

Cross test using Operating Model based on Life History

L Kell

23 June, 2018

OEM

```
setGeneric("cpue", function(object, ...) standardGeneric("cpue"))
```

```
[1] "cpue"
```

```
#' @rdname cpue
#' @aliases cpue,FLStock-method
setMethod("cpue", signature(object = "FLStock"), function(object, sel = catch.sel(object),
  effort = c("f", "h"), mass = TRUE) {

  if (effort[1] == "h")
    E <- catch(object)/%stock(object) else E <- fbar(object)

  cpue <- (catch.n(object) %%% sel)/%E

  if (mass)
    cpue <- cpue * catch.wt(object)

  return(cpue)
})
```

```
[1] "cpue"
```

```
# survey
setGeneric("survey", function(object, ...) standardGeneric("survey"))
```

```
[1] "survey"
```

```
setMethod("survey", signature(object = "FLStock"), function(object, sel = stock.n(object) %%%
  1, wt = stock.wt(object), timing = 0.5, mass = FALSE) {

  timing = pmax(pmin(timing, 1), 0)

  stock.n = stock.n(object) * exp(-(harvest(object) * timing + m(object) *
    timing))

  res = stock.n %%% sel

  if (mass)
    res = res %%% wt

  return(res)
})
```

```
[1] "survey"
```

```
#### Indicators
#### #####
```

```

setGeneric("mnSwt", function(object, ...) standardGeneric("mnSwt"))

[1] "mnSwt"
setGeneric("mnCwt", function(object, ...) standardGeneric("mnCwt"))

[1] "mnCwt"
setGeneric("mnLwt", function(object, ...) standardGeneric("mnLwt"))

[1] "mnLwt"
setGeneric("mnLen", function(object, ...) standardGeneric("mnLen"))

[1] "mnLen"
setGeneric("wt2z", function(object, ...) standardGeneric("wt2z"))

[1] "wt2z"
setGeneric("ln2z", function(object, Linf, ...) standardGeneric("ln2z"))

[1] "ln2z"
setMethod("mnSwt", signature(object = "FLStock"), function(object) apply(stock.wt(object) *
  stock.n(object), 2:6, sum)/apply(stock.n(object), 2:6, sum))

[1] "mnSwt"
setMethod("mnCwt", signature(object = "FLStock"), function(object) apply(catch.wt(object) *
  catch.n(object), 2:6, sum)/apply(catch.n(object), 2:6, sum))

[1] "mnCwt"
setMethod("mnLwt", signature(object = "FLStock"), function(object) apply(landings.wt(object) *
  landings.n(object), 2:6, sum)/apply(landings.n(object), 2:6, sum))

[1] "mnLwt"
setMethod("mnLen", signature(object = "FLStock"), function(object, a = 0.001,
  b = 3, wt = "stock.wt") mnLenFunc(object, a, b, wt))

[1] "mnLen"
setMethod("wt2z", signature(object = "FLStock"), function(object, a = 0.001,
  b = 3, wt = "stock.wt") wt2zFunc(object, a, b, wt))

[1] "wt2z"
setMethod("ln2z", signature(object = "numeric", Linf = "numeric"), function(object,
  Linf, Lc, k) ln2zFunc(object, Linf, Lc, k))

[1] "ln2z"
setMethod("ln2z", signature(object = "numeric", Linf = "FLPar"), function(object,
  Linf) ln2zFunc(object, Linf[Linf], Linf[Lc], Linf[k]))

[1] "ln2z"
mnLenFunc <- function(object, a = 0.001, b = 3, wt = "stock.wt") {
  wt. = slot(object, wt)
  n. = slot(object, gsub(".wt", ".n", wt))

```

```

    apply((wt./a)^(1/b) * n., c(2, 6), sum)/apply(n., c(2, 6), sum)
}

wt2zFunc <- function(object, Linf, Lc, k, a = 0.001, b = 3, wt = "stock.wt") {
  mnSz <- mnSzStock(object, a, b, wt)
  k * (Linf - mnSz)/(mnSz - Lc)
}

ln2zFunc <- function(object, Linf, Lc, k) {
  k * (Linf - object)/(object - Lc)
}
#####

