

Formato Archivo Control.ss

abril, 18, 2023

Contents

1	Contexto	2
1.1	Identificamos los directorio de trabajo	2
1.2	Leer los archivos de Stock Synthesis con la función <code>SS_read()</code>	2
1.3	Revisamos los nombres de los componentes de la lista del archivo control que deseamos modificar	2
1.4	Especificaciones iniciales	4
1.5	Datos de los archivos	4
1.6	Especificaciones del crecimiento	4
1.7	Distribución del reclutamiento	4
1.8	Bloques	5
1.9	Parámetros que varían en el tiempo	6
1.10	Parámetros biológicos	7
1.11	Relación stock recluta	18
1.12	Desvíos de los reclutamientos	20
1.13	Mortalidad por pesca	23
1.14	Capturabilidad	24
1.15	Selectividad	26
1.16	more stddev reporting	33

1 Contexto

1.1 Identificamos los directorio de trabajo

```
dirname.base <- here("10a_anchcadiz")
```

1.1.0.1 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)
```

1.2 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
```

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

```
names(ctl1)  
## [1] "warnings"      "Comments"  
## [3] "nseas"         "N_areas"  
## [5] "Nages"         "Nsexes"  
## [7] "Npopbins"      "Nfleets"  
## [9] "Do_AgeKey"     "fleetnames"  
## [11] "sourcefile"    "type"  
## [13] "ReadVersion"   "eof"  
## [15] "EmpiricalWAA"  "N_GP"  
## [17] "N_platoon"     "recr_dist_method"  
## [19] "recr_global_area" "recr_dist_read"  
## [21] "recr_dist_inx" "recr_dist_pattern"  
## [23] "N_Block_Designs" "blocks_per_pattern"  
## [25] "Block_Design"  "time_vary_adjust_method"  
## [27] "time_vary_auto_generation" "natM_type"  
## [29] "GrowthModel"   "Growth_Age_for_L1"  
## [31] "Growth_Age_for_L2" "Exp_Decay"  
## [33] "Growth_Placeholder" "N_natMparms"  
## [35] "SD_add_to_LAA"  "CV_Growth_Pattern"  
## [37] "maturity_option" "First_Mature_Age"  
## [39] "fecundity_option" "hermaphroditism_option"  
## [41] "parameter_offset_approach" "MG_parms"  
## [43] "MGparm_seas_effects" "SR_function"  
## [45] "Use_steep_init_equi" "Sigma_R_FofCurvature"  
## [47] "SR_parms"      "do_recdev"  
## [49] "MainRdevYrFirst" "MainRdevYrLast"
```

```

## [51] "recdev_phase"          "recdev_adv"
## [53] "F_ballpark"           "F_ballpark_year"
## [55] "F_Method"             "maxF"
## [57] "F_iter"               "Q_options"
## [59] "Q_parms"              "size_selex_types"
## [61] "age_selex_types"      "size_selex_parms"
## [63] "age_selex_parms"     "Use_2D_AR1_selectivity"
## [65] "TG_custom"            "DoVar_adjust"
## [67] "Variance_adjustment_list" "maxlambdaphase"
## [69] "sd_offset"            "N_lambdas"
## [71] "more_stddev_reporting"

```

1.4 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
```

1.5 Datos de los archivos

```
ctl1$fleetnames <- c("Fishery","PELAGO","ECOCADIZ")
ctl1$Comments<-"#C 2023 Boqueron Cádiz control file MODELO TRIMESTRAL"
ctl1$eof        <- TRUE
```

1.6 Especificaciones del crecimiento

```
#-----
# EmpiricalWAA
#-----
# 0 means do not read wtatage.ss;
# 1 means read and use wtatage.ss and
# also read and use growth parameters
ctl1$EmpiricalWAA <- 0
#-----
# N_GP
#_N_Growth_Patterns
#-----
# (Growth Patterns,
# Morphs,
# Bio Patterns,
# GP
# are terms used interchangeably in SS3)
ctl1$N_GP <- 1
#-----
#N_platoon
#_N_platoons_Within_GrowthPattern
#-----
ctl1$N_platoon <- 1
```

1.7 Distribución del reclutamiento

```
#-----
# recr_dist_method for parameters:
#-----
# 2=main effects for GP, Area, Settle timing;
# 3=each Settle entity;
# 4=none (only when N_GP*Nsettle*pop==1)
ctl1$recr_dist_method<-3
#-----
# not yet implemented; Future usage: Spawner-Recruitment:
```

