# *RcppOctave* package: Seamless Interface to Octave – and Matlab

Renaud Gaujoux

*RcppOctave* package – Version 0.7.2 [April 5, 2012]

The *RcppOctave* package package provides a direct interface to Octave from R, implemented using the *Rcpp* package[1] [1]. It allows to call Octave functions in a similar way as calling C/C++/Fortran functions using the R core function `.Call`. Since Octave uses a language that is mostly compatible with Matlab, the *RcppOctave* package may also be used to run Matlab m-files. As a matter of fact, this package was originally developed to facilitates the port and comparison of R and Matlab code.

The package provides Octave modules that allow calling R core random number generators from Octave, which enables to reproduce and compare stochastic computations.

## 1 Calling Octave functions from R

The *p*ackage provides the function `.CallOctave` to call Octave functions from R, mimicking the way native *C/C++* functions are called with `.Call`:

```
.O$eye(3)

##      [,1] [,2] [,3]
## [1,]    1    0    0
## [2,]    0    1    0
## [3,]    0    0    1

.O$svd(matrix(1:9, 3))

##           [,1]
## [1,] 1.685e+01
## [2,] 1.068e+00
## [3,] 5.543e-16
```

## 2 Calling R functions from Octave

TODO

## 3 Direct interface: the `.O` object

A convenient shortcut interface in defined by the object `.O` of class `Octave`, exported from *RcppOctave* package namespace:

[1] http://cran.r-project.org/package=Rcpp

```
.O$eye(3)

##      [,1] [,2] [,3]
## [1,]    1    0    0
## [2,]    0    1    0
## [3,]    0    0    1

.O$svd(matrix(1:9, 3))

##           [,1]
## [1,] 1.685e+01
## [2,] 1.068e+00
## [3,] 5.543e-16
```

# 4 Sample session

Comparing equivalent R and Octave functions is therefore as easy as comparing two R functions. For example, one can compare the respective functions `svd` with the following code:

```
# define random data
X <- matrix(runif(25), 5)

# run SVD in R
svd.R <- svd(X)
# run SVD in Octave
svd.O <- .O$svd(X)
# check results
svd.O

##        [,1]
## [1,] 2.3467
## [2,] 0.9458
## [3,] 0.8666
## [4,] 0.4380
## [5,] 0.1596

all.equal(svd.R$d, as.numeric(svd.O))

## [1] TRUE

# but not exactly identical
all.equal(svd.R$d, as.numeric(svd.O), tol = 10^-16)

## [1] "Mean relative difference: 2.879e-16"
```

We notice here that Octave default `svd` returns only the eigen values as a column vector. This is documented in its documentation that is accessible via the function `o_help`, which will show it in a similar way as R documentation:

```
# show Octave help for svd
o_help(svd)
```

The documentation for – Octave – `svd` states that the complete decomposition is returned, if three output values are provided. This can be done using argument `argout`:

```
# get full output from Octave svd
.O$svd(X, argout = 3)

## [[1]]
##         [,1]     [,2]    [,3]     [,4]     [,5]
## [1,] -0.3222  0.78366 -0.3214 -0.05106 -0.41972
## [2,] -0.2893 -0.09965  0.1416 -0.94051  0.04197
## [3,] -0.4209 -0.41952  0.3078  0.18818 -0.71883
## [4,] -0.5131 -0.37720 -0.7277  0.09871  0.23483
## [5,] -0.6100  0.24014  0.5023  0.26016  0.50022
##
## [[2]]
##       [,1]   [,2]   [,3]  [,4]   [,5]
## [1,] 2.347 0.0000 0.0000 0.000 0.0000
## [2,] 0.000 0.9458 0.0000 0.000 0.0000
## [3,] 0.000 0.0000 0.8666 0.000 0.0000
## [4,] 0.000 0.0000 0.0000 0.438 0.0000
## [5,] 0.000 0.0000 0.0000 0.000 0.1596
##
## [[3]]
##         [,1]     [,2]    [,3]      [,4]    [,5]
## [1,] -0.3898  0.81421 -0.1255  0.052488 -0.4082
## [2,] -0.5636  0.02137 -0.4267 -0.121157  0.6965
## [3,] -0.4763 -0.56957 -0.3405 -0.001904 -0.5769
## [4,] -0.4349 -0.10317  0.5735  0.675888  0.1202
## [5,] -0.3382 -0.03931  0.5977 -0.725077 -0.0324
##
```

## Session information

```
R version 2.14.2 (2012-02-29)
Platform: x86_64-pc-linux-gnu (64-bit)

locale:
 [1] LC_CTYPE=en_ZA.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=en_ZA.UTF-8        LC_COLLATE=en_ZA.UTF-8
 [5] LC_MONETARY=en_ZA.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=C                 LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_ZA.UTF-8 LC_IDENTIFICATION=C

attached base packages:
[1] methods   stats     graphics  grDevices utils     datasets  base

other attached packages:
[1] RcppOctave_0.7.2 Rcpp_0.9.10      knitr_0.4

loaded via a namespace (and not attached):
[1] codetools_0.2-8 digest_0.5.2    evaluate_0.4.1  formatR_0.3-4
[5] highlight_0.3.1 parser_0.0-14   plyr_1.7.1      stringr_0.6
[9] tools_2.14.2
```

# References

[1] Dirk Eddelbuettel and Romain François. "Rcpp: Seamless R and C++ Integration". In: *Journal of Statistical Software* 40.8 (2011), pp. 1–18. URL: http://www.jstatsoft.org/v40/i08/.