The required packages for this module include:

| library(rattle) | # The weatherAUS dataset. |
| library(plyr)  | # Group by operations. |

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 \texttt{?} command as in:

\texttt{?read.csv}

We can obtain documentation on a particular package using the \texttt{help=} option of \texttt{library()}:

\texttt{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 Load the Data

We use the full weatherAUS dataset from rattle (Williams, 2014) to illustrate data summarisation over a more complex dataset.

ds <- weatherAUS
names(ds) <- normVarNames(names(ds))  # Lower case variable names.

names(ds)
## [1] "date" "location" "min_temp"
## [4] "max_temp" "rainfall" "evaporation"
## [7] "sunshine" "wind_gust_dir" "wind_gust_speed"
## [10] "wind_dir_9am" "wind_dir_3pm" "wind_speed_9am"

head(ds)
## date location min_temp max_temp rainfall evaporation sunshine
## 1 2008-12-01 Albury 13.4 22.9 0.6 NA NA
## 2 2008-12-02 Albury 7.4 25.1 0.0 NA NA
## 3 2008-12-03 Albury 12.9 25.7 0.0 NA NA

head(ds)
## date location min_temp max_temp rainfall evaporation sunshine
## 88763 2014-04-20 Uluru 10.3 29.6 0 NA NA
## 88764 2014-04-21 Uluru 11.3 30.5 0 NA NA
## 88765 2014-04-22 Uluru 10.1 31.6 0 NA NA

head(ds[sample(nrow(ds), 6),])
## date location min_temp max_temp rainfall evaporation
## 42691 2011-01-30 Melbourne 16.4 38.1 0.0 7.4
## 74988 2010-03-21 Perth 19.5 33.1 0.0 6.0
## 46982 2011-08-14 Portland 2.5 15.4 0.2 1.0

str(ds)
## 'data.frame': 88768 obs. of 24 variables:
## $ date : Date, format: "2008-12-01" "2008-12-02" ...
## $ location : Factor w/ 49 levels "Adelaide","Albany",...: 3 3 3 3 3 ...
## $ min_temp : num 13.4 7.4 12.9 9.2 17.5 14.6 14.3 7.7 9.7 13.1 ...

summary(ds)
## date location min_temp max_temp
## Min. :2007-11-01 Canberra: 2279 Min. :-8.5 Min. :-3.8
## 1st Qu.:2010-03-08 Sydney : 2187 1st Qu.: 7.6 1st Qu.:18.0
## Median :2011-08-03 Adelaide: 2036 Median :12.0 Median :22.5
##
2 Dataset Indexing

Often we will be on the lookout for oddities or data typing that need fixing up. Once identified we will use the operations covered in a separate session on Transforming data.

We start by looking at some of the data. This introduces the concept of indexing our data frame.

```r
ds[1,]  # First observation.
## date location min_temp max_temp rainfall evaporation sunshine
## 1 2008-12-01 Albury 13.4 22.9 0.6 NA NA
## wind_gust_dir wind_gust_speed wind_dir_9am wind_dir_3pm wind_speed_9am
## 1 W 44 W WNW 20
....
ds[1:2,]  # First two observations.
## date location min_temp max_temp rainfall evaporation sunshine
## 1 2008-12-01 Albury 13.4 22.9 0.6 NA NA
## 2 2008-12-02 Albury 7.4 25.1 0.0 NA NA
## wind_gust_dir wind_gust_speed wind_dir_9am wind_dir_3pm wind_speed_9am
## 1 W 44 W WNW 20
....
ds[1:2, 3:4]  # First two observations and variables 3 and 4.
## min_temp max_temp
## 1 13.4 22.9
## 2 7.4 25.1
head(ds[3:4], 2)  # Single dimension treated as variable index.
## min_temp max_temp
## 1 13.4 22.9
## 2 7.4 25.1
head(ds[,3:4], 2)  # Or we can leave the observation index empty.
## min_temp max_temp
## 1 13.4 22.9
## 2 7.4 25.1
```
### Summary