```

#-----
# 1=global;
# 2=by area
ctl1$recre_global_area<-1
#-----
# number of recruitment settlement assignments
#-----
ctl1$recre_dist_read<-1
#-----
# unused option
#-----
ctl1$recre_dist_inx<-0
#-----
# GPattern month area age (for each settlement assignment)
#-----
rec_pattern<-data.frame(row.names="recre_dist_pattern1",
                        "GPattern" = 1,
                        "month"    = 1,
                        "area"     = 1,
                        "age"      = 0)
ctl1$recre_dist_pattern<-rec_pattern
#-----
#_Cond 0 # N_movement_definitions goes here if Nareas > 1
#-----

#-----
#_Cond 1.0 # first age that moves (real age at begin of season, not integer)
# also cond on do_migration>0
#-----

#-----
#_Cond 1 1 1 2 4 10 # example move definition for
#-----
# seas=1,
# morph=1,
# source=1
# dest=2,
# age1=4,
# age2=10

```

1.8 Bloques

```

#-----
#_Nbblock_Patterns
#-----
ctl1$N_Block_Designs <- 1
#-----
#_blocks_per_pattern
#-----
ctl1$blocks_per_pattern<-1
#-----
# begin and end years of blocks

```

```
#-----
ctl1$Block_Design<-c(1989,1989)
```

1.9 Parámetros que varían en el tiempo

```
#-----
# controls for all timevary parameters
#-----
#_time-vary parm bound check (
  # 1=warn relative to base parm bounds;
  # 3=no bound check);
  # Also see env (3) and dev (5) options to constrain with base bounds
ctl1$time_vary_adjust_method <- 1

#-----
# AUTOGEN
#-----
# autogen:
  # 1st element for biology,
  # 2nd for SR,
  # 3rd for Q,
  # 4th reserved,
  # 5th for seleæ
# where: 0 = autogen time-varying parms of this category;
#         1 = read each time-varying parm line;
#         2 = read then autogen if parm min==12345
#
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

#-----
#_Available timevary codes
#-----
#_Block types:
#-----
  # 0: P_block=P_base*exp(TVP);
  # 1: P_block=P_base+TVP;
  # 2: P_block=TVP;
  # 3: P_block=P_block(-1) + TVP

#-----
#_Block_trends:
#-----
  # -1: trend bounded by base parm min-max and parms in transformed units (beware);
  # -2: endtrend and infl_year direct values;
  # -3: end and infl as fraction of base range

#-----
#_EnvLinks:
#-----
  # 1: P(y)=P_base*exp(TVP*env(y));
```

```

# 2: P(y)=P_base+TVP*env(y);
# 3: P(y)=f(TVP,env_Zscore) w/ logit to stay in min-max;
# 4: P(y)=2.0/(1.0+exp(-TVP1*env(y) - TVP2))

#-----
#_DevLinks:
#-----
# 1: P(y)*=exp(dev(y)*dev_se;
# 2: P(y)+=dev(y)*dev_se;
# 3: random walk;
# 4: zero-reverting random walk with rho;
# 5: like 4 with logit transform to stay in base min-max

#-----
#_DevLinks(more):
#-----
# 21-25 keep last dev for rest of years
#
#-----
#_Prior_codes:
#-----
# 0=none;
# 6=normal;
# 1=symmetric beta;
# 2=CASAL's beta;
# 3=lognormal;
# 4=lognormal with biascorr;
# 5=gamma
#

```

1.10 Parámetros biológicos

```

# setup for M, growth, wt-len, maturity, fecundity, (hermaphro), recr_distr, cohort_grow, (movement), (
#-----
#_NATMORT
#-----
#_natM_type:
#_0=1Parm;
#_1=N_breakpoints;
#_2=Lorenzen;
#_3=agespecific;
#_4=agespec_withseasinterpolate;
#_5=BETA:_Maunder_link_to_maturity;
#_6=Lorenzen_range
#_no additional input for selected M option; read 1P per morph
#
ctl1$natM_type<-0

#-----
# GrowthModel:
#-----
# 1=vonBert 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
ctl1$GrowthModel<-1
#-----
#_Age(post-settlement)_for_L1;linear growth below this
#-----
ctl1$Growth_Age_for_L1<-0.1
#-----
#_Growth_Age_for_L2 (999 to use as Linf)
#-----
ctl1$Growth_Age_for_L2<-4
#-----
#_exponential decay for growth above maxage
#(value should approx initial Z;
# -999 replicates 3.24;
# -998 to not allow growth above maxage)
#-----
ctl1$Exp_Decay<- -999
#-----
#_placeholder for future growth feature
#-----
ctl1$Growth_Placeholder <- 0

#####?????????
ctl1$N_natMparms <- 1 # Buscar----- modelo simple no lo tiene, revisar
#####?????????

