# Package: FLRef (via r-universe)

September 7, 2024

**Title** Reference point computation for advice rules

**Version** 1.10.7

**Maintainer** Henning Winker <Henning.Winker@gmail.com>

**Description** Blah

**Depends** R (>= 4.0), ggplot2, FLCore, ggplotFL, FLBRP, FLasher, mse, FLSRTMB

**Imports** methods, ggplot2

**License** EUPL

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.3.1

**Repository** https://ices-tools-prod.r-universe.dev

**RemoteUrl** https://github.com/Henning-Winker/FLRef

**RemoteRef** HEAD

**RemoteSha** 602e0181be1e45d1b6d8e26ce98f235c991a47b4

## Contents


ABItgt . . . 3
ALK . . . 3
alk.sample . . . 4
ALKs . . . 4
applyALK . . . 5
asem2spm . . . 5
bioidx.sim . . . 6
blag . . . 7
ca.sim . . . 7
computeFbrp . . . 8
computeFbrps . . . 9
Fbrp . . . 10
Fe40 . . . 11

flr2stars . . . 11
Fmmy . . . 12
Fp05 . . . 13
Fsim . . . 14
fudc . . . 15
fwd2stars . . . 16
getF . . . 16
huecol . . . 17
iALK . . . 17
idx.sim . . . 18
jabba2FLStockR . . . 19
jabba2stars . . . 20
len.sim . . . 20
lfd.sim . . . 21
Mlorenzen . . . 22
newselex . . . 23
opt.bisect . . . 24
pgquant . . . 25
plotAdvice . . . 26
plotAR . . . 27
plotbioage . . . 28
plotbioyr . . . 29
plotdyn . . . 30
ploteq . . . 30
plotFsim . . . 31
plotMajuro . . . 32
plotpf . . . 34
plotspr . . . 34
plotWKREF . . . 35
rc4 . . . 37
rffwd . . . 37
rGclass . . . 38
schaefer.sim . . . 38
sops . . . 40
spict2FLQuant . . . 40
spict2FLStockR . . . 41
spict2stars . . . 42
ss2FLStockR . . . 42
ss2stars . . . 43
ss3col . . . 43
ssmvln . . . 44
stock2ratios . . . 45
stockMedians . . . 45
updsr . . . 46
updstars . . . 46

**Index** **48**

## ABItgt *ABItgt() Computes ABI for target F, e.g. ABImsy (Griffith et al. 2023)*

### Description

ABItgt() Computes ABI for target F, e.g. ABImsy (Griffith et al. 2023)

### Usage

```
ABItgt(stock, ftgt = 0.2, thresh = 0.9, ...)
```

### Arguments

| | |
|---|---|
| stock | object of class FLStock |
| ftgt | target F at equilibrium, e.g. Fmsy |
| thresh | quantile ageref treshold, default 0.9 |

### Value

*FLQuant*

### Examples

```
data(ple4)
ABImsy = ABItgt(ple4,ftgt=0.22,thresh=0.9)
plot(ABImsy)+ylim(0,2)+
 geom_hline(yintercept = c(0.8,1),col=c(2,1),linetype=c(2,1))+ylab(expression(ABI[MSY]))
```

## ALK *ALK function*

### Description

ALK function

### Usage

```
ALK(N_a, iALK)
```

### Arguments

| | |
|---|---|
| N_a | numbers at age sample for single event |
| iALK | from iALK() outout |

### Value

FLPar of ALK

## alk.sample *generates annual ALK sample with length stratified sampling*

### Description

generates annual ALK sample with length stratified sampling

### Usage

```
alk.sample(lfds, alks, nbin = 20, n.sample = 1)
```

### Arguments

| | |
|---|---|
| lfds | length frequency *FLQuant* |
| alks | annual ALK proportions at age output form ALKs() *FLPars* |
| nbin | number of samples per length bin |
| n.sample | sample size of lfd |

### Value

FLPars of sampled ALK

## ALKs *annual ALK function*

### Description

annual ALK function

### Usage

```
ALKs(object, iALK)
```

### Arguments

| | |
|---|---|
| object | FLQuant with numbers at age |
| iALK | from iALK() outout |

### Value

FLPars of ALK

`applyALK` *applyALK function to length to age*

## Description

applyALK function to length to age

## Usage

```
applyALK(lfds, alks)
```

## Arguments

| | |
|---|---|
| `alks` | *FLPars* annual ALKs |
| `lfd` | *FLQuant* with numbers at length |

## Value

FLQuant for numbers at age

`asem2spm` *asem2spm()*

## Description

asem2spm()

## Usage

```
asem2spm(
  object,
  quant = c("vb", "ssb"),
  fmsy = NULL,
  rel = FALSE,
  spcurve = FALSE
)
```

## Arguments

| | |
|---|---|
| `object` | An *FLBRP* |
| `quant` | choose between vb and ssb |
| `rel` | if TRUE ratios are produced for spcurve |
| `spcurve` | if TRUE a data.frame is added |

## Value

prior means for r and m *FLPar*

## Examples

```
data(ple4)
sr <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
brp = FLBRP(ple4,sr)
asem2spm(brp)[1:4]
plotpf(brp)
plotpf(brp,rel=TRUE)
```

---

# bioidx.sim *generates FLIndexBiomass with random observation error from an FLStock*

---

## Description

generates FLIndexBiomass with random observation error from an FLStock

## Usage

```
bioidx.sim(object, sel = catch.sel(object), sigma = 0.2, q = 0.001)
```

## Arguments

| | |
|---|---|
| object | FLStock |
| sel | FLQuant with selectivity.pattern |
| sigma | observation error for log(index) |
| q | catchability coefficient for scaling |

## Value

FLIndexBiomass

## Examples

```
data(ple4)
sel = newselex(catch.sel(ple4),FLPar(S50=1.5,S95=2.1,Smax=4.5,Dcv=1,Dmin=0.1))
ggplot(sel)+geom_line(aes(age,data))+ylab("Selectivity")+xlab("Age")
object = propagate(ple4,10)
sel = newselex(catch.sel(object),FLPar(S50=2.5,S95=3.2,Smax=3.5,Dcv=0.6,Dmin=0.2))
idx = bioidx.sim(object,sel=sel,q=0.0001)
# Checks
ggplot(idx@sel.pattern)+geom_line(aes(age,data))+ylab("Selectivity")+xlab("Age")
ggplot(idx@index)+geom_line(aes(year,data,col=ac(iter)))+theme(legend.position = "none")+ylab("Index")
```

## blag *blag()*

### Description

function to assign B[y+1] to B[y]. Warning correlation structure of B[y+1] and F[y] is meaningless