```

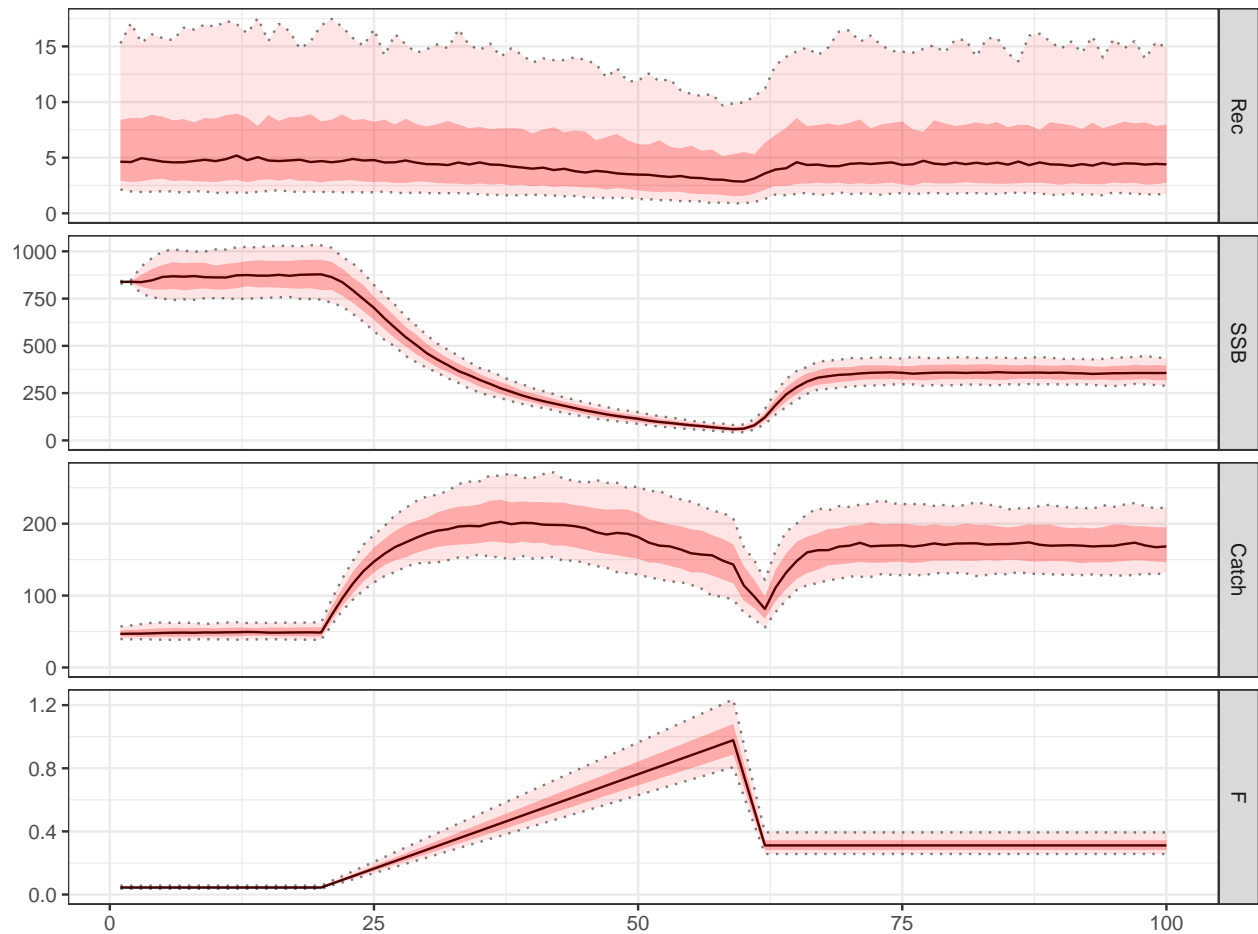


Figure 1, Operating Model Time series.

