

DATA SCIENCE WITH R

INTRODUCING DATA MINING WITH RATTLE AND R

Graham.Williams@togaware.com

Senior Director and Chief Data Miner, Analytics
Australian Taxation Office

Visiting Professor, SIAT, Chinese Academy of Sciences
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

- 1 AN INTRODUCTION TO DATA MINING
- 2 THE RATTLE PACKAGE FOR DATA MINING
- 3 MOVING INTO R
- 4 GETTING STARTED WITH RATTLE

OVERVIEW

- 1 AN INTRODUCTION TO DATA MINING
- 2 THE RATTLE PACKAGE FOR DATA MINING
- 3 MOVING INTO R
- 4 GETTING STARTED WITH RATTLE

DATA MINING AND BIG DATA

- Application of
 - Machine Learning
 - Statistics
 - Software Engineering and Programming with Data
 - Intuition
- To Big Data — Volume, Velocity, Variety
- ...to discover new knowledge
- ...to improve business outcomes
- ...to deliver better tailored services

THE BUSINESS OF DATA MINING

- Australian Taxation Office
 - Lodgment (\$110M)
 - Tax Havens (\$150M)
 - Tax Fraud (\$250M)

- IBM Buys SPSS for \$1.2B in 2009
- SAS has annual revenue approaching \$3B
- Analytics is >\$100B business
- Amazon, eBay/PayPal, Google ...

BASIC TOOLS: DATA MINING ALGORITHMS

- Linear Discriminant Analysis (lda)
- Logistic Regression (glm)
- Decision Trees (rpart, wsrpart)
- Random Forests (randomForest, wsrfr)
- 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

- 1 AN INTRODUCTION TO DATA MINING
- 2 THE RATTLE PACKAGE FOR DATA MINING
- 3 MOVING INTO R
- 4 GETTING STARTED WITH RATTLE

WHY A GUI?

- 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

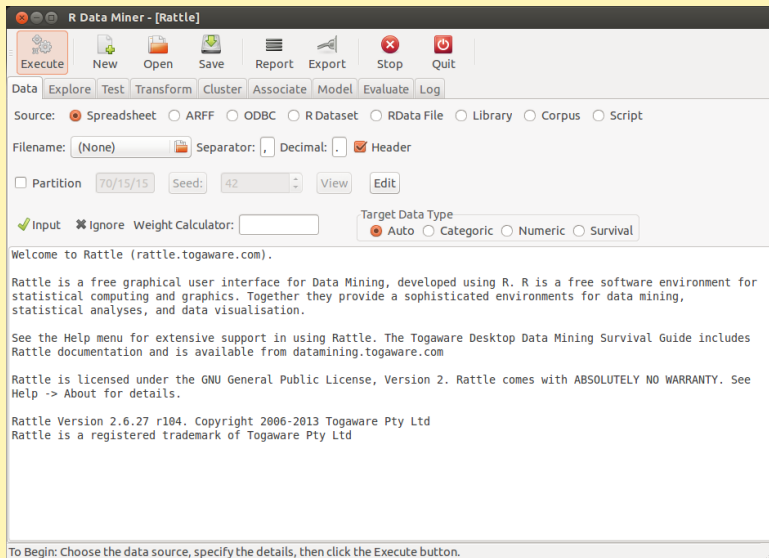
It is used by

- Consultants and Analytics Teams across business
- Universities to teach Data Mining

It is and will remain freely available.

