Demographic: Classes and Methods for Data about Populations

John Bryant[1][*]

1. Statistics New Zealand
[*]Contact author: john.bryant@stats.govt.nz

Population data cross-classified by variables such as age, sex, and time is ubiquitous in official statistics and elsewhere. The presentation introduces **Demographic**, an *R* package under development that aims to ease some of the burden of manipulating this sort of data.

It is natural to represent cross-classified data as a multidimensional array, and to describe operations on cross-classified data as manipuations of arrays, such as splitting, collapsing, or expanding. Package **Demographic** facilitates this approach by providing an S4 class, `Demographic`, that builds on standard *R* arrays. Among other things, an object of class `Demographic`

- holds metadata such as the precise definitions of age intervals, or key words describing of the nature of each dimension (eg 'time', 'origin', or 'category'),

- has methods for most standard functions, many of which take advantage of the metadata by, for instance, checking that age intervals align correctly,

- encourages the use of names rather than numeric indices, in the interest of more transparent, less error-prone code,

- permits data.frame-like expressions such as `subset(x, age >= 65)`.

There are two main subclasses of `Demographic`: `Counts` and `Rates`. Objects from these two classes behave differently, reflecting differences in the way that analysts conventionally treat counts and rates data. For instance, when one `Counts` object is divided by another `Counts` object, a `Rates` object is produced, and if one of the `Counts` objects has a dimension that the other does not, the extra dimension is collapsed before the division is carried out.

The presentation will show how **Demographic** can be used for common tasks such as tidying messy data or doing simple projections. It will also provide an example of how a specialised package can take advanage of the general-purpose services provided by **Demographic**.