.04236487
Deductions -0.05741445  0.01365124  1.00000000  0.1835169  0.08399899
Adjusted   -0.20079885  0.21068160  0.18351693  1.0000000  0.23400690
Age        -0.24391339  0.04236487  0.08399899  0.2340069  1.00000000

Generated by Rattle 2006-08-18 21:17:50 gjw
======================================================================
```

Correlation plot and summary generated.

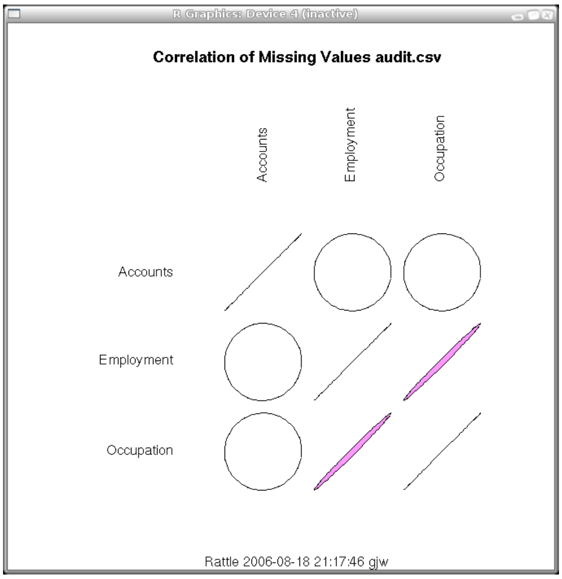
A correlation plot will display correlations between the values of variables in the data set. In addition to the usual correlation calculated between values of different variables, the correlation between missing values can be explored by checking the Explore Missing check box.

Rattle uses the default R correlation calculation known as Pearson's correlation, a common measure of correlation.



The first thing to notice for this correlation plot is that only the numeric variables appear. Rattle only computes correlations between numeric variables at this time. The second thing to note about the graphic is that it is symmetric about the diagonal. The correlation between two variables is the same, irrespective of the order in which we view the two variables. The third thing to note is that the order of the variables does not correspond to the order in the data set, but to the order of the strength of any correlations, from the least to the greatest. This is done simply to achieve a more pleasing graphic which is easier to take in.

We interpret the degree of any correlation by both the shape and colour of the graphic elements. Any variable is, of course, perfectly correlated with itself, and this is reflected as the diagonal lies on the diagonal of the graphic. Where the graphic element is a perfect circle, then there is no correlation between the variables, as is the case in the correlation between Hours and Deductions—although in fact there is a correlation, just a very weak one. XXXX Explain Colour Intensity XXXX