### Usage

```
blag(mvn, verbose = TRUE)
```

### Arguments

mvn

### Value

output list of quant posteriors and mle's

### Author(s)

Henning Winker (GFCM)

## ca.sim *generates catch.n with lognormal annual and multinomial age composition observation error*

### Description

generates catch.n with lognormal annual and multinomial age composition observation error

### Usage

```
ca.sim(object, ess = 200, what = c("catch", "landings", "discards")[1])
```

### Arguments

| | |
|---|---|
| object | FLQuant |
| ess | effective sample size for age composition |
| what | c("catch", "landings", "discards") |
| sel | FLQuant with selectivity.pattern e.g. catch.sel() |

### Value

FLQuant with catch.n samples

## Examples

```
data(ple4)
object = propagate(ple4,10)
ca = ca.sim(object,ess=200)
# Checks
ggplot(ca)+geom_line(aes(year,data,col=ac(iter)))+facet_wrap(~age)+
theme(legend.position = "none")+ylab("Index")
```

---

`computeFbrp` *computeFbrp() Computes biological reference points corresponding to the proxy Fbrp*

---

## Description

computeFbrp() Computes biological reference points corresponding to the proxy Fbrp

## Usage

```
computeFbrp(
  stock,
  sr = "missing",
  proxy = NULL,
  x = NULL,
  blim = 0.1,
  type = c("b0", "btgt", "value"),
  btri = "missing",
  bpa = "missing",
  bthresh = "missing",
  verbose = T,
  fmax = 10,
  ...
)
```

## Arguments

| | |
|---|---|
| `stock` | object of class FLStock |
| `sr` | stock recruitment model of class FLSR |
| `proxy` | choice of Fmsy proxies (combinations permitted)<br>• "sprx" spawning potential ratio spr/spr0 with basis x<br>• "bx" SSB as fraction xSSB0<br>• "f0.1" 10<br>• "fe40" Patterns estimator for Fmsy<br>• "msy" maximum surplus production (not defined for segreg)<br>• numeric user value |
| `x` | basis in percent for sprx and bx, e.g. 40 for spr40 |

| | |
|---|---|
| blim | values < 1 are taken as fraction to B0 and blim > 1 as absolute values unless specified otherwise |
| type | type of blim input, values < 1 are<br>• "b0" fraction to B0<br>• "btgt" fraction to Btarget (first occurring in proxy)<br>• "value" absolute value |
| btri | Btrigger can specified as absolute value |
| bpa | Bpa can specified as absolute value |
| bthresh | Bthresh (GFCM) interchangeable use with Bpa |
| verbose | |
| fmax | maximum Flim = max(Flim,fmax*Fbrp) |

## Value

brp object of class FLBRP with computed Fbrp reference points

## Examples

```
data(ple4)
srr = srrTMB(as.FLSR(ple4,model=rickerSV),spr0=spr0y(ple4))
brp = computeFbrp(stock=ple4,sr=srr,proxy=c("sprx","f0.1"),blim=0.1,type="b0")
ploteq(brp,obs=TRUE,refpts="msy")
```

---

# computeFbrps *computeFbrps() Computes biological reference points corresponding to the proxy Fbrp*

---

## Description

computeFbrps() Computes biological reference points corresponding to the proxy Fbrp

## Usage

```
computeFbrps(
  stock,
  sr = "missing",
  proxy = c("sprx", "bx", "all"),
  fmsy = FALSE,
  f0.1 = TRUE,
  fmax = 5,
  verbose = T,
  ...
)
```

### Arguments

| | |
|---|---|
| stock | object of class FLStock |
| sr | stock recruitment model of class FLSR |
| fmsy | if TRUE, Fmsy is computed (not suggest for segreg or geomean sr) |
| f0.1 | if TRUE, F0.1 is computed |
| fmax | maximum Flim = minfmax*Fbrp) |
| verbose | |
| proxies | choice of Fmsy proxies<br>• "all" both sprx and bx<br>• "sprx" spawning potential ratio spr/spr0 with basis x<br>• "bx" SSB as fraction xSSB0 |

### Value

brp object of class FLBRP with computed Fbrp reference points

---

## Fbrp *Fbrp() Extract Fbrp based reference points from output of computeFbrp*

---

### Description

Fbrp() Extract Fbrp based reference points from output of computeFbrp

### Usage

```
Fbrp(brp)
```

### Arguments

| | |
|---|---|
| brp | input of class FLBRP from ComputeFbrp |

### Value

FLPar object with computed Fbrp reference points

`Fe40` *Fe40() Patterson estimator for Fmsy*

## Description

Fe40() Patterson estimator for Fmsy

## Usage

```
Fe40(stock, nyears = 3)
```

## Arguments

| | |
|---|---|
| `stock` | input of class FLStock |
| `nyears` | number of years to average |

## Value

value

`flr2stars` *flr2stars()*

## Description

flr2stars()

## Usage

```
flr2stars(object, quantiles = c(0.05, 0.95))
```

## Arguments

| | |
|---|---|
| `object` | of class FLStockR |
| `quantities` | default is 90CIs as c(0.05,0.95) |

## Value

STARS list with $timeseris and $refpts

`Fmmy` *Fmmy() Uses opt.bisect to derive the F at Maximum Median Yield from stochastic simulations*

## Description

Fmmy() Uses opt.bisect to derive the F at Maximum Median Yield from stochastic simulations

## Usage

```
Fmmy(
  brp,
  sigmaR = 0.5,
  rho = 0,
  nyears = 100,
  iters = 250,
  yrs.eval = NULL,
  range = "missing",
  tol = 0.001,
  maxit = 15,
  verbose = TRUE
)
```

## Arguments

| | |
|---|---|
| `brp` | output object from computeFbrp() of class FLBRP |
| `sigmaR` | lognormal recruitment standard deviation |
| `rho` | AR1 recruitment autocorrelation coefficient |
| `nyears` | number of simulation years |
| `iters` | number simulation iterations |
| `yrs.eval` | last years to be used evaluation period, default nyears/2 |
| `range` | range of Fbar value to be evaluated |
| `tol` | tolerance |
| `maxit` | number of steps |
| `verbose` | cat comments |

## Value

list of FLPar, FLStock and FLBRP objects

## Examples

```
data(ple4)
bh = srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=spr0y(ple4))
brp = computeFbrp(ple4,bh,proxy=c("bx","msy"),x=35,blim=0.1)
fmmy = Fmmy(brp,sigmaR=0.7,rho=0.3)
getF(fmmy) # FMMY value
plotFsim(fmmy)
brpfmmy = computeFbrp(ple4,bh,proxy=getF(fmmy),blim=0.1)
fsim = Fsim(brpfmmy,sigmaR=0.7,rho=0.3)
plotFsim(fsim)
```

