

# Examples of using the plotMSY

Einar Hjörleifsson

February 2, 2014

- R version 3.0.2 (2013-09-25), `x86_64-redhat-linux-gnu`
- Base packages: base, datasets, graphics, grDevices, grid, methods, stats, utils
- Other packages: data.table 1.8.8, FLCore 2.5.20140123, ggplot2 0.9.3.1, gridExtra 0.9.1, knitr 1.2, lattice 0.20-15, lubridate 1.3.0, MASS 7.3-27, mgcv 1.7-27, msy 0.1.9, nlme 3.1-110, plyr 1.8, R2admb 0.7.10, RColorBrewer 1.0-5, reshape2 1.2.2, scales 0.2.3, scam 1.1-6, stringr 0.6.2, xtable 1.7-1
- 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, Matrix 1.0-12, munsell 0.4.2, proto 0.3-10, stats4 3.0.2, tools 3.0.2

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

# 1 plotMSY

## 2 Quick start

Compile:

```
srmsync_compile()
```

Convert FLStock object to srmsync data files, use last three years as the basis for biological and selection information (not done here, use prepared .dat files):

```
stk <- icesStocks$codNS
#FLS2srmsync(stk,y=3,"codNS","age.dat",1)
cmd <- paste('cp',paste(path.package("msy"),'extdata/age.dat',sep='/'),'.')
system(cmd)
cmd <- paste('cp',paste(path.package("msy"),'extdata/srmsync.dat',sep='/'),'.')
system(cmd)
rby <- fls2rby(stk)
rby[,c(1,2,4,5)] <- rby[,c(1,2,4,5)]/1e3
```

Run the srmsync

```
srmsync_run(sr=1,path="ricker",echo=FALSE, windose=FALSE)
srmsync_run(sr=2,path="bevholt",echo=FALSE, windose=FALSE)
srmsync_run(sr=3,path="segreg",echo=FALSE, windose=FALSE)
```

Plot the ssb recruitment fits (png files):

```
srmsync_plotcomposit_ssbr(rby)
```

```
## pdf
## 2
```

