Data Analytics and Business Intelligence (8696/8697)

Introducing Data Science with Rattle and R

Graham.Williams@togaware.com

Chief Data Scientist
Australian Taxation Office

Adjunct Professor, Australian National University
Adjunct Professor, University of Canberra
Fellow, Institute of Analytics Professionals of Australia

Graham.Williams@togaware.com
http://datamining.togaware.com

Overview

Overview

Data Mining

A data driven analysis to uncover otherwise unknown but useful patterns in large datasets, to discover new knowledge and to develop predictive models, turning data and information into knowledge and (one day perhaps) wisdom, in a timely manner.

Data Mining

- Application of
  - Machine Learning
  - Statistics
  - Software Engineering and Programming with Data
  - Effective Communications and Intuition
- . . . to Datasets that vary by
  Volume, Velocity, Variety, Value, Veracity
- . . . to discover new knowledge
- . . . to improve business outcomes
- . . . to deliver better tailored services

Data Mining in Research

- **Health Research**
  Adverse reactions using linked Pharmaceutical, General Practitioner, Hospital, Pathology datasets.
- **Astronomy**
  Microlensing events in the Large Magellanic Cloud of several million observed stars (out of 10 billion).
- **Psychology**
  Investigation of age-of-onset for Alzheimer's disease from 75 variables for 800 people.
- **Social Sciences**
  Survey evaluation. Social network analysis - identifying key influencers.

Data Mining in Government

- **Australian Taxation Office**
  - Lodgment ($110M)
  - Tax Havens ($150M)
  - Tax Fraud ($250M)
- **Immigration and Border Control**
  - Check passengers before boarding
- **Health and Human Services**
  - Doctor shoppers
  - Over servicing

The Business of Data Mining

- SAS has annual revenues of $3B (2013)
- IBM bought SPSS for $1.2B (2009)

- Analytics is >$100B business and >$320B by 2020
- Amazon, eBay/PayPal, Google, Facebook, LinkedIn, ...
- Shortage of 180,000 data scientists in US in 2018 (McKinsey) ...

Basic Data Mining Algorithms

- Cluster Analysis (kmeans, wskm)
- Association Analysis (arules)
- Linear Discriminant Analysis (lda)
- Logistic Regression (glm)
- Decision Trees (rpart, wsrpart)
- Random Forests (randomForest, wsrf)
- Boosted Stumps (ada)
- Neural Networks (nnet)
- Support Vector Machines (kernlab)
- ...

That's a lot of tools to learn in R!
Many with different interfaces and options.

Overview

Installing R and Rattle

- **First task is to install R**

  As free/libre open source software (FLOSS or FOSS), R and Rattle are available to all, with no limitations on our freedom to use and share the software, except to share and share alike.

- Visit CRAN at `http://cran.rstudio.com`
- Visit Rattle at `http://rattle.togaware.com`

- Linux: Install packages (Ubuntu is recommended)
  `$ wajig install r-recommended r-cran-rattle`
- Windows: Download and install from CRAN
- MacOSX: Download and install from CRAN

Why do Data Science with R?

- Most widely used Data Mining and Machine Learning Package
  - Machine Learning
  - Statistics
  - Software Engineering and Programming with Data
  - But not the nicest of languages for a Computer Scientist!
- Free (Libre) Open Source Statistical Software
  - . . . all modern statistical approaches
  - . . . many/most machine learning algorithms
  - . . . opportunity to readily add new algorithms
- That is important for us in the research community
  Get our algorithms out there and being used—impact!!!

Why do Data Science with R?

- Most widely used Data Mining and Machine Learning Package
  - Machine Learning
  - Statistics
  - Software Engineering and Programming with Data
  - But not the nicest of languages for a Computer Scientist!
- Free (Libre) Open Source Statistical Software
  - . . . all modern statistical approaches
  - . . . many/most machine learning algorithms
  - . . . opportunity to readily add new algorithms
- That is important for us in the research community
  Get our algorithms out there and being used—impact!!!

Why do Data Science with R?

- Most widely used Data Mining and Machine Learning Package
  - Machine Learning
  - Statistics
  - Software Engineering and Programming with Data
  - But not the nicest of languages for a Computer Scientist!
- Free (Libre) Open Source Statistical Software
  - . . . all modern statistical approaches
  - . . . many/most machine learning algorithms
  - . . . opportunity to readily add new algorithms
- That is important for us in the research community
  Get our algorithms out there and being used—impact!!!

Why do Data Science with R?

- Most widely used Data Mining and Machine Learning Package
  - Machine Learning
  - Statistics
  - Software Engineering and Programming with Data
  - But not the nicest of languages for a Computer Scientist!
- Free (Libre) Open Source Statistical Software
  - . . . all modern statistical approaches
  - . . . many/most machine learning algorithms
  - . . . opportunity to readily add new algorithms
- That is important for us in the research community
  Get our algorithms out there and being used—impact!!!

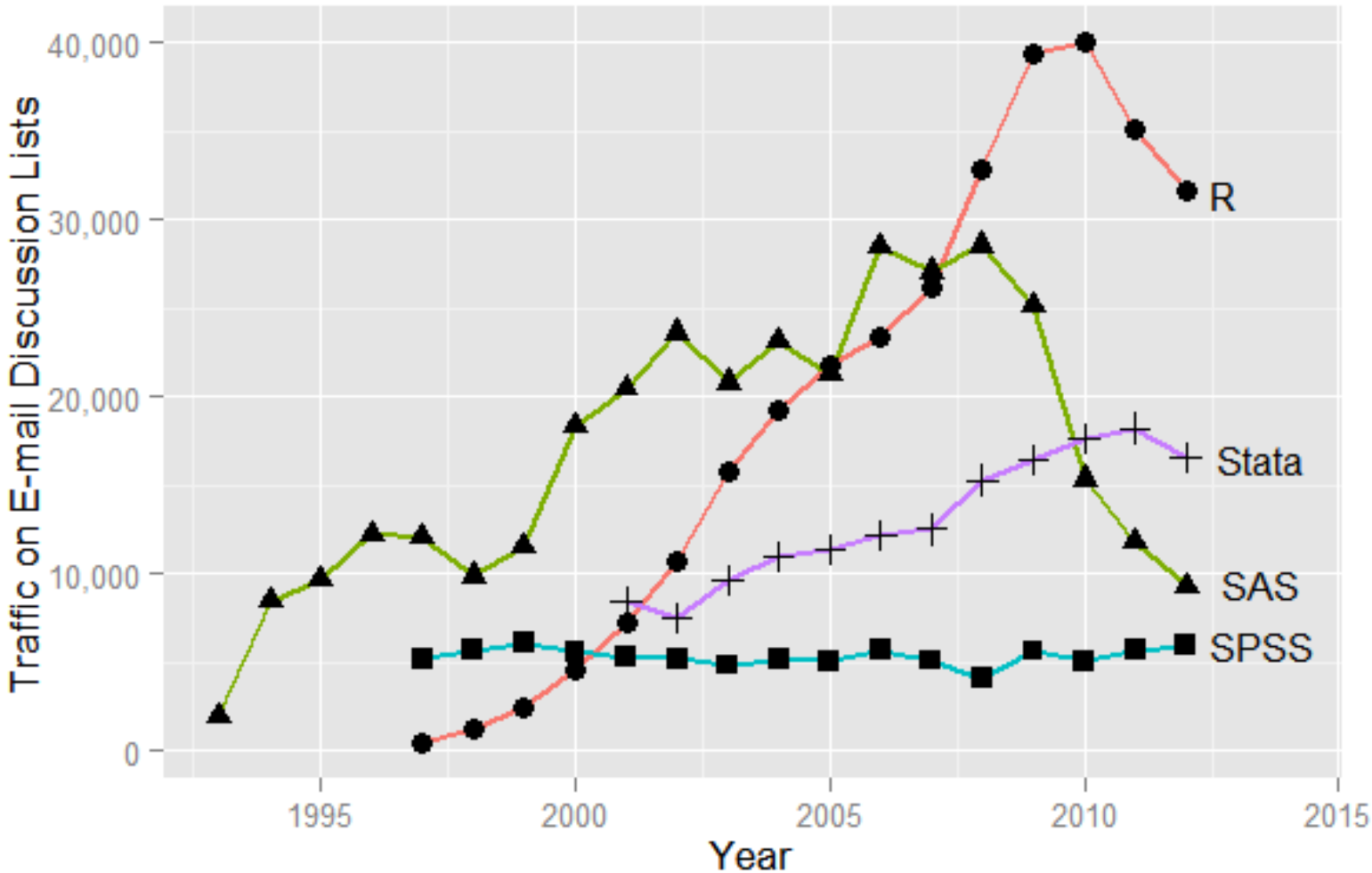
R: A Dangerous Tool?

"I think it addresses a niche market for high-end data analysts that want free, readily available code. We have customers who build engines for aircraft. I am happy they are not using freeware when I get on a jet." Anne H. Milley, director of technology product marketing at SAS (New York Times, 7 January 2009).

It's interesting that SAS Institute feels that non-peer-reviewed software with hidden implementations of analytic methods that cannot be reproduced by others should be trusted when building aircraft engines. (Frank Harrell)

How Popular is R? Discussion List Traffic

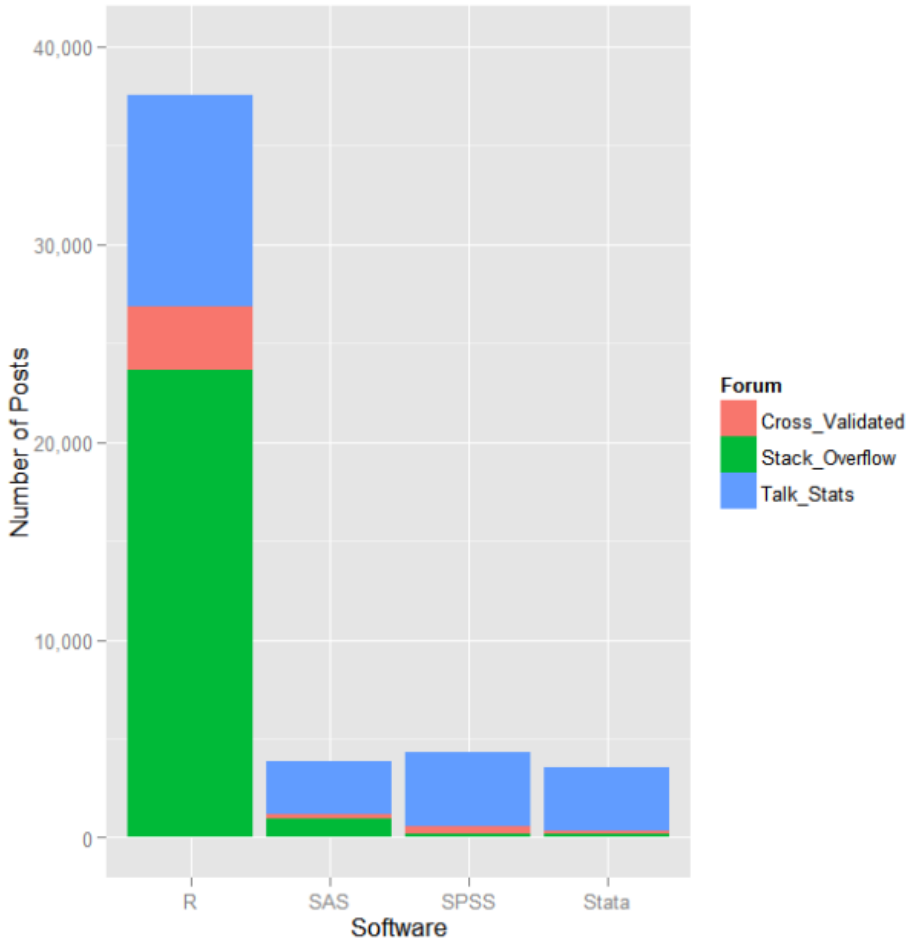
Monthly email traffic on software's main discussion list.



Source: http://r4stats.com/articles/popularity/

How Popular is R? Discussion Topics

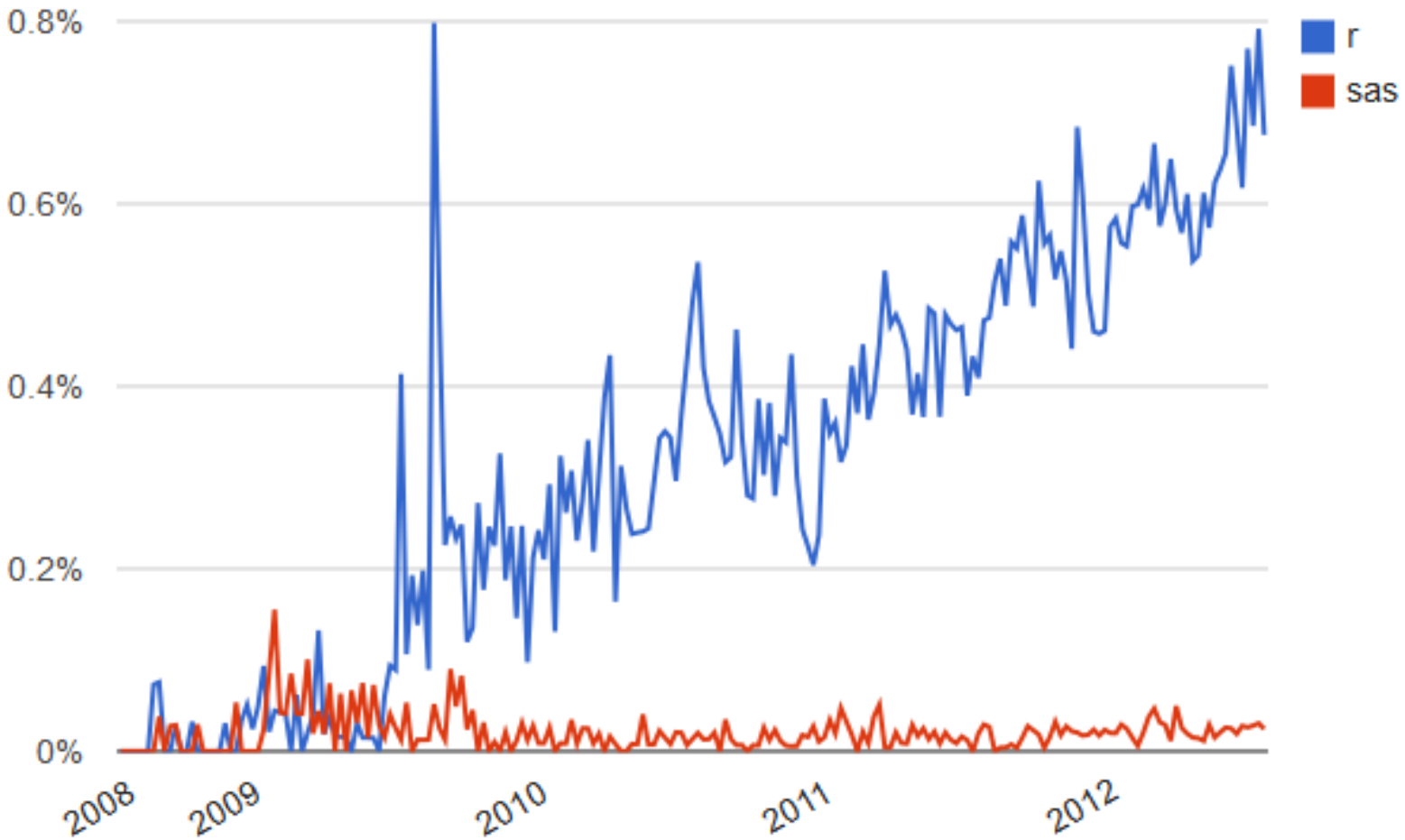
Number of discussions on popular QandA forums 2013.



Source: http://r4stats.com/articles/popularity/

How Popular is R? R versus SAS

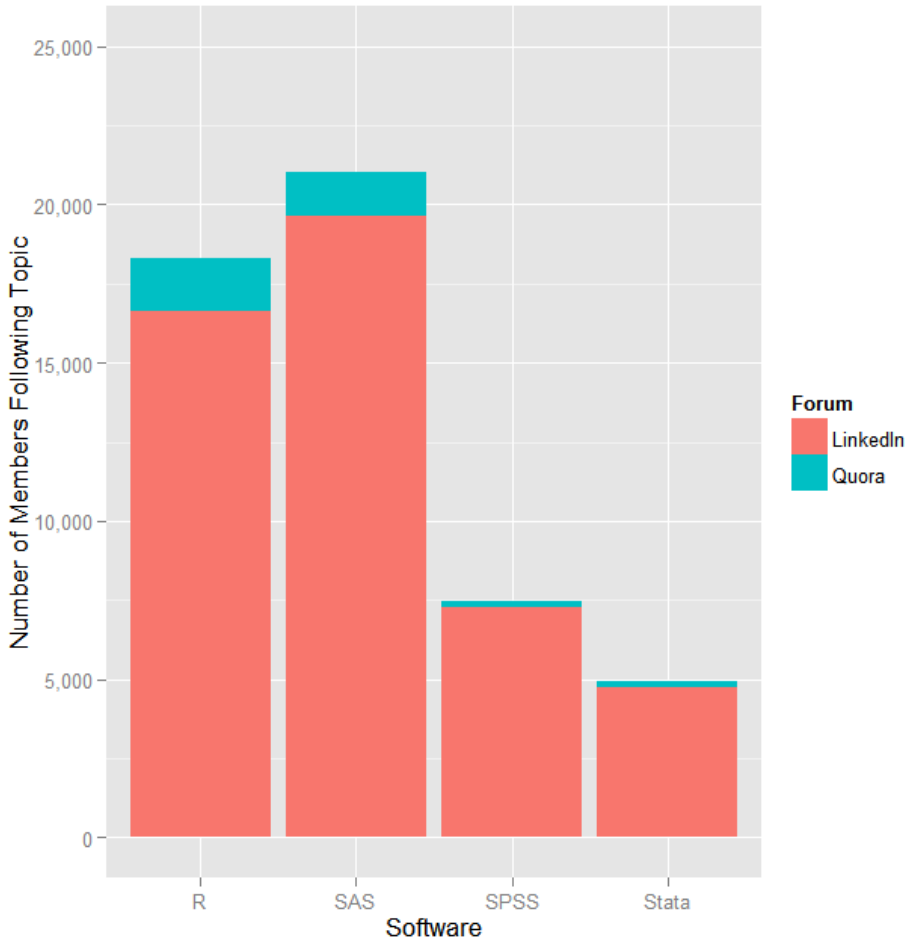
Number of R/SAS related posts to Stack Overflow by week.