---

# Fp05 *Calculates the Fbar value giving a maximum probability of ssb being below Blim of 5 percent*

---

## Description

Calculates the Fbar value giving a maximum probability of ssb being below Blim of 5 percent

## Usage

```
Fp05(
  object,
  iters = "missing",
  range = "missing",
  tol = 0.001,
  maxit = 20,
  verbose = TRUE
)
```

## Arguments

| | |
|---|---|
| object | output from Fsim() |
| iters | Number of iterations, cannot exceed input object |
| range | range of Fbar value to be evaluated |
| verbose | Should progress be shown, TRUE. |

## Value

list

## Examples

```
data(ple4)
bh = srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=spr0y(ple4))
brp = computeFbrp(ple4,bh,proxy="bx",x=35,blim=0.2) # set Blim higher
fsim = Fsim(brp,sigmaR=0.7,rho=0.3,iters=500)
plotFsim(fsim)
fp.05 = Fp05(fsim)
plotFsim(fp.05,panels=c(2,4)) # black line is Fp0.05
getF(fp.05)
```

---

`Fsim` *Fsim() Simulates stochastic stock dynamics under under constant Fbrp*

---

## Description

Fsim() Simulates stochastic stock dynamics under under constant Fbrp

## Usage

```
Fsim(
  brp,
  Ftgt = NULL,
  sigmaR = 0.5,
  rho = 0,
  nyears = 100,
  iters = 250,
  yrs.eval = NULL,
  verbose = TRUE
)
```

## Arguments

| | |
|---|---|
| `brp` | output object from computeFbrp() of class FLBRP |
| `sigmaR` | lognormal recruitment standard deviation |
| `rho` | AR1 recruitment autocorrelation coefficient |
| `nyears` | number of simulation years |
| `iters` | number simulation iterations |
| `yrs.eval` | last years to be used evaluation period, default nyears/2 |
| `verbose` | cat comments |

## Value

list of FLPar, FLStock and FLBRP objects

## Examples

```
data(ple4)
hs = srrTMB(as.FLSR(ple4,model=segreg),spr0=spr0y(ple4),lplim=0.05,uplim=0.25)
blim = params(hs)[[2]]
brp = computeFbrp(ple4,hs,proxy=c("sprx","f0.1","msy"),x=40,blim=blim)
ploteq(brp)
fsim = Fsim(brp,sigmaR=0.7,rho=0.3)
plotFsim(fsim)
plotFsim(fsim,panels=2)
```

---

# fudc *fudc()*

---

## Description

generates an up-down-constant F-pattern

## Usage

```
fudc(
  object,
  fref = 0.2,
  fhi = 2.5,
  flo = 0.8,
  sigmaF = 0.2,
  breaks = c(0.5, 0.75)
)
```

## Arguments

| | |
|---|---|
| object | An *FLStock* |
| fref | reference denominator for fbar |
| fhi | factor for high F as fhi = fbar/fref |
| flo | factor for low F as flo = fbar/fref |
| sigmaF | variation on fbar |
| breaks | relative location of directional change |

## Value

FLQuant

## Examples

```
data(ple4)
sr <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
brp = computeFbrp(ple4,sr,proxy="msy")
fmsy = Fbrp(brp)["Fmsy"]
stki = propagate(ple4,100)
fy = fudc(ple4,fhi=2,flo=0.9,fref=fmsy,sigmaF=0)
fyi = fudc(stki,fhi=2,flo=0.9,fref=fmsy,sigmaF=0.2)
plot(fy,fyi)+ylab("F")
#Forcasting
om <- FLStockR(ffwd(stki,sr,fbar=fyi))
om@refpts = Fbrp(brp)
plotAdvice(window(om,start=1960))
```

---

# fwd2stars *Function to summarise forecast results*

---

## Description

Function to summarise forecast results

## Usage

```
fwd2stars(object, eval.yrs = NULL, rel = TRUE)
```

## Arguments

| | |
|---|---|
| object | *FLStocks* with list of *FLStockR* objects |
| eval.yrs | evaluation years of forecast |
| rel | if TRUE ratios B/Btgt and F/Ftgt are shown |

## Value

data.frame

---

# getF *getF()*

---

## Description

Helper functio to extract F from various FLRef output

## Usage

```
getF(x)
```

### Arguments

| | |
|---|---|
| x | output object from computeFbrp() of class FLBRP |

---

`huecol` *huecol*

---

### Description

huecol

### Usage

```
huecol(n, alpha = 1)
```

### Arguments

| | |
|---|---|
| n | number of colors |
| alpha | transluscency |

---

`iALK` *inverse ALK function with lmin added to FLCore::invALK*

---

### Description

inverse ALK function with lmin added to FLCore::invALK

### Usage

```
iALK(
  params,
  model = vonbert,
  age,
  cv = 0.1,
  lmin = 5,
  lmax = 1.2,
  bin = 1,
  max = ceiling(linf * lmax),
  reflen = NULL
)
```

### Arguments

| | |
|---|---|
| params | growth parameter, default FLPar(linf,k,t0) |
| model | growth model, only option currently vonbert |
| age | age vector |
| cv | of length-at-age |
| lmin | minimum length |
| lmax | maximum upper length specified lmax*linf |
| bin | length bin size, dafault 1 |
| max | maximum size value |
| reflen | evokes fixed sd for L_a at sd = cv*reflen |
| timing | t0 assumed 1st January, default seq(0,11/12,1/12), but can be single event 0.5 |
| unit | default is "cm" |

### Value

FLPar age-length matrix

---

## idx.sim *generates FLIndex with lognormal annual and multinomial age composition observation error*

---

### Description

generates FLIndex with lognormal annual and multinomial age composition observation error

### Usage

```
idx.sim(
  object,
  sel = catch.sel(object),
  ages = NULL,
  years = NULL,
  ess = 200,
  sigma = 0.2,
  q = 0.01
)
```

### Arguments

| | |
|---|---|
| object | FLStock |
| sel | FLQuant with selectivity.pattern |
| ages | define age range |
| years | define year range |
| ess | effective sample size for age composition sample |
| sigma | annual observation error for log(q) |
| q | catchability coefficient for scaling |

