The concept of building multiple decision trees to produce a better model can be dated back to the concept of Multiple Inductive Learning or the MIL algorithm (Williams, 1988). An ensemble of trees was found to produce a more accurate model than a single tree.

In this module we explore the use of ada (Culp et al., 2012) and randomForest (Breiman et al., 2012), as well as two newer packages wsrpart (Zhalama and Williams, 2014) and wsr (Meng et al., 2014).

The required packages for this module include:

```r
library(rattle)  # The weather dataset.
library(ada)     # Build boosted trees model with ada().
library(randomForest) # Impute missing values with na.roughfix().
library(wsrpart) # Weighted subspace using RPart.
library(wsr)     # Weighted subspace implemented in Cpp.
library(party)   # Conditional random forest cforest().
```

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 Prepare Weather Data for Modelling

See the separate Data and Model modules for template for preparing data and building models. In brief, we set ourselves up for modelling the weather dataset with the following commands, extending the simpler example we have just seen.

```r
set.seed(1426)
library(rattle)
data(weather)
dsname <- "weather"
ds <- get(dsname)
id <- c("Date", "Location")
target <- "RainTomorrow"
risk <- "RISK_MM"
ignore <- c(id, if (exists("risk")) risk)
(vars <- setdiff(names(ds), ignore))

## [1] "MinTemp"  "MaxTemp"  "Rainfall"  "Evaporation"
## [5] "Sunshine" "WindGustDir" "WindGustSpeed" "WindDir9am"
## [9] "WindDir3pm" "WindSpeed9am" "WindSpeed3pm" "Humidity9am"
## [13] "Humidity3pm" "Pressure9am" "Pressure3pm" "Cloud9am"

....
inputs <- setdiff(vars, target)
ds[vars] <- na.roughfix(ds[vars])  # Impute missing values, roughly.
(nobs <- nrow(ds))

## [1] 366
(numerics <- intersect(inputs, names(ds)[which(sapply(ds[vars], is.numeric))]))

## [1] "MinTemp"  "MaxTemp"  "Rainfall"  "Sunshine"
## [5] "WindDir9am" "WindDir3pm" "WindSpeed9am" "WindSpeed3pm"
## [9] "Humidity9am" "Humidity3pm" "Pressure9am" "Pressure3pm"
## [13] "Cloud9am"  "Cloud3pm"

....
(categorics <- intersect(inputs, names(ds)[which(sapply(ds[vars], is.factor))]))

## [1] "Evaporation" "WindGustDir" "WindGustSpeed" "Temp9am"
## [5] "Temp3pm"
(form <- formula(paste(target, " ~ .")))

## RainTomorrow ~ .

length(train <- sample(nobs, 0.7*nobs))

## [1] 256
length(test <- setdiff(seq_len(nobs), train))

## [1] 110
actual <- ds[test, target]
```
2 Review the Dataset

It is always a good idea to review the data.

```r
dim(ds)
## [1] 366 24
names(ds)
## [1] "Date"     "Location" "MinTemp" "MaxTemp"
## [5] "Rainfall" "Evaporation" "Sunshine" "WindGustDir"
## [9] "WindGustSpeed" "WindDir9am" "WindDir3pm" "WindSpeed9am"
## [13] "WindSpeed3pm" "Humidity9am" "Humidity3pm" "Pressure9am"
## [17] "Pressure3pm" "Cloud9am" "Cloud3pm" "Temp9am"
...
head(ds)
## Date Location MinTemp MaxTemp Rainfall Evaporation Sunshine
## 1 2007-11-01 Canberra 8.0 24.3 0.0 3.4 6.3
## 2 2007-11-02 Canberra 14.0 26.9 3.6 4.4 9.7
## 3 2007-11-03 Canberra 13.7 23.4 3.6 5.8 3.3
## 4 2007-11-04 Canberra 13.3 15.5 39.8 7.2 9.1
## ...
...
tail(ds)
## Date Location MinTemp MaxTemp Rainfall Evaporation Sunshine
## 361 2008-10-26 Canberra 7.9 26.1 0 6.8 3.5
## 362 2008-10-27 Canberra 9.0 30.7 0 7.6 12.1
## 363 2008-10-28 Canberra 7.1 28.4 0 11.6 12.7
## 364 2008-10-29 Canberra 12.5 19.9 0 8.4 5.3
## ...
str(ds)
## 'data.frame': 366 obs. of 24 variables:
## $ Date     : Date, format: "2007-11-01" "2007-11-02" ...
## $ Location : Factor w/ 49 levels "Adelaide","Albany",...: 10 10 10 10 ...
## $ MinTemp  : num 8 14 13.7 13.3 7.6 6.2 6.1 8.3 8.8 8.4 ...
## $ MaxTemp  : num 24.3 26.9 23.4 15.5 16.1 16.9 16.2 17 19.5 22.8 ...
## ...
summary(ds)
## Date Location MinTemp MaxTemp
## Min.:2007-11-01 Canberra :366 Min.: -5.30 Min.: 7.6
## 1st Qu.:2008-01-31 Adelaide :0 1st Qu.: 2.30 1st Qu.:15.0
## Median :2008-05-01 Albany :0 Median: 7.45 Median:19.6
## Mean :2008-05-01 Albury :0 Mean: 7.27 Mean:20.6
## ...
```
3 Decision Tree for Comparison

