Life History Relationship

Laurence Kell

August 13th, 2014

Introduction

Many studies have shown relationships between life history traits such as growth, maturity and natural mortality. This knowledge has been used to provide advice for data poor stocks, develop priors or fix values for data rich stock assesments and to parameterise ecological models. `FLife` package brings together a variety of methods for modelling life history traits and ecological processes and can be used to create `FLR` objects such as `FLBRP` and `FLStock` in order to model species, population or stock dynamics.

```
Warning: multiple methods tables found for
'ages'
```

```
Warning: multiple methods tables found for
'dnormal'
```

```
Warning: multiple methods tables found for
'len2wt'
```

```
Warning: multiple methods tables found for
'logistic'
```

```
Warning: multiple methods tables found for
'vonB'
```

```
Warning: multiple methods tables found for
'wt2len'
```

Life history relationships

`lhPar`

In data poor situtuations only the maximum size (l_{max}) may be known. Life history relationships can be used to derive the missing parameters (Gislason et al. 2008). For example k of the (Von Bertalanffy 1957) growth equation

$$k = 3.15 l_\infty^{-0.64} \tag{1}$$

and the length at which 50% of the population mature

$$l_{50} = 0.72 l_\infty^{0.93} \tag{2}$$

The `lhPar` method takes as its first argument an `FLPar` object with as a minimum a value for `linf` and uses these relationships to derive parameters such as `k` and $l50$

```
par=lhPar(FLPar(linf=100))
```

Natural mortality can be estimated from length

$$M = 0.55(l - 1.66 l_{\infty}^{1.44} k \tag{3}$$

or mass-at-age (Lorenzen and Enberg 2002)

$$M = m_1 * w_2^m \tag{4}$$

where $m_1 = 0.55(l_{\infty}^{1.44})k$ and $m_2 = -1.61$

There are defaults for other values which can not be derived from life history theory. These include a and b from the length weight relationship $w = al^b$, `ato95` the age at which 95% of fish are mature, offset to age at which 50% are mature, `sl` selectivity-at-age parameter, standard deviation of lefthand limb of double normal, `sr` stock recruitment relationship, `s` steepness of stock recruitment relationship, `v` virgin biomass.

Biological processes as growth, maturity and natural mortality can be modelled using functions such as `vonB`, `sigmoid`, and `lorezen` methods. These take as arguments an object for age, length or weight and an `FLPar` with the life history parameters.

```
age=FLQuant(0:20,dimnames=list(age=0:20))

ln =vonB(age,par)
mat=sigmoid(age,par)
wt =par["a"]*ln^par["b"]
wt =len2wt(ln,par)
```

Selection pattern can be modelled as flat topped or dome shaped by using the double normal function

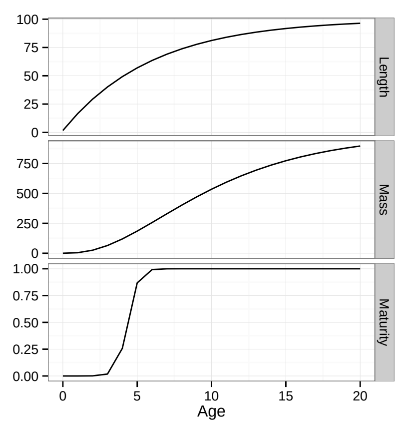


Figure 1: Biological age vectors

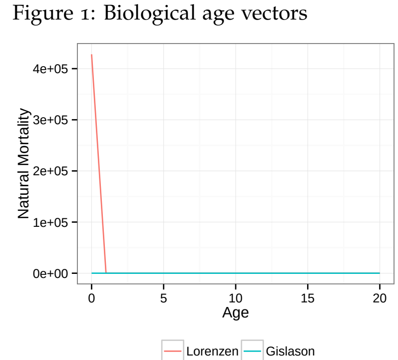


Figure 2: Natural Mortality

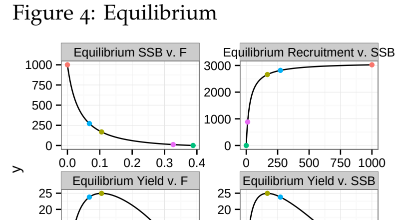
Creation of objects

FLBRP

```
Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
```

```
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore

Warning in .local(x, i, j, ...): using a
local copy of '[[' which will be removed in
later versions of FLCore
```

