

# Montaje de los archivos de entrada al modelo SS3, formatos requeridos y aprovechamiento de los utilizados en las entradas de a4a.

May, 08, 2023

## Contents

<b>1</b>	<b>Contexto</b>	<b>3</b>
<b>2</b>	<b>Tareas a desarrollar.</b>	<b>3</b>
2.1	Temporalización y dinámica de las actividades propuestas. . . . .	3
<b>3</b>	<b>Descripción general del modelo de merluza europea (<i>Merluccius merluccius</i>) de la GSA1 (mar de Alborán).</b>	<b>4</b>
3.1	Modelo conceptual . . . . .	4
3.2	Descripción general del modelo implementado en <b>a4a</b> . . . . .	4
<b>4</b>	<b>Revisión de los archivos requeridos para cada enfoque de modelación</b>	<b>5</b>
4.1	Archivos utilizado para enfoque de modelación <b>a4a</b> . . . . .	5
4.2	Archivos utilizado para enfoque de modelación <b>SS3</b> . . . . .	6
<b>5</b>	<b>Descripción del Formado de entrada de datos para cada enfoque de modelación</b>	<b>7</b>
5.1	Archivo data . . . . .	7
5.1.1	Información general del modelo Formato <b>a4a</b> . . . . .	7
5.1.2	Información general del modelo Formato <b>SS3</b> . . . . .	7
5.1.3	Capturas en formato <b>a4a</b> . . . . .	8
5.1.4	Capturas en formato <b>SS3</b> . . . . .	9
5.1.5	Indices de abundancia formato <b>a4a</b> . . . . .	12
5.1.6	Indices de abundancia formato <b>SS3</b> . . . . .	13
5.1.7	Descartes y tallas medias Formato <b>a4a</b> . . . . .	16
5.1.8	Descartes y tallas medias Formato <b>SS3</b> . . . . .	16
5.1.9	Composición de tallas Formato <b>a4a</b> . . . . .	17
5.1.10	Composición de tallas Formato <b>SS3</b> . . . . .	21
5.1.11	Composición de edad Formato <b>a4a</b> . . . . .	24
5.1.12	Composición de edad Formato <b>SS3</b> . . . . .	26
5.1.13	Otros datos . . . . .	30
5.1.14	Escribir archivo de datos modificado con la función <b>SS_write</b> para el enfoque de modelación <b>SS3</b> . . . . .	32
5.2	Archivo control.ss . . . . .	33
5.2.1	Especificaciones iniciales . . . . .	33
5.2.2	Datos de los archivos . . . . .	33
5.2.3	Especificaciones del crecimiento . . . . .	33
5.2.4	Distribución del reclutamiento . . . . .	34
5.2.5	Bloques . . . . .	35
5.2.6	Parámetros que varían en el tiempo . . . . .	35
5.2.7	Parámetros biológicos . . . . .	36
5.2.8	Parámetros biológicos . . . . .	40
5.2.9	Relación stock recluta . . . . .	54

5.2.10	Desvíos de los reclutamientos . . . . .	56
5.2.11	Mortalidad por pesca . . . . .	60
5.2.12	Capturabilidad . . . . .	62
5.2.13	Selectividad . . . . .	65
5.2.14	Lambdas . . . . .	73
5.2.15	more stddev reporting . . . . .	75
5.2.16	Escribir archivo de control modificado con la función <b>SS_write</b> para el enfoque de modelación <b>SS3</b> . . . . .	77
5.3	Archivo starter.ss . . . . .	78
5.3.1	Escribir archivo de starter modificado con la función <b>SS_write</b> para el enfoque de modelación <b>SS3</b> . . . . .	82
5.4	Archivo forecast.ss . . . . .	83
5.4.1	Escribir archivo de forecast modificado con la función <b>SS_write</b> para el enfoque de modelación <b>SS3</b> . . . . .	88
5.5	Archivo watatege . . . . .	89
5.5.1	Formato <b>a4a</b> . . . . .	89
5.5.2	Formato <b>SS3</b> . . . . .	93
<b>6</b>	<b>GSA6 (Levante-Norte de España).</b>	<b>100</b>
6.1	Descripción del modelo . . . . .	100
6.2	Archivos utilizado para enfoque de modelación <b>a4a</b> . . . . .	101

# 1 Contexto

Se pretende desarrollar un modelado con Stock Synthesis de los stocks de merluza europea (*Merluccius merluccius*) de la GSA1 (mar de Alborán) y GSA6 (Levante-Norte de España). Ambos stocks se encuentran definidos en el GFCM en el Mediterráneo español y llevan varios años siendo evaluados con metodología analítica SCAA utilizando el modelo **a4a**, con evaluaciones validadas y aprobadas en el GFCM en los últimos años. Se trabajaría con la serie histórica actualizada en el último WGSAD de diciembre de 2022 (2002-2021).

Los dos objetivos principales serían por un lado Intentar testar el paso de un modelo **SCAA** a uno integrado en el entorno **GFCM** con stocks españoles, algo hasta el momento inédito, y por otro lado presentar los resultados al próximo Simposio de Modelado y Evaluación de Recursos Pesqueros (SIMERPE II) que se celebrará en Cádiz desde el 24 al 27 de octubre de 2023.

## 2 Tareas a desarrollar.

Dentro de la propuesta se han identificado las posibles tareas a desarrollar que serían por orden cronológico:

1. Montaje de los archivos de entrada al modelo SS3, formatos requeridos y aprovechamiento si fuera posible de los utilizados en las entradas de a4a.
2. Identificar tareas adicionales que requieran una recopilación de datos de entrada extra o con formatos diferentes a los usados en a4a.
3. Puesta a punto de ficheros de entrada SS3.
4. Primeras rodadas tentativas.
5. Análisis de resultados. Comparación con salidas XSA y a4a.
6. Presentación de los resultados en el SIMERPE II
7. Estudio de la viabilidad y/o idoneidad de estructurar la evaluación con estos stocks usando Stock Synthesis para los grupos de trabajo del GFCM el año que viene (WGSAD 2023).

### 2.1 Temporalización y dinámica de las actividades propuestas.

El flujo de trabajo tendría tres fases bien definidas, una inicial en la cual de forma telemática se completarían los puntos 1, 2 y 3 de la propuesta. Se propone completar esta fase antes del 31 de marzo de 2023.

En una segunda fase de desarrollo, se realizarían las primeras rodadas tentativas y se adquirirían las competencias necesarias para rodar el modelo de forma autónoma, al menos en sus requerimientos más básicos. Esta fase se realizaría en el C.O de Cádiz y Encarnación García y José Luis Pérez se desplazarían a este centro para tal fin para trabajar con María José Zúñiga bajo la supervisión de Margarita Rincón. Esta segunda fase tendría que estar completada el 30 de abril.

En una última fase se analizarían telemáticamente los resultados obtenidos y la viabilidad e idoneidad del paso de a4a a Stock Synthesis para los dos stocks evaluados. Esta fase tendría dos fechas diferenciadas, una a final de mayo para confeccionar y enviar los resultados al SIMERPE II y otra en noviembre para dejar lista la evaluación de ambos stocks para su posible presentación en el WGSAD-GFCM en diciembre de 2023.

### 3 Descripción general del modelo de merluza europea (*Merluccius merluccius*) de la GSA1 (mar de Alborán).

#### 3.1 Modelo conceptual

La conceptualización del modelo biológico de **GSA1** considera los siguientes componentes de la dinámica poblacional:

- Estructura geográfica
- Reproducción
- Reclutamiento
- Mortalidad natural
- Crecimiento
- otros...

#### 3.2 Descripción general del modelo implementado en **a4a**

El modelo de evaluación de stock de **GSA1** se basa en un análisis estadístico de la dinámica de estructuras de edad anual que incorpora información biológica y pesquera. La información que ingresa al modelo consiste en ...

- Años de evaluación de stock = 2003\_2021
- número de edades = 6 años ?

## 4 Revisión de los archivos requeridos para cada enfoque de modelación

### 4.1 Archivos utilizado para enfoque de modelación a4a

El modelo SCAA a4a está implementado en FLR ([https://flr-project.org/doc/Loading\\_your\\_data\\_into\\_FLR.html](https://flr-project.org/doc/Loading_your_data_into_FLR.html)), por lo tanto la arquitectura de sus ficheros input sigue se estructura en torno a la clase FLStock con los siguientes slots:

- Catch
- catch.n
- catch.wt
- discards
- discards.n
- discards.wt
- landings
- landings.n
- landings.wt
- stock
- stock.n
- stock.wt
- m
- mat
- harvest
- harvest.spwn
- m.spwn

No obstante, para facilitar la importación de los archivos input, usamos ficheros con formato .DAT que mediante el archivo índice “LOWIND.DAT” y la sentencia “stk <- readFLStock(“LOWIND.DAT”)”, nos lee nuestros ficheros creándonos el objeto FLStock.

A continuación describimos brevemente cada uno de los archivos .DAT que usamos como inputs:

- CATCH: Capturas anuales del stock (En nuestro caso GSA correspondiente). Toneladas.
- CATNUM: Matriz de número de individuos por edad/año de las capturas. (Miles de individuos).
- CATWT: Peso medio por edad y año de tu matriz de captura. (Kilos).
- STOCWT: medio por edad y año asumida para el stock (Normalmente = CATWT). (Kilos).
- TUNEFF: Fichero con los índices de abundancia para calibrar (CPUEs o Campañas, o ambas). Normalmente para índices de campaña empleamos número/KM2.
- NATMOR: Vector de mortalidad Natural por edad.
- PROPMAT: Ogiva de madurez por edad.
- PROPM: Factor de proporcionalidad de M antes de la puesta. Sería el slot m.spwn. Normalmente computamos 0.5 si la puesta es alrededor de Junio.
- PROPF: Factor de proporcionalidad de F antes de la puesta. Sería el slot harvest.spwn. Normalmente computamos 0.5 si la puesta es alrededor de Junio.
- LOWIND: Fichero índice.
- INCHECK: Este no es un fichero input propiamente dicho. Se trata del fichero resumen tras realizar el slicing con el programa L2age. Se recogen aquí el set de parámetros de crecimiento utilizados.

```
dir_GSA1 <- here("hke-GSA1-a4a_format")
dir(dir_GSA1)
## [1] "CATCH.DAT"
## [2] "CATNUM.DAT"
## [3] "CATWT.DAT"
## [4] "Definiciones_Ficheros_imput_a4a.docx"
## [5] "INCHECK.TXT"
```

```
## [6] "LOWIND.DAT"
## [7] "NATMOR.DAT"
## [8] "PROPF.DAT"
## [9] "PROPM.DAT"
## [10] "PROPMAT.DAT"
## [11] "STOCWT.DAT"
## [12] "TUNEFF.DAT"
```

## 4.2 Archivos utilizado para enfoque de modelación SS3

1. Identificamos el directorio donde se encuentra el modelo base simple

```
dirname.base <- here("simple")
```

3. Creamos un nuevo directorio para la nueva versión del modelo modificado

```
dirname.simple_mod <- here("GSA6_SS3")
dir.create(path=dirname.simple_mod, showWarnings = TRUE, recursive = TRUE)
```

5. Copiamos los archivos para el modelo que vamos a modificar

```
copy_SS_inputs(dir.old = dirname.base,
               dir.new = dirname.simple_mod,
               copy_exe = TRUE,
               verbose = FALSE)
## [1] FALSE
```

## 5 Descripción del Formado de entrada de datos para cada enfoque de modelación

### 5.1 Archivo data

#### 5.1.1 Información general del modelo Formato a4a

Buscar esta información en archivos entregados!!!

#### 5.1.2 Información general del modelo Formato SS3

En la parte superior se especifica información general del modelo: los años del modelo, número de temporadas, número de sexos, edad máxima, número de áreas, número de flotas

[Consulte la Guía de usuario de SS3: Sección 7.5 “Model Dimensions”](#).

Revisamos los nombres de los componentes de la lista del archivo .dat

```
dat <- r4ss::SS_readdat(here(dirname.base,"data.ss")) #base
dat1<-dat # para modificar
#names(dat1) # muestra los objetos de la lista

#Especificaciones iniciales
dat1$styr <-2003 #_StartYr
dat1$endyr <-2021 #_EndYr
dat1$nseas <-1 #_Nseas
dat1$months_per_seas<-12 #_months/season
dat1$Nsubseasons <-2 #_Nsubseasons (even number, minimum is 2)
dat1$spawn_month <-6 #_spawn_month (puesta alrededor de junio)
dat1$Ngenders <-1 #_Ngenders: 1, 2, -1 (use -1 for 1 sex setup with SSB
# multiplied by female_frac parameter)

dat1$Nsexes <-1
dat1$Nages <-6 #_Nages=accumulator age, first age is always age 0
dat1$N_areas <-1 #_Nareas
dat1$Nfleets <-4 #_Nfleets (including surveys)
```

### 5.1.3 Capturas en formato a4a

Capturas anuales del stock (toneladas)

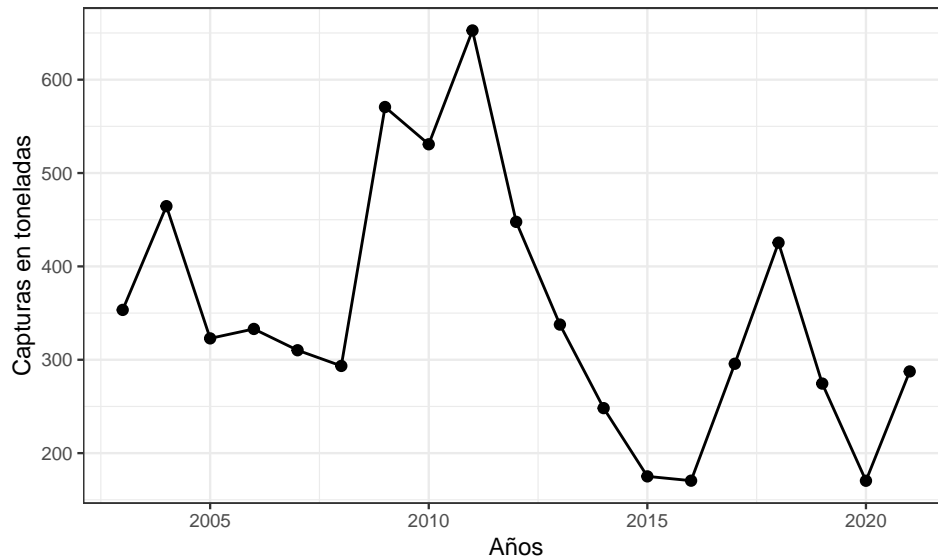
```
dir(dir_GSA1)[1]
## [1] "CATCH.DAT"
CATCH.DAT <- read.table(paste(dir_GSA1,dir(dir_GSA1)[1],sep="/"),
                        header=T,sep=" ",na="NA",fill=T,skip = 4)

CATCH.DAT
##      X5
## 1  353.4
## 2  464.5
## 3  322.9
## 4  333.0
## 5  310.2
## 6  293.4
## 7  570.7
## 8  530.8
## 9  652.7
## 10 447.7
## 11 337.7
## 12 248.2
## 13 175.1
## 14 170.5
## 15 295.7
## 16 425.4
## 17 274.5
## 18 170.4
## 19 287.5
```

```
catchdat<-as.data.frame(CATCH.DAT) %>%
  mutate(Yr=2003:2021)

ggplot(catchdat,aes(x=Yr,y=X5))+
  geom_line()+
  geom_point()+
  labs(x="Años", y="Capturas en toneladas") +
  theme_bw(base_size=9)
```





### 5.1.4 Capturas en formato SS3

Primero ingresamos las especificaciones de los Datos de captura de la flota

[Consulte la Guía de usuario de SS3: Sección 7.9 “Catch”.](#)

#### 5.1.4.1 Especificaciones de datos de captura

```
#-----
#_fleet_type: 1=catch fleet; 2=bycatch only fleet; 3=survey; 4=ignore
#_sample_timing: -1 for fishing fleet to use season-long catch-at-age for
#observations, or 1 to use observation month; (always 1 for surveys)
#_fleet_area: area the fleet/survey operates in
#_units of catch: 1=bio; 2=num (ignored for surveys; their units read later)
#_catch_mult: 0=no; 1=yes
#_rows are fleets
#_fleet_type fishery_timing area catch_units need_catch_mult fleetname
#-----

# Arreglo de datos
fleetnames1<-c("FISHERY", "SURVEY1", "SURVEY2","SURVEY3")
type1<-c(1,3,3,3)
surveytiming1<-c(-1,1,1,1)
units_of_catch1<-c(1,2,2,2)
areas1<-c(1,1,1,1)
need_catch_mult1<-c(0,0,0,0)
#-----

# crear data.frame
fleetinfo1<-data.frame(type = type1,
                      surveytiming =surveytiming1,
                      area=areas1,
                      units=units_of_catch1,
                      need_catch_mult =need_catch_mult1,
                      fleetname=fleetnames1)
#-----

dat1$fleetinfo<-fleetinfo1
dat1$fleetinfo
```

```
##      type surveytiming area units need_catch_mult fleetname
## 1      1           -1    1      1                0  FISHERY
## 2      3            1    1      2                0  SURVEY1
## 3      3            1    1      2                0  SURVEY2
## 4      3            1    1      2                0  SURVEY3
#-----
```

#### 5.1.4.2 Datos de captura

```
#-----
#_Catch data: yr, seas, fleet, catch, catch_se
#_catch_se: standard error of log(catch)
#_NOTE: catch data is ignored for survey fleets
#-----

# Arreglo de Datos
year<-2003:2021
nyear<-length(year)
catch_year<-c(-999,year)
catch_seas<-rep(1,nyear+1)
catch_fleet<-rep(1,nyear+1)
catch_catch<-c(0,CATCH.DAT$X5)
catch_catch_se<-rep(0.01,nyear+1) # se asume cv = 0.01 Revisar!!!!
#-----

# crear data.frame
catch1<-data.frame(year=catch_year,
                   seas=catch_seas,
                   fleet=catch_fleet,
                   catch=catch_catch,
                   catch_se=catch_catch_se)

#-----

dat1$catch<-catch1
dat1$catch
##      year seas fleet catch catch_se
## 1   -999    1    1   0.0   0.01
## 2   2003    1    1 353.4   0.01
## 3   2004    1    1 464.5   0.01
## 4   2005    1    1 322.9   0.01
## 5   2006    1    1 333.0   0.01
## 6   2007    1    1 310.2   0.01
## 7   2008    1    1 293.4   0.01
## 8   2009    1    1 570.7   0.01
## 9   2010    1    1 530.8   0.01
## 10  2011    1    1 652.7   0.01
## 11  2012    1    1 447.7   0.01
## 12  2013    1    1 337.7   0.01
## 13  2014    1    1 248.2   0.01
## 14  2015    1    1 175.1   0.01
## 15  2016    1    1 170.5   0.01
## 16  2017    1    1 295.7   0.01
## 17  2018    1    1 425.4   0.01
## 18  2019    1    1 274.5   0.01
## 19  2020    1    1 170.4   0.01
## 20  2021    1    1 287.5   0.01
#-----
```

- La primera línea del fragmento de código anterior muestra los encabezados de columna para los datos de captura.
- Tenga en cuenta que toda la captura proviene de la pesquería. La línea `-999 1 1 0 0.01` especifica la captura de equilibrio para los años anteriores al inicio del modelo; en este caso, no hay captura de equilibrio porque la columna de captura es 0.