Source: http://r4stats.com/articles/popularity/

How Popular is R? Professional Forums

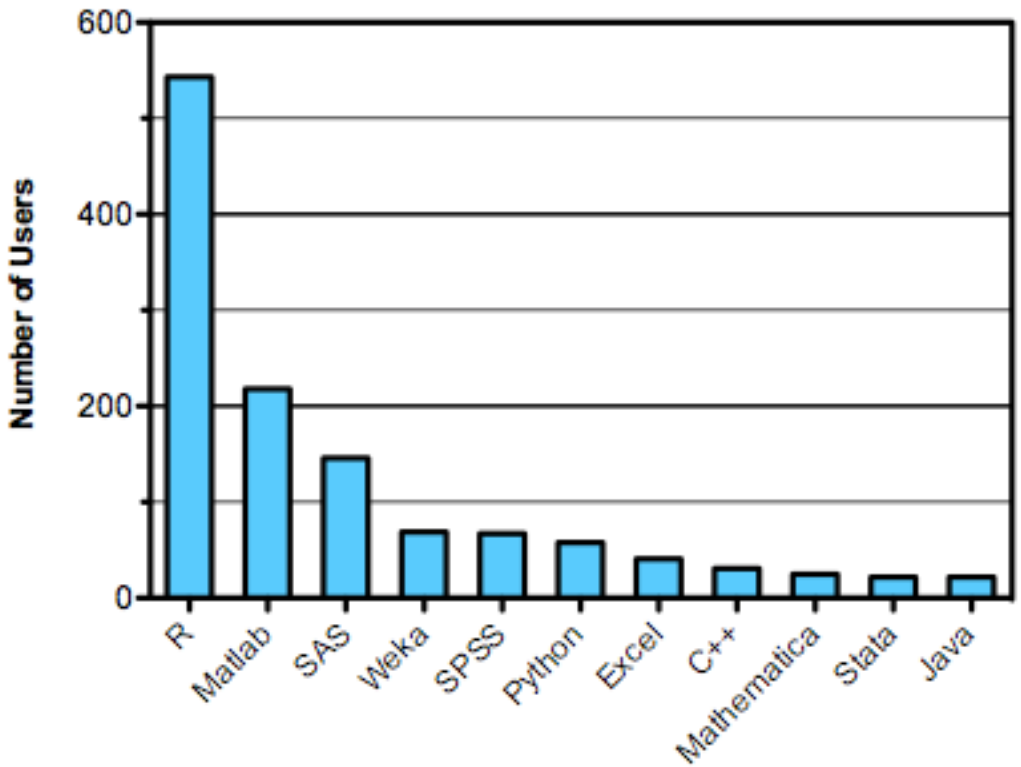
Registered for the main discussion group for each software.



Source: http://r4stats.com/articles/popularity/

How Popular is R? Used in Analytics Competitions

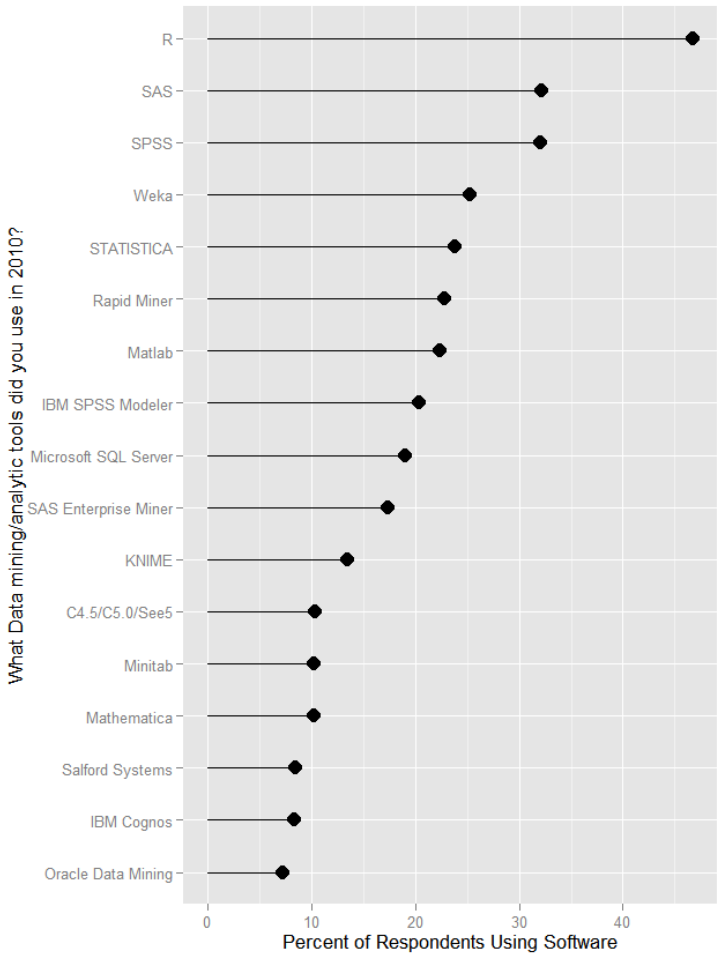
Software used in data analysis competitions in 2011.



Source: http://r4stats.com/articles/popularity/

How Popular is R? User Survey

Rexer Analytics Survey 2010 results for data mining/analytic tools.



Source: http://r4stats.com/articles/popularity/

R Skills Attract Good Salaries

- 2014 survey of average US tech salaries by Dice Tech puts R at the top of the list at $115,531

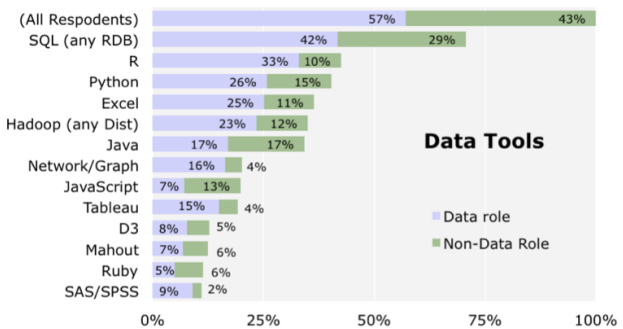
  ComputerWorld

- 2013 O'Rielly Strata Conference: Data Scientists use R over other data programming languages and Data Scientists using Open Source earn $130,000 on average.

  Revolution Analytcs

AVERAGE SALARY FOR High Paying Skills and Experience

SKILL	2013	YR/YR CHANGE
R	$ 115,531	n/a
NoSQL	$ 114,796	1.6%
MapReduce	$ 114,396	n/a
PMBok	$ 112,382	1.3%
Cassandra	$ 112,382	n/a
Omnigraffle	$ 111,039	0.3%
Pig	$ 109,561	n/a
SOA (Service Oriented Architecture)	$ 108,997	-0.5%
Hadoop	$ 108,669	-5.6%
Mongo DB	$ 107,825	-0.4%



What is R?

R — The Video

A 90 Second Promo from Revolution Analytics

`http://www.revolutionanalytics.com/what-is-open-source-r/`

Choosing R for Data Science

- Data Science is about Analysing Data;
- R is freely available to all to analyse data;
- R has the most extensive suite of functionality available;
- Nothing else is any longer even close.

This document, sourced from StartL.Rnw revision 436, 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 2014-06-21 20:26:13.

Overview

Why a GUI for Data Science in R?

- Statistics can be complex and traps await
- **So many** tools in R to deliver insights
- Effective analyses should be scripted
- Scripting also required for repeatability
- R is a language for **programming** with data

How to remember how to do all of this in R?
How to skill up 150 data analysts with Data Mining?

Users of Rattle

Today, Rattle is used world wide in many industries

- Health analytics
- Customer segmentation and marketing
- Fraud detection
- Government

It is used by

- Universities to teach Data Mining
- Within research projects for basic analyses
- Consultants and Analytics Teams across business

It is and will remain freely available.

CRAN and `http://rattle.togaware.com`

Installation

- Rattle is built using R
- Need to download and install R from cran.r-project.org
- Recommend also install RStudio from www.rstudio.org
- Then start up RStudio and install Rattle:

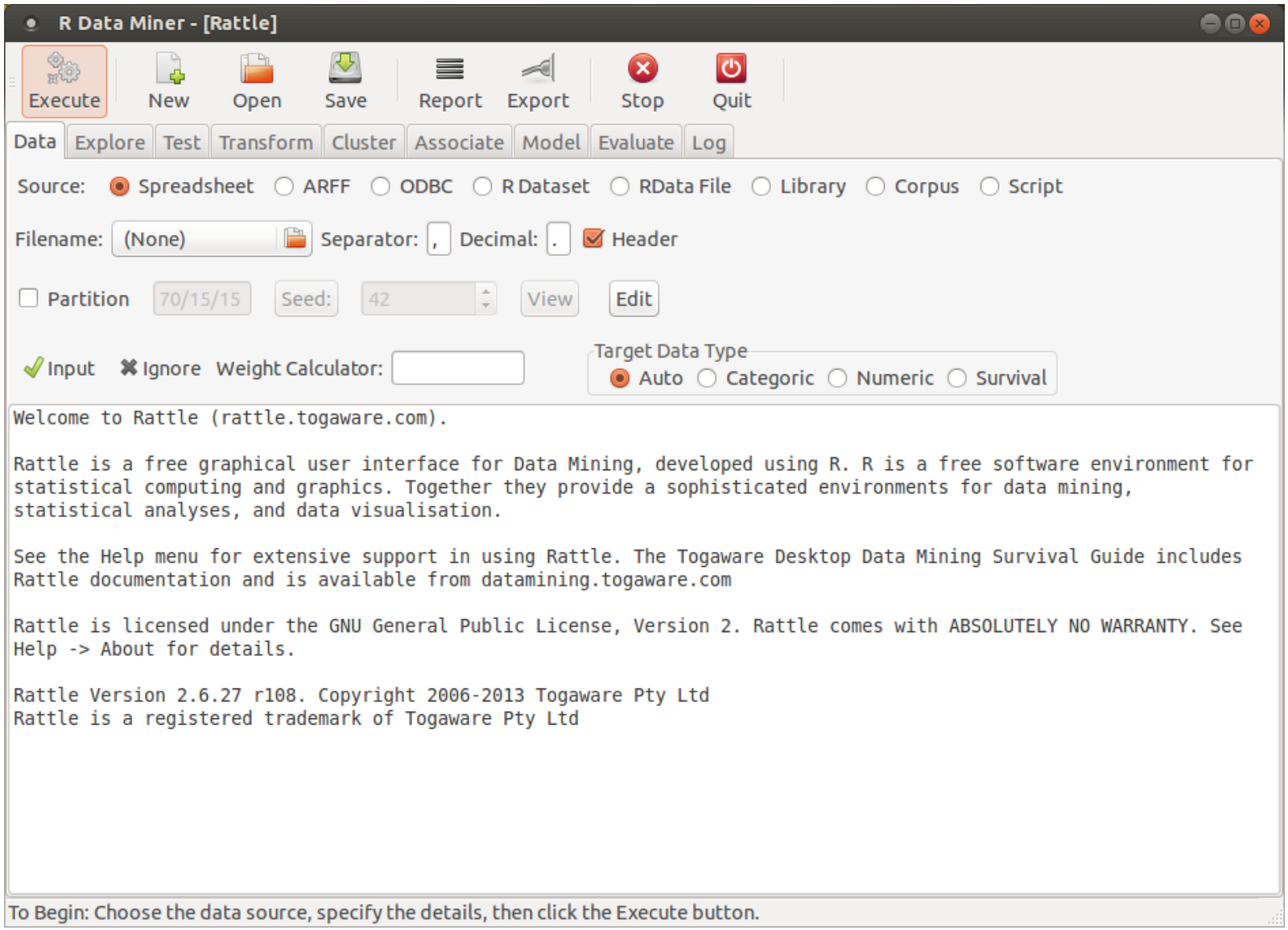
```
install.packages("rattle")
```

- Then we can start up Rattle:

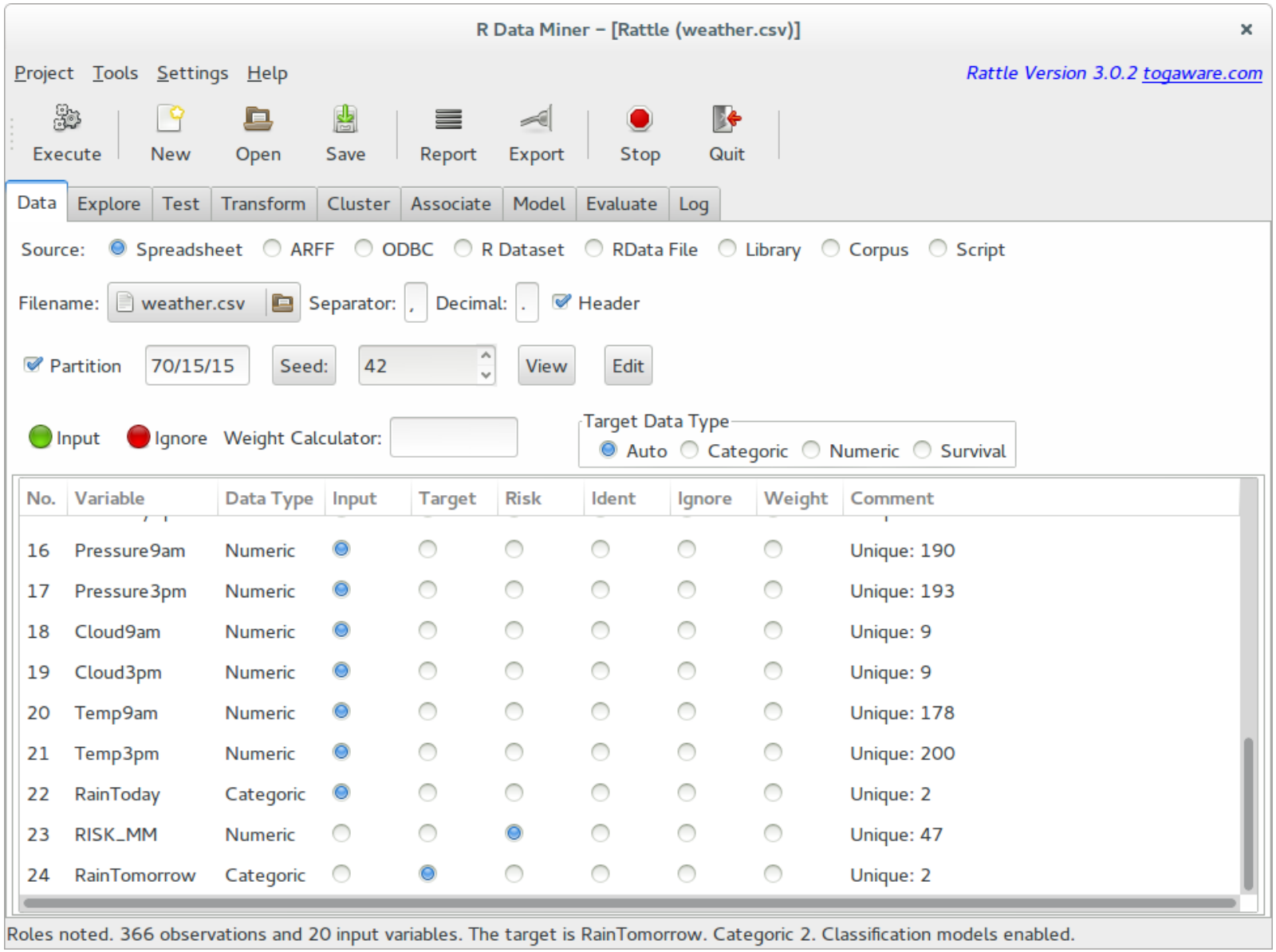
```
rattle()
```

- Required packages are loaded as needed.