Figure 4: Equilibrium



FLStock

Stability

An important factor determining a population's response to perturbation is stability which can be measured in a variety of ways (e.g. Pimm 1984). In its simplest form, a population can be considered stable if it returns to equilibrium after a perturbation. Other definitions expand on this and involve the time taken to return to equilibrium after a perturbation, known as the characteristic return time or population resilience. The lower the characteristic return time, or higher the resilience, the more stable the population. The stability of a population is strongly influenced by the life history of the population and also the pattern of density dependence. For some population models the stability is a good indicator of a population's response to noise (Taylor 1992), but generally stability is insufficient on its own to predict the response (Horwood 1993). Here we use it to indicate how quickly management can cause an effect in a population, e.g. to recover a stock to a level that would support MSY. In this way, stability can be used as a guide to how controllable the stock is. For discrete, structured populations this can be calculated using the magnitude of the dominant eigenvalue of the Jacobian matrix evaluated at the equilibrium point (Beddington 1974; Caswell 2001). If the magnitude of this is less than 1 the population will return to equilibrium after a disturbance, with the stability decreasing as the magnitude approaches 1. When the magnitude of the dominant eigenvalue is 1 there is a bifurcation and past this point non-equilibrium dynamics, including extinction, are seen.

```
Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used
```

```
Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array
```

```
Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc
```

```
Warning in .local(params, ...): Should be no
need to specify mode of FLPar element
```

```
Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work
```

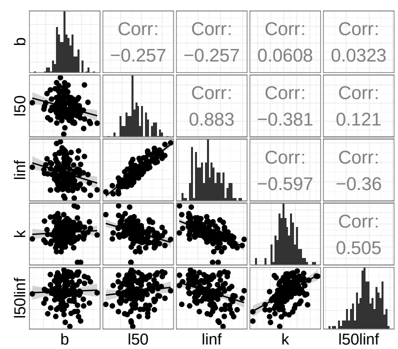


Figure 6: Overfish

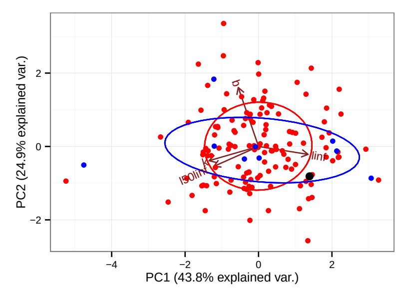


Figure 7: Rebuild

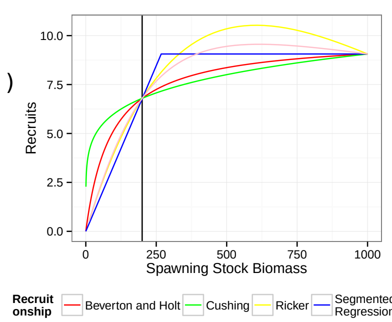
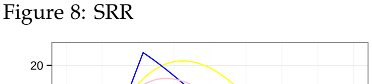


Figure 8: SRR



```
Warning in .local(params, ...): iter(FLBRP,i)
doesn’t work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn’t work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn’t work
```

```
Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used
```

```
Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn’t work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn’t work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc
```

```
Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work
```

```
Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work
```

```
Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn't work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used
```

```
Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn’t work

Warning in if (spwn > 1 | spwn < 0 | fish
> 1 | fish < 0) stop("spwn and fish must be
in the range 0 to 1\n"): the condition has
length > 1 and only the first element will be
used

Warning in .local(x, i, ...): Selected
elements do not form a coherent 6D array

Warning in .local(params, ...): Scarab, iters
dont work for SRR:sv/ab etc

Warning in .local(params, ...): Should be no
need to specify mode of FLPar element

Warning in .local(params, ...):
iter(params(res),i)=ab(params[c(s,v),i],sr,spr0=iter(spr0(res),i))
[c(a,b)] assignment doesnt work

Warning in .local(params, ...): iter(FLBRP,i)
doesn’t work


Warning: Removed 2250 rows containing missing
values (geom_path).

Warning: Removed 2250 rows containing missing
values (geom_path).

Warning: Removed 2250 rows containing missing
values (geom_path).
```

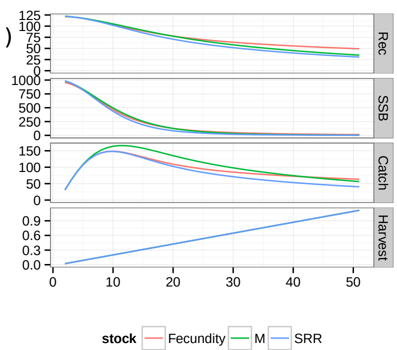


Figure 10: Natural Mortality

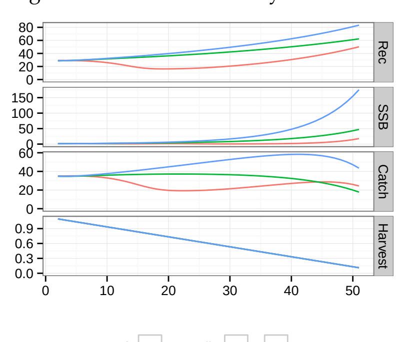
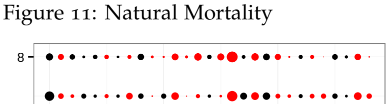


