How to use the **ss3sim** package to run simulations in SS3

Sean C. Anderson, ...

First start by installing the latest version of ss3sim and loading the package:

```
> # install.packages(c("r4ss", "MCMCpack")) # dependencies, if needed
> # install.packages("devtools")
> # devtools::install_github("ss3sim", username="seananderson")
> library(ss3sim)
```

You can read the help files and access this vignette again with:

```
> help(package = "ss3sim")
> vignette("ss3sim")
```

Setting up the file structure

This package is set up assuming that you have an established base case operating model and estimation model to work with. Each operating model and estimation model should be in their own folder. The operating model folder should have the files:

```
yourmodel.ctl
yourmodel.dat
ss3.par
starter.ss
forecast.ss
```

The estimation model folder should have:

```
yourmodel.ctl
yourmodel.dat # optional; not used
starter.ss
forecast.ss
```

In both cases, nothing more and nothing less. The names of the `.ctl` and `.dat` files are not important. The package functions will rename them after they are copied to appropriate folders. These files should be derived from the `.ss_new` files but named as listed above. It's important to use these `.ss_new` files so they have consistent formatting. Many of the functions in this package depend on that formatting.

For the purposes of the Fish 600 project, we have unique case identifiers which combine to create unique scenarios. The types of cases are: natural mortality (M), fishing mortality (F), data quality (D), and retrospective (R). And the species are cod (cod), sardine-like (sar), and flatfish (fla). It's important to use these three letter abbreviations for the species since the functions assume that the last three letters represent a species (or some other identifier for a different project).

The different version of each case are identified with numbers. So, for example, the base case scenario for cod is identified as `M1-F1-D1-R1-cod`. We will have a spreadsheet that describes each of these in plain language.

The function `copy_ss3models` creates a folder structure and copies over the operating and estimation models. The folder structure looks like:

```
M1-F1-D1-R1-cod/1/om
M1-F1-D1-R1-cod/1/em
M1-F1-D1-R1-cod/2/om
M1-F1-D1-R1-cod/2/em
...
```

If you are using bias correction (`bias_correct = TRUE`) then there will be some additional folders. In that case the folders will look like:

```
M1-F1-D1-R1-cod/bias/1/om
M1-F1-D1-R1-cod/bias/1/em
M1-F1-D1-R1-cod/bias/2/om
M1-F1-D1-R1-cod/bias/2/em
...
M1-F1-D1-R1-cod/1/om
M1-F1-D1-R1-cod/1/em
M1-F1-D1-R1-cod/2/om
```

```
  M1-F1-D1-R1-cod/2/em

   ...
```

Note that the operating and estimating model folders have been renamed `om` and
`em` within each iteration, the operating and estimation models have been checked to
make sure they contain the minimal files (as listed above), the filenames have been
made all lowercase, the data file has been renamed `data.dat`, the control files have
been renamed `om.ctl` or `em.ctl`, and the starter and control files have been adjusted
to reflect these new file names.

   The functions in this package assume you've set your working directory in R to
be the base folder where you will store the scenario folders.

Creating the input configuration files

You will need to have a folder containing "case" argument definitions. These plain
text files are read by `get_caseval` and turned into argument lists that are passed to
`run_ss3sim`. You can create template input files by running `create_argfiles`. It
reads the various functions and parses the arguments and default values into plain
text files. The default settings create these files:

1. `M0-spp.txt`

2. `F0-spp.txt`

3. `index0-spp.txt`

4. `agecomp0-spp.txt`

5. `lcomp0-spp.txt`

6. `R0-spp.txt` (not implemented yet)

   Look in your working directory for the template files. Change the case ID number
(defaults to `0`) and the species identifier to a three letter identifier. For the Fish 600
project use one of `cod`, `sar`, or `fla` for cod, sardine, or flatfish. An example filename
would be `M1-sar.txt` or `lcomp2-fla.txt`. The case `D1` corresponds to the files
`index1-spp.txt`, `agecomp1-spp.txt`, and `lcomp0-spp.txt`. The other case types
have single argument files.

   The first column in the text files denotes the argument to be passed to a function.
The second argument denotes the value to be passed. You can use any simple R syntax. For example: `c(1, 2, 4)`, or `seq(1, 100)` or `1:100` or `matrix()`. Character

objects don't need to be quoted, but can be if you'd like. However, be careful not to use the delimiter (set up as a semicolon) anywhere else in the file besides to denote columns. You can add comments after any **#** symbols. Internally, the functions evaluate in R any entries that have no character values (e.g. `1:100`) or have an alphanumeric character followed by a (. Anything that is character only or has character mixed with numeric but doesn't have the regular expression `"[A-Za-z0-9](" ` gets turned into a character argument.