## Value

FLIndex

## Examples

```
data(ple4)
sel = newselex(catch.sel(ple4),FLPar(S50=1.5,S95=2.1,Smax=4.5,Dcv=1,Dmin=0.1))
ggplot(sel)+geom_line(aes(age,data))+ylab("Selectivity")+xlab("Age")
object = propagate(ple4,10)
idx = idx.sim(object,sel=sel,ess=200,sigma=0.2,q=0.01,years=1994:2017)
# Checks
ggplot(idx@sel.pattern)+geom_line(aes(age,data))+ylab("Selectivity")+xlab("Age")
ggplot(idx@index)+geom_line(aes(year,data,col=ac(iter)))+facet_wrap(~age,scales="free_y")+
theme(legend.position = "none")+ylab("Index")
```

---

# jabba2FLStockR *jabba2FLStockR()*

---

## Description

jabba2FLStockR()

## Usage

```
jabba2FLStockR(jabba, blim = 0.3, bthr = 0.5, thin = 10, rel = FALSE)
```

## Arguments

| | |
|---|---|
| jabba | fit from JABBA fit_jabba() or jabba$kbtrj |
| blim | biomass limit reference point as fraction of Bmsy |
| thin | thinnig rate of retained iters |
| rel | if TRUE ratios BBmsy and FFmsy are stored |
| bpa | biomass precautionary reference point as fraction of Bmsy |

## Value

FLStockR with refpts

## jabba2stars *jabba2stars()*

### Description

jabba2stars()

### Usage

```
jabba2stars(jabba, quantiles = c(0.05, 0.95), blim = 0.3, bthr = 0.5)
```

### Arguments

| | |
|---|---|
| jabba | fit from JABBA fit_jabba() or jabba$kbtrj |
| quantiles | default is 90CIs as c(0.05,0.95) |
| blim | biomass limit point as fraction of Bmsy, default 0.3Bmsy (ICES) |
| bthr | biomass precautionary point as fraction of Bmsy, default 0.5Bmsy (ICES) |

### Value

STARS list with $timeseris and $refpts

## len.sim *function to generate survey (pulse) and continuous LFDs*

### Description

function to generate survey (pulse) and continuous LFDs

### Usage

```
len.sim(
  N_a,
  params,
  model = vonbert,
  ess = 250,
  timing = seq(0, 11/12, 1/12),
  unit = "cm",
  scale = TRUE,
  reflen = NULL,
  bin = 1,
  cv = 0.1,
  lmin = 5,
  lmax = 1.2
)
```

### Arguments

| | |
|---|---|
| N_a | numbers at age sample |
| params | growth parameter, default FLPar(linf,k,t0) |
| model | growth model, only option currently vonbert |
| ess | effective sample size |
| timing | t0 assumed 1st January, default seq(0,11/12,1/12), but can be single event 0.5 |
| unit | default is "cm" |
| scale | if TRUE scaled to N_a input |
| reflen | evokes fixed sd for L_a at sd = cv*reflen |
| bin | length bin size, dafault 1 |
| cv | variation in L_a |
| lmin | minimum length |
| lmax | maximum upper length specified lmax*linf |

### Value

FLQuant for length

---

## lfd.sim *function to generate survey (pulse) and continuous LFDs*

---

### Description

function to generate survey (pulse) and continuous LFDs

### Usage

```
lfd.sim(
  object,
  stock,
  sel = catch.sel(stock),
  params,
  model = vonbert,
  ess = 250,
  timing = seq(0, 11/12, 1/12),
  timeref = 0.5,
  unit = "cm",
  scale = TRUE,
  reflen = NULL,
  bin = 1,
  cv = 0.1,
  lmin = 5,
  lmax = 1.2
)
```

### Arguments

| | |
|---|---|
| object | *FLQuant* numbers at age sample |
| stock | *FLStock* object |
| sel | selectivity, default catch.sel(stock) |
| params | growth parameter, default FLPar(linf,k,t0) |
| model | growth model, only option currently vonbert |
| ess | effective sample size |
| timing | default constinoues seq(0,11/12,1/12), but can be single event 0.5 |
| timeref | reference timing of the sample, default 0.5 (e.g. survey or catch.n) |
| unit | default is "cm" |
| scale | if TRUE scaled to N_a input |
| reflen | evokes fixed sd for L_a at sd = cv*reflen |
| bin | length bin size, dafault 1 |
| cv | variation in L_a |
| lmin | minimum length |
| lmax | maximum upper length specified lmax*linf |

### Value

FLQuant for length

---

## Mlorenzen *Mlorenzen*

---

### Description

computes Lorenzen M with scaling option

### Usage

```
Mlorenzen(object, Mref = "missing", Aref = 2)
```

### Arguments

| | |
|---|---|
| object | weight-at-age of class *FLQuant* |
| Mref | reference M for scaling |
| Aref | reference Age for scaling |

### Value

FLQuant m()

## Examples

```
data(ple4)
Ml = Mlorenzen(stock.wt(ple4))
# Scale
Ms = Mlorenzen(stock.wt(ple4),Mref=0.2,Aref=2)
flqs = FLQuants(Lorenzen=Ml,Scaled=Ms)
```

---

# newselex *generates flexible 5-paramater selex curves*

---

## Description

generates flexible 5-paramater selex curves

## Usage

```
newselex(object, selexpars)
```

## Arguments

| | |
|---|---|
| object | FLQuant from catch.sel() or sel.pattern() |
| selexpars | Selectivity Parameters selexpars S50, S95, Smax, Dcv, Dmin |

- S50: age at 50
- S95: age at 50
- Smax: age at peak of selectivity before descending limb
- Dcv: CV demeterming the steepness of the descending half-normal slope
- Dmin: determines the minimum retention of oldest fishes

## Value

FLquant with selectivity pattern

## Examples

```
data(ple4)
sel = newselex(catch.sel(ple4),FLPar(S50=2,S95=3,Smax=4.5,Dcv=0.6,Dmin=0.3))
ggplot(sel)+geom_line(aes(age,data))+ylab("Selectivity")+xlab("Age")
# Simulate
harvest(ple4)[] = sel
sr <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
brp = computeFbrp(ple4,sr,proxy="msy")
fbar(brp) = FLQuant(rep(0.01,70))
stk = as(brp,"FLStock")
units(stk) = standardUnits(stk)
its = 100
stk <- FLStockR(propagate(stk, its))
stk@refpts= Fbrp(brp)
```

```
b0=an(Fbrp(brp)["B0"])
control = FLPar(Feq=0.15,Frate=0.1,Fsigma=0.15,SB0=b0,minyear=2,maxyear=70,its=its)
run <- rffwd(stk, sr=sr,control=control,deviances=ar1rlnorm(0.3, 1:70, its, 0, 0.6))
plotAdvice(run)
```

---

`opt.bisect` *Bisection approach to optimise x for maximising y*

---

## Description

The plain bisection algorithm (Burden & Douglas, 1985) is employed here to find the value of a given forecast target quantity (e.g. 'fbar') for which a selected value of a performance statistic is obtained over a chosen period.

