Demonstrating EcoState via simulation

James Thorson

```
library(EcoState)
```

ecostate is an R package for fitting the mass-balance dynamics specified by EcoSim as a state-space model. We here highlight a few features in particular.

Simulation demonstration

We first simulate new data. To do so, we specify a 5-species ecosystem:

```
# Time-interval and number of species
n_years = 40
n_species = 5

# Ecopath-with-EcoSim parameters
DC_ij = matrix( c(
  0, 1, 0, 0.5, 0,
  0, 0, 1, 0.5, 0,
  0, 0, 0, 0,   0.5,
  0, 0, 0, 0,   0.5,
  0, 0, 0, 0,   0
), byrow=TRUE, ncol=n_species)
PB_i = c( 90, 5, 0.2, 3, 0.1 )         # Reciprocal of mean age according to Polovina-1984 ~= M
QB_i = c( 150, 10, 3, 15, 4 )
V_ij = matrix( 2, nrow=n_species, ncol=n_species )
B0_i = c(1, 1, 1, 1, 1)

# Simulation parameters
fished_i = c( FALSE, FALSE, TRUE, FALSE, TRUE )
sigmaB_i = c(0.1, 0.1, 0.1, 0.1, 0.1)
```

We then loop through years while projecting dynamics, given that the two consumers are subject to increased fishing over time:

```
# Simulate process errors
rarray = \(x,dims=dim(x),sd) array( sd*rnorm(prod(dims)), dim=dims )
deltaB_ti = rarray(dims=c(n_years,n_species),sd=1) * outer(rep(1,n_years),sigmaB_i)

# Choose solver and integration method
n_steps = 10
project_vars = abm3pc_sys

# Project forward
```

```
Bobs_ti = Cobs_ti = B_ti = C_ti = matrix(NA, nrow=n_years, ncol=n_species )
B_ti[1,] = B0_i
C_ti[1,] = NA
for( t in seq_len(n_years)[-1] ){
  # Fishing mortality ramps up for two predators
  F_t = c( 0, 0, t/n_years*0.2, 0, t/n_years*0.1 )

  # Integrate dynamics annually
  sim = project_vars(
        f = dBdt,
        a = 0,
        b = 1,
        n = n_steps,
        Pars = list( logB_i = log(B0_i),
                     logPB_i = log(PB_i),
                     logQB_i = log(QB_i),
                     logV_ij = log(V_ij),
                     DC_ij = DC_ij,
                     deltaB_i = deltaB_ti[t,],
                     logF_i = log(F_t),
                     n_species = n_species ),
        y0 = c( B_ti[t-1,], rep(0,n_species) ) )

  # Record results
  B_ti[t,] = sim$y[nrow(sim$y),seq_len(n_species)]
  C_ti[t,] = sim$y[nrow(sim$y),n_species+seq_len(n_species)]

  # Simulate measurement errors
  Bobs_ti[t,] = B_ti[t,] * exp(0.1*rnorm(n_species))
  Cobs_ti[t,] = ifelse(C_ti[t,]==0,NA,C_ti[t,]) * exp(0.1*rnorm(n_species))
}

#
dimnames(Bobs_ti) = dimnames(Cobs_ti) = list( "Year" = seq_len(n_years),
                                              "Taxon" = seq_len(n_species) )
```

