Management Strategy Evaluation of Biomass Dynamic Management Procedures

Laurence Kell

August 13th, 2016

Introduction

Management Strategy Evaluation (MSE) involves using simulation to compare the relative effectiveness for achieving management objectives of different combinations of data collection schemes, methods of analysis and subsequent processes leading to management actions. MSE can be used to identify a ???best??? management strategy among a set of candidate strategies, or to determine how well an existing strategy performs André E Punt et al. (2014).

Conducting an MSE requires six steps; namely i) identification of management objectives; ii) selection of hypotheses for the OM; iii) conditioning the OM based on data and knowledge; iv) identifying candidate management strategies and coding these as a Management Procedure (MP); v) running the MP as feedback control in order to simulate the long-term impact of management; and then vi) identifying the MP that robustly meets the management objectives.

mpb

The mpb package includes methods for biomass based stock assessment, simulating harvest control rules (HCRs) and simulation testing them using MSE. It is part of FLR Kell et al. (2007).

```r
library(FLCore)
library(mpb)
library(ggplotFL)
```

Management objectives

The performance of alternative stategies are compared with respect to their ability to robustly deliver a range of management objectives, e.g. high catch, high catch rate, low variability in catches, high fishery and population resilience to environmental and other ???shocks???, low catches of nominated by-catch or by- product species.

Operating Model

The OM represents the simulated version of reality for a range of hypotheses about resource dynamics. There are many alternative ways to do this (e.g. **?**). For example a stock assessment could be used to develop the OM, or an OM could be conditioned on range of ecological and economic hypotheses about the factors that drive the system Andr?? E. Punt et al. (2014).

The use of an assessment model as an OM implies that assessment models describe nature almost perfectly, however, if a Management Procedure (MP) cannot perform well when reality is as simple as implied by an assessment model it is unlikely to perform adequately for more realistic representations of uncertainty about resource dynamics. Basing an OM on the current assessment model also has arguably the lowest demands for knowledge and data and allows RFMOs to make a phased transition from the stock assessment paradigm to a risk based approach.

There are many important processes, however, that are not modelled in stock assessments and affect the robustness of control systems (e.g. density dependence regulation in processes other than stock recruitment relationship **?**). Therefore to ensure a control system is robust also requires OMs to be conditioned based on expert beliefs and other apriori information about the processes that may affect the behaviour of management systems in the future. I.e. the focus is on the future, not on fitting historical data as when conditioning an OM on a stock assessment. This is a less data, and more hypothesis-orientated approach.

```
library(FLBRP)

library(diags)

data(om)
data(eql)

plot(eql)+
  theme_bw()+theme(legend.position="bottom")
```

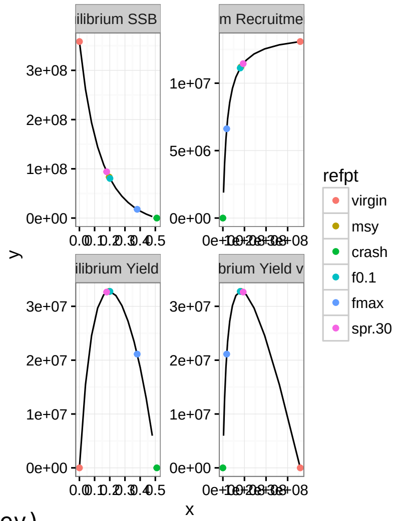


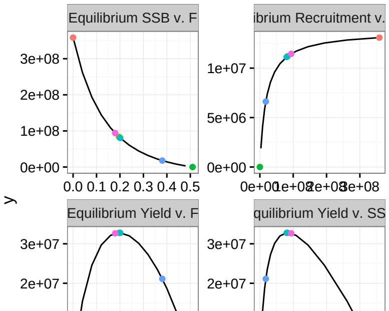
Figure 1: Expected Dynamics

Add stochasticity

```
srDev=FLQuant(0,dimnames=list(year=2000:2040))
srDev=rlnorm(100,srDev,0.3)


om=propagate(om,100)
om=fwd(om,catch=catch(om)[,ac(2000:2011)],sr=eql,sr.residuals=srDev)
```

Extend object into the future



```
options(warn=-1)
om=fwdWindow(om,eql,end=2040)
```

```
plot(om)+
  scale_x_continuous(limits=c(1930,2010))+
  theme_bw()
```