Figure 11: Natural Mortality



```
Warning: Removed 2250 rows containing missing
values (geom_path).

Warning: Removed 2250 rows containing missing
values (geom_path).

Warning: Removed 2250 rows containing missing
values (geom_path).

[1] 1

Warning in ‘[[<-‘(‘*tmp*‘, ac(args[,
1]), value = structure(list(stock.wt =
structure(c(0.0219656777887209, : using a
local copy of ’[[<-’ which will be removed in
later versions of FLCore

=================== 101 ===================
=================== 104 ===================
=================== 107 ===================
=================== 110 ===================
=================== 113 ===================
=================== 116 ===================
=================== 119 ===================
```

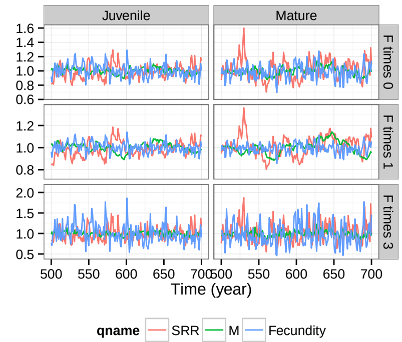


Figure 14: Natural Mortality

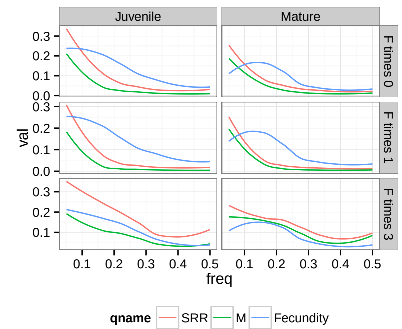


Figure 15: Natural Mortality

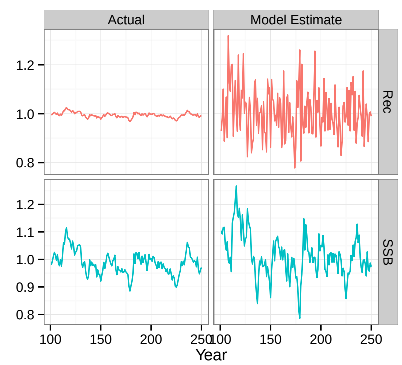


Figure 16: Natural Mortality Mis-specification

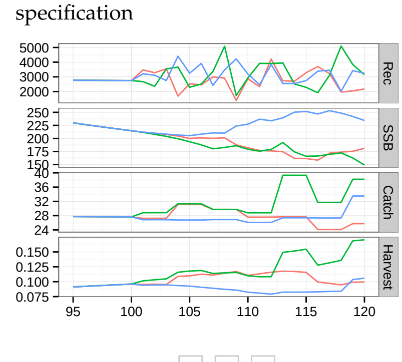
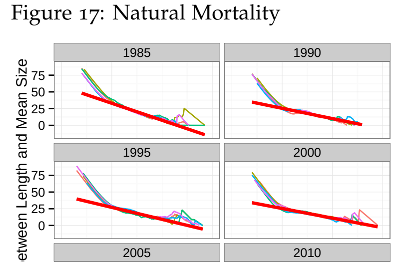


Figure 17: Natural Mortality



Elasticity

A measure of proportional effect, i.e., the effect that a change in a given matrix element has as a proportional to the change in that element

Gislason, H., J.G. Pope, J.C. Rice, and N. Daan. 2008. “Coexistence in North Sea Fish Communities: implications for Growth and Natural Mortality.” *ICES Journal of Marine Science: Journal Du Conseil* 65 (4): 514–530.

Lorenzen, Kai, and Katja Enberg. 2002. “Density-Dependent Growth as a Key Mechanism in the Regulation of Fish Populations: evidence from Among-Population Comparisons.” *Proceedings of the Royal Society of London. Series B: Biological Sciences* 269 (1486): 49–54.

Von Bertalanffy, L. 1957. “Quantitative Laws in Metabolism and Growth.” *Quarterly Review of Biology*: 217–231.