### 5.1.5 Índices de abundancia formato a4a

Índices de abundancia para calibrar (CPUEs o Campañas, o ambas). Normalmente para índices de campaña empleamos número/KM2.

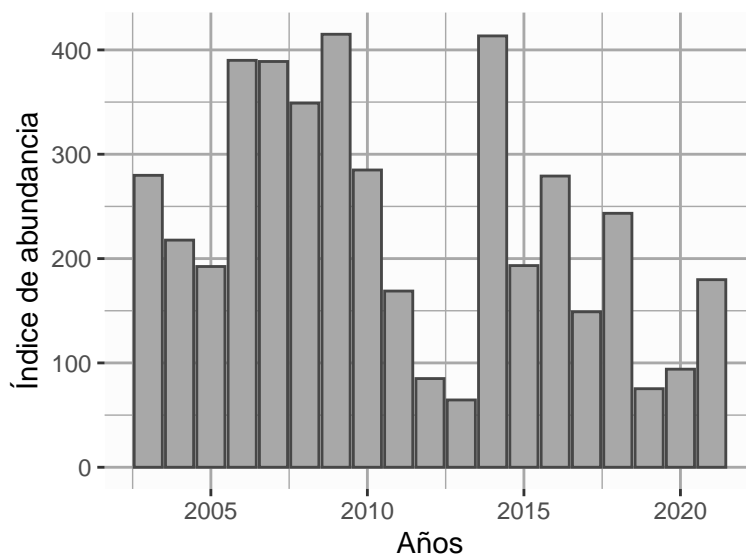
```
dir(dir_GSA1)[12]
## [1] "TUNEFF.DAT"
TUNEFF.DAT<-read.table(paste(dir_GSA1,dir(dir_GSA1)[12],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 6)
```

```
TUNEFF.DAT
##      V1      V2      V3      V4 V5 V6      V7
## 1  1 242.2  33.6  2.7 0.2 0.1 0.001
## 2  1 187.9  25.2  3.3 0.2 0.1 0.001
## 3  1 168.1  20.2  2.8 0.2 0.1 0.001
## 4  1 350.2  34.7  3.6 0.3 0.2 0.001
## 5  1 356.6  27.9  2.0 1.2 0.2 0.001
## 6  1 306.3  36.5  4.5 0.6 0.1 0.001
## 7  1 329.6  78.8  5.2 0.3 0.1 0.001
## 8  1 147.0 125.6 11.0 0.2 0.1 0.001
## 9  1 116.9  47.7  3.0 0.2 0.1 0.001
## 10 1  62.9  17.4  2.9 0.5 0.3 0.001
## 11 1  38.4  22.3  2.5 0.2 0.1 0.001
## 12 1 383.7  23.1  3.1 1.7 0.8 0.001
## 13 1 161.7  24.9  4.9 0.6 0.2 0.001
## 14 1 238.7  37.8  1.3 0.2 0.1 0.001
## 15 1  80.7  64.4  2.4 0.3 0.2 0.001
## 16 1 143.9  96.3  1.7 0.4 0.1 0.001
## 17 1  28.6  42.5  2.6 0.5 0.1 0.001
## 18 1  50.6  40.7  1.3 0.3 0.1 0.001
## 19 1 104.9  71.7  1.9 0.2 0.1 0.001
```

```
names_ind<-c('index_1','Yr')
```

```
indeexdat<-data.frame(Yr=2003:2021,suma=rowSums(TUNEFF.DAT))
```

```
ggplot() +
  geom_bar(data=indeexdat, aes(x=Yr, y =suma),
           stat="identity", fill='gray66', color = 'gray28') +
  labs(x="Años", y="Índice de abundancia") +
  theme(panel.background = element_rect(fill ="gray99")) +
  theme(panel.grid=element_line(color="gray66"))
```



### 5.1.6 Índices de abundancia formato SS3

Luego viene la especificación de los índices de abundancia. Primero está la configuración para todas las flotas. Consulte la Guía de usuario de SS3: Sección 7.10 “Índices”.

#### 5.1.6.1 Especificaciones de los índices de abundancia

```
#-----
#_CPUE_and_surveyabundance_observations
#_Units: 0=numbers; 1=biomass; 2=F; >=30 for special types
#_Errtype: -1=normal; 0=lognormal; >0=T
#_SD_Report: 0=no sdreport; 1=enable sdreport
#_Fleet Units Errtype SD_Report
#-----

# Arreglo de datos
CPUEinfo_Fleet<-c(1,2)
CPUEinfo_Units<-c(1,1) # unidades la dejamos en 1=biomass Revisar!!!
CPUEinfo_Errtype<-c(0,0) # en general se trabaja lognormal
CPUEinfo_SD_Report<-c(0,0) # esto se puede cambiar después si se necesita
                        # SD_Report, por ahora no es necesario...

CPUEinfo_names<-c("FISHERY", "SURVEY1")
#-----

# crear data.frame
CPUEinfo1<-data.frame(Fleet=CPUEinfo_Fleet,
                      Units=CPUEinfo_Units,
                      Errtype=CPUEinfo_Errtype,
                      SD_Report=CPUEinfo_SD_Report)

row.names(CPUEinfo1)<-CPUEinfo_names
#-----

dat1$CPUEinfo<-CPUEinfo1
dat1$CPUEinfo
##           Fleet Units Errtype SD_Report
## FISHERY      1      1        0         0
## SURVEY1      2      1        0         0
```

```
#-----
```

- Los encabezados de las columnas de esta sección están directamente encima de los números. Tenga en cuenta que aquí se definen todas las flotas (es decir, cada flota necesita una línea), incluida la pesquería, y se enumeran en el mismo orden que cuando se especificaron los tipos de flota.
- Lo más importante en esta sección es que se especifican las unidades y el tipo de error que se utilizará al leer los índices de abundancia.
- En este caso, la pesquería y las campañas tienen unidades de biomasa. Revisar que pasa si lo cambiamos a número. Se asume un error logarítmico normal para las 3 flotas.
- Inmediatamente después de su encabezado, se incluyen los datos de índices de abundancia:

### 5.1.6.2 Datos de índices de abundancia

```
#-----
```

```
# Arreglo de datos
yearf1<-2003:2021
CPUE_year<-c(yearf1)
#CPUE_seas = fecha e las campañas, se asume 1 (enero),
#corregir por el mes correspondiente
CPUE_seas<-rep(1,length(CPUE_year))
# Los números de "CPUE_index" son los mismos números de "Fleet"
# que se especifican en "CPUEinfo"
CPUE_index<-c(rep(2,length(yearf1)))
CPUE_obs<-c(TUNEFF.DAT$V2)
CPUE_se_log<-c(TUNEFF.DAT$V5)
```

```
#-----
```

```
# crear data.frame
CPUE1<-data.frame(year=CPUE_year,
                  seas=CPUE_seas,
                  index=CPUE_index,
                  obs=CPUE_obs,
                  se_log=CPUE_se_log)
```

```
#-----
```

```
dat1$CPUE<-CPUE1
dat1$CPUE
##   year seas index  obs se_log
## 1  2003    1    2 242.2   0.2
## 2  2004    1    2 187.9   0.2
## 3  2005    1    2 168.1   0.2
## 4  2006    1    2 350.2   0.3
## 5  2007    1    2 356.6   1.2
## 6  2008    1    2 306.3   0.6
## 7  2009    1    2 329.6   0.3
## 8  2010    1    2 147.0   0.2
## 9  2011    1    2 116.9   0.2
## 10 2012    1    2  62.9   0.5
## 11 2013    1    2  38.4   0.2
## 12 2014    1    2 383.7   1.7
## 13 2015    1    2 161.7   0.6
## 14 2016    1    2 238.7   0.2
## 15 2017    1    2  80.7   0.3
## 16 2018    1    2 143.9   0.4
## 17 2019    1    2  28.6   0.5
```

## 18 2020	1	2	50.6	0.3
## 19 2021	1	2	104.9	0.2
#-----				

### 5.1.7 Descartes y tallas medias Formato a4a

En este modelo no se ingresan datos de estructuras de tallas.

### 5.1.8 Descartes y tallas medias Formato SS3

A continuación, se podrían especificar los datos de descartes y tallas media.

[Consulte la Guía de usuario de SS3: Sección 7.11 “Discard”](#).

#### 5.1.8.1 Descarte

```
dat1$N_discard_fleets<-0 #_N_fleets_with_discard
#-----
#_discard_units (1=same_as_catchunits(bio/num);
#                2=fraction;
#                3=numbers)
#_discard_errtype: >0 for DF of T-dist(read CV below);
#                  0 for normal with CV;
#                  -1 for normal with se;
#                  -2 for lognormal;
#                  -3 for trunc normal with CV
# note: only enter units and errtype for fleets with discard
# note: discard data is the total for an entire season, so input of month
#       here must be to a month in that season
#_Fleet units errtype
#-----
# -9999 0 0 0.0 0.0 # terminator for discard data
```

#### 5.1.8.2 Tallas medias

```
dat1$use_meanbodywt<-0 #_use meanbodysize_data (0/1)
#-----
#_COND_0 #_DF_for_meanbodysize_T-distribution_like
# note: type=1 for mean length; type=2 for mean body weight
#_yr month fleet part type obs stderr
#-----
# -9999 0 0 0 0 0 0 # terminator for mean body size data
```



### 5.1.9 Composición de tallas Formato a4a

```
excel<-here("Indice_abundancia_tallas_meditis_GSA1_hake.xlsx")

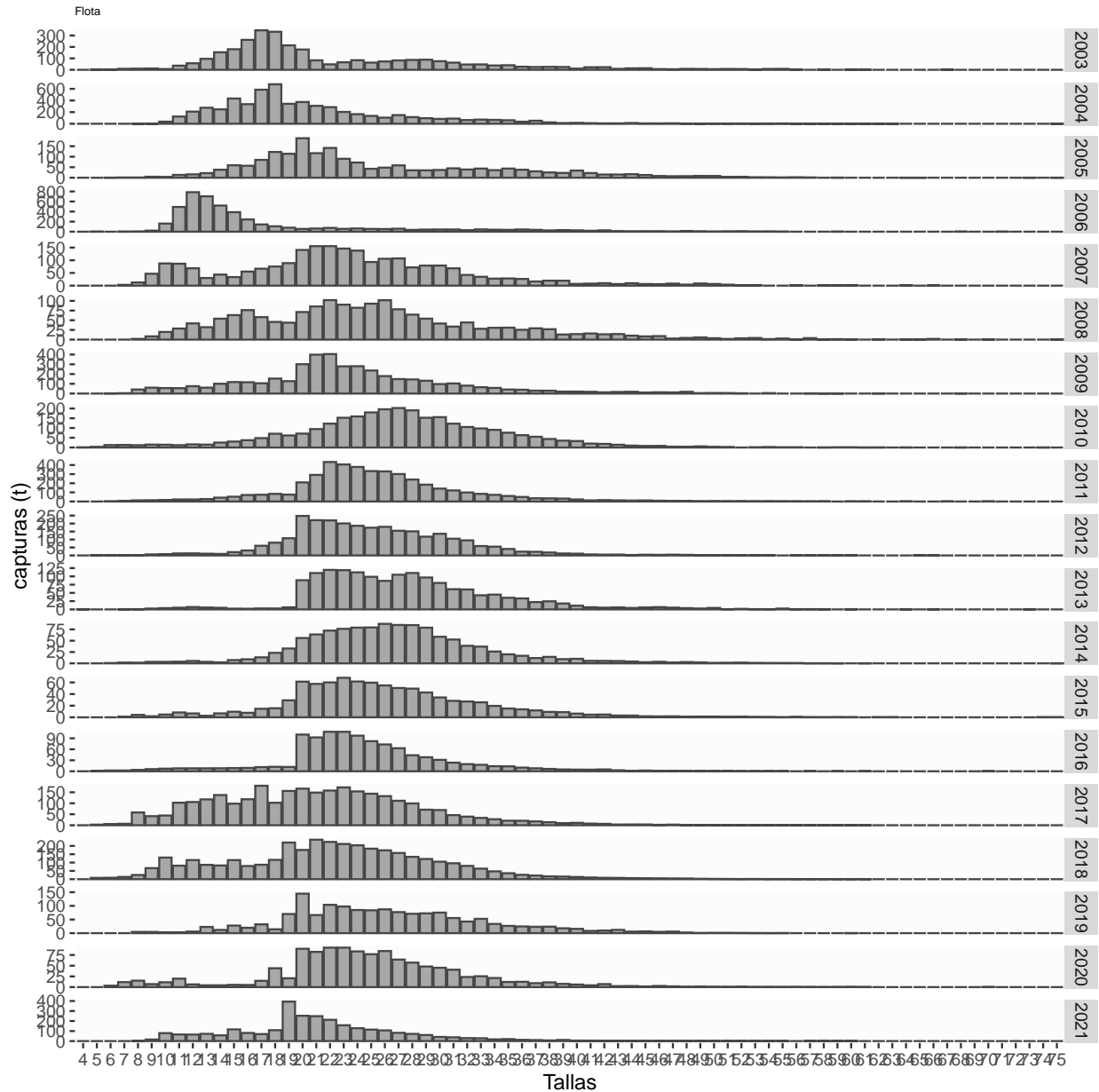
comptallas<-read_excel(excel, sheet = "catch",col_names= TRUE,col_types=NULL,na="",skip= 0)

yearcatch<-seq(2003,2021,1)
tallas<-seq(4,75,1)
comptallas<-comptallas[2:73,2:20]

CompTallas<-data.frame(row.names = NULL,year=yearcatch,t(comptallas))
colnames(CompTallas)<-c("year",tallas)

plot_tallas<-CompTallas %>% melt(id.vars=c("year"))

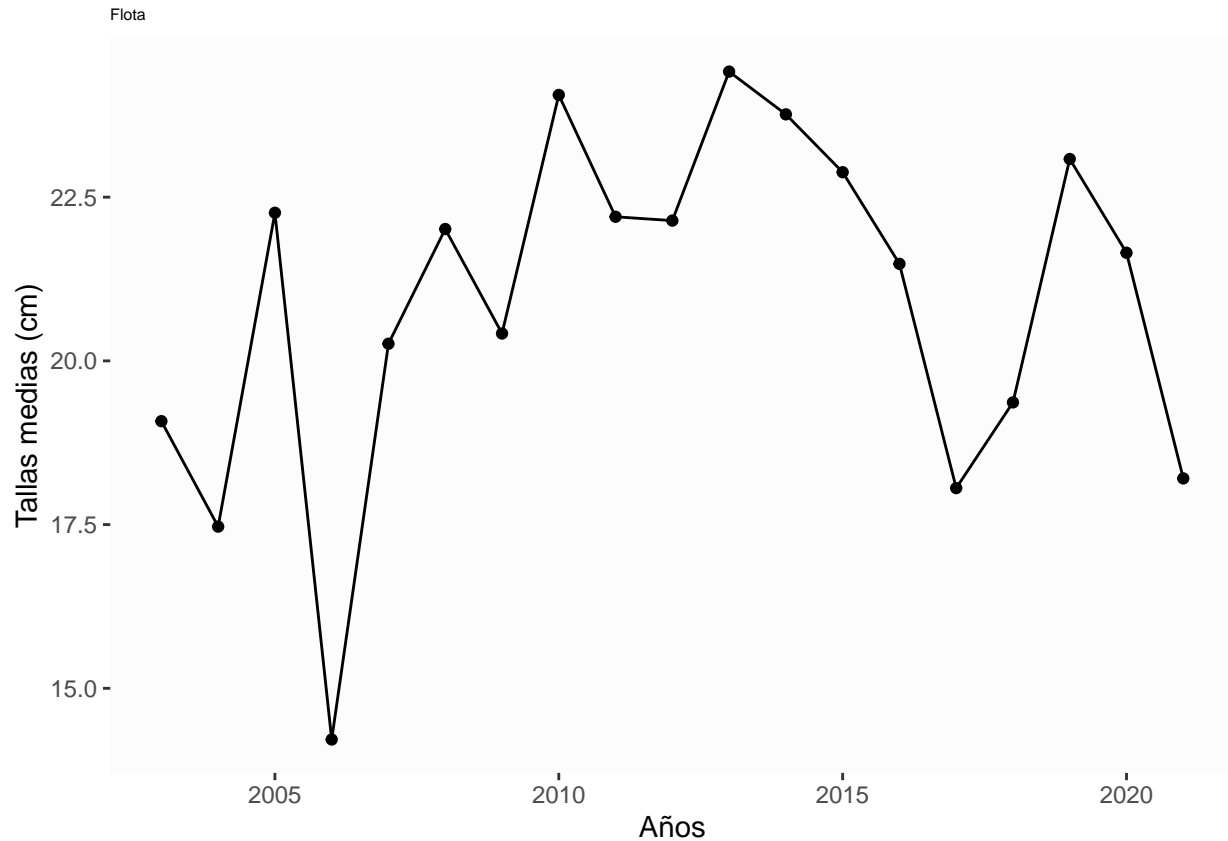
#-----
ggplot(plot_tallas) +
  geom_bar(aes(x = variable, y = value), stat="identity", fill='gray66', color = 'gray28') +
  facet_grid(rows = vars(year), as.table = TRUE,scales = "free") +
  labs(x = 'Tallas', y = 'capturas (t)') +
  theme(panel.background = element_rect(fill = "gray99")) +
  theme(panel.grid=element_line(color=NA)) +
  ggtitle('Flota')+
  theme(plot.title = element_text(size = 6))
```



```
mediaTalla_flota<-as.numeric(plot_tallas$variable)*as.numeric(plot_tallas$value)

TallasMedias_flota<-plot_tallas %>% mutate(value2=mediaTalla_flota) %>% group_by(year) %>%
  summarize(sum1 = sum(value),sum2 = sum(value2)) %>% mutate(Lmed=sum2/sum1)

ggplot(TallasMedias_flota,aes(x=year,y=Lmed))+ geom_point()+geom_line()+
  labs(x = 'Años', y = 'Tallas medias (cm)') +
  theme(panel.background = element_rect(fill = "gray99")) +
  theme(panel.grid=element_line(color=NA)) +
  ggtitle('Flota')+
  theme(plot.title = element_text(size = 6))
```