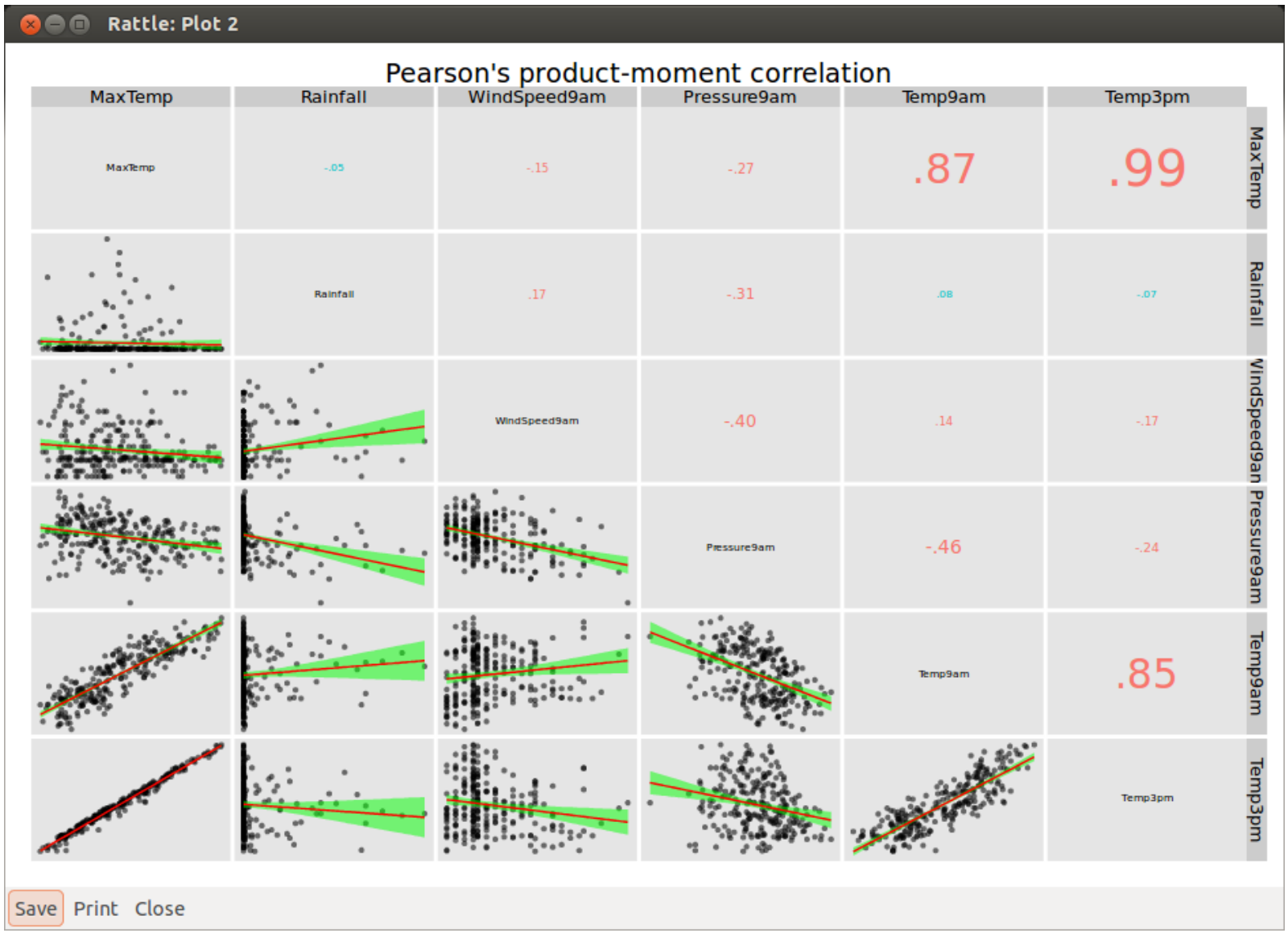
A Tour Thru Rattle: Startup



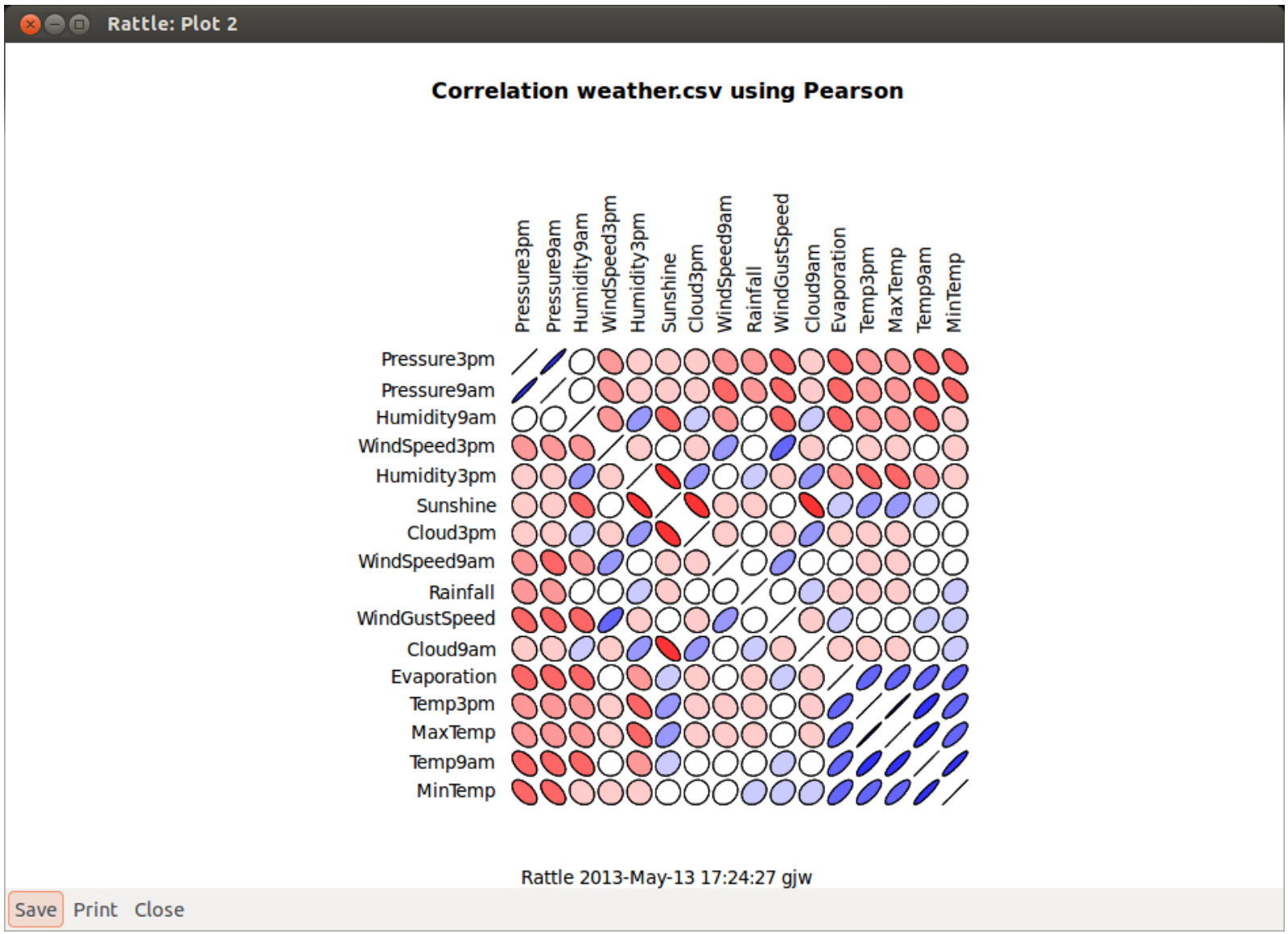
A Tour Thru Rattle: Loading Data



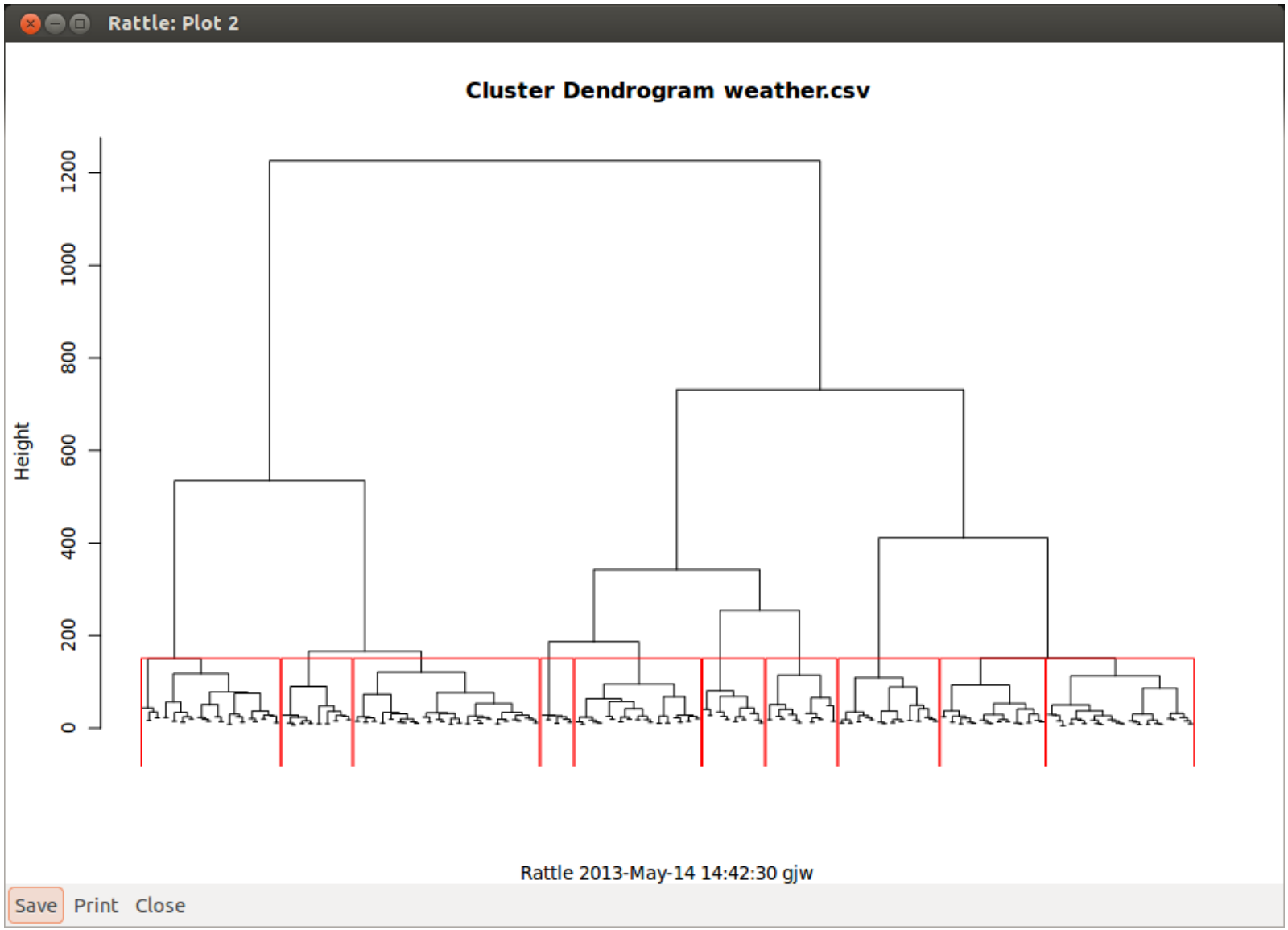
A Tour Thru Rattle: Explore Distribution



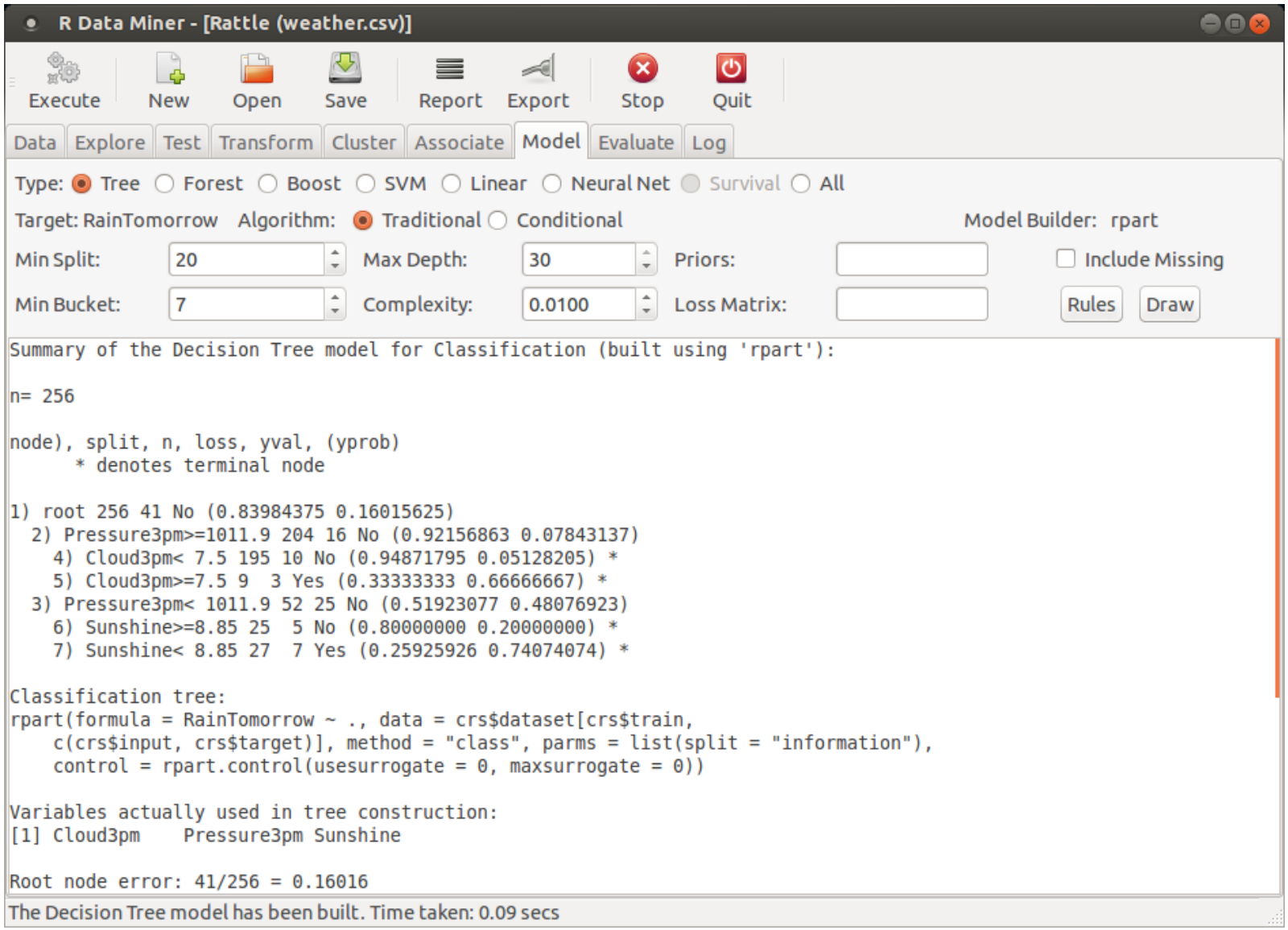
A Tour Thru Rattle: Explore Correlations



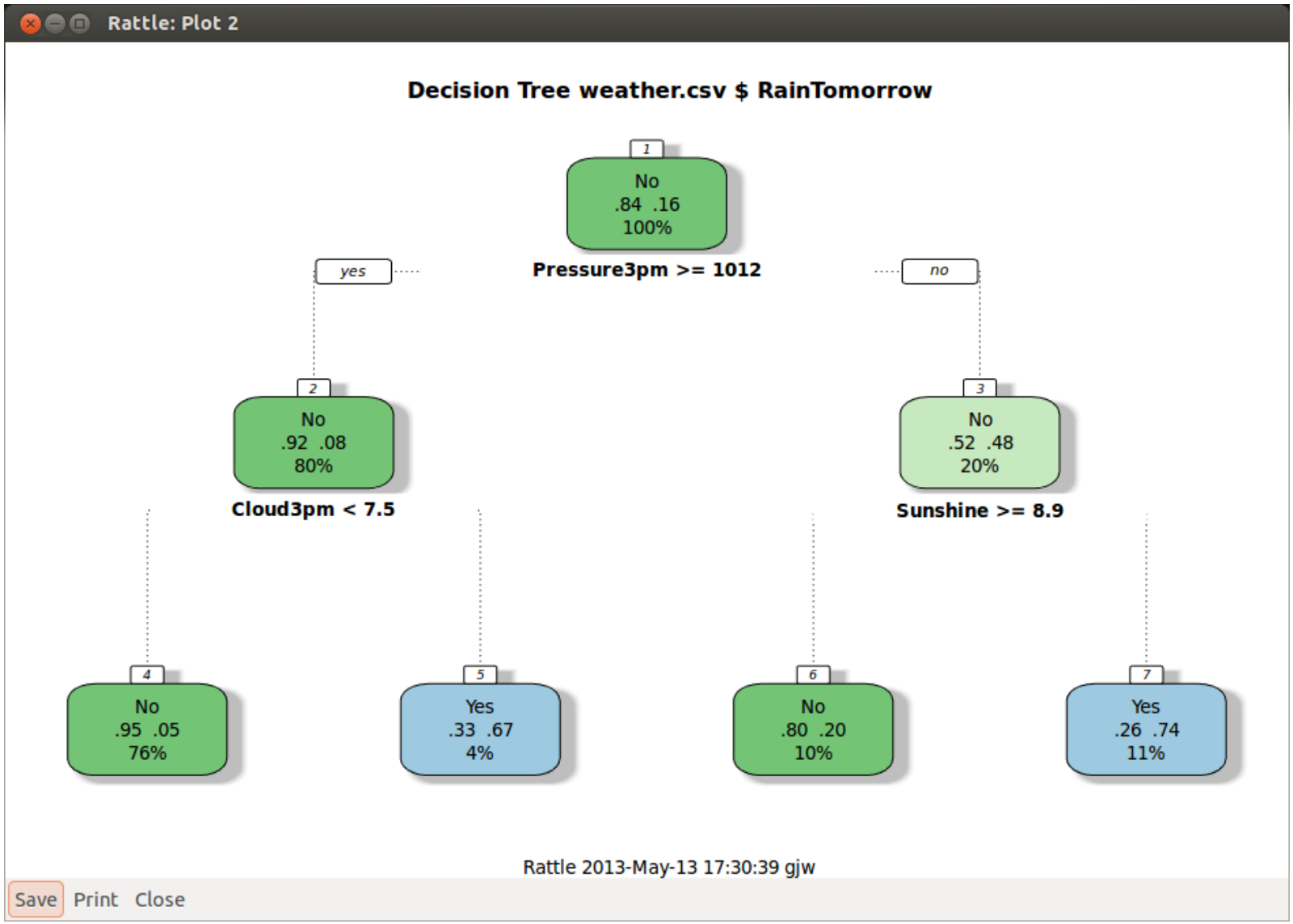
A Tour Thru Rattle: Hierarchical Cluster