#-----
#_SD_add_to_LAA (set to 0.1 for SS2 V1.x compatibility)
#-----
ctl1$SD_add_to_LAA <- 0

#-----
#_CV_Growth_Pattern:
#-----
# 0 CV=f(LAA);
# 1 CV=F(A);
# 2 SD=F(LAA);
# 3 SD=F(A);
# 4 logSD=F(A)
#-----
ctl1$CV_Growth_Pattern <- 0
##-----
#_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
##-----
ctl1$maturity_option <- 1
##-----
#_First_Mature_Age
ctl1$First_Mature_Age <- 1
##-----
#_fecundity_at_length option:
##-----
# (1)eggs=Wt*(a+b*Wt);
# (2)eggs=a*L^b;
# (3)eggs=a*Wt^b;
# (4)eggs=a+b*L;
# (5)eggs=a+b*W
##-----
ctl1$fecundity_option <- 1
##-----
#_hermaphroditism option:
##-----
# 0=none;
# 1=female-to-male age-specific fxn;
# -1=male-to-female age-specific fxn
##-----
ctl1$hermaphroditism_option <- 0
##-----
#_parameter_offset_approach for M, G, CV_G:
##-----
# 1- direct, no offset**
# 2- male=fem_parm*exp(male_parm);
# 3: male=female*exp(parm) then old=young*exp(parm)
##-----
ctl1$parameter_offset_approach <- 1
#_** in option 1, any male parameter with value = 0.0 and phase <0 is set equal to female parameter
#

```

1.10.1 Parámetros biológicos

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] "RecrDist_GP_1_area_1_month_1" "CohortGrowDev"
## [15] "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(ctl$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 vinculo 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

```
##-----
#NatM_p_1_Fem_GP_1
##-----
# t(ctl$MG_parms[1,])

MG_parms1<-data.frame(row.names="NatM_p_1_Fem_GP_1",
                      "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[1,] <- MG_parms1
##-----

##-----
#L_at_Amin_Fem_GP_1
##-----
# t(ctl$MG_parms[2,])
MG_parms2<-data.frame(row.names="L_at_Amin_Fem_GP_1",
```

```

        "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[2,]<-MG_parms2
##-----

##-----
#L_at_Amax_Fem_GP_1
##-----
# t(ctl$MG_parms[3,])
MG_parms3<-data.frame(row.names="L_at_Amax_Fem_GP_1",
        "LO"          = 5,
        "HI"          = 22,
        "INIT"        = 18,
        "PRIOR"       = 50,
        "PR_SD"       = 99,
        "PR_type"     = 0,
        "PHASE"       = 5,
        "env_var&link" = 0,
        "dev_link"    = 0,
        "dev_minyr"   = 0,
        "dev_maxyr"   = 0,
        "dev_PH"      = 0,
        "Block"       = 0,
        "Block_Fxn"   = 0)
ctl1$MG_parms[3,]<-MG_parms3
##-----

##-----
#VonBert_K_Fem_GP_1
##-----
# t(ctl$MG_parms[4,])
MG_parms4<-data.frame(row.names="VonBert_K_Fem_GP_1",
        "LO"          = 0.1,
        "HI"          = 2.0,
        "INIT"        = 0.5,
        "PRIOR"       = 0.3,
        "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[4,]<-MG_parms4
##-----

##-----
#CV_young_Fem_GP_1
##-----
# t(ctl$MG_parms[5,])
MG_parms5<-data.frame(row.names="CV_young_Fem_GP_1",
        "LO"          = 0.03,
        "HI"          = 0.150,
        "INIT"        = 0.066,
        "PRIOR"       = 0.1,
        "PR_SD"       = 99,
        "PR_type"     = 0,
        "PHASE"       = 5,
        "env_var&link" = 0,
        "dev_link"    = 0,
        "dev_minyr"   = 0,
        "dev_maxyr"   = 0,
        "dev_PH"      = 0,
        "Block"       = 0,
        "Block_Fxn"   = 0)
ctl1$MG_parms[5,]<-MG_parms5
##-----

##-----
#CV_old_Fem_GP_1
##-----
# t(ctl$MG_parms[6,])
MG_parms6<-data.frame(row.names="CV_old_Fem_GP_1",
        "LO"          = 0.03,
        "HI"          = 0.150,
        "INIT"        = 0.066,
        "PRIOR"       = 0.1,
        "PR_SD"       = 99,
        "PR_type"     = 0,
        "PHASE"       = 5,
        "env_var&link" = 0,
        "dev_link"    = 0,
        "dev_minyr"   = 0,
        "dev_maxyr"   = 0,
        "dev_PH"      = 0,
        "Block"       = 0,
        "Block_Fxn"   = 0)
ctl1$MG_parms[6,]<-MG_parms6
##-----

