

Formato Archivo Control.ss

abril, 27, 2023

Contents

0.1	Identificamos los directorio de trabajo	2
0.2	Leer los archivos de Stock Synthesis con la función <code>SS_read()</code>	3
0.3	Revisamos los nombres de los componentes de la lista del archivo control que deseamos modificar	3
0.4	Especificaciones iniciales	4
0.5	Especificaciones del crecimiento	5
0.6	Distribución del reclutamiento	5
0.7	Bloques	6
0.8	Parámetros que varían en el tiempo	7
0.9	Parámetros biológicos	7
0.10	Relación stock recluta	17
0.11	Desvíos de los reclutamientos y sesgo	19
0.12	Mortalidad por pesca	21
0.13	Capturabilidad	21
0.14	Selectividad	23

Los archivos requerido para modelos simple los puedes descargar del siguiente link [Repositorio modelo simple](#)
 Los archivos modificados para el boqueron los puedes descargar del siguiente link [Repositorio modelo boqueron](#)

0.1 Identificamos los directorio de trabajo

```
#dirname.base <- here("10a_anchcadiz")
dirname.base <- here("modelos_SS3", "simple")

dirsardina <- here("assess_ref", "sardine.dat")

dat_Sard<- SS_readdat(here("assess_ref", "sardine.dat"), verbose= FALSE, version = 3.24)

ctl_Sard <- readLines(here("assess_ref", "sardine.ctl"))

ctl_Sard <- r4ss::SS_readctl(
  file = here("assess_ref", "sardine.ctl"),
  version="3.24",
  verbose = FALSE,
  datlist = dat_Sard, use_datlist = TRUE)

names(ctl_Sard)
## [1] "warnings"           "nseas"
## [3] "N_areas"            "Nages"
## [5] "Nsexes"             "Npopbins"
## [7] "Nfleet"             "Nsurveys"
## [9] "Do_AgeKey"          "N_CPUE_obs"
## [11] "fleetnames"         "sourcefile"
## [13] "type"               "ReadVersion"
## [15] "eof"                "N_GP"
## [17] "N_platoon"          "sd_ratio"
## [19] "submorphdist"       "recr_dist_inx"
## [21] "recr_dist_method"   "N_Block_Designs"
## [23] "blocks_per_pattern" "Block_Design"
## [25] "time_vary_adjust_method" "time_vary_auto_generation"
## [27] "fracfemale"         "natM_type"
## [29] "natM"               "GrowthModel"
## [31] "Growth_Age_for_L1"  "Growth_Age_for_L2"
## [33] "Exp_Decay"          "N_natMparms"
## [35] "SD_add_to_LAA"      "CV_Growth_Pattern"
## [37] "maturity_option"    "EmpiricalWAA"
## [39] "First_Mature_Age"   "fecundity_option"
## [41] "hermaphroditism_option" "parameter_offset_approach"
## [43] "env_block_dev_adjust_method" "MG_parms"
## [45] "MGparm_seas_effects" "SR_function"
## [47] "SR_parms"           "SR_env_link"
## [49] "SR_env_target"      "do_recdev"
## [51] "MainRdevYrFirst"    "MainRdevYrLast"
## [53] "recdev_phase"       "recdev_adv"
## [55] "recdev_early_start" "recdev_early_phase"
## [57] "Fcast_recr_phase"   "lambda4Fcast_recr_like"
## [59] "last_early_yr_nobias_adj" "first_yr_fullbias_adj"
## [61] "last_yr_fullbias_adj" "first_recent_yr_nobias_adj"
```

```
## [63] "max_bias_adj"          "period_of_cycles_in_recr"
## [65] "min_rec_dev"           "max_rec_dev"
## [67] "N_Read_recdevs"        "F_ballpark"
## [69] "F_ballpark_year"       "F_Method"
## [71] "maxF"                  "F_iter"
## [73] "init_F"                "Q_setup"
## [75] "Q_parms"               "size_selex_types"
## [77] "age_selex_types"        "age_selex_parms"
## [79] "DoCustom_sel_blk_setup" "custom_sel_blk_setup"
## [81] "DoAdjust"              "selex_adjust_method"
## [83] "TG_custom"             "DoVar_adjust"
## [85] "Variance_adjustments"  "maxlambdaphase"
## [87] "sd_offset"             "N_lambdas"
## [89] "lambdas"               "more_stddev_reporting"
## [91] "stddev_reporting_specs" "stddev_reporting_selex"
## [93] "stddev_reporting_growth" "stddev_reporting_N_at_A"
```

0.1.0.1 Creamos un nuevo directorio para la nueva versión del modelo modificado

```
dirname.mod <- here("boqueron_SS3")
dir.create(path=dirname.mod, showWarnings = TRUE, recursive = TRUE)
```

0.2 Leer los archivos de Stock Synthesis con la función SS_read()

Cada uno de los archivos de entrada se lee en R como una lista.

Use `names()` para ver todos los componentes de la lista

```
inputs <- r4ss::SS_read(dir = dirname.base)
names(inputs)
## [1] "dir"      "path"     "dat"      "ctl"      "start"    "fore"     "wtatage"

#trabajaremos con archivo ctl
ctl<-inputs$ctl # archivo base
ctl1<-ctl # archivo modificado
```

0.3 Revisamos los nombres de los componentes de la lista del archivo control que deseamos modificar

El archivo `control.ss` contiene las especificaciones de los valores de los parámetros del modelo (tanto valores fijos como valores estimados).

```
#matrix(names(ctl1),ncol=1)
```

0.4 Especificaciones iniciales

```
#-----
ctl1$nseas      <- 4
ctl1$N_areas   <- 1
ctl1$Nages     <- 4
ctl1$Nsexes    <- 1
ctl1$Npopbins  <- 44
ctl1$Nfleets   <- 3
ctl1$Do_AgeKey <- 0
ctl1$fleetnames <- c("Fishery","PELAGO","ECOCADIZ")
ctl1$Comments<-"#C 2023 Boqueron Cádiz control file MODELO TRIMESTRAL"
ctl1$eof       <- TRUE
#-----

# revisa salidas
ctl1[1:14]
## $warnings
## [1] ""
##
## $Comments
## [1] "#C 2023 Boqueron Cádiz control file MODELO TRIMESTRAL"
##
## $nseas
## [1] 4
##
## $N_areas
## [1] 1
##
## $Nages
## [1] 4
##
## $Nsexes
## [1] 1
##
## $Npopbins
## [1] 44
##
## $Nfleets
## [1] 3
##
## $Do_AgeKey
## [1] 0
##
## $fleetnames
## [1] "Fishery" "PELAGO" "ECOCADIZ"
##
## $sourcefile
## [1] "C:/SS3_boqueron/SS3_ane27.9a/modelos_SS3/simple/control.ss"
##
## $type
## [1] "Stock_Synthesis_control_file"
##
## $ReadVersion
## [1] "3.30"
```

```
##
## $eof
## [1] TRUE
#-----
```

0.5 Especificaciones del crecimiento

Consulte la Guía de usuario de SS3: [8.4.1 Weight-at-Age](#).

En este caso, no se usa, por lo que se establece en 0.

Si se usara el peso empírico por edad, SS3 ignoraría todas las entradas relacionadas con el crecimiento, la madurez y la fecundidad que se especifican más adelante en el archivo de `control.ss` (aunque todavía espera entradas).

Luego están las opciones para el número de patrones de crecimiento y platoons. Estos se establecen en 1 porque asumimos que toda la población tiene el mismo patrón de crecimiento y no hay platoons dentro de los patrones de crecimiento.

Opciones y especificaciones:

- `EmpiricalWAA = 0` means do not read `wtatage.ss`; 1 means read and use `wtatage.ss` and also read and use growth parameters
- `N_GP = N_Growth_Patterns` (Growth Patterns, Morphs, Bio Patterns, GP are terms used interchangeably in SS3)
- `N_platoon = N_platoons_Within_GrowthPattern`

```
#-----
ctl1$EmpiricalWAA <- 0
ctl1$N_GP          <- 1
ctl1$N_platoon     <- 1
#-----
# revisa salidas
ctl1[15:17]
## $EmpiricalWAA
## [1] 0
##
## $N_GP
## [1] 1
##
## $N_platoon
## [1] 1
#-----
```

0.6 Distribución del reclutamiento

Consulte la Guía de usuario de SS3: [8.4.2 Settlement Timing for Recruits and Distribution](#).

Luego se especifica la distribución del reclutamiento. Suponemos que para esta población solo hay 1 patrón de crecimiento, 1 tiempo de asentamiento (settlement) y 1 área, por lo que podemos usar la opción 4. Esta es la más simple, porque no requiere líneas completas de parámetros más adelante en el archivo `control.ss`.