The `summary()` command provides a quick univariate overview of our dataset.

```r
summary(ds, digits=6)
```

<table>
<thead>
<tr>
<th></th>
<th>date</th>
<th>location</th>
<th>min_temp</th>
<th>max_temp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>2007-11-01</td>
<td>Canberra</td>
<td>Min. -8.5</td>
<td>Min. -3.8</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>2010-03-08</td>
<td>Sydney</td>
<td>1st Qu. 7.6</td>
<td>1st Qu. 18.0</td>
</tr>
<tr>
<td>Median</td>
<td>2011-08-03</td>
<td>Adelaide</td>
<td>Median :12.0</td>
<td>Median :22.5</td>
</tr>
<tr>
<td>Mean</td>
<td>2011-07-29</td>
<td>Brisbane</td>
<td>Mean :12.2</td>
<td>Mean :23.1</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>2012-11-27</td>
<td>Darwin</td>
<td>3rd Qu.: 16.8</td>
<td>3rd Qu.: 28.0</td>
</tr>
<tr>
<td>Max.</td>
<td>2014-04-25</td>
<td>Hobart</td>
<td>Max. :33.9</td>
<td>Max. :48.1</td>
</tr>
<tr>
<td>(Other)</td>
<td>76158</td>
<td></td>
<td>NA's :669</td>
<td>NA's :493</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>rainfall</th>
<th>evaporation</th>
<th>sunshine</th>
<th>wind_gust_dir</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>0</td>
<td>Min. 0</td>
<td>Min. 0</td>
<td>W : 5850</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>0.0</td>
<td>1st Qu. 3</td>
<td>1st Qu. 5</td>
<td>SE : 5796</td>
</tr>
<tr>
<td>Median</td>
<td>2.5</td>
<td>Median 5</td>
<td>Median 8</td>
<td>N : 5739</td>
</tr>
<tr>
<td>Mean</td>
<td>0.8</td>
<td>Mean 5</td>
<td>Mean 8</td>
<td>S : 5669</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>7</td>
<td>3rd Qu. 11</td>
<td>3rd Qu. 11</td>
<td>SSE : 5583</td>
</tr>
<tr>
<td>Max.</td>
<td>371.0</td>
<td>Max. 14</td>
<td>(Other) :53353</td>
<td></td>
</tr>
<tr>
<td>NA's</td>
<td>1656</td>
<td>NA's :32687</td>
<td>NA's :35530</td>
<td>NA's :6778</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>wind_gust_speed</th>
<th>wind_dir_9am</th>
<th>wind_dir_3pm</th>
<th>wind_speed_9am</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>6</td>
<td>N : 7200</td>
<td>SE : 6928</td>
<td>Min. 0</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>31</td>
<td>1st Qu. 5</td>
<td>1st Qu. 7</td>
<td>1st Qu. 7.0</td>
</tr>
<tr>
<td>Median</td>
<td>39</td>
<td>E : 5568</td>
<td>S : 6071</td>
<td>Median :13.0</td>
</tr>
<tr>
<td>Mean</td>
<td>40</td>
<td>SSE : 5508</td>
<td>WSW : 5846</td>
<td>Mean :14.2</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>48</td>
<td>S : 5345</td>
<td>SSE : 5804</td>
<td>3rd Qu. 20.0</td>
</tr>
<tr>
<td>Max.</td>
<td>135</td>
<td>(Other) :52838</td>
<td>(Other) :56040</td>
<td>Max. :87.0</td>
</tr>
<tr>
<td>NA's</td>
<td>6738</td>
<td>NA's :6641</td>
<td>NA's :1956</td>
<td>NA's :1159</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>wind_speed_3pm</th>
<th>humidity_9am</th>
<th>humidity_3pm</th>
<th>pressure_9am</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>0.0</td>
<td>Min. 0</td>
<td>Min. 0</td>
<td>Min. 980</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>13.0</td>
<td>1st Qu. 57.0</td>
<td>1st Qu. 37.0</td>
<td>1st Qu. 1013</td>
</tr>
<tr>
<td>Median</td>
<td>19.0</td>
<td>Median 70.0</td>
<td>Median 52.0</td>
<td>Median :1017</td>
</tr>
<tr>
<td>Mean</td>
<td>18.8</td>
<td>Mean 68.7</td>
<td>Mean 51.6</td>
<td>Mean :1017</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>24.0</td>
<td>3rd Qu. 83.0</td>
<td>3rd Qu. 66.0</td>
<td>3rd Qu. 1022</td>
</tr>
<tr>
<td>Max.</td>
<td>87.0</td>
<td>Max. 100.0</td>
<td>Max. 100.0</td>
<td>Max. :1041</td>
</tr>
<tr>
<td>NA's</td>
<td>1170</td>
<td>NA's :1495</td>
<td>NA's :1389</td>
<td>NA's :8273</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>pressure_3pm</th>
<th>cloud_9am</th>
<th>cloud_3pm</th>
<th>temp_9am</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min.</td>
<td>979</td>
<td>Min. 0</td>
<td>Min. 0</td>
<td>Min. -5.9</td>
</tr>
<tr>
<td>1st Qu.</td>
<td>1010</td>
<td>1st Qu. 1</td>
<td>1st Qu. 2</td>
<td>1st Qu. 12.3</td>
</tr>
<tr>
<td>Median</td>
<td>1015</td>
<td>Median 5</td>
<td>Median 5</td>
<td>Median :16.7</td>
</tr>
<tr>
<td>Mean</td>
<td>1015</td>
<td>Mean 4</td>
<td>Mean 4</td>
<td>Mean :17.0</td>
</tr>
<tr>
<td>3rd Qu.</td>
<td>1020</td>
<td>3rd Qu. 7</td>
<td>3rd Qu. 7</td>
<td>3rd Qu. 21.5</td>
</tr>
<tr>
<td>Max.</td>
<td>1040</td>
<td>Max. 9</td>
<td>Max. 9</td>
<td>Max. :40.2</td>
</tr>
<tr>
<td>NA's</td>
<td>8245</td>
<td>NA's :32337</td>
<td>NA's :33339</td>
<td>NA's :1040</td>
</tr>
</tbody>
</table>
```

---

Copyright © 2013-2014 Graham@togaware.com  Module: SummaryO  Page: 3 of 10
4 Textual Summaries—Warning

Do be weary of the results provided by `summary()`. The `summary()` command rounds the results to 4 digits by default. This can surprise us sometimes when we find `min()` and the reported minimum value from `summary()` disagree! Let’s look at some random data and notice the reported minimum value.