We begin by using our basic decision tree model as a base to compare the performance of the ensemble decision trees. See the DTrees module for details.

```r
model <- m.rp <- rpart(form, ds[train, vars])
```

```r
# n= 256
# node), split, n, loss, yval, (yprob)
# * denotes terminal node
# 1) root 256 38 No (0.85156 0.14844)
#  2) Humidity3pm< 71 238 25 No (0.89496 0.10504)
#  4) Pressure3pm>=1010 208 13 No (0.93750 0.06250) *
#  5) Pressure3pm< 1010 30 12 No (0.60000 0.40000)
# 10) Sunshine>=9.95 14 1 No (0.92857 0.07143) *
# 11) Sunshine< 9.95 16 5 Yes (0.31250 0.68750) *
# 3) Humidity3pm>=71 18 5 Yes (0.27778 0.72222) *
```

```r
fancyRpartPlot(model)
```

![Decision Tree Diagram]

Copyright © 2013-2014 Graham@togaware.com  Module: EnsemblesO  Page: 3 of 16
4 Decision Tree Performance

predicted <- predict(model, ds[test, vars], type="prob")[,2]
riskchart(predicted, actual)
5 Random Forest Model

```r
model <- m.rf <- randomForest(form, ds[train, vars])
model
```

---

Notice the out-of-bag (OOB) estimate of the error rate.

**Exercise:** Explain what an out-of-bag estimate is. How is it calculated for the random forest?
6 Random Forest Performance—Error Matrix

An error matrix shows, clockwise from the top left, the percentages of true negatives, false positives, true positives, and false negatives.

```r
predicted <- predict(model, ds[test, vars])
sum(actual != predicted)/length(predicted) # Overall error rate
## [1] 0.1545
round(100*table(actual, predicted, dnn=c("Actual", "Predicted"))/length(predicted))
##   Predicted
## Actual No Yes
## No    71   4
## Yes   12  14
## ....
```

Compare the matrix here with the OOB matrix from the randomForest() call itself. The data here is based on the 30% test dataset. The OOB estimate is based on the 70% sampled used as the training dataset.
7 Random Forest Performance—Risk Chart

```r
predicted <- predict(model, ds[test, vars], type="prob")[,2]
riskchart(predicted, actual)
```

![Risk Chart Diagram](image)
8 Conditional Random Forest

```r
model <- m.cf <- cforest(form, ds[train, vars])
model
#
## Random Forest using Conditional Inference Trees
## Number of trees: 500
....
```

```r
model <- m.cf <- cforest(form, ds[train, vars],
  controls=cforest_control(ntree=500,
    mtry=2,
    replace=FALSE,
    teststat="quad",
    testtype="Univ",
    mincriterion=0,
    fraction = 0.632,
    msplit=2,
    minbucket=1))
model
#
## Random Forest using Conditional Inference Trees
## Number of trees: 500
....
```
9 Weighted Subspace with RPart Decision Trees

```r
model <- m.wsrp <- wsrpart(form, ds[train, vars], ntrees=100)

model
```

```r
## A multiple rpart model with 100 trees.
## Variables used (11): MinTemp, Temp3pm, Rainfall, Cloud9am, Pressure3pm,
##                      WindSpeed9am, Pressure9am, Cloud3pm,
##                      Humidity3pm, Temp9am, Sunshine.
## # n= 256
## # node), split, n, loss, yval, (yprob)
## # * denotes terminal node
## # 1) root 256 33 No (0.871094 0.128906)
## 2) Pressure3pm>=1011 213 14 No (0.934272 0.065728)
## 4) Cloud3pm< 6.5 152 1 No (0.993421 0.006579) *
## 5) Cloud3pm>=6.5 61 13 No (0.786885 0.213115)
```
10 Weighted Subspace RPart Performance