By selecting the Explore Missing check box you can obtain a correlation plot that will show any correlations between the missing values of variables. This is particularly useful to understand how missing values in one variable are related to missing values in another.

We notice immediately that only three variables are included in this correlation plot. Rattle has identified that the other variables in fact have no missing values, and so there is no point including them in the plot. We also notice that a categorical variable, Accounts, is included in the plot even though it was not included in the usual correlation plot. In this case we can obtain a correlation for categorical variables since we only measure missing and presence of a value, which is easily interpreted as numeric.

The graphic shows us that Employment and Occupation are highly correlated in their presence of missing values. That is, when Employment has a missing value, so does Occupation, and vice versa, at least in general. The actual correlation is 0.995 (which can be read from the Rattle text view window), which is very close to 1.

On the other hand, there is no (in fact very little at 0.013) correlation between Accounts and the other two variables, with regard missing values.

5.8.4 Hierarchical Correlation Option

5.8.5 Principal Components Option

5.9 Cluster Tab

5.9.1 K Means Option

5.9.2 Hierarchical Option

5.10 Model Tab

5.10.1 Decision Tree Option

5.10.2 Random Forest Option

5.10.3 SVM Option

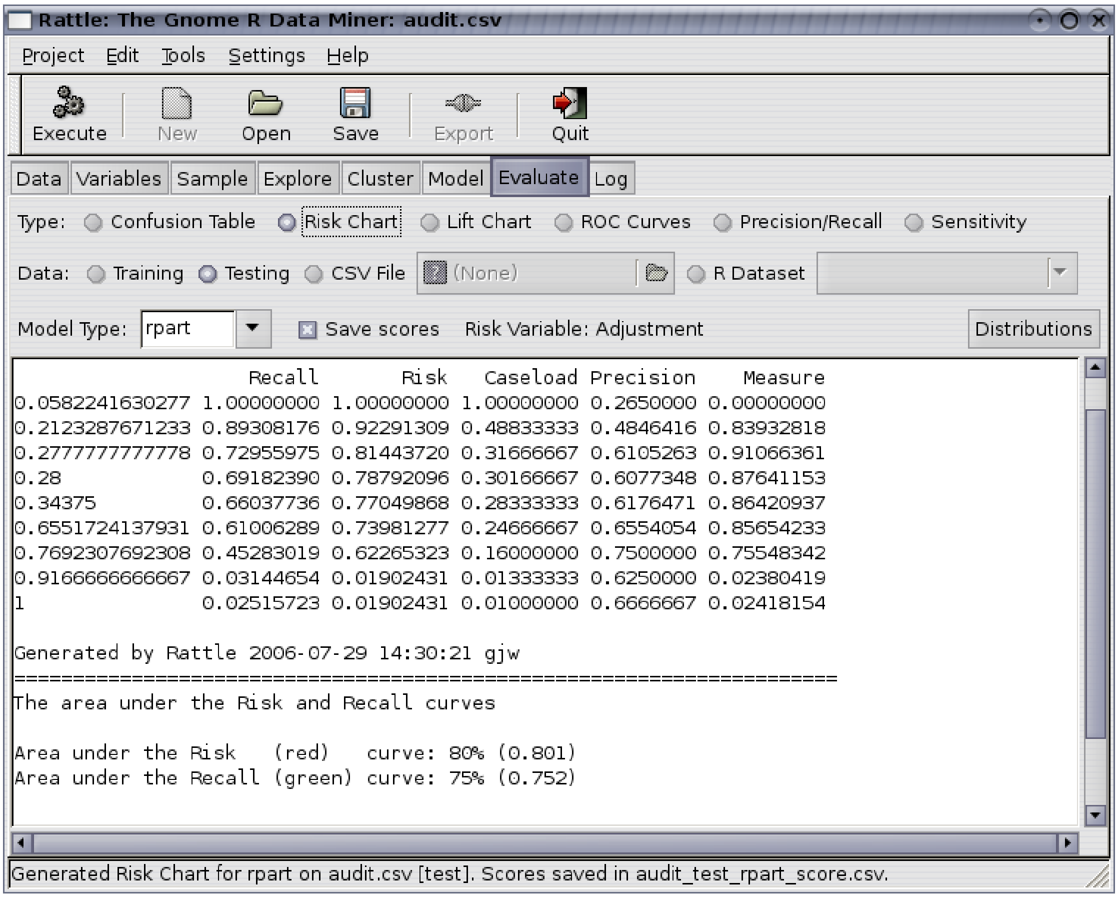
5.10.4 Regression Option

5.10.5 Boosting Option

5.11 Evaluate Tab

5.11.1 Confusion Option

5.11.2 Risk Option


Rattle: The Gnome R Data Miner: audit.csv
Project Edit Tools Settings Help
Execute
New
Open
Save
Export
Quit
Data Variables Sample Explore Cluster Model Evaluate Log
Distributions

Type: ○ Confusion Table ● Risk Chart ○ Lift Chart ○ ROC Curves ○ Precision/Recall ○ Sensitivity

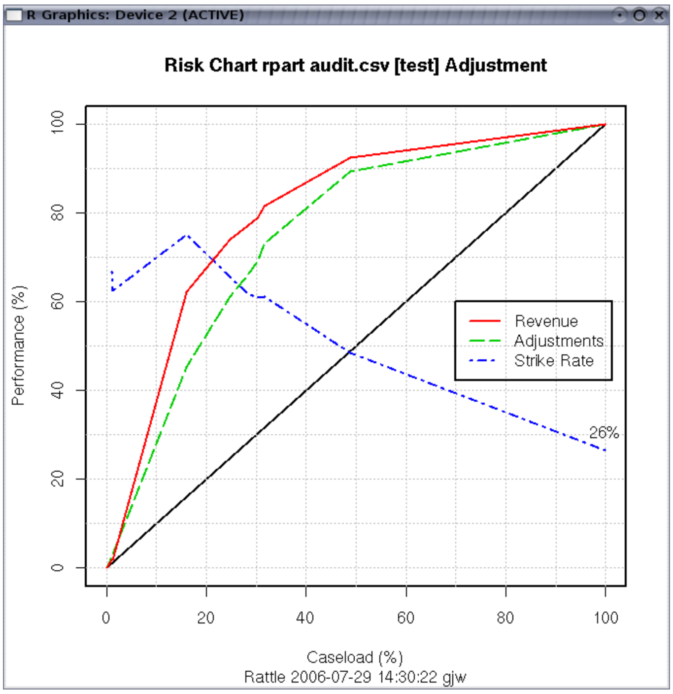
Data: ○ Training ● Testing ○ CSV File (None) ○ R Dataset

Model Type: rpart ☒ Save scores Risk Variable: Adjustment

