Multiple-table data in R

SC Walker

September 1, 2011

The standard data management paradigm in R is based on `data.frame` objects, which are two dimensional tables with rows and columns representing replicates and variables respectively. My experience in the field of community ecology has led me to data sets that do not easily fit within this paradigm. A common example is the fourth-corner problem (refs?), in which three tables are collected: a sites-by-species table of abundances or occurrences; a table of environmental variables at each site; and a table of traits for each species (Fig. 1). Such data are characterized by a conspicuous (lower-right) 'fourth-corner', where there are no data. And this fourth-corner problem will not go away if we somehow rearrange the data. It is not possible to put such a data set into a data frame without large holes of missing values. The fourth-corner problem is therefore inherently multi-tabular. Hence, it is not obvious how best to include such data into an R workflow.

One possible solution is to develop new R analysis functions—or new software packages altogether—that are specifically designed to accept several tables as input. There has been a fair amount of work in this direction, focusing on data with a fourth-corner problem (e.g. Chessel et al. 1996; Legendre et al. 1997; Ives and Godfray 2006; Dray and Legendre 2008; Pillar and Duarte 2010; Leibold et al. 2010; Ives and Helmus 2011). However, this work does not apply to data sets that have other more complex multiple-table data structures (e.g. zooplankton communities in Lac Croche, Fig. 2). One approach to such issues would be to build new data analysis functions for each new data structure. But such an approach is less than ideal, as it would require that new methods be learned for each new structure. The `multitable` package provides an alternative approach, by introducing a multiple-table generalization of data frames—called data lists—which can be analyzed with virtually any function that can be used to analyze a data frame. Thus, instead of providing new methods of analysis, `multitable` provides new methods of data management.

There are several existing R packages that are designed to make data management easier (e.g. `reshape2`; etc.??). In particular, the `mefa` and `mefa4` packages have been developed to organize data with a slight generalization[1] of the fourth-corner problem. The `multitable` package has much in common with `mefa`, but there are noticeable differences; for example, `mefa` provides more extensive tools

[1] Several community matrices—called segments—with identical dimensions are allowed in `mefa`.