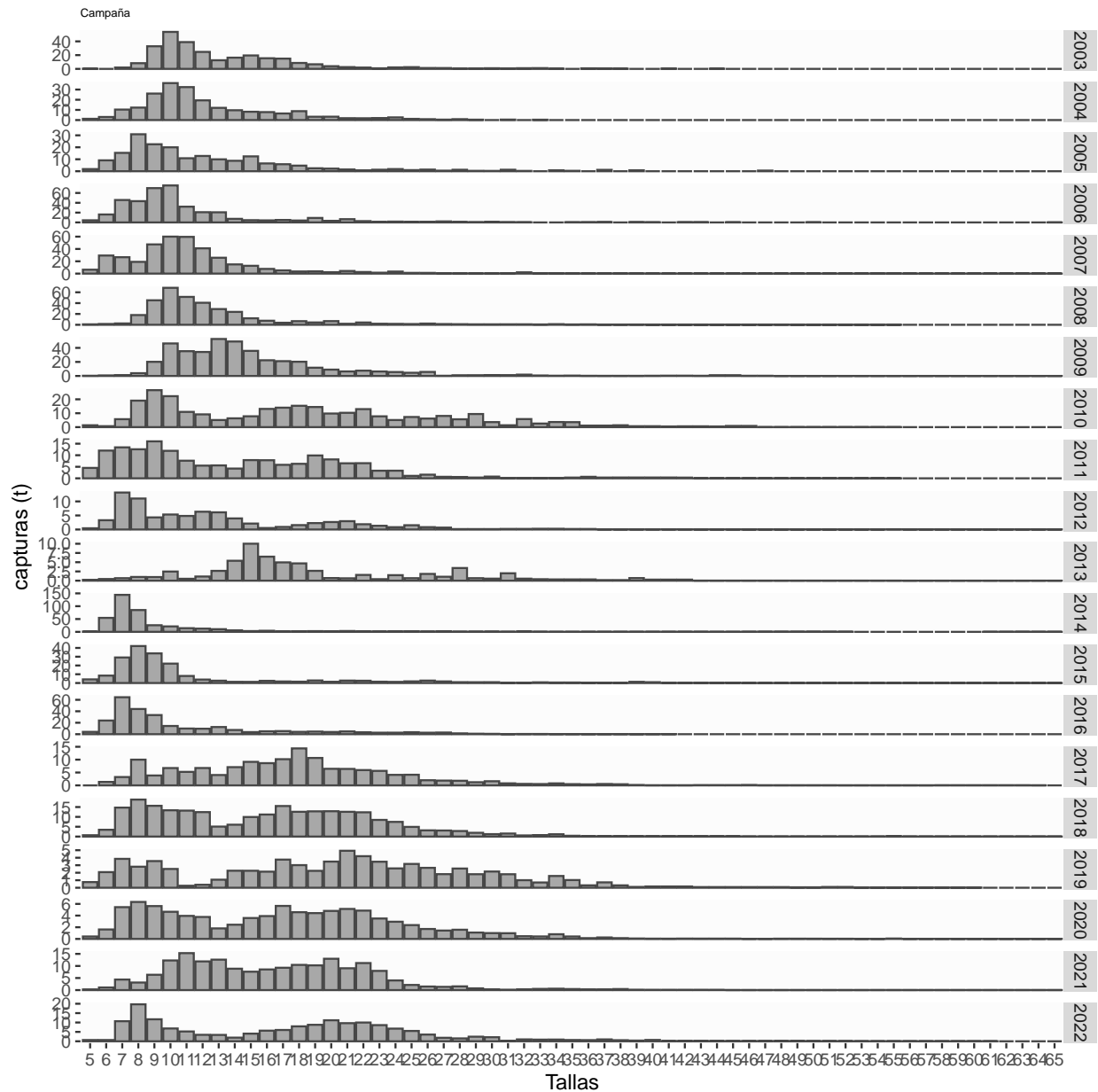


```
Indtallas<-read_excel(excel, sheet = "indices",col_names= TRUE,col_types=NULL,na="",skip= 1)

yearcatch<-seq(2003,2022,1)
tallas<-seq(5,65,1)
Indtallas<-Indtallas[2:62,10:29]

IndicesTallas<-data.frame(row.names = NULL,year=yearcatch,t(Indtallas))
colnames(IndicesTallas)<-c("year",tallas)
plot_Indtallas<-IndicesTallas %>% melt(id.vars=c("year"))

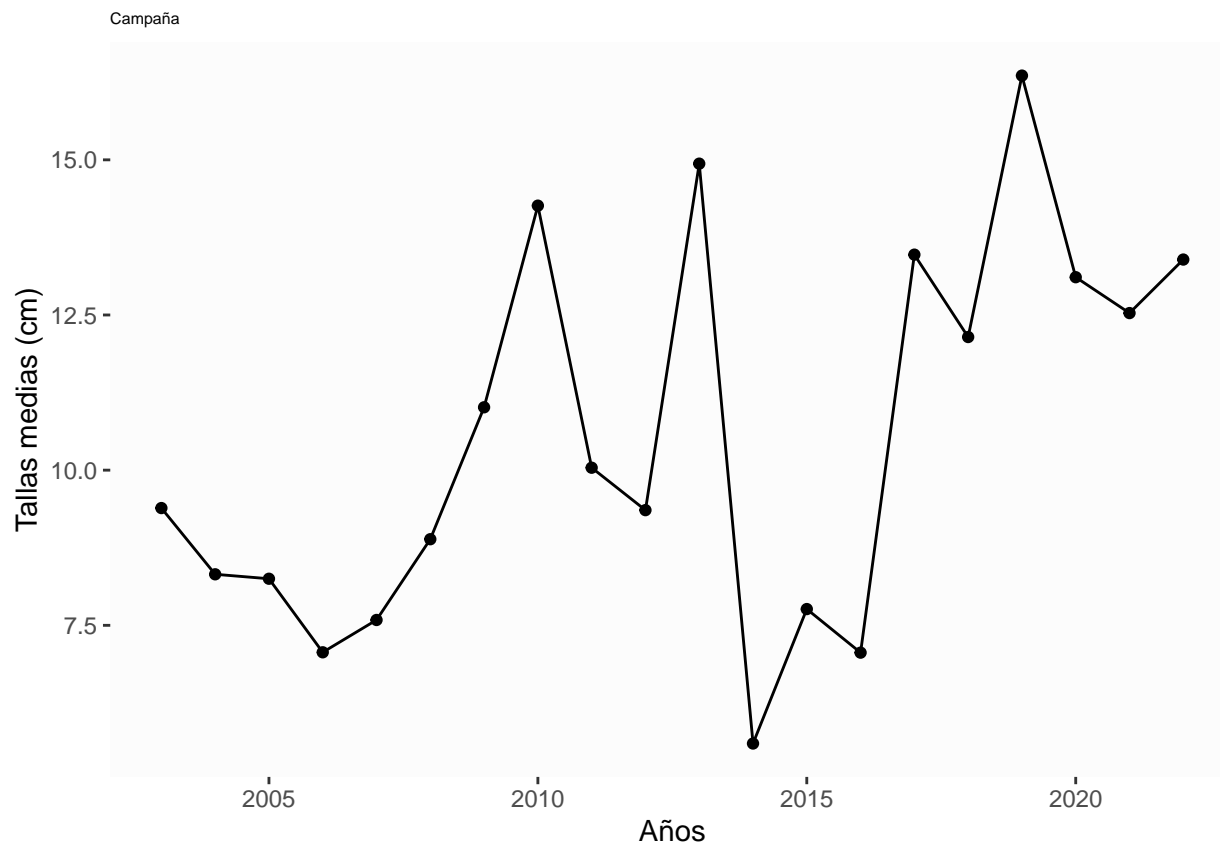
ggplot(plot_Indtallas) +
  geom_bar(aes(x = variable, y = value), stat="identity", fill='gray66', color = 'gray28')+
  facet_grid(rows = vars(year), as.table = TRUE,scales = "free") +
  labs(x = 'Tallas', y = 'capturas (t)') +
  theme(panel.background = element_rect(fill ="gray99")) +
  theme(panel.grid=element_line(color=NA)) +
  ggtitle('Campaña')+
  theme(plot.title = element_text(size = 6))
```



```
mediaTalla<-as.numeric(plot_Indtallas$variable)*as.numeric(plot_Indtallas$value)

TallasMedias_Ind<-plot_Indtallas %>% mutate(value2=mediaTalla) %>% group_by(year) %>%
  summarize(sum1 = sum(value),sum2 = sum(value2)) %>% mutate(Lmed=sum2/sum1)

ggplot(TallasMedias_Ind,aes(x=year,y=Lmed))+ geom_point()+geom_line()+
  labs(x = 'Años', y = 'Tallas medias (cm)') +
  theme(panel.background = element_rect(fill ="gray99")) +
  theme(panel.grid=element_line(color=NA)) +
  ggtitle('Campaña')+
  theme(plot.title = element_text(size = 6))
```



#### 5.1.10 Composición de tallas Formato SS3

La siguiente sección configura los intervalos de talla (`length bin`) de la población.

Esto debe especificarse ya sea que se utilicen o no datos de composiciones de tallas (aunque podría generar los intervalos de longitud de la población a partir de los intervalos de datos de composiciones de tallas).

[Consulte la Guía de usuario de SS3: Sección 7.14 “Length Composition Data Structure”.](#)

##### 5.1.10.1 Bins tallas

```
# set up population length bin structure (note - irrelevant if not using size
#data and using empirical wtatage
dat1$lbin_method<-2      # length bin method:1=use databins;
                          # 2=generate from binwidth,min,max below;
                          # 3=read vector
dat1$binwidth<-2         # binwidth for population size comp
dat1$minimum_size<-10    # minimum size in the population
                          # (lower edge of first bin and size at age 0.00)
dat1$maximum_size<-70    # maximum size in the population (lower edge of last bin)
                          # Valor asumido, Revisar!!! aunque no se usan datos de tallas
                          # en este recurso.
dat1$use_lencomp<-0      # use length composition data (0/1)
```

Después de los intervalos de tallas de la población está la especificación para la composición de tallas (asumiendo 1 línea por flota):

##### 5.1.10.2 Especificación composición de tallas

```

#-----
#_mintailcomp:  upper and lower distribution for females and males separately are
#                accumulated until exceeding this level.
#_addtocomp:    after accumulation of tails; this value added to all bins
#_combM+F:      males and females treated as combined gender below this bin number
#_compressbins: accumulate upper tail by this number of bins; acts simultaneous with
#                mintailcomp; set=0 for no forced accumulation
#_Comp_Error:   0=multinomial,
#                1=dirichlet using Theta*n,
#                2=dirichlet using beta,
#                3=MV_Tweedie
#_ParmSelect:   consecutive index for dirichlet or MV_Tweedie
#_minsamplesize: minimum sample size; set to 1 to match 3.24, minimum value is 0.001
#
#_mintailcomp addtocomp combM+F CompressBins CompError ParmSelect minsamplesize
#-----
# Arreglo de datos
len_info_mintailcomp<-rep(-1,4)
len_info_addtocomp<-rep(0.001,4)
len_info_combine_M_F<-rep(0,4)
len_info_CompressBins<-rep(0,4)
len_info_CompError<-rep(0,4)
len_info_ParmSelect<-rep(0,4)
len_info_minsamplesize<-rep(1,4)
#-----
# crear data.frame
len_info1<-data.frame(mintailcomp=len_info_mintailcomp,
                      addtocomp=len_info_addtocomp,
                      combine_M_F=len_info_combine_M_F,
                      CompressBins=len_info_CompressBins,
                      CompError=len_info_CompError,
                      ParmSelect=len_info_ParmSelect,
                      minsamplesize=len_info_minsamplesize)

row.names(len_info1)<-c("FISHERY", "SURVEY1", "SURVEY2","SURVEY3")
#-----
dat1$len_info<-len_info1 #data.frame
dat1$len_info
##           mintailcomp addtocomp combine_M_F CompressBins CompError ParmSelect
## FISHERY           -1     0.001           0           0           0           0
## SURVEY1           -1     0.001           0           0           0           0
## SURVEY2           -1     0.001           0           0           0           0
## SURVEY3           -1     0.001           0           0           0           0
##           minsamplesize
## FISHERY              1
## SURVEY1              1
## SURVEY2              1
## SURVEY3              1
#-----

```

### 5.1.10.3 Especificación del vector de tallas

```

dat1$N_lbins<-26
dat1$lbin_vector<-seq(20,70,2)

```

#### 5.1.10.4 Datos de composición de tallas

```
#-----
# sex codes:  0=combined;
#             1=use female only;
#             2=use male only;
#             3=use both as joint sexlength distribution
# partition codes: (0=combined;
#                  1=discard;
#                  2=retained
#-----
# Arreglo de datos
new_lencomp <- data.frame(Yr = "# -9999",
                        Seas = 0,
                        FltSvy = 0,
                        Gender = 0,
                        Part = 0,
                        Nsamp = 0)

dat_rows_names <- paste("L",seq(20,70,2),sep="")
dat_rows <- as.data.frame(matrix(data = 0,
                                nrow = nrow(new_lencomp),
                                ncol = length(dat_rows_names)))
names(dat_rows)<-dat_rows_names
#-----
# crear data.frame
new_lencomp1<-cbind(new_lencomp, dat_rows)
#-----
dat1$lencomp<-new_lencomp1
dat1$lencomp
##           Yr Seas FltSvy Gender Part Nsamp L20 L22 L24 L26 L28 L30 L32 L34 L36 L38
## 1 # -9999    0      0      0    0    0    0  0  0  0  0  0  0  0  0  0  0
##   L40 L42 L44 L46 L48 L50 L52 L54 L56 L58 L60 L62 L64 L66 L68 L70
## 1   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0   0
#-----
```

### 5.1.11 Composición de edad Formato a4a

Matriz de número de individuos por edad/año de las capturas (Miles de individuos)

```
dir(dir_GSA1)[2]
## [1] "CATNUM.DAT"
CATNUM.DAT<-read.table(paste(dir_GSA1,dir(dir_GSA1)[2],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)

CATNUM.DAT
##      V1      V2      V3      V4      V5      V6
## 1  746.7 1502.0 216.0 44.0 2.1 0.5
## 2 1425.9 2837.4 231.9 20.5 1.7 0.5
## 3  212.8 1132.5 277.0 36.6 1.2 0.7
## 4 2848.6  861.6 256.2 30.9 1.2 1.4
## 5  421.0 1409.5 163.7 25.2 4.6 1.2
## 6  306.3  986.0 208.9 28.1 4.7 1.2
## 7  650.6 2902.1 291.0 37.6 2.1 0.8
## 8  197.6 1962.0 404.5 18.4 2.2 0.3
## 9  278.9 3410.3 324.9 15.8 1.1 0.1
## 10 133.8 2464.1 198.4 11.7 0.7 0.1
## 11  36.6 1360.9 223.6 19.4 0.3 0.1
## 12  48.2 1086.1 136.4 13.3 0.3 0.2
## 13  61.7  791.1 106.1  7.7 0.7 0.3
## 14  79.5  989.3  78.1  2.7 0.1 0.1
## 15 809.8 1750.2 126.9  4.6 0.2 0.1
## 16 754.5 2320.0 189.8 13.7 0.3 0.1
## 17 107.4 1171.8 197.6  5.5 0.2 0.1
## 18 123.8  996.9  78.3  3.0 0.3 0.1
## 19 731.8 2299.1  87.5  4.6 0.1 0.1

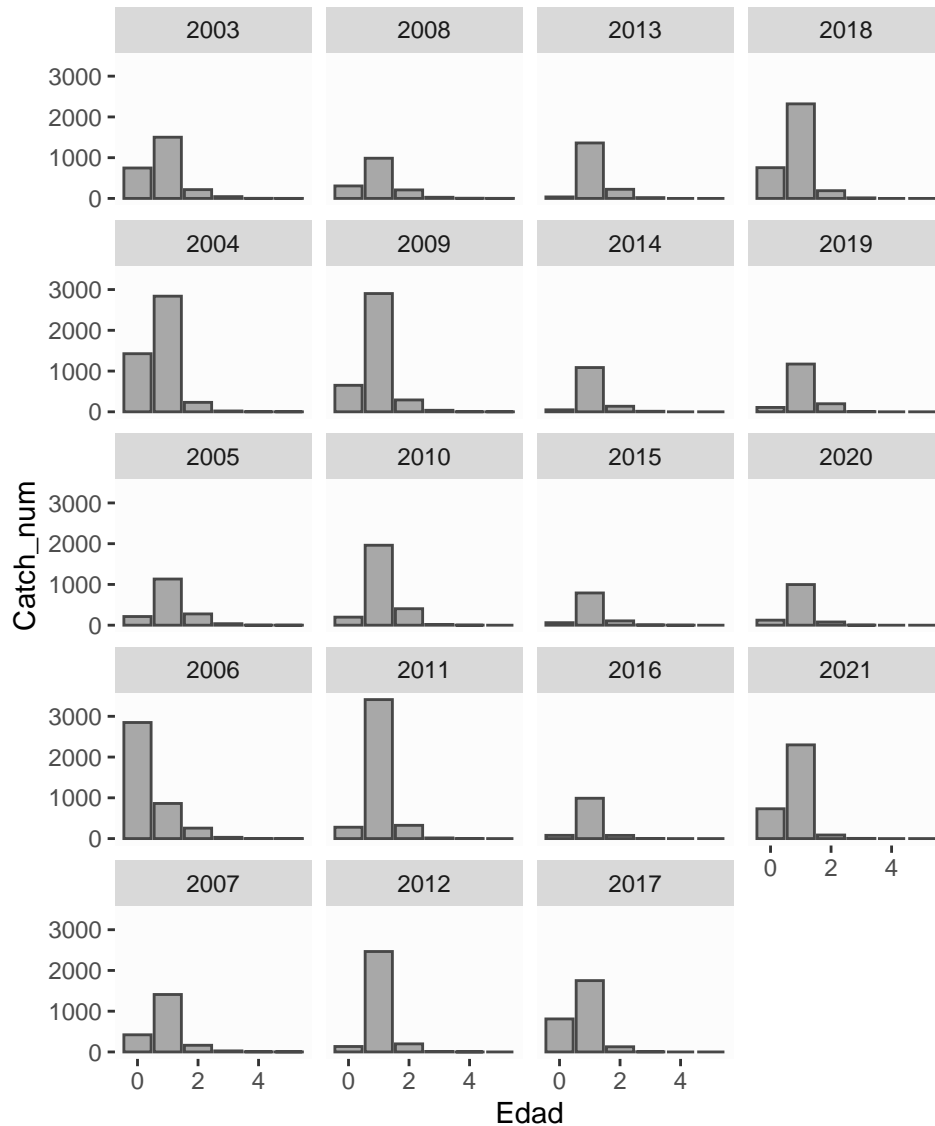
age<-seq(0,5,1)
years<-2003:2021
nyears<-length(years)

CATNUM.DAT<-read.table(paste(dir_GSA1,dir(dir_GSA1)[2],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)

Catnum<-as.data.frame(CATNUM.DAT) %>%
  mutate(Yr=years) %>%
  melt(id.vars="Yr") %>%
  mutate(edad=rep(age,each=nyears)) %>%
  mutate(type='Catch')

ggplot(Catnum) +
  geom_bar(aes(x = edad, y = value), stat="identity", fill='gray66', color = 'gray28') +
  facet_wrap(vars(Yr), dir = 'v', as.table = TRUE) +
  labs(x = 'Edad', y = 'Catch_num') +
  theme(panel.background = element_rect(fill ="gray99")) +
  theme(panel.grid=element_line(color=NA)) + theme(plot.title = element_text(size = 12))
```