Tenga en cuenta que también usamos 1 asignación de asentamiento (settlement) y especificamos que los peces se establecieron en enero (1) a la edad 0 (año calendario). Esto podría modificarse si los peces se establecieron a una edad o momento diferente.

Opciones y especificaciones:

- `recr_dist_method = 2`=main effects for GP, Area, Settle timing; `3`=each Settle entity; `4`=none (only when `N_GP$settlepop==1`)
- `recr_global_area = 1`=global; `2`=by area
- `recr_dist_read` = number of recruitment settlement assignments
- `recr_dist_inx` = unused option

`recr_dist_pattern` es una Matriz que contiene la siguiente información:

- GPattern:
- month:
- area:
- age:

(for each settlement assignment)

```
#-----
ctl1$recr_dist_method<-3
ctl1$recr_global_area<-1
ctl1$recr_dist_read  <-1
ctl1$recr_dist_inx   <-0
#-----
      rec_pattern<-data.frame(row.names="recr_dist_pattern1",
                             "GPattern" = 1,
                             "month"    = 1,
                             "area"     = 1,
                             "age"      = 0)
ctl1$recr_dist_pattern<-rec_pattern
#-----
```

0.7 Bloques

No estamos usando bloques de tiempo, pero hay algunos especificados en el modelo. Tenga en cuenta que cualquier bloque puede especificarse y no usarse (los bloques se especifican aquí, pero no se usan hasta que se especifica dentro de una línea de parámetro), por lo que podría haber cualquier cantidad de bloques especificados aquí.

[Consulte la Guía de usuario de SS3: 8.4.4 Time Blocks.](#)

Opciones y especificaciones:

- `N_Block_Designs` = Number of block patterns. These patterns can be referred to in the parameter sections to create a separate parameter value for each block. Following inputs are omitted if the number of block patterns equals 0.
- `blocks_per_pattern` =
- `Block_Design` =

```
#-----
ctl1$N_Block_Designs  <- 1
ctl1$blocks_per_pattern <- 1
ctl1$Block_Design     <- c(1989,1989)
#-----
# revisa salidas
ctl1[23:25]
## $N_Block_Designs
## [1] 1
##
## $blocks_per_pattern
## [1] 1
```

```
##
## $Block_Design
## [1] 1989 1989
#-----
```

0.8 Parámetros que varían en el tiempo

Así mismo, no se utilizan parámetros variables en el tiempo, por lo que no es importante lo que se especifica en los parámetros de control variables en el tiempo.

Consulte la Guía de usuario de SS3: [8.4.5 Auto-generation](#).

Opciones y especificaciones:

```
#-----
ctl1$time_vary_adjust_method <- 1
ctl1$time_vary_adjust_method
## [1] 1
#-----
# arreglo de datos para "time_vary_auto_generation"
time_auto<-data.frame(matrix(rep(1,5),nrow=1,ncol=5))
colnames(time_auto)<-paste("time_vary_auto_generation_",seq(1,5,1),sep="")

ctl1$time_vary_auto_generation<-time_auto
ctl1$time_vary_auto_generation
##   time_vary_auto_generation_1 time_vary_auto_generation_2
## 1                          1                          1
##   time_vary_auto_generation_3 time_vary_auto_generation_4
## 1                          1                          1
##   time_vary_auto_generation_5
## 1                          1
#-----
```

0.9 Parámetros biológicos

Consulte la Guía de usuario de SS3: [8.5 Biology](#).

Las líneas de parámetros resultantes de la mortalidad natural, el crecimiento y la madurez (esta sección a veces se denomina parámetros MG) se especifican a continuación. El número de líneas de parámetros depende de las opciones seleccionadas en las líneas de especificación. Los parámetros también deben especificarse en un orden particular, con los parámetros de las hembras antes que los de machos en un modelo de 2 sexos.

Tenga en cuenta que la primera línea en el bloque de entrada SS anterior muestra los encabezados de columna. Todas las secciones con líneas de parámetros largas dentro del archivo de control tienen estos mismo encabezados. Hay muchas especificaciones en estas largas líneas de parámetros, pero algunas de particular importancia son:

- Todo lo que tenga fase negativa (séptimo valor en una línea larga de parámetros) no se estima y se establece en el valor inicial (tercer valor en la línea), mientras que las fases positivas sí se estiman.
- La mortalidad natural tanto para machos como para hembras se especifican en 0,1.
- Se estiman los tres parámetros de la curva de crecimiento de von Bertalanffy.

A continuación se presentan algunas opciones no utilizadas, porque no hay estacionalidad en los parámetros biológicos:

Opciones y especificaciones:

- **natM_type:** 0=1Parm;1=N_breakpoints;2=Lorenzen;3=agespecific;4=agespec_withseasinterpolate;8.5.1 [Natural Mortality](#). 5=BETA: Maunders link to maturity;6=Lorenzen_rangen additional input for selected M option; read 1P per morph
- **GrowthModel:** 1= von Bert with L1&L2; 2=Richards with L1&L2; 3=age_specific_K_incr; 4=age_specific_K_decr; 5=age_specific_K_each; 6=NA; 7=NA; 8=growth cessation
- **Growth_Age_for_L1:** Age(post-settlement)_for_L1; linear growth below this
- **Growth_Age_for_L2:** 999 to use as Linf
- **Exp_Decay:** exponential decay for growth above maxage (value should approx initial Z; -999 replicates 3.24; -998 to not allow growth above maxage)
- **Growth_Placeholder:** placeholder for future growth feature
- **N_natMparms:**
- **SD_add_to_LAA:** SD_add_to_LAA (set to 0.1 for SS2 V1.x compatibility)
- **CV_Growth_Pattern:** 0=(CV=f(LAA)); 1=(CV=F(A)); 2=(SD=F(LAA)); 3=(SD=F(A)); 4=(logSD=F(A))
- **maturity_option:** 1=length logistic; 2=age logistic; 3=read age-maturity matrix by growth_pattern; 4=read age-fecundity; 5=disabled; 6=read length-maturity
- **First_Mature_Age:**
- **fecundity_option:** (1)eggs=Wt(a+bWt); (2)eggs=aL^b; (3)eggs=aWt^b; (4)eggs=a+bL; (5)eggs=a+bW
- **hermaphroditism_option:** 0=none; 1=female-to-male age-specific fxn; -1=male-to-female age-specific fxn
- **parameter_offset_approach:** 1- direct, no offset**; 2- male=fem_parmexp(male_parm); 3: male=femaleexp(parm) then old=young*exp(parm)

El siguiente código permite modificar las opciones antes especificadas:

```
ctl1$natM_type      <- 3
#incluір vector de mortalidad natural-----
# leer vector de M natural desde archivos de GADGET
medad0 <- list()
medad1<-data.frame(matrix(c(2.21,1.3,1.3,1.3),nrow=1,ncol=4),row.names="#_natM1")
colnames(medad1)<-paste("Age_",seq(0,3,1),sep="")

medad0[[1]] <- medad1
names(medad0)<-"natM"
ctl1 <- append(ctl1, medad0, after = 28)
#-----

ctl1$GrowthModel      <- 1
ctl1$Growth_Age_for_L1 <- 0.1
ctl1$Growth_Age_for_L2 <- 4
ctl1$Exp_Decay        <- -999
ctl1$Growth_Placeholder <- 0
ctl1$N_natMparms      <- 1
ctl1$SD_add_to_LAA    <- 0
ctl1$CV_Growth_Pattern <- 0
ctl1$maturity_option  <- 1
ctl1$First_Mature_Age <- 1
ctl1$fecundity_option  <- 1
ctl1$hermaphroditism_option <- 0
ctl1$parameter_offset_approach <- 1

#-----
#revisa salidas
ctl1[28:41]
```



```
## $natM_type
## [1] 3
##
## $natM
##      Age_0 Age_1 Age_2 Age_3
## #_natM1 2.21  1.3  1.3  1.3
##
## $GrowthModel
## [1] 1
##
## $Growth_Age_for_L1
## [1] 0.1
##
## $Growth_Age_for_L2
## [1] 4
##
## $Exp_Decay
## [1] -999
##
## $Growth_Placeholder
## [1] 0
##
## $N_natMparms
## [1] 1
##
## $SD_add_to_LAA
## [1] 0
##
## $CV_Growth_Pattern
## [1] 0
##
## $maturity_option
## [1] 1
##
## $First_Mature_Age
## [1] 1
##
## $fecundity_option
## [1] 1
##
## $hermaphroditism_option
## [1] 0
##-----
```

