An_introduction_to_MSE_with_FLR

16 February, 2017

This tutorial introduces a basic MSE: conditioning the operating model (starting from an existing stock assessment), setting up the observation error model, constructing a simple model-based HCR (based on the ICES MSY approach), performing the MSE simulations, producing performance statistics

Required packages

To follow this tutorial you should have installed the following packages:

- CRAN: ggplot2
- FLR: FLCore, FLash,FLXSA,FLBRP, ggplotFL

You can do so as follows,

```
install.packages(c("ggplot2"))
install.packages(c("FLa4a","FLash","FLXSA","FLBRP","ggplotFL"), repos="http://flr-project.org/R")
```

```
# This chunk loads all necessary packages, trims pkg messages
library(FLa4a)
library(FLash)
library(FLXSA)
library(FLBRP)
library(ggplotFL)
```

CONDITIONING THE OPERATING MODEL

Read in stock assessment data

```
data(ple4)
data(ple4.index)
stk <- ple4; rm("ple4")
idx <- FLIndices(idx=ple4.index); rm("ple4.index")
```

Set up the iteration and projection window parameters

```
it <- 20 # iterations
y0 <- range(stk)["minyear"] # initial data year
dy <- range(stk)["maxyear"] # final data year
iy <- dy+1  # initial year of projection (also intermediate)
fy <- dy+12 # final year
ny <- fy - iy + 1 # number of years to project from intial year
nsqy <- 3 # number of years to compute status quo metrics
```

Fit stock assessment model a4a

```
qmod <- list(~s(age, k=6))
fmod <- ~ te(replace(age, age>9,9), year, k=c(6,8))
mcmc <- 2000
mcsave <- mcmc / it  #this needs to be an integer value
fit <- a4aSCA(stk, idx, fmodel=fmod, qmodel=qmod, fit="MCMC", mcmc = SCAMCMC(mcmc = mcmc, mcsave = mcsa
```

```
stk <- stk + fit
stk0 <- qapply(stk, iterMedians) #reduce to keep one iteration only
```

Fit stock-recruit model

```
srbh <- fmle(as.FLSR(stk, model="bevholt"), method="L-BFGS-B", lower=c(1e-6, 1e-6), upper=c(max(rec(stk
```

```
## final  value -13.199792
## converged
## final  value -13.199792
## converged
## final  value -13.161960
## converged
## final  value -13.117131
## converged
## final  value -12.519496
## converged
## final  value -12.925271
## converged
## final  value -13.017653
## converged
## final  value -14.434271
## converged
## final  value -15.942283
## converged
## final  value -13.203449
## converged
## final  value -12.670650
## converged
## final  value -11.813228
## converged
## final  value -11.395789
## converged
## final  value -13.100922
## converged
## final  value -12.684184
## converged
## final  value -11.656310
## converged
## final  value -11.042795
## converged
## final  value -11.929659
## converged
## final  value -11.272703
## converged
## final  value -11.107311
## converged
```

```
srbh0 <- fmle(as.FLSR(stk0, model="bevholt"), method="L-BFGS-B", lower=c(1e-6, 1e-6), upper=c(max(rec(st
```

```
## final  value -13.625097
## converged
```

```
srbh.res <- rnorm(it, FLQuant(0, dimnames=list(year=iy:fy)), c(apply(residuals(srbh), 6, mad)))
```

Calculate reference points and set up the operating model for the projection window

```
brp <- brp(FLBRP(stk0, srbh0))
Fmsy <- c(refpts(brp)["msy","harvest"])
Bpa <- 0.5*c(refpts(brp)["msy","ssb"])
stk <- stf(stk, fy-dy, nsqy, nsqy)
```

SET UP OBSERVATION ERROR MODEL ELEMENTS

Estimate the indices catchability from the a4a fit (without simulation)

```
idcs <- FLIndices()
for (i in 1:length(idx)){
    lst <- mcf(list(idx[[i]]@index, stock.n(stk0)))
    idx.lq <- log(lst[[1]]/lst[[2]]) # log catchability of index
    idx.qmu <- idx.qsig <- stock.n(iter(stk,1)) # empty quant
    idx.qmu[] <- yearMeans(idx.lq) # Every year has the same mean catchability
    idx.qsig[] <- log((sqrt(yearVars(idx.lq))/yearMeans(idx.lq))^2 + 1) # Every year has same sd
    idx.q <- FLQuant(NA, dimnames=dimnames(stock.n(stk)))
    idx.q[,ac(dimnames(stock.n(stk))$year[1]:dy)] <- propagate(exp(idx.lq[,ac(dimnames(stock.n(stk))$yea
    idx.q <- rlnorm(it, idx.qmu, idx.qsig) # Build FLQ of index catchability based on lognormal distrib
    idx_temp <- idx.q * stock.n(stk)
    idx_temp <- FLIndex(index=idx_temp, index.q=idx.q) # generate initial index
    range(idx_temp)[c("startf", "endf")] <- c(0, 0)
    idcs[[i]] <- idx_temp
}
names(idcs) <- names(idx)
idx<-idcs[1]
```