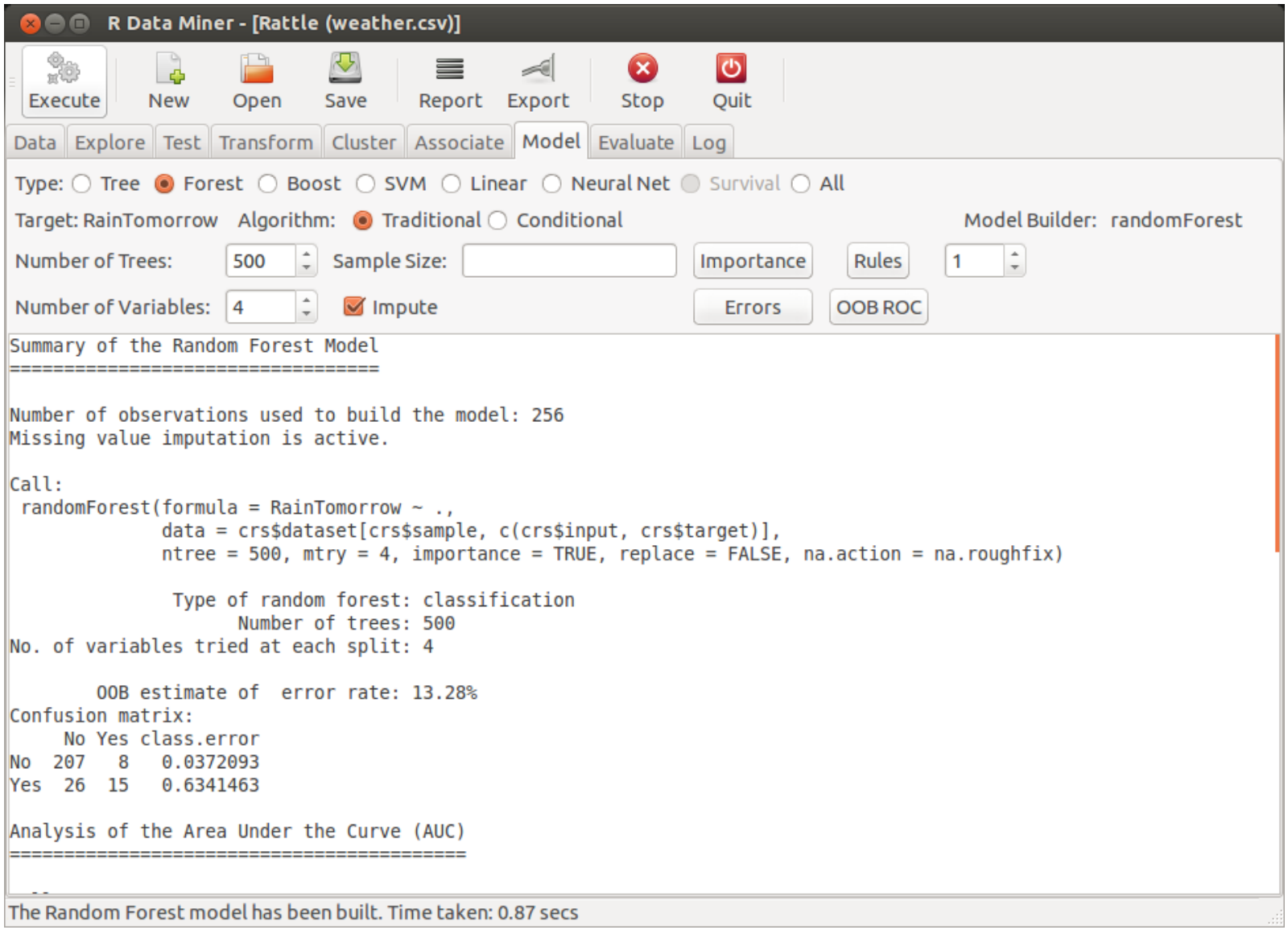
A Tour Thru Rattle: Decision Tree



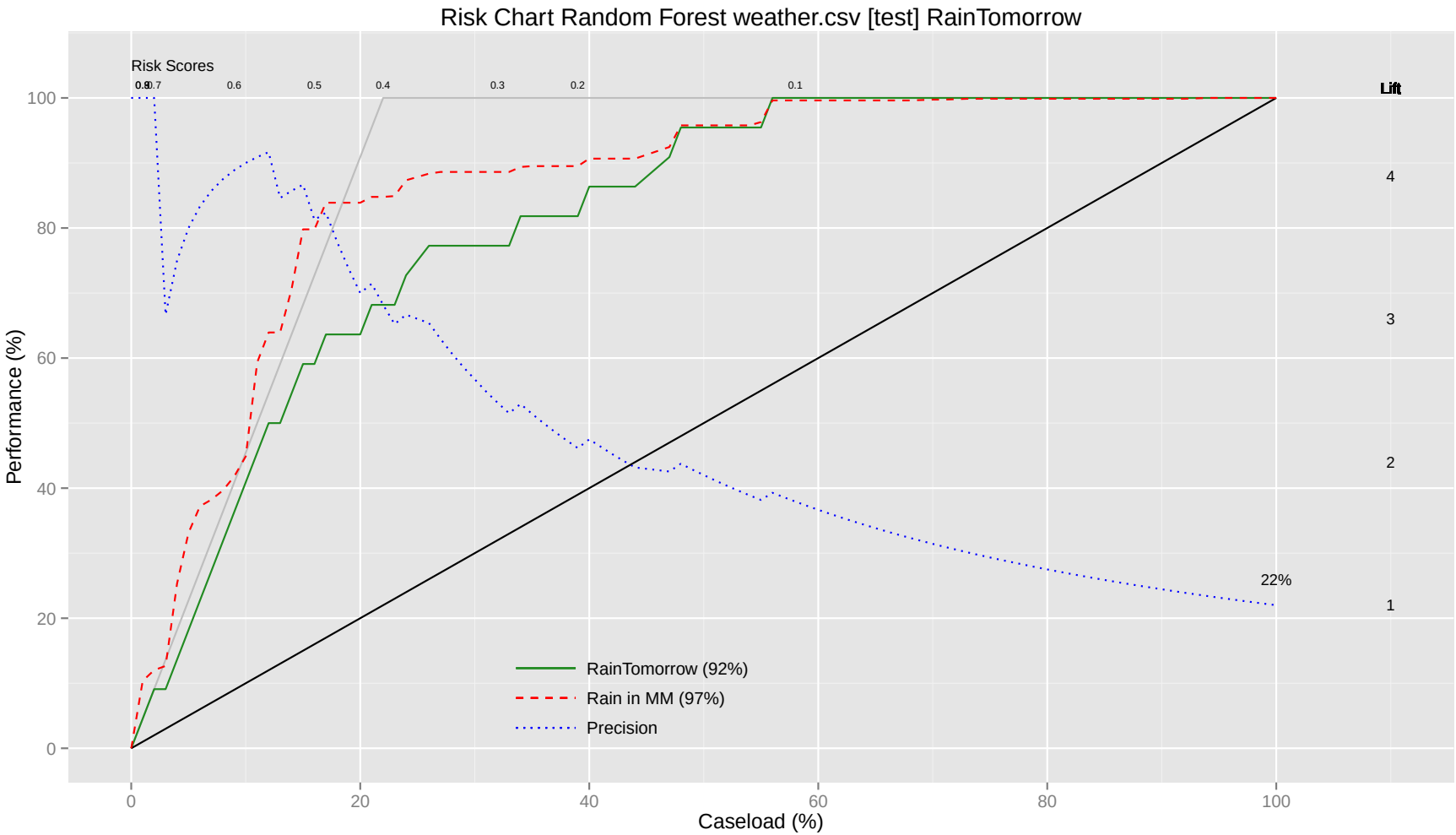
A Tour Thru Rattle: Decision Tree Plot



A Tour Thru Rattle: Random Forest



A Tour Thru Rattle: Risk Chart



Rattle Interface Notes

- Work through the tabs from left to right
- After setting up a tab we need to Execute it
- Projects save the current Rattle state
- Projects can be restored at a later time

Overview

Data Scientists are Programmers of Data

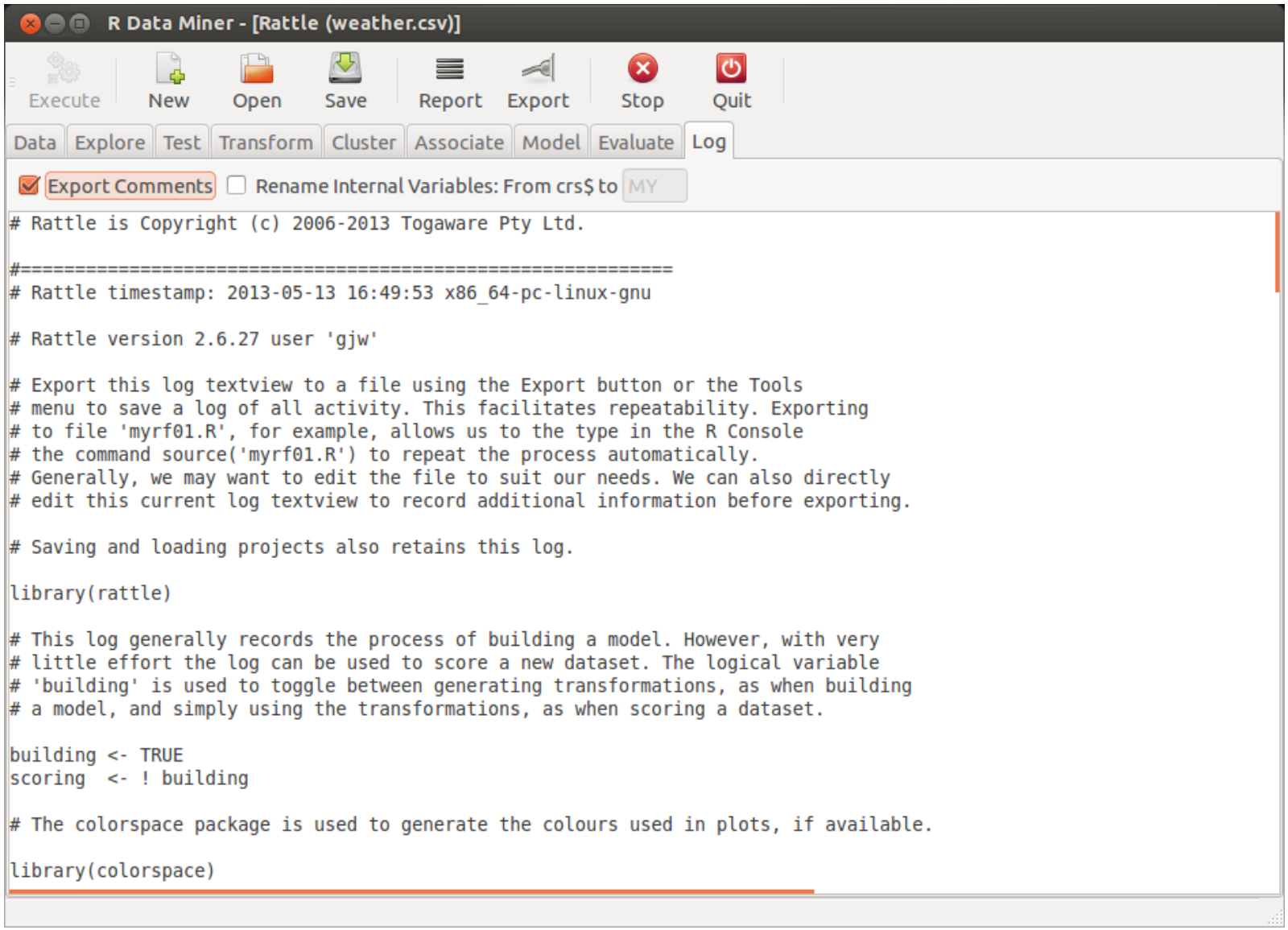
But...

- Data scientists are programmers of data
- A GUI can only do so much
- R is a powerful statistical language

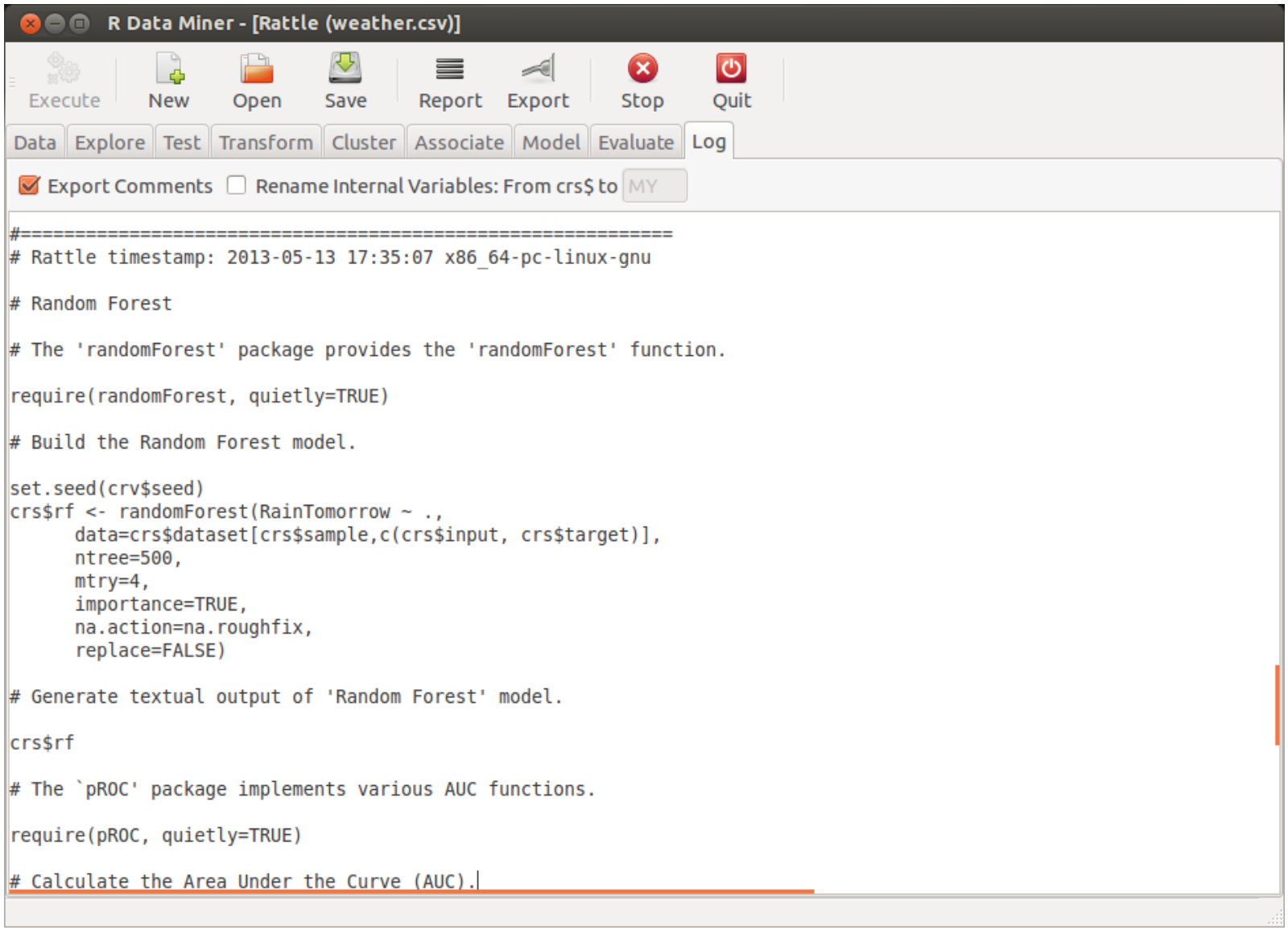
Data Scientists Desire...

- Scripting
- Transparency
- Repeatability
- Sharing

From GUI to CLI — Rattle's Log Tab



From GUI to CLI — Rattle's Log Tab



Step 1: Load the Dataset

```
dsname <- "weather"
ds     <- get(dsname)
dim(ds)
```

```
## [1] 366  24
```

```
names(ds)
```

```
##  [1] "Date"          "Location"      "MinTemp"       "...
##  [5] "Rainfall"      "Evaporation"   "Sunshine"      "...
##  [9] "WindGustSpeed" "WindDir9am"    "WindDir3pm"    "...
## [13] "WindSpeed3pm"  "Humidity9am"   "Humidity3pm"   "...
....
```

Step 2: Observe the Data — Observations

```
head(ds)
```

```
##         Date Location MinTemp MaxTemp Rainfall Evapora...
## 1 2007-11-01 Canberra     8.0    24.3      0.0         ...
## 2 2007-11-02 Canberra    14.0    26.9      3.6         ...
## 3 2007-11-03 Canberra    13.7    23.4      3.6         ...
....
```

```
tail(ds)
```

```
##           Date Location MinTemp MaxTemp Rainfall Evapo...
## 361 2008-10-26 Canberra     7.9    26.1        0       ...
## 362 2008-10-27 Canberra     9.0    30.7        0       ...
## 363 2008-10-28 Canberra     7.1    28.4        0       ...
....
```

Step 2: Observe the Data — Structure

```
str(ds)
```

```
## 'data.frame': 366 obs. of  24 variables:
##  $ Date          : Date, format: "2007-11-01" "2007-11-...
##  $ Location      : Factor w/ 49 levels "Adelaide","Alba...
##  $ MinTemp       : num  8 14 13.7 13.3 7.6 6.2 6.1 8.3 ...
##  $ MaxTemp       : num  24.3 26.9 23.4 15.5 16.1 16.9 1...
##  $ Rainfall      : num  0 3.6 3.6 39.8 2.8 0 0.2 0 0 16...
##  $ Evaporation   : num  3.4 4.4 5.8 7.2 5.6 5.8 4.2 5.6...
##  $ Sunshine      : num  6.3 9.7 3.3 9.1 10.6 8.2 8.4 4....
##  $ WindGustDir   : Ord.factor w/ 16 levels "N"<"NNE"<"N...
##  $ WindGustSpeed: num  30 39 85 54 50 44 43 41 48 31 ...
##  $ WindDir9am    : Ord.factor w/ 16 levels "N"<"NNE"<"N...
##  $ WindDir3pm    : Ord.factor w/ 16 levels "N"<"NNE"<"N...
....
```

Step 2: Observe the Data — Summary

```
summary(ds)
```

```
##       Date                    Location      MinTemp  ...
##  Min.   :2007-11-01   Canberra     :366   Min.   :-5.3...
##  1st Qu.:2008-01-31   Adelaide     :  0   1st Qu.: 2.3...
##  Median :2008-05-01   Albany       :  0   Median : 7.4...
##  Mean   :2008-05-01   Albury       :  0   Mean   : 7.2...
##  3rd Qu.:2008-07-31   AliceSprings :  0   3rd Qu.:12.5...
##  Max.   :2008-10-31   BadgerysCreek:  0   Max.   :20.9...
##                       (Other)      :  0                ...
##     Rainfall      Evaporation       Sunshine       Wind...
##  Min.   : 0.00   Min.   : 0.20   Min.   : 0.00   NW   ...
##  1st Qu.: 0.00   1st Qu.: 2.20   1st Qu.: 5.95   NNW  ...
##  Median : 0.00   Median : 4.20   Median : 8.60   E    ...
....
```

Step 2: Observe the Data — Variables

```
id     <- c("Date", "Location")
target <- "RainTomorrow"
risk   <- "RISK_MM"
(ignore <- union(id, risk))
```

```
## [1] "Date"     "Location" "RISK_MM"
```

```
(vars   <- setdiff(names(ds), ignore))
```

```
##  [1] "MinTemp"       "MaxTemp"       "Rainfall"      "...
##  [5] "Sunshine"      "WindGustDir"   "WindGustSpeed" "...
##  [9] "WindDir3pm"    "WindSpeed9am"  "WindSpeed3pm"  "...
## [13] "Humidity3pm"   "Pressure9am"   "Pressure3pm"   "...
....
```

Step 3: Clean the Data — Remove Missing

```
dim(ds)
```

```
## [1] 366  24
```

```
sum(is.na(ds[vars]))
```

```
## [1] 47
```

```
ds <- ds[-attr(na.omit(ds[vars]), "na.action"),]
```

Step 3: Clean the Data — Remove Missing

```
dim(ds)
```

```
## [1] 328  24
```

```
sum(is.na(ds[vars]))
```

```
## [1] 0
```

Step 3: Clean the Data—Target as Categoric

```
summary(ds[target])
```

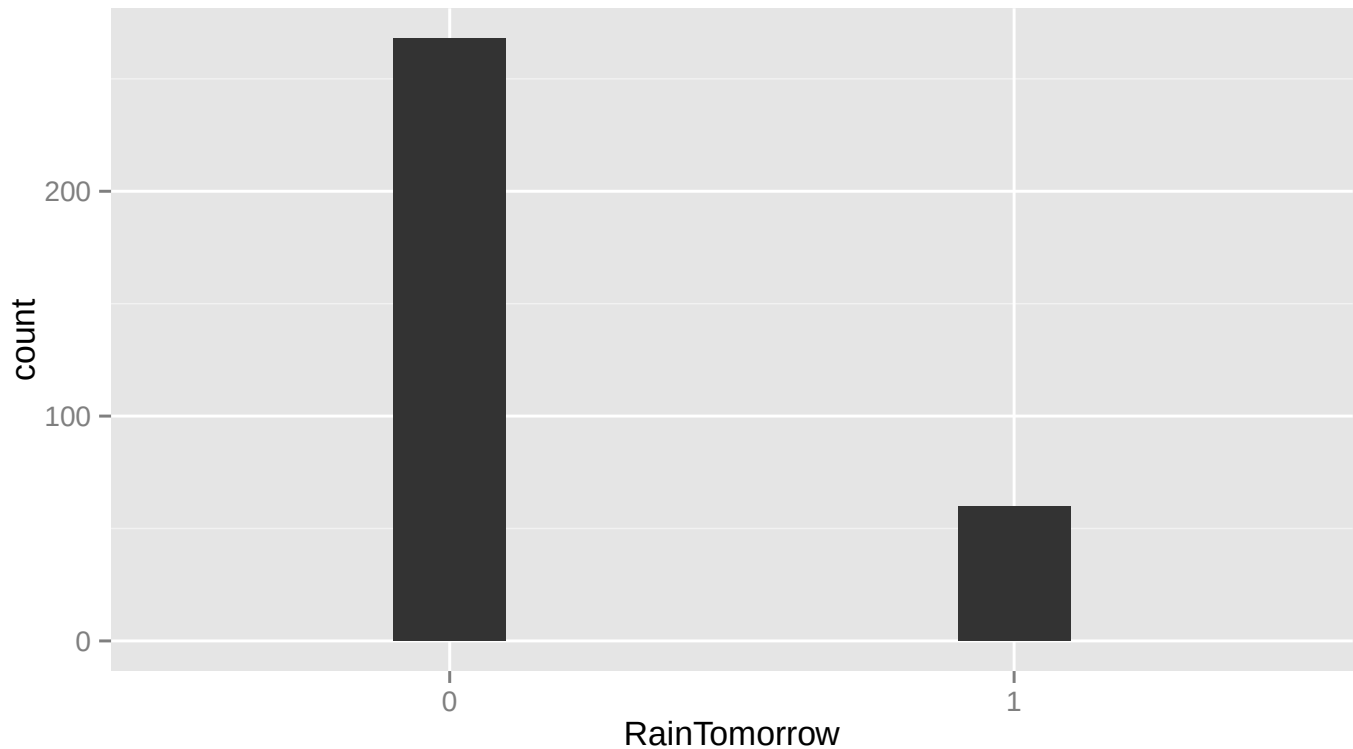
```
##   RainTomorrow
##  Min.   :0.000
##  1st Qu.:0.000
##  Median :0.000
##  Mean   :0.183
##  3rd Qu.:0.000
##  Max.   :1.000
....
```

```
ds[target] <- as.factor(ds[[target]])
levels(ds[target]) <- c("No", "Yes")
```

