Package: ss3om (via r-universe)

August 13, 2024

Title Tools for Conditioning Fisheries Operating Models Using Stock Synthesis 3

Version 0.5.2.9016

Description Tools for loading Stock Synthesis (SS3) models into FLR. Used in conditioning of Operating Models based on SS3 by considering structural uncertainty in input parameters and assumptions. A grid of SS3 runs can be created and results loaded on objects of various FLR classes.

X-schema.org-keywords fisheries, ss3, flr, R

Depends R (>= 3.3.2), FLCore (>= 2.6.5), r4ss (>= 1.43.0)

Imports methods, stats, data.table, foreach, FLFishery, mse, mvtnorm

Additional_repositories `https://flr.r-universe.dev`

Remotes https://r4ss.r-universe.dev

BugReports `https://github.com/flr/ss3om/issues`

Suggests doParallel, testthat, rlang, knitr, rmarkdown

License EUPL

VignetteBuilder knitr

Roxygen list(markdown = TRUE)

Repository https://flr.r-universe.dev

RemoteUrl https://github.com/flr/ss3om

RemoteRef HEAD

RemoteSha 2b2b421ab313f133f613ad79987fe662ca417b87

Contents

extractSSB 2
nameGrid 3
readFLIBss3 3
readFLSRss3 4
readFLSss3 5
ss3slot 6

Index **9**

`extractSSB` *Extracts derived quantities from SS output*

Description

A number of derived quantities are available in the *Report.sso* file, and are useful for checking that the generated FLR objects lead to the same values. These functions are called on the list returned by *readOutputss3* or r4ss::SS_output, and extract the yearly values from the following rows in the *derived_quants* data.frame:

Usage

```
extractSSB(out)

extractRec(out)

extractFbar(out)

extractZatage(out)

extractDevs(out)
```

Arguments

`out`	A list as returned by r4ss::SS_output.

Details

- *extractSSB*: `SSB_y` for y between `startyr` and `endyr`.
- *extractRec*: `Recr_y` for y between `startyr` and `endyr`.
- *extractFbar*: `F_y` for y between `startyr` and `endyr`.
- *extractZatage*: `Z_ay` for y between `startyr` and `endyr` and ages but last.

For 2 sex models (`nsexes`), *extractRec* will return a two-unit *FLQuant*, with the `Recr_y` values split according to the recruitment sex ratio. This is extracted from `recruitment_dist[[1]][, "Frac/sex"]`.

The value returned by `extractFbar` is the actual mean F over the age range, and the value in `derived_quants` is corrected according to `F_report_basis` for models where F reporting basis is F/FMSY.

Value

An *FLQuant* object of the requested quantity.

Examples

```
out <- readOutputss3(system.file("ext-data", "herring", package="ss3om"))

extractSSB(out)

extractFbar(out)
```

`nameGrid` *nameGrid*

Description

Creates folder names from a 'grid' df of scenarios

Usage

```
nameGrid(df, dir, from = 1)
```

Arguments

`df`	Model grid data.frame
`dir`	Folder name
`from`	Starting number

`readFLIBss3` *A function to read the CPUE series from an SS3 run into an `FLIndex` object*

Description

A function to read the CPUE series from an SS3 run into an `FLIndex` object

Usage

```
readFLIBss3(
  dir,
  fleets,
  birthseas = out$birthseas,
  repfile = "Report.sso",
  compfile = "CompReport.sso",
  ...
)
```

Arguments

`dir` Directory containing the SS3 output files

`birthseas` The birthseasons for this stock as a numeric vector.

`...` Any other argument to be passed to `r4ss::SS_output`

Value

An object of class FLStock

Author(s)

Iago Mosqueira, EC JRC

References

Methot RD Jr, Wetzel CR (2013) Stock Synthesis: A biological and statistical framework for fish stock assessment and fishery management. Fisheries Research 142: 86-99.

See Also

FLComp

`readFLSRss3` *A function to read the stock-recruit relationships from an SS3 run into an `FLSR` object*

Description

A function to read the stock-recruit relationships from an SS3 run into an `FLSR` object

Usage

```
readFLSRss3(
  dir,
  birthseas = out$birthseas,
  repfile = "Report.sso",
  compfile = "CompReport.sso",
  ...
)
```

Arguments

`dir` Directory containing the SS3 output files

`birthseas` The birthseasons for this stock as a numeric vector.

`...` Any other argument to be passed to `r4ss::SS_output`

Value

An object of class FLStock

Author(s)

Iago Mosqueira, EC JRC

References

Methot RD Jr, Wetzel CR (2013) Stock Synthesis: A biological and statistical framework for fish stock assessment and fishery management. Fisheries Research 142: 86-99.

See Also

FLComp

`readFLSss3` *A function to read SS3 results as an FLStock object*

Description

Results of a run of the Stock Synthesis sofware, SS3 (Methot & Wetzel, 2013), can be loaded into an object of class `FLStock`. The code makes use of the r4ss::SS_output function to obtain a list from Report.sso. The following elements of that list are used to generate the necessary information for the slots in `FLStock`: "catage", "natage", "ageselex", "endgrowth", "catch_units", "nsexes", "nseasons", "nareas", "IsFishFleet", "fleet_ID", "FleetNames", "spawnseas", "inputs" and "SS_version".