CRAN and <http://rattle.togaware.com>

A TOUR THRU RATTLE: STARTUP



A TOUR THRU RATTLE: LOADING DATA

R Data Miner - [Rattle (weather.csv)]

Execute New Open Save Report Export Stop Quit

Data Explore Test Transform Cluster Associate Model Evaluate Log

Source: ☒ Spreadsheet ☐ ARFF ☐ ODBC ☐ R Dataset ☐ RData File ☐ Library ☐ Corpus ☐ Script

Filename: (None) Separator: , Decimal: . ☒ Header

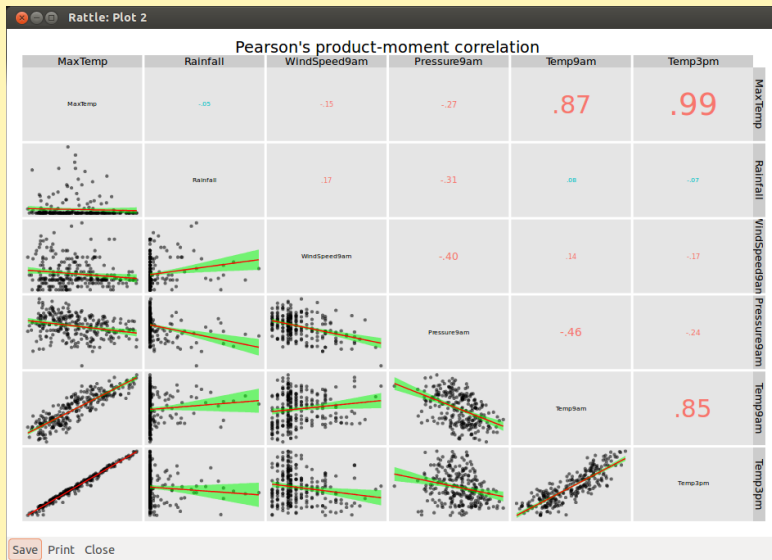
☒ Partition 70/15/15 Seed: 42 View Edit

☒ Input ☒ Ignore Weight Calculator: Target Data Type ☒ Auto ☐ Categorical ☐ Numeric ☐ Survival

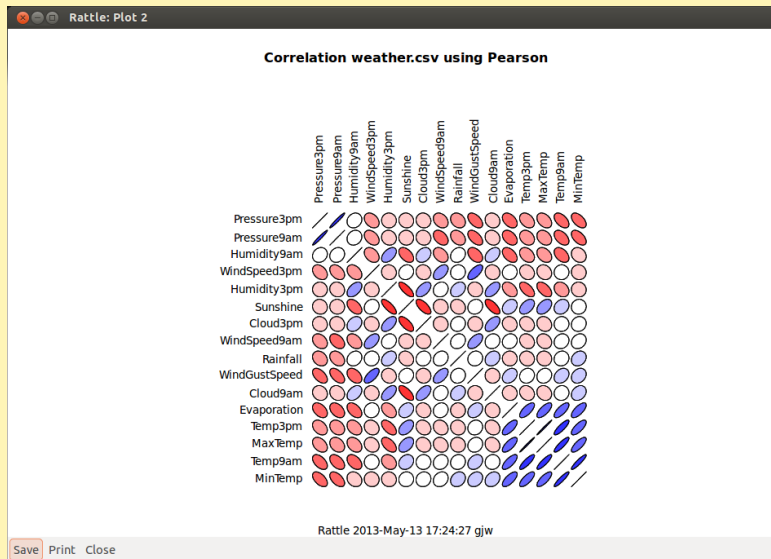
No.	Variable	Data Type	Input	Target	Risk	Ident	Ignore	Weight	Comment
1	Date	Ident	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 366
2	Location	Constant	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	Unique: 1
3	MinTemp	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 180
4	MaxTemp	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 187
5	Rainfall	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 47
6	Evaporation	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 55
7	Sunshine	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 114 Missing: 3
8	WindGustDir	Categorical	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 16 Missing: 3
9	WindGustSpeed	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 35 Missing: 2
10	WindDir9am	Categorical	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 16 Missing: 31
11	WindDir3pm	Categorical	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 16 Missing: 1
12	WindSpeed9am	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 22 Missing: 7
13	WindSpeed3pm	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 26
14	Humidity9am	Numeric	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	Unique: 60

Roles noted. 366 observations and 20 input variables. The target is RainTomorrow. Categorical 2. Classification models enabled.

