Very large numbers in R: Introducing package Brobdingnag

Robin K. S. Hankin
University of Cambridge

Abstract

This vignette shows how to use the **Brobdingnag** package to manipulate very large numbers; it is based on Hankin (2007c).

The other vignette shows how to use S4 methods in the context of a simple package.

Keywords: S4 methods, Brobdingnag, R.

1. Introduction

The largest floating point number representable in standard double precision arithmetic is a little under 2^{1024}, or about 1.79×10^{308}. This is too small for some applications.

The R package **Brobdingnag** (Swift 1726) overcomes this limit by representing a real number x using a double precision variable with value $\log|x|$, and a logical corresponding to $x \geq 0$; the S4 class of such objects is `brob`. Complex numbers with large absolute values (class `glub`) may be represented using a pair of `brob`s to represent the real and imaginary components.

The package allows user-transparent access to the large numbers allowed by Brobdingnagian arithmetic. The package also includes a vignette—`brob`—which documents the S4 methods used and includes a step-by-step tutorial. The vignette also functions as a "Hello, World!" example of S4 methods as used in a simple package. It also includes a full description of the `glub` class.

2. Package Brobdingnag in use

Most readers will be aware of a googol which is equal to 10^{100}:

```
> require(Brobdingnag)
```

```
> googol <- as.brob(10)^100
```

```
[1] +exp(230.26)
```

Note the coercion of `double` value `10` to an object of class `brob` using function `as.brob()`: raising this to the power 100 (also double) results in another `brob`. The result is printed using exponential notation, which is convenient for very large numbers.

A googol is well within the capabilities of standard double precision arithmetic. Now, however, suppose we wish to compute its factorial. Taking the first term of Stirling's series gives

```
> stirling <- function(n) {
+     n^n * exp(-n) * sqrt(2 * pi * n)
+ }
```

which then yields

```
> stirling(googol)
```

```
[1] +exp(2.2926e+102)
```

Note the transparent coercion to `brob` form within function `stirling()`.

It is also possible to represent numbers very close to 1. Thus

```
> 2^(1/googol)
```

```
[1] +exp(6.9315e-101)
```

It is worth noting that if x has an exact representation in double precision, then e^x is exactly representable using the system described here. Thus e and e^{1000} may be represented exactly.

2.1. Accuracy

For small numbers (that is, representable using standard double precision floating point arithmetic), **Brobdingnag** suffers a slight loss of precision compared to normal representation. Consider the following function, whose return value for nonzero arguments is algebraically zero:

```
f <- function(x){
   as.numeric( (pi*x -3*x -(pi-3)*x)/x)
}
```

This function combines multiplication and addition; one might expect a logarithmic system such as described here to have difficulty with it.

```
> f(1/7)
```

```
[1] 1.700029e-16
```

```
> f(as.brob(1/7))
```

```
[1] -1.886393e-16
```

This typical example shows that Brobdingnagian numbers suffer a slight loss of precision for numbers of moderate magnitude. This degradation increases with the magnitude of the argument:

```
> f(1e+100)

[1] -2.185503e-16

> f(as.brob(1e+100))

[1] -3.219444e-14
```

Here, the brobs' accuracy is about two orders of magnitude worse than double precision arithmetic: this would be expected, as the number of bits required to specify the exponent goes as $\log\log x$.

Compare

```
> f(as.brob(10)^1000)

[1] 1.931667e-13
```

showing a further degradation of precision. However, observe that conventional double precision arithmetic cannot deal with numbers this big, and the package returns about 12 correct significant figures.

3. A practical example

In the field of population dynamics, and especially the modelling of biodiversity (Hankin 2007a; Hubbell 2001), complicated combinatorial formulae often arise.

Etienne (2005), for example, considers a sample of N individual organisms taken from some natural population; the sample includes S distinct species, and each individual is assigned a label in the range 1 to S. The sample comprises n_i members of species i, with $1 \leq i \leq S$ and $\sum n_i = N$. For a given sample D Etienne defines, amongst other terms, $K(D, A)$ for $1 \leq A \leq N - S + 1$ as

$$\sum_{\left\{a_1,\ldots,a_S \middle| \sum_{i=1}^{S} a_i = A\right\}} \prod_{i=1}^{S} \frac{\overline{s}(n_i, a_i)\overline{s}(a_i, 1)}{\overline{s}(n_i, 1)} \qquad (1)$$

where $\overline{s}(n, a)$ is the Stirling number of the second kind (Abramowitz and Stegun 1965). The summation is over $a_i = 1, \ldots, n_i$ with the restriction that the a_i sum to A, as carried out by `blockparts()` of the **partitions** package (Hankin 2006, 2007b).

Taking an intermediate-sized dataset due to Saunders[1] of only 5903 individuals—a relatively small dataset in this context—the maximal element of $K(D, A)$ is about 1.435×10^{1165}. The

[1] The dataset comprises species counts on kelp holdfasts; here `saunders.exposed.tot` of package **untb** (Hankin 2007a), is used.

accuracy of package **Brobdingnag** in this context may be assessed by comparing it with that computed by PARI/GP (Batut, Belabas, Bernardi, Cohen, and Olivier 2000) with a working precision of 100 decimal places; the natural logs of the two values are 2682.8725605988689 and 2682.87256059887 respectively: identical to 14 significant figures.

4. Conclusions

The **Brobdingnag** package allows representation and manipulation of numbers larger than those covered by standard double precision arithmetic, although accuracy is eroded for very large numbers. This facility is useful in several contexts, including combinatorial computations such as encountered in theoretical modelling of biodiversity.

Acknowledgments

I would like to acknowledge the many stimulating and helpful comments made by the R-help list over the years.

References

Abramowitz M, Stegun IA (1965). *Handbook of Mathematical Functions*. New York: Dover.

Batut C, Belabas K, Bernardi D, Cohen H, Olivier M (2000). "User's Guide to PARI/GP." Technical Reference Manual. URL http://www.parigp-home.de/.

Etienne RS (2005). "A New Sampling Formula for Neutral Biodiversity." *Ecology Letters*, **8**, 253–260. doi:10.111/j.1461-0248.2004.00717.x.

Hankin RKS (2006). "Additive Integer Partitions in R." *Journal of Statistical Software*, **16**(Code Snippet 1).

Hankin RKS (2007a). "Introducing **untb**, an R package for simulating ecological drift under the Unified Neutral Theory of Biodiversity."

Hankin RKS (2007b). "Urn Sampling Without Replacement: Enumerative Combinatorics in R." *Journal of Statistical Software*, **17**(Code Snippet 1).

Hankin RKS (2007c). "Very Large Numbers in R: Introducing Package **Brobdingnag**." *R News*, **7**(3), 15–16. URL http://CRAN.R-project.org/doc/Rnews/.

Hubbell SP (2001). *The Unified Neutral Theory of Biodiversity and Biogeography*. Princeton University Press.

Swift J (1726). *Gulliver's Travels*. Benjamin Motte.

Affiliation:

Robin K. S. Hankin
The University of Cambridge

19 Silver Street
Cambridge CB3 9EP
United Kingdom
E-mail: rksh1@cam.ac.uk