## Usage

```
opt.bisect(
  stock,
  sr,
  deviances = rec(stock) %=% 1,
  metrics,
  statistic,
  years,
  pyears = years,
  tune,
  tol = 0.001,
  maxit = 15,
  log = TRUE,
  verbose = TRUE
)
```

## Arguments

| | |
|---|---|
| `stock` | object class FLStock |
| `sr` | object class FLSR |
| `metrics` | FLQuant of FLStock to be defined |
| `statistic` | |
| `years` | years to be evaluated |
| `tune` | range for input x |
| `tol` | tolerance level |
| `maxit` | number of optimisation steps |
| `log` | if TRUE, optimise on log-scale |

## Author(s)

Credits to Iago Mosqueira

## References

Burden, Richard L.; Faires, J. Douglas (1985), "2.1 The Bisection Algorithm", Numerical Analysis (3rd ed.), PWS Publishers, ISBN 0-87150-857-5

## Examples

```
data(ple4)
stock <- propagate(stf(ple4, end=2118), 200)
srr <- predictModel(model=rec ~ ifelse(ssb <= b, a * ssb, a * b), params=FLPar(a=1.29, b=1.35e+06))
# GENERATE SRR deviances
devs <- ar1rlnorm(rho=0.4, 2018:2118, iters=200, meanlog=0, sdlog=0.5)
# DEFINE MMY statistic
statistic <- list(MMY=list(~apply(L,1,median), name="MMY",
  desc="ICES Maximum Median Yield"))
# CALL bisect over 100 years, Fmmy calculated over last 50.
fmmy <- opt.bisect(stock, sr=srr, deviances=devs, metrics=list(L=landings),
statistic=statistic, years=2018:2118,
pyears=2069:2118, tune=list(fbar=c(0.01, 0.2)))
# fmmy
mean(fbar(fmmy)[,ac(2069:2118)])
```

---

# pgquant *sets plus group on FLQuant*

---

## Description

sets plus group on FLQuant

## Usage

```
pgquant(object, pg)
```

## Arguments

| | |
|---|---|
| object | FLQuant |
| pg | |

## Value

FLQuant

plotAdvice *plotAdvice Plots stochastic stock dynamics against refpts for constant Fsim()*

## Description

plotAdvice Plots stochastic stock dynamics against refpts for constant Fsim()

## Usage

```
plotAdvice(
  object,
  rpts = "missing",
  type = NULL,
  plotrefs = TRUE,
  probs = c(0.05, 0.2, 0.5, 0.8, 0.95),
  colour = "dodgerblue",
  ncol = NULL,
  label.size = 2.5
)
```

## Arguments

| | |
|---|---|
| type | age-structured "asm" or surplus production "spm" plotting style |
| plotrefs | if TRUE reference points are plotted |
| probs | determine credibility intervals, default 80th, 90th percentiles #' @param ncol number of plot panel columns |
| colour | color of CIs |
| label.size | size of refpts labels |
| stock | FLStock or FLStockR |
| refpts | as FLPar or Fbrp() if FLStockR is not provided or should be overwritten |

## Value

ggplot

## Examples

```
data(ple4)
srr = srrTMB(as.FLSR(ple4,model=rickerSV),spr0=spr0y(ple4))
brp = computeFbrp(stock=ple4,sr=srr,proxy=c("sprx","f0.1","fe40"),blim=0.1,type="b0")
plotAdvice (ple4,brp)
```

plotAR *plotAR Plots the new proposed ICES advice rule*

## Description

plotAR Plots the new proposed ICES advice rule

## Usage

```
plotAR(
  pars,
  ftgt = 1,
  btrigger = "missing",
  bpa = "missing",
  bthresh = "missing",
  fpa = "missing",
  fthresh = "missing",
  bclose = 0,
  fmin = 0,
  obs = "missing",
  kobe = TRUE,
  alpha = 1,
  xmax = 1.2,
  ymax = 1.5,
  ylab = "missing",
  xlab = "missing",
  rel = FALSE,
  expand = TRUE,
  labels = TRUE,
  label.cex = 3.5,
  critical = TRUE
)
```

## Arguments

| | |
|---|---|
| pars | FLPar object or computeFbrp() ouput<br>• 1: "Fbrp" # "F.." must first<br>• 2: "Btgt"<br>• 3: "Blim"<br>• 4: "B0" |
| ftgt | factor to adjust Fmsy or its proxy e.g. 0.8Fmsy |
| btrigger | biomass trigger below which F is linearly reduced, if > 10 value, else factor*Btgt |
| bpa | precautionary biomass threshold, if > 10 value, else factor*Blim |
| fpa | option to input Fpa value |
| bclose | biomass that invokes fishing closure |

| | |
|---|---|
| fmin | minimum allowable (bycatch) fishing mortality under closure |
| obs | obtion to show observation with input class 'FLStock' |
| kobe | add kobe colour-coding |
| alpha | transparency of shading |
| xmax | multiplier for upper default xlim |
| ymax | multiplier for upper default ylim |
| ylab | option customize ylab |
| xlab | option customize xlab |
| rel | option to denote x,y labs as relative B/Btgt and F/Ftgt |
| expand | option to expand the plot area to border - default TRUE |
| labels | annotate reference point labels |
| critical | option to highlight critical zone below blim |
| labelslabel.cex=3.5 | set size of labels |

## Value

ggplot

## Examples

```
data(ple4)
srr = srrTMB(as.FLSR(ple4,model=segreg),spr0=spr0y(ple4))
blim = params(srr)[[2]]
brp = computeFbrp(stock=ple4,sr=srr,proxy="f0.1",blim=blim)
rpt = Fbrp(brp)
plotAR(rpt,btrigger=an(0.8*rpt["Btgt"]))
# Use Bpa as trigger (ICES style)
plotAR(rpt,obs=ple4,bpa=1.4)
# Change kobe to greyscale
plotAR(rpt,obs=ple4,bpa=1.4,kobe=FALSE)
# add fishing closure with minimum unavoidable F and Btrigger
plotAR(rpt,obs=ple4,bpa=1.4,btrigger=0.7,kobe=TRUE,bclose=1,fmin=0.01)
# show a relative
plotAR(rpt,obs=ple4,rel=TRUE,bpa=1.4,btrigger=0.7,kobe=TRUE,bclose=1,fmin=0.02)
```