##-----

```

```

#Wtlen_1_Fem_GP_1
##-----
# t(ctl$MG_parms[7,])
MG_parms7<-data.frame(row.names="Wtlen_1_Fem_GP_1",
                      "LO"          = -3.0,
                      "HI"          =  3.0,
                      "INIT"        = 0.00563,
                      "PRIOR"       = 0.00563,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = -50,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms[7,<-MG_parms7
##-----

##-----
#Wtlen_2_Fem_GP_1
##-----
# t(ctl$MG_parms[8,])
MG_parms8<-data.frame(row.names="Wtlen_2_Fem_GP_1",
                      "LO"          = -3.0,
                      "HI"          =  3.0,
                      "INIT"        = 3.1591,
                      "PRIOR"       = 3.1591,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = -50,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_minyr"   = 0,
                      "dev_maxyr"   = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
                      "Block_Fxn"   = 0)
ctl1$MG_parms[8,<-MG_parms8
##-----

##-----
#Mat50%_Fem_GP_1
##-----
# t(ctl$MG_parms[9,])
MG_parms9<-data.frame(row.names="Wtlen_2_Fem_GP_1",
                      "LO"          = -3.0,
                      "HI"          =  3.0,
                      "INIT"        = 3.1591,
                      "PRIOR"       = 3.1591,
                      "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[9,]<-MG_parms9
##-----

##-----
#Mat_slope_Fem_GP_1
##-----
# t(ctl$MG_parms[10,])
MG_parms10<-data.frame(row.names="Mat_slope_Fem_GP_1",
        "LO"          = -3.0,
        "HI"          = 3.0,
        "INIT"        = -0.45,
        "PRIOR"       = -0.45,
        "PR_SD"       = 99,
        "PR_type"     = 0,
        "PHASE"       = -50,
        "env_var&link" = 0,
        "dev_link"    = 0,
        "dev_minyr"   = 0,
        "dev_maxyr"   = 0,
        "dev_PH"      = 0,
        "Block"       = 0,
        "Block_Fxn"   = 0)
ctl1$MG_parms[10,]<-MG_parms10
##-----

##-----
#Eggs/kg_inter_Fem_GP_1
##-----
# t(ctl$MG_parms[11,])
MG_parms11<-data.frame(row.names="Eggs/kg_inter_Fem_GP_1",
        "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[11,]<-MG_parms11
##-----

##-----
#Eggs/kg_slope_wt_Fem_GP_1
##-----
# t(ctl$MG_parms[12,])
MG_parms12<-data.frame(row.names="Eggs/kg_slope_wt_Fem_GP_1",
                        "LO"          = -3.0,
                        "HI"          = 3.0,
                        "INIT"         = 1,
                        "PRIOR"        = 1,
                        "PR_SD"        = 99,
                        "PR_type"       = 0,
                        "PHASE"        = -50,
                        "env_var&link" = 0,
                        "dev_link"     = 0,
                        "dev_minyr"    = 0,
                        "dev_maxyr"    = 0,
                        "dev_PH"       = 0,
                        "Block"        = 0,
                        "Block_Fxn"    = 0)
ctl1$MG_parms[12,]<-MG_parms12
##-----

##-----
#RecrDist_GP_1_area_1_month_1
##-----
# t(ctl$MG_parms[13,])
MG_parms13<-data.frame(row.names="RecrDist_GP_1_area_1_month_1",
                        "LO"          = 0,
                        "HI"          = 10,
                        "INIT"         = 1,
                        "PRIOR"        = 1,
                        "PR_SD"        = 99,
                        "PR_type"       = 0,
                        "PHASE"        = -3,
                        "env_var&link" = 0,
                        "dev_link"     = 0,
                        "dev_minyr"    = 0,
                        "dev_maxyr"    = 0,
                        "dev_PH"       = 0,
                        "Block"        = 0,
                        "Block_Fxn"    = 0)
ctl1$MG_parms[13,]<-MG_parms13
##-----

##-----
#CohortGrowDev
##-----
# t(ctl$MG_parms[14,])
MG_parms14<-data.frame(row.names="CohortGrowDev",

```