Step 3: Clean the Data—Target as Categoric

```
summary(ds[target])
```

```
##  RainTomorrow
##  0:268
##  1: 60
```



Step 4: Prepare for Modelling

```
(form <- formula(paste(target, "~ .")))
```

```
## RainTomorrow ~ .
```

```
(nobs <- nrow(ds))
```

```
## [1] 328
```

```
train <- sample(nobs, 0.70*nobs)
length(train)
```

```
## [1] 229
```

```
test  <- setdiff(1:nobs, train)
length(test)
```

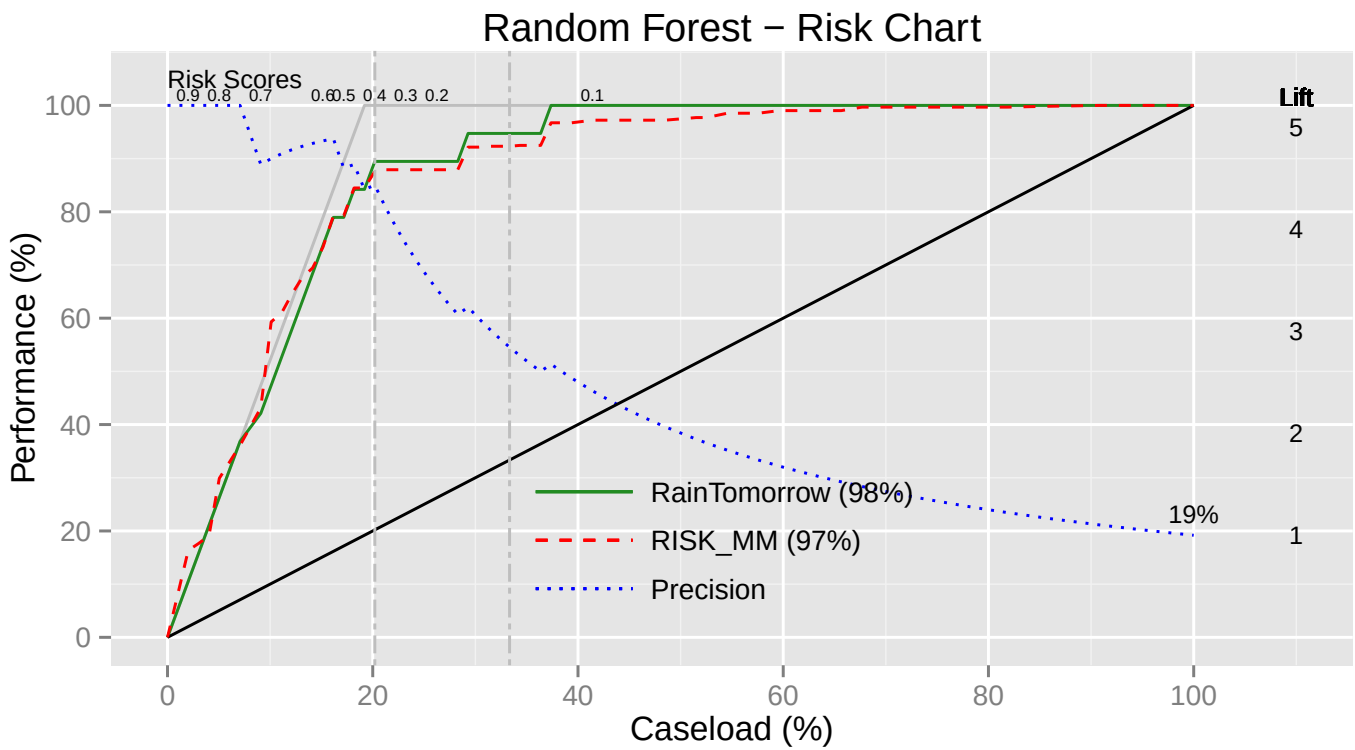
```
## [1] 99
```

Step 5: Build the Model—Random Forest

```
library(randomForest)
model <- randomForest(form, ds[train, vars], na.action=na.omit)
model

##
## Call:
##  randomForest(formula=form, data=ds[train, vars], ...
##                Type of random forest: classification
##                      Number of trees: 500
## No. of variables tried at each split: 4
....
```

Step 6: Evaluate the Model—Risk Chart

```
pr <- predict(model, ds[test,], type="prob")[,2]
riskchart(pr, ds[test, target], ds[test, risk],
          title="Random Forest - Risk Chart",
          risk=risk, recall=target, thresholds=c(0.35, 0.15))
```



Tools

- Ubuntu GNU/Linux operating system
  - Feature rich toolkit, up-to-date, easy to install, FLOSS
- RStudio
  - Easy to use integrated development environment, FLOSS
  - Powerful alternative is Emacs (Speaks Statistics), FLOSS
- R Statistical Software Language
  - Extensive, powerful, thousands of contributors, FLOSS
- KnitR and LaTeX
  - Produce beautiful documents, easily reproducible, FLOSS

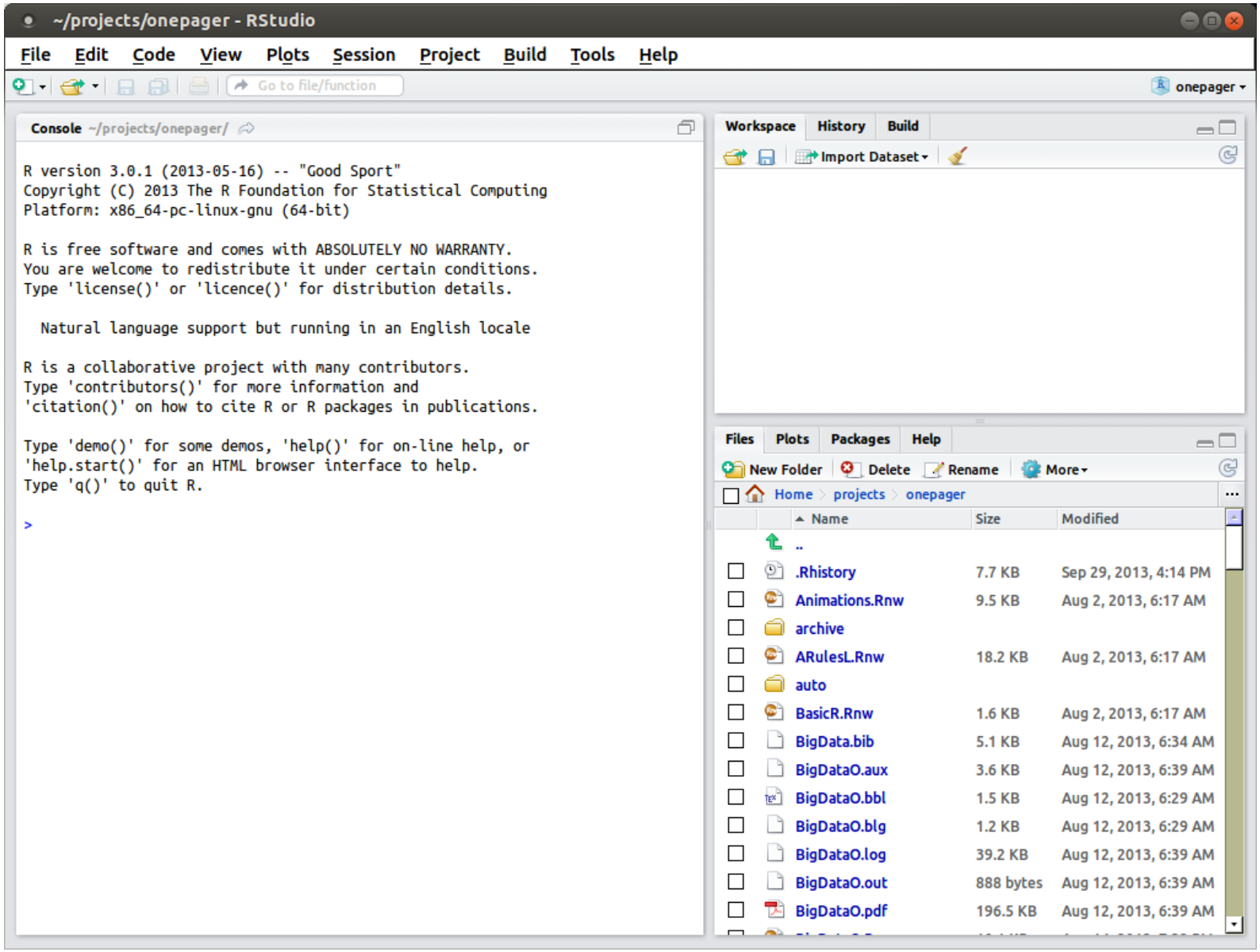
Using Ubuntu

- Desktop Operating System (GNU/Linux)
- Replacing Windows and OSX

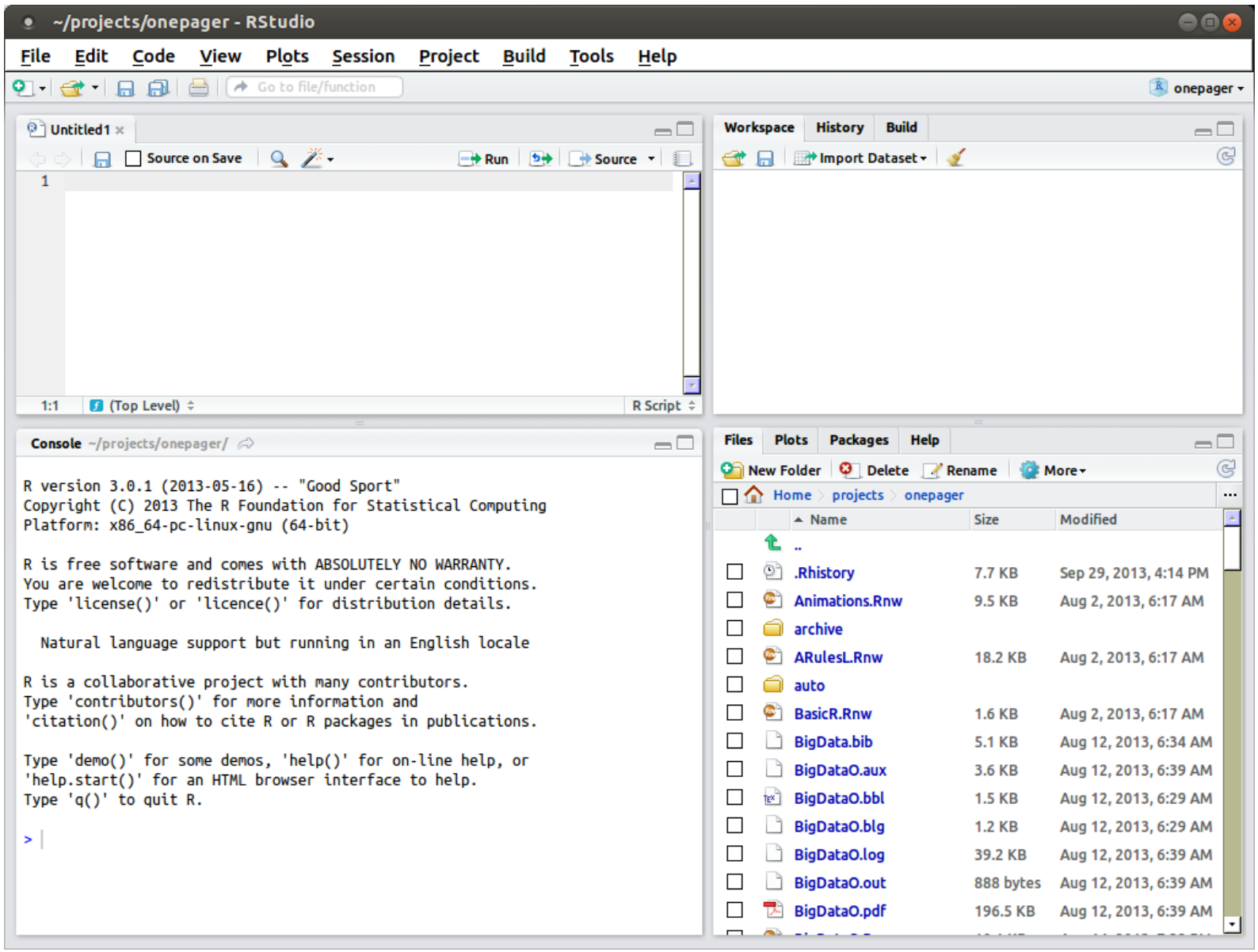
- The GNU Tool Suite based on Unix — significant heritage
- Multiple specialised single task tools, working well together
- Compared to single application trying to do it all
- Powerful data processing from the command line:
  grep, awk, head, tail, wc, sed, perl, python, most, diff, make, paste, join, patch, ...

- For interacting with R — start up RStudio from the Dash

RStudio—The Default Three Panels



RStudio—With R Script File—Editor Panel



Scatterplot—R Code

Our first little bit of R code:

- Load a couple of *packages* into the R *library*

```
library(rattle)  # Provides the weather dataset
library(ggplot2) # Provides the qplot() function
```

- Then produce a quick plot using `qplot()`

```
ds <- weather
qplot(MinTemp, MaxTemp, data=ds)
```

- Your turn: give it a go.

Scatterplot—R Code

Our first little bit of R code:

- Load a couple of *packages* into the R *library*

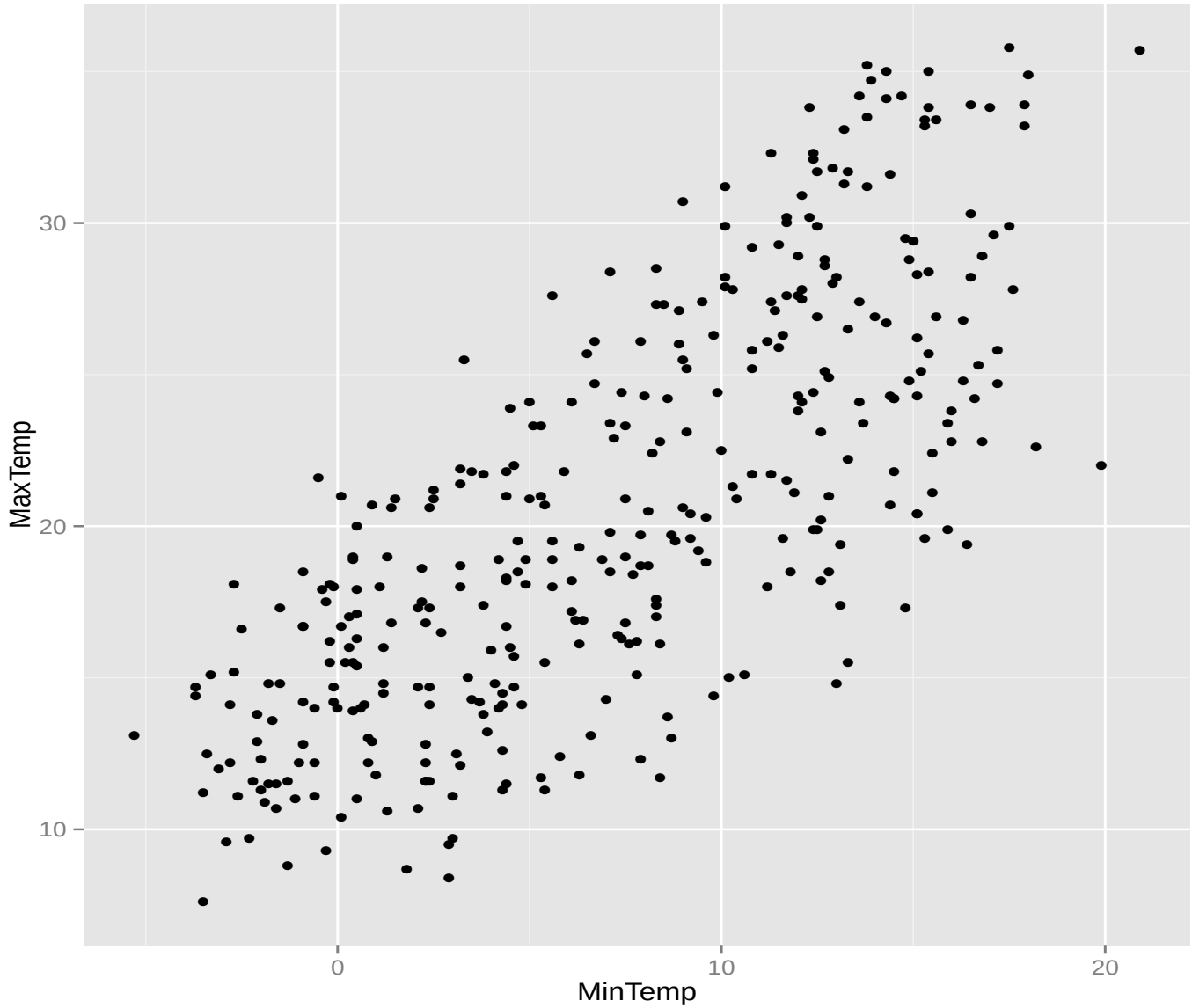
```
library(rattle)  # Provides the weather dataset
library(ggplot2) # Provides the qplot() function
```

- Then produce a quick plot using `qplot()`

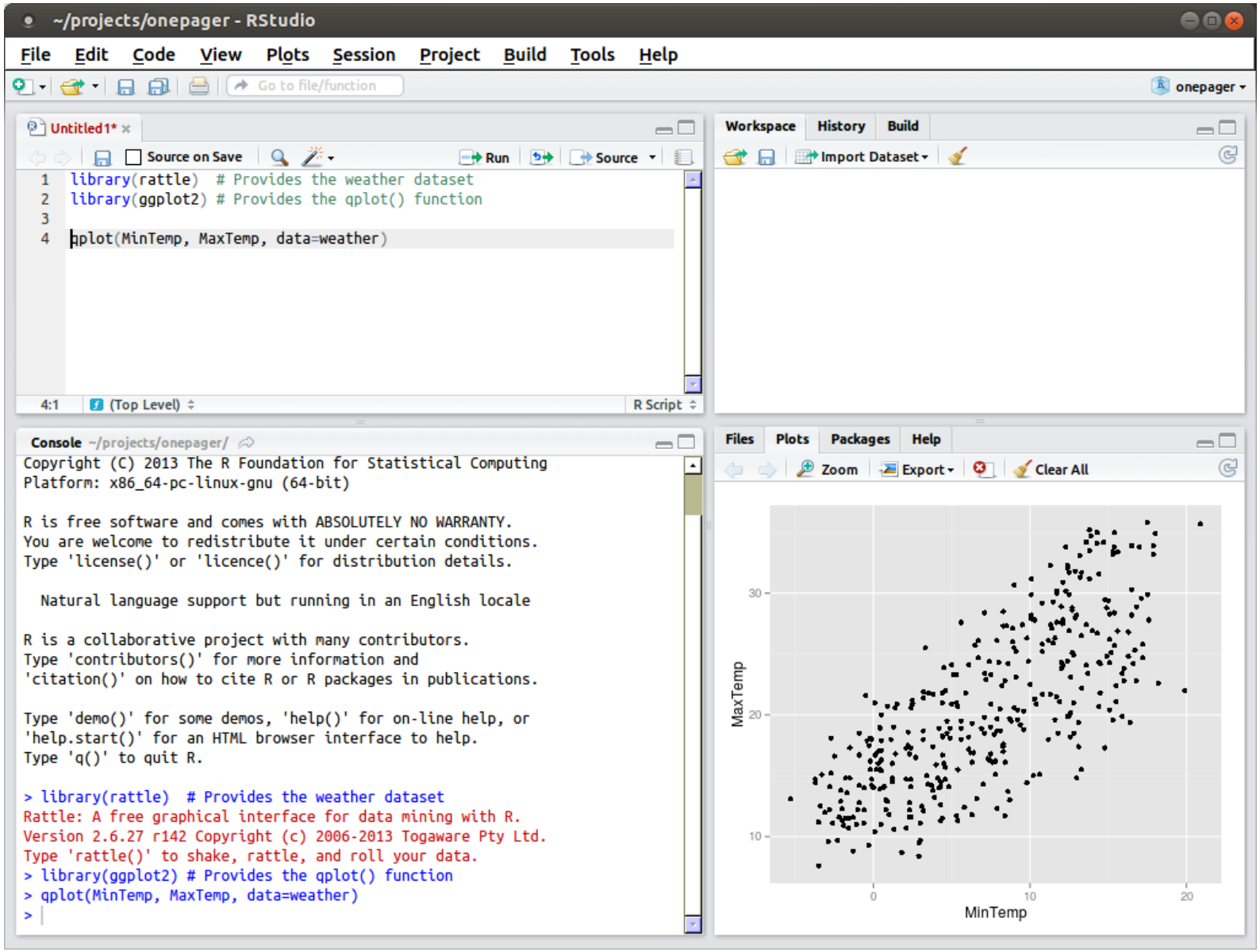
```
ds <- weather
qplot(MinTemp, MaxTemp, data=ds)
```

- Your turn: give it a go.