La función principal del archivo de control es definir los parámetros que utilizará el modelo. A continuación se indican las líneas de parámetros biológicos que serán modificadas:

```
row.names(ctl$MG_parms)
## [1] "NatM_p_1_Fem_GP_1"      "L_at_Amin_Fem_GP_1"
## [3] "L_at_Amax_Fem_GP_1"     "VonBert_K_Fem_GP_1"
## [5] "CV_young_Fem_GP_1"      "CV_old_Fem_GP_1"
## [7] "Wtlen_1_Fem_GP_1"       "Wtlen_2_Fem_GP_1"
## [9] "Mat50%_Fem_GP_1"        "Mat_slope_Fem_GP_1"
## [11] "Eggs/kg_inter_Fem_GP_1" "Eggs/kg_slope_wt_Fem_GP_1"
```

```
## [13] "NatM_p_1_Mal_GP_1"      "L_at_Amin_Mal_GP_1"
## [15] "L_at_Amax_Mal_GP_1"      "VonBert_K_Mal_GP_1"
## [17] "CV_young_Mal_GP_1"       "CV_old_Mal_GP_1"
## [19] "Wtlen_1_Mal_GP_1"        "Wtlen_2_Mal_GP_1"
## [21] "RecrDist_GP_1"           "RecrDist_Area_1"
## [23] "RecrDist_month_1"        "CohortGrowDev"
## [25] "FracFemale_GP_1"
```

La definición de cada línea es la siguiente:

- `NatM_p_1_Fem_GP_1` = Mortalidad natural para el patrón de crecimiento de hembras (Fem) 1, donde el número de parámetros de mortalidad natural depende de la opción seleccionada.
- `L_at_Amin_Fem_GP_1` = Longitud en Amin (unidades en cm) para hembra, patrón de crecimiento 1.
- `L_at_Amax_Fem_GP_1` = Longitud en Amax (unidades en cm) para hembra, patrón de crecimiento 1.
- `VonBert_K_Fem_GP_1` = Coeficiente de crecimiento de von Bertalanffy (las unidades son por año) para las hembras, patrón de crecimiento 1.
- `CV_young_Fem_GP_1` = variabilidad para el tamaño a la edad \leq Amin para las hembras, patrón de crecimiento 1. Tenga en cuenta que CV no puede variar con el tiempo, por lo que no figure env-link o un vector de desviación. Además, las unidades son como CV o como desviación estándar, dependiendo del valor asignado del patrón CV.
- `CV_old_Fem_GP_1` = variabilidad para el tamaño a la edad \geq Amax para las hembras, patrón de crecimiento 1. Para edades intermedias, haga una interpolación lineal de CV en el tamaño medio a la edad. Tenga en cuenta que las unidades para CV dependerán del patrón CV y del valor del parámetro mortalidad-crecimiento como compensación. El valor del CV no puede variar con el tiempo.
- `Wtlen_1_Fem_GP_1` = coeficiente para convertir la longitud en cm en peso en kg para las hembras.
- `Wtlen_2_Fem_GP_1` = exponente en convertir la longitud a peso para hembras.
- `Mat50%_Fem_GP_1` = inflexión logística de madurez (en cm o años) donde la madurez femenina en logitud (o edad) es una función logística.
- `Mat_slope_Fem_GP_1` = pendiente logística (debe tener valor negativo).
- `Eggs_alpha_Fem_GP_1` = parámetros de fecundidad. El uso depende de la opción de fecundidad seleccionada.
- `Eggs_beta_Fem_GP_1`

Cada línea de parámetro biológicos contiene la siguiente información:

```
names(ct1$MG_parms)
## [1] "LO"      "HI"      "INIT"     "PRIOR"    "PR_SD"
## [6] "PR_type" "PHASE"   "env_var&link" "dev_link" "dev_minyr"
## [11] "dev_maxyr" "dev_PH"  "Block"    "Block_Fxn" "PType"
```

La definición de cada línea es la siguiente:

- `LO` = un valor mínimo para el parámetro
- `HI` = un valor máximo para el parámetro
- `INIT` = valor inicial para el parámetro. Si la fase (descrita a continuación) para el parámetro es negativa, el parámetro se fija en este valor. Si se lee el archivo ss.par, sobrescribe estos valores INIT.
- `PRIOR` = valor esperado para el parámetro. Este valor se ignora si el `PR_type` es 0 (no prior) o 1 (symmetric beta). Si `PR_type` es lognormal (descrito a continuación).
- `PR_SD` = desviación standar de la `PRIOR`, utilizado para calcular la likelihood del valor del parámetro actual. Este valor es ignorado si la `PR_type` es 0.
- `PR_type` = tipo de distribución de error del valor esperado:

0 = ninguno

```

1 = symmetric beta
2 = full beta
3 = lognormal sin ajuste de sesgo
4 = lognormal con ajuste de sesgo
5 = gamma, y
6 = normal.

```

- PHASE = fase en la que se empieza a estimar el parámetro. Un valor negativo hace que el parámetro conserve su valor INIT (o valor leído del archivo ss.par)
- env_var&link = crea un vínculo a una serie temporal ingresada al archivo de datos
- dev_link = invoca el uso del vector de desviación en la función linkage
- dev_minyr= año de inicio del vector de desviación
- dev_maxyr= año final para el vector de desviación
- dev_PH = fase de estimación para elementos en el vector de desviación
- Block = bloque de tiempo o tendencia a aplicar
- Block_Fxn= forma funcional para el desplazamiento de bloques

0.9.1 Mortalidad natural

La opción 0 se usa para la mortalidad natural porque solo se asume 1 valor.

0.9.2 Crecimiento

[Consulte la Guía de usuario de SS3: 8.5.2 Growth.](#)

El modelo de crecimiento 1 se usa para especificar un modelo de crecimiento de von Bertalanffy, seguido de líneas que especifican detalles sobre el crecimiento.

0.9.3 Madurez-fecundidad

[Consulte la Guía de usuario de SS3: 8.5.3 Maturity-Fecundity.](#)

0.9.4 Hermafroditismo

[Consulte la Guía de usuario de SS3: 8.5.4 Hermaphroditism.](#)

```

##-----
MG_parms1<-data.frame("LO"          = 0.05,
                      "HI"          = 1.6,
                      "INIT"        = 0.7,
                      "PRIOR"       = -1.60944,
                      "PR_SD"       = 0.1,
                      "PR_type"     = 0,
                      "PHASE"       = -4,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms["NatM_p_1_Fem_GP_1",] <- "#"#MG_parms1 # dejar en null si se cambia a opción 3

```

```

##-----
MG_parms2<-data.frame("LO"          = 0,
                      "HI"          = 10,
                      "INIT"        = 6,
                      "PRIOR"       = 32,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = 5,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms["L_at_Amin_Fem_GP_1",]<-MG_parms2
##-----
MG_parms3<-data.frame("LO"          = 5,
                      "HI"          = 22,
                      "INIT"        = 18.95,
                      "PRIOR"       = 19,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = 5,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms["L_at_Amax_Fem_GP_1",]<-MG_parms3
##-----
MG_parms4<-data.frame("LO"          = 0.1,
                      "HI"          = 2.0,
                      "INIT"        = 0.89,
                      "PRIOR"       = 0.9,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = 3,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms["VonBert_K_Fem_GP_1",]<-MG_parms4
##-----
MG_parms5<-data.frame("LO"          = 0.03,
                      "HI"          = 0.150,
                      "INIT"        = 0.066,
                      "PRIOR"       = 0.1,

```

```

        "PR_SD"          = 99,
        "PR_type"        = 0,
        "PHASE"          = 5,
        "env_var&link"    = 0,
        "dev_link"       = 0,
        "dev_minyr"      = 0,
        "dev_maxyr"      = 0,
        "dev_PH"         = 0,
        "Block"          = 0,
        "Block_Fxn"      = 0)
ctl1$MG_parms["CV_young_Fem_GP_1",]<-MG_parms5
##-----
MG_parms6<-data.frame("LO"          = 0.03,
                      "HI"          = 0.150,
                      "INIT"        = 0.066,
                      "PRIOR"       = 0.1,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = 5,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms["CV_old_Fem_GP_1",]<-MG_parms6
##-----
MG_parms7<-data.frame("LO"          = -3.0,
                      "HI"          = 3.0,
                      "INIT"        = 0.00312895,
                      "PRIOR"       = 0.00312895,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = -50,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms["Wtlen_1_Fem_GP_1",]<-MG_parms7
##-----
MG_parms8<-data.frame(row.names="Wtlen_2_Fem_GP_1",
                      "LO"          = -3.0,
                      "HI"          = 3.0,
                      "INIT"        = 3.278,
                      "PRIOR"       = 3.278,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = -50,
                      "env_var&link" = 0,