```

        "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[14,]<-MG_parms14
##-----

##-----
#FracFemale_GP_1
##-----
# t(ctl$MG_parms[15,])
MG_parms15<-data.frame(row.names="FracFemale_GP_1",
        "LO"          = 0.000001,
        "HI"          = 0.999999,
        "INIT"        = 0.5,
        "PRIOR"       = 0.5,
        "PR_SD"       = 0.5,
        "PR_type"     = 0,
        "PHASE"       = -99,
        "env_var&link" = 0,
        "dev_link"    = 0,
        "dev_minyr"   = 0,
        "dev_maxyr"   = 0,
        "dev_PH"      = 0,
        "Block"       = 0,
        "Block_Fxn"   = 0)
ctl1$MG_parms[15,]<-MG_parms15
##-----

1.10.1.0.1 Mortalidad natural Fem GP_1 #####_seasonal_effects_on_biology_parms
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

##-----
#_ LO HI INIT PRIOR PR_SD PR_type PHASE
#_Cond -2 2 0 0 -1 99 -2 #_placeholder when no seasonal MG parameters
#
##-----

```

1.11 Relación stock recluta

```
##-----
#_Spawner-Recruitment; Options:
##-----
# 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
ctl1$SR_function <- 4

##-----
# # 0/1 to use steepness in initial equ recruitment calculation
##-----
ctl1$Use_steep_init_equi <- 0

##-----
# future feature:
# 0/1 to make realized sigmaR a function of SR curvature
##-----
ctl1$Sigma_R_FofCurvature <- 0

#-----
#_LO_HI_INIT_PRIOR_PR_SD_PR_type_PHASE_env-var_use_dev_dev_mnyr_dev_mxyr_dev_PH_Block_Blk_Fxn_#_parm_na
#names(ctl1$SR_parms)
#row.names(ctl1$SR_parms)
#-----

##-----
#SR_LN(R0)
##-----
# t(ctl1$SR_parms[1,])
SR_parms1<-data.frame(row.names="SR_LN(R0)",
                      "LO"           = 5,
                      "HI"           = 20,
                      "INIT"          = 13,
                      "PRIOR"         = 0,
                      "PR_SD"         = 0,
                      "PR_type"       = 0,
                      "PHASE"         = 1,
                      "env_var&link"  = 0,
                      "dev_link"      = 0,
                      "dev_minyr"     = 0,
                      "dev_maxyr"     = 0,
                      "dev_PH"        = 0,
                      "Block"         = 0,
                      "Block_Fxn"     = 0)
ctl1$SR_parms[1,]<-SR_parms1
##-----
```

```

##-----
#SR_BH_steep
# t(ctl$SR_parms[2,])
##-----
SR_parms2<-data.frame(row.names="SR_SCAA_null",
                      "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$SR_parms[2,] <-SR_parms2
##-----
##-----
#SR_sigmaR
# t(ctl$SR_parms[3,])
##-----
SR_parms3<-data.frame(row.names="SR_sigmaR",
                      "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$SR_parms[3,] <-SR_parms3
##-----
##-----
#SR_regime
# t(ctl$SR_parms[4,])
##-----
SR_parms4<-data.frame(row.names="SR_regime",
                      "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$SR_parms[4,] <-SR_parms4
##-----
##-----
#SR_autocorr
# t(ctl$SR_parms[5,])
##-----
SR_parms5<-data.frame(row.names="SR_autocorr",
        "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

ctl1$SR_parms
##          LO    HI  INIT PRIOR  PR_SD PR_type PHASE env_var&link dev_link
## SR_LN(R0)   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(R0)          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

```

1.12 Desvíos de los reclutamientos

```

##-----
#_no timevary SR parameters
##-----

```

```

##-----
#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
ctl1$do_recdev <- 1

##-----
# first year of main recr_devs;
#early devs can precede this era
##-----
ctl1$MainRdevYrFirst <- 1989

##-----
# last year of main recr_devs;
#forecast devs start in following year
##-----
ctl1$MainRdevYrLast <- 2021

##-----
#_recdev phase
##-----
ctl1$recdev_phase <- 1

##-----
# (0/1) to read 13 advanced options
##-----
ctl1$recdev_adv <- 0

##-----
#_recdev_early_start
#(0=none; neg value makes relative to recdev_start)
##-----
ctl1$recdev_early_start <- NULL

##-----
#_recdev_early_phase
##-----
ctl1$recdev_early_phase <- NULL

##-----
#_forecast_recruitment phase
#(incl. late recr) (0 value resets to maxphase+1)
##-----
ctl1$Fcast_recr_phase <- NULL

```

1.12.0.1 Sesgo

```

##-----
#_lambda for Fcast_recr_like occurring before endyr+1

```

```

##-----
ctl1$lambda4Fcast_recr_like <- NULL
##-----
#_last_yr_nobias_adj_in_MPD;
#begin of ramp
##-----
ctl1$last_early_yr_nobias_adj <- NULL