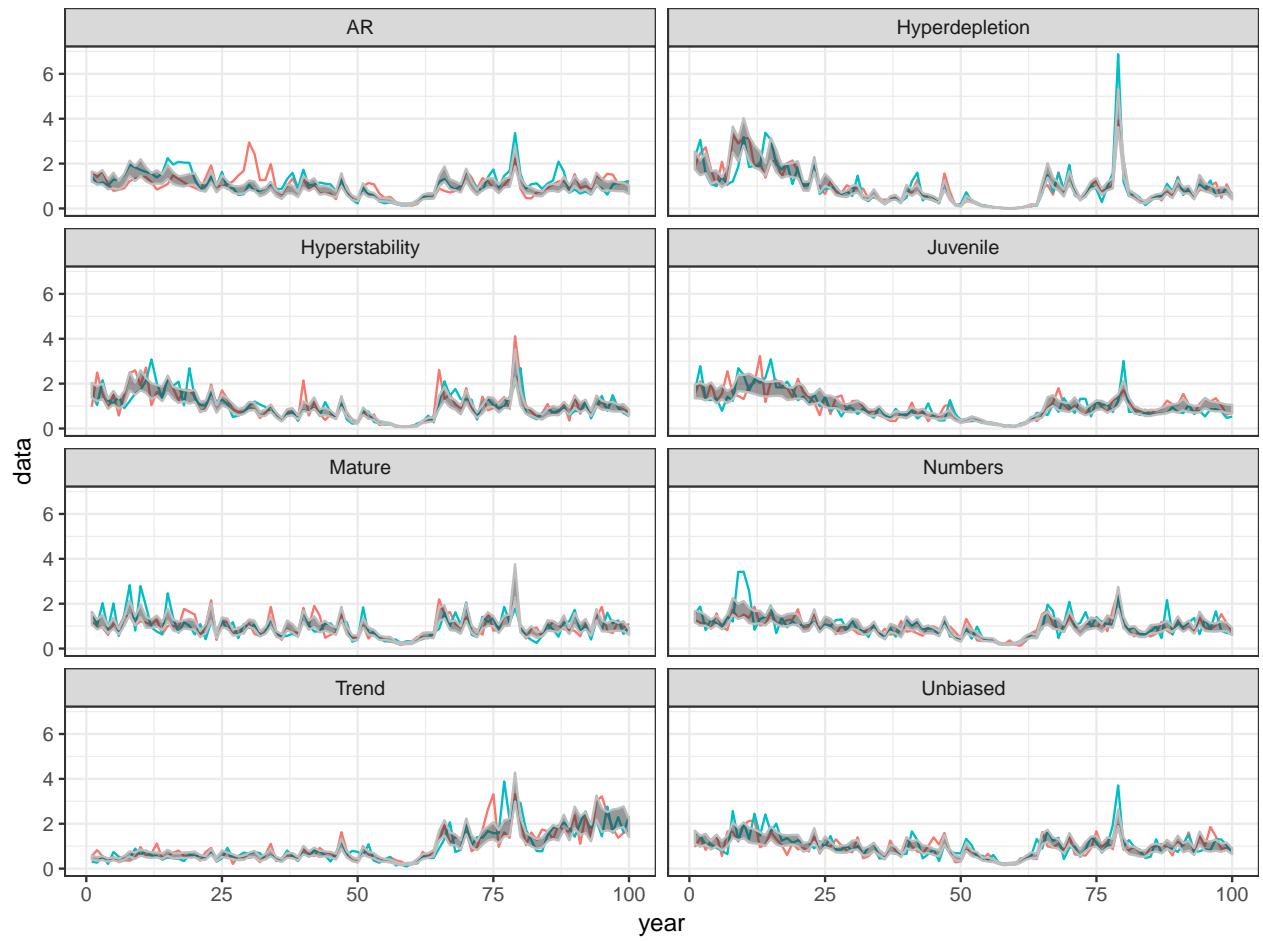


Figure 2, Commercial indices of abundance.