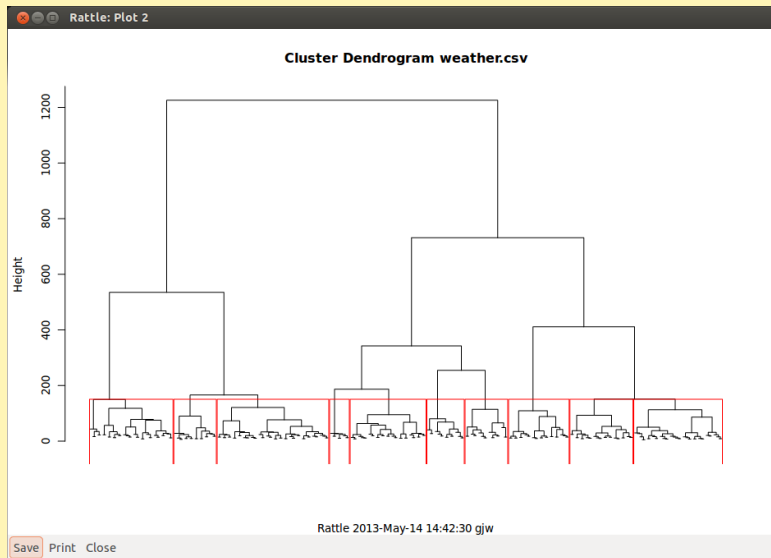
A TOUR THRU RATTLE: EXPLORE DISTRIBUTION



A TOUR THRU RATTLE: EXPLORE CORRELATIONS



A TOUR THRU RATTLE: HIERARCHICAL CLUSTER



A TOUR THRU RATTLE: DECISION TREE

R Data Miner - [Rattle (weather.csv)]

Buttons: Execute, New, Open, Save, Report, Export, Stop, Quit

Tabs: Data, Explore, Test, Transform, Cluster, Associate, **Model**, Evaluate, Log

Type: ☒ Tree ☐ Forest ☐ Boost ☐ SVM ☐ Linear ☐ Neural Net ☐ Survival ☐ All

Target: RainTomorrow Algorithm: ☒ Traditional ☐ Conditional Model Builder: rpart

Min Split: 20 Max Depth: 30 Priors: ☐ Include Missing

Min Bucket: 7 Complexity: 0.0100 Loss Matrix:

Summary of the Decision Tree model for Classification (built using 'rpart'):

n= 256

node), split, n, loss, yval, (yprob)
* denotes terminal node

- 1) root 256 41 No (0.83984375 0.16015625)
- 2) Pressure3pm>=1011.9 204 16 No (0.92156863 0.07843137)
 - 4) Cloud3pm< 7.5 195 10 No (0.94871795 0.05128205) *
 - 5) Cloud3pm>=7.5 9 3 Yes (0.33333333 0.66666667) *
- 3) Pressure3pm< 1011.9 52 25 No (0.51923077 0.48076923)
 - 6) Sunshine>=8.85 25 5 No (0.80000000 0.20000000) *
 - 7) Sunshine< 8.85 27 7 Yes (0.25925926 0.74074074) *

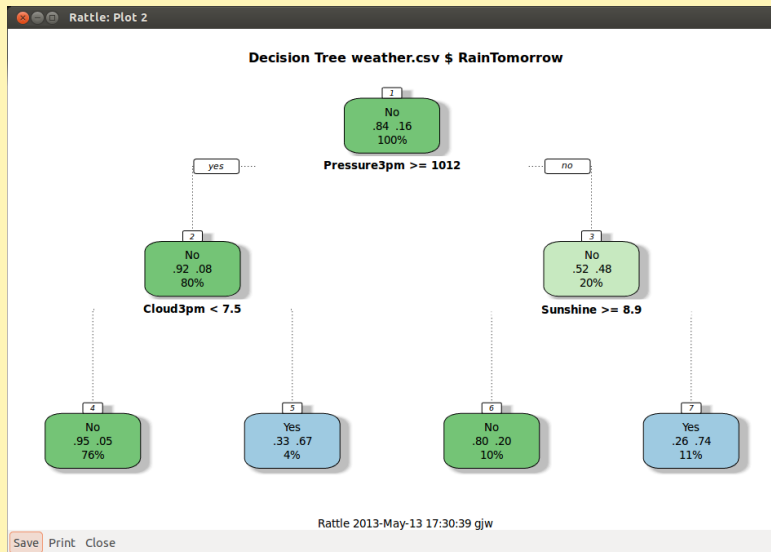
Classification tree:
rpart(formula = RainTomorrow ~ ., data = crs\$dataset[crs\$train,
c(crs\$input, crs\$target)], method = "class", parms = list(split = "information"),
control = rpart.control(usesurrogate = 0, maxsurrogate = 0))