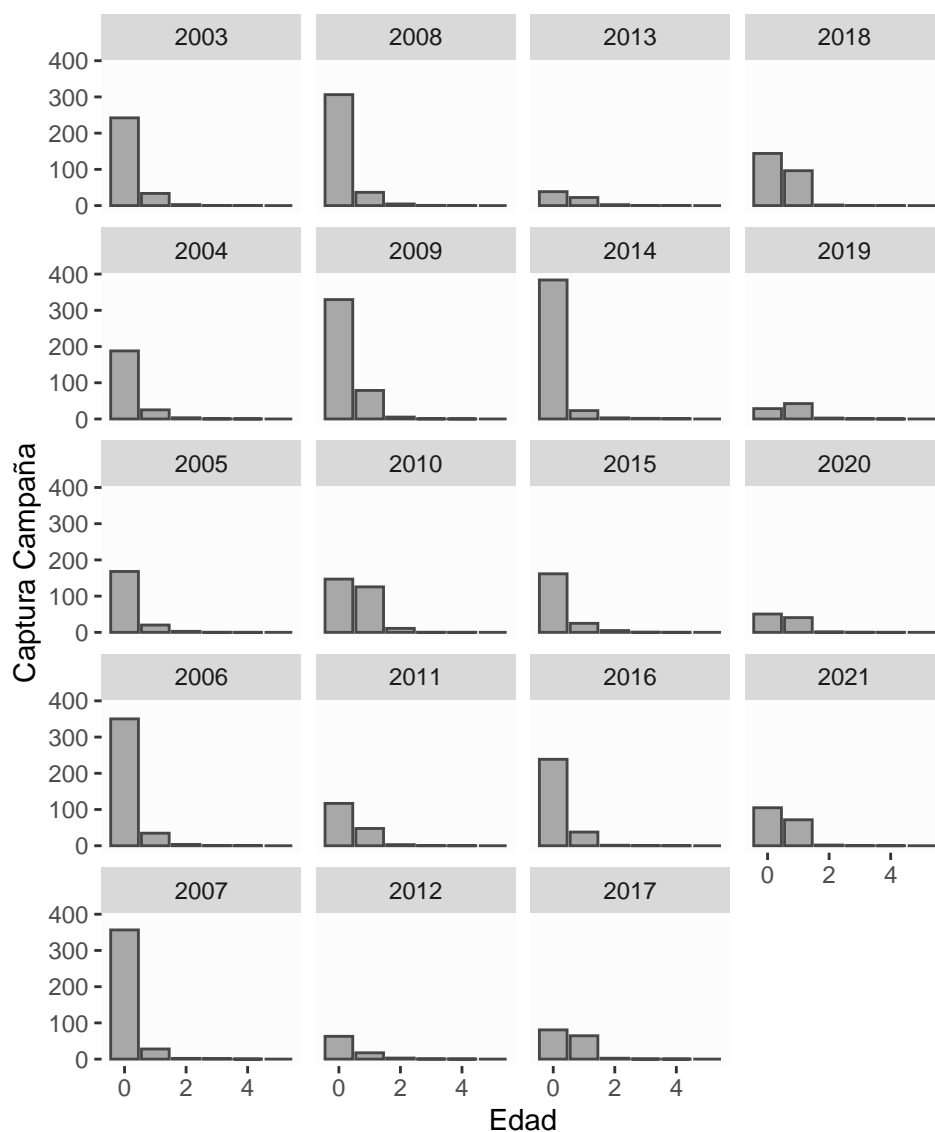




```
age<-seq(0,5,1)
years<-2003:2021
nyears<-length(years)
```

```
Indnum<-as.data.frame(TUNEFF.DAT[,2:7]) %>%
  mutate(Yr=years) %>%
  melt(id.vars="Yr") %>%
  mutate(edad=rep(age,each=nyears)) %>%
  mutate(type='Catch')
```

```
ggplot(Indnum) +
  geom_bar(aes(x = edad, y = value), stat="identity", fill='gray66', color = 'gray28') +
  facet_wrap(vars(Yr), dir = 'v', as.table = TRUE) +
  labs(x = 'Edad', y = 'Captura Campaña') +
  theme(panel.background = element_rect(fill = "gray99")) +
  theme(panel.grid=element_line(color=NA)) + theme(plot.title = element_text(size = 12))
```



### 5.1.12 Composición de edad Formato SS3

A continuación se presentan los datos de composición por edad. En primer lugar, se establecen las categorías de edad y las definiciones de error de edad.

[Consulte la Guía de usuario de SS3: Sección 7.16 “Age Composition Option”.](#)

#### 5.1.12.1 Bins de edad

```
dat1$N_agebins<-6
dat1$N_agebins
## [1] 6
dat1$agebin_vector<-seq(0,5,1)
dat1$agebin_vector
## [1] 0 1 2 3 4 5
```

### 5.1.12.2 Datos de error edad

```
# Arreglo de datos
matrix_ageerror<-rbind(rep(-1,26),rep(0.001,26))
#-----
dat1$N_ageerror_definitions<-1    #_N_ageerror_definitions
dat1$N_ageerror_definitions
## [1] 1
dat1$ageerror<-matrix_ageerror #_ageerror_definitions
dat1$ageerror
##          [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## [1,] -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000
## [2,]  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001
##          [,11] [,12] [,13] [,14] [,15] [,16] [,17] [,18] [,19] [,20]
## [1,] -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000 -1.000
## [2,]  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001
##          [,21] [,22] [,23] [,24] [,25] [,26]
## [1,] -1.000 -1.000 -1.000 -1.000 -1.000 -1.000
## [2,]  0.001  0.001  0.001  0.001  0.001  0.001
```

### 5.1.12.3 Especificaciones de los datos de composicion de edad

```
#-----
#_mintailcomp:  upper and lower distribution for females and males separately are
#               accumulated until exceeding this level.
#_addtocomp:    after accumulation of tails; this value added to all bins
#_combM+F:     males and females treated as combined gender below this bin number
#_compressbins: accumulate upper tail by this number of bins; acts simultaneous
#               with mintailcomp; set=0 for no forced accumulation
#_Comp_Error:  0=multinomial,
#               1=dirichlet using Theta*n,
#               2=dirichlet using beta,
#               3=MV_Tweedie
#_ParmSelect:  consecutive index for dirichlet or MV_Tweedie
#_minsamplesize: minimum sample size; set to 1 to match 3.24, minimum value is 0.001
#
#_mintailcomp addtocomp combM+F CompressBins CompError ParmSelect minsamplesize
#-----
# crear data.frame
age_info1<-data.frame(mintailcomp=rep(-1,4),
                      addtocomp=rep(0.001,4),
                      combine_M_F=rep(0,4),
                      CompressBins=rep(0,4),
                      CompError=rep(0,4),
                      ParmSelect=rep(0,4),
                      minsamplesize=rep(1,4))

row.names(age_info1)<-c("FISHERY", "SURVEY1", "SURVEY2", "SURVEY3")
#-----
dat1$age_info<-age_info1
dat1$age_info
##          mintailcomp addtocomp combine_M_F CompressBins CompError ParmSelect
## FISHERY             -1     0.001           0           0           0           0
## SURVEY1             -1     0.001           0           0           0           0
## SURVEY2             -1     0.001           0           0           0           0
```

```
## SURVEY3          -1      0.001          0          0          0          0
##          minsamplesize
## FISHERY          1
## SURVEY1          1
## SURVEY2          1
## SURVEY3          1
#-----
```

#### 5.1.12.4 Datos de composicion de edad

```
#-----
dat1$Lbin_method <- 1 #_Lbin_method_for_Age_Data: 1=poplenbins; 2=datalenbins; 3=lengths
#-----
# sex codes: 0=combined;
#             1=use female only;
#             2=use male only;
#             3=use both as joint sexlength distribution
# partition codes: (0=combined;
#                   1=discard;
#                   2=retained)
#_yr month fleet sex part ageerr Lbin_lo Lbin_hi Nsamp datavector(female-male)
#-----
# Arreglo de datos
new_agecomp<-data.frame(Yr=2003:2021,
                        Seas=7,
                        FltSvy=-1,
                        Gender=3,
                        Part=0,
                        Ageerr=2,
                        Lbin_lo=-1,
                        Lbin_hi=-1,
                        Nsamp=75)

dat_rows_names<-paste("E",seq(0,5,1),sep="")

#dat_rows<-as.data.frame(matrix(data = CATNUM.DAT,
#                                nrow = nrow(new_agecomp),
#                                ncol = length(dat_rows_names)))

names(CATNUM.DAT)<-dat_rows_names

new_agecomp<-cbind(new_agecomp, CATNUM.DAT)
#-----
dat1$agecomp<-new_agecomp
dat1$agecomp
##      Yr Seas FltSvy Gender Part Ageerr Lbin_lo Lbin_hi Nsamp      E0      E1
## 1  2003    7    -1     3     0     2     -1     -1    75  746.7 1502.0
## 2  2004    7    -1     3     0     2     -1     -1    75 1425.9 2837.4
## 3  2005    7    -1     3     0     2     -1     -1    75  212.8 1132.5
## 4  2006    7    -1     3     0     2     -1     -1    75 2848.6  861.6
## 5  2007    7    -1     3     0     2     -1     -1    75  421.0 1409.5
## 6  2008    7    -1     3     0     2     -1     -1    75  306.3  986.0
## 7  2009    7    -1     3     0     2     -1     -1    75  650.6 2902.1
```

```

## 8 2010 7 -1 3 0 2 -1 -1 75 197.6 1962.0
## 9 2011 7 -1 3 0 2 -1 -1 75 278.9 3410.3
## 10 2012 7 -1 3 0 2 -1 -1 75 133.8 2464.1
## 11 2013 7 -1 3 0 2 -1 -1 75 36.6 1360.9
## 12 2014 7 -1 3 0 2 -1 -1 75 48.2 1086.1
## 13 2015 7 -1 3 0 2 -1 -1 75 61.7 791.1
## 14 2016 7 -1 3 0 2 -1 -1 75 79.5 989.3
## 15 2017 7 -1 3 0 2 -1 -1 75 809.8 1750.2
## 16 2018 7 -1 3 0 2 -1 -1 75 754.5 2320.0
## 17 2019 7 -1 3 0 2 -1 -1 75 107.4 1171.8
## 18 2020 7 -1 3 0 2 -1 -1 75 123.8 996.9
## 19 2021 7 -1 3 0 2 -1 -1 75 731.8 2299.1
##      E2  E3  E4  E5
## 1 216.0 44.0 2.1 0.5
## 2 231.9 20.5 1.7 0.5
## 3 277.0 36.6 1.2 0.7
## 4 256.2 30.9 1.2 1.4
## 5 163.7 25.2 4.6 1.2
## 6 208.9 28.1 4.7 1.2
## 7 291.0 37.6 2.1 0.8
## 8 404.5 18.4 2.2 0.3
## 9 324.9 15.8 1.1 0.1
## 10 198.4 11.7 0.7 0.1
## 11 223.6 19.4 0.3 0.1
## 12 136.4 13.3 0.3 0.2
## 13 106.1 7.7 0.7 0.3
## 14 78.1 2.7 0.1 0.1
## 15 126.9 4.6 0.2 0.1
## 16 189.8 13.7 0.3 0.1
## 17 197.6 5.5 0.2 0.1
## 18 78.3 3.0 0.3 0.1
## 19 87.5 4.6 0.1 0.1
#-----

```

### 5.1.13 Otros datos

#### 5.1.13.1 Tallas medias a la edad

```
#-----
dat1$use_MeanSize_at_Age_obs <- 0 #_Use_MeanSize-at-Age_obs (0/1)
dat1$use_MeanSize_at_Age_obs
## [1] 0
#-----
# sex codes: 0=combined;
#             1=use female only;
#             2=use male only;
#             3=use both as joint sexlength distribution
# partition codes: (0=combined;
#                   1=discard;
#                   2=retained
# ageerr codes: positive means mean length-at-age; negative means mean bodywt_at_age
#_yr month fleet sex part ageerr ignore datavector(female-male)
#                                     samplesize(female-male)
#-----
# Arreglo de datos
MeanSize_at_Age_obs_esp<-data.frame(Yr="# -9999",
                                     Seas=0,
                                     FltSvy=0,
                                     Gender=0,
                                     Part=0,
                                     AgeErr=0,
                                     Ignore=0)

dat_rows_names <- c(paste("L",seq(0,5,1),sep=""),paste("nmL",seq(0,5,1),sep="")) # los datos de esta ma
dat_rows <- as.data.frame(matrix(data = 0,
                                nrow = nrow(MeanSize_at_Age_obs_esp),
                                ncol = length(dat_rows_names)))
names(dat_rows)<-dat_rows_names
#-----
# crear data.frame
MeanSize_at_Age_obs1 <-cbind(MeanSize_at_Age_obs_esp, dat_rows)
#-----
dat1$MeanSize_at_Age_obs<-MeanSize_at_Age_obs1
dat1$MeanSize_at_Age_obs
##           Yr Seas FltSvy Gender Part AgeErr Ignore L0 L1 L2 L3 L4 L5 nmL0 nmL1
## 1 # -9999    0      0      0    0      0      0  0  0  0  0  0  0  0  0
##   nmL2 nmL3 nmL4 nmL5
## 1    0    0    0    0
#-----
```

#### 5.1.13.2 Datos ambientales

(tarea = buscar ejemplo donde ingresen estos datos)

```
dat1$N_environ_variables<-0 #_N_environ_variables
dat1$N_environ_variables
## [1] 0
# -2 in yr will subtract mean for that env_var;
# -1 will subtract mean and divide by stddev (e.g. Z-score)
#Yr Variable Value
```

```
#
```

### 5.1.13.3 Sizefreq data

(tarea = buscar ejemplo donde ingresen estos datos)

```
# Sizefreq data. Defined by method because a fleet can use multiple methods
dat1$N_sizefreq_methods<-0 # N sizefreq methods to read (or -1 for expanded options)
dat1$N_sizefreq_methods
## [1] 0
```

### 5.1.13.4 Datos de tags

(tarea = buscar ejemplo donde ingresen estos datos)

```
dat1$do_tags<-0 # do tags (0/1/2); where 2 allows entry of TG_min_recap
dat1$do_tags
## [1] 0
```

### 5.1.13.5 Datos de morfos

(tarea = buscar ejemplo donde ingresen estos datos)

```
dat1$morphcomp_data<-0 # morphcomp data(0/1)
dat1$morphcomp_data
## [1] 0
# Nobs, Nmorphs, mincomp
# yr, seas, type, partition, Nsamp, datavector_by_Nmorphs
#
```

#### 5.1.13.5.1 Priors de selectividad

(tarea = buscar ejemplo donde ingresen estos datos)

```
dat1$use_selectivity_priors<-0 # Do dataread for selectivity priors(0/1)
dat1$use_selectivity_priors
## [1] 0
# Yr, Seas, Fleet, Age/Size, Bin, selex_prior, prior_sd
# feature not yet implemented
#
```

### 5.1.13.6 Final de archivo data.ss

imprime línea final

```
dat1$eof<-TRUE
dat1$eof
## [1] TRUE
```

#### 5.1.14 Escribir archivo de datos modificado con la función SS\_write para el enfoque de modelación SS3

```
r4ss::SS_writedat(dat1,outfile=here(dirname.simple_mod,"data.ss"),overwrite = TRUE)
```



## 5.2 Archivo control.ss

Revisamos los nombres de los componentes de la lista del archivo .dat

```
ctl1 <- r4ss::SS_readctl(here(dirname.base, "control.ss"),
                        verbose = FALSE,
                        use_datlist = TRUE,
                        datlist = dat) #base

ctl1<-ctl1 # para modificar
#names(ctl1) # muestra los objetos de la lista
```

### 5.2.1 Especificaciones iniciales

```
ctl1$nseas <-1
ctl1$N_areas <-1
ctl1$Nages <-6
ctl1$Nsexes <-2
ctl1$Npopbins <-70
ctl1$Nfleets <-4
ctl1$Do_AgeKey<-0
```

### 5.2.2 Datos de los archivos

```
ctl1$fleetnames <- c("FISHERY", "SURVEY1", "SURVEY2", "SURVEY3")
ctl1$eof <- TRUE
```