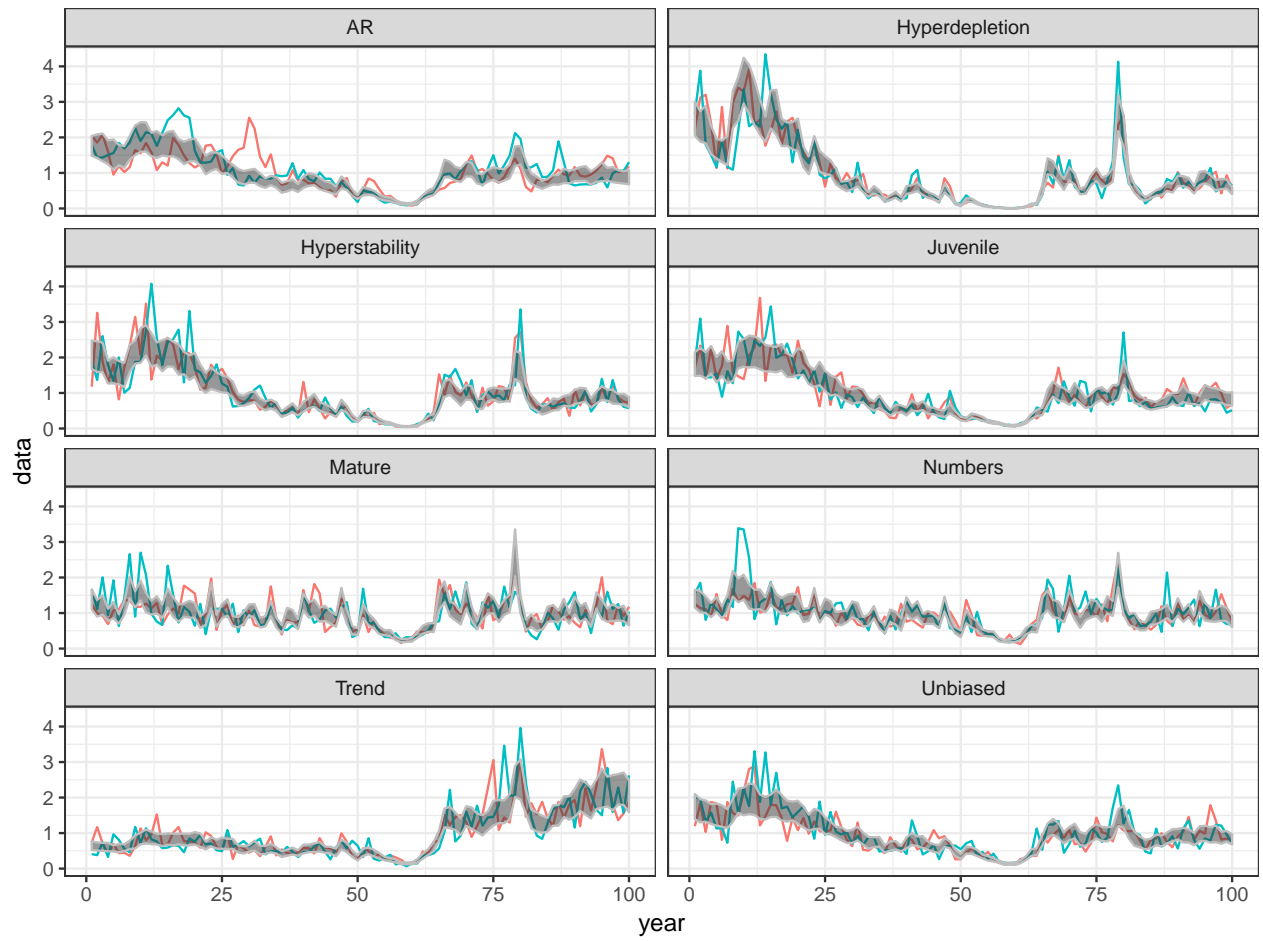


Figure 3, Survey indices of abundance.

