Package: FLSRTMB (via r-universe)

November 26, 2024

Title FLSR in TMB

Version 1.1.4.9015

Description Estimates FLR spawner recruitment relationships in TMB

X-schema.org-keywords stock-recruit, fisheries, flr, R, TMB, AD

License EUPL

Encoding UTF-8

LazyData true

Roxygen list(markdown = TRUE)

RoxygenNote 7.3.3

Depends FLCore (>= 2.6.16), ggplot2, ggplotFL, doFuture

Imports TMB (>= 1.7.19), progressr, scales

LinkingTo TMB, RcppEigen

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

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

RemoteRef HEAD

RemoteSha 4099048f5ebe719a0175b9e51ba7006d892a7808

Contents

`blimprior` *blimprior plot Plots the bounds of the hockey-stick break-point*

Description

blimprior plot Plots the bounds of the hockey-stick break-point

Usage

```
blimprior(
  lplim = 0.01,
  uplim = 0.3,
  s.logitsd = 50,
  par = "plim",
  bias.correct = TRUE
)
```

Arguments

`lplim`	steepness, default 0.6 for a approx. uniform prior with s.logistsd = 20
`uplim`	s steepness, default 20 for a approx. uniform prior with s = 0.6
`par`	parameter on x-axis default "plim", else "sstar"
`s.sdlogit`	default 20

Value

ggplot

Examples

```
blimprior() # approx. uniform with some curving on bounds
blimprior(lplim=0.001,uplim=0.3,s.logitsd=20)
# Non-bias corrected
blimprior(lplim=0.001,uplim=0.3,s.logitsd=20,bias.correct=FALSE)
```

bootstrapSR *Bootstrap fits of mutliple stock-recruits relationships*

Description

Definition ...

Usage

```
bootstrapSR(
  x,
  iters = 100,
  method = c("best", "logLik", "relative"),
  models = c("bevholt", "ricker", "segreg"),
  verbose = TRUE,
  ...
)
```

Arguments

x	An object of class 'FLStock'.
models	Name(s) of model(s) to fit, 'character'. See Details.
verbose	Should progress be reported, 'logical'.
iter	Number of bootstrap iterations, 'numeric'.

Details

The returned 'FLPar' object contains

Value

A list with elements An object or class 'FLPar' containing the estimated paramaters.

Author(s)

Iago Mosqueira (WMR)

See Also

FLSR, linksrrTMB

Examples

```
data(ple4)
mods <- bootstrapSR(ple4, iter=50, model=c("bevholt", "segreg"))
```

dprior *dprior plot Plots the logit prior distribution for depensation*

Description

dprior plot Plots the logit prior distribution for depensation

Usage

```
dprior(d = 1, d.logitsd = 100, ll = 0.5, ul = 3)
```

Arguments

d	depensation, default 1 for a approx. uniform prior with s.logistsd = 20
d.logitsd	dependation sd, default 20 for a approx. uniform prior with s = 0.6
ll	lower bound of d = 0.25
ul	lower bound of d = 4

Value

ggplot

Examples

```
dprior() # approx. uniform with some curving on bounds
dprior(d=1,d.logitsd=2)
```

from_logitd *from_logitd()*

Description

convert depensation d from logit

Usage

```
from_logitd(logit_d, ll = 0.5, ul = 3)
```

Arguments

logit_d	logit(depensation)
ll	defines lower prior bound of d
ul	defines upper prior bound of d

Value

depensation d

from_logits *from_logits()*

Description

convert steepness from logit

Usage

```
from_logits(logit_s, ll = 0.2, ul = 1)
```

Arguments

logit_s	logit(steepness)
ll	defines lower prior bound of s
ul	defines upper prior bound of s

Value

steepness s

gm *gm()*

Description

Generic geometric mean function

Usage

```
gm(x)
```

Arguments

x	object

Value

FLQuant with annual sprFy

hsblim *hsblim()*

Description

helper to quickly extract blim, r0 and srp0 = blim/b0 from hockeystick

Usage

```
hsblim(object)
```

Arguments

object	class FLSR fitted with model segreg

Value

FLPar

plotjitter *plots Jitter results*

Description

plots Jitter results

Usage

```
plotjitter(jitter)
```

Arguments

jitter	output from srrjitter()

Value

ggplot

Examples

```
data(ple4)
hs = srrTMB(as.FLSR(ple4,model=segreg),spr0=spr0y(ple4),lplim=0.07,uplim=0.2)
plotsrs(hs)
jitter = srrjitter(hs)
plotsrs(jitter$groups)
plotjitter(jitter)
bh = srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=spr0y(ple4))
jitbh = srrjitter(bh)
plotjitter(jitbh)
plotsrs(jitbh$groups)
```

`plotsrs` *plotsrs based on plot(FLSRs) Plots FLSRs, i.e. multiple S-R relationships*