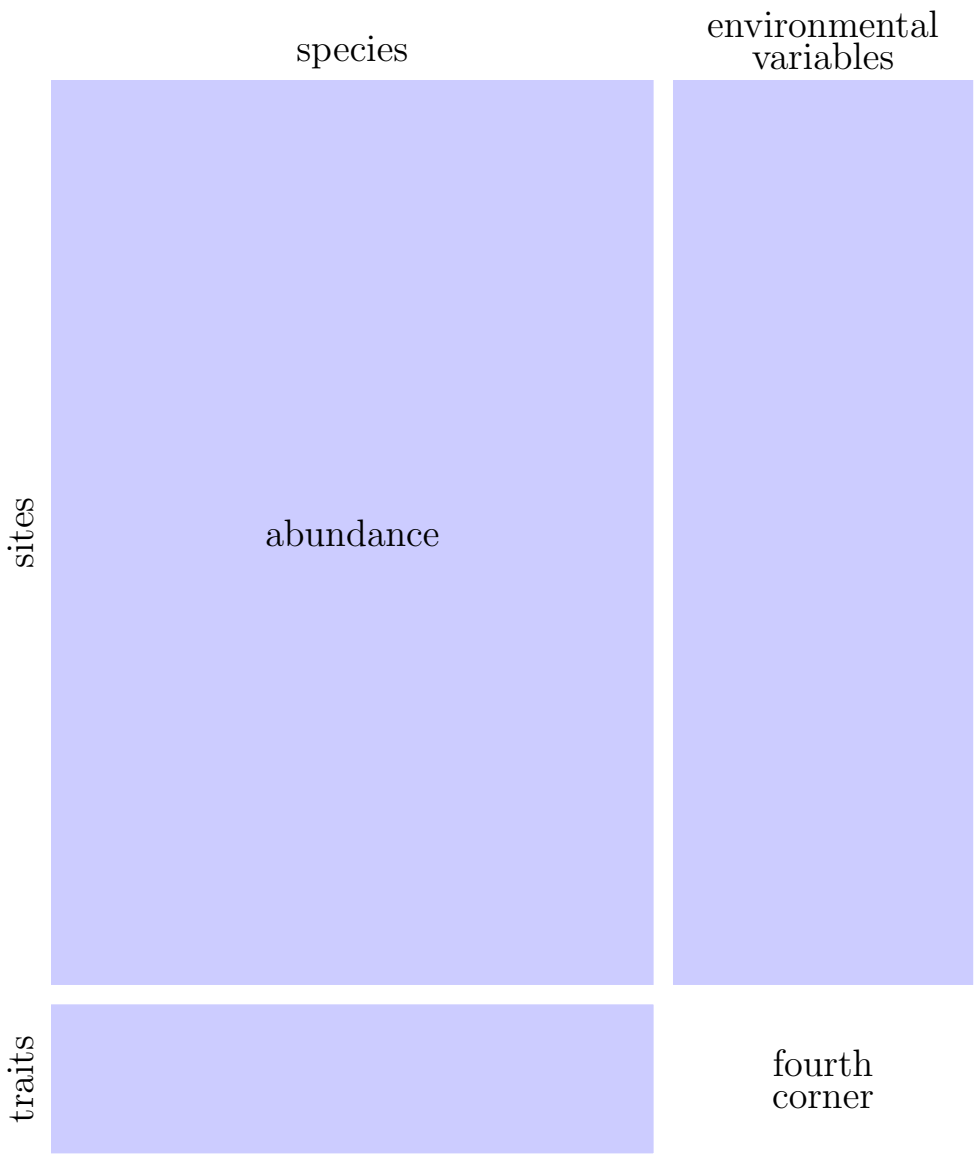

Figure 1: Fourth corner problem.