Variables actually used in tree construction:
[1] Cloud3pm Pressure3pm Sunshine

Root node error: 41/256 = 0.16016

The Decision Tree model has been built. Time taken: 0.03 secs

A TOUR THRU RATTLE: DECISION TREE PLOT



A TOUR THRU RATTLE: RANDOM FOREST

R Data Miner - [Rattle (weather.csv)]

Execute New Open Save Report Export Stop Quit

Data Explore Test Transform Cluster Associate Model Evaluate Log

Type: ☐ Tree ☒ Forest ☐ Boost ☐ SVM ☐ Linear ☐ Neural Net ☐ Survival ☐ All

Target: RainTomorrow Algorithm: ☒ Traditional ☐ Conditional Model Builder: randomForest

Number of Trees: 500 Sample Size: Importance Rules 1

Number of Variables: 4 ☒ Impute Errors OOB ROC

Summary of the Random Forest Model

=====

Number of observations used to build the model: 256
Missing value imputation is active.

Call:
randomForest(formula = RainTomorrow ~ .,
data = crs\$dataset[crs\$sample, c(crs\$input, crs\$target)],
ntree = 500, mtry = 4, importance = TRUE, replace = FALSE, na.action = na.roughfix)

Type of random forest: classification
Number of trees: 500
No. of variables tried at each split: 4

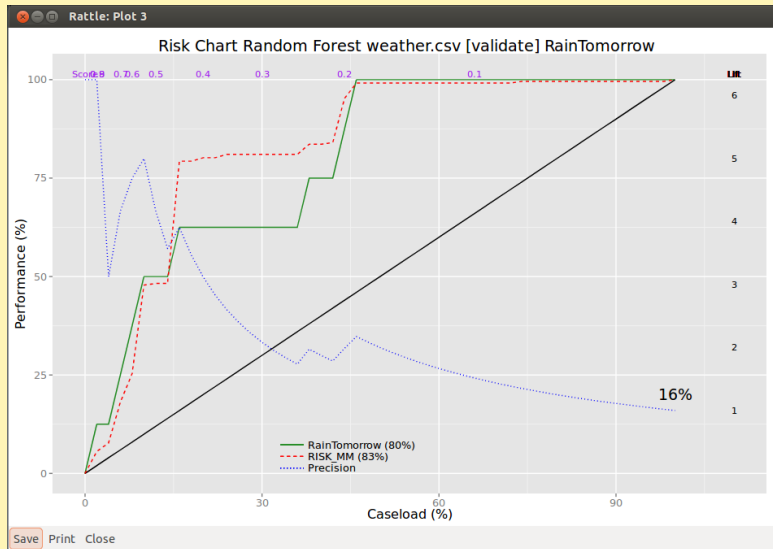
OOB estimate of error rate: 13.28%

Confusion matrix:
No Yes class.error
No 207 8 0.0372093
Yes 26 15 0.6341463

Analysis of the Area Under the Curve (AUC)
=====

The Random Forest model has been built. Time taken: 0.87 secs

A TOUR THRU RATTLE: RISK CHART



OVERVIEW

- 1 AN INTRODUCTION TO DATA MINING
- 2 THE RATTLE PACKAGE FOR DATA MINING
- 3 MOVING INTO R**
- 4 GETTING STARTED WITH RATTLE