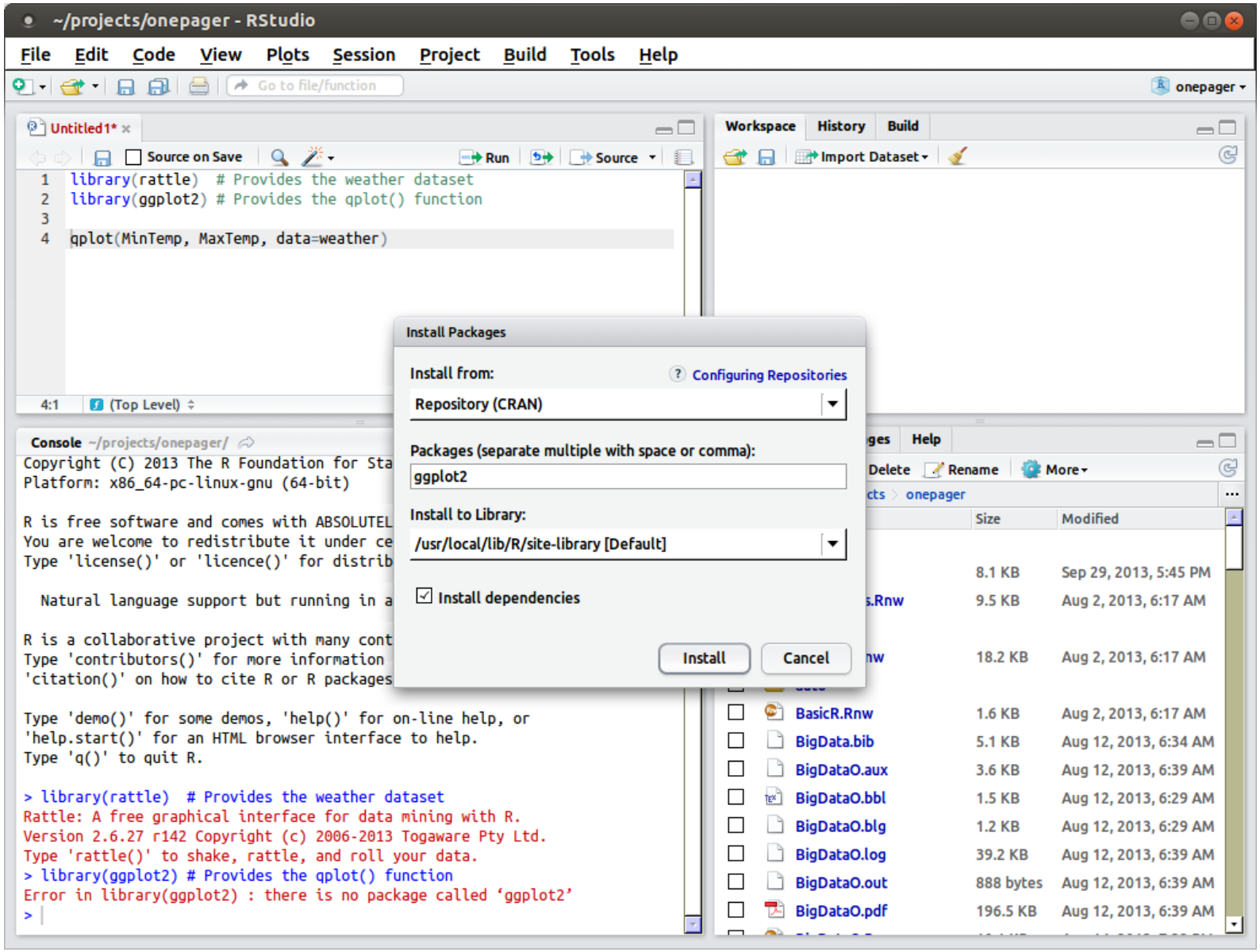
Scatterplot—Plot



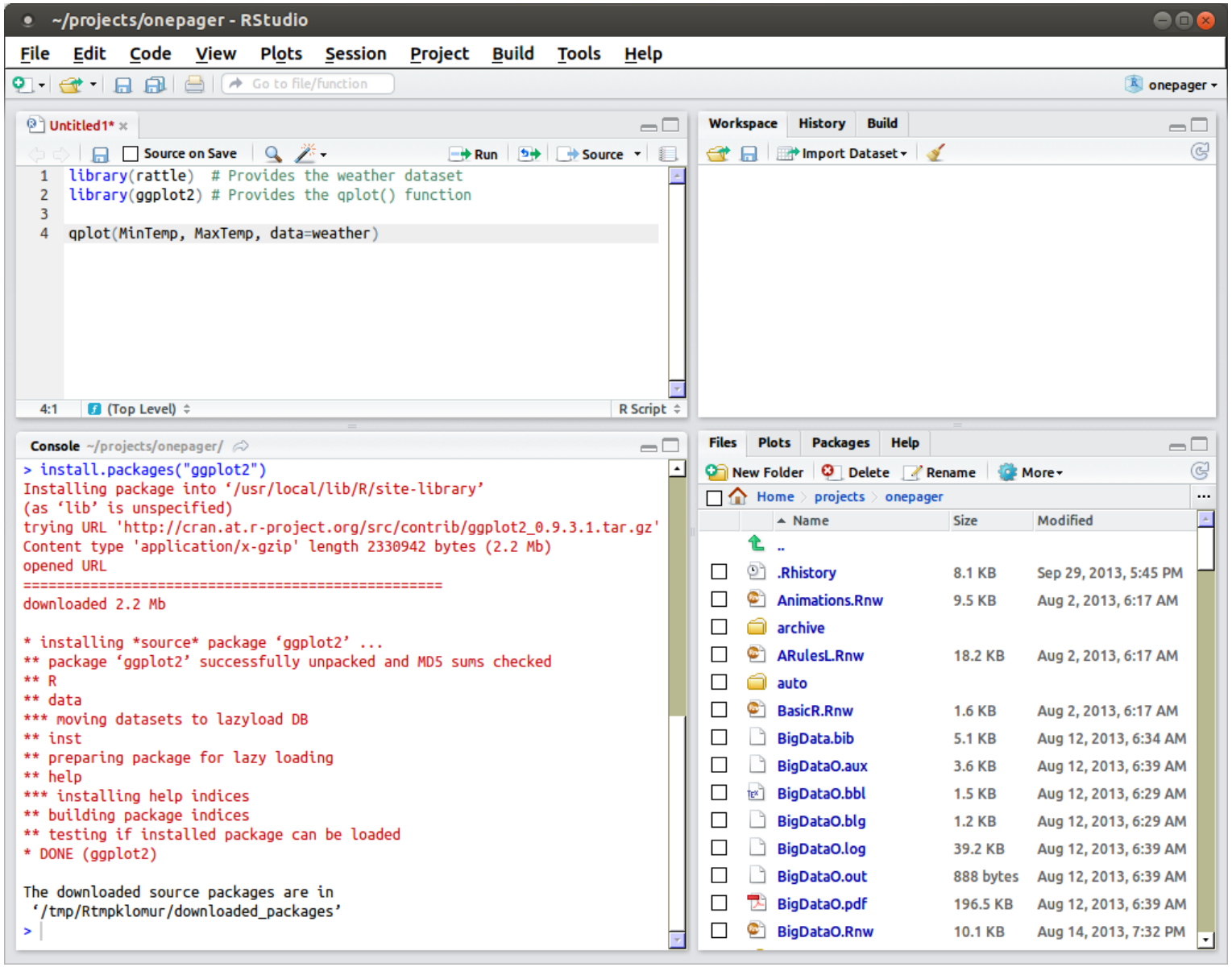
Scatterplot—RStudio



Missing Packages–Tools→Install Packages...



RStudio—Installing ggplot2



RStudio—Keyboard Shortcuts

These will become very useful!

- Editor:
  - Ctrl-Enter will send the line of code to the R console
  - Ctrl-2 will move the cursor to the Console
- Console:
  - UpArrow will cycle through previous commands
  - Ctrl-UpArrow will search previous commands
  - Tab will complete function names and list the arguments
  - Ctrl-1 will move the cursor to the Editor

Your turn: try them out.

RStudio—Keyboard Shortcuts

These will become very useful!

- Editor:
  - Ctrl-Enter will send the line of code to the R console
  - Ctrl-2 will move the cursor to the Console
- Console:
  - UpArrow will cycle through previous commands
  - Ctrl-UpArrow will search previous commands
  - Tab will complete function names and list the arguments
  - Ctrl-1 will move the cursor to the Editor

Your turn: try them out.

Basic R

```
library(rattle)    # Load the weather dataset.
head(weather)      # First 6 observations of the dataset.
```

```
##         Date Location MinTemp MaxTemp Rainfall Evapora...
## 1 2007-11-01 Canberra     8.0    24.3      0.0         ...
## 2 2007-11-02 Canberra    14.0    26.9      3.6         ...
## 3 2007-11-03 Canberra    13.7    23.4      3.6         ...
....
```

```
str(weather)       # Struncture of the variables in the dataset.
```

```
## 'data.frame': 366 obs. of  24 variables:
##  $ Date          : Date, format: "2007-11-01" "2007-11-...
##  $ Location      : Factor w/ 49 levels "Adelaide","Alba...
##  $ MinTemp       : num  8 14 13.7 13.3 7.6 6.2 6.1 8.3 ...
....
```

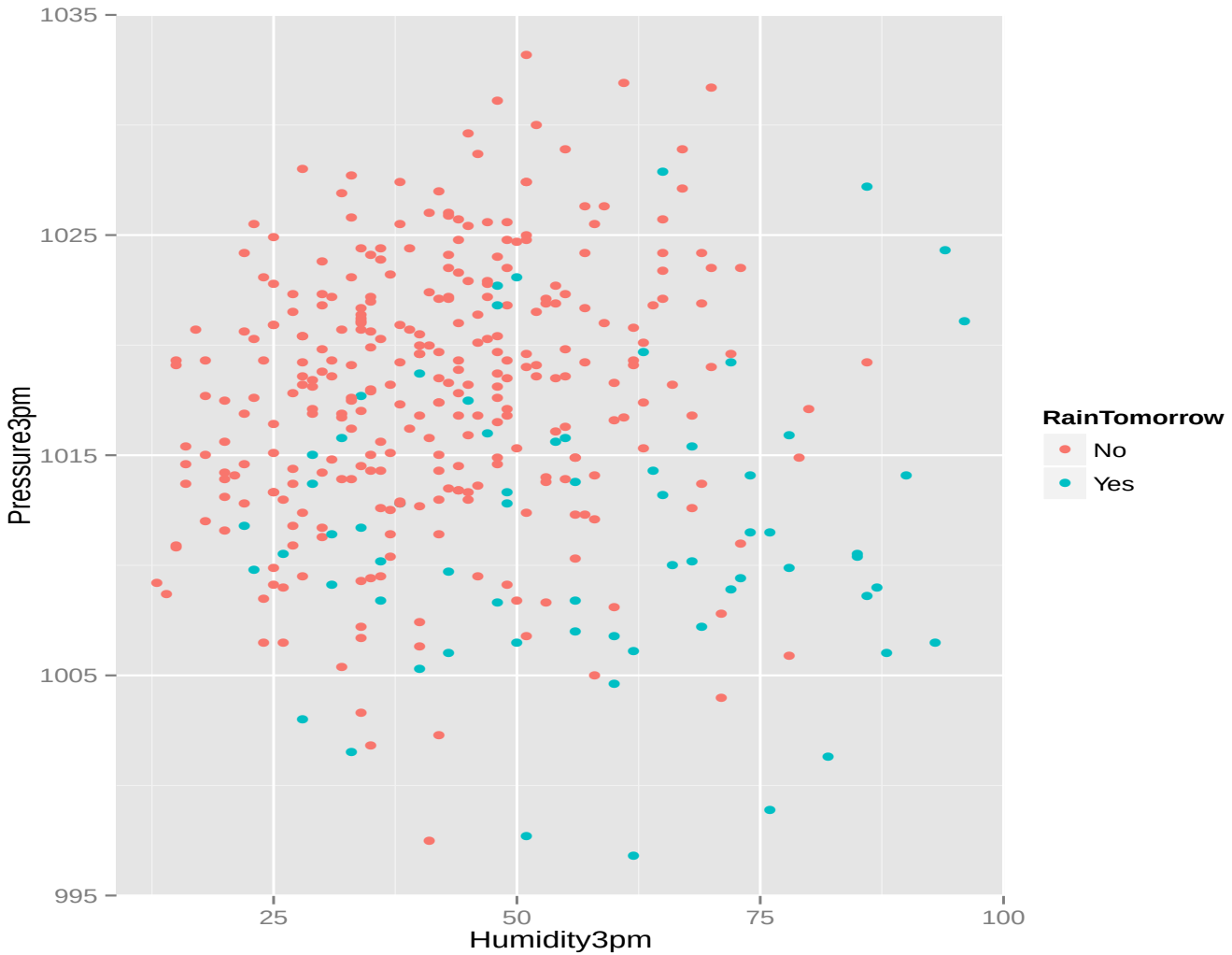
Basic R

```
summary(weather)  # Univariate summary of the variables.
```

```
##       Date                    Location      MinTemp          ...
##  Min.   :2007-11-01   Canberra     :366   Min.   :-5.30    ...
##  1st Qu.:2008-01-31   Adelaide     :  0   1st Qu.: 2.30    ...
##  Median :2008-05-01   Albany       :  0   Median : 7.45    ...
##  Mean   :2008-05-01   Albury       :  0   Mean   : 7.27    ...
##  3rd Qu.:2008-07-31   AliceSprings :  0   3rd Qu.:12.50    ...
##  Max.   :2008-10-31   BadgerysCreek:  0   Max.   :20.90    ...
##                       (Other)      :  0                    ...
##     Rainfall       Evaporation        Sunshine       WindGust...
##  Min.   : 0.00   Min.   : 0.20   Min.   : 0.00   NW     : ...
##  1st Qu.: 0.00   1st Qu.: 2.20   1st Qu.: 5.95   NNW    : ...
##  Median : 0.00   Median : 4.20   Median : 8.60   E      : ...
##  Mean   : 1.43   Mean   : 4.52   Mean   : 7.91   WNW    : ...
##  3rd Qu.: 0.20   3rd Qu.: 6.40   3rd Qu.:10.50   ENE    : ...
....
```

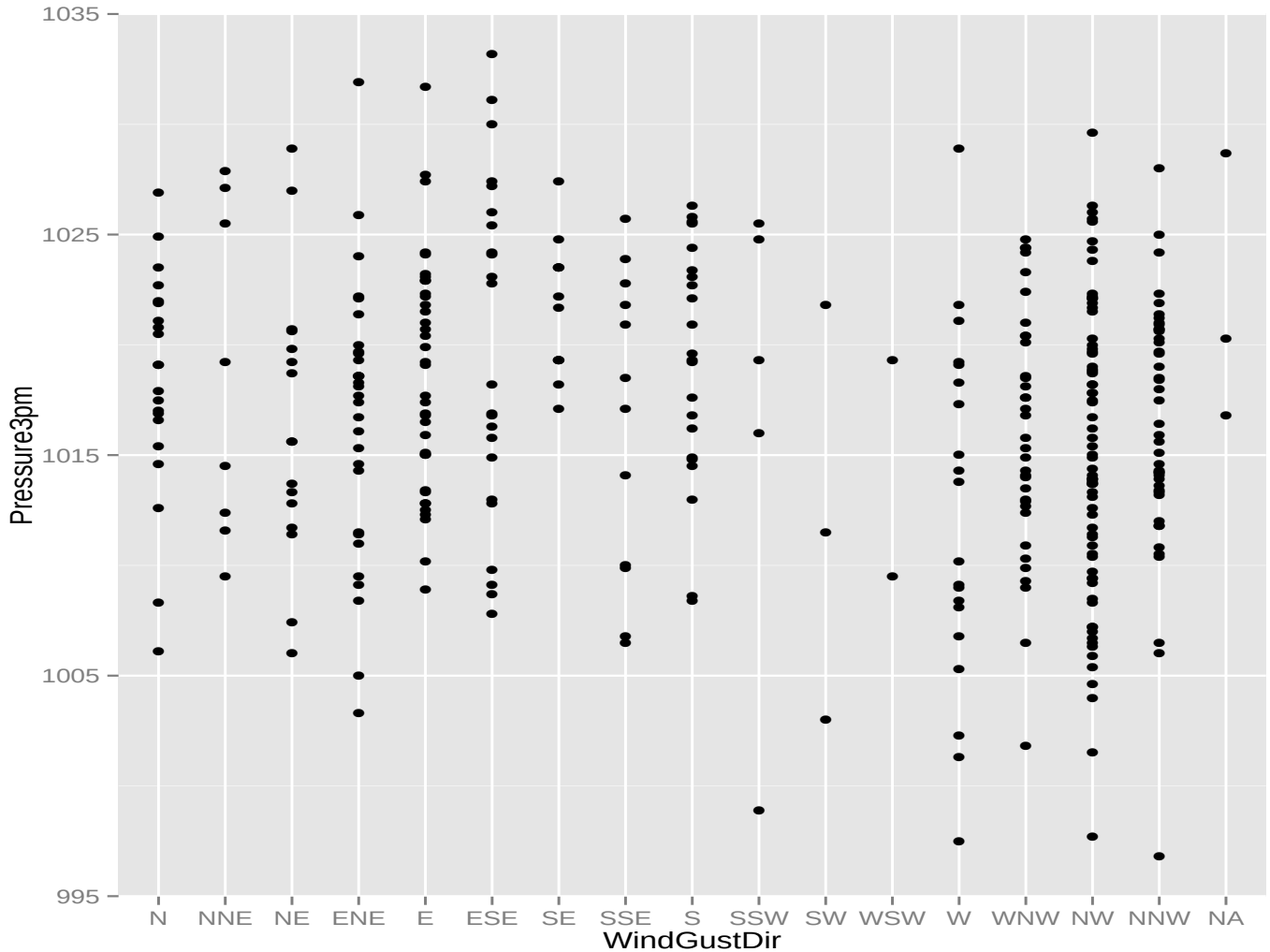
Visual Summaries—Add A Little Colour

```
qplot(Humidity3pm, Pressure3pm, colour=RainTomorrow, data=ds)
```