```

```

        "dev_link"      = 0,
        "dev_minyr"     = 0,
        "dev_maxyr"     = 0,
        "dev_PH"        = 0,
        "Block"         = 0,
        "Block_Fxn"     = 0)
ctl1$MG_parms[8,]<-MG_parms8
##-----
MG_parms10<-data.frame("LO"          = -3.0,
        "HI"              = 3.0,
        "INIT"            = -0.45,
        "PRIOR"           = -0.45,
        "PR_SD"           = 99,
        "PR_type"         = 0,
        "PHASE"           = -50,
        "env_var&link"    = 0,
        "dev_link"        = 0,
        "dev_minyr"       = 0,
        "dev_maxyr"       = 0,
        "dev_PH"          = 0,
        "Block"           = 0,
        "Block_Fxn"       = 0)
ctl1$MG_parms["Mat_slope_Fem_GP_1",]<-MG_parms10
##-----
MG_parms11<-data.frame("LO"          = -3.0,
        "HI"              = 3.0,
        "INIT"            = 1,
        "PRIOR"           = 1,
        "PR_SD"           = 99,
        "PR_type"         = 0,
        "PHASE"           = -50,
        "env_var&link"    = 0,
        "dev_link"        = 0,
        "dev_minyr"       = 0,
        "dev_maxyr"       = 0,
        "dev_PH"          = 0,
        "Block"           = 0,
        "Block_Fxn"       = 0)
ctl1$MG_parms["Eggs/kg_inter_Fem_GP_1",]<-MG_parms11
##-----
MG_parms12<-data.frame("LO"          = -3.0,
        "HI"              = 3.0,
        "INIT"            = 1,
        "PRIOR"           = 1,
        "PR_SD"           = 99,
        "PR_type"         = 0,
        "PHASE"           = -50,
        "env_var&link"    = 0,
        "dev_link"        = 0,
        "dev_minyr"       = 0,
        "dev_maxyr"       = 0,
        "dev_PH"          = 0,
        "Block"           = 0,

```

```

      "Block_Fxn"      = 0)
ctl1$MG_parms["Eggs/kg_slope_wt_Fem_GP_1",]<-MG_parms12
##-----
MG_parms13<-data.frame("LO"          = 0,
      "HI"            = 10,
      "INIT"          = 1,
      "PRIOR"         = 1,
      "PR_SD"         = 99,
      "PR_type"       = 0,
      "PHASE"         = -3,
      "env_var&link"  = 0,
      "dev_link"      = 0,
      "dev_minyr"     = 0,
      "dev_maxyr"     = 0,
      "dev_PH"        = 0,
      "Block"         = 0,
      "Block_Fxn"     = 0)
ctl1$MG_parms["RecrDist_GP_1_area_1_month_1",]<-MG_parms13
##-----
MG_parms14<-data.frame("LO"          = 1,
      "HI"            = 1,
      "INIT"          = 1,
      "PRIOR"         = 1,
      "PR_SD"         = 99,
      "PR_type"       = 0,
      "PHASE"         = -1,
      "env_var&link"  = 0,
      "dev_link"      = 0,
      "dev_minyr"     = 0,
      "dev_maxyr"     = 0,
      "dev_PH"        = 0,
      "Block"         = 0,
      "Block_Fxn"     = 0)
ctl1$MG_parms["CohortGrowDev",]<-MG_parms14
##-----
MG_parms15<-data.frame("LO"          = 0.000001,
      "HI"            = 0.999999,
      "INIT"          = 0.5,
      "PRIOR"         = 0.5,
      "PR_SD"         = 0.5,
      "PR_type"       = 0,
      "PHASE"         = -99,
      "env_var&link"  = 0,
      "dev_link"      = 0,
      "dev_minyr"     = 0,
      "dev_maxyr"     = 0,
      "dev_PH"        = 0,
      "Block"         = 0,
      "Block_Fxn"     = 0)
ctl1$MG_parms["FracFemale_GP_1",]<-MG_parms15
##-----
#elimino los parámetros que no se utilizaran en el modelo trimestral por el momento
patron_eliminar<-rownames(ctl1$MG_parms)[grep("_Mal_",rownames(ctl1$MG_parms))]

```

```

ctl1$MG_parms <- subset(ctl1$MG_parms, !rownames(ctl1$MG_parms) %in%
                        c(patron_eliminar,"RecrDist_GP_1","RecrDist_Area_1"))

# revisa salidas primeras columnas

ctl1$MG_parms[,1:6]
##              LO              HI              INIT              PRIOR PR_SD PR_type
## NatM_p_1_Fem_GP_1          #          #          #          #          #          #
## L_at_Amin_Fem_GP_1          0          10          6          32          99          0
## L_at_Amax_Fem_GP_1          5          22         18.95          19          99          0
## VonBert_K_Fem_GP_1         0.1          2          0.89          0.9          99          0
## CV_young_Fem_GP_1          0.03         0.15         0.066          0.1          99          0
## CV_old_Fem_GP_1            0.03         0.15         0.066          0.1          99          0
## Wtlen_1_Fem_GP_1           -3          3 0.00312895 0.00312895          99          0
## Wtlen_2_Fem_GP_1           -3          3          3.278          3.278          99          0
## Mat50%_Fem_GP_1            50          60          55          55          0.8          0
## Mat_slope_Fem_GP_1         -3          3          -0.45          -0.45          99          0
## Eggs/kg_inter_Fem_GP_1     -3          3          1          1          99          0
## Eggs/kg_slope_wt_Fem_GP_1  -3          3          1          1          99          0
## RecrDist_month_1           0          0          0          0          0          0
## CohortGrowDev              1          1          1          1          99          0
## FracFemale_GP_1            1e-06 0.999999          0.5          0.5          0.5          0
## RecrDist_GP_1_area_1_month_1 0          10          1          1          99          0
##-----

```

0.9.4.0.1 Seasonal_effects_on_biology_parms

Los efectos estacionales están disponibles para los parámetros de peso-longitud, madurez, fecundidad y para el parámetro de crecimiento K. Los valores de los parámetros estacionales ajustan el valor del parámetro base para esa temporada.

MGparm_seas_effects: Estacionalidad para parámetros biológicos seleccionados (no una entrada condicional). Lea 10 enteros para especificar qué parámetros biológicos tienen estacionalidad: female-wtlen1, female-wtlen2, maturity1, maturity2, fecundity1, fecundity2, male-wtlen1, male-wtlen2, L1, K. La lectura de un valor positivo selecciona ese factor para la estacionalidad.

Consulte la Guía de usuario de SS3: 8.5.12 Seasonal Biology Parameters.

```

##-----
MGparm_seas_effects1<-data.frame(matrix(rep(0,10),nrow=1,ncol=10))
colnames(MGparm_seas_effects1)<-paste("MGparm_seas_effects_",seq(1,10,1),sep="")

ctl1$MGparm_seas_effects<-MGparm_seas_effects1
##-----
# para revisar
ctl1$MGparm_seas_effects
##   MGparm_seas_effects_1 MGparm_seas_effects_2 MGparm_seas_effects_3
## 1                      0                      0                      0
##   MGparm_seas_effects_4 MGparm_seas_effects_5 MGparm_seas_effects_6
## 1                      0                      0                      0
##   MGparm_seas_effects_7 MGparm_seas_effects_8 MGparm_seas_effects_9
## 1                      0                      0                      0
##   MGparm_seas_effects_10
## 1                      0
##-----

```


0.10 Relación stock recluta

Consulte la Guía de usuario de SS3: 8.6 Spawner-Recruitment.