SET UP MSE LOOP

Needed Functions

Observation error model

```
o <- function(stk, idx, assessmentYear, dataYears) {
    # dataYears is a position vector, not the years themselves
    stk0 <- stk[, dataYears]
    # add small amount to avoid zeros
    catch.n(stk0) <- catch.n(stk0) + 0.1
    # Generate the indices - Just data years
    idx0 <- lapply(idx, function(x) x[,dataYears])
  # Generate objserved index
    for (i in 1:length(idx)) index(idx[[i]])[, assessmentYear] <- stock.n(stk)[, assessmentYear]*index.c
  list(stk=stk0, idx=idx0, idx.om=idx)
}
```

XSA assessment model

```
xsa <- function(stk0, idx0){
  # Use default XSA settings
    control  <- FLXSA.control(tol = 1e-09, maxit=99, min.nse=0.3, fse=2.0,
                         rage = -1, qage = stk0@range["max"]-1, shk.n = TRUE, shk.f = TRUE,
```

```
                                shk.yrs = 5, shk.ages= 5, window = 100, tsrange = 99, tspower = 0)
  # Fit XSA
  fit0 <- FLXSA(stk0, idx0, control)
  # convergence diagnostic (quick and dirty)
  maxit <- c("maxit" = fit0@control@maxit)
  # Update stk0
  stk0   <- transform(stk0, harvest = fit0@harvest, stock.n = fit0@stock.n)
  return(list(stk0 = stk0, converge = maxit))
}
```

Control object for projections

```
getCtrl <- function(values, quantity, years, it){
    dnms <- list(iter=1:it, year=years, c("min", "val", "max"))
    arr0 <- array(NA, dimnames=dnms, dim=unlist(lapply(dnms, length)))
    arr0[,,"val"] <- unlist(values)
    arr0 <- aperm(arr0, c(2,3,1))
    ctrl <- fwdControl(data.frame(year=years, quantity=quantity, val=NA))
    ctrl@trgtArray <- arr0
    ctrl
}
```

MSE initialisation

```
vy <- ac(iy:fy)
TAC <- FLQuant(NA, dimnames=list(TAC="all", year=c(dy,vy), iter=1:it))
TAC[,ac(dy)] <- catch(stk)[,ac(dy)]
TAC[,ac(iy)] <- TAC[,ac(dy)] #assume same TAC in the first intermediate year
ctrl <- getCtrl(c(TAC[,ac(iy)]), "catch", iy, it)
stk <- fwd(stk, control=ctrl, sr=srbh, sr.residuals = srbh.res, sr.residuals.mult = FALSE)
```

Start the MSE loop

```
for(i in vy[-length(vy)]){
  # set up simulations parameters
    ay <- an(i)
    cat(i, " > ")
    vy0 <- 1:(ay-y0) # data years (positions vector) - one less than current year
    sqy <- (ay-y0-nsqy+1):(ay-y0) # status quo years (positions vector) - one less than current year

  # apply observation error
    oem <- o(stk, idx, i, vy0)
    stk0 <- oem$stk
    idx0 <- oem$idx
    idx <- oem$idx.om

  # perform assessment
  out.assess <- eval(call("xsa", stk0, idx0))
  stk0 <- out.assess$stk0

  # apply ICES MSY-like Rule to obtain Ftrgt
```