DATA MINERS ARE PROGRAMMERS OF DATA

- Data miners are programmers of data
 - A GUI can only do so much
 - R is a powerful statistical language
-
- Professional data mining
 - Scripting
 - Transparency
 - Repeatability

FROM GUI TO CLI — RATTLE'S LOG TAB

```

R Data Miner - [Rattle (weather.csv)]

Execute New Open Save Report Export Stop Quit

Data Explore Test Transform Cluster Associate Model Evaluate Log

☒ Export Comments ☐ Rename Internal Variables: From crs$ to MY

# Rattle is Copyright (c) 2006-2013 Togaware Pty Ltd.

#=====
# Rattle timestamp: 2013-05-13 16:49:53 x86_64-pc-linux-gnu

# Rattle version 2.6.27 user 'gju'

# Export this log textview to a file using the Export button or the Tools
# menu to save a log of all activity. This facilitates repeatability. Exporting
# to file 'myrf01.R', for example, allows us to the type in the R Console
# the command source('myrf01.R') to repeat the process automatically.
# Generally, we may want to edit the file to suit our needs. We can also directly
# edit this current log textview to record additional information before exporting.

# Saving and loading projects also retains this log.

library(rattle)

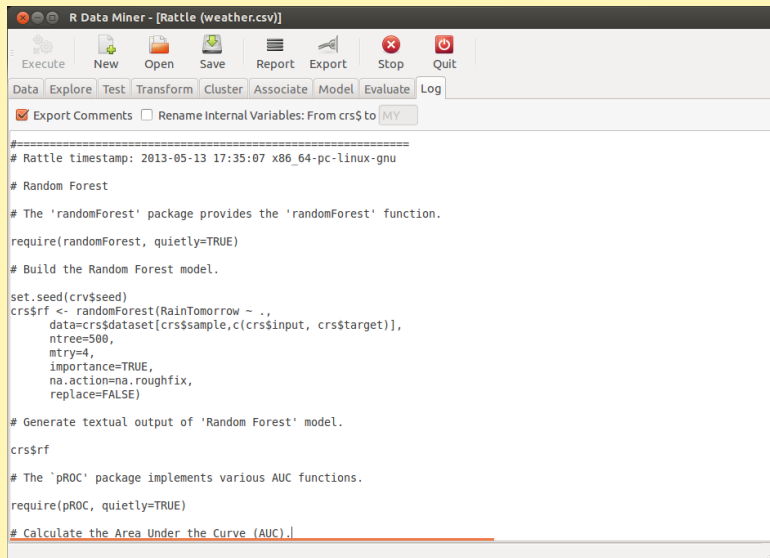
# This log generally records the process of building a model. However, with very
# little effort the log can be used to score a new dataset. The logical variable
# 'building' is used to toggle between generating transformations, as when building
# a model, and simply using the transformations, as when scoring a dataset.

building <- TRUE
scoring <- ! building

# The colorspace package is used to generate the colours used in plots, if available.

library(colorspace)
  
```

FROM GUI TO CLI — RATTLE'S LOG TAB



The screenshot shows the Rattle GUI window titled "R Data Miner - [Rattle (weather.csv)]". The "Log" tab is selected, displaying the following R code:

```

=====
# Rattle timestamp: 2013-05-13 17:35:07 x86_64-pc-linux-gnu

# Random Forest

# The 'randomForest' package provides the 'randomForest' function.

require(randomForest, quietly=TRUE)

# Build the Random Forest model.

set.seed(crv$seed)
crs$rfr <- randomForest(RainTomorrow ~ .,
  data=crs$dataset[crs$sample,c(crs$input, crs$target)],
  ntree=500,
  mtry=4,
  importance=TRUE,
  na.action=na.roughfix,
  replace=FALSE)

# Generate textual output of 'Random Forest' model.

crs$rfr

# The 'pROC' package implements various AUC functions.

require(pROC, quietly=TRUE)

# Calculate the Area Under the Curve (AUC).|