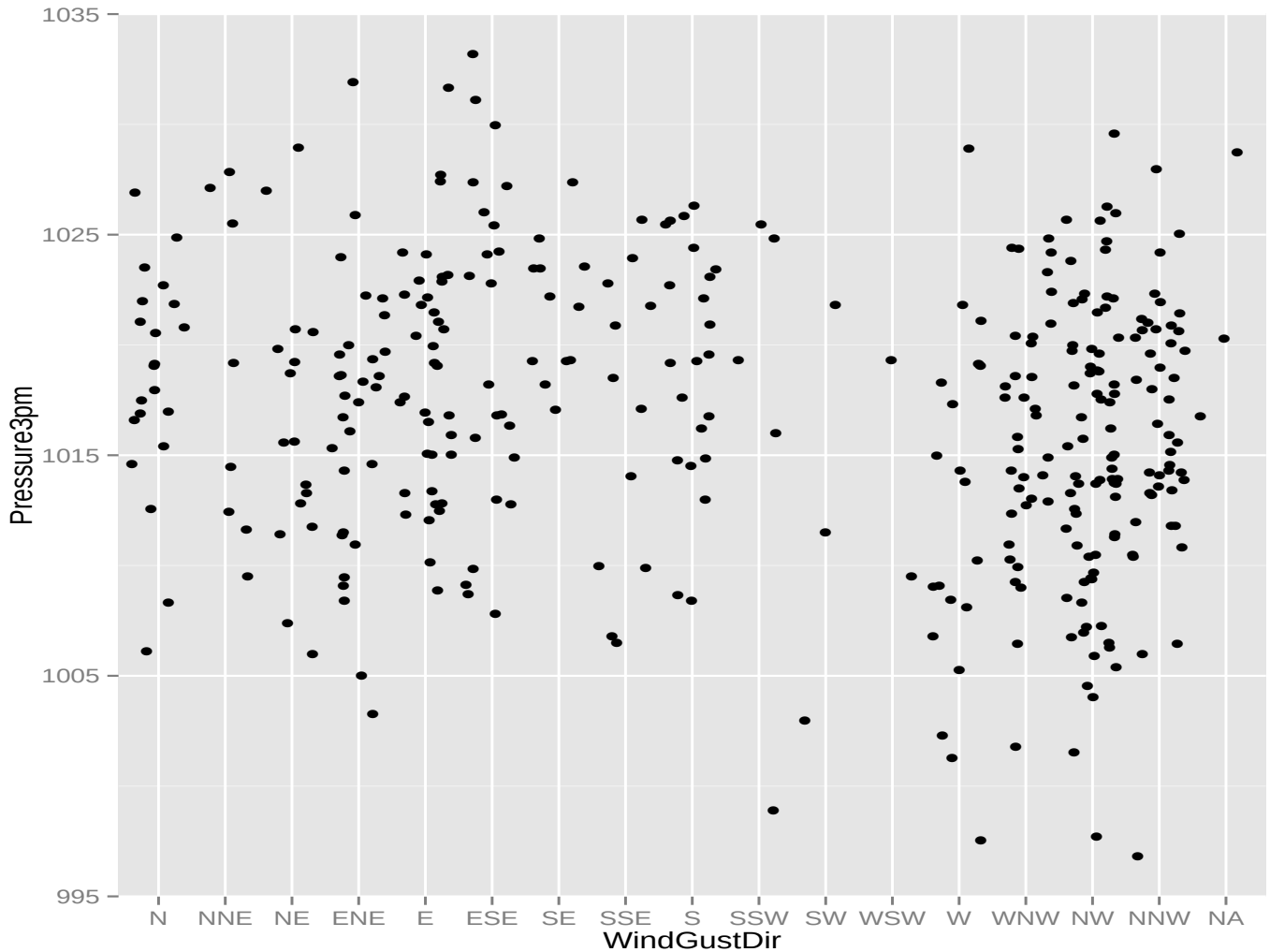
Visual Summaries—Careful with Categorics

```
qplot(WindGustDir, Pressure3pm, data=ds)
```



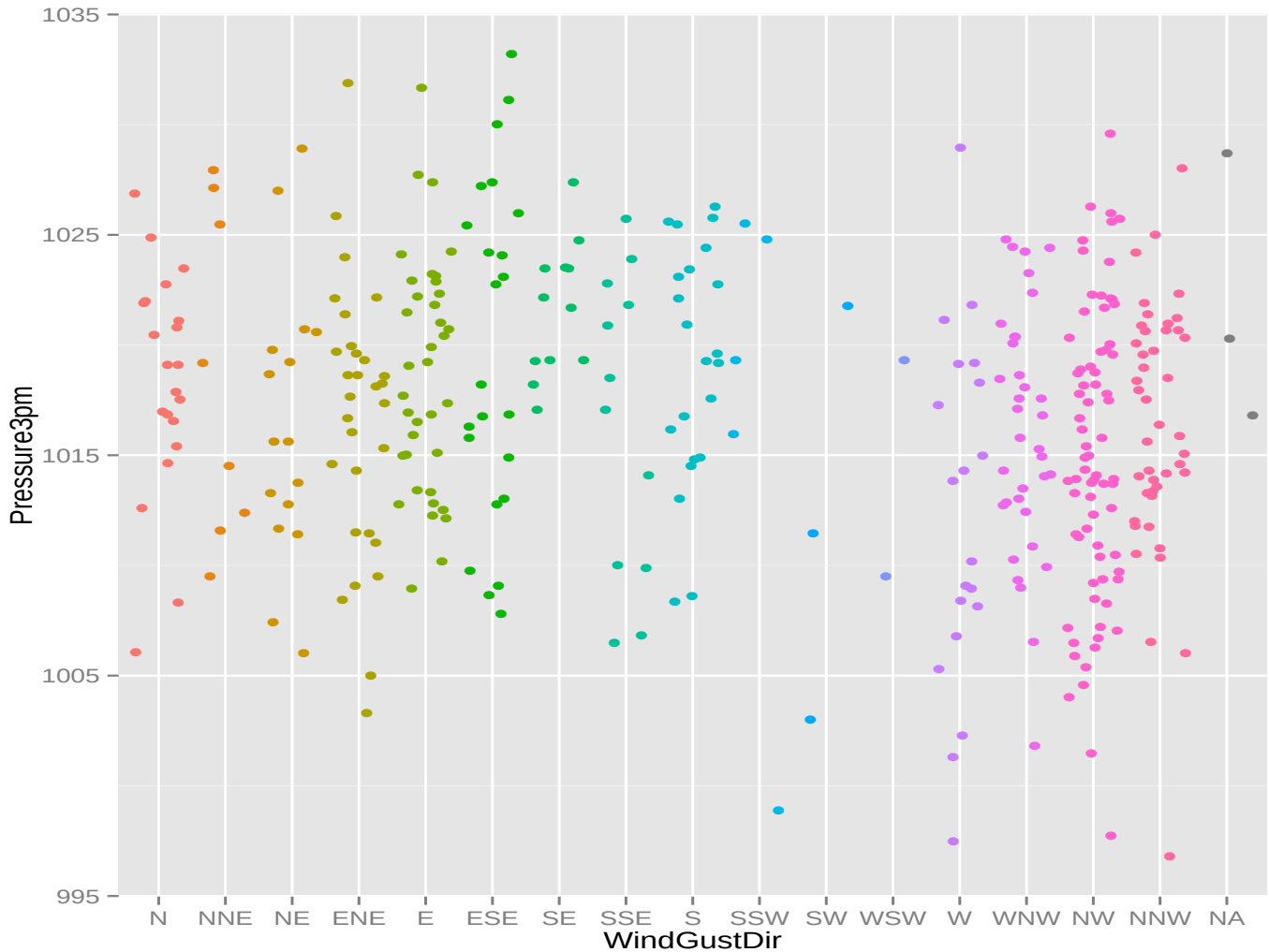
Visual Summaries—Add A Little Jitter

```
qplot(WindGustDir, Pressure3pm, data=ds, geom="jitter")
```

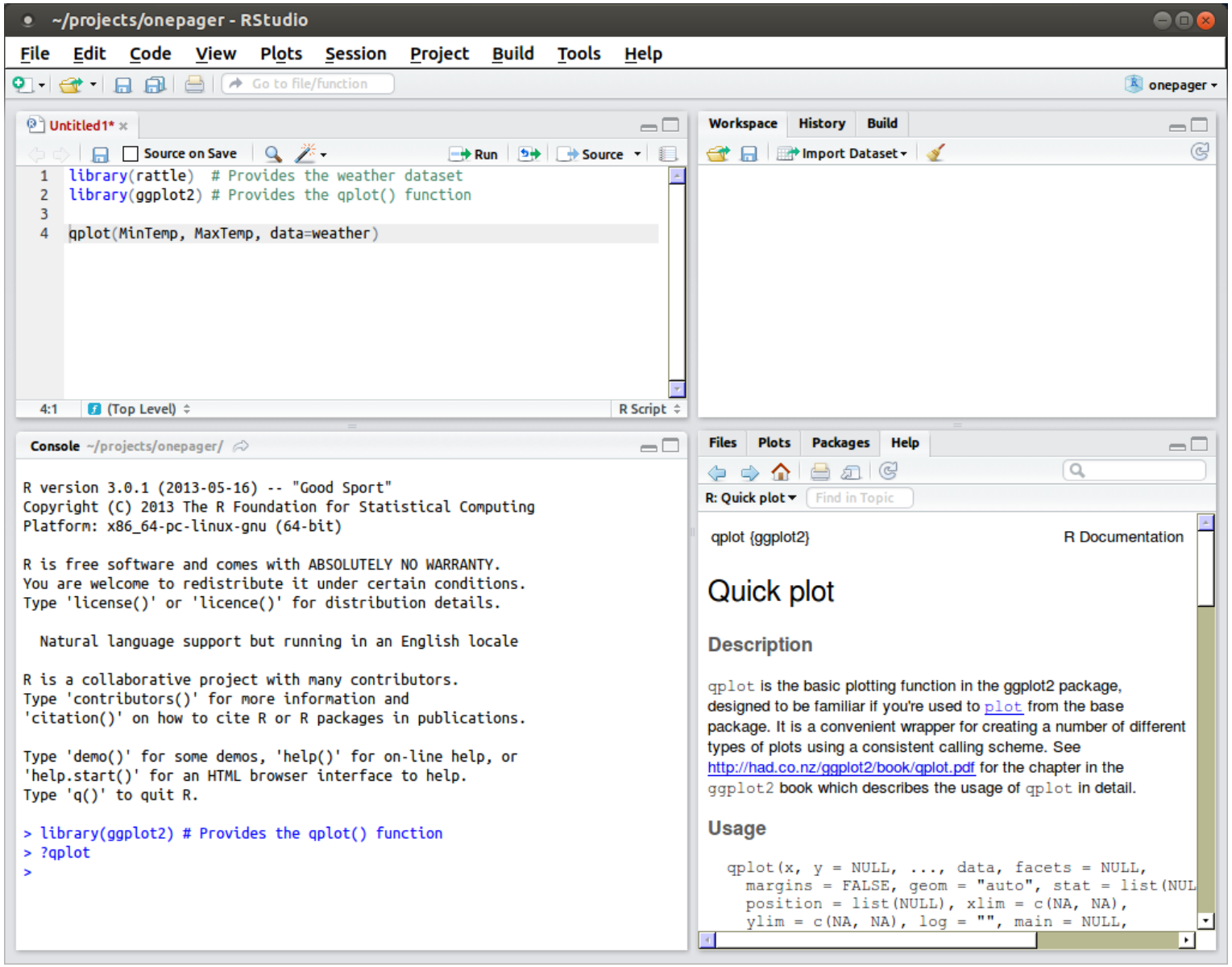


Visual Summaries—And Some Colour

```
qplot(WindGustDir, Pressure3pm, data=ds, colour=WindGustDir, geom="jitter")
```

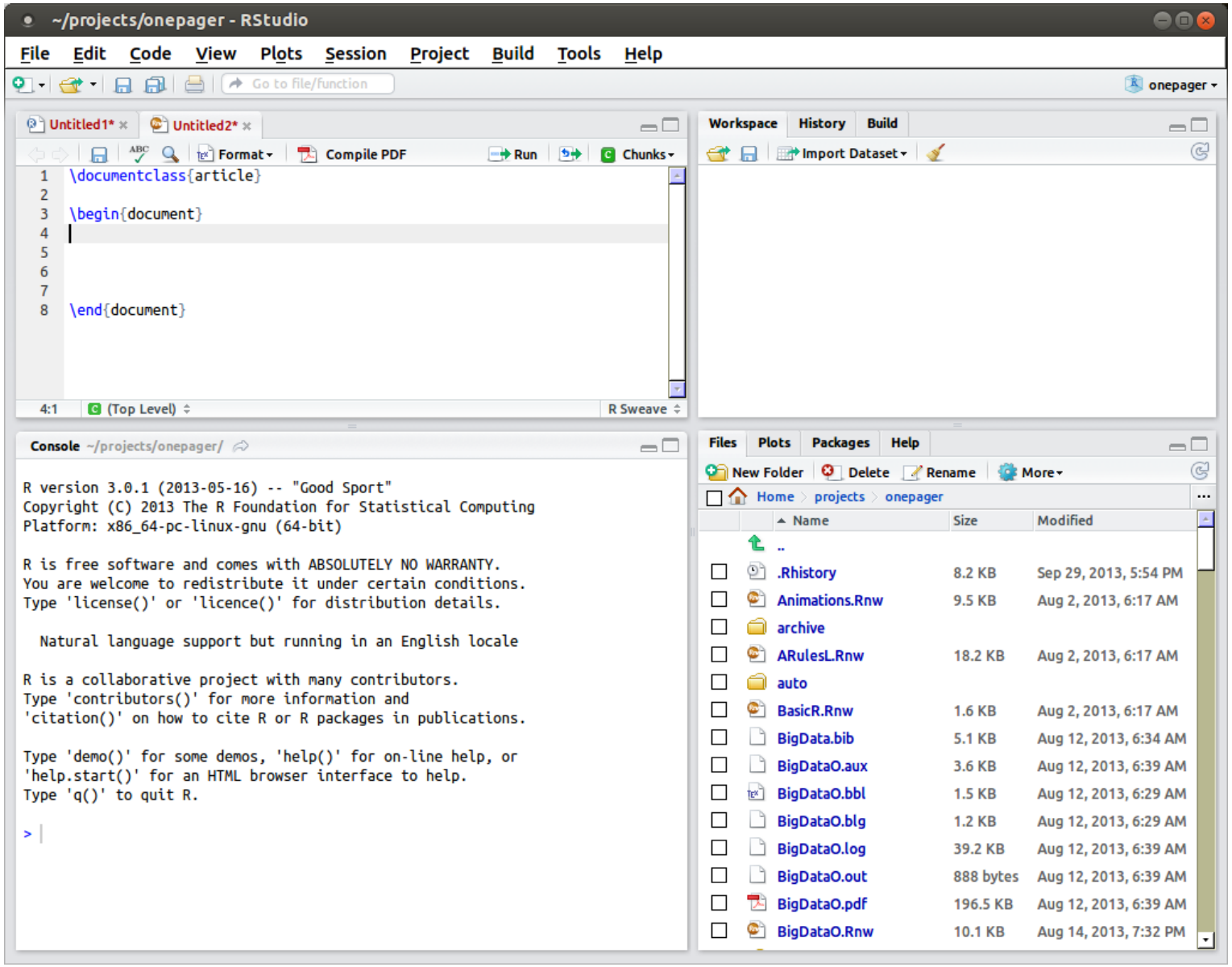


Getting Help—Precede Command with ?



Overview

Create a KnitR Document: New→R Sweave



Setup KnitR

We wish to use KnitR rather than the older Sweave processor

In RStudio we can configure the options to use knitr:

- Select Tools→Options
- Choose the Sweave group
- Choose **knitr** for *Weave Rnw files using:*
- The remaining defaults should be okay
- Click **Apply** and then **OK**

Simple KnitR Document

Insert the following into your new KnitR document:

```
\title{Sample KnitR Document}
\author{Graham Williams}
\maketitle

\section*{My First Section}

This is some text that is automatically typeset
by the LaTeX processor to produce well formatted
quality output as PDF.
```

Your turn—Click **Compile PDF** to view the result.

