Rmarkup – a very simple tool for literate programming

Søren Højsgaard

Department of Mathematical Sciences
Aalborg University
Denmark

useR! 2012 conference
Nashville, TN, USA

14. juni 2012

- `Rmarkup()` very leight weight tool for literate programming.
- Not unlike `R Markdown` presented earlier by JJ Allarie.
- Input: Plain text file containing
  - R code
  - Comment lines (starting with ##) with descriptive text. Simple text markups are implemented.
- Output: HTML document containing
  - Descriptive text (with possible markups),
  - program code together with
  - graphics and results from the computations.
- Genesis: Teaching R in life sciences
- The `Rmarkup()` function is in the **doBy** package.
- `Rmarkup()` is implemented by using the `RweaveHTML` driver in the **R2HTML**

```
## == __Rmarkup Example 1__ ==
## === Søren Højsgaard ===
## %%date

## In this __example__ we //embed// parts of the **examples**
## from the &&kruskal.test&& help page into an HTML document:

##@@
data ( airquality )
kruskal.test ( Ozone ~ Month , data = airquality )
##@

## which shows that the location parameter of the Ozone
## distribution varies significantly from month to month.
## Finally we include a boxplot of the data :

##@@@
boxplot ( Ozone ~ Month , data = airquality )
##@
```

- Run `Rmarkup()` as:

```
R> Rmarkup("SweaveEx1.R",
+       encoding = "latin1",      # because of the 'ø's
+       cssfile  = "R2HTML.css") # optional css file
```

- Will produce `SweaveEx1.html` in working directory (where the `css` file must reside).
- If a pdf-file is needed, the utility `wkhtmltopdf` is useful, see

  http://code.google.com/p/wkhtmltopdf/

- Simply do:

```
wkhtmltopdf SweaveEx1.html SweaveEx1.pdf
```

Rmarkup Example 1

Søren Højsgaard


```
[1] "2012-06-14 23:08:07"
```

In this example we *embed* parts of the **examples** from the `kruskal.test` help page into an HTML document:

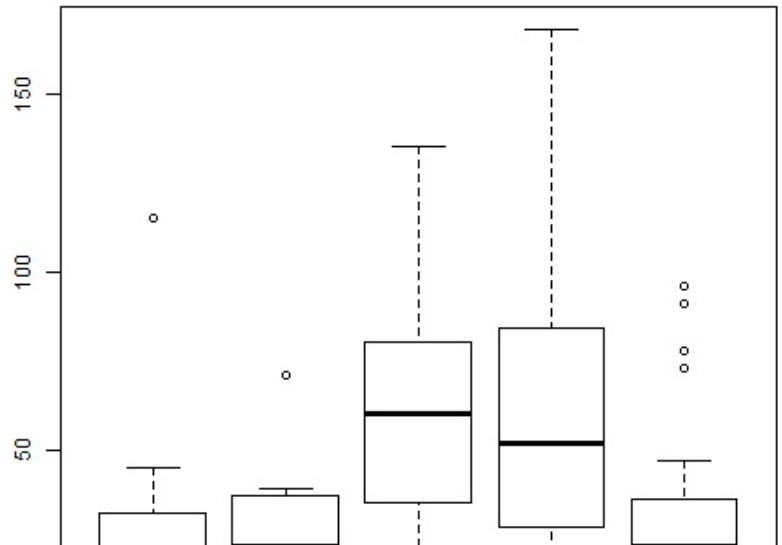
```
> data(airquality)
```

```
> kruskal.test(Ozone ~ Month, data = airquality)
```

```

	Kruskal-Wallis rank sum test

data:  Ozone by Month
Kruskal-Wallis chi-squared = 29.2666, df = 4, p-value =
6.901e-06
```

which shows that the location parameter of the Ozone distribution varies significantly from month to month. Finally we include a boxplot of the data :

```
> boxplot(Ozone ~ Month, data = airquality)
```



All text lines start with one or more hashes (#) because these lines are to be regarded as comments by R.

Lines starting with:

- One or two hashes: Regarded as a text which is transferred (possibly after some additional processing; see below) to the resulting HTML document.
- Three or more hashes: Not transferred to the HTML document. (This is useful e.g. for TODOs).

- `Rmarkup()` allows some markup facilities for the text inspired by `txt2tags` markups (see `http://txt2tags.org/`).
- Headings at different font sizes are produced with (there can be 6 levels of headings):
  `= Title level 1 =`, `== Title level 2 ==`, ...
- The time of creation of the HTML document is produced with `%%date`.

- Beautifiers:
  boldface, *italics*, underline, `monospace` :
- Produced with:
  boldface, //italics// __underline__, &&monospace&&
- Beautifiers can be combined in any way, e.g.
  __some text__.
- Beautifiers can be used in the headings.