---

# plotbioage *plotbioage() Plots stock N_a, W_a, M_a and Mat_a by year*

---

## Description

plotbioage() Plots stock N_a, W_a, M_a and Mat_a by year

## Usage

```
plotbioage(stk, ncol = 2)
```

### Arguments

| | |
|---|---|
| stk | stock object class FLStock |
| ncol | number of columns in multiplot |

### Value

ggplot

### Examples

```
data(ple4)
plotbioage(ple4)
```

---

## plotbioyr *plotbioyr() Plots stock N_a, W_a, M_a and Mat_a across years*

---

### Description

plotbioyr() Plots stock N_a, W_a, M_a and Mat_a across years

### Usage

```
plotbioyr(stk, ncol = 2)
```

### Arguments

| | |
|---|---|
| stk | stock object class FLStock |
| ncol | number of columns in multiplot |

### Value

ggplot

### Examples

```
data(ple4)
plotbioyr(ple4)
```

plotdyn *plotdyn() Plots stock trajectories at age*

## Description

plotdyn() Plots stock trajectories at age

## Usage

```
plotdyn(stk, ncol = 2)
```

## Arguments

| | |
|---|---|
| stk | stock object class FLStock |
| ncol | number of columns in multiplot |

## Value

ggplot

## Examples

```
data(ple4)
plotdyn(ple4)
```

ploteq *ploteq() Modification of method plot('FLBRP') to plot equilibrium output of computeFbrp()*

## Description

ploteq() Modification of method plot('FLBRP') to plot equilibrium output of computeFbrp()

## Usage

```
ploteq(
  brps,
  refpts = "missing",
  obs = FALSE,
  rel = FALSE,
  rpf = TRUE,
  dashed = rpf,
  colours = "missing",
  panels = NULL,
  ncol = 2
)
```

### Arguments

| | |
|---|---|
| brps | output object from computeFbrp of class FLBRP |
| refpts | Reference points, defaults are computed refpts from computeFbrp()<br>• Fbrp<br>• Blim<br>• B0<br>• Btri |
| obs | Should observations be plotted? Defaults to 'FALSE'. |
| rel | option to denote x,y labs as relative B/Btgt and F/Ftgt |
| rpf | adds refpts in plots |
| dashed | plots vertical dashed lines to highlight refpts locations |
| colours | refpts colours, default is designed for computeFbrp() output |
| panels | plot panel option 1:4 |
| ncol | number of plot panel columns |

### Value

ggplot

### Examples

```
data(ple4)
srr = srrTMB(as.FLSR(ple4,model=rickerSV),spr0=spr0y(ple4))
brp = computeFbrp(stock=ple4,sr=srr,proxy=c("sprx","f0.1","msy"),blim=0.1,type="b0")
ploteq(brp,obs=TRUE)
ploteq(brp,obs=TRUE,refpts="msy",rel=TRUE)
brp.pa = computeFbrp(stock=ple4,sr=srr,proxy=c("msy","sprx","f0.1"),blim=0.1,bpa=Fbrp(brp)["Blim"]*2,type="b0")
ploteq(brp.pa,obs=TRUE,rel=TRUE)
```

---

## plotFsim *plotFsim Plots stochastic stock dynamics against refpts for constant Fsim()*

---

### Description

plotFsim Plots stochastic stock dynamics against refpts for constant Fsim()

### Usage

```
plotFsim(
  object,
  worms = TRUE,
  thinning = 10,
  probs = c(0.05, 0.2, 0.5, 0.8, 0.95),
```

```
  plotrefs = TRUE,
  colour = "missing",
  ncol = "missing",
  label.size = 3,
  yrs.eval = NULL,
  panels = "missing"
)
```

## Arguments

| | |
|---|---|
| object | output object from Fsim() |
| worms | option to show individual iterations |
| thinning | thinning rate of iterations shows, e.g. 10 shows every 10th |
| probs | determine credibility intervals, default 80th, 90th percentiles |
| plotrefs | if TRUE reference points are plotted |
| colour | color of CIs |
| ncol | number of plot panel columns |
| label.size | size of reference points |
| yrs.eval | last years to be used evaluation period, default half nyears |

## Value

ggplot

---

# plotMajuro *Plots the new proposed ICES advice rule*

---

## Description

Plots the new proposed ICES advice rule

## Usage

```
plotMajuro(
  ftgt = 1,
  fthresh = 1.1,
  btgt = 1,
  blim = 0.1,
  btrigger = 0.8 * btgt,
  bthresh = 0.5 * btgt,
  bclose = 0,
  fmin = 0,
  obs = "missing",
  kobe = TRUE,
  alpha = 1,
```

```
  xmax = 1.5,
  ymax = 1.5,
  ylab = "missing",
  xlab = "missing",
  rel = FALSE,
  expand = TRUE,
  labels = TRUE,
  critical = kobe
)
```

## Arguments

| | |
|---|---|
| ftgt | Target F = min(Fbrp,Fp0.5) |
| btgt | Biomass target corresponding to Fbrp |
| blim | biomass limit |
| btrigger | biomass trigger below which F is linearly reduced |
| bthresh | biomass threshold beyond which biomass is classified sustainable |
| bclose | biomass that invokes fishing closure |
| fmin | minimum allowable (bycatch) fishing mortality under closure |
| obs | obtion to show observation with input class 'FLStock' |
| kobe | add kobe colour-coding |
| alpha | transparency of shading |
| xmax | multiplier for upper default xlim |
| ymax | multiplier for upper default ylim |
| ylab | option customize ylab |
| xlab | option customize xlab |
| rel | option to denote x,y labs as relative B/Btgt and F/Ftgt |
| expand | option to expand the plot area to border - default TRUE |
| labels | annotate reference point labels |
| critical | option to highlight critical zone below blim |

## Value

ggplot

## Examples

```
plotMajuro()
```

## plotpf *plotpf()*

### Description

plots production functions

### Usage

```
plotpf(object, quant = c("vb", "ssb"), fmsy = NULL, rel = FALSE)
```

### Arguments

| | |
|---|---|
| object | An *FLBRP* |
| quant | choose between vb and ssb or both |
| fmsy | default if Fmsy |
| rel | if TRUE ratios are produced for spcurve |

### Value

ggplot

### Examples

```
data(ple4)
sr <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
brp = FLBRP(ple4,sr)
asem2spm(brp)[1:4]
plotpf(brp)
plotpf(brp,rel=TRUE)
```

## plotspr *plotspr() Plots current vs unfished spawning biomass per recruit at age*

### Description

plotspr() Plots current vs unfished spawning biomass per recruit at age

### Usage

```
plotspr(stk, nyears = 3)
```

### Arguments

| | |
|---|---|
| stk | stock object class FLStock |
| nyears | number of current last years, default is 3 |
| ncol | number of columns in multiplot |