Plot the refs and yield and ssb as a funtion of fishing mortality (png file):

```
srmsync_plotcomposit_yield(path="ricker",rby=rby)

## pdf
## 2

srmsync_plotcomposit_yield(path="bevholt",rby=rby)

## pdf
## 2

srmsync_plotcomposit_yield(path="segreg",rby=rby)

## pdf
## 2
```

```
x1 <- srmsync_read_yield("ricker")
x2 <- srmsync_read_yield("bevholt")
x3 <- srmsync_read_yield("segreg")
qs <- rbind(x1$quantiles,x2$quantiles,x3$quantiles)
d <- rbind(x1$stochastic,x2$stochastic,x3$stochastic)

plotTEST <- function(type="yield") {
  if(type == "yield") {
```

```

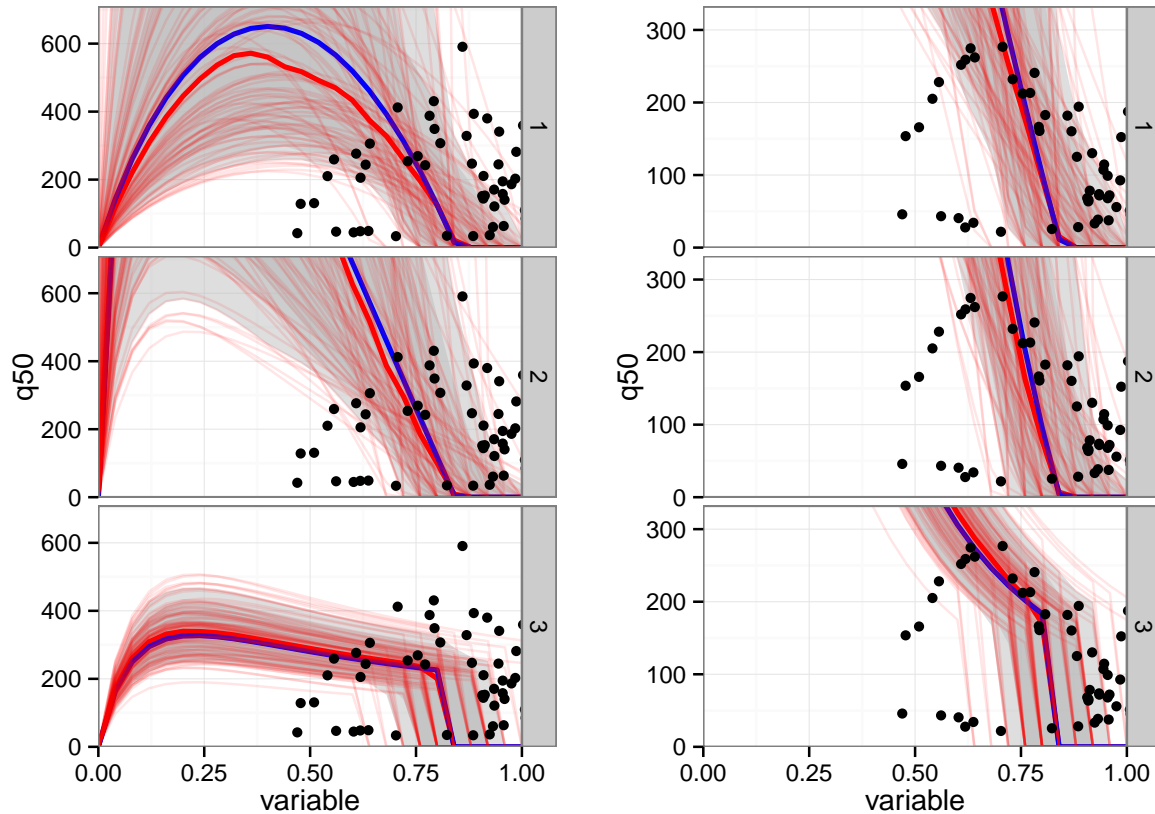
  x1 <- srmsymc_read_yield("ricker")
  x2 <- srmsymc_read_yield("bevholt")
  x3 <- srmsymc_read_yield("segreg")
}
if(type == "ssb") {
  x1 <- srmsymc_read_ssb("ricker")
  x2 <- srmsymc_read_ssb("bevholt")
  x3 <- srmsymc_read_ssb("segreg")
}

qs <- rbind(x1$quantiles,x2$quantiles,x3$quantiles)
d <- rbind(x1$stochastic,x2$stochastic,x3$stochastic)

p <- ggplot(qs,aes(variable)) +
  geom_ribbon(aes(ymin=q05,ymax=q95),fill='grey',alpha=1/2) +
  geom_ribbon(aes(ymin=q10,ymax=q90),fill='grey',alpha=1/2) +
  geom_ribbon(aes(ymin=q16,ymax=q84),fill='grey',alpha=1/2) +
  geom_line(aes(y=q50),col='red',lwd=1) +
  geom_line(aes(y=mean),col='blue',lwd=1) +
  geom_line(data=d,aes(variable,value,group=iter),alpha = 0.1,col='red') +
  facet_grid(srno ~ .)
return(p)
}

p <- plotTEST("yield")
p1 <- p + geom_point(data=rby,aes(fbar,catch)) +
  coord_cartesian(xlim=c(0,1),ylim=c(0,1.2*max(rby$catch)))
p <- plotTEST("ssb")
p2 <- p + geom_point(data=rby,aes(fbar,ssb)) +
  coord_cartesian(xlim=c(0,1),ylim=c(0,1.2*max(rby$ssb)))
vplayout <- function(x, y) viewport(layout.pos.row = x, layout.pos.col = y)
pushViewport(viewport(layout = grid.layout(1, 2)))
print(p1, vp=vplayout(1,1))
print(p2, vp=vplayout(1,2))

```



## 2.1 Setup

### Compiling the ADMB code

The uncompiled srmsync ADMB-codes come with the `msy-package`. They can be located on your computer using the following command:

```
paste(path.package("msy"), "tpl", sep="/")
```

To compile the code as well as for further communication with the program you need to have the `R2admb` library installed and loaded as well of course the ADMB-program itself. Once these are in place compiling the code is done by:

```
srmsync_compile()
```

This command copies the `tpl`-files (uncompiled ADMB codes, named `srmsync.tpl` and `srmsync2.tpl`) into your current working directory and compiles them. You should now have four new files in the working directory, the two `.tpl` files and then `srmsync` and `srmsync2` (or if on windose `srmsync.exe` and `srmsync2.exe`). I.e. you should have something like this in your directory:

```
## [1] "srmsync"      "srmsync.dat"  "srmsync.tpl"  "srmsync2"
## [5] "srmsync2.tpl"
```

### The data files

The `srmsync` programs read data from separate files. Examples are provided with the `msy-package` in the following directory your computer using the following command:

```
paste(path.package("msy"), "extdata", sep="/")
```

Example output of these files are provided in the appendix. Further information about the meaning of various xxx can be found in xxxx.

