# DynDocs with Word Example

Duncan Temple Lang

In this example, we embed parts of the examples from the kruskal.test help page into a Word document. This is taken as an almost direct copy of the example in the Sweave manual to aid understanding the similarities and differences between Sweave and this approach. Then we can process the document to

1. extract the code from the document to run directly in R or save to a file(s)
2. evaluate the code in order and insert the results into a new document along with the original text.

We start with the code and put it in a paragraph (with soft-line breaks) and set the style to be Rcode:

data(airquality)  
library(stats)  
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

This shows that the location parameter of the Ozone distribution varies significantly from month to month. We end by including a boxplot of the data:

boxplot(Ozone ~ Month, data = airquality)

## The Motor Trends cars data!

Here we look at the age-old mtcars dataset in R. We compute a simple two-way frequency table for the variables cyl and gear.

data(mtcars)  
xtabs( ~ cyl + gear, mtcars)

|  |  |  |  |
| --- | --- | --- | --- |
|  | 3 | 4 | 5 |
| 4 | 1 | 8 | 2 |
| 6 | 2 | 4 | 1 |
| 8 | 12 | 0 | 2 |

This illustrates the method for converting an R object of class table to a Word table.

# Adding styling information for R output

This illustrates how we can generate content from R and put it into an existing and formatted element of the Word document. In this case, the element is a table.

sapply(c(3, 10, 20), function(lambda) rpois(6, lambda))

[,1] [,2] [,3]  
[1,] 1 7 15  
[2,] 3 11 23  
[3,] 0 18 19  
[4,] 3 7 12  
[5,] 2 12 26  
[6,] 6 15 26

|  |  |  |
| --- | --- | --- |
| 1 | 7 | 13 |
| 2 | 8 | 14 |
| 3 | 9 | 15 |
| 4 | 10 | 16 |
| 5 | 11 | 17 |
| 6 | 12 | 18 |

This is another table that we can fill in from the result of the R code

replicate(3, round(rnorm(5), 5))

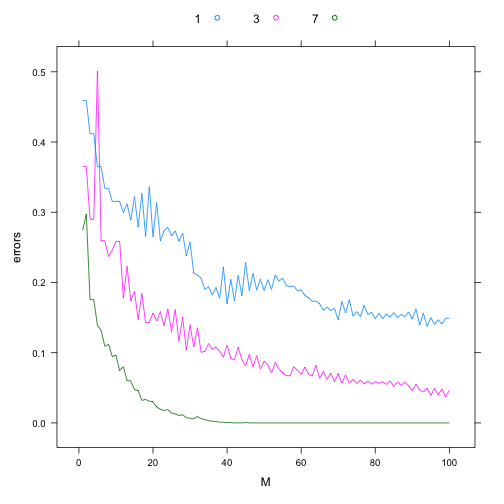
|  |  |  |
| --- | --- | --- |
| -0.55367 | -1.05628 | -0.34143 |
| -1.49091 | -0.26065 | -0.58247 |
| 0.25216 | 0.5084 | -0.32784 |
| -0.29706 | -0.43068 | 0.27642 |
| 0.34505 | 0.62902 | -0.09203 |

How can we control the number of digits in the output? Can we have 1.0123456789 appear as 1.014 without explicitly having to round? Or is explicitly round so terrible?

Note that the table above is formatted by the author to have the left column left-aligned, the middle column centered and the right column right-aligned. Also, he has interactively changed the column widths from the default by dragging the column separator lines.

Next we insert a picture

plot(1:10)



This is intentionally inserted within an “R plot output” style that has been customized with a border. (Very ugly!)