### 5.2.3 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
```

### 5.2.4 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$recr_dist_method <- 2
#-----
# not yet implemented; Future usage: Spawner-Recruitment:
#-----
# 1=global;
# 2=by area
ctl1$recr_global_area <- 1
#-----
# number of recruitment settlement assignments
#-----
ctl1$recr_dist_read <- 1
#-----
# unused option
#-----
ctl1$recr_dist_inx <- 0
#-----
# GPattern month area age (for each settlement assignment)
#-----
rec_pattern<-data.frame(row.names="recr_dist_pattern1",
                        "GPattern" = 1,
                        "month"    = 1,
                        "area"     = 1,
                        "age"      = 0)
ctl1$recr_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
```

### 5.2.5 Bloques

```
#-----  
#_Nblock_Patterns  
#-----  
ctl1$N_Block_Designs    <- 1  
  
#-----  
#_blocks_per_pattern  
#-----  
ctl1$blocks_per_pattern <- 1  
  
#-----  
# begin and end years of blocks  
#-----  
ctl1$Block_Design[[1]]  <- c(2003,2003)
```

### 5.2.6 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 sele  
# 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
#

```

### 5.2.7 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 <- 1

#-----
#_Growth_Age_for_L2 (999 to use as Linf)
#-----
ctl1$Growth_Age_for_L2 <- 25

#-----
#_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

#-----
#_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 <- 5

##-----
#_First_Mature_Age
ctl1$First_Mature_Age <- 2

##-----
#_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  
#*

### 5.2.8 Parámetros biológicos

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

- `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 longitud (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:

- `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

### 5.2.8.1 Parámetros hembras

#### 5.2.8.1.1 Mortalidad natural Fem GP\_1

```
##-----  
# NatM_uniform_Fem_GP_1  
##-----  
MG_parms1<-data.frame(row.names="NatM_p_1_Fem_GP_1",  
                        "LO"          = 0.05,  
                        "HI"          = 0.4,  
                        "INIT"        = 0.18,  
                        "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  
##-----
```

#### 5.2.8.1.2 longitud a la edad mínima Fem GP\_1

```
##-----  
# Sex: 1 BioPattern: 1 Growth  
# L_at_Amin_Fem_GP_1  
##-----  
MG_parms2<-data.frame(row.names="L_at_Amin_Fem_GP_1",  
                        "LO"          = 2,  
                        "HI"          = 15,  
                        "INIT"        = 5,  
                        "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  
##-----
```

#### 5.2.8.1.3 longitud a la edad máxima Fem GP\_1

```
##-----  
# Sex: 1 BioPattern: 1 Growth
```

```

# L_at_Amax_Fem_GP_1
##-----
MG_parms3<-data.frame(row.names="L_at_Amax_Fem_GP_1",
                      "LO"          = 45,
                      "HI"          = 60,
                      "INIT"        = 53,
                      "PRIOR"       = 50,
                      "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[3,] <- MG_parms3
##-----

```

#### 5.2.8.1.4 Tasa de crecimiento Fem GP\_1

```

##-----
# VonBert_K_Fem_GP_1
##-----
MG_parms4<-data.frame(row.names="VonBert_K_Fem_GP_1",
                      "LO"          = 0.2,
                      "HI"          = 0.4,
                      "INIT"        = 0.3,
                      "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
##-----

```

#### 5.2.8.1.5 CV crecimiento Fem GP\_1

```

##-----
# CV_young_Fem_GP_1
##-----
MG_parms5<-data.frame(row.names="CV_young_Fem_GP_1",
                      "LO"          = 0.03,
                      "HI"          = 0.16,
                      "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
##-----
MG_parms6<-data.frame(row.names="CV_old_Fem_GP_1",
        "LO"            = 0.03,
        "HI"            = 0.16,
        "INIT"          = 0.062,
        "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
##-----

```

#### 5.2.8.1.6 Relación longitud-peso Fem GP\_1

```

##-----
# Sex: 1 BioPattern: 1 WtLen
# Wtlen_1_Fem_GP_1
##-----
MG_parms7<-data.frame(row.names="Wtlen_1_Fem_GP_1",
        "LO"            = -3,
        "HI"            = 3,
        "INIT"          = 7e-06,
        "PRIOR"          = 7e-06,
        "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
##-----

##-----
# Sex: 1 BioPattern: 1 WtLen
# Wtlen_2_Fem_GP_1
##-----
MG_parms8<-data.frame(row.names="Wtlen_2_Fem_GP_1",
        "LO"             = -3,
        "HI"             = 3,
        "INIT"           = 2.9624,
        "PRIOR"          = 2.9624,
        "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
##-----

```

#### 5.2.8.1.7 Relación Madurez Fem GP\_1

```

##-----
# Sex: 1 BioPattern: 1 Maturity&Fecundity
# Mat50%_Fem_GP_1
##-----
MG_parms9<-data.frame(row.names="Mat50%_Fem_GP_1",
        "LO"             = -3,
        "HI"             = 43,
        "INIT"           = 37,
        "PRIOR"          = 37,
        "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

```

```
##-----
##-----
# Sex: 1 BioPattern: 1 Maturity&Fecundity
# Mat_slope_Fem_GP_1
##-----
MG_parms10<-data.frame(row.names="Mat_slope_Fem_GP_1",
                        "LO"          = -3,
                        "HI"          = 3,
                        "INIT"        = -0.48,
                        "PRIOR"       = -0.48,
                        "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
##-----
```

#### 5.2.8.1.8 Eggs/kg Fem GP\_1

```
##-----
# Eggs/kg_inter_Fem_GP_1
##-----
MG_parms11<-data.frame(row.names="Eggs/kg_inter_Fem_GP_1",
                        "LO"          = -3,
                        "HI"          = 3,
                        "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
##-----
MG_parms12<-data.frame(row.names="Eggs/kg_slope_wt_Fem_GP_1",
                        "LO"          = -3,
                        "HI"          = 3,
```

```

        "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$MG_parms[12,] <- MG_parms12
##-----

```

## 5.2.8.2 Parámetros Machos

### 5.2.8.2.1 Mortalidad natural Mal GP\_1

```

##-----
# Sex: 2 BioPattern: 1 NatMort
# NatM_uniform_Mal_GP_1
##-----
MG_parms13<-data.frame(row.names="NatM_p_1_Mal_GP_1",
        "LO"          = 0,
        "HI"          = 0,
        "INIT"        = 0,
        "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,
        "Block"       = 0,
        "Block_Fxn"   = 0)

##-----
ctl1$MG_parms[13,] <- ""
##-----

```

### 5.2.8.2.2 longitud a la edad Mal GP\_1

```

##-----
# Sex: 2 BioPattern: 1 Growth
# L_at_Amin_Mal_GP_1
##-----
MG_parms14<-data.frame(row.names="L_at_Amin_Mal_GP_1",
        "LO"          = 0,
        "HI"          = 0,
        "INIT"        = 0,
        "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,
        "Block"          = 0,
        "Block_Fxn"      = 0)

##-----
ctl1$MG_parms[14,] <- "##"
##-----

##-----
# Sex: 2 BioPattern: 1 Growth
# L_at_Amax_Mal_GP_1
##-----
MG_parms15<-data.frame(row.names="L_at_Amax_Mal_GP_1",
        "LO"            = 0,
        "HI"            = 0,
        "INIT"          = 0,
        "PRIOR"         = 0,
        "PR_SD"         = 0,
        "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[15,] <- "##"
##-----

```

#### 5.2.8.2.3 Tasa de crecimiento Mal GP\_1

```

##-----
# Sex: 2 BioPattern: 1 Growth
# VonBert_K_Mal_GP_1
##-----
MG_parms16<-data.frame(row.names="VonBert_K_Mal_GP_1",
        "LO"            = 0,
        "HI"            = 0,
        "INIT"          = 0,
        "PRIOR"         = 0,
        "PR_SD"         = 0,
        "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[16,] <-""
##-----

```

#### 5.2.8.2.4 CV crecimiento Mal GP\_1

```

##-----
# Sex: 2 BioPattern: 1 Growth
# CV_young_Mal_GP_1
##-----
MG_parms17<-data.frame(row.names="CV_young_Mal_GP_1",
        "LO"            = 0,
        "HI"            = 0,
        "INIT"          = 0,
        "PRIOR"         = 0,
        "PR_SD"         = 0,
        "PR_type"       = 0,
        "PHASE"         = -0,
        "env_var&link"   = 0,
        "dev_link"      = 0,
        "dev_minyr"     = 0,
        "dev_maxyr"     = 0,
        "dev_PH"        = 0,
        "Block"         = 0,
        "Block_Fxn"     = 0)
##-----
ctl1$MG_parms[17,] <- ""
##-----

```

```

##-----
# Sex: 2 BioPattern: 1 Growth
# CV_old_Mal_GP_1
##-----
MG_parms18<-data.frame(row.names="CV_old_Mal_GP_1",
        "LO"            = 0,
        "HI"            = 0,
        "INIT"          = 0,
        "PRIOR"         = 0,
        "PR_SD"         = 0,
        "PR_type"       = 0,
        "PHASE"         = -0,
        "env_var&link"   = 0,
        "dev_link"      = 0,
        "dev_minyr"     = 0,
        "dev_maxyr"     = 0,
        "dev_PH"        = 0,
        "Block"         = 0,
        "Block_Fxn"     = 0)
##-----
ctl1$MG_parms[18,] <-""

```

```
##-----
```

#### 5.2.8.2.5 Relación longitud-peso Mal GP\_1

```
##-----
```

```
# Sex: 2 BioPattern: 1 WtLen
```

```
# Wtlen_1_Mal_GP_1
```

```
##-----
```

```
MG_parms19<-data.frame(row.names="Wtlen_1_Mal_GP_1",  
                        "LO"          = 0,  
                        "HI"          = 0,  
                        "INIT"        = 0,  
                        "PRIOR"        = 0,  
                        "PR_SD"        = 0,  
                        "PR_type"      = 0,  
                        "PHASE"        = -0,  
                        "env_var&link" = 0,  
                        "dev_link"     = 0,  
                        "dev_minyr"    = 0,  
                        "dev_maxyr"    = 0,  
                        "dev_PH"       = 0,  
                        "Block"        = 0,  
                        "Block_Fxn"    = 0)
```

```
##-----
```

```
ctl1$MG_parms[19,] <- ""
```

```
##-----
```

```
##-----
```

```
# Sex: 2 BioPattern: 1 WtLen
```

```
# Wtlen_2_Mal_GP_1
```

```
##-----
```

```
MG_parms20<-data.frame(row.names="Wtlen_2_Mal_GP_1",  
                        "LO"          = 0,  
                        "HI"          = 0,  
                        "INIT"        = 0,  
                        "PRIOR"        = 0,  
                        "PR_SD"        = 0,  
                        "PR_type"      = 0,  
                        "PHASE"        = -0,  
                        "env_var&link" = 0,  
                        "dev_link"     = 0,  
                        "dev_minyr"    = 0,  
                        "dev_maxyr"    = 0,  
                        "dev_PH"       = 0,  
                        "Block"        = 0,  
                        "Block_Fxn"    = 0)
```

```
##-----
```

```
ctl1$MG_parms[20,] <- ""
```

```
##-----
```

#### 5.2.8.2.6 Hermaphroditism

Revisar como se ingresa esta información, en ejemplo simple no se utiliza

```
##-----
# Hermaphroditism
##-----
```

### 5.2.8.2.7 Distribución del reclutamiento

```
##-----
# Recruitment Distribution
# RecrDist_GP_1
##-----
MG_parms21<-data.frame(row.names="RecrDist_GP_1",
                        "LO"          = 0,
                        "HI"          = 2,
                        "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[21,] <- MG_parms21
##-----
```

```
##-----
# Recruitment Distribution
# RecrDist_Area_1
##-----
MG_parms22<-data.frame(row.names="RecrDist_Area_1",
                        "LO"          = 0,
                        "HI"          = 2,
                        "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[22,]<-MG_parms22
##-----
```

```
##-----
# Recruitment Distribution
# RecrDist_month_1
```

```
##-----
MG_parms23<-data.frame(row.names="RecrDist_month_1",
                        "LO"          = 0,
                        "HI"          = 2,
                        "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[23,]<- MG_parms23
##-----
```

#### 5.2.8.2.8 Cohort growth dev base

```
##-----
# Cohort growth dev base
# CohortGrowDev
##-----
MG_parms24<-data.frame(row.names="CohortGrowDev",
                        "LO"          = 1,
                        "HI"          = 1,
                        "INIT"         = 1,
                        "PRIOR"        = 1,
                        "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$MG_parms[24,] <- MG_parms24
##-----
```

Revisar como se ingresa esta información, en ejemplo simple no se utiliza

```
##-----
# Movement
##-----

##-----
# Age Error from parameters
##-----
```

```
##-----
# catch multiplier
##-----
```

#### 5.2.8.2.9 fraction female, by GP

```
##-----
# fraction female, by GP
# FracFemale_GP_1
##-----
MG_parms25<-data.frame(row.names="FracFemale_GP_1",
                        "LO"          = 0.000001,
                        "HI"          = 0.999999,
                        "INIT"        = 0.999999,
                        "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[25,] <-MG_parms25
##-----
```

#### 5.2.8.2.10 M2 parameter for each predator fleet

Revisar como se ingresa esta información, en ejemplo simple no se utiliza

```
##-----
# M2 parameter for each predator fleet
##-----
#
```

#####\_no timevary MG parameters    Revisar como se ingresa esta información, en ejemplo simple no se utiliza

```
##-----
#_no timevary MG parameters
##-----
#
```

#####\_seasonal\_effects\_on\_biology\_parms

```
##-----
#_femwtlen1,femwtlen2,mat1,mat2,fec1,fec2,Malewtlen1,malewtlen2,L1,K
##-----
```

```
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
#
##-----
```

### 5.2.9 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$Sigma_R_FofCurvature <-0
#-----
#_LO_HI_INIT_PRIOR_PR_SD_PR_type_PHASE_env-var_use_dev_dev_mnyr_dev_mxyr_dev_PH_Block_Black_Fxn_#_parm_name
#names(ctl1$SR_parms)
#row.names(ctl1$SR_parms)
#-----

##-----
#SR_LN(R0)
##-----
SR_parms1<-data.frame(row.names="SR_LN(R0)",
                      "LO"          = 5,
                      "HI"          = 30,
                      "INIT"        = 20,
                      "PRIOR"       = 15,
                      "PR_SD"       = 99,
                      "PR_type"     = 0,
                      "PHASE"       = 1,
                      "env_var&link" = 0,
                      "dev_link"    = 0,
                      "dev_mnyr"    = 0,
                      "dev_mxyr"    = 0,
                      "dev_PH"      = 0,
                      "Block"       = 0,
```

```

                                "Block_Fxn"    = 0)
ctl1$SR_parms[1,]<-SR_parms1

##-----
#SR_BH_steep
##-----
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
##-----
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
##-----
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
##-----
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 30.0 20.00 15.000 99.000      0      1          0          0
## SR_BH_steep 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
## SR_LN(R0)          0          0      0      0          0
## SR_BH_steep          0          0      0      0          0
## SR_sigmaR           0          0      0      0          0
## SR_regime           0          0      0      0          0
## SR_autocorr         0          0      0      0          0

```

## 5.2.10 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 preceed this era
##-----
ctl1$MainRdevYrFirst <- 2003

##-----
# 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 <- "#0"

##-----
#_recdev_early_phase
##-----
ctl1$recdev_early_phase <- "#-4"

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

```

#### 5.2.10.1 Sesgo

```

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

```

```

ctl1$lambda4Fcast_recr_like <-"#1"
##-----
#_last_yr_nobias_adj_in_MPD;
#begin of ramp
##-----
ctl1$last_early_yr_nobias_adj <-"#1900"
##-----
#_first_yr_fullbias_adj_in_MPD;
#begin of plateau
##-----
ctl1$first_yr_fullbias_adj <-"#1900"
##-----
#_last_yr_fullbias_adj_in_MPD
##-----
ctl1$last_yr_fullbias_adj <-"#2001"
##-----
#_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$first_recent_yr_nobias_adj <-"#2002"
##-----
#_max_bias_adj_in_MPD
#(typical ~0.8;
#-3 sets all years to 0.0;
#-2 sets all non-forecast yrs w/ estimated recdevs to 1.0;
#-1 sets biasadj=1.0 for all yrs w/ recdevs)
##-----
ctl1$max_bias_adj<-"#1"

```

### 5.2.10.2 Desvíos

```

#-----
#_period of cycles in recruitment (N parms read below)
#-----
ctl1$period_of_cycles_in_recr <-"#0"
#-----
#min_rec_dev
#-----
ctl1$min_rec_dev <- "#-5"
#-----
#max_rec_dev
#-----
ctl1$max_rec_dev <- "#5"
#-----
#_read_recdevs
#-----
ctl1$N_Read_recdevs <- "#0"
#-----
#_end of advanced SR options
#-----
#
#_placeholder for full parameter lines for recruitment cycles
# read specified recr devs

```

```

#_Yr Input_value
#
# all recruitment deviations
# 1971R 1972R 1973R 1974R 1975R 1976R 1977R 1978R 1979R 1980R 1981R 1982R 1983R 1984R 1985R
# 1986R 1987R 1988R 1989R 1990R 1991R 1992R 1993R 1994R 1995R 1996R 1997R 1998R 1999R 2000R
# 2001R 2002F 2003F 2004F 2005F 2006F 2007F 2008F 2009F 2010F 2011F
# 0.1268 -0.0629442 0.0998014 -0.174095 0.0306484 0.714818 -0.0228752 0.00379775 0.261267
# 0.173626 0.0900049 -0.226622 -0.439888 -0.312088 0.393112 0.551725 0.21891 0.154932
# -0.384786 0.596744 -0.682432 -0.273424 -0.829665 0.365024 -0.605267 0.455103 1.11072
# -0.546499 -0.656469 0.171606 -0.301581 0 0 0 0 0 0 0 0 0 0
#

```

[illegible]

#

## 5.2.12 Capturabilidad

### 5.2.12.1 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
#-----
Q_options1<-data.frame(row.names="SURVEY1",
                        "fleet"      = 2,
                        "link"       = 1,
                        "link_info"  = 0,
                        "extra_se"   = 0,
                        "biasadj"    = 0,
                        "float"      = 0)

#-----
ctl1$Q_options[1,<-Q_options1
#-----
Q_options2<-data.frame(row.names="SURVEY2",
                        "fleet"      = 3,
                        "link"       = 1,
                        "link_info"  = 0,
                        "extra_se"   = 0,
                        "biasadj"    = 0,
                        "float"      = 0)

#-----
ctl1$Q_options[2,<-Q_options2
#-----
Q_options3<-data.frame(row.names="SURVEY3",
                        "fleet"      = 4,
                        "link"       = 1,
                        "link_info"  = 0,
                        "extra_se"   = 0,
                        "biasadj"    = 0,
                        "float"      = 0)

#-----
ctl1$Q_options[3,<-Q_options3
#-----
```

