Some notes on setting up HCR simulations using the fishvise package

Einar Hjörleifsson

November 21, 2013

```
## Loading required package:  fishvise
## Loading required package:  FLCore
## Loading required package:  grid
## Loading required package:  lattice
## Loading required package:  MASS
## FLCore 2.5.0 development version
##
## Attaching package:  'FLCore'
## The following object is masked from 'package:plyr':
##
##    desc
## The following object is masked from 'package:ggplot2':
##
##    %+%
## The following object is masked from 'package:base':
##
##    cbind, rbind
```

- R version 3.0.1 (2013-05-16), `x86_64-redhat-linux-gnu`
- Base packages: base, datasets, graphics, grDevices, grid, methods, stats, utils
- Other packages: data.table 1.8.8, fishvise 0.01, FLCore 2.5.20131114, ggplot2 0.9.3.1, knitr 1.2, lattice 0.20-15, lubridate 1.3.0, MASS 7.3-27, plyr 1.8, RColorBrewer 1.0-5, reshape2 1.2.2, scales 0.2.3, stringr 0.6.2
- Loaded via a namespace (and not attached): colorspace 1.2-2, dichromat 2.0-0, digest 0.6.3, evaluate 0.4.4, formatR 0.8, gtable 0.1.2, labeling 0.2, munsell 0.4.2, proto 0.3-10, stats4 3.0.1, tools 3.0.1

```
[1] "This document was created in knitr"
```

Contents

1 Preamble

some nice stuff

2 A quick start

Just to give a hint where we are heading:

```
suppressPackageStartupMessages(require(fishvise))
ctr_dim <- list(a1 = 0, a2 = 14, y1 = 2011, y2 = 2031, iter = 1000,
    f1 = 5, f2 = 10, nR = 1, HRATE = c(0.001, 0.05, 0.1, 0.125, seq(0.15,
        0.3, by = 0.01)))
ctr_str <- list(tac_y1 = 170, tac_y2 = 180, y1Bias = 1)
ctr_rec <- list(cv = 0.32, rho = 0, model = 1, r_mean = 128, ssb_break = 220)
ctr_ass <- list(cv = 0.15, rho = 0.45, error_type = 1, bias = 1)
ctr_wgt <- list(cv = 0.12, rho = 0.6, error_type = 2, refBweight = 1)
ctr_hcr <- list(alphaTAC = 0.5, Btrigger = 0, delay = 0, model_number = 3)
ctr_imp <- list(beta1 = 1e-10, beta2 = 1)

d <- hcr_set_dimensions(ctr_dim)
d$cvcW <- hcr_set_wgtErrors(d$cvcW, ctr_wgt)
d$cvsW <- d$cvcW
d$cvR <- hcr_set_recErrors(d$cvR, ctr_rec)
d$assError <- hcr_set_assErrors(d$assError, ctr_ass)
file <- paste(path.package("fishvise"), "extdata/hcr_iCod.dat", sep = "/")
dat_y1 <- hcr_read_startfile(file)
d <- hcr_set_starting_conditions(dat_y1, d, ctr_str, wgt_error_type = ctr_wgt$error_type,
    refBweight = ctr_wgt$refBweight)

HRATE <- ctr_dim$HRATE
n_years <- ctr_dim$y2 - ctr_dim$y1 + 1
n_iters <- ctr_dim$iter
```

Do the run:

```
# First year
y <- 1
for (h in 1:length(HRATE)) {
    # Estimates the true fishing mortality
    Fmult <- hcr_TAC_to_Fmult(d, y, h)
    # Operating model
    d <- hcr_operating_model(d, y, h, ctr_rec, Fmult, nR = 1)
}
# The inbetween years
for (h in 1:length(HRATE)) {
    for (y in 2:(n_years - 1)) {
        # Implementation error
        d$TAC[y, h, ] <- hcr_implementation(d$TAC[y, h, ], d$TAC[y -
            1, h, ], ctr_imp$beta1, ctr_imp$beta2)
        # Estimates the true fishing mortality
        Fmult <- hcr_TAC_to_Fmult(d, y, h)
        # Operating model
        d <- hcr_operating_model(d, y, h, ctr_rec, Fmult, nR = 1)
        # Observation model
        hat <- hcr_observation_error(d, y, h, HRATE[h], Fmult, ctr_hcr$delay,
            ctr_ass$bias, ctr_ass$error_type)
        # Decision model (TAC next year)
        d$TAC[y + 1, h, ] <- switch(ctr_hcr$model_number, stop("not yet implemented"),
```