Description

plotsrs based on plot(FLSRs) Plots FLSRs, i.e. multiple S-R relationships

Usage

```
plotsrs(object, path = TRUE, b0 = FALSE, rel = FALSE)
```

Arguments

`object`	of class FLSRS
`path`	connect points sequentially

Value

ggpplot

Examples

```
data(ple4)
bh <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
plotsrs(bh)
plotsrs(bh,b0=TRUE) # plot through to B0
plotsrs(bh,b0=TRUE,rel=TRUE) # plot relative to B0
# Try more models
hs <- srrTMB(as.FLSR(ple4,model=segreg),spr0=(spr0y(ple4)),plim=0.05,pmax=0.2)
ri <- srrTMB(as.FLSR(ple4,model=rickerSV),spr0=mean(spr0y(ple4)))
bh.tv <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=spr0y(ple4))
ri.tv <- srrTMB(as.FLSR(ple4,model=rickerSV),spr0=spr0y(ple4))
all= FLSRs(hs=hs,bh=bh,ri=ri,bh.tv=bh.tv,ri.tv=ri.tv)
plotsrs(all)
plotsrs(all,path=FALSE)
# plot through to b0
```

```
plotsrs(all,b0=TRUE)
# plot all relative relative to B0
plotsrs(all,rel=TRUE)
plotsrs(all,rel=TRUE,b0=TRUE)
```

---

plotsrts *Plots time series of observed and predicted recruitment*

Description

Plots time series of observed and predicted recruitment

Usage

```
plotsrts(object, relative = TRUE)
```

Arguments

object	Input FLSR of FLSRs object

Value

ggplot

Examples

```
data(ple4)
bh <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
plotsrts(bh)
# Try more models
hs <- srrTMB(as.FLSR(ple4,model=segreg),spr0=(spr0y(ple4)),plim=0.05,pmax=0.2)
ri <- srrTMB(as.FLSR(ple4,model=rickerSV),spr0=mean(spr0y(ple4)))
bh.tv <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=spr0y(ple4))
ri.tv <- srrTMB(as.FLSR(ple4,model=rickerSV),spr0=spr0y(ple4))
all= FLSRs(hs=hs,bh=bh,ri=ri,bh.tv=bh.tv,ri.tv=ri.tv)
plotsrts(all)
do.call(c,lapply(all,function(x)AIC(x))) # AIC
```

productivity *productivity()*

Description

Function to compute r, generation time (gt) and annual reproductive rate (alpha)

Usage

```
productivity(object, s = 0.7)
```

Arguments

object	class FLStock
s	steepness of the stock recruitment relationship

Value

FLQuants with FLQuant r, gt and alpha

Author(s)

Henning Winker and Laurence Kell

spr0y *spr0y()*

Description

Function to compute annual spr0

Usage

```
spr0y(object, byage = FALSE, simplify = TRUE)
```

Arguments

object	class FLStock
byage	if TRUE it return spr0_at_age

Value

FLQuant with annual spr0y

Author(s)

Laurence Kell

sprFy *sprFy()*

Description

Function to compute annual sprF as function of F_a

Usage

```
sprFy(object, byage = FALSE)
```

Arguments

object	class FLStock
byage	if TRUE it return sprF_at_age

Value

FLQuant with annual sprFy

Author(s)

Henning Winker and Laurence Kell

sprior *sprior plot Plots the logit prior distribution for steepness*

Description

sprior plot Plots the logit prior distribution for steepness

Usage

```
sprior(s = 0.6, s.logitsd = 20, ll = 0.2, ul = 1)
```

Arguments

s	steepness, default 0.6 for a approx. uniform prior with s.logistsd = 20
s.logitsd	s steepness, default 20 for a approx. uniform prior with s = 0.6
ll	lower bound of s = 0.2
ul	lower bound of s = 1

Value

ggplot

Examples

```
sprior() # approx. uniform with some curving on bounds
sprior(s=0.8,s.logitsd=0.5)
```

`srrjitter` *Jitter of S-R fits*

Description

Jitter of S-R fits

Usage

```
srrjitter(fit, steps = 100)
```

Arguments

`fit`	fit of srrTBM()
`steps`	number of jitter steps

Value

list

Examples

```
data(ple4)
hs = srrTMB(as.FLSR(ple4,model=segreg),spr0=spr0y(ple4),lplim=0.07,uplim=0.5)
plotsrs(hs)
jitter = srrjitter(hs)
plotsrs(jitter$groups)
plotsrs(jitter$best) # Best
# Relax lower bound
plotsrs(FLSRs(init=hs,best=jitter$best))
```