- SR_function: 1=NA;2=Ricker;3=std_B-H;4=SCAA;5=Hockey; 6=B-H_flattop; 7=survival_3Parm; 8=Shepherd_3Parm; 9=RickerPower_3parm
- Use_steep_init_equi: 0/1 to use steepness in initial equ recruitment calculation
- ctl1\$Sigma_R_FofCurvature: future feature: 0/1 to make realized sigmaR a function of SR curvature
- Sigma_R_FofCurvature: future feature: 0/1 to make realized sigmaR a function of SR curvature

```
# Especificaciones iniciales
##-----
ctl1$SR_function      <- 4
ctl1$Use_steep_init_equi <- 0
ctl1$Sigma_R_FofCurvature <- 0
##-----

# revisar salidas
ctl1[44:46]
## $MGparm_seas_effects
##   MGparm_seas_effects_1 MGparm_seas_effects_2 MGparm_seas_effects_3
## 1                      0                      0                      0
##   MGparm_seas_effects_4 MGparm_seas_effects_5 MGparm_seas_effects_6
## 1                      0                      0                      0
##   MGparm_seas_effects_7 MGparm_seas_effects_8 MGparm_seas_effects_9
## 1                      0                      0                      0
##   MGparm_seas_effects_10
## 1                      0
##
## $SR_function
## [1] 4
##
## $Use_steep_init_equi
## [1] 0

# Sección donde se ingresan los parámetros
##-----
SR_parms1<-data.frame("LO"          = 5,
                      "HI"          = 20,
                      "INIT"        = 13,
                      "PRIOR"        = 0,
                      "PR_SD"        = 0,
                      "PR_type"      = 0,
                      "PHASE"        = 1,
                      "env_var&link" = 0,
                      "dev_link"     = 0,
                      "dev_minyr"    = 0,
                      "dev_maxyr"    = 0,
                      "dev_PH"       = 0,
                      "Block"        = 0,
                      "Block_Fxn"    = 0)
ctl1$SR_parms[1,]<-SR_parms1
rownames(ctl1$SR_parms)[1]<-"SR_LN(R0)"
##-----
SR_parms2<-data.frame("LO"          = 0.2,
                      "HI"          = 1,
```

```

        "INIT"          = 0.88,
        "PRIOR"         = 0.777,
        "PR_SD"         = 0.113,
        "PR_type"       = 2,
        "PHASE"         = -4,
        "env_var&link"  = 0,
        "dev_link"      = 0,
        "dev_minyr"     = 0,
        "dev_maxyr"     = 0,
        "dev_PH"        = 0,
        "Block"         = 0,
        "Block_Fxn"     = 0)
ctl1$SR_parms[2,] <-SR_parms2;
rownames(ctl1$SR_parms)[2]<-"SR_SCAA_null"
##-----
SR_parms3<-data.frame("LO"          = 0.3,
                      "HI"          = 1.6,
                      "INIT"        = 0.6,
                      "PRIOR"        = 1.1,
                      "PR_SD"        = 99,
                      "PR_type"      = 0,
                      "PHASE"        = -6,
                      "env_var&link" = 0,
                      "dev_link"     = 0,
                      "dev_minyr"    = 0,
                      "dev_maxyr"    = 0,
                      "dev_PH"       = 0,
                      "Block"        = 0,
                      "Block_Fxn"    = 0)
ctl1$SR_parms[3,] <-SR_parms3
rownames(ctl1$SR_parms)[3]<-"SR_sigmaR"
##-----
SR_parms4<-data.frame("LO"          = -5,
                      "HI"          = 5,
                      "INIT"        = 0,
                      "PRIOR"        = 0,
                      "PR_SD"        = 99,
                      "PR_type"      = 0,
                      "PHASE"        = -50,
                      "env_var&link" = 0,
                      "dev_link"     = 0,
                      "dev_minyr"    = 0,
                      "dev_maxyr"    = 0,
                      "dev_PH"       = 0,
                      "Block"        = 0,
                      "Block_Fxn"    = 0)
ctl1$SR_parms[4,] <-SR_parms4
rownames(ctl1$SR_parms)[4]<-"SR_regime"
##-----
SR_parms5<-data.frame("LO"          = 0,
                      "HI"          = 2,
                      "INIT"        = 0,

```

```

        "PRIOR"          = 1,
        "PR_SD"          = 99,
        "PR_type"        = 0,
        "PHASE"          = -50,
        "env_var&link"    = 0,
        "dev_link"        = 0,
        "dev_minyr"       = 0,
        "dev_maxyr"       = 0,
        "dev_PH"          = 0,
        "Block"           = 0,
        "Block_Fxn"       = 0)
ctl1$SR_parms[5,] <- SR_parms5
rownames(ctl1$SR_parms)[5]<-"SR_autocorr"
##-----
# para revisar primeras columnas
ctl1$SR_parms[,1:6]
##          LO    HI   INIT PRIOR  PR_SD PR_type
## SR_LN(R0)    5.0 20.0 13.00 0.000  0.000      0
## SR_SCAA_null 0.2  1.0  0.88 0.777  0.113      2
## SR_sigmaR    0.3  1.6  0.60 1.100 99.000      0
## SR_regime    -5.0  5.0  0.00 0.000 99.000      0
## SR_autocorr   0.0  2.0  0.00 1.000 99.000      0
##-----

```

0.11 Desvíos de los reclutamientos y sesgo

Consulte la Guía de usuario de SS3: [8.6.5 Recruitment Deviation Setup](#).

Las opciones de desviación de reclutamiento se especifican después de los parámetros de SR. Primero están las opciones de desviaciones estándar de reclutamiento:

- **do_recdev:** 0=none; 1=devvector ($R=F(SSB)+dev$); 2=deviations ($R=F(SSB)+dev$); 3=deviations ($R=R0*dev$; $dev2=R-f(SSB)$); 4=like 3 with sum(dev2) adding penalty
- **MainRdevYrFirst:** first year of main recr_devs; early devs can precede this era
- **MainRdevYrLast:** last year of main recr_devs; forecast devs start in following year
- **recdev_phase:**

Estos definen las principales desviaciones de reclutamientos, que en este caso duran desde el primer año del modelo hasta el último año. También se leen opciones avanzadas:

- **recdev_adv:** (0/1) to read 13 advanced options
- **recdev_early_start:** (0=none; neg value makes relative to recdev_start)
- **recdev_early_phase:**
- **Fcast_recrr_phase:** (incl. late recr) (0 value resets to maxphase+1)
- **lambda4Fcast_recrr_like:** lambda for Fcast_recrr_like occurring before endyr+1
- **last_early_yr_nobias_adj:** last_yr_nobias_adj_in_MPD; begin of ramp
- **first_yr_fullbias_adj:** first_yr_fullbias_adj_in_MPD; begin of plateau
- **last_yr_fullbias_adj:** last_yr_fullbias_adj_in_MPD
- **first_recent_yr_nobias_adj:** end_yr_for_ramp_in_MPD (can be in forecast to shape ramp, but SS3 sets bias_adj to 0.0 for fcast yrs)

- `max_bias_adj`: `max_bias_adj_in_MPD`; (typical ~0.8; -3 sets all years to 0.0; -2 sets all non-forecast yrs w/ estimated recdevs to 1.0; -1 sets `biasadj=1.0` for all yrs w/ recdevs).
- `period_of_cycles_in_recr`: period of cycles in recruitment (N parms read below)
- `min_rec_dev`:
- `max_rec_dev`:
- `N_Read_recdevs`:

Las opciones avanzadas permiten al usuario sesgar el ajuste de las desviaciones de reclutamiento. Hay más información sobre el ajuste del sesgo en el manual del usuario de SS, pero la idea general es tener en cuenta el hecho de que las desviaciones de reclutamiento anteriores y posteriores probablemente tengan menos información que las del medio.

La rampa de ajuste de sesgo tiene esto en cuenta y, por lo general, se “ajusta” observando la rampa de sesgo en los resultados del modelo después de ejecutarlo, volviendo a especificar la rampa de sesgo según sea necesario y volviendo a ejecutar el modelo

```
##-----
ctl1$do_recdev <- 1
ctl1$MainRdevYrFirst <- 1989
ctl1$MainRdevYrLast <- 2022
ctl1$recdev_phase <- 1
ctl1$recdev_adv <- 0
ctl1$recdev_early_start <- NULL
ctl1$recdev_early_phase <- NULL
ctl1$Fcast_recr_phase <- NULL
ctl1$lambda4Fcast_recr_like <- NULL
ctl1$last_early_yr_nobias_adj <- NULL
ctl1$first_yr_fullbias_adj <- NULL
ctl1$last_yr_fullbias_adj <- NULL
ctl1$first_recent_yr_nobias_adj <- NULL
ctl1$max_bias_adj <- NULL
ctl1$period_of_cycles_in_recr <- NULL
ctl1$min_rec_dev <- NULL
ctl1$max_rec_dev <- NULL
ctl1$N_Read_recdevs <- NULL
##-----
# revisar salidas
ctl1[48:52]
## $SR_parms
##
##      LO    HI   INIT PRIOR  PR_SD PR_type PHASE env_var&link dev_link
## SR_LN(RO)   5.0 20.0 13.00 0.000  0.000      0      1          0         0
## SR_SCAA_null 0.2  1.0  0.88 0.777  0.113      2     -4          0         0
## SR_sigmaR    0.3  1.6  0.60 1.100 99.000      0     -6          0         0
## SR_regime   -5.0  5.0  0.00 0.000 99.000      0    -50          0         0
## SR_autocorr  0.0  2.0  0.00 1.000 99.000      0    -50          0         0
##
##      dev_minyr dev_maxyr dev_PH Block Block_Fxn PType
## SR_LN(RO)      0        0      0      0          0  5.0
## SR_SCAA_null    0        0      0      0          0  0.2
## SR_sigmaR       0        0      0      0          0  0.3
## SR_regime       0        0      0      0          0 -5.0
## SR_autocorr     0        0      0      0          0  0.0
##
## $do_recdev
```

```
## [1] 1
##
## $MainRdevYrFirst
## [1] 1989
##
## $MainRdevYrLast
## [1] 2022
##
## $recdev_phase
## [1] 1
```

0.12 Mortalidad por pesca

Consulte la Guía de usuario de SS3: [8.7 Fishing Mortality Method](#).