### 5.2.12.2 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_SURVEY1(2)",
                      "LO"          = -7,
                      "HI"          = 5,
                      "INIT"        = 0.516018,
                      "PRIOR"        = 0,
                      "PR_SD"        = 1,
                      "PR_type"      = 0,
                      "PHASE"        = 1,
                      "env_var&link" = 0,
                      "dev_link"     = 0,
                      "dev_mnyr"     = 0,
                      "dev_mxyr"     = 0,
                      "dev_PH"       = 0,
                      "Block"        = 0,
                      "Block_Fxn"    = 0)

#-----
ctl1$Q_parms[1,]<-Q_parms1
#-----
Q_parms2<-data.frame(row.names="LnQ_base_SURVEY2(3)",
                      "LO"          = -7,
                      "HI"          = 5,
                      "INIT"        = 0.516018,
                      "PRIOR"        = 0,
                      "PR_SD"        = 1,
                      "PR_type"      = 0,
                      "PHASE"        = 1,
                      "env_var&link" = 0,
                      "dev_link"     = 0,
                      "dev_mnyr"     = 0,
                      "dev_mxyr"     = 0,
                      "dev_PH"       = 0,
                      "Block"        = 0,
                      "Block_Fxn"    = 0)

#-----
ctl1$Q_parms[2,]<-Q_parms2
#-----
Q_parms3<-data.frame(row.names="LnQ_base_SURVEY2(4)",
                      "LO"          = -7,
                      "HI"          = 5,
                      "INIT"        = -6.628100,
                      "PRIOR"        = 0,
                      "PR_SD"        = 1,
                      "PR_type"      = 0,
                      "PHASE"        = 1,
                      "env_var&link" = 0,
                      "dev_link"     = 0,
                      "dev_mnyr"     = 0,
                      "dev_mxyr"     = 0,
                      "dev_PH"       = 0,
                      "Block"        = 0,

```

```
                                "Block_Fxn"    = 0)
#-----
ctl1$Q_parms[3,]<-Q_parms3
#-----
#_no timevary Q parameters
#-----
#
```



## 5.2.13 Selectividad

### 5.2.13.1 tipos de patrones 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$size_selex_types[1,]<-size_selex_types1
#-----
size_selex_types2<-data.frame(row.names="SURVEY1",
                             "Pattern" = 1,
                             "Discard" = 0,
                             "Male"    = 0,
                             "Special" = 0)
ctl1$size_selex_types[2,]<-size_selex_types2
#-----
size_selex_types3<-data.frame(row.names="SURVEY2",
```

```

        "Pattern" = 0,
        "Discard" = 0,
        "Male"     = 0,
        "Special"  = 0)
ctl1$size_selex_types[3,]<-size_selex_types3
#-----
size_selex_types4<-data.frame(row.names="SURVEY3",
        "Pattern" = 0,
        "Discard" = 0,
        "Male"     = 0,
        "Special"  = 0)
ctl1$size_selex_types[4,]<-size_selex_types4
#-----

```

### 5.2.13.2 tipos de patrones 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; Coleraime - 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
#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 = 11,
        Discard = 0,
        Male     = 0,
        Special  = 0)
ctl1$age_selex_types[1,]<-age_selex_types1
#-----
age_selex_types2<-data.frame(row.names="SURVEY1",
        Pattern = 11,
        Discard = 0,
        Male     = 0,
        Special  = 0)
ctl1$age_selex_types[2,]<-age_selex_types2
#-----
age_selex_types3<-data.frame(row.names="SURVEY2",
        Pattern = 11,

```

```

Discard = 0,
Male     = 0,
Special = 0)
ctl1$age_selex_types[3,]<-age_selex_types3
#-----
age_selex_types4<-data.frame(row.names="SURVEY3",
                             Pattern = 11,
                             Discard = 0,
                             Male     = 0,
                             Special = 0)
ctl1$age_selex_types[4,]<-age_selex_types4
#-----
ctl1$age_selex_types
##           Pattern Discard Male Special
## FISHERY      11      0    0      0
## SURVEY1      11      0    0      0
## SURVEY2      11      0    0      0
## SURVEY3      11      0    0      0

```

### 5.2.13.3 parametros de selectividad a la talla

```

#
#_LO_HI_INIT_PRIOR_PR_SD_PR_type_PHASE_env-var_use_dev-dev_mnyr-dev_mxyr-dev_PH-Block-Blk_Fxn#parm_name
#-----
# 1  FISHERY LenSelex
#-----
size_selex_parms1<-data.frame(row.names="SizeSel_P_1_FISHERY(1)",
                              "LO"      = 19.00,
                              "HI"      = 80,
                              "INIT"    = 53.64110,
                              "PRIOR"    = 50,
                              "PR_SD"    = 0.01,
                              "PR_type"  = 1,
                              "PHASE"    = 2,
                              "env_var&link" = 0,
                              "dev_link"  = 0,
                              "dev_mnyr"  = 0,
                              "dev_mxyr"  = 0,
                              "dev_PH"    = 0,
                              "Block"     = 0,
                              "Block_Fxn" = 0)
ctl1$size_selex_parms[1,]<-size_selex_parms1
#-----
size_selex_parms2<-data.frame(row.names="SizeSel_P_2_FISHERY(1)",
                              "LO"      = 0.01,
                              "HI"      = 60,
                              "INIT"    = 18.92320,
                              "PRIOR"    = 15,
                              "PR_SD"    = 0.01,
                              "PR_type"  = 1,
                              "PHASE"    = 3,
                              "env_var&link" = 0,
                              "dev_link"  = 0,
                              "dev_mnyr"  = 0,

```

```

        "dev_maxyr"    = 0,
        "dev_PH"       = 0,
        "Block"        = 0,
        "Block_Fxn"    = 0)
ctl1$size_selex_parms[2,]<-size_selex_parms2
#-----
size_selex_parms3<-data.frame(row.names="SizeSel_P_1_SURVEY1(2)",
        "LO"          = 19.00,
        "HI"          = 70,
        "INIT"        = 36.65300,
        "PRIOR"       = 30,
        "PR_SD"       = 0.01,
        "PR_type"     = 1,
        "PHASE"       = 2,
        "env_var&link" = 0,
        "dev_link"    = 0,
        "dev_minyr"   = 0,
        "dev_maxyr"   = 0,
        "dev_PH"      = 0,
        "Block"       = 0,
        "Block_Fxn"   = 0)
ctl1$size_selex_parms[3,]<-size_selex_parms3
#-----
size_selex_parms4<-data.frame(row.names="SizeSel_P_2_SURVEY1(2)",
        "LO"          = 0.01,
        "HI"          = 60,
        "INIT"        = 6.59179,
        "PRIOR"       = 10,
        "PR_SD"       = 0.01,
        "PR_type"     = 1,
        "PHASE"       = 3,
        "env_var&link" = 0,
        "dev_link"    = 0,
        "dev_minyr"   = 0,
        "dev_maxyr"   = 0,
        "dev_PH"      = 0,
        "Block"       = 0,
        "Block_Fxn"   = 0)
ctl1$size_selex_parms[4,]<-size_selex_parms4

ctl1$size_selex_parms
##              LO HI      INIT PRIOR PR_SD PR_type PHASE env_var&link
## SizeSel_P_1_FISHERY(1) 19.00 80 53.64110   50 0.01      1    2      0
## SizeSel_P_2_FISHERY(1)  0.01 60 18.92320   15 0.01      1    3      0
## SizeSel_P_1_SURVEY1(2) 19.00 70 36.65300   30 0.01      1    2      0
## SizeSel_P_2_SURVEY1(2)  0.01 60  6.59179   10 0.01      1    3      0
##              dev_link dev_minyr dev_maxyr dev_PH Block Block_Fxn
## SizeSel_P_1_FISHERY(1)      0      0      0      0    0      0
## SizeSel_P_2_FISHERY(1)      0      0      0      0    0      0
## SizeSel_P_1_SURVEY1(2)      0      0      0      0    0      0
## SizeSel_P_2_SURVEY1(2)      0      0      0      0    0      0

```

#### 5.2.13.4 parametros de selectividad a la edad

```

#-----
age_selex_parms1<-data.frame(row.names="AgeSel_P_1_FISHERY(1)",
                             "LO"           = 0,
                             "HI"           = 40,
                             "INIT"          = 0,
                             "PRIOR"         = 5,
                             "PR_SD"         = 99,
                             "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$age_selex_parms[1,]<-age_selex_parms1
#-----
age_selex_parms2<-data.frame(row.names="AgeSel_P_2_FISHERY(1)",
                             "LO"           = 0,
                             "HI"           = 40,
                             "INIT"          = 40,
                             "PRIOR"         = 6,
                             "PR_SD"         = 99,
                             "PR_type"       = 0,
                             "PHASE"         = -99,
                             "env_var&link"  = 0,
                             "dev_link"      = 0,
                             "dev_minyr"     = 0,
                             "dev_maxyr"     = 0,
                             "dev_PH"        = 0,
                             "Block"         = 0,
                             "Block_Fxn"     = 0)
ctl1$age_selex_parms[2,]<-age_selex_parms2
#-----
age_selex_parms3<-data.frame(row.names="AgeSel_P_1_SURVEY1(2)",
                             "LO"           = 0,
                             "HI"           = 40,
                             "INIT"          = 0,
                             "PRIOR"         = 5,
                             "PR_SD"         = 99,
                             "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$age_selex_parms[3,]<-age_selex_parms3
#-----
age_selex_parms4<-data.frame(row.names="AgeSel_P_2_SURVEY1(2)",

```

```

        "LO"           = 0,
        "HI"           = 40,
        "INIT"          = 40,
        "PRIOR"          = 6,
        "PR_SD"          = 99,
        "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$age_selex_parms[4,]<-age_selex_parms4
#-----
age_selex_parms5<-data.frame(row.names="AgeSel_P_1_SURVEY2(3)",
        "LO"           = 0,
        "HI"           = 40,
        "INIT"          = 0,
        "PRIOR"          = 5,
        "PR_SD"          = 99,
        "PR_type"        = 0,
        "PHASE"          = -99,
        "env_var&link"   = 0,
        "dev_link"       = 0,
        "dev_minyr"      = 0,
        "dev_maxyr"      = 0,
        "dev_PH"         = 0,
        "Block"          = 0,
        "Block_Fxn"      = 0)
ctl1$age_selex_parms[5,]<-age_selex_parms5
#-----
age_selex_parms6<-data.frame(row.names="AgeSel_P_2_SURVEY2(3)",
        "LO"           = 0,
        "HI"           = 40,
        "INIT"          = 0,
        "PRIOR"          = 6,
        "PR_SD"          = 99,
        "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$age_selex_parms[6,]<-age_selex_parms6
#-----
ctl1$age_selex_parms
##           LO HI INIT PRIOR PR_SD PR_type PHASE env_var&link
## AgeSel_P_1_FISHERY(1) 0 40  0   5   99      0  -99          0

```

## AgeSel_P_2_FISHERY(1)	0	40	40	6	99	0	-99	0
## AgeSel_P_1_SURVEY1(2)	0	40	0	5	99	0	-99	0
## AgeSel_P_2_SURVEY1(2)	0	40	40	6	99	0	-99	0
## AgeSel_P_1_SURVEY2(3)	0	40	0	5	99	0	-99	0
## AgeSel_P_2_SURVEY2(3)	0	40	0	6	99	0	-99	0
##	dev_link	dev_minyr	dev_maxyr	dev_PH	Block	Block_Fxn		
## AgeSel_P_1_FISHERY(1)	0	0	0	0	0	0	0	0
## AgeSel_P_2_FISHERY(1)	0	0	0	0	0	0	0	0
## AgeSel_P_1_SURVEY1(2)	0	0	0	0	0	0	0	0
## AgeSel_P_2_SURVEY1(2)	0	0	0	0	0	0	0	0
## AgeSel_P_1_SURVEY2(3)	0	0	0	0	0	0	0	0
## AgeSel_P_2_SURVEY2(3)	0	0	0	0	0	0	0	0

### 5.2.13.5 otros parámetros ??? REVISAR

```
#_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 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 <- 0

#-----
#_maxlambdaphase
#-----
ctl1$maxlambdaphase <- 4

#-----
#_sd_offset;
#-----
# must be 1 if any growthCV, sigmaR, or survey extraSD is an estimated parameter
ctl1$sd_offset <- 1
```



### 5.2.14 Lambdas

```
#-----
# read 3 changes to default Lambdas (default value is 1.0)
#-----
ctl1$N_lambdas <- 3

#-----
# Like_comp codes:
#-----
# 1=surv;
# 2=disc;
# 3=mnwt;
# 4=length;
# 5=age;
# 6=SizeFreq;
# 7=sizeage;
# 8=catch;
# 9=init_equ_catch;
# 10=recrdev;
# 11=parm_prior;
# 12=parm_dev;
# 13=CrashPen;
# 14=Morphcomp;
# 15=Tag-comp;
# 16=Tag-negbin;
# 17=F_ballpark;
# 18=initEQregime

#-----
#like_comp fleet phase value sizefreq_method
#-----
lambdas1<-data.frame(row.names="Surv_SURVEY1_Ph2",
                     like_comp=1,
                     fleet=2,
                     phase=2,
                     value=1,
                     sizefreq_method=1)
ctl1$lambdas[1,]<-lambdas1
#-----
lambdas2<-data.frame(row.names="length_SURVEY1_sizefreq_method_1_Ph2",
                     like_comp=4,
                     fleet=2,
                     phase=2,
                     value=1,
                     sizefreq_method=1)
ctl1$lambdas[2,]<-lambdas2
#-----
lambdas3<-data.frame(row.names="length_SURVEY1_sizefreq_method_1_Ph3",
                     like_comp=4,
                     fleet=2,
                     phase=3,
                     value=1,
                     sizefreq_method=1)
ctl1$lambdas[3,]<-lambdas3
```

```

#-----
ctl1$lambda
##                                like_comp fleet phase value
## Surv_SURVEY1_Ph2                1      2      2      1
## length_SURVEY1_sizefreq_method_1_Ph2      4      2      2      1
## length_SURVEY1_sizefreq_method_1_Ph3      4      2      3      1
##                                sizefreq_method
## Surv_SURVEY1_Ph2                                1
## length_SURVEY1_sizefreq_method_1_Ph2            1
## length_SURVEY1_sizefreq_method_1_Ph3            1
#-----

```

## REVISAR ESTA PARTE

### 5.2.15 more stddev reporting

```
# (0/1/2) read specs for more stddev reporting:
# 0 = skip,
# 1 = read specs for reporting stdev for selectivity, size, and numbers,
# 2 = add options for M,Dyn. Bzero, SmryBio
ctl1$more_stddev_reporting <- 1
```

#### 5.2.15.1 specs ?

```
#-----
# Selectivity:
#-----
# (1) 0 to skip or fleet,
# (2) 1=len/2=age/3=combined,
# (3) year,
# (4) N selex bins;
#NOTE: combined reports in age bins
#-----
stddev_reporting_specs1<-data.frame(matrix(c(1,1,-1,5,1,5,1,-1,5),
                                             nrow=1,
                                             ncol=9))
colnames(stddev_reporting_specs1)<-paste("stddev_reporting_specs_",seq(1,9,1),sep="")

ctl1$stddev_reporting_specs<-stddev_reporting_specs1
```

#### 5.2.15.2 Crecimiento

```
#-----
# Growth:
#-----
#(1) 0 to skip or growth pattern,
#(2) growth ages;
#NOTE: does each sex
#-----
stddev_reporting_growth1<-data.frame(matrix(c(1,2,14,26,40),
                                             nrow=1,
                                             ncol=5))
colnames(stddev_reporting_growth1)<-paste("stddev_reporting_growth_",seq(1,5,1),sep="")

ctl1$stddev_reporting_growth <-stddev_reporting_growth1
```

#### 5.2.15.3 Abundancia a la edad

```
#-----
# Numbers-at-age:
#-----
#(1) 0 or area(-1 for all),
#(2) year,
#(3) N ages;
#NOTE: sums across morphs
#-----
```

```

stddev_reporting_N_at_A1<-data.frame(matrix(c(1,2,14,26,40),
                                             nrow=1,
                                             ncol=5))
colnames(stddev_reporting_N_at_A1)<-paste("stddev_reporting_N_at_A_",seq(1,5,1),sep="")
ctl1$stddev_reporting_N_at_A <- stddev_reporting_N_at_A1

```

#### 5.2.15.4 Selectividad

```

#-----
# vector with selex std bins (-1 in first bin to self-generate)
#-----
stddev_reporting_selex1<-data.frame(matrix(c(5,15,25,35,43),
                                             nrow=1,
                                             ncol=5))
colnames(stddev_reporting_selex1)<-paste("stddev_reporting_selex_",seq(1,5,1),sep="")
ctl1$stddev_reporting_selex <- stddev_reporting_selex1

```

#### 5.2.16 Escribir archivo de control modificado con la función SS\_write para el enfoque de modelación SS3

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

### 5.3 Archivo starter.ss

Revisamos los nombres de los componentes de la lista del archivo .dat

```
#-----
start <- r4ss::SS_readstarter(here(dirname.base,"starter.ss"),
                             verbose = FALSE) #base
#-----

start1<-start # para modificar
#names(start1) # muestra los objetos de la lista

start1$datfile<-"data.ss"
start1$ctlfile<-"control.ss"

#-----
# 0=use init values in control file;
# 1=use ss.par
#-----
start1$init_values_src<-0

#-----
# run display detail (0,1,2)
#-----
start1$run_display_detail<-0

#-----
# detailed output
#-----
# (0=minimal for data-limited,
# 1=high (w/ wtatage.ss_new),
# 2=brief,
# 3=custom)
#-----
start1$detailed_age_structure<-1

#-----
# custom report options:
#-----
# -100 to start with minimal;
# -101 to start with all;
# -number to remove, +number to add, -999 to end
#-----

#-----
# write 1st iteration details to echoinput.sso file (0,1)
#-----
start1$checkup<-0