- Writing

```
## @@
data(airquality)
## @
```

  is equivalent to

```
## <<>>=
data(airquality)
## @
```

- Other valid specs can go into <<>>.

- Writing

```
## @@@
plot(airquality)
## @
```

  is equivalent to

```
## <<fig=T>>=
data(airquality)
## @
```

- Other valid specs can go into <<>>.

```
## == __Rmarkup Example 2__ ==
## === Søren Højsgaard ===
## %%date

## In this example we embed parts of the examples from
## the &&kruskal.test&& help page into an HTML document:

##@@
data ( airquality )
kruskal.test ( Ozone ~ Month , data = airquality )
##@

## which shows that the location parameter of the Ozone
## distribution varies significantly from month to month.
##
## Notice that there are __\Sexpr{nrow(airquality)}__
## observations.
##
## Finally we include a boxplot of the data :

##@@@
boxplot ( Ozone ~ Month , data = airquality )
##@
```

Run Rmarkup() as:

```
R> Rmarkup("script/SweaveEx2.R",
+          destdir  = "./report",
+          encoding = "latin1",
+          cssfile  = "R2HTML.css",
+          parms = list(height = 200, width = 200)
+          )
```

Rmarkup Example 2

Søren Højsgaard


```
[1] "2012-06-14 23:08:08"
```

In this example we embed parts of the examples from the `kruskal.test` help page into an HTML document:

```
> data(airquality)
```

```
> kruskal.test(Ozone ~ Month, data = airquality)
```

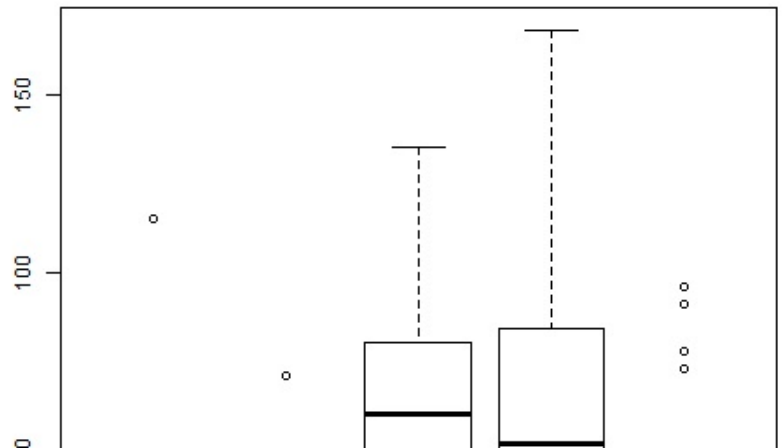
```

Kruskal-Wallis rank sum test

data:  Ozone by Month
Kruskal-Wallis chi-squared = 29.2666, df = 4, p-value =
6.901e-06
```

which shows that the location parameter of the Ozone distribution varies significantly from month to month.

Notice that there are 153 observations.

Finally we include a boxplot of the data :

```
> boxplot(Ozone ~ Month, data = airquality)
```



- A text markup must be completed in one line
- Example: a heading in a large font can be obtained with

```
## = HERE COMES A TITLE =
```

- – whereas this is not obtained if one writes e.g.

```
## =
##   HERE COMES A TITLE
## =
```

- Comments in code chunks will not appear in output

`Rmarkup()` grew out of teaching R to graduate students and others in the life sciences.

- Mainly Microsoft Office users
- Learning R itself is a hurdle.
- Hesitant to install to much software on their computers. Installing R and a suitable editor (e.g. Notepad++ for Windows users) is about as much as we can ask for.
- (`Rstudio` could well be a better choice today.)
- Over the top to ask students to learn LaTeX (also irrelevant; they collaborate on / submit papers in MS Office format)
- Even **odfWeave** is problematic, as there is no Windows binary on CRAN (depends on **XML**?).

Why would anyone possibly want to use `Rmarkup()` when so many fancy literate programming tools are available?

- Often we work with a script file as a sandbox for e.g. data manipulation tasks, exploratory data analysis etc.
- To document the work one may choose to create a LaTeXfile with describing such activities using `Sweave`.
- But most people maintain their R scripts. So one may have two different files describing essentially the same tasks.
- `Rmarkup()` allows one to create a reasonably nice looking report from an R script itself.
- Hence the task of documenting the work (in particular the more tedious parts of the work) is more likely to be done.

- `Rmarkup()` may be useful in teaching students "good habbits"
- `Rmarkup()` may help others doing the same...
- `Rmarkup()` will never be developed to a fancy tools with all sorts of bells and whistles.
- Still, open to suggestions - to pick the "low hanging fruits"
- A small demo!!

Thank you for your attention!