- `F_ballpark`: F ballpark value in units of annual_F
- `F_ballpark_year`: F ballpark year (neg value to disable)
- `F_Method`: 1=Pope midseason rate; 2=F as parameter; 3=F as hybrid; 4=fleet-specific parm/hybrid (#4 is superset of #2 and #3 and is recommended)
- `maxF`: max F (methods 2-4) or harvest fraction (method 1)
- `F_iter`: N iterations for tuning in hybrid mode; recommend 3 (faster) to 5 (more precise if many fleets)

```
#-----
ctl1$F_ballpark      <- 0.1
ctl1$F_ballpark_year <- -1989
ctl1$F_Method        <- 3
ctl1$maxF            <- 4
ctl1$F_iter          <- 3
#-----
# revisa salidas
ctl1[53:57]
## $recdev_adv
## [1] 0
##
## $F_ballpark
## [1] 0.1
##
## $F_ballpark_year
## [1] -1989
##
## $F_Method
## [1] 3
##
## $maxF
## [1] 4
#-----
```

0.13 Capturabilidad

Consulte la Guía de usuario de SS3: [8.8 Catchability](#).

0.13.1 Opciones de capturabilidad por flota (CPUE o CAMPAÑAS)

- `fleet`: number

- `link_type`: (1=simple q, 1 parm;2=mirror simple q, 1 mirrored parm;3=q and power, 2 parm;4=mirror with offset, 2 parm)
- `link_info`: extra input for link, i.e. mirror fleet# or dev index number
- `extra_se`: 0/1 to select extra sd parameter
- `biasadj`: 0/1 for biasadj or not
- `float`: 0/1 to float

```
#-----
Q_options1<-data.frame("fleet"      = 2,
                      "link"       = 1,
                      "link_info"  = 0,
                      "extra_se"   = 0,
                      "biasadj"    = 0,
                      "float"      = 0)

ctl1$Q_options[1,]<-Q_options1
rownames(ctl1$Q_options)[1]<-"PELAGO"
#-----
Q_options2<-data.frame("fleet"      = 3,
                      "link"       = 1,
                      "link_info"  = 0,
                      "extra_se"   = 0,
                      "biasadj"    = 0,
                      "float"      = 0)

ctl1$Q_options[2,]<-Q_options2
rownames(ctl1$Q_options)[2]<-"ECOCADIZ"
#-----
#revisa salidas
ctl1$Q_options
##      fleet link link_info extra_se biasadj float
## PELAGO      2   1         0         0         0   0
## ECOCADIZ     3   1         0         0         0   0
```

0.13.2 Parámetros

```
#-----
Q_parms1<-data.frame("LO"          = -30,
                    "HI"          = 15,
                    "INIT"        = -0.158817,
                    "PRIOR"       = 0,
                    "PR_SD"       = 1,
                    "PR_type"     = 0,
                    "PHASE"       = 1,
                    "env_var&link" = 0,
                    "dev_link"    = 0,
                    "dev_minyr"   = 0,
                    "dev_maxyr"   = 0,
                    "dev_PH"      = 0,
                    "Block"       = 0,
                    "Block_Fxn"   = 0)

ctl1$Q_parms[1,]<-Q_parms1
rownames(ctl1$Q_parms)[1]<-"LnQ_base_PELAGO(2)"
#-----
Q_parms2<-data.frame("LO"          = -30,
                    "HI"          = 15,
```

```

      "INIT"          = 0.08918,
      "PRIOR"         = 0,
      "PR_SD"         = 1,
      "PR_type"       = 0,
      "PHASE"         = 1,
      "env_var&link"  = 0,
      "dev_link"      = 0,
      "dev_minyr"     = 0,
      "dev_maxyr"     = 0,
      "dev_PH"        = 0,
      "Block"         = 0,
      "Block_Fxn"     = 0)

ctl1$Q_parms[2,]<-Q_parms2
rownames(ctl1$Q_parms)[2]<-"LnQ_base_ECOCADIZ(3)"
#-----
#elimino survey 3 del modelo base
ctl1$Q_parms <- subset(ctl1$Q_parms, !rownames(ctl1$Q_parms) %in% c("LnQ_base_SURVEY2(3)"))
#revisa salidas
ctl1$Q_parms
##
##      LO HI      INIT PRIOR PR_SD PR_type PHASE env_var&link
## LnQ_base_PELAGO(2) -30 15 -0.158817      0      1      0      1      0
## LnQ_base_ECOCADIZ(3) -30 15  0.089180      0      1      0      1      0
##
##      dev_link dev_minyr dev_maxyr dev_PH Block Block_Fxn
## LnQ_base_PELAGO(2)      0      0      0      0      0      0
## LnQ_base_ECOCADIZ(3)      0      0      0      0      0      0

```

0.14 Selectividad

Consulte la Guía de usuario de SS3: [8.9 Selectivity and Discard](#).

0.14.1 tipos de selectividad a la talla

- Pattern: _0; parm=0; selex=1.0 for all sizes
- Pattern: _1; parm=2; logistic; with 95% width specification
- Pattern: _5; parm=2; mirror another size selex; PARMS pick the min-max bin to mirror
- Pattern: _11; parm=2; selex=1.0 for specified min-max population length bin range
- Pattern: _15; parm=0; mirror another age or length selex
- Pattern: _6; parm=2+special; non-parm len selex
- Pattern: _43; parm=2+special+2; like 6, with 2 additional param for scaling (average over bin range)
- Pattern: _8; parm=8; double _logistic with smooth transitions and constant above Linf option
- Pattern: _9; parm=6; simple 4-param double logistic with starting length; parm 5 is first length; parm 6=1 does desc as offset
- Pattern: _21; parm=2+special; non-parm len selex, read as pairs of size, then selex
- Pattern: _22; parm=4; double _normal as in CASAL
- Pattern: _23; parm=6; double _normal where final value is directly equal to sp(6) so can be >1.0
- Pattern: _24; parm=6; double _normal with sel(minL) and sel(maxL), using joiners
- Pattern: _2; parm=6; double _normal with sel(minL) and sel(maxL), using joiners, back compatible version of 24 with 3.30.18 and older
- Pattern: _25; parm=3; exponential-logistic in length
- Pattern: _27; parm=special+3; cubic spline in length; parm1==1 resets knots; parm1==2 resets all
- Pattern: _42; parm=special+3+2; cubic spline; like 27, with 2 additional param for scaling (average over bin range)
- discard_options: _0=none; _1=define_retention; _2=retention&mortality; _3=all_discarded_dead; _4=define_dome-shaped_retention

```

#-----
size_selex_types1<-data.frame("Pattern" = 1,
                              "Discard" = 0,
                              "Male"    = 0,
                              "Special" = 0)
ctl1$size_selex_types[1,]<-size_selex_types1
rownames(ctl1$size_selex_types)[1]<-"Fishery"
#-----
size_selex_types2<-data.frame("Pattern" = 1,
                              "Discard" = 0,
                              "Male"    = 0,
                              "Special" = 0)
ctl1$size_selex_types[2,]<-size_selex_types2
rownames(ctl1$size_selex_types)[2]<-"PELAGO"
#-----
size_selex_types3<-data.frame("Pattern" = 1,
                              "Discard" = 0,
                              "Male"    = 0,
                              "Special" = 0)
ctl1$size_selex_types[3,]<-size_selex_types3
rownames(ctl1$size_selex_types)[3]<-"ECOCADIZ"
#-----
#revisa salidas
ctl1$size_selex_types
##           Pattern Discard Male Special
## Fishery         1         0    0        0
## PELAGO          1         0    0        0
## ECOCADIZ        1         0    0        0

```

0.14.2 tipos de selectividad a la edad

- Pattern: _0; parm=0; selex=1.0 for ages 0 to maxage
- Pattern: _10; parm=0; selex=1.0 for ages 1 to maxage
- Pattern: _11; parm=2; selex=1.0 for specified min-max age
- Pattern: _12; parm=2; age logistic
- Pattern: _13; parm=8; age double logistic. Recommend using pattern 18 instead.
- Pattern: _14; parm=nages+1; age empirical
- Pattern: _15; parm=0; mirror another age or length selex
- Pattern: _16; parm=2; Coleraine - Gaussian
- Pattern: _17; parm=nages+1; empirical as random walk N parameters to read can be overridden by setting special to non-zero
- Pattern: _41; parm=2+nages+1; // like 17, with 2 additional param for scaling (average over bin range)
- Pattern: _18; parm=8; double logistic - smooth transition
- Pattern: _19; parm=6; simple 4-param double logistic with starting age
- Pattern: _20; parm=6; double_normal,using joiners
- Pattern: _26; parm=3; exponential-logistic in age
- Pattern: _27; parm=3+special; cubic spline in age; parm1==1 resets knots; parm1==2 resets all
- Pattern: _42; parm=2+special+3; // cubic spline; with 2 additional param for scaling (average over bin range)
- Age patterns entered with value >100 create Min_selage from first digit and pattern from remainder

```

#-----
age_selex_types1<-data.frame(Pattern = 12,

