

# An exposé of naming conventions in R.

Rasmus Bååth<sup>1\*</sup>, Martin Jönsson<sup>2</sup>

1. Cognitive Science, Department of Philosophy, Lund University, Sweden.

2. Department of Mathematical Statistics, Lund University, Sweden.

\*Contact author: [rasmus.baath@lucs.lu.se](mailto:rasmus.baath@lucs.lu.se)

**Keywords:** R programming, Naming conventions, Coding Style

R is one of the most heterogeneous programming languages when it comes to naming conventions. An example of this is the conventions for function names where most other modern languages are divided between using underscore-separated and lowerCamelCase names and have official guidelines stating which convention to prefer. In the R community there are no less than five different naming conventions for function names in use and many unofficial guidelines disagreeing on which one to prefer. R is rapidly gaining in popularity and there is a steady stream of newcomers having to decide what naming conventions to adopt. If you are a newcomer to R or if you are a package developer you would probably want to adhere to the current naming conventions of the R community, but how to know what the most common convention is? While there are no official guidelines there fortunately exist ample information regarding what conventions are used *in practice* as the *Comprehensive R Archive Network* (<http://cran.r-project.org/>) contains the code and documentation of over 4000 R packages.

We downloaded the documentation of each package on CRAN and counted what proportion of function and parameter names that followed different naming conventions. A summary of the results were published in the *Programmer's Niche* section in the R journal Bååth (2012) with the main finding being that the most common naming convention for function names was lowerCamelCase (55% matched this convention) and that most argument names were period.separated (83 %). It was also found that many packages use mixed naming conventions (28 % of the packages mix three or more conventions). A lot of interesting findings did not make it into the *Programmer's Niche* article however. Using information regarding when a package was first released it is possible to study naming convention usage over time (see figure 1) which seems to be rather stable with a possible downward trend for using period.separated names. The CRAN documentation is also full of examples of function names that don't seem to follow any naming convention, for example, the function names `Balanced.Initialization` and `mv_truthTable` and parameter names `To.prime`, `missing.PlugIn` and `outputSGP_INDIVIDUAL.content_areas`. Except for how to handle identifiers consisting of many words there are also many other implicit naming conventions worth pointing out.

## References

Bååth, R. (2012, December). The state of naming conventions in R. *The R Journal* 4(2), 65–73.

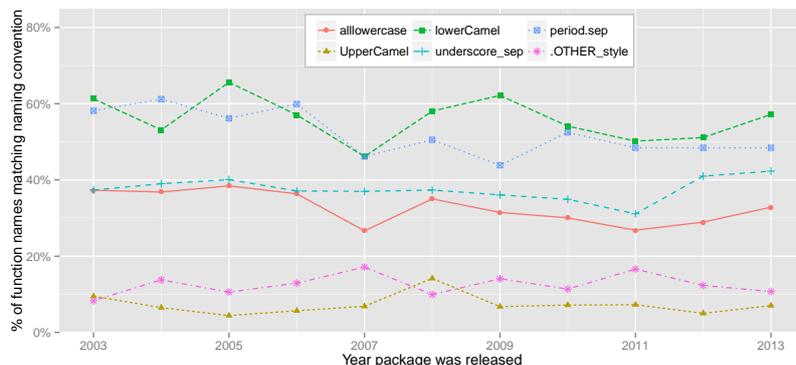


Figure 1: Naming convention usage on CRAN from 2003 to March 2013.