### Value

ggplot

### Examples

```
data(ple4)
plotbioage(ple4)
```

---

## plotWKREF *plotWKREF Plots the new proposed ICES advice rule*

---

### Description

plotWKREF Plots the new proposed ICES advice rule

### Usage

```
plotWKREF(
  ftgt = 1,
  btgt = 1,
  blim = 0.2,
  btrigger = 0.9 * btgt,
  bthresh = 0.8 * btgt,
  bclose = 0,
  fmin = 0,
  obs = "missing",
  kobe = TRUE,
  alpha = 1,
  xmax = 1.3,
  ymax = 1.5,
  ylab = "missing",
  xlab = "missing",
  rel = FALSE,
  expand = TRUE,
  labels = TRUE,
  critical = kobe
)
```

## Arguments

| | |
|---|---|
| ftgt | Target F = min(Fbrp,Fp0.5) |
| btgt | Biomass target corresponding to Fbrp |
| blim | biomass limit |
| btrigger | biomass trigger below which F is linearly reduced |
| bthresh | biomass threshold beyond which biomass is classified sustainable |
| bclose | ratio biomass/blim that invokes fishing closure relative to blim |
| fmin | minimum allowable (bycatch) fishing mortality under closure |
| obs | obtion to show observation with input class 'FLStock' |
| kobe | add kobe colour-coding |
| alpha | transparency of shading |
| xmax | multiplier for upper default xlim |
| ymax | multiplier for upper default ylim |
| ylab | option customize ylab |
| xlab | option customize xlab |
| rel | option to denote x,y labs as relative B/Btgt and F/Ftgt |
| expand | option to expand the plot area to border - default TRUE |
| labels | annotate reference point labels |
| critical | option to highlight critical zone below blim |

## Value

ggplot

## Examples

```
plotWKREF()
# Close fishery at Blim and adjust axis labels to relative
plotWKREF(blim=0.2,bclose=0.2,rel=TRUE)
# Close fishery at Blim, but allow fmin (e.g. bycatch)
plotWKREF(blim=0.2,bclose=0.2,fmin=0.1,rel=TRUE)
# Change Btrigger above Btgt
plotWKREF(blim=0.2,bclose=0.2,fmin=0.1,btrigger=0.80,rel=TRUE)
# Plot stock data
data(ple4)
plotWKREF(ftgt=0.25,btgt=8e+05,btrigger = 0.9*8e+05, blim=2e5,bclose=3e5,fmin=0.03,obs=ple4)
```

## rc4 *r4sscol*

### Description

r4sscol

### Usage

```
rc4(n, alpha = 1)
```

### Arguments

| | |
|---|---|
| n | number of colors |
| alpha | transluscency |

### Value

vector of color codes

## rffwd *rffwd() Project forward an FLStock with evolutionary Fbar*

### Description

rffwd() Project forward an FLStock with evolutionary Fbar

### Usage

```
rffwd(object, sr, fbar = control, control = fbar, deviances = "missing")
```

### Arguments

| | |
|---|---|
| object | An *FLStock* |
| sr | A stock-recruit relationship, *FLSR* or *predictModel*. |
| fbar | Yearly target for average fishing mortality, *FLQuant*. |
| control | Yearly target for average fishing mortality, *FLPar*. |
| deviances | Deviances for the strock-recruit relationsip, *FLQuant*. |

### Value

The projected *FLStock* object.

## Examples

```
data(ple4)
sr <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
brp = computeFbrp(ple4,sr,proxy="msy")
fbar(brp) = FLQuant(rep(0.01,70))
stk = as(brp,"FLStock")
units(stk) = standardUnits(stk)
its = 100
stk <- FLStockR(propagate(stk, its))
stk@refpts= Fbrp(brp)
b0=an(Fbrp(brp)["B0"])
control = FLPar(Feq=0.15,Frate=0.1,Fsigma=0.15,SB0=b0,minyear=2,maxyear=70,its=its)
run <- rffwd(stk, sr=sr,control=control,deviances=ar1rlnorm(0.3, 1:70, its, 0, 0.6))
plotAdvice(run)
```

---

# rGclass *Function to characterize Productivity and refpts based on r and Generation*

---

## Description

Function to characterize Productivity and refpts based on r and Generation

## Usage

```
rGclass(r = NULL, gt = NULL)
```

## Arguments

| | |
|---|---|
| r | value of the intrinsic rate of population increase |
| gt | generation time G |

## Value

list with Productivity category and suggest Fbrps

---

# schaefer.sim *schaefer.sim()*

---

## Description

generates a Schafer surplus production model with process and observation error

## Usage

```
schaefer.sim(
  k = 10000,
  r = 0.3,
  q = 0.5,
  pe = 0.1,
  oe = 0.2,
  bk = 0.9,
  years = 1980:2022,
  f0 = 0.2,
  fhi = 2.2,
  flo = 0.8,
  sigmaF = 0.15,
  iters = 1,
  blim = 0.3,
  bthr = 0.5,
  rel = FALSE
)
```

## Arguments

| | |
|---|---|
| k | carrying capacity |
| r | intrinsic rate of population increase |
| q | catchability coefficient |
| pe | process error |
| oe | process error |
| bk | initial fraction of b/k |
| years | time horizon |
| f0 | factor for initial year as f0 = f/fmsy |
| fhi | factor for high F as fhi = f/fmsy |
| flo | factor for low F as flo = fbar/fmsy |
| sigmaF | variation on f trajectory |
| iters | number of iterations |
| rel | if TRUE metrics B/Bmsy and F/Fmsy are produced |

## Value

FLQuants

## Examples

```
stk = schaefer.sim(iters=100,q=0.5)
plotAdvice(stk)
plot(FLIndex(index=iter(stk@stock,1))) # index
```

## sops *scales catch-at-age to total catch with error (optional)*

### Description

scales catch-at-age to total catch with error (optional)

### Usage

```
sops(object, stock, sigma = 0.1, what = c("catch", "landings", "discards")[1])
```

### Arguments

| | |
|---|---|
| object | FLQuant catch.n, discard.n, landings.n |
| stock | FLStock |
| sigma | observation error |
| what | type c("catch", "landings", "discards") |

### Value

FLQuant

## spict2FLQuant *spict2FLQuant()*

### Description

spict2FLQuant()

### Usage

```
spict2FLQuant(
  x,
  metric = c("ssb", "fbar", "catch", "stock", "harvest")[1],
  osa = FALSE,
  forecast = F
)
```