Usage

```
readFLSss3(
  dir,
  repfile = "Report.sso",
  compfile = "CompReport.sso",
  wtatage = out$wtatage_switch,
  ...
)
```

Arguments

`dir`	Directory holding the SS3 output files
`...`	Any other argument to be passed to `r4ss::SS_output`
`birthseas`	Birth seasons for this stock, defaults to spawnseas
`name`	Name of the output object to fil the name slot
`desc`	Description of the output object to fill the desc slot

Value

An object of class \link{FLStock}

Author(s)

The FLR Team

References

Methot RD Jr, Wetzel CR (2013) Stock Synthesis: A biological and statistical framework for fish stock assessment and fishery management. Fisheries Research 142: 86-99.

See Also

FLComp

---

ss3slot *Functions to convert SS3 output into FLQuant(s)*

Description

A series of auxiliary functions that convert one or more elements, typically of class `data.frame`. in the list returned by `r4ss::SS_output` into particular FLQuant or FLQuants objects.

Usage

```
ss3index(cpue, fleets)

ss3index.res(cpue, fleets)

ss3index.var(cpue, fleets)

ss3index.q(cpue, fleets)

ss3sel.pattern(selex, years, fleets, morphs, factor = "Asel2")

ss3wt(endgrowth, dmns, birthseas)

ss3mat(endgrowth, dmns, birthseas, option = 3)

ss3m(endgrowth, dmns, birthseas)

ss3n(n, dmns, birthseas)

ss3catch(catage, wtatage, dmns, birthseas, idx)

ss3mat30(endgrowth, dmns, spawnseas, option = 3)
```

```
ss3m30(endgrowth, dmns, birthseas)

ss3n30(n, dmns)

ss3catch30(catage, wtatage, dmns, birthseas, idx)
```

Arguments

`cpue`	A data frame obtained from SS_output$cpue.
`fleets`	Named vector of fleets (numeric) codes
`selex`	A data frame obtained from SS_output$ageselex.
`years`	Vector of years for which the index applies
`morphs`	Vector of morphs to use
`endgrowth`	A data frame obtained from SS_output$endgrowth.
`dmns`	dimnames of the output object, usually obatined using `getDimnames`.
`birthseas`	The birthseasons for this stock as a numeric vector.
`n`	A data frame obtained from SS_output$natage.
`catage`	A data frame obtained from SS_output$catage.
`wtatage`	A data frame obtained from SS_output$endgrowth but subset for `birthseas` and `RetWt:_idx`.
`idx`	The fishing fleets, as in `SS_output$fleet_ID[SS_output$IsFishFleet]`.

Details

- `ss3index` returns the `index` slot of each survey/CPUE fleet.
- `ss3index.res` returns the `index.res` slot of each survey/CPUE fleet.
- `ss3index.var` returns the `index.var` slot of each survey/CPUE fleet.
- `ss3index.q` returns the `index.q` slot of each survey/CPUE fleet.
- `ss3sel.pattern` returns the `sel.pattern` slot of each survey/CPUE fleet.
- `ss3wt` returns the `stock.wt` slot.
- `ss3mat` returns the `mat` slot.
- `ss3m` returns the `m` slot.
- `ss3m` returns the `m` slot.
- `ss3catch` currently returns the `landings.n` slot, equal to `catch.n` as discards are not being parsed.
- `ss3mat30` returns the `mat` slot.

- `ss3m` returns the `m` slot.
- `ss3n30` returns the `stock.n` slot.
- `ss3catch` currently returns the `landings.n` slot, equal to `catch.n` as discards are not being parsed.

Value

An FLQuant or FLQuants object, depending on the converted data structure

Author(s)

Iago Mosqueira, EC JRC D02

See Also

`FLQuant` `readFLSss3`

Index

∗ **classes**

readFLIBss3, 3
readFLSRss3, 4
readFLSss3, 5
ss3slot, 6

extractDevs *(*extractSSB*)*, 2
extractFbar *(*extractSSB*)*, 2
extractRec *(*extractSSB*)*, 2
extractSSB, 2
extractZatage *(*extractSSB*)*, 2

FLComp, *4–6*
FLQuant, *8*
FLStock, *4*, *5*

nameGrid, 3

readFLIBss3, 3
readFLSRss3, 4
readFLSss3, 5, *8*

ss3catch *(*ss3slot*)*, 6
ss3catch30 *(*ss3slot*)*, 6
ss3index *(*ss3slot*)*, 6
ss3m *(*ss3slot*)*, 6
ss3m30 *(*ss3slot*)*, 6
ss3mat *(*ss3slot*)*, 6
ss3mat30 *(*ss3slot*)*, 6
ss3n *(*ss3slot*)*, 6
ss3n30 *(*ss3slot*)*, 6
ss3sel.pattern *(*ss3slot*)*, 6
ss3slot, 6
ss3wt *(*ss3slot*)*, 6