```
                    Recall       Risk   Caseload Precision    Measure
0.0582241630277 1.00000000 1.00000000 1.00000000 0.2650000 0.00000000
0.212328767123 0.89308176 0.92291309 0.48833333 0.4846416 0.83932818
0.277777777778 0.72955975 0.81443720 0.31666667 0.6105263 0.91066361
0.28           0.69182390 0.78792096 0.30166667 0.6077348 0.87641153
0.34375        0.66037736 0.77049868 0.28333333 0.6176471 0.86420937
0.6551724137931 0.61006289 0.73981277 0.24666667 0.6554054 0.85654233
0.7692307692308 0.45283019 0.62265323 0.16000000 0.7500000 0.75548342
0.916666666667 0.03144654 0.01902431 0.01333333 0.6250000 0.02380419
1              0.02515723 0.01902431 0.01000000 0.6666667 0.02418154

Generated by Rattle 2006-07-29 14:30:21 gjw
======================================================================
The area under the Risk and Recall curves

Area under the Risk   (red)   curve: 80% (0.801)
Area under the Recall (green) curve: 75% (0.752)
```

Generated Risk Chart for rpart on audit.csv [test]. Scores saved in audit_test_rpart_score.csv.

We have introduced the idea of a Risk Chart to communicate the effectiveness of a model when each entity has associated with it a risk value. For example, for revenue (or tax) authorities, the outcomes of audits include a dollar amount by which the tax obligation of the taxpayer has been changed (which may be a change in favour of the revenue authority or in favour of the taxpayer). For fraud investigations, the outcome might be the dollar amount recovered from the fraud. It is often useful to see the tradeoff between the return on investment and the number of cases investigated.



When a Risk Chart is generated the text window in Rattle will display the aggregated data that is used to construct the plot. This data consists of a row for each level of the probability distribution that is output from the model, ordered from the lowest probability value to a value of 1. For each row we record the model performance in terms of predicting a class of 1 if the probability cutoff was set to the corresponding value.