```r
eg <- sample(1e6:(1e7-1), 100)
max(eg)
## [1] 9882363
min(eg)
## [1] 1146522
summary(eg)
## Min. 1st Qu. Median Mean 3rd Qu. Max.
## 1150000 3050000 5120000 5350000 8050000 9880000
summary(eg, digits=4)
## Min. 1st Qu. Median Mean 3rd Qu. Max.
## 1147000 3051000 5123000 5348000 8051000 9882000
summary(eg, digits=5)
## Min. 1st Qu. Median Mean 3rd Qu. Max.
## 1146500 3050700 5123000 5348100 8050880 9882360
summary(eg, digits=6)
## Min. 1st Qu. Median Mean 3rd Qu. Max.
## 1146520 3050710 5123028 5348103 8050881 9882363
```
5 PlyR: Summarise per Group to new Data Frame

The plyr (Wickham, 2014) package provides a clean and consistent approach to transforming data. We can easily, for example, transform a data frame into a new smaller data frame grouped by the location.

```r
temps <- ddply(ds, "location", summarise,
                max=max(max_temp, na.rm=TRUE),
                min=min(min_temp, na.rm=TRUE))
temps
## location max min
## 1 Adelaide 45.7 0.7
## 2 Albany 38.9 1.8
## 3 Albury 44.8 -2.5
....
```

The plyr package also provides the .() function as a convenient mechanism for listing variable names without the need to quote them. The function becomes more convenient when we have multiple variables to list.