### Arguments

| | |
|---|---|
| x | fit from SPICT |
| osa | add one-step-ahead forecast |
| forecast | TRUE/FALSE |

### Value

FLQuant

### Author(s)

adopted from Laurie Kell (biodyn)

---

`spict2FLStockR` *spict2FLStockR()*

---

### Description

spict2FLStockR()

### Usage

```
spict2FLStockR(
  res,
  blim = 0.3,
  bthr = 0.5,
  rel = FALSE,
  osa = FALSE,
  forecast = NULL
)
```

### Arguments

| | |
|---|---|
| `res` | fit from SPICT |
| `blim` | biomass limit reference point as fraction of Bmsy |
| `bthr` | biomass precautionary reference point as fraction of Bmsy |
| `rel` | if TRUE ratios BBmsy and FFmsy are stored |
| `osa` | add one-step-ahead forecast |
| `forecast` | extract forecast TRUE/FALSE |

### Value

FLStockR with refpts

## spict2stars *spict2stars()*

### Description

spict2stars()

### Usage

```
spict2stars(spict, blim = 0.3, bthr = 0.5)
```

### Arguments

| | |
|---|---|
| spict | fit from fit.spict() |
| blim | biomass limit point as fraction of Bmsy, default 0.3Bmsy (ICES) |
| bthr | biomass precautionary point as fraction of Bmsy, default 0.5Bmsy (ICES) |

### Value

STARS list with $timeseris and $refpts

## ss2FLStockR *ss2FLStockR()*

### Description

ss2FLStockR()

### Usage

```
ss2FLStockR(mvln, thin = 10, output = NULL)
```

### Arguments

| | |
|---|---|
| mvln | output from ssmvln() |
| thin | thinnig rate of retained iters |
| output | expected outputs presented as "mle" or median of "iters" |

### Value

FLStockR with refpts

## ss2stars *ss2stars()*

### Description

ss2stars()

### Usage

```
ss2stars(mvln, output = c("iters", "mle")[1], quantiles = c(0.025, 0.975))
```

### Arguments

| | |
|---|---|
| mvln | output of ssmvln() |
| output | choice c("iters","mle")[1] |
| quantiles | default is 95CIs as c(0.025,0.975) |

### Value

STARS list with $timeseris and $refpts

## ss3col *ss3col*

### Description

ss3col

### Usage

```
ss3col(n, alpha = 1)
```

### Arguments

| | |
|---|---|
| n | number of colors |
| alpha | transluscency |

### Value

vector of color codes

ssmvln *ssmvln()*

## Description

function to generate uncertainty for Stock Synthesis

## Usage

```
ssmvln(
  ss3rep,
  Fref = NULL,
  years = NULL,
  virgin = FALSE,
  mc = 1000,
  weight = 1,
  run = "MVLN",
  addprj = FALSE,
  ymax = NULL,
  xmax = NULL,
  legendcex = 1,
  verbose = TRUE,
  seed = 123
)
```

## Arguments

| | |
|---|---|
| ss3rep | from r4ss::SS_output |
| Fref | Choice of Fratio c("MSY","Btgt","SPR","F01"), correponding to F_MSY and F_Btgt |
| years | single year or vector of years for mvln |
| virgin | if FALSE (default) the B0 base for Bratio is SSB_unfished |
| mc | number of monte-carlo simulations |
| weight | weighting option for model ensembles weight*mc |
| run | qualifier for model run |
| addprj | include forecast years |
| ymax | ylim maximum |
| xmax | xlim maximum |
| verbose | Report progress to R GUI? |
| seed | retains interannual correlation structure like MCMC |
| out | choice c("iters","mle") |
| plot | option to show plot |
| legendcex=1 | Allows to adjust legend cex |

### Value

output list of quant posteriors and mle's

### Author(s)

Henning Winker (GFCM)

---

`stock2ratios` *stock2ratios()*

---

### Description

stock2ratios()

### Usage

```
stock2ratios(object)
```

### Arguments

| | |
|---|---|
| `object` | of class *FLStockR* |

### Value

FLStockR with ratios F/Ftgt and B/Btgt

---

`stockMedians` *stockMedians*

---

### Description

stockMedians

### Usage

```
stockMedians(stock)
```

### Arguments

| | |
|---|---|
| `stock` | class FLStock or FLStockR |

### Value

medians of all FLstock FlQuants

## updsr *updsr()*

### Description

updates sr in brp after changing biology

### Usage

```
updsr(object, s = 0.7, v = NULL)
```

### Arguments

| | |
|---|---|
| object | An *FLBRP* |
| s | assumed steepness s |
| v | input option new SB0 |

### Value

FLBRP

### Examples

```
data(ple4)
sr <- srrTMB(as.FLSR(ple4,model=bevholtSV),spr0=mean(spr0y(ple4)))
brp = FLBRP(ple4,sr)
s = sr@SV[[1]]
params(brp)
# change
m(brp) = Mlorenzen(stock.wt(brp),Mref=0.15)
brpupd =updsr(brp,s)
params(brp)
```

## updstars *updstars()*

### Description

updstars()

### Usage

```
updstars(star, newrefpts)
```

## Arguments

| | |
|---|---|
| star | output of star list |
| newrefpts | manually adjusted reference points |

## Value

STARS list with $timeseris and $refpts

# Index


ABItgt, 3
ALK, 3
alk.sample, 4
ALKs, 4
applyALK, 5
asem2spm, 5

bioidx.sim, 6
blag, 7

ca.sim, 7
computeFbrp, 8
computeFbrps, 9

Fbrp, 10
Fe40, 11
flr2stars, 11
Fmmy, 12
Fp05, 13
Fsim, 14
fudc, 15
fwd2stars, 16

getF, 16

huecol, 17

iALK, 17
idx.sim, 18

jabba2FLStockR, 19
jabba2stars, 20

len.sim, 20
lfd.sim, 21

Mlorenzen, 22

newselex, 23

opt.bisect, 24

pgquant, 25
plotAdvice, 26
plotAR, 27
plotbioage, 28
plotbioyr, 29
plotdyn, 30
ploteq, 30
plotFsim, 31
plotMajuro, 32
plotpf, 34
plotspr, 34
plotWKREF, 35

rc4, 37
rffwd, 37
rGclass, 38

schaefer.sim, 38
sops, 40
spict2FLQuant, 40
spict2FLStockR, 41
spict2stars, 42
ss2FLStockR, 42
ss2stars, 43
ss3col, 43
ssmvln, 44
stock2ratios, 45
stockMedians, 45

updsr, 46
updstars, 46