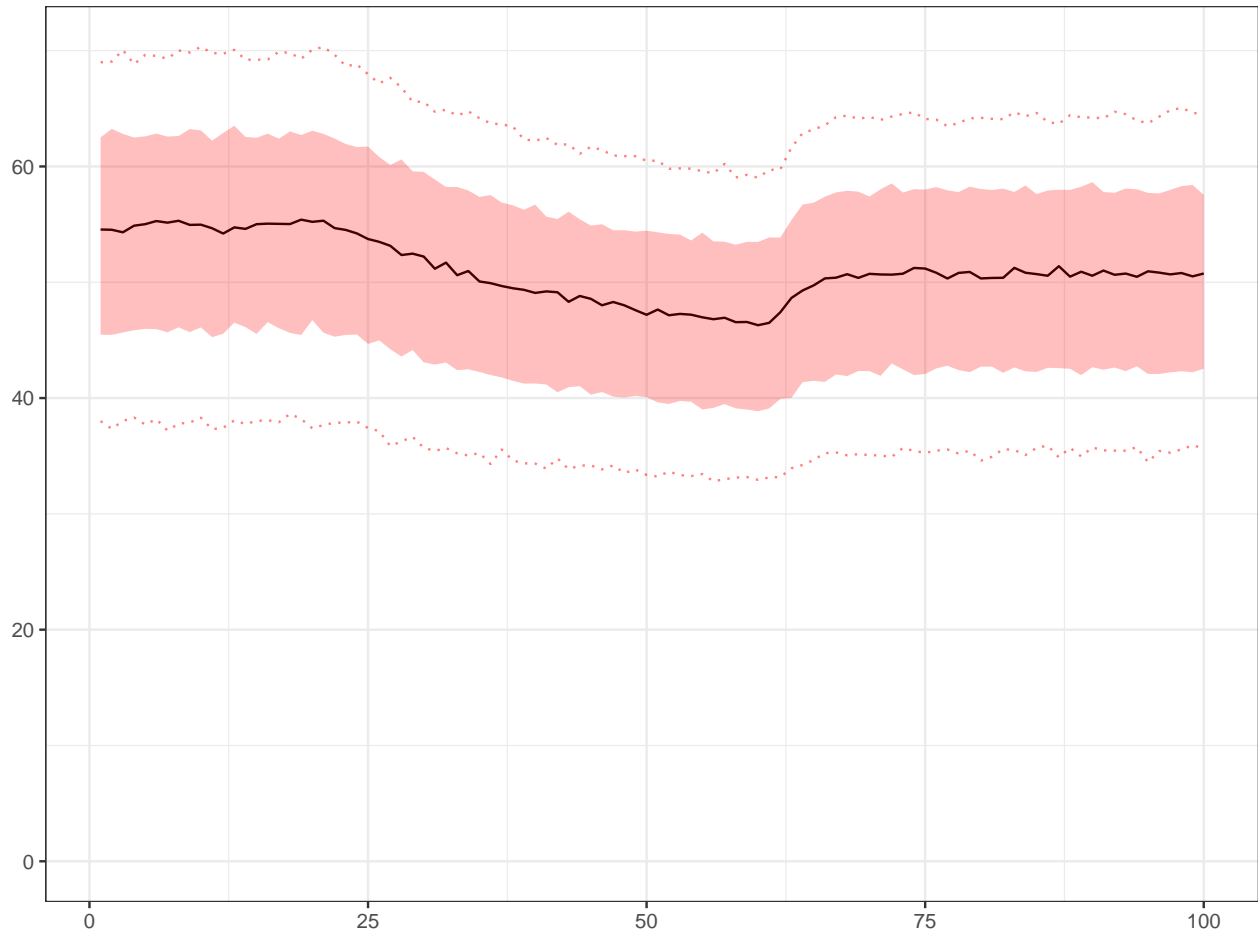


Figure 4, Index of mean length.

Stock Assessment Models