```

STEP 1: IDENTIFY THE DATA

```
dsname <- "weather"
target <- "RainTomorrow"
risk    <- "RISK_MM"
form    <- formula(paste(target, "~ ."))
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

```
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        ...
## 4 2007-11-04 Canberra     13.3     15.5       39.8        ...
## 5 2007-11-05 Canberra      7.6     16.1        2.8        ...
....
```

```
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        ...
## 364 2008-10-29 Canberra     12.5     19.9         0        ...
## 365 2008-10-30 Canberra     12.5     26.9         0        ...
....
```


STEP 2: OBSERVE THE DATA

```
str(ds)
```

```
## 'data.frame': 366 obs. of  24 variables:
##  $ Date          : Date, format: "2007-11-01" "2007-11-..."
##  $ Location       : Factor w/ 46 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...
....
```

```
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...
....
```

STEP 3: CLEAN THE DATA — IDENTIFY VARIABLES

```
(ignore <- c(names(ds)[c(1,2)], 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"   "...
....

dim(ds[vars])

## [1] 366 21
```

STEP 3: CLEAN THE DATA — REMOVE MISSING

```
dim(ds[vars])  
  
## [1] 366 21  
  
sum(is.na(ds[vars]))  
  
## [1] 47  
  
ds <- na.omit(ds[vars])  
sum(is.na(ds))  
  
## [1] 0  
  
dim(ds)  
  
## [1] 328 21
```

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

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

STEP 4: BUILD THE MODEL—TRAIN/TEST

```
(n <- nrow(ds))  
  
## [1] 328  
  
train <- sample(1:n, 0.70*n)  
length(train)  
  
## [1] 229  
  
test  <- setdiff(1:n, train)  
length(test)  
  
## [1] 99
```

STEP 4: BUILD THE MODEL—RANDOM FOREST

```
library(randomForest)

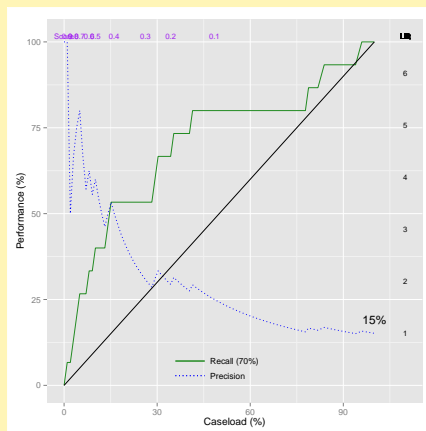
## randomForest 4.6-7
## Type rfNews() to see new features/changes/bug fixes.

m <- randomForest(form, ds[train,])
m

##
## Call:
## randomForest(formula=form, data=ds[train, ])
##               Type of random forest: classification
##               Number of trees: 500
##               No. of variables tried at each split: 4
##
##               OOB estimate of  error rate: 12.23%
## Confusion matrix:
.....
```

STEP 5: EVALUATE THE MODEL—RISK CHART

```
pr <- predict(m, ds[test,], type="prob")[,2]
ev <- evaluateRisk(pr, ds[test, target], ds[test, risk])
riskchart(ev)
```



OVERVIEW

- 1 AN INTRODUCTION TO DATA MINING
- 2 THE RATTLE PACKAGE FOR DATA MINING
- 3 MOVING INTO R
- 4 GETTING STARTED WITH RATTLE

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")
```

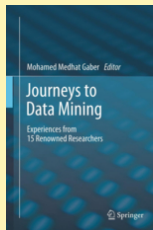
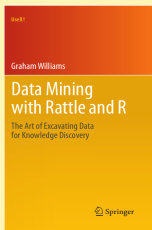
- Then we can start up Rattle:

```
rattle()
```

- Required packages are loaded as needed.

RESOURCES AND REFERENCES

- Rattle: <http://rattle.togaware.com>
- **OnePageR**: <http://onepager.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)



Thank You