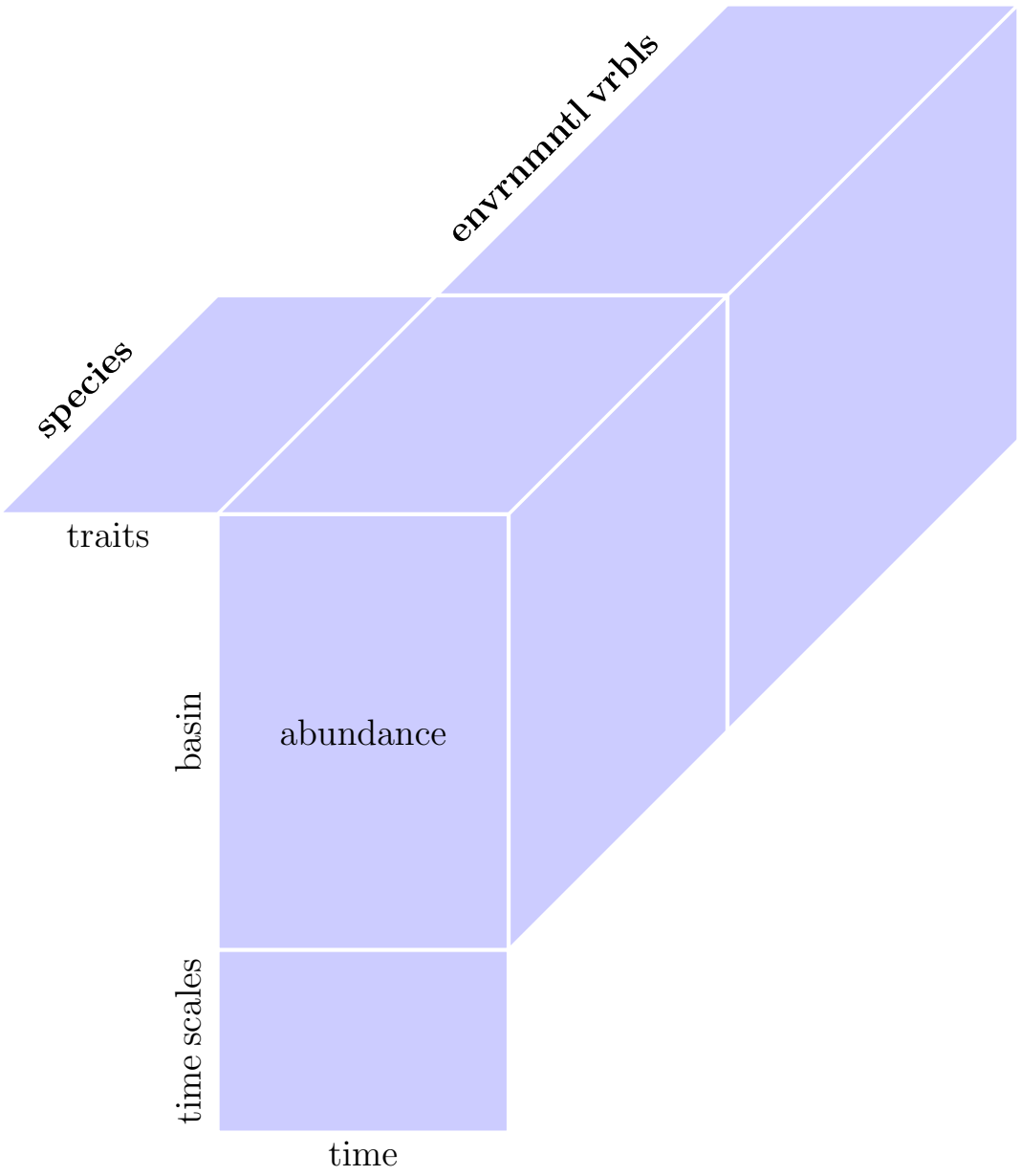

Figure 2: The structure of the Lac Croche zooplankton community data.

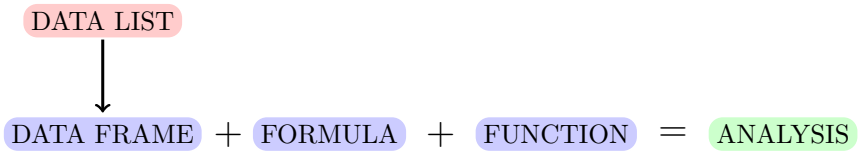

Figure 3: The `multitable` paradigm for including multiple-table data into the standard R workflow. Data lists are used to organize and manipulate multiple-table data. When such data are required for analysis, they are coerced into a data frame. Once in data frame form, they can be used in analyses by combining them with formulas (to specify hypothetical relationships between variables) and functions (to call computational methods).

for data summarization than `multitable`, while `multitable` is designed to handle more general data structures than `mefa`. However, we recognize that `mefa` and `multitable` will be complementary, not competitive.

The specific aim of the `multitable` package is to make multiple-table data analysis as similar as possible to single-table analysis in R. The standard single-table R workflow involves organizing data into a data frame; expressing hypotheses about the relationships between variables in the data frames via `formula` objects; and combining data frames and formulas by passing them to functions that produce analyses (e.g. plots; fitted models; summary statistics). This framework allows ecologists to concentrate on their primary interests—the relationships between ecological variables—without explicit reference to complex mathematical and algorithmic details. It also provides access to those details, which are required (1) for more effective analyses and (2) to develop new methods of analysis within the framework. As new methods are developed, researchers simply pass their data frames to new functions in much the same way they would pass them to older functions. Thus, by separating low-level methods development from high-level data analysis, R fosters the formation of a community of researchers where both methodologists and analysts can have mutually beneficial interactions. Our overarching design principle is to use this standard R paradigm with multiple-table data, even though such data do not fit into data frames.

1 Organizing multiple-table data in data list objects

The `multitable` model of data management is illustrated in Figure 3.