1 setRNG Functions

In R, the functions in this package are made available with

> library("setRNG")

As of R-2.1.0 the code from the vignette that generates this guide can be
loaded into an editor with edit(vignette("setRNG")). This uses the default edi-
tor, which can be changed using options(). Also, it should be possible to view
the pdf version of the guide for this package with print(vignette("setRNG")).

This library provides tools to simplify recording and resetting the random
number generator, to help make monte carlo experiments easily reproducible. It
uses the R/S tools for setting the seed, but also records and sets the mechanism
for converting uniform numbers to normally distributed numbers. (It could be
extended to other transformations, but I have not done that.)

The setRNG function would typically be called by simulation programs. For
example, if rng=NULL is an argument to the function then the code

> if(!require("setRNG")) {
    stop("This function requires the setRNG package.")
} else
> if(is.null(rng)) rng <- setRNG() else
    {old.rng <- setRNG(rng); on.exit(setRNG(old.rng))}

should be used before the random number generator is used. This will set the
RNG information if given, and in all cases record the RNG information which
can then be returned with the result of the simulation. (setRNG() returns the
setting so do not skip this if rng=NULL.) With the information recorded the
simulation can always be reproduced if necessary. In the case where the rng is
set to a specified value it is good practice to set it back to the original value on
exit. This prevents other random experiments from accidentally being affected
by the rng setting.

The library also implements an approach to random number generation
which allows the same random experiments to be replicated in S and R. The
functions in the S directory allow the R results using Wichmann-Hill and Box-
Muller to be replicated in S. These were done with the aid of an example from
B. D. Ripley. (The files in the S directory of the package are for use with S not
R.) These functions are intended primarily as a way to confirm that simulations
and estimations with simulated data work in the same way in both S and R,
not as an improved RNG. (It has only been tested in Splus 3.3) Default and
other RNGs can still be used and are preferred for both speed and theoretical
reasons.