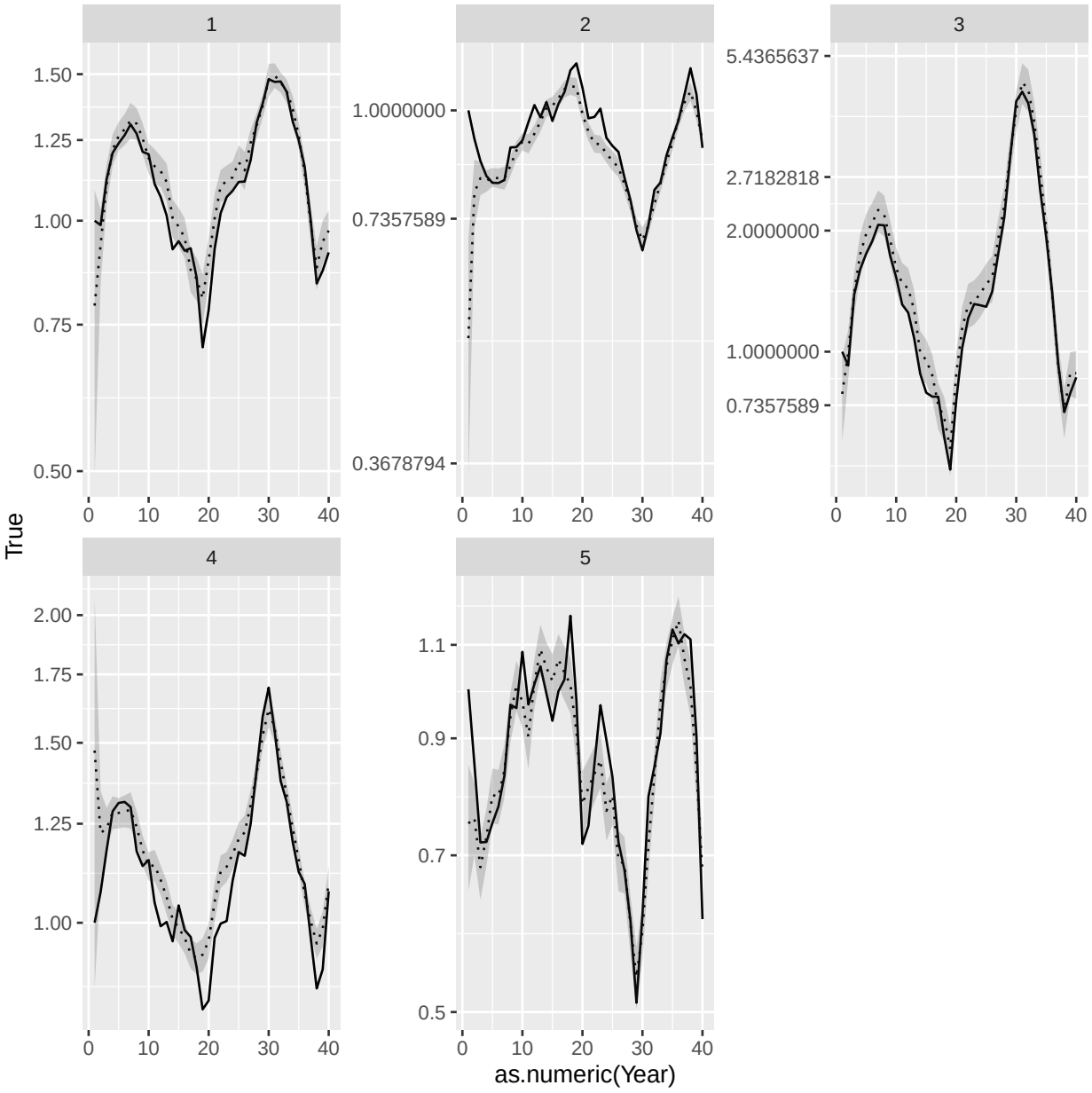
Next, we reformat simulated biomass and catch time-series into long-form data frames and fit them with `ecostate`

```
# reformat to longform data-frame
Catch = na.omit(data.frame(expand.grid(dimnames(Cobs_ti)), "Mass"=as.vector(Cobs_ti)))
Biomass = na.omit(data.frame(expand.grid(dimnames(Bobs_ti)), "Mass"=as.vector(Bobs_ti)))

# Domain
taxa = seq_len(n_species)
years = seq_len(n_years)

# Settings: specify what parameters to estimate
fit_delta = 1:5   # process errors
fit_Q = 1:5       # catchability coefficient
fit_B0 = 1:5      # non-equilibrium initial condition
fit_B = c()       # equilibrium biomass
```

```
# Label EwE inputs for each taxon as expected (so users can easily change taxa)
names(PB_i) = names(QB_i) = names(B0_i) = taxa
  dimnames(DC_ij) = list("Prey"=taxa, "Predator"=taxa)

# Run model
out = ecostate( taxa = taxa,
                years = years,
                catch = Catch,
                biomass = Biomass,
                PB = PB_i,
                QB = QB_i,
                DC = DC_ij,
                B = B0_i,
                fit_B = fit_B,
                fit_Q = fit_Q,
                fit_eps = fit_delta,
                fit_B0 = fit_B0,
                control = ecostate_control( trace=0 ) )
```

Finally, we can extract elements from the fitted model, and plot them easily using ggplot2 to compare them with known (simulated) values:

```
# Extract estimated biomass
Bhat_ti = as.list(out$sdrep, what="Estimate", report=TRUE )$Bhat_ti
Bse_ti = as.list(out$sdrep, what="Std. Error", report=TRUE )$Bhat_ti

# Reformat to long-form data frame for ggplot
results = expand.grid(dimnames(Bobs_ti))
results = cbind( results,
                 "True" = as.vector(B_ti),
                 "Est" = as.vector(Bhat_ti),
                 "SE" = as.vector(Bse_ti) )

# Plot using ggplot
library(ggplot2)
ggplot(results) +
  geom_line( aes(x=as.numeric(Year), y=True) ) +
  facet_wrap( vars(Taxon), scale="free" ) +
  geom_line( aes(x=as.numeric(Year), y=Est), linetype="dotted" ) +
  geom_ribbon( aes(x=as.numeric(Year), ymin=Est-SE, ymax=Est+SE), alpha=0.2) +
  scale_y_continuous(trans='log')
```


1
2
3
4
5
True
as.numeric(Year)
1.50
1.25
1.00
0.75
0.50
1.0000000
0.7357589
0.3678794
5.4365637
2.7182818
2.0000000
1.0000000
0.7357589
2.00
1.75
1.50
1.25
1.00
1.1
0.9
0.7
0.5
0
10
20
30
40