Putting that all together, here's what an example `F1-cod.txt` file might look like:

```
years; 1:100
years_alter; NA
fvals; NA
```

Running the models

The `run_ss3sim` function is a wrapper function. It adjusts the natural mortality, adjusts the fishing mortality, adds recruitment deviations, calls `run_ss3model` to run the operating model, samples various survey estimates from the operating model, changes the age composition data as specified, changes the length composition data as specified, copies and renames files as necessary, and calls `run_ss3model` again to run the estimation model.

The `run_fish600` function is a high-level wrapper that deals with parsing the case input files and then passes these arguments on to `run_ss3sim`.

Say you have your input case files setup and you want to run the first 50 iterations of those scenarios. You could run them like this:

```
> # First grab the example package data:
> d <- system.file("extdata", package = "ss3sim")
> f <- paste0(d, "/run_ss3sim_eg/")
> om_model_dir <- paste0(f, "cod_om")
> em_model_dir <- paste0(f, "cod_em")
> case_folder <- paste0(f, "case-arguments")
> #
> #
> # Without bias correction:
> run_fish600(iterations = 1, scenarios = c("M1-F1-D1-R1-cod"),
+ case_folder = case_folder, om_model_dir = om_model_dir,
```

```
+ em_model_dir = em_model_dir)
> #
> #
> # With bias correction:
> # (Note that bias_nsim should be bigger, say 5, but it is set to 1
> # here so the example runs faster.)
> run_fish600(iterations = 1, scenarios = c("M1-F1-D1-R1-cod"),
+ case_folder = case_folder, om_model_dir = om_model_dir,
+ em_model_dir = em_model_dir, bias_correct = TRUE,
+ bias_nsim = 1)
```
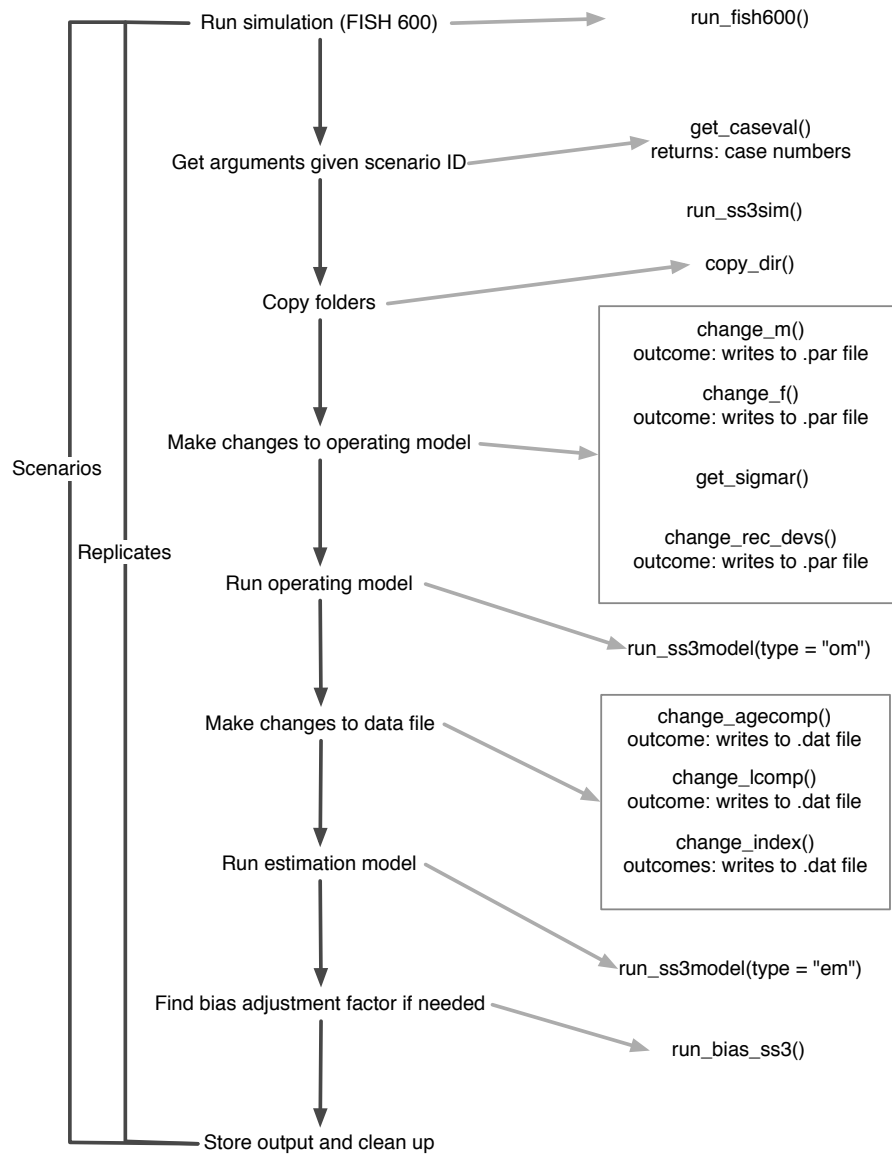
Figure 1: Simulation steps. Higher-level function calls are shown on the right.