#-----
# write parm values to ParmTrace.sso
#-----
# (0=no,
# 1=good,active;
# 2=good,all;
# 3=every_iter,all_parms;
```

```

# 4=every,active)
#-----
start1$parmtrace<-0

#-----
# write to cumreport.sso
#-----
#(0=no,
# 1=like&timeseries;
# 2=add survey fits
#-----
start1$cumreport<-0

#-----
# Include prior_like for non-estimated parameters (0,1)
#-----
start1$prior_like<-1

#-----
# Use Soft Boundaries to aid convergence (0,1) (recommended)
#-----
start1$soft_bounds<-1

#-----
# Number of datafiles to produce:
#-----
# 0 turns off all *.ss_new;
# 1st is data_echo.ss_new,
# 2nd is data_expval.ss,
# 3rd and higher are data_boot_**N.ss,
#-----
start1$N_bootstraps<-1

#-----
# Turn off estimation for parameters entering after this phase
#-----
start1$last_estimation_phase<-10

#-----
# MCEval burn interval
#-----
start1$MCMCburn<-0

#-----
# MCEval thin interval
#-----
start1$MCMCthin<-1

#-----
# jitter initial parm value by this fraction
#-----
start1$jitter_fraction<-0

```

```

#-----
# min yr for sdreport outputs (-1 for styr); #_1969
#-----
start1$minyr_sdreport<-1969

#-----
# max yr for sdreport outputs
# (-1 for endyr+1;
#  -2 for endyr+Nforecastyrs);
#  #_2011
#-----
start1$maxyr_sdreport<-2011

#-----
# N individual STD years
#-----
start1$N_STD_yrs<-0

#-----
#vector of year values
#-----

#-----
# final convergence criteria (e.g. 1.0e-04)
#-----
start1$converge_criterion<-1e-04

#-----
# retrospective year relative to end year (e.g. -4)
#-----
start1$retro_yr<-0

#-----
# min age for calc of summary biomass
#-----
start1$min_age_summary_bio<-1

#-----
# Depletion basis:
#-----
# denom is:
# 0=skip;
# 1=rel X*SPBvirgin;
# 2=rel SPBmsy;
# 3=rel X*SPB_styr;
# 4=rel X*SPB_endyr;
# values;
# >=11 invoke N multiyr (up to 9!) with 10's digit;
# >100 invokes log(ratio)
#-----
start1$depl_basis<-2

#-----

```



```

# Fraction (X) for Depletion denominator (e.g. 0.4)
#-----
start1$depl_denom_frac<-1

#-----
# SPR_report_basis:
#-----
# 0=skip;
# 1=(1-SPR)/(1-SPR_tgt);
# 2=(1-SPR)/(1-SPR_MSY);
# 3=(1-SPR)/(1-SPR_Btarget);
# 4=rawSPR
#-----
start1$SPR_basis<-4

#-----
# F_reporting_units:
#-----
# 0=skip;
# 1=exploitation(Bio);
# 2=exploitation(Num);
# 3=sum(Apical_F's);
# 4=true F for range of ages;
# 5=unweighted avg.
# F for range of ages
#-----
start1$F_report_units<-3

#-----
#COND 10 15
#_min and max age over which average F will be calculated with F_reporting=4 or 5
#-----
start1$F_age_range<-"#"

#-----
# F_std_basis:
#-----
# 0=raw_annual_F;
# 1=F/Fspr;
# 2=F/Fmsy;
# 3=F/Ftgt;
# where F means annual_F;
# values >=11 invoke N multiyr (up to 9!) with 10's digit;
# >100 invokes log(ratio)
#-----
start1$F_report_basis<-0

#-----
# MCMC output detail:
#-----
# integer part
# (0 = default;
# 1 = adds obj func components;

```

```

# 2 = +write_report_for_each_mceval);
# and decimal part (added to SR_LN(R0) on first call to mcmc)
#-----
start1$MCMC_output_detail<-0

#-----
# ALK tolerance
#-----
# ***disabled in code (example 0.0001)
#-----
start1$ALK_tolerance<-0

#-----
# random number seed for bootstrap data
# (-1 to use long(time) as seed): # 1664576434
#-----
start1$seed<- -1

#-----
# check value for end of file and for version control
#-----
start1$final<-3.3

```

### 5.3.1 Escribir archivo de starter modificado con la función SS\_write para el enfoque de modelación SS3

```

r4ss::SS_writestarter(mylist = start1,
                      dir = here(dirname.simple_mod),
                      file = "starter.ss",
                      overwrite = TRUE,
                      verbose = TRUE)

```

## 5.4 Archivo forecast.ss

Revisamos los nombres de los componentes de la lista del archivo forecast

```
fore <- r4ss::SS_readforecast(here(dirname.base,"forecast.ss"),
                             verbose = FALSE) #base
fore1<-fore # para modificar
#names(fore1) # muestra los objetos de la lista

#-----
# Benchmarks:
#-----
# 0=skip;
# 1=calc F_spr,F_btgt,F_msy;
# 2=calc F_spr,F0.1,F_msy;
# 3=add F_Blimit;
#-----
fore1$benchmarks <- 1

#-----
# Do_MSY:
#-----
# 1= set to F(SPR);
# 2=calc F(MSY);
# 3=set to F(Btgt) or F0.1;
# 4=set to F(endyr);
# 5=calc F(MEY) with MSY_unit options
#-----
fore1$MSY <- 2

# if Do_MSY=5, enter MSY_Units; then list fleet_ID, cost/F, price/mt, include_in_Fmey_scaling;
# -fleet_ID to fill; -9999 to terminate

#-----
# SPR target (e.g. 0.40)
#-----
fore1$SPRtarget <- 0.4

#-----
# Biomass target (e.g. 0.40)
#-----
fore1$Btarget <- 0.342

#-----
#_Bmark_years:
#-----
# beg_bio, end_bio,
# beg_selex, end_selex,
# beg_relF, end_relF,
# beg_recr_dist, end_recr_dist,
# beg_SRparm, end_SRparm
# (enter actual year, or values of 0 or -integer to be rel. endyr)
#-----
fore1$Bmark_years
```

```

## #_Bmark_years_1 #_Bmark_years_2 #_Bmark_years_3 #_Bmark_years_4
##           2001           2001           2001           2001
## #_Bmark_years_5 #_Bmark_years_6 #_Bmark_years_7 #_Bmark_years_8
##           2001           2001           1971           2001
## #_Bmark_years_9 #_Bmark_years_10
##           1971           2001

# 2001 2001 2001 2001 2001 2001 1971 2001 1971 2001
# value <0 convert to endyr-value;
# except -999 converts to start_yr;
# must be >=start_yr and <=endyr

#-----
#Bmark_relF_Basis:
#-----
# 1 = use year range;
# 2 = set relF same as forecast below
#-----
fore1$Bmark_relF_Basis <- 1

#-----
# Forecast:
#-----
# -1=none;
# 0=simple_1yr;
# 1=F(SPR);
# 2=F(MSY)
# 3=F(Btgt) or F0.1;
# 4=Ave F (uses first-last relF yrs);
# 5=input annual F scalar
#-----
fore1$Forecast <- 1

# where none and simple require no input after this line;
# simple sets forecast F same as end year F

#-----
# N forecast years
#-----
fore1$Nforecastyrs <- 10

#-----
# Fmult (only used for Do_Forecast==5) such that apical_F(f)=Fmult*relF(f)
#-----
fore1$F_scalar <- 0.2

#-----
#_Fcast_years:
#-----
# beg_selex, end_selex,
# beg_relF, end_relF,
# beg_mean_recruits, end_recruits
# (enter actual year, or values of 0 or -integer to be rel. endyr)

```

```

fore1$Fcast_years
## #_Fcast_years_1 #_Fcast_years_2 #_Fcast_years_3 #_Fcast_years_4 #_Fcast_years_5
##          0          0          -10          0          -999
## #_Fcast_years_6
##          0

# 2001 2001 1991 2001 1971 2001

#-----
# Forecast selectivity
#-----
#(0=fcast sele $x$  is mean from year range;
# 1=fcast selectivity from annual time-vary parms)
#-----
fore1$Fcast_selex <- 0

#-----
# Control rule method
#-----
#(0: none;
# 1: ramp does catch=f(SSB), buffer on F;
# 2: ramp does F=f(SSB), buffer on F;
# 3: ramp does catch=f(SSB), buffer on catch;
# 4: ramp does F=f(SSB), buffer on catch)
# values for top, bottom and buffer exist, but not used when Policy=0
#-----
fore1$ControlRuleMethod <- 1

#-----
# Control rule inflection for constant F
#-----
# (as frac of Bzero, e.g. 0.40);
# must be > control rule cutoff, or set to -1 to use Bmsy/SSB_unf
#-----
fore1$BforconstantF <- 0.4

#-----
# Control rule cutoff for no F
#-----
# (as frac of Bzero, e.g. 0.10)
#-----
fore1$BfornoF <- 0.1

#-----
# Buffer:
#-----
# enter Control rule target as fraction of Flimit (e.g. 0.75),
# negative value invokes list of [year, scalar] with filling from year to YrMax
#-----
fore1$Flimitfraction <- 0.75

#-----
#_N forecast loops

```

```

#-----
# (1=OFL only;
# 2=ABC;
# 3=get F from forecast ABC catch with allocations applied)
#-----
fore1$N_forecast_loops <- 3

#-----
#_First forecast loop with stochastic recruitment
#-----
fore1$First_forecast_loop_with_stochastic_recruitment <- 3

#-----
#_Forecast recruitment:
#-----
# 0= spawn_recr;
# 1=value*spawn_recr_fn;
# 2=value*VirginRecr;
# 3=recent mean from yr range above
# (need to set phase to -1 in control to get constant recruitment in MCMC)
#-----
fore1$fcst_rec_option <- 1

#-----
# value is multiplier of SRR
#-----
fore1$fcst_rec_val <- 1

#-----
#_Forecast loop control
#-----
# 5 (reserved for future bells&whistles)
#-----
fore1$Forecast_loop_control_5 <- 0

#-----
#FirstYear for caps and allocations
#-----
# (should be after years with fixed inputs)
#-----
fore1$FirstYear_for_caps_and_allocations <- 2010

#-----
# stddev of log(realized catch/target catch)
#-----
# in forecast (set value>0.0 to cause active impl_error)
#-----
fore1$stddev_of_log_catch_ratio <- 0

#-----
# Do West Coast gfish rebuilder output:
#-----
# 0=no;

```

```

# 1=yes
#-----
fore1$Do_West_Coast_gfish_rebuilder_output <- 0

#-----
# Rebuilder:
#-----
# first year catch could have been set to zero (Ydecl)(-1 to set to 1999)
fore1$Ydecl <- 1999

#-----
# Rebuilder:
#-----
# year for current age structure (Yinit) (-1 to set to endyear+1)
fore1$Yinit <- 2022

#-----
# fleet relative F:
#-----
# 1=use first-last alloc year;
# 2=read seas, fleet, alloc list below
#-----
fore1$fleet_relative_F <- 1

# Note that fleet allocation is used directly as average F if Do_Forecast=4
#-----
# basis for fcast catch tuning and for fcast catch caps and allocation
#-----
# (2=deadbio;
# 3=retainbio;
# 5=deadnum;
# 6=retainnum);
# NOTE: same units for all fleets
#-----
fore1$basis_for_fcast_catch_tuning <- 2

# Conditional input if relative F choice = 2
# enter list of: season, fleet, relF; if used, terminate with season=-9999
# 1 1 1
# -9999 0 0 # terminator for list of relF
# enter list of: fleet number, max annual catch for fleets with a max; terminate with fleet=-9999
# -9999 -1
# enter list of area ID and max annual catch; terminate with area=-9999
# -9999 -1
# enter list of fleet number and allocation group assignment, if any; terminate with fleet=-9999
# -9999 -1

#-----
#_if N allocation groups >0,
# list year, allocation fraction for each group
#-----
fore1$N_allocation_groups <- 0

```

```

# list sequentially because read values fill to end of N forecast
# terminate with -9999 in year field
# no allocation groups
#

#-----
# basis for input Fcast catch:
#-----
# -1=read basis with each obs;
# 2=dead catch;
# 3=retained catch;
# 99=input apical_F;
# NOTE: bio vs num based on fleet's catchunits
#-----
fore1$InputBasis <- 2

#-----
#enter list of Fcast catches or Fa;
# terminate with line having year=-9999
#-----
#_Yr Seas Fleet Catch(or_F)
#-9999 1 1 0
#
#-----
# verify end of input
#-----
fore1$eof<-TRUE

```

#### 5.4.1 Escribir archivo de forecast modificado con la función SS\_write para el enfoque de modelación SS3

```

r4ss::SS_writeforecast(mylist = fore1,
                      dir = here(dirname.simple_mod),
                      file = "forecast.ss",
                      overwrite = TRUE,
                      verbose = TRUE)

```



## 5.5 Archivo watatege

### 5.5.1 Formato a4a

#### 5.5.1.1 Peso medio por edad y año de tu matriz de captura (Kilos).

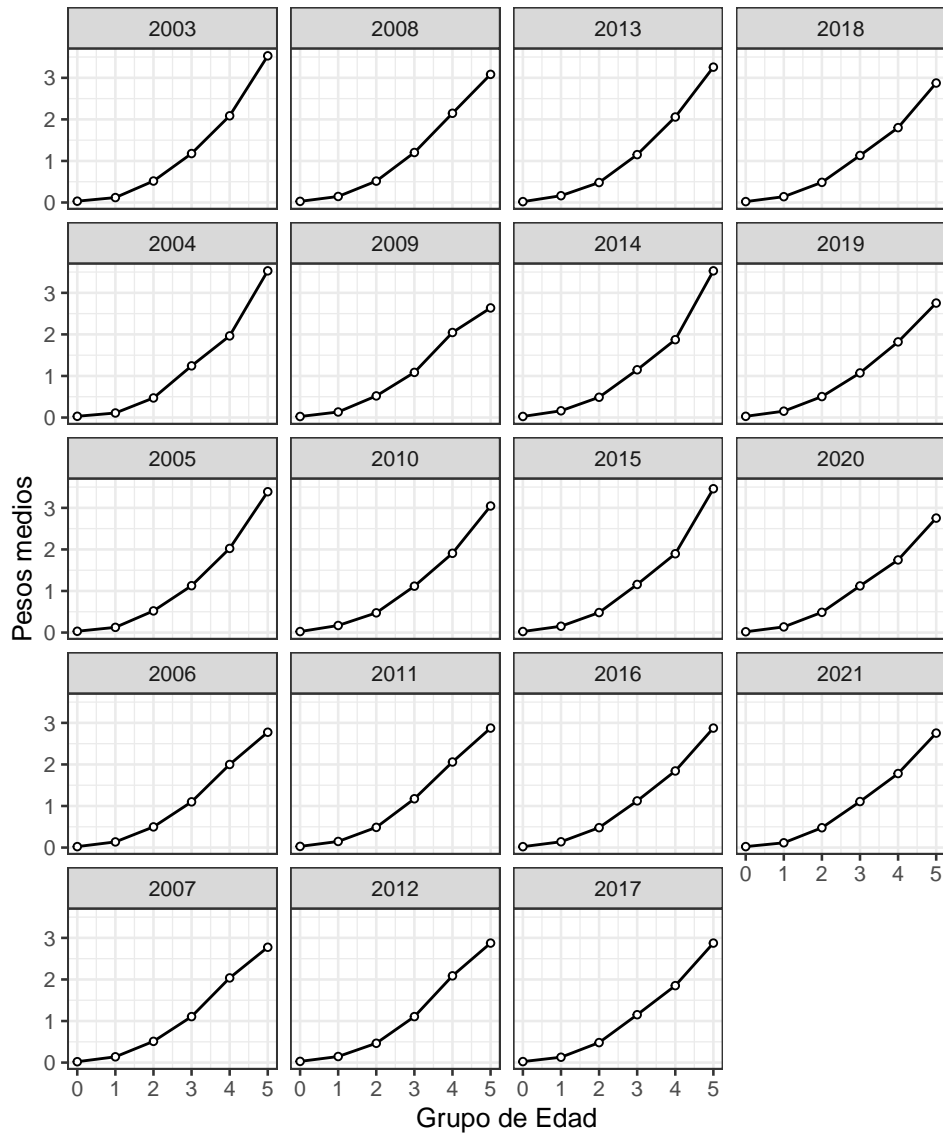
```
dir(dir_GSA1)[3]
## [1] "CATWT.DAT"
CATWT.DAT <- read.table(paste(dir_GSA1,dir(dir_GSA1)[3],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)
CATWT.DAT
##      V1      V2      V3      V4      V5      V6
## 1  0.032 0.119 0.517 1.178 2.086 3.529
## 2  0.030 0.109 0.471 1.243 1.964 3.529
## 3  0.031 0.125 0.521 1.127 2.024 3.388
## 4  0.023 0.136 0.497 1.100 1.999 2.775
## 5  0.021 0.139 0.511 1.106 2.037 2.775
## 6  0.028 0.145 0.515 1.203 2.147 3.083
## 7  0.025 0.133 0.520 1.086 2.046 2.637
## 8  0.024 0.170 0.474 1.116 1.907 3.045
## 9  0.028 0.147 0.486 1.174 2.059 2.875
## 10 0.029 0.146 0.465 1.106 2.087 2.875
## 11 0.021 0.164 0.481 1.152 2.057 3.257
## 12 0.026 0.162 0.488 1.149 1.873 3.529
## 13 0.025 0.152 0.481 1.157 1.896 3.460
## 14 0.022 0.139 0.479 1.123 1.843 2.875
## 15 0.024 0.129 0.481 1.152 1.851 2.875
## 16 0.022 0.140 0.485 1.132 1.801 2.875
## 17 0.029 0.153 0.504 1.073 1.820 2.754
## 18 0.019 0.136 0.487 1.122 1.747 2.754
## 19 0.023 0.115 0.477 1.107 1.781 2.754
```

### 5.5.1.2 Medio por edad y año asumida para el stock (Normalmente = CATWT) (Kilos).