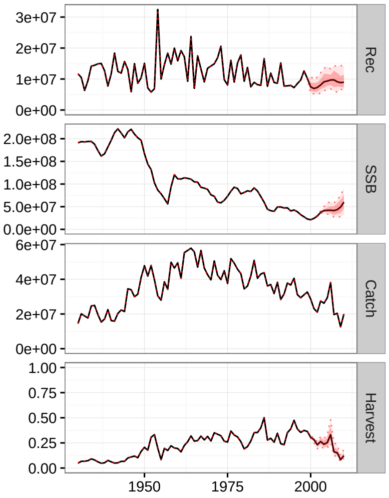


Figure 3: Time series

Alternative hypotheses about the dynamics can be modelled by FLife package

```
options(warn=-1)
library(FLife)
```

Observation Error Model

Observations that are available from the fishery and stock are simulated (i.e. psuedo data) using an Observation Error Model (OEM).

```
uDev=FLQuant(0,dimnames=list(year=ac(dims(eql)["minyear"]):ac(dims(eql)["maxyear"])))
uDev=rlnorm(100,uDev,0.3)

cpue=oem(om)
plot(cpue)
```

```
Warning: Removed 28 rows containing missing
values (geom_path).

Warning: Removed 28 rows containing missing
values (geom_path).

Warning: Removed 28 rows containing missing
values (geom_path).
```

```
#geom_path(aes(year,data),data=as.data.frame(iter(cpue,11)))
```

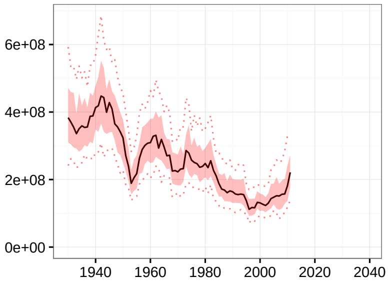


Figure 4: Simulated CPUE series

Management strategies

Manangement Procedures use a harvest control rule (HCR), based using a stock assessment (or an empirical index) that provides estimates of stock status relative to reference points, to set preagreed management regulations such as a total allowable catch (TAC).

The biodyn class can be used to estimate stock status and reference points based on biomass, to simulate HCRs.

First a object has to be created, the easiest way to do this is coerce the eql object holding the age based population parameters

```
library(popbio)
library(FLBRP)
library(mpb)
data(eql)

source('~/Desktop/flr/mpb/R/biodyn-coerce.R')

mp=FLBRP2biodyn(eql,"biodyn")
```

Initial guesses are provided for the production function, from which guesses can be derived for the nuisance parameters for catchability (q) and (σ)

```
modelParams=mpb:::modelParams
setParams(mp)=cpue
```

The `control` slot provides with the initial guesses, upper and lower bounds (`min` and `max`), and the `phase` for each parameter.

```
setControl(mp)=params(mp)
```