##-----
#_first_yr_fullbias_adj_in_MPD;
#begin of plateau
##-----
ctl1$first_yr_fullbias_adj <- NULL
##-----
#_last_yr_fullbias_adj_in_MPD
##-----
ctl1$last_yr_fullbias_adj <- NULL
##-----
#_end_yr_for_ramp_in_MPD
#(can be in forecast to shape ramp, but SS3 sets bias_adj to 0.0 for fcast yrs)
##-----
ctl1$first_recent_yr_nobias_adj <- NULL
##-----
#_max_bias_adj_in_MPD
#(typical ~0.8;
#-3 sets all years to 0.0;
#-2 sets all non-forecast yrs w/ estimated recdevs to 1.0;
#-1 sets biasadj=1.0 for all yrs w/ recdevs)
##-----
ctl1$max_bias_adj <- NULL

```

1.12.0.2 Desvíos

```

#-----
#_period of cycles in recruitment (N parms read below)
#-----
ctl1$period_of_cycles_in_recr <- NULL
#-----
#min_rec_dev
#-----
ctl1$min_rec_dev <- NULL
#-----
#max_rec_dev
#-----
ctl1$max_rec_dev <- NULL
#-----
#_read_recdevs
#-----
ctl1$N_Read_recdevs <- NULL

```

1.13 Mortalidad por pesca

```
#-----  
#Fishing Mortality info  
#-----  
  
#-----  
# F ballpark value in units of annual_F  
#-----  
ctl1$F_ballpark <- 0.1  
#-----  
# F ballpark year (neg value to disable)  
#-----  
ctl1$F_ballpark_year <- -1989  
#-----  
# 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)  
ctl1$F_Method <- 3  
#-----  
# max F (methods 2-4) or harvest fraction (method 1)  
#-----  
ctl1$maxF <- 4  
#-----  
# N iterations for tuning in hybrid mode;  
# recommend 3 (faster) to 5 (more precise if many fleets)  
#-----  
ctl1$F_iter <- 3
```

1.14 Capturabilidad

1.14.1 Opciones de capturabilidad

```
#-----
#_Q_setup for fleets with cpue or survey data
#-----
#_1:  fleet number
#_2:  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)
#_3:  extra input for link, i.e. mirror fleet# or dev index number
#_4:  0/1 to select extra sd parameter
#_5:  0/1 for biasadj or not
#_6:  0/1 to float
#_fleet_link_link_info_extra_se_biasadj_float #  fleetname
#-----

#-----
#t(ctl1$Q_options[1,])
Q_options1<-data.frame(row.names="PELAGO",
                       "fleet"      = 2,
                       "link"       = 1,
                       "link_info"  = 0,
                       "extra_se"   = 0,
                       "biasadj"    = 0,
                       "float"      = 0)

#-----
ctl1$Q_options[1,]<-Q_options1

#-----
#t(ctl1$Q_options[2,])
Q_options2<-data.frame(row.names="ECOCADIZ",
                       "fleet"      = 3,
                       "link"       = 1,
                       "link_info"  = 0,
                       "extra_se"   = 0,
                       "biasadj"    = 0,
                       "float"      = 0)

#-----
ctl1$Q_options[2,]<-Q_options2
```

1.14.2 Parámetros

```
#-----
#_Q_parms(if_any);
#_Qunits_are_ln(q)
#-----
#_LO_HI_INIT_PRIOR_PR_SD_PR_type_PHASE_env-var_use_dev_dev_mnyr_dev_mxyr_dev_PH_Block_Blk_Fxn#parm_name
#-----
Q_parms1<-data.frame(row.names="LnQ_base_PELAGO(2)",
                     "LQ"          = -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$Q_parms[1,]<-Q_parms1
#-----
Q_parms2<-data.frame(row.names="LnQ_base_ECOCADIZ(3)",
        "LO"            = -30,
        "HI"            = 15,
        "INIT"          = 0.08918,
        "PRIOR"         = 0,
        "PR_SD"         = 1,
        "PR_type"       = 0,
        "PHASE"         = 1,
        "env_var&link"  = 0,
        "dev_link"      = 0,
        "dev_minyr"     = 0,
        "dev_maxyr"     = 0,
        "dev_PH"        = 0,
        "Block"         = 0,
        "Block_Fxn"     = 0)
#-----
ctl1$Q_parms[2,]<-Q_parms2
#-----

```

1.15 Selectividad

1.15.1 tipos de selectividad a la talla

