Duncan Temple Lang, University of California at Davis

Table of Contents

The *save*() and corresponding *load*() functions allow people to easily collect one or more R objects in a single portable archive and to distribute these and restore them in different R sessions. Often people store a single object in such a file and the name of the file is sufficient to describe the contents. In other cases, however, there are many variable stored in a single object and the name does not adequately describe the individual elements. To find the names of the variables in such a file, we have to restore the contents of the entire file. This can be time consuming and unnecessary as we may not want the values themselves, but just the names. Even when we want the contents, we may not want the value for all variables, but just a couple. Again, we have to read the entire contents of the file, allocating memory unnecessarily.

In this short paper, we discuss different approaches to provide better access to i) accessing the names of the variables within an R data file, ii) accessing individual elements directly without having to read through the entire file up to the object of interest. It takes only a moment to see that if we added a character vector containing the names of the objects within the file, we could access this directly and use this to address i), i.e. providing a table of contents for the RDA file. If we also included the offset in bytes from the start of the RDA file at which each of the top-level objects started, we could use that to rapidly access and extract individual elements within the RDA file. We know the names of the variables before we write the RDA file. We don't know the offsets of the i-th variable until we write the first i - 1 elements to the file. We can put the names of the variables into a "hidden" variable at the beginning of the file. It makes most sense to put the offsets as a value at the end of the RDA file. Since we place the offsets at the end, we might also place the names at the end and leave the original format as is and merely append to the end of that format.

One approach to this format is to add the character vector of names as the 3rd last element of the RDA file, after all the regular objects are written. We can put the offsets after this and include the offset of the character vector of names within that vector. We need one other piece of information.

We could avoid adding the vector of names. Instead, once we can locate the offsets, we can jump to each of these and read the SYMSXP immediately there. Unfortunately, the symbol may not be located immediately at the offset as some objects may have attributes that are written out before the object itself, e.g. LANGSXP, PROMSXP, DOTSXP and LISTSXP - all somewhat non-standard objects. There are also issues with references to a common table of shared objects.

Instead of merely including the vector of names of the objects and a separate vector of offsets, we might include a data frame with a row for each object. We might include the the class, type, length and name of the object. We might also include whether it is an S4 object, whether it has attributes.