Running

The Management Procedure (MP) as a

FEEDBACK CONTROL in order to simulate the long-term impact of management;

```
save(om,eql,mp,file="/home/laurie/Desktop/tmp/mpb-mse.RData")
```

```
?mseBiodyn
```

```
load("/home/laurie/Desktop/tmp/mpb-mse.RData")
source('~/Desktop/flr/mpb/R/biodyn-msy.R')

srDev=FLQuant(0,dimnames=list(year=2000:2040))
srDev=rlnorm(100,srDev,0.3)

uDev=FLQuant(0,dimnames=list(year=ac(dims(eql)["minyear"]):ac(dims(eql)["maxyear"])))
uDev=rlnorm(100,uDev,0.3)

end=range(om)["maxyear"]
start=end-30
interval=3

hcrPar=function(mp,ftar=0.70,btrig=0.60,fmin=0-01,blim=0.001)
```

```
          hcrParam(ftar =ftar *fmsy(mp),
                   btrig=btrig*bmsy(mp),
                   fmin =fmin *fmsy(mp),
                   blim =blim *bmsy(mp))
bndF=NULL;bndTac=NULL;maxF=1.0
omega =1;refB  =1
qTrend=0

  ## Get number of iterations in OM
nits=c(om=dims(om)$iter, eql=dims(params(eql))$iter, rsdl=dims(srDev)$iter)
if (length(unique(nits))>=2 & !(1 %in% nits)) ("Stop, iters not ’1 or n’ in OM")
if (nits[’om’]==1) stock(om)=propagate(stock(om),max(nits))

## Cut in capacity
maxF=mean(apply(fbar(window(om,end=start)),6,max)*maxF)

## open loop feed forward
mou=om

#### Observation Error (OEM) setup #######################
## Random variation for CPUE
cpue=oem(window(om,end=start),cv=uDev,fishDepend=TRUE)
if (!("FLQuant"%in%is(qTrend)))
  qTrend=FLQuant(cumprod(rep(1+qTrend,(end+interval-as.numeric(dims(cpue)["minyear"])+1))),
                   dimnames=list(year=range(om)["minyear"]:(end+interval)))
cpue=cpue*qTrend[,dimnames(cpue)$year]

plot(cpue)

mpSmry =NULL
hcrSmry=NULL

## Loop round years
#for (iYr in seq(start,range(om,’maxyear’)-interval,interval)){

  iYr = seq(start,range(om,’maxyear’),interval)[1]
  cat(’\n===================’, iYr, ’===================\n’)

    #### OEM
    ## use data from last year
    catch(mp)=window(catch(om),end=iYr-1)
    cpue=window(cpue,end=iYr-1)
    cpue[,ac(iYr-(interval:1))]=oem(om[,ac(iYr-(interval:0))],uDev,fishDepend=TRUE)
    cpue[,ac(iYr-(interval:1))]=cpue[,ac(iYr-(interval:1))]*qTrend[,ac(iYr-(interval:1))]
    #### Management Procedure
    ## fit
    mp =fit(mp,cpue)
    mp =mpb:::fwd(mp,catch=catch(om)[,ac(iYr)])

    ## HCR
    hcrP=hcrPar(mp)
    tac=mpb:::hcr(mp,yr=iYr-1,hyr=iYr+1:3,tac=TRUE)[,-1]

    #### Operating Model feedforward
    om =fwd(om,catch=tac,maxF=maxF,sr=eql,sr.residuals=srDev)

    #### Summary Statistics
    ## HCR actions, i.e. is biomass<Btrig?, what is F?, ..
    hcrSmry=rbind(hcrSmry,data.frame(yearHcr=min(as.numeric(dimnames(hcrOutcome$hvt)$year)),
                          #yearAss=rep(range(bd)[2],dims(bd)$iter),
                          model.frame(           hcrPar,drop=T)[,-5],
                          tac    =as.data.frame(apply(hcrOutcome$tac,6,mean),drop=T)[,’data’],
                          harvest=as.data.frame(apply(hcrOutcome$hvt,6,mean),drop=T)[,’data’],
                          stock  =as.data.frame(hcrOutcome$stock,drop=T)[,2]))

    ## Assessment parameters and reference points
    mpSmry=rbind(mpSmry,cbind(cbind(year=iYr,model.frame(params(bd))),
                       model.frame(refpts(bd))[,-4],
                       hcr))

  ## save OM, projection without feedback, last assessment and MP summary
  res=list(om=om,mp=mpSmry,hcr=hcrSmry)
```

Identifying the MPs that robustly meet management objectives.

```
library(kobe)
```

References

Kell, L.T., I. Mosqueira, P. Grosjean, J.M. Fromentin, D. Garcia, R. Hillary, E. Jardim, et al. 2007. "FLR: An Open-Source Framework for the Evaluation and Development of Management Strategies." *ICES J. Mar. Sci.* 64 (4): 640.

Punt, Andr?? E., Teresa A'mar, Nicholas A. Bond, Douglas S. Butterworth, Carryn L. de Moor, Jos?? A. A. De Oliveira, Melissa A. Haltuch, Anne B. Hollowed, and Cody Szuwalski. 2014. "Fisheries Management Under Climate and Environmental Uncertainty: Control Rules and Performance Simulation." *ICES Journal of Marine Science: Journal Du Conseil* 71 (8): 2208–20. doi:10.1093/icesjms/fst057.

Punt, André E, Doug S Butterworth, Moorrryn L, José AA De Oliveira, and Malcolm Haddon. 2014. "Management Strategy Evaluation: Best Practices." *Fish and Fisheries*. Wiley Online Library.