APPENDIX They look like this:

```
readLines("ricke/srmsymc.dat")

## [1] "# Header: Some nice description"
## [2] "codNS # stkname: Name of the stock"
## [3] "age.dat # filename: Name of the option file (2nd file"
## [4] "1963 # ybeg: First year"
## [5] "2011 # yend: Last year"
## [6] "1 # r: Recruitment age"
## [7] "7 # A: Plus group age"
## [8] "1 # Ropt: S-R function type"
## [9] "1 # simopt: 0=no simulation, 1=simulation"
## [10] "1 # senopt"
## [11] "1 # penopt"
## [12] "# r ssb"
## [13] " 465.56 151.90"
## [14] " 852.56 164.23"
## [15] "1069.82 203.82"
## [16] "1379.18 227.29"
## [17] "1271.87 251.45"
## [18] " 656.71 262.24"
## [19] " 606.22 258.59"
## [20] "1839.49 273.76"
## [21] "2369.05 276.23"
## [22] " 584.20 241.35"
## [23] " 875.02 213.20"
## [24] " 807.74 232.35"
## [25] "1377.80 212.99"
## [26] " 849.16 182.96"
## [27] "2096.96 161.14"
## [28] "1271.87 160.33"
## [29] "1435.47 166.71"
## [30] "2273.88 181.50"
## [31] " 885.58 194.66"
## [32] "1407.04 188.34"
## [33] " 819.13 154.97"
## [34] "1426.88 132.46"
## [35] " 378.51 126.63"
## [36] "1692.98 115.73"
## [37] " 671.32 108.99"
## [38] " 462.31 100.61"
## [39] " 767.58 94.28"
## [40] " 333.70 80.18"
## [41] " 370.28 73.64"
## [42] " 792.54 71.11"
## [43] " 446.86 68.60"
## [44] " 944.11 72.11"
## [45] " 557.94 81.06"
## [46] " 403.53 79.22"
## [47] "1059.17 75.21"
## [48] " 170.59 61.27"
## [49] " 303.46 55.94"
## [50] " 548.53 49.66"
## [51] " 208.98 41.73"
## [52] " 255.76 42.57"
## [53] " 119.13 36.90"
## [54] " 200.19 31.98"
## [55] " 137.45 29.76"
```

```
(yearRange[1])
(yearRange[2])
(senhead[1])"
(senhead[2])"
(ifelse(nits==0,0,
```

```
## [56] " 344.90  26.24"
## [57] " 147.12  32.83"
## [58] " 178.80  38.25"
## [59] " 191.19  47.19"
## [60] " 326.44  51.79"
## [61] " 165.71  56.33"
```

and:

```
readLines("ricker/age.dat")

## [1] "#Header: Some nice description"
## [2] "2 # fno: Number of fleets (nstocks)"
## [3] "1 # sno: Fleets for yield per recruit stats (always 1)"
## [4] "0 # f: proportional fishing mortality before spawning time (pf)"
## [5] "0 # m: proportional natural mortality before spawning time (pm)"
## [6] "# Selection pattern"
## [7] "0.0273348519362187 0.2186788154897490 "
## [8] "0.3006833712984050 0.4783599088838270 "
## [9] "0.8251708428246010 0.2716400911161730 "
## [10] "1.0216400911161700 0.1025056947608200 "
## [11] "1.1070615034168600 0.0358769931662870 "
## [12] "1.1924829157175400 0.0307517084282460 "
## [13] "1.1788154897494300 0.0444191343963554 "
## [14] "# cv Selection pattern"
## [15] "0.175 0.175 "
## [16] "0.097 0.097 "
## [17] "0.076 0.076 "
## [18] "0.079 0.079 "
## [19] "0.085 0.085 "
## [20] "0.109 0.109 "
## [21] "0.109 0.109 "
## [22] "# Weight at age"
## [23] "0.747 0.286 "
## [24] "1.306 0.755 "
## [25] "2.634 1.726 "
## [26] "4.236 3.542 "
## [27] "6.104 5.964 "
## [28] "7.642 6.674 "
## [29] "9.701 8.823"
## [30] "# cv Weight at age"
## [31] "0.112 0.255 "
## [32] "0.143 0.282 "
## [33] "0.137 0.419 "
## [34] "0.097 0.542 "
## [35] "0.062 0.480 "
## [36] "0.050 0.345 "
## [37] "0.040 0.792 "
## [38] "# Biological data"
## [39] "#      M,      mat,      wSSB"
## [40] "1.0385 0.01 0.339"
## [41] "0.6975 0.05 0.964"
## [42] "0.4895 0.23 2.404"
## [43] "0.2325 0.62 4.173"
## [44] "0.2000 0.86 6.112"
## [45] "0.2000 1.00 7.647"
## [46] "0.2000 1.00 9.770"
## [47] "# cv Biological data"
```

```
## [48] "# cvM, cvmat, cvwSSB"  
## [49] "0.1 0.0 0.112"  
## [50] "0.1 0.1 0.143"  
## [51] "0.1 0.1 0.137"  
## [52] "0.1 0.1 0.097"  
## [53] "0.1 0.0 0.062"  
## [54] "0.1 0.0 0.050"  
## [55] "0.1 0.0 0.040"
```