```
#-----
#_size_selex_patterns
#-----
#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
#-----
#_Pattern Discard Male Special
#-----
size_selex_types1<-data.frame(row.names="Fishery",
                             "Pattern" = 1,
                             "Discard" = 0,
                             "Male"    = 0,
                             "Special" = 0)
ctl1$size_selex_types[1,]<-size_selex_types1
#-----
size_selex_types2<-data.frame(row.names="PELAGO",
                              "Pattern" = 1,
                              "Discard" = 0,
                              "Male"    = 0,
                              "Special" = 0)
ctl1$size_selex_types[2,]<-size_selex_types2
#-----
```

```
size_selex_types3<-data.frame(row.names="ECOCADIZ",
                             "Pattern" = 1,
                             "Discard" = 0,
                             "Male"    = 0,
                             "Special" = 0)
ctl1$size_selex_types[3,]<-size_selex_types3
#-----
```

1.15.2 tipos de selectividad a la edad

```
#-----
#_age_selex_patterns
#-----
#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
#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
#_Pattern Discard Male Special
#-----
age_selex_types1<-data.frame(row.names="Fishery",
                             Pattern = 12,
                             Discard = 0,
                             Male    = 0,
                             Special = 0)
ctl1$age_selex_types[1,]<-age_selex_types1
#-----
age_selex_types2<-data.frame(row.names="PELAGO",
                             Pattern = 12,
                             Discard = 0,
                             Male    = 0,
                             Special = 0)
ctl1$age_selex_types[2,]<-age_selex_types2
#-----
age_selex_types3<-data.frame(row.names="ECOCADIZ",
                             Pattern = 12,
                             Discard = 0,
                             Male    = 0,
                             Special = 0)
ctl1$age_selex_types[3,]<-age_selex_types3
```

1.15.3 parametros de selectividad a la talla

```
#
#_LO_HI_INIT_PRIOR_PR_SD_PR_type_PHASE_env-var_use_dev-dev_mnyr-dev_maxyr-dev_PH-Block-Blk_Fxn#parm_name
#-----
# 1 FISHERY LenSelex
#-----
#t(ctl1$size_selex_parms[1,])
size_selex_parms1<-data.frame(row.names="SizeSel_P_1_Fishery(1)",
                              "LO"          = -1,
                              "HI"          = 20,
                              "INIT"         = 12,
                              "PRIOR"        = 0,
                              "PR_SD"        = 0,
                              "PR_type"      = 0,
                              "PHASE"        = 2,
                              "env_var&link" = 0,
                              "dev_link"     = 0,
                              "dev_mnyr"     = 0,
                              "dev_maxyr"    = 0,
                              "dev_PH"       = 0.5,
                              "Block"        = 0,
                              "Block_Fxn"    = 0)
ctl1$size_selex_parms[1,<-size_selex_parms1
#-----
#t(ctl1$size_selex_parms[2,])
size_selex_parms2<-data.frame(row.names="SizeSel_P_1_Fishery(1)",
                              "LO"          = -1,
                              "HI"          = 20,
                              "INIT"         = 18,
                              "PRIOR"        = 0,
                              "PR_SD"        = 0,
                              "PR_type"      = 0,
                              "PHASE"        = 2,
                              "env_var&link" = 0,
                              "dev_link"     = 0,
                              "dev_mnyr"     = 0,
                              "dev_maxyr"    = 0,
                              "dev_PH"       = 0.5,
                              "Block"        = 0,
                              "Block_Fxn"    = 0)
ctl1$size_selex_parms[2,<-size_selex_parms2
#-----
#t(ctl1$size_selex_parms[3,])
size_selex_parms3<-data.frame(row.names="SizeSel_P_1_PELAGO(2)",
                              "LO"          = -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$size_selex_parms[3,]<-size_selex_parms3
#-----
#t(ctl1$size_selex_parms[4,])
size_selex_parms4<-data.frame(row.names="SizeSel_P_2_PELAGO(2)",
        "LO"          = -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$size_selex_parms[4,]<-size_selex_parms4
#-----
#t(ctl1$size_selex_parms[5,])
size_selex_parms5<-data.frame(row.names="SizeSel_P_1_ECOCADIZ(3)",
        "LO"          = -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$size_selex_parms[5,]<-size_selex_parms5
#-----
#t(ctl1$size_selex_parms[6,])
size_selex_parms6<-data.frame(row.names="SizeSel_P_2_ECOCADIZ(3)",
        "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

```

1.15.4 parametros de selectividad a la edad