For example, we might choose a cutoff to be a probability of 0.28 so that anything predicted to be in class 1 with a probability of 0.28 or more will be regarded as in class 1. Then the number of predicted positives (or the Caseload) will be 30% (0.301667) of all cases. Amongst this 30% of cases are 69% of all true positives and they account for 79% of the total of the risk scores. The strike rate (number of true positives amongst the positives predicted by the model) is 61%. Finally, the measure reports the sum of the distances of the risk and recall from the baseline (the diagonal line). This measure can indicate the optimal caseload in terms of maximising both risk recovery and recall.

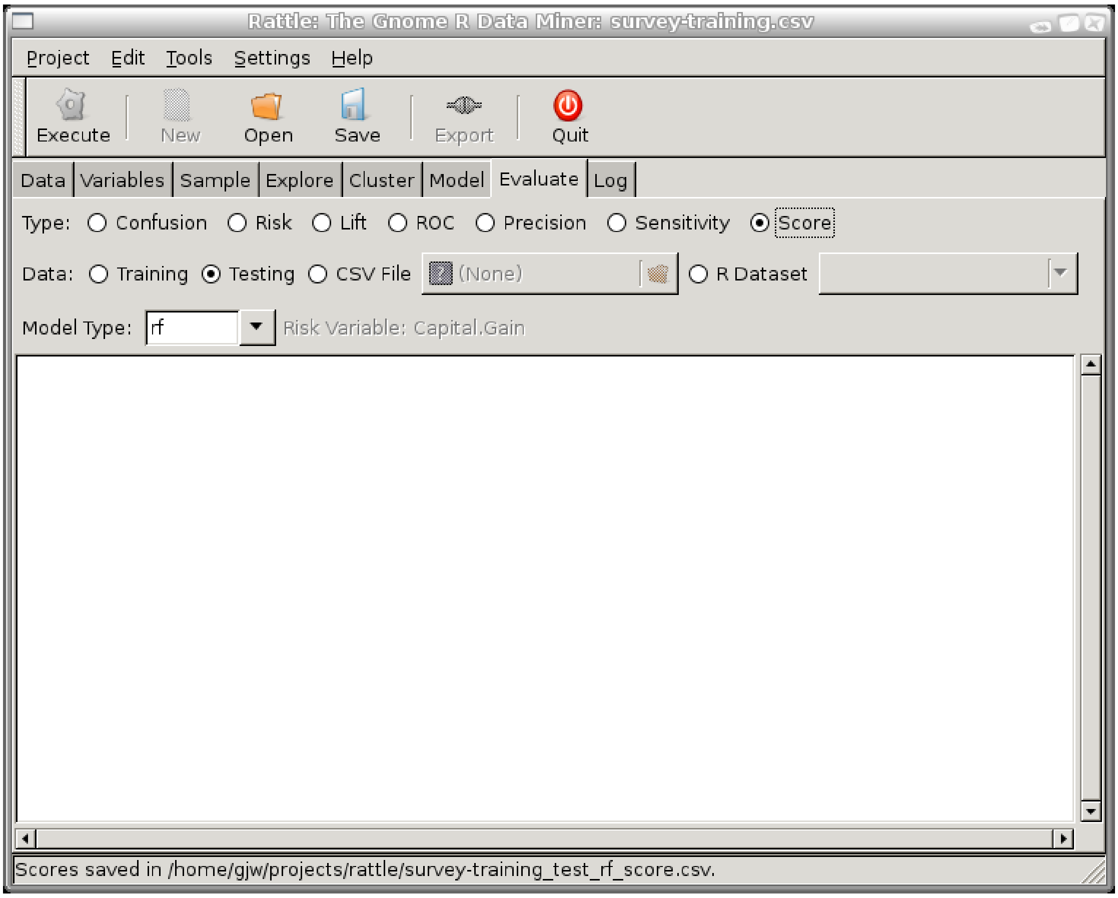
5.11.3 Lift Option

5.11.4 ROC Option

5.11.5 Precision Option

5.11.6 Sensitivity Option

5.11.7 Score Option


Rattle: The Gnome R Data Miner: survey-training.csv
Project Edit Tools Settings Help
Execute New Open Save Export Quit
Data Variables Sample Explore Cluster Model Evaluate Log

Type: ○ Confusion ○ Risk ○ Lift ○ ROC ○ Precision ○ Sensitivity ● Score

Data: ○ Training ● Testing ○ CSV File (None) ○ R Dataset

Model Type: rf

Risk Variable: Capital.Gain

Scores saved in /home/gjw/projects/rattle/survey-training_test_rf_score.csv.

Often you will want to apply a model to a dataaset to generate scores for use in other tools. The Score radio button allows you to score (i.e., to generate probabilities for each entry in) a dataset. Rattle will generate a CSV file containing these "scores," and we refer to this process as scoring a dataset. Each line of the CSV file will consist of a comma separated list of all of the variables that have been identified as *Ident*s in the Variables tab, followed by the score.

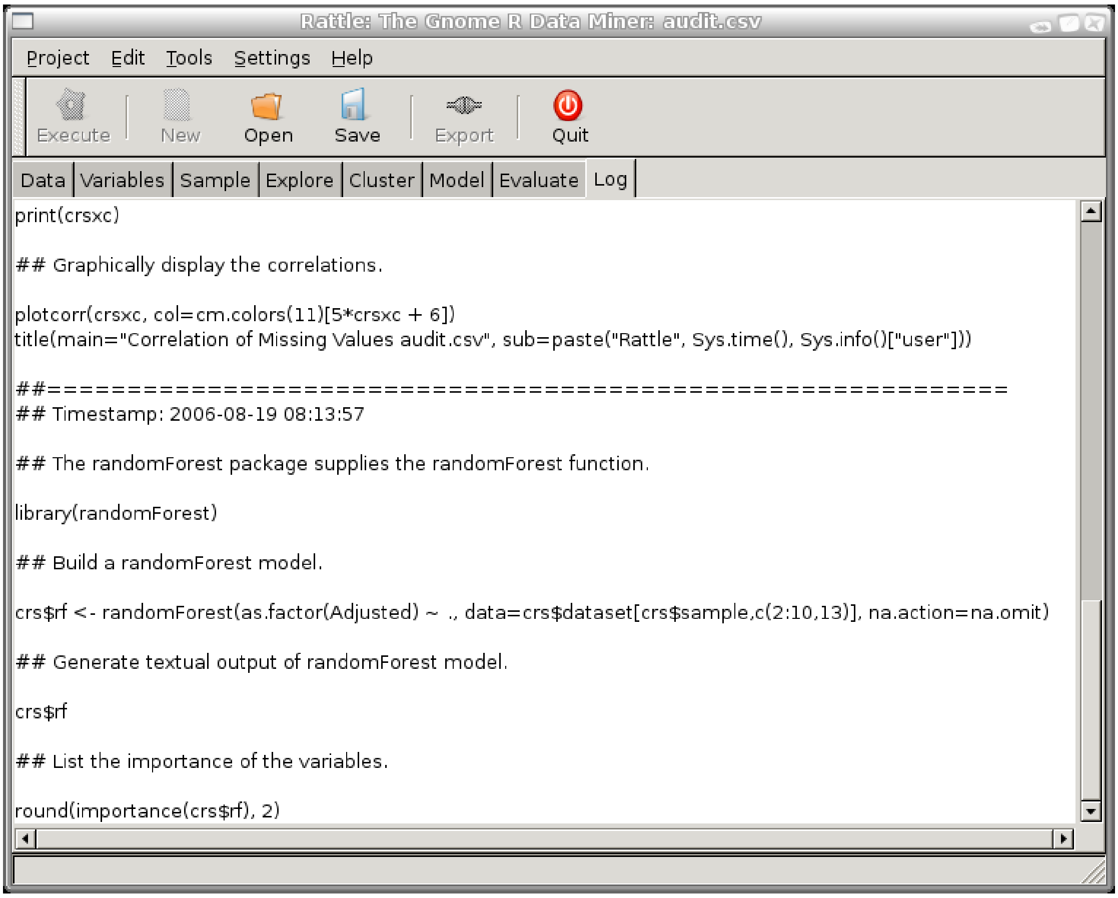
Note the status bar in the sample screenshot has identified that the score file has been saved to the file. The file name is derived from name of the dataset (perhaps a source data csv filename of the name of an R data frame), whether it is a test or training dataset, the type of model and the type of score.

The output looks like:

```
ID,predict
98953270,0.104
12161980,NA
```

```
96316627,0.014
54464140,0.346
57742269,0.648
19307037,0.07
61179245,0.004
36044473,0.338
19156946,0.33
```

5.12 Log Tab



```
print(crsxc)

## Graphically display the correlations.

plotcorr(crsxc, col=cm.colors(11)[5*crsxc + 6])
title(main="Correlation of Missing Values audit.csv", sub=paste("Rattle", Sys.time(), Sys.info()["user"]))

##============================================================
## Timestamp: 2006-08-19 08:13:57

## The randomForest package supplies the randomForest function.

library(randomForest)

## Build a randomForest model.

crs$rf <- randomForest(as.factor(Adjusted) ~ ., data=crs$dataset[crs$sample,c(2:10,13)], na.action=na.omit)

## Generate textual output of randomForest model.

crs$rf

## List the importance of the variables.

round(importance(crs$rf), 2)
```

All R commands that Rattle runs underneath are exposed through the text view of the Log tab. The intention is that the R commands be available for copying into the R console so that where Rattle only exposes a limited number of options, further options can be tuned via the R console.

The Log tab aims to be educational as much as possible. Informative comments are included to describe the steps involved.

Also, the whole log can be saved to a script file (with a R filename extension) and in principle, loaded into R to repeat the exact steps of the Rattle interactions. In general, you may want to review the steps and fine tune them to suit your purposes. After pasting the contents of the Log text view into a file, perhaps with a filename of `audit-rf-risk.R`, you can have the file execute as a script in R with:

```
> source("audit-rf-risk.R")
```

Internally, Rattle uses a variable called crs to store its current state, and you can modify this variable directly. Generally, changes you make will be reflected within Rattle and vice versa.

5.13 Troubleshooting

5.13.1 A factor has new levels

This occurs when the training dataset does not contain examples of all of the levels of particular factor and the testing set contains examples of these other levels.