predicted <- predict(model, ds[test, vars], type="prob")[,2]
riskchart(predicted, actual)
11 Weighted Subspace Random Forest

model <- m.wsrf <- wsrf(form, ds[train, vars], ntrees=500, nvars=20)

model
##
## Tree 1 with 29 nodes:
## 1) root
##   ..2) Sunshine <= 6.15
##   ...3) Pressure3pm <= 1016.5
##   ... .4) WindSpeed9am <= 2 No (1 0)
##   ... .5) WindSpeed9am > 2
##   ... . .6) Temp3pm <= 7.75 No (0.5 0.5)
##   ... . . .7) Temp3pm > 7.75 Yes (0 1)
##   ... . .8) Pressure3pm > 1016.5
##   ... . . .9) WindGustSpeed <= 23
##   ... . . . .10) Pressure3pm <= 1021.6 Yes (0 1)
##   ... . . . .11) Pressure3pm > 1021.6 No (0.5 0.5)
##   ... . . . .12) WindGustSpeed > 23
##   ... . . . .13) Temp3pm <= 10.2 No (0.5 0.5)
##   ... . . . .14) Temp3pm > 10.2 No (0 1)
##   ... . .15) Sunshine > 6.15
##   ... . .16) Cloud3pm <= 3.5 No (1 0)
##   ... . .17) Cloud3pm > 3.5
##   ... . .18) MaxTemp <= 28.85
##   ... . . .19) Humidity3pm <= 64
##   ... . . . .20) Pressure3pm <= 1012.05
##   ... . . . . .21) Pressure3pm <= 1011.5
##   ... . . . . . .22) Temp3pm <= 17.2 No (0.5 0.5)
##   ... . . . . . .23) Temp3pm > 17.2 No (1 0)
##   ... . . . . .24) Pressure3pm > 1011.5 Yes (0 1)
##   ... . . . . .25) Pressure3pm > 1012.05 No (1 0)
##   ... . . . .26) Humidity3pm > 64 Yes (0.333 0.667)
##   ... . . . .27) MaxTemp > 28.85
##   ... . . . .28) Sunshine <= 9.25 Yes (0 1)
##   ... . . . .29) Sunshine > 9.25 No (1 0)
##
## Tree 2 with 39 nodes:
## 1) root
##   ..2) Humidity3pm <= 72.5
##   ...3) Pressure3pm <= 1010.3
##   ... .4) Humidity3pm <= 53.5
##   ... . .5) WindSpeed3pm <= 10 Yes (0 1)
##   ... . . .6) WindSpeed3pm > 10
##   ... . . . .7) Cloud3pm <= 7.5
##   ... . . . .8) MaxTemp <= 28.85
##   ... . . . . .8) MaxTemp > 28.85
##   ... . . . . .9) Sunshine <= 9.25 Yes (0 1)
##   ... . . . .10) Sunshine > 9.25 No (1 0)
##   ... . .11) Cloud3pm <= 3.5 No (1 0)
##   ... . .12) Cloud3pm > 3.5
...
12 Weighted Subspace Random Forest Performance

predicted <- predict(model, ds[test, vars], type="prob")[,2]

riskchart(predicted, actual)
13 Wide Datasets

The weighted subspace algorithms target datasets with very many variables.

Exercise: Obtain a sample dataset with vary many variables, such as a text mining dataset, and compare the performances of rp, rf, wsrpart, and wsrf.
14 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 chapter is one of many chapters available from http://HandsOnDataScience.com. In particular follow the links on the website with a * which indicates the generally more developed chapters.
15 References