```



```

Discard = 0,
Male     = 0,
Special = 0)
ctl1$age_selex_types[1,]<-age_selex_types1
rownames(ctl1$age_selex_types)[1]<-"Fishery"
#-----
age_selex_types2<-data.frame(Pattern = 12,
                             Discard = 0,
                             Male     = 0,
                             Special = 0)
ctl1$age_selex_types[2,]<-age_selex_types2
rownames(ctl1$age_selex_types)[2]<-"PELAGO"
#-----
age_selex_types3<-data.frame(Pattern = 12,
                             Discard = 0,
                             Male     = 0,
                             Special = 0)
ctl1$age_selex_types[3,]<-age_selex_types3
rownames(ctl1$age_selex_types)[3]<-"ECOCADIZ"
#-----
#revisa salidas
ctl1$age_selex_types
##           Pattern Discard Male Special
## Fishery         12      0    0      0
## PELAGO          12      0    0      0
## ECOCADIZ        12      0    0      0

```

0.14.3 parametros de selectividad a la talla

```

#-----
size_selex_parms1<-data.frame("LO"           = -1,
                              "HI"           = 20,
                              "INIT"         = 12,
                              "PRIOR"        = 0,
                              "PR_SD"        = 0,
                              "PR_type"      = 0,
                              "PHASE"        = 2,
                              "env_var&link" = 0,
                              "dev_link"     = 0,
                              "dev_minyr"    = 0,
                              "dev_maxyr"    = 0,
                              "dev_PH"       = 0.5,
                              "Block"        = 0,
                              "Block_Fxn"    = 0)
ctl1$size_selex_parms[1,]<-size_selex_parms1
rownames(ctl1$size_selex_parms)[1]<-"SizeSel_P_1_Fishery(1)"
#-----
size_selex_parms2<-data.frame("LO"           = -1,
                              "HI"           = 20,
                              "INIT"         = 18,
                              "PRIOR"        = 0,
                              "PR_SD"        = 0,
                              "PR_type"      = 0,

```

```

        "PHASE"           = 2,
        "env_var&link"    = 0,
        "dev_link"       = 0,
        "dev_minyr"      = 0,
        "dev_maxyr"      = 0,
        "dev_PH"         = 0.5,
        "Block"          = 0,
        "Block_Fxn"      = 0)
ctl1$size_selex_parms[2,]<-size_selex_parms2
rownames(ctl1$size_selex_parms)[2]<-"SizeSel_P_2_Fishery(1)"
#-----
size_selex_parms3<-data.frame("L0"           = -3,
                              "HI"           = 8,
                              "INIT"         = 6,
                              "PRIOR"        = 0,
                              "PR_SD"        = 0,
                              "PR_type"      = 0,
                              "PHASE"        = 3,
                              "env_var&link" = 0,
                              "dev_link"     = 0,
                              "dev_minyr"    = 0,
                              "dev_maxyr"    = 0,
                              "dev_PH"       = 0.5,
                              "Block"        = 0,
                              "Block_Fxn"    = 0)
ctl1$size_selex_parms[3,]<-size_selex_parms3
rownames(ctl1$size_selex_parms)[3]<-"SizeSel_P_1_PELAGO(2)"
#-----
size_selex_parms4<-data.frame("L0"           = -3,
                              "HI"           = 16,
                              "INIT"         = 10,
                              "PRIOR"        = 0,
                              "PR_SD"        = 0,
                              "PR_type"      = 0,
                              "PHASE"        = 3,
                              "env_var&link" = 0,
                              "dev_link"     = 0,
                              "dev_minyr"    = 0,
                              "dev_maxyr"    = 0,
                              "dev_PH"       = 0.5,
                              "Block"        = 0,
                              "Block_Fxn"    = 0)
ctl1$size_selex_parms[4,]<-size_selex_parms4
rownames(ctl1$size_selex_parms)[4]<-"SizeSel_P_2_PELAGO(2)"
#-----
size_selex_parms5<-data.frame("L0"           = -1,
                              "HI"           = 10,
                              "INIT"         = 8,
                              "PRIOR"        = 0,
                              "PR_SD"        = 0,
                              "PR_type"      = 0,
                              "PHASE"        = 3,
                              "env_var&link" = 0,

```

```

        "dev_link"      = 0,
        "dev_minyr"     = 0,
        "dev_maxyr"     = 0,
        "dev_PH"        = 0.5,
        "Block"         = 0,
        "Block_Fxn"     = 0)
ctl1$size_selex_parms[5,]<-size_selex_parms5
rownames(ctl1$size_selex_parms)[5]<-"SizeSel_P_1_ECOCADIZ(3)"
#-----
size_selex_parms6<-data.frame("LO"          = -1,
                              "HI"          = 20.5,
                              "INIT"        = 15,
                              "PRIOR"       = 0,
                              "PR_SD"       = 0,
                              "PR_type"     = 0,
                              "PHASE"       = 3,
                              "env_var&link" = 0,
                              "dev_link"    = 0,
                              "dev_minyr"   = 0,
                              "dev_maxyr"   = 0,
                              "dev_PH"      = 0.5,
                              "Block"       = 0,
                              "Block_Fxn"   = 0)
ctl1$size_selex_parms[6,]<-size_selex_parms6
rownames(ctl1$size_selex_parms)[6]<-"SizeSel_P_2_ECOCADIZ(3)"
#-----
# revisa salidas
ctl1$size_selex_parms
##
##      LO    HI  INIT PRIOR PR_SD PR_type PHASE env_var&link
## SizeSel_P_1_Fishery(1) -1 20.0  12    0    0    0    2      0
## SizeSel_P_2_Fishery(1) -1 20.0  18    0    0    0    2      0
## SizeSel_P_1_PELAGO(2) -3  8.0   6    0    0    0    3      0
## SizeSel_P_2_PELAGO(2) -3 16.0  10    0    0    0    3      0
## SizeSel_P_1_ECOCADIZ(3) -1 10.0   8    0    0    0    3      0
## SizeSel_P_2_ECOCADIZ(3) -1 20.5  15    0    0    0    3      0
##
##      dev_link dev_minyr dev_maxyr dev_PH Block Block_Fxn
## SizeSel_P_1_Fishery(1)      0      0      0  0.5    0      0
## SizeSel_P_2_Fishery(1)      0      0      0  0.5    0      0
## SizeSel_P_1_PELAGO(2)      0      0      0  0.5    0      0
## SizeSel_P_2_PELAGO(2)      0      0      0  0.5    0      0
## SizeSel_P_1_ECOCADIZ(3)      0      0      0  0.5    0      0
## SizeSel_P_2_ECOCADIZ(3)      0      0      0  0.5    0      0

```

0.14.4 parametros de selectividad a la edad

```

age_selex_parms1<-data.frame("LO"          = -2.0,
                              "HI"          = 5.50,
                              "INIT"        = 0.10,
                              "PRIOR"       = 0,
                              "PR_SD"       = 0.01,
                              "PR_type"     = 0,
                              "PHASE"       = -1,
                              "env_var&link" = 0,

```

```

        "dev_link"      = 0,
        "dev_minyr"     = 0,
        "dev_maxyr"     = 0,
        "dev_PH"        = 0.5,
        "Block"         = 0,
        "Block_Fxn"     = 0)
ctl1$age_selex_parms[1,]<-age_selex_parms1
rownames(ctl1$age_selex_parms)[1]<-"AgeSel_P_1_Fishery(1)"
#-----
age_selex_parms2<-data.frame("L0"          = -1.0,
                             "HI"          = 5.50,
                             "INIT"        = 5.0,
                             "PRIOR"       = 0,
                             "PR_SD"       = 0.01,
                             "PR_type"     = 0,
                             "PHASE"       = -1,
                             "env_var&link" = 0,
                             "dev_link"    = 0,
                             "dev_minyr"   = 0,
                             "dev_maxyr"   = 0,
                             "dev_PH"      = 0.5,
                             "Block"       = 0,
                             "Block_Fxn"   = 0)
ctl1$age_selex_parms[2,]<-age_selex_parms2
rownames(ctl1$age_selex_parms)[2]<-"AgeSel_P_2_Fishery(1)"
#-----
age_selex_parms3<-data.frame("L0"          = -2.0,
                             "HI"          = 5.50,
                             "INIT"        = 0.10,
                             "PRIOR"       = 0,
                             "PR_SD"       = 0.01,
                             "PR_type"     = 0,
                             "PHASE"       = -1,
                             "env_var&link" = 0,
                             "dev_link"    = 0,
                             "dev_minyr"   = 0,
                             "dev_maxyr"   = 0,
                             "dev_PH"      = 0.5,
                             "Block"       = 0,
                             "Block_Fxn"   = 0)
ctl1$age_selex_parms[3,]<-age_selex_parms3
rownames(ctl1$age_selex_parms)[3]<-"AgeSel_P_1_PELAGO(2)"
#-----
age_selex_parms4<-data.frame("L0"          = -1.0,
                             "HI"          = 5.50,
                             "INIT"        = 5.00,
                             "PRIOR"       = 0,
                             "PR_SD"       = 0.01,
                             "PR_type"     = 0,
                             "PHASE"       = -1,
                             "env_var&link" = 0,
                             "dev_link"    = 0,
                             "dev_minyr"   = 0,