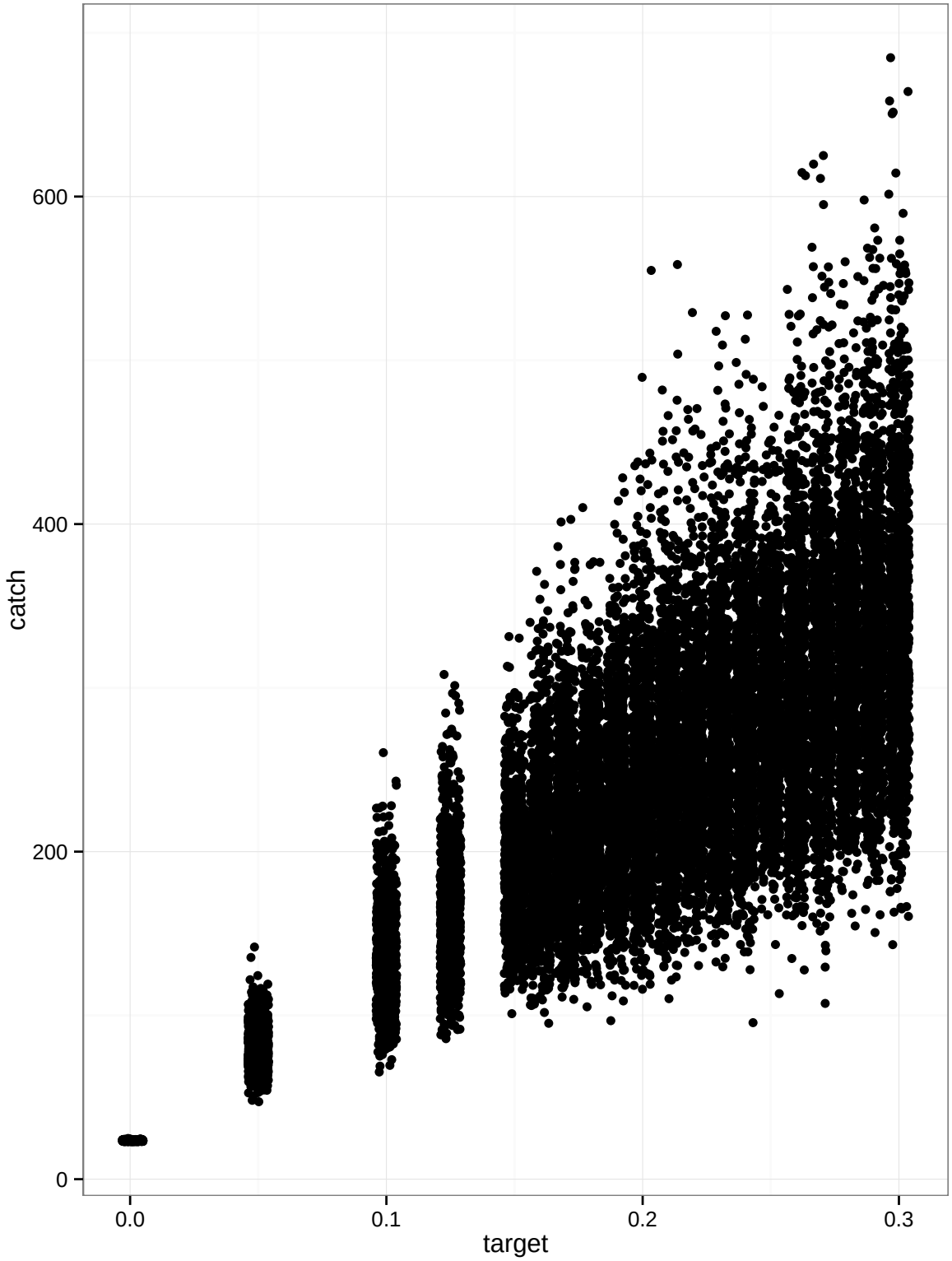
```
            hcr_management_fmort(), hcr_management_bio(hat$bio, hat$ssb,
                rep(HRATE[h], n_iters), ctr_hcr$Btrigger, d$TAC[y, h,
                  ], ctr_hcr$alphaTAC))
    }
}
# Last year
y <- n_years
for (h in 1:length(HRATE)) {
    # Estimates the true fishing mortality
    Fmult <- hcr_TAC_to_Fmult(d, y, h)
    # Operating model
    d <- hcr_operating_model(d, y, h, ctr_rec, Fmult, nR = 1)
}
```

Summarise the data:

```
sY <- melt(colSums(d$C * d$cW))
sB <- melt(colSums(d$N * d$cW * d$selB))
sS <- melt(colSums(d$N * exp(-(d$pM * d$M + d$pF * d$tF)) * d$sW * d$mat))
sR <- melt(drop(d$N[1, , , ]))
sF <- melt(colMeans(d$tF[(ctr_dim$f1 + 1):(ctr_dim$f2 + 1), , , ]))
dat <- data.frame(year = sY$year, iter = sY$iter, target = sY$hrate,
    rec = sR$value, refbio = sB$value, ssb = sS$value, relssb = rep(NA,
        nrow(sS)), fbar = sF$value, catch = sY$value, hr = sY$value/sB$value)
```

And plot the illusive MSY:

```
i <- dat$year %in% 2015
ggplot(dat[i, ], aes(target, catch)) + geom_jitter()
```


600
400
200
0
catch
0.0
0.1
0.2
0.3
target