```

#t(ctl1$age_selex_parms[1,])
#-----
age_selex_parms1<-data.frame(row.names="AgeSel_P_1_Fishery(1)",
        "LO"           = -2.0,
        "HI"           = 5.50,
        "INIT"         = 0.10,
        "PRIOR"        = 0,
        "PR_SD"        = 0.01,
        "PR_type"      = 0,
        "PHASE"        = -1,
        "env_var&link"  = 0,
        "dev_link"     = 0,
        "dev_minyr"    = 0,
        "dev_maxyr"    = 0,
        "dev_PH"       = 0.5,
        "Block"        = 0,
        "Block_Fxn"    = 0)
ctl1$age_selex_parms[1,]<-age_selex_parms1
#-----
#t(ctl1$age_selex_parms[2,])
#-----
age_selex_parms2<-data.frame(row.names="AgeSel_P_2_Fishery(1)",
        "LO"           = -1.0,
        "HI"           = 5.50,
        "INIT"         = 5.0,
        "PRIOR"        = 0,
        "PR_SD"        = 0.01,
        "PR_type"      = 0,
        "PHASE"        = -1,
        "env_var&link"  = 0,
        "dev_link"     = 0,
        "dev_minyr"    = 0,
        "dev_maxyr"    = 0,
        "dev_PH"       = 0.5,
        "Block"        = 0,
        "Block_Fxn"    = 0)
ctl1$age_selex_parms[2,]<-age_selex_parms2
#-----
#t(ctl1$age_selex_parms[3,])
#-----
age_selex_parms3<-data.frame(row.names="AgeSel_P_1_PELAGO(2)",
        "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$age_selex_parms[3,]<-age_selex_parms3
#-----
#t(ctl1$age_selex_parms[4,])
#-----
age_selex_parms4<-data.frame(row.names="AgeSel_P_2_PELAGO(2)",
        "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$age_selex_parms[4,]<-age_selex_parms4
#-----
#t(ctl1$age_selex_parms[5,])
#-----
age_selex_parms5<-data.frame(row.names="AgeSel_P_1_ECOCADIZ(3)",
        "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$age_selex_parms[5,]<-age_selex_parms5
#-----
#t(ctl1$age_selex_parms[6,])

```

```

#-----
age_selex_parms6<-data.frame(row.names="AgeSel_P_2_ECOCADIZ(3)",
                             "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

```

1.15.5 otros parámetros

```

#_No_Dirichlet parameters
#_no timevary selex parameters
#
#-----
# use 2D_AR1 selectivity(0/1)
#-----
ctl1$Use_2D_AR1_selectivity<-0
#-----
#_no 2D_AR1 selex offset used
#
#-----
# Tag loss and Tag reporting parameters go next
#-----
# TG_custom:
#-----
# 0=no read and autogen if tag data exist;
# 1=read
ctl1$TG_custom<-0
#-----
#_Cond -6 6 1 1 2 0.01 -4 0 0 0 0 0 0 #_placeholder if no parameters
#
# no timevary parameters
#
#
#-----
# Input variance adjustments factors:
#-----
#_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
#_Factor Fleet Value

ctl1$DoVar_adjust<-1

#-----
varadj1<-data.frame(row.names="Variance_adjustment_list1",
                    "Factor" = 4,
                    "Fleet"  = 1,
                    "Value"  = 0.0045)
ctl1$Variance_adjustment_list[1,<-varadj1
#-----
varadj1<-data.frame(row.names="Variance_adjustment_list2",
                    "Factor" = 4,
                    "Fleet"  = 2,
                    "Value"  = 0.0051)
ctl1$Variance_adjustment_list[1,<-varadj1
#-----
varadj1<-data.frame(row.names="Variance_adjustment_list3",
                    "Factor" = 4,
                    "Fleet"  = 3,
                    "Value"  = 0.0047)
ctl1$Variance_adjustment_list[1,<-varadj1
#-----
#_maxlambdaphase
#-----
ctl1$maxlambdaphase<-1
#-----
#_sd_offset;
#-----
# must be 1 if any growthCV, sigmaR, or survey extraSD is an estimated parameter
ctl1$sd_offset<-1

```

1.15.6 Lambdas

```
ctl1$lambdas <- NULL
```

```
ctl1$N_lambdas<- 0
```

1.16 more stddev reporting

```
ctl1$more_stddev_reporting<-0
```

1.16.1 epecs

```
ctl1$stddev_reporting_specs<-NULL
```

1.16.2 selectividad

```
ctl1$stddev_reporting_selex<-NULL
```

1.16.3 crecimiento

```
ctl1$stddev_reporting_growth<-NULL
```

1.16.4 abundancia a la edad

```
ctl1$stddev_reporting_N_at_A<-NULL
```

1.16.5 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)  
#-----
```