`srrTMB` *Fits Stock Recruitment Relationships (SRR) in TMB*

Description

Fits Stock Recruitment Relationships (SRR) in TMB

Usage

```
srrTMB(object, ...)

## S4 method for signature 'FLSRs'
srrTMB(object, ...)

## S4 method for signature 'FLSR'
srrTMB(
  object,
  spr0 = "missing",
  s = NULL,
  s.est = TRUE,
  s.logitsd = 50,
  r0.pr = "missing",
  lplim = 0.01,
  uplim = 0.3,
  Blim = "missing",
  d = 1,
  d.est = TRUE,
  d.logitsd = 100,
  ld = 0.5,
  ud = 3,
  plim = lplim,
  pmax = uplim,
  nyears = NULL,
  report.sR0 = FALSE,
  inits = NULL,
  lower = NULL,
  upper = NULL,
  SDreport = TRUE,
  verbose = FALSE,
  rm.yrs = "missing",
  bias.correct = TRUE
)
```

Arguments

object	Input FLSR = as.FLSR(stock,model) object with current model options • bevholtSV • bevholtDa • rickerSV • segreg • geomean
spr0	unfished spawning biomass per recruit from FLCore::spr0(FLStock)
s	steepness parameter of SRR (fixed or prior mean)
s.est	option to estimate steepness

s.logitsd	prior sd for logit(s), default is 1.4 (flat) if s.est = TRUE
r0.pr	option to condition models on r0 priors (NULL = geomean)
lplim	lower bound of spawning ratio potential SRP, default 0.0001
uplim	upper bound of plausible spawning ratio potential SRP , default 0.3
Blim	fixing Blim, only works with segreg
d	depensation parameter (default = 1)
d.est	option to estimate depensation d
d.logitsd	priod sd for logit(d)
ld	lower bound of depensation parameter d
ud	upper bound of depensation parameter d
plim	depreciated plim = usrp
pmax	depreciated pmax = lsrp
nyears	yearMeans from the tail used to compute a,b from the reference spr0 (default all years)
report.sR0	option to report s and R0 instead of a,b
inits	option to specify initial values of log(r0), log(SigR) and logit(s)
lower	option to specify lower bounds of log(r0), log(SigR) and logit(s)
upper	option to specify upper bounds of log(r0), log(SigR) and logit(s)
SDreport	option to converge hessian and get vcov
verbose	if TRUE, it shows tracing
rm.yrs	remove recruitment years from model fit
bias.correct	if TRUE, bias correction of the uniform logistic hockey-stick prior for Blim/B0 (recommended)

Value

A list containing elements 'FLSR', of class *FLSR*

Examples

```
data(ple4)
gm <- srrTMB(as.FLSR(ple4,model=geomean),spr0=mean(spr0y(ple4)))
bh <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=spr0y(ple4))
ri <- srrTMB(as.FLSR(ple4,model=rickerSV),spr0=spr0y(ple4))
hs <- srrTMB(as.FLSR(ple4,model=segreg),spr0=spr0y(ple4),lplim=0.05,uplim=0.2)
srs = FLSRs(gm=gm,bh=bh,ri=ri,hs=hs) # combine
plotsrs(srs)
plotsrts(srs)  # relative
plotsrs(srs[2:4],b0=TRUE) # through to B0
plotsrs(srs[2:4],b0=TRUE,rel=TRUE)  # relative
gm@SV # estimates
do.call(rbind,lapply(srs,AIC))
# Bias-correction test for Hockey-Stick
srs=FLSRs(
```

```
 correct = srrTMB(as.FLSR(ple4,model=segreg),spr0=mean(spr0y(ple4)),lplim=0.001,uplim=0.08),
 bias =srrTMB(as.FLSR(ple4,model=segreg),spr0=mean(spr0y(ple4)),lplim=0.001,uplim=0.08,bias.correct=F,s.logitsd
plotsrs(srs,rel=T)
# Depensation
d.srs = FLSRs(
uniform = srrTMB(as.FLSR(ple4,model=bevholtDa),spr0=spr0y(ple4)),
larger1 = srrTMB(as.FLSR(ple4,model=bevholtDa),spr0=spr0y(ple4),ld=1),
prior1 = srrTMB(as.FLSR(ple4,model=bevholtDa),spr0=spr0y(ple4),d=1.5,d.logitsd=1.5),
prior1.5 = srrTMB(as.FLSR(ple4,model=bevholtDa),spr0=spr0y(ple4),d=1.5,d.logitsd=1.5),
fixed1.5 = srrTMB(as.FLSR(ple4,model=bevholtDa),spr0=spr0y(ple4),d=1.5,d.est=FALSE),
fixed2.5 = srrTMB(as.FLSR(ple4,model=bevholtDa),spr0=spr0y(ple4),d=2.5,d.est=FALSE)
)
plotsrs(d.srs)
```

`to_logitd` *to_logitd()*

Description

convert steepness to logit

Usage

```
to_logitd(d, ll = 0.5, ul = 3)
```

Arguments

`d`	steepness s
`ll`	defines lower prior bound of s
`ul`	defines upper prior bound of s

Value

logit transformed steepness s

`to_logits` *to_logits()*

Description

convert steepness to logit

Usage

```
to_logits(s, ll = 0.2, ul = 1)
```

Arguments

s	steepness s
lim	defines lower prior bound of s
max	defines upper prior bound of s

Value

logit transformed steepness s

Index