```

```

        "dev_maxyr"      = 0,
        "dev_PH"        = 0.5,
        "Block"         = 0,
        "Block_Fxn"     = 0)
ctl1$age_selex_parms[4,]<-age_selex_parms4
rownames(ctl1$age_selex_parms)[4]<-"AgeSel_P_2_PELAGO(2)"
#-----
age_selex_parms5<-data.frame("LO"          = -2.0,
                             "HI"          = 5.50,
                             "INIT"        = 0.10,
                             "PRIOR"       = 0,
                             "PR_SD"       = 0.01,
                             "PR_type"     = 0,
                             "PHASE"       = -1,
                             "env_var&link" = 0,
                             "dev_link"    = 0,
                             "dev_minyr"   = 0,
                             "dev_maxyr"   = 0,
                             "dev_PH"      = 0.5,
                             "Block"       = 0,
                             "Block_Fxn"   = 0)
ctl1$age_selex_parms[5,]<-age_selex_parms5
rownames(ctl1$age_selex_parms)[5]<-"AgeSel_P_1_ECOCADIZ(3)"
#-----
age_selex_parms6<-data.frame("LO"          = -1.0,
                             "HI"          = 5.50,
                             "INIT"        = 5.00,
                             "PRIOR"       = 0,
                             "PR_SD"       = 0.01,
                             "PR_type"     = 0,
                             "PHASE"       = -1,
                             "env_var&link" = 0,
                             "dev_link"    = 0,
                             "dev_minyr"   = 0,
                             "dev_maxyr"   = 0,
                             "dev_PH"      = 0.5,
                             "Block"       = 0,
                             "Block_Fxn"   = 0)
ctl1$age_selex_parms[6,]<-age_selex_parms6
rownames(ctl1$age_selex_parms)[6]<-"AgeSel_P_2_ECOCADIZ(3)"
#-----
# revisa salidas
ctl1$age_selex_parms
##
##      LO  HI  INIT  PRIOR  PR_SD  PR_type  PHASE  env_var&link
## AgeSel_P_1_Fishery(1) -2 5.5  0.1    0  0.01    0    -1          0
## AgeSel_P_2_Fishery(1) -1 5.5  5.0    0  0.01    0    -1          0
## AgeSel_P_1_PELAGO(2) -2 5.5  0.1    0  0.01    0    -1          0
## AgeSel_P_2_PELAGO(2) -1 5.5  5.0    0  0.01    0    -1          0
## AgeSel_P_1_ECOCADIZ(3) -2 5.5  0.1    0  0.01    0    -1          0
## AgeSel_P_2_ECOCADIZ(3) -1 5.5  5.0    0  0.01    0    -1          0
##
##      dev_link dev_minyr dev_maxyr dev_PH Block Block_Fxn
## AgeSel_P_1_Fishery(1)      0      0      0  0.5    0      0
## AgeSel_P_2_Fishery(1)      0      0      0  0.5    0      0

```

## AgeSel_P_1_PELAGO(2)	0	0	0	0.5	0	0
## AgeSel_P_2_PELAGO(2)	0	0	0	0.5	0	0
## AgeSel_P_1_ECOCADIZ(3)	0	0	0	0.5	0	0
## AgeSel_P_2_ECOCADIZ(3)	0	0	0	0.5	0	0

0.14.5 otros parámetros

- Use_2D_AR1_selectivity: use 2D_AR1 selectivity(0/1)
- TG_custom: 0=no read and autogen if tag data exist; 1=read
- DoVar_adjust: 1= add_to_survey_CV; 2=add_to_discard_stddev; 3=add_to_bodywt_CV; 4=mult_by_lencomp_N; 5=mult_by_agecomp_N; 6=mult_by_size-at-age_N; 7=mult_by_generalized_sizecomp
- Variance_adjustment_list:
- maxlambdaphase:
- sd_offset: must be 1 if any growthCV, sigmaR, or survey extraSD is an estimated parameter
- maxlambdaphase:
- sd_offset:
- lambdas:
- N_lambdas:
- more_stddev_reporting:
- stddev_reporting_specs:
- stddev_reporting_selex:
- stddev_reporting_growth:
- stddev_reporting_N_at_A:

```
#-----
ctl1$Use_2D_AR1_selectivity <-0
ctl1$TG_custom <-0
ctl1$DoVar_adjust <-1
# revisa salidas
ctl1[64:66]
## $age_selex_parms
##          LO  HI  INIT PRIOR PR_SD PR_type PHASE env_var&link
## AgeSel_P_1_Fishery(1) -2 5.5  0.1    0 0.01    0   -1         0
## AgeSel_P_2_Fishery(1) -1 5.5  5.0    0 0.01    0   -1         0
## AgeSel_P_1_PELAGO(2) -2 5.5  0.1    0 0.01    0   -1         0
## AgeSel_P_2_PELAGO(2) -1 5.5  5.0    0 0.01    0   -1         0
## AgeSel_P_1_ECOCADIZ(3) -2 5.5  0.1    0 0.01    0   -1         0
## AgeSel_P_2_ECOCADIZ(3) -1 5.5  5.0    0 0.01    0   -1         0
##          dev_link dev_minyr dev_maxyr dev_PH Block Block_Fxn
## AgeSel_P_1_Fishery(1)    0        0        0  0.5    0        0
## AgeSel_P_2_Fishery(1)    0        0        0  0.5    0        0
## AgeSel_P_1_PELAGO(2)    0        0        0  0.5    0        0
## AgeSel_P_2_PELAGO(2)    0        0        0  0.5    0        0
## AgeSel_P_1_ECOCADIZ(3)  0        0        0  0.5    0        0
## AgeSel_P_2_ECOCADIZ(3)  0        0        0  0.5    0        0
##
## $Use_2D_AR1_selectivity
## [1] 0
##
## $TG_custom
## [1] 0
#-----
```

```

varadj0 <- list()
varadj1<-data.frame(row.names=c("Variance_adjustment_list1",
                                "Variance_adjustment_list2",
                                "Variance_adjustment_list3"),
                    "Factor"   = c(4,4,4),
                    "Fleet"    = c(1,2,3),
                    "Value"    = c(0.0045,0.0051,0.0047))

varadj0[[1]] <- varadj1
names(varadj0)<-"Variance_adjustment_list"

ctl1 <- append(ctl1, varadj0, after = 66)

ctl1$Variance_adjustment_list
##                               Factor Fleet  Value
## Variance_adjustment_list1      4      1 0.0045
## Variance_adjustment_list2      4      2 0.0051
## Variance_adjustment_list3      4      3 0.0047

ctl1[67]
## $Variance_adjustment_list
##                               Factor Fleet  Value
## Variance_adjustment_list1      4      1 0.0045
## Variance_adjustment_list2      4      2 0.0051
## Variance_adjustment_list3      4      3 0.0047

#-----
ctl1$maxlambdaphase      <- 1
ctl1$sd_offset           <- 1
ctl1$lambdas             <- NULL
ctl1$N_lambdas           <- 0
ctl1$more_stddev_reporting <- 0
ctl1$stddev_reporting_specs <- NULL
ctl1$stddev_reporting_selex <- NULL
ctl1$stddev_reporting_growth <- NULL
ctl1$stddev_reporting_N_at_A <- NULL
#-----

ctl1[68:71]
## $DoVar_adjust
## [1] 1
##
## $maxlambdaphase
## [1] 1
##
## $sd_offset
## [1] 1
##
## $N_lambdas
## [1] 0

```

0.14.6 Escribir archivo de control modificado con la función SS_write para el enfoque de modelación SS3

```
#-----  
r4ss::SS_writectl(ctl1,  
                  outfile=here(dirname.mod,"control.ss"),  
                  overwrite = TRUE)  
#-----
```