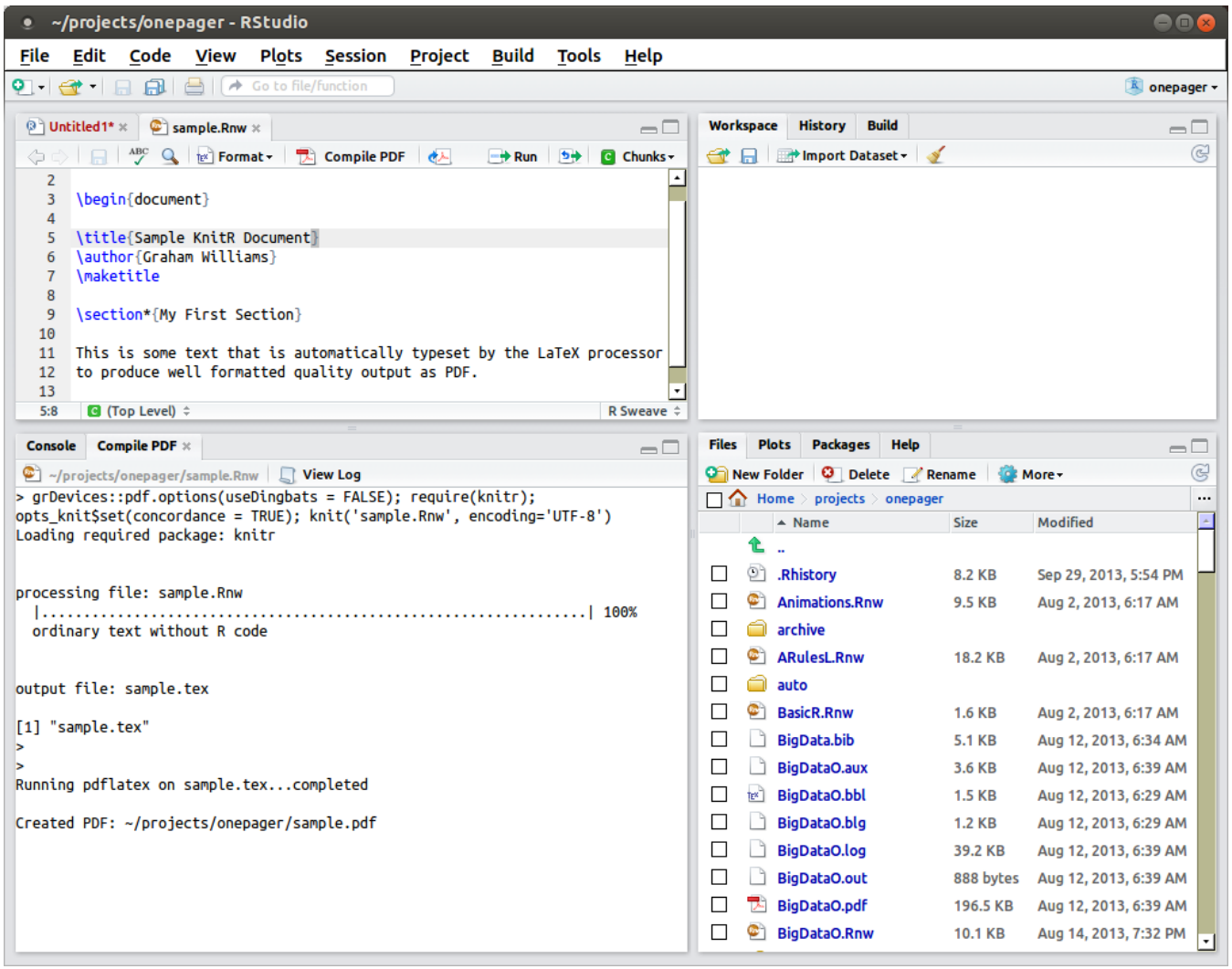
Simple KnitR Document

Insert the following into your new KnitR document:

```
\title{Sample KnitR Document}
\author{Graham Williams}
\maketitle

\section*{My First Section}

This is some text that is automatically typeset
by the LaTeX processor to produce well formatted
quality output as PDF.
```

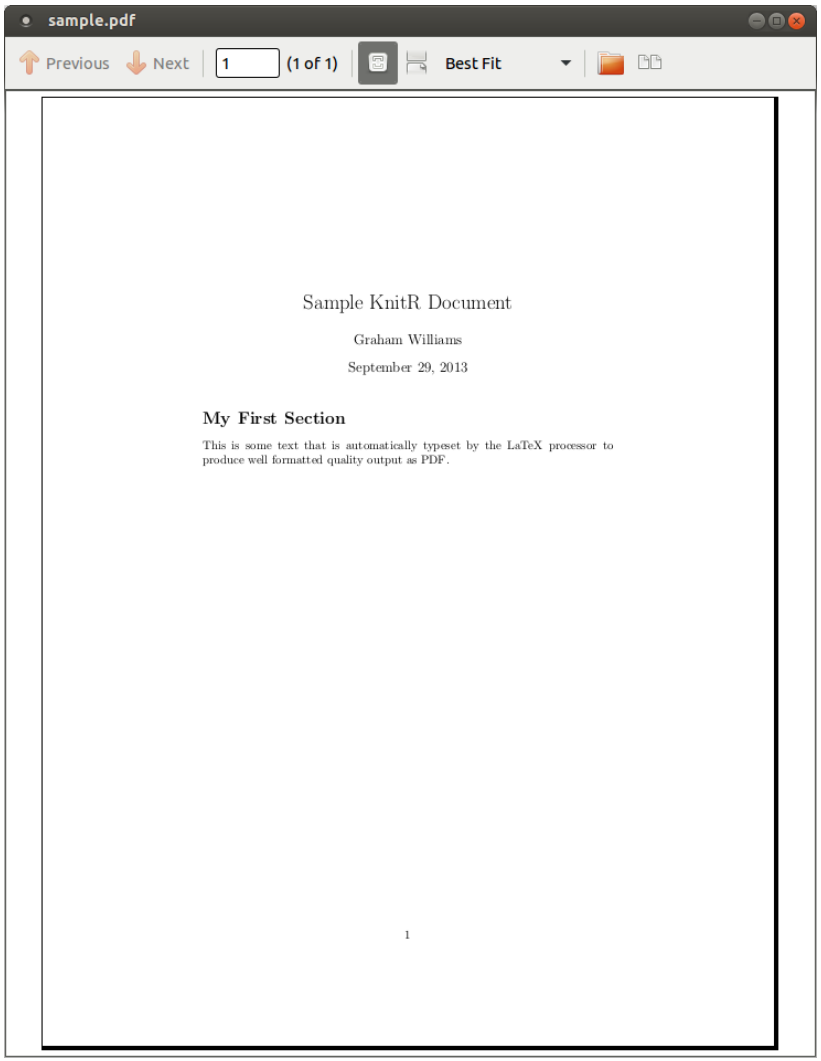
Your turn—Click **Compile PDF** to view the result.

Simple KnitR Document



Simple KnitR Document—Resulting PDF

Result of **Compile PDF**



KnitR: Add R Commands

R code can be used to generate results into the document:

```
<<echo=FALSE, message=FALSE>>=
library(rattle)  # Provides the weather dataset
library(ggplot2) # Provides the qplot() function

ds <- weather
qplot(MinTemp, MaxTemp, data=ds)
@
```

Your turn—Click **Compile PDF** to view the result.

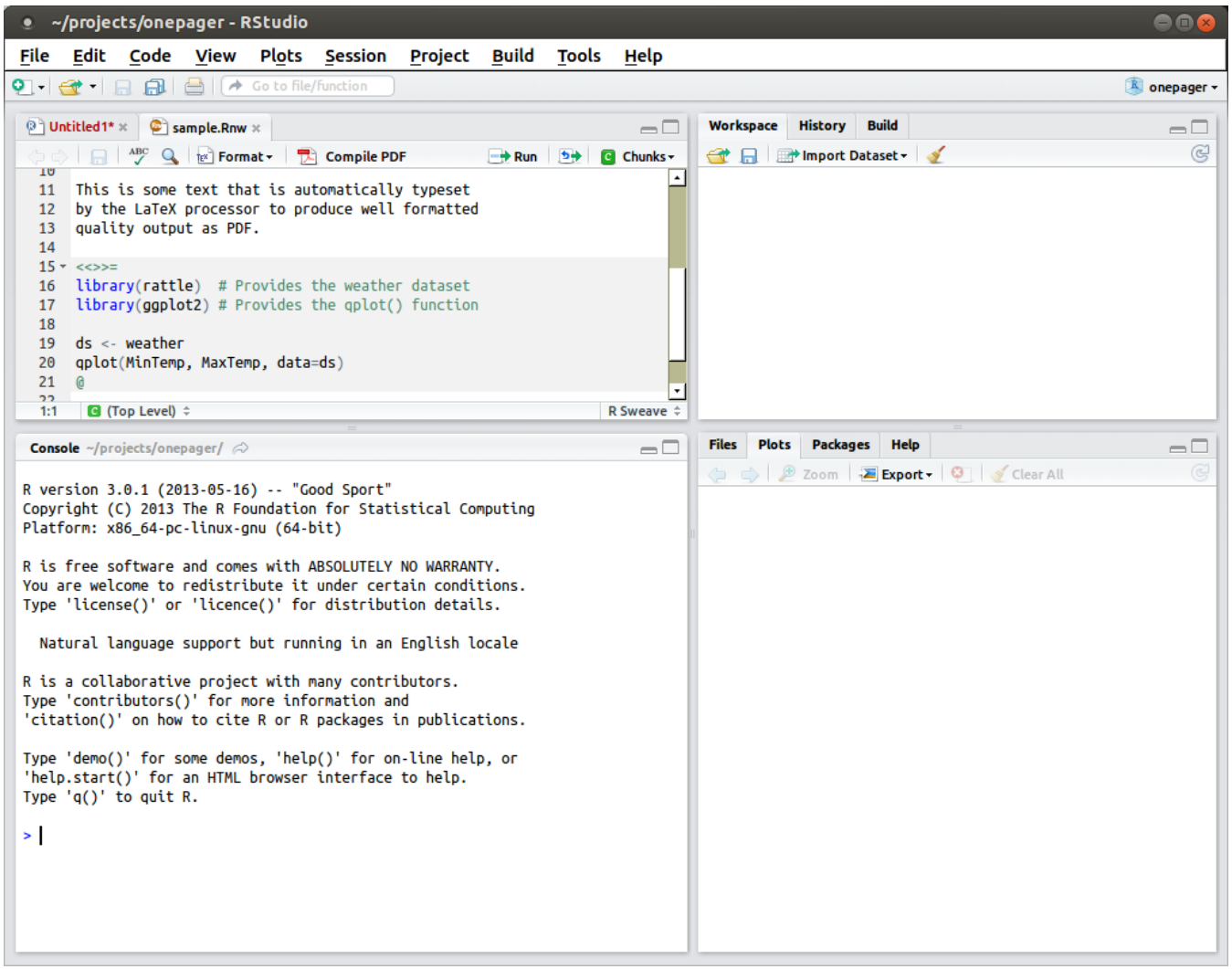
KnitR: Add R Commands

R code can be used to generate results into the document:

```
<<echo=FALSE, message=FALSE>>=
library(rattle)  # Provides the weather dataset
library(ggplot2) # Provides the qplot() function

ds <- weather
qplot(MinTemp, MaxTemp, data=ds)
@
```

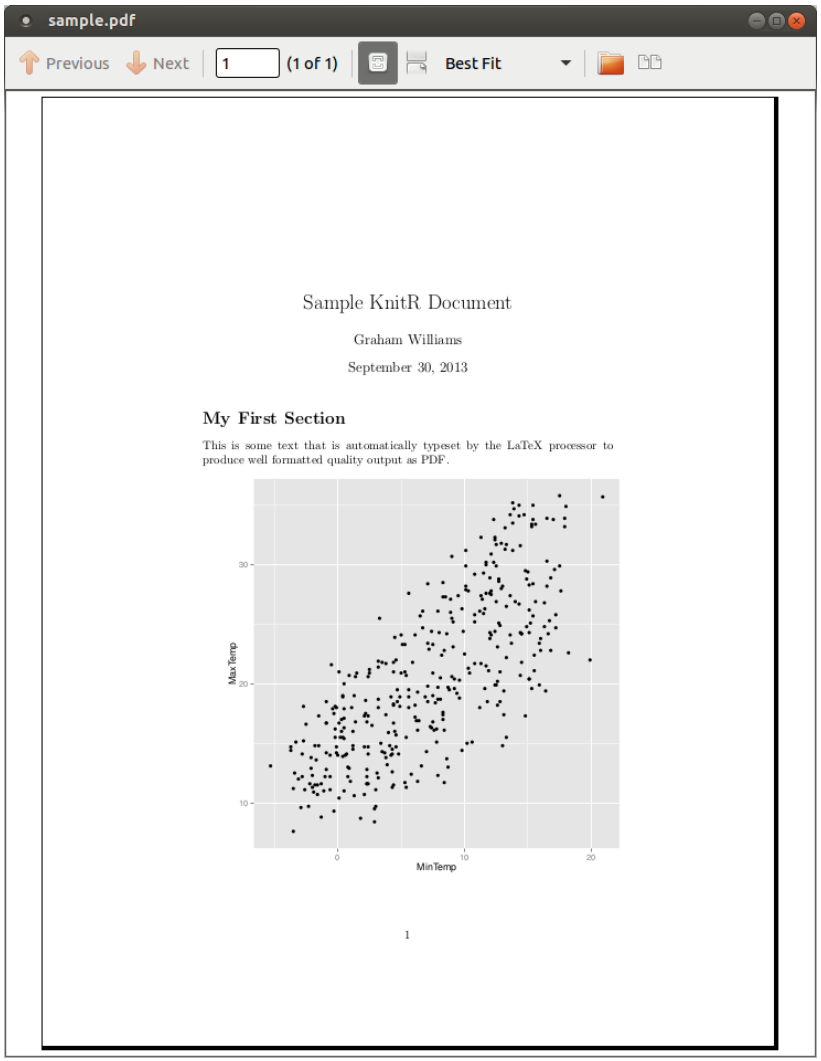
Your turn—Click **Compile PDF** to view the result.

KnitR Document With R Code



Simple KnitR Document—PDF with Plot

Result of **Compile PDF**



LaTeX Basics

```
\subsection*{...}          % Introduce a Sub Section

\subsubsection*{...}       % Introduce a Sub Sub Section

\textbf{...}               % Bold font
\textit{...}               % Italic font

\begin{itemize}            % A bullet list
  \item ...
  \item ...
\end{itemize}
```

Plus an extensive collection of other markup and capabilities.

KnitR Basics

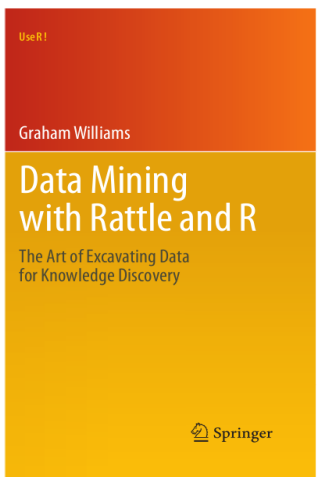
```
echo=FALSE          # Do not display the R code
eval=TRUE           # Evaluate the R code

results="hide"      # Hide the results of the R commands

fig.width=10        # Extend figure width from 7 to 10 inches
fig.height=8        # Extend figure height from 7 to 8 inches

out.width="0.8\\textwidth"     # Fit figure 80% page width
out.height="0.5\\textheight"   # Fit figure 50% page height
```

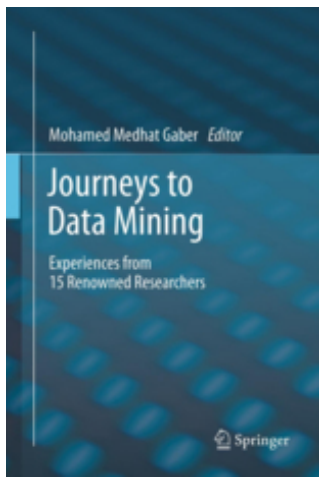
Plus an extensive collection of other options.

Resources and References

- **OnePageR**: `http://onepager.togaware.com` – Tutorial Notes
- Rattle: `http://rattle.togaware.com`
- Guides: `http://datamining.togaware.com`
- Practise: `http://analystfirst.com`

- Book: Data Mining using Rattle/R
- Chapter: Rattle and Other Tales
- Paper: A Data Mining GUI for R — R Journal, Volume 1(2)






Contributed Research Articles 45

Rattle: A Data Mining GUI for R

by Graham J Williams

Abstract Data mining delivers insights, patterns, and descriptive and predictive models from the large amounts of data available today in many organisations. The data miner draws heavily on methodologies, techniques and algorithms from statistics, machine learning, and computer science. R increasingly provides a powerful platform for data mining. However, scripting and programming is sometimes a challenge for data analysts moving into data mining. The Rattle package provides a graphical user interface specifically for data mining using R. It also provides a stepping stone toward using R as a programming language for data analysis.

Introduction

Data mining combines concepts, tools, and algorithms from machine learning and statistics for the analysis of very large datasets, so as to gain insights, understanding, and actionable knowledge.

Closed source data mining products have facilitated the uptake of data mining in many organisations. These products offer off-the-shelf ease-of-use that makes them attractive to the many new data miners in a market place desperately seeking high levels of analytical skills.

R is ideally suited to the many challenging tasks associated with data mining. R offers a breadth and depth in statistical computing beyond what is available in commercial closed source products. Yet R remains, primarily, a programming language for the highly skilled statistician, and out of the reach of many.

Rattle (the R Analytical Tool To Learn Easily) is a graphical data mining application written in and providing a pathway into R (Williams, 2009b). It has been developed specifically to ease the transition from basic data mining, as necessarily offered by GUIs, to sophisticated data analyses using a powerful statistical language.

Rattle brings together a multitude of R packages that are essential for the data miner but often not easy for the novice to use. An understanding of R is not required in order to get started with Rattle—this will gradually grow as we add sophistication to our data mining. Rattle's user interface provides an entree into the power of R as a data mining tool.

Rattle is used for teaching data mining at numerous universities and is in daily use by consultants and data mining teams world wide. It is also available as a product within Information Builders' WebFocus business intelligence suite as RStat.

Rattle is one of several open source data mining tools (Chen et al., 2007). Many of these tools are also directly available within R (and hence Rattle) through packages like **RWeka** (Hornik et al., 2009) and **arulos** (Hahsler et al., 2009).

Implementation

Rattle uses the Gnome graphical user interface as provided through the **RGtk2** package (Lawrence and Lang, 2008). It runs under various operating systems, including GNU/Linux, Macintosh OS/X, and MS/Windows.

The GUI itself has been developed using the Glade interactive interface builder. This produces a programming-language-independent XML description of the layout of the widgets that make up the user interface. The XML file is then simply loaded by an application and the GUI is rendered.

Glade allows the developer to freely choose to implement the functionality (i.e., the widget callbacks) in a programming language of choice, and for Rattle that is R. It is interesting to note that the first implementation of Rattle actually used Python for implementing the callbacks and R for the statistics, using **rpy**. The release of **RGtk2** allowed the interface elements of Rattle to be written directly in R so that Rattle is a fully R-based application.

Underneath, Rattle relies upon an extensive collection of R packages. This is a testament to the power of R—it delivers a breadth and depth of statistical analysis that is hard to find anywhere else. Some of the packages underlying Rattle include **ada**, **arules**, **doBy**, **ellipse**, **fBasics**, **fpc**, **gplots**, **Hmisc**, **kernlab**, **mice**, **network**, **party**, **playwith**, **pscl**, **randomForest**, **reshape**, **rggobi**, **RGtk2**, **ROCR**, **RODBC**, and **spart**. These packages are all available from the Comprehensive R Archive Network (CRAN). If a package is not installed but we ask through Rattle for some functionality provided by that package, Rattle will popup a message indicating that the package needs to be installed.

Rattle is not only an interface though. Additional functionality that is desired by a data miner has been written for use in Rattle, and is available from the **rattle** package without using the Rattle GUI. The **pmml** package (Guazzelli et al., 2009) is an offshoot of the development of Rattle and supports the export of models from Rattle using the open standard XML based PMML, or Predictive Model Markup Language (Data Mining Group, 2008). Models exported from R in this way can be imported into tools like the ADAPA decision engine running on cloud computers, Teradata's Warehouse Miner for deployment as SQL over a very large database, and Information Builder's WebFocus which handles

The R Journal Vol. 1/2, December 2009 ISSN 2073-4859

Lecture Summary

- Data Science—Analytics—Data Mining;
- Rattle as a GUI for Quickly Analysing Data;
- The Power is with R.

This document, sourced from StartL.Rnw revision 436, was processed by KnitR version 1.6 of 2014-05-24 and took 4.9 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 2014-06-21 20:26:17.