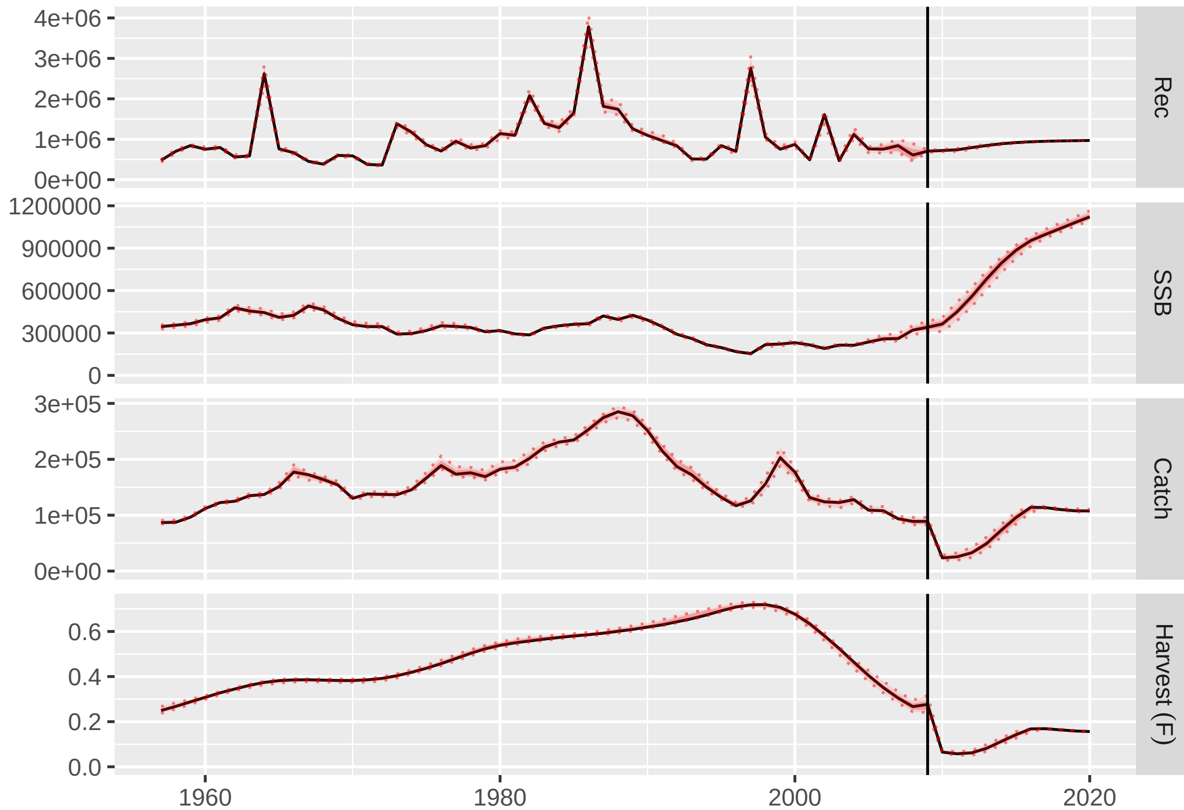

Figure 1: Figure. Operating model results for applying an ICES MSY-like rule

```
  flag <- ssb(stk0)[,ac(ay-1)]<Bpa
  Ftrgt <- ifelse(flag,ssb(stk0)[,ac(ay-1)]*Fmsy/Bpa,Fmsy)

  # project the perceived stock to get the TAC for ay+1
  fsq0 <- yearMeans(fbar(stk0)[,sqy]) # Use status quo years defined above
  ctrl <- getCtrl(c(fsq0, Ftrgt), "f", c(ay, ay+1), it)
  stk0 <- stf(stk0, 2)
  gmean_rec <- c(exp(yearMeans(log(rec(stk0)))))
  stk0 <- fwd(stk0, control=ctrl, sr=list(model="mean", params = FLPar(gmean_rec,iter=it)))
  TAC[,ac(ay+1)] <- catch(stk0)[,ac(ay+1)]

  # apply the TAC to the operating model stock
  ctrl <- getCtrl(c(TAC[,ac(ay+1)]), "catch", ay+1, it)
  stk <- fwd(stk, control=ctrl,sr=srbh, sr.residuals = srbh.res, sr.residuals.mult = FALSE)
}
```

```
## 2009  > 2010  > 2011  > 2012  > 2013  > 2014  > 2015  > 2016  > 2017  > 2018  > 2019  >
```

```
plot(stk)+geom_vline(aes(xintercept=as.numeric(ISOdate(iy,1,1))))
```

References

More information

- You can submit bug reports, questions or suggestions on this tutorial at https://github.com/flr/doc/issues.
- Or send a pull request to https://github.com/flr/doc/
- For more information on the FLR Project for Quantitative Fisheries Science in R, visit the FLR webpage, http://flr-project.org.

Software Versions

- R version 3.3.2 (2016-10-31)
- FLCore: 2.6.0.20170214
- ggplotFL: 2.5.20161007
- ggplot2: 2.2.1
- **Compiled**: Thu Feb 16 09:54:16 2017

License

Author information

Iago MOSQUEIRA. European Commission Joint Research Centre (JRC), Institute for the Protection and Security of the Citizen (IPSC), Maritime Affairs Unit, Via E. Fermi 2749, 21027 Ispra VA, Italy. https://ec.europa.eu/jrc/