```r
temps <- ddply(ds, .(location), summarise,
                max=max(max_temp, na.rm=TRUE),
                min=min(min_temp, na.rm=TRUE))
temps
## location max min
## 1 Adelaide 45.7 0.7
## 2 Albany 38.9 1.8
## 3 Albury 44.8 -2.5
....
```

We can review the resulting values, ordered by the maximum temperature.

```r
temps[order(temps$max, decreasing=TRUE),]
## location max min
## 49 Woomera 48.1 0.7
## 22 Moree 47.3 -3.3
## 20 MelbourneAirport 46.8 -0.4
....
```

Similarly, but ordered by the minimum temperature.

```r
head(temps[order(temps$min),])
## location max min
## 24 MountGinini 31.1 -8.5
## 41 Tuggeranong 40.1 -8.2
## 10 Canberra 42.0 -8.0
....
```
6 PlyR: Summarise per Group to Original Data Frame

Transform a data frame by adding the group summaries per original observation, simply by replacing `summarise` with `transform`.

```r
temps <- ddply(ds, .(location), transform,
               max=max(max_temp, na.rm=TRUE),
               min=min(min_temp, na.rm=TRUE))
```

Now notice that the top few values for `min` and `max` are constant, since they belong to the same group (`Adelaide`).

```r
head(temps[c("date", "location", "min_temp", "min", "max_temp", "max")])
```

```
  date location min_temp min max_temp max
  1 2008-07-01 Adelaide 8.8 0.7 15.7 45.7
  2 2008-07-02 Adelaide 12.7 0.7 15.8 45.7
  3 2008-07-03 Adelaide 6.2 0.7 15.1 45.7
  ....
```

If we same a few observations we see the various values of `min` and `max` across different `locations`.

```r
temps[sample(nrow(temps), 10),
      c("date", "location", "min_temp", "min", "max_temp", "max")]
```

```
  date location min_temp min_max_temp max
  # 88579 2013-10-18 Woomera 7.7 0.7 27.8 48.1
  # 87470 2010-07-08 Woomera 0.7 0.7 15.3 48.1
  # 77879 2009-09-08 Walpole 5.5 3.4 17.8 39.3
  ....
```
7 PlyR: Select One Observation Per Group

We can also select a single observation per group, using some criteria to decide which observation to pick. We replace the `summarise` or `transform` with a function to select the observation of interest.

```r
temps <- ddply(ds, .(location),
               function(x) x[x$max_temp == max(x$max_temp, na.rm=TRUE),])
head(temps[1:7])
```

```
## date location min_temp max_temp rainfall evaporation sunshine
## 1 <NA> <NA> NA NA NA NA NA
## 2 2009-01-28 Adelaide 30.7 45.7 0 13.0 12.5
## 3 2010-01-18 Albany 17.8 38.9 0 11.8 12.8
## 4 2009-02-07 Albury 22.3 44.8 0 NA NA
```

Notice the unexpected rows of missing values. The vector comparison, using `==()`, will return `NA` whenever comparing `NA`'s and an index to `[()]` of `NA` will return an NA row for each observation. We can get around this issue of missing values by testing whether we get `TRUE` from the comparison, rather than `FALSE` or `NA` by using `identical()`.

```r
temps <- ddply(ds, .(location),
               function(x) x[sapply(x$max_temp == max(x$max_temp, na.rm=TRUE), identical, TRUE),])
head(temps[1:7])
```

```
## date location min_temp max_temp rainfall evaporation sunshine
## 1 2009-01-28 Adelaide 30.7 45.7 0 13.0 12.5
## 2 2010-01-18 Albany 17.8 38.9 0 11.8 12.8
## 3 2009-02-07 Albury 22.3 44.8 0 NA NA
```

8 Further Reading

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

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