```
age<-seq(0,5,1)
years<-2003:2021
nyears<-length(years)
#dir(dir_GSA1)[11]
STOCWT.DAT <- read.table(paste(dir_GSA1,dir(dir_GSA1)[11],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)
#STOCWT.DAT

stockWt<-as.data.frame(STOCWT.DAT) %>%
  mutate(Yr=years) %>%
  melt(id.vars="Yr") %>%
  mutate(edad=rep(age,each=nyears)) %>%
  mutate(type='Wt')

ggplot(stockWt, aes(edad,value))+
  geom_line()+
  geom_point(size=1,shape=21,fill="white")+
  facet_wrap(vars(Yr),dir='v',as.table=TRUE)+
  labs(x="Grupo de Edad", y="Pesos medios") +
  theme(panel.background = element_rect(fill ="gray95")) +
  theme_bw(base_size=10)
```

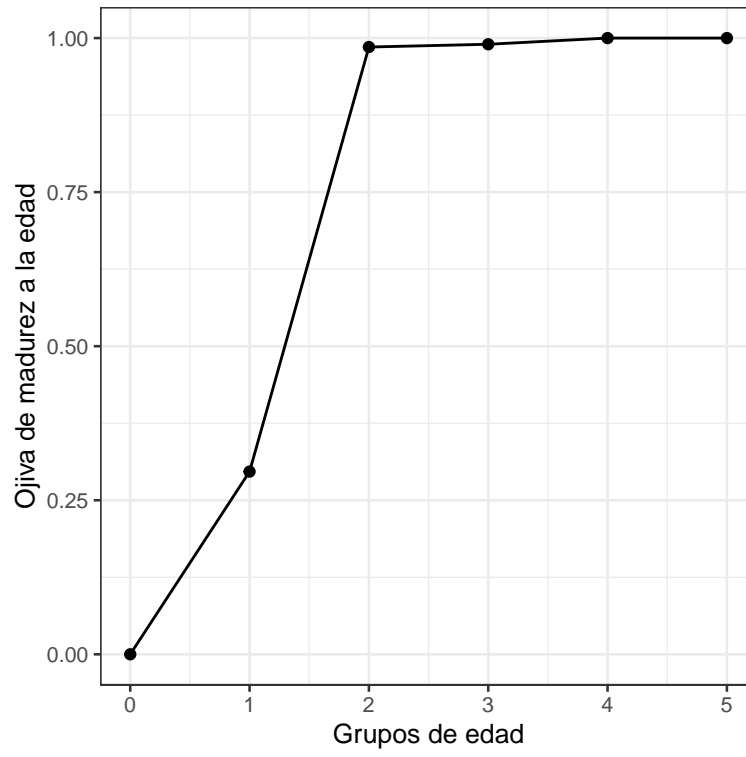


### 5.5.1.3 Ojiva de madurez por edad.

```
dir(dir_GSA1)[10]
## [1] "PROPMAT.DAT"
PROPMAT.DAT <- read.table(paste(dir_GSA1,dir(dir_GSA1)[10],sep="/"),
                           sep=" ",na="NA",fill=T,skip = 5)

ojivamadurez<-data.frame(edad=seq(0,5,1),
                          prop=as.numeric(PROPMAT.DAT[1,]))

ggplot(ojivamadurez,aes(edad,prop))+
  geom_line()+
  geom_point()+
  labs(x="Grupos de edad", y="Ojiva de madurez a la edad") +
  theme(panel.background = element_rect(fill ="gray95")) +
  theme_bw(base_size=10)
```



### 5.5.2 Formato SS3

```
wtat <- r4ss::SS_read(here("empirical_wtatage_and_age_selex"),
                     verbose = FALSE) #base
wtat1<-wtat$wtatage # para modificar
#names(wtat1) # muestra los objetos de la lista

# if Yr is negative, then fill remaining years for that Seas, growpattern, Bio_Pattern, Fleet
# if season is negative, then fill remaining fleets for that Seas, Bio_Pattern, Sex, Fleet
# will fill through forecast years, so be careful
# fleet 0 contains begin season pop WT
# fleet -1 contains mid season pop WT
# fleet -2 contains maturity*fecundity
#Yr Seas Sex Bio_Pattern BirthSeas Fleet

# Arreglo de datos

maturity1<-as.numeric(PROPMAT.DAT[1,])
data_wt<-as.data.frame(CATWT.DAT)

wt_flt<-data.frame(do.call(rbind,replicate(4,data_wt,simplify = FALSE))) %>% mutate(type=c(rep("#wt_flt",4)),
colnames(wt_flt)<-c("E0", "E1", "E2", "E3", "E4", "E5", "#type")

fecundity<-data.frame(do.call(rbind,replicate(nyears,maturity1,simplify = FALSE))) %>% mutate(type=c(rep("#fecundity",nyears)),
colnames(fecundity)<-c("E0", "E1", "E2", "E3", "E4", "E5", "#type")

popwt<-data.frame(do.call(rbind,replicate(2,data_wt,simplify = FALSE))) %>% mutate(type=c(rep("#popwt_b",2)),
colnames(popwt)<-c("E0", "E1", "E2", "E3", "E4", "E5", "#type")

wtatege_f1<-data.frame(Yr=c(rep(years,7)),
                      Seas=1,
                      Sex=1,
                      Bio_Pattern=1,
                      BirthSeas=1,
                      Fleet=c(rep(1,nyears),rep(2,nyears),rep(3,nyears),rep(4,nyears),
                              rep(-2,nyears),
                              rep(0,nyears),rep(-1,nyears)))

wtatege_f1<-cbind(wtatege_f1,rbind(wt_flt,fecundity,popwt))

wtatege_f1[order(wtatege_f1$Yr),]
##      Yr Seas Sex Bio_Pattern BirthSeas Fleet    E0    E1    E2    E3    E4
## 1  2003    1  1      1          1      1  0.032 0.1190 0.5170 1.178 2.086
## 20 2003    1  1      1          1      2  0.032 0.1190 0.5170 1.178 2.086
## 39 2003    1  1      1          1      3  0.032 0.1190 0.5170 1.178 2.086
## 58 2003    1  1      1          1      4  0.032 0.1190 0.5170 1.178 2.086
## 77 2003    1  1      1          1      -2 0.000 0.2965 0.9855 0.990 1.000
## 96 2003    1  1      1          1      0  0.032 0.1190 0.5170 1.178 2.086
## 115 2003    1  1      1          1     -1 0.032 0.1190 0.5170 1.178 2.086
## 2  2004    1  1      1          1      1  0.030 0.1090 0.4710 1.243 1.964
## 21 2004    1  1      1          1      2  0.030 0.1090 0.4710 1.243 1.964
## 40 2004    1  1      1          1      3  0.030 0.1090 0.4710 1.243 1.964
## 59 2004    1  1      1          1      4  0.030 0.1090 0.4710 1.243 1.964
```

## 78	2004	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 97	2004	1	1	1	1	0	0.030	0.1090	0.4710	1.243	1.964
## 116	2004	1	1	1	1	-1	0.030	0.1090	0.4710	1.243	1.964
## 3	2005	1	1	1	1	1	0.031	0.1250	0.5210	1.127	2.024
## 22	2005	1	1	1	1	2	0.031	0.1250	0.5210	1.127	2.024
## 41	2005	1	1	1	1	3	0.031	0.1250	0.5210	1.127	2.024
## 60	2005	1	1	1	1	4	0.031	0.1250	0.5210	1.127	2.024
## 79	2005	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 98	2005	1	1	1	1	0	0.031	0.1250	0.5210	1.127	2.024
## 117	2005	1	1	1	1	-1	0.031	0.1250	0.5210	1.127	2.024
## 4	2006	1	1	1	1	1	0.023	0.1360	0.4970	1.100	1.999
## 23	2006	1	1	1	1	2	0.023	0.1360	0.4970	1.100	1.999
## 42	2006	1	1	1	1	3	0.023	0.1360	0.4970	1.100	1.999
## 61	2006	1	1	1	1	4	0.023	0.1360	0.4970	1.100	1.999
## 80	2006	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 99	2006	1	1	1	1	0	0.023	0.1360	0.4970	1.100	1.999
## 118	2006	1	1	1	1	-1	0.023	0.1360	0.4970	1.100	1.999
## 5	2007	1	1	1	1	1	0.021	0.1390	0.5110	1.106	2.037
## 24	2007	1	1	1	1	2	0.021	0.1390	0.5110	1.106	2.037
## 43	2007	1	1	1	1	3	0.021	0.1390	0.5110	1.106	2.037
## 62	2007	1	1	1	1	4	0.021	0.1390	0.5110	1.106	2.037
## 81	2007	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 100	2007	1	1	1	1	0	0.021	0.1390	0.5110	1.106	2.037
## 119	2007	1	1	1	1	-1	0.021	0.1390	0.5110	1.106	2.037
## 6	2008	1	1	1	1	1	0.028	0.1450	0.5150	1.203	2.147
## 25	2008	1	1	1	1	2	0.028	0.1450	0.5150	1.203	2.147
## 44	2008	1	1	1	1	3	0.028	0.1450	0.5150	1.203	2.147
## 63	2008	1	1	1	1	4	0.028	0.1450	0.5150	1.203	2.147
## 82	2008	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 101	2008	1	1	1	1	0	0.028	0.1450	0.5150	1.203	2.147
## 120	2008	1	1	1	1	-1	0.028	0.1450	0.5150	1.203	2.147
## 7	2009	1	1	1	1	1	0.025	0.1330	0.5200	1.086	2.046
## 26	2009	1	1	1	1	2	0.025	0.1330	0.5200	1.086	2.046
## 45	2009	1	1	1	1	3	0.025	0.1330	0.5200	1.086	2.046
## 64	2009	1	1	1	1	4	0.025	0.1330	0.5200	1.086	2.046
## 83	2009	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 102	2009	1	1	1	1	0	0.025	0.1330	0.5200	1.086	2.046
## 121	2009	1	1	1	1	-1	0.025	0.1330	0.5200	1.086	2.046
## 8	2010	1	1	1	1	1	0.024	0.1700	0.4740	1.116	1.907
## 27	2010	1	1	1	1	2	0.024	0.1700	0.4740	1.116	1.907
## 46	2010	1	1	1	1	3	0.024	0.1700	0.4740	1.116	1.907
## 65	2010	1	1	1	1	4	0.024	0.1700	0.4740	1.116	1.907
## 84	2010	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 103	2010	1	1	1	1	0	0.024	0.1700	0.4740	1.116	1.907
## 122	2010	1	1	1	1	-1	0.024	0.1700	0.4740	1.116	1.907
## 9	2011	1	1	1	1	1	0.028	0.1470	0.4860	1.174	2.059
## 28	2011	1	1	1	1	2	0.028	0.1470	0.4860	1.174	2.059
## 47	2011	1	1	1	1	3	0.028	0.1470	0.4860	1.174	2.059
## 66	2011	1	1	1	1	4	0.028	0.1470	0.4860	1.174	2.059
## 85	2011	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 104	2011	1	1	1	1	0	0.028	0.1470	0.4860	1.174	2.059
## 123	2011	1	1	1	1	-1	0.028	0.1470	0.4860	1.174	2.059
## 10	2012	1	1	1	1	1	0.029	0.1460	0.4650	1.106	2.087

## 29	2012	1	1	1	1	2	0.029	0.1460	0.4650	1.106	2.087
## 48	2012	1	1	1	1	3	0.029	0.1460	0.4650	1.106	2.087
## 67	2012	1	1	1	1	4	0.029	0.1460	0.4650	1.106	2.087
## 86	2012	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 105	2012	1	1	1	1	0	0.029	0.1460	0.4650	1.106	2.087
## 124	2012	1	1	1	1	-1	0.029	0.1460	0.4650	1.106	2.087
## 11	2013	1	1	1	1	1	0.021	0.1640	0.4810	1.152	2.057
## 30	2013	1	1	1	1	2	0.021	0.1640	0.4810	1.152	2.057
## 49	2013	1	1	1	1	3	0.021	0.1640	0.4810	1.152	2.057
## 68	2013	1	1	1	1	4	0.021	0.1640	0.4810	1.152	2.057
## 87	2013	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 106	2013	1	1	1	1	0	0.021	0.1640	0.4810	1.152	2.057
## 125	2013	1	1	1	1	-1	0.021	0.1640	0.4810	1.152	2.057
## 12	2014	1	1	1	1	1	0.026	0.1620	0.4880	1.149	1.873
## 31	2014	1	1	1	1	2	0.026	0.1620	0.4880	1.149	1.873
## 50	2014	1	1	1	1	3	0.026	0.1620	0.4880	1.149	1.873
## 69	2014	1	1	1	1	4	0.026	0.1620	0.4880	1.149	1.873
## 88	2014	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 107	2014	1	1	1	1	0	0.026	0.1620	0.4880	1.149	1.873
## 126	2014	1	1	1	1	-1	0.026	0.1620	0.4880	1.149	1.873
## 13	2015	1	1	1	1	1	0.025	0.1520	0.4810	1.157	1.896
## 32	2015	1	1	1	1	2	0.025	0.1520	0.4810	1.157	1.896
## 51	2015	1	1	1	1	3	0.025	0.1520	0.4810	1.157	1.896
## 70	2015	1	1	1	1	4	0.025	0.1520	0.4810	1.157	1.896
## 89	2015	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 108	2015	1	1	1	1	0	0.025	0.1520	0.4810	1.157	1.896
## 127	2015	1	1	1	1	-1	0.025	0.1520	0.4810	1.157	1.896
## 14	2016	1	1	1	1	1	0.022	0.1390	0.4790	1.123	1.843
## 33	2016	1	1	1	1	2	0.022	0.1390	0.4790	1.123	1.843
## 52	2016	1	1	1	1	3	0.022	0.1390	0.4790	1.123	1.843
## 71	2016	1	1	1	1	4	0.022	0.1390	0.4790	1.123	1.843
## 90	2016	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 109	2016	1	1	1	1	0	0.022	0.1390	0.4790	1.123	1.843
## 128	2016	1	1	1	1	-1	0.022	0.1390	0.4790	1.123	1.843
## 15	2017	1	1	1	1	1	0.024	0.1290	0.4810	1.152	1.851
## 34	2017	1	1	1	1	2	0.024	0.1290	0.4810	1.152	1.851
## 53	2017	1	1	1	1	3	0.024	0.1290	0.4810	1.152	1.851
## 72	2017	1	1	1	1	4	0.024	0.1290	0.4810	1.152	1.851
## 91	2017	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 110	2017	1	1	1	1	0	0.024	0.1290	0.4810	1.152	1.851
## 129	2017	1	1	1	1	-1	0.024	0.1290	0.4810	1.152	1.851
## 16	2018	1	1	1	1	1	0.022	0.1400	0.4850	1.132	1.801
## 35	2018	1	1	1	1	2	0.022	0.1400	0.4850	1.132	1.801
## 54	2018	1	1	1	1	3	0.022	0.1400	0.4850	1.132	1.801
## 73	2018	1	1	1	1	4	0.022	0.1400	0.4850	1.132	1.801
## 92	2018	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000
## 111	2018	1	1	1	1	0	0.022	0.1400	0.4850	1.132	1.801
## 130	2018	1	1	1	1	-1	0.022	0.1400	0.4850	1.132	1.801
## 17	2019	1	1	1	1	1	0.029	0.1530	0.5040	1.073	1.820
## 36	2019	1	1	1	1	2	0.029	0.1530	0.5040	1.073	1.820
## 55	2019	1	1	1	1	3	0.029	0.1530	0.5040	1.073	1.820
## 74	2019	1	1	1	1	4	0.029	0.1530	0.5040	1.073	1.820
## 93	2019	1	1	1	1	-2	0.000	0.2965	0.9855	0.990	1.000

```

## 112 2019      1      1      1      1      0 0.029 0.1530 0.5040 1.073 1.820
## 131 2019      1      1      1      1     -1 0.029 0.1530 0.5040 1.073 1.820
## 18  2020      1      1      1      1      1 0.019 0.1360 0.4870 1.122 1.747
## 37  2020      1      1      1      1      2 0.019 0.1360 0.4870 1.122 1.747
## 56  2020      1      1      1      1      3 0.019 0.1360 0.4870 1.122 1.747
## 75  2020      1      1      1      1      4 0.019 0.1360 0.4870 1.122 1.747
## 94  2020      1      1      1      1     -2 0.000 0.2965 0.9855 0.990 1.000
## 113 2020      1      1      1      1      0 0.019 0.1360 0.4870 1.122 1.747
## 132 2020      1      1      1      1     -1 0.019 0.1360 0.4870 1.122 1.747
## 19  2021      1      1      1      1      1 0.023 0.1150 0.4770 1.107 1.781
## 38  2021      1      1      1      1      2 0.023 0.1150 0.4770 1.107 1.781
## 57  2021      1      1      1      1      3 0.023 0.1150 0.4770 1.107 1.781
## 76  2021      1      1      1      1      4 0.023 0.1150 0.4770 1.107 1.781
## 95  2021      1      1      1      1     -2 0.000 0.2965 0.9855 0.990 1.000
## 114 2021      1      1      1      1      0 0.023 0.1150 0.4770 1.107 1.781
## 133 2021      1      1      1      1     -1 0.023 0.1150 0.4770 1.107 1.781
##      E5      #type
## 1    3.529 #wt_flt_1
## 20   3.529 #wt_flt_2
## 39   3.529 #wt_flt_3
## 58   3.529 #wt_flt_4
## 77   1.000 #fecundity
## 96   3.529 #popwt_beg
## 115  3.529 #popwt_mid
## 2    3.529 #wt_flt_1
## 21   3.529 #wt_flt_2
## 40   3.529 #wt_flt_3
## 59   3.529 #wt_flt_4
## 78   1.000 #fecundity
## 97   3.529 #popwt_beg
## 116  3.529 #popwt_mid
## 3    3.388 #wt_flt_1
## 22   3.388 #wt_flt_2
## 41   3.388 #wt_flt_3
## 60   3.388 #wt_flt_4
## 79   1.000 #fecundity
## 98   3.388 #popwt_beg
## 117  3.388 #popwt_mid
## 4    2.775 #wt_flt_1
## 23   2.775 #wt_flt_2
## 42   2.775 #wt_flt_3
## 61   2.775 #wt_flt_4
## 80   1.000 #fecundity
## 99   2.775 #popwt_beg
## 118  2.775 #popwt_mid
## 5    2.775 #wt_flt_1
## 24   2.775 #wt_flt_2
## 43   2.775 #wt_flt_3
## 62   2.775 #wt_flt_4
## 81   1.000 #fecundity
## 100  2.775 #popwt_beg
## 119  2.775 #popwt_mid
## 6    3.083 #wt_flt_1

```



```

## 25 3.083 #wt_flt_2
## 44 3.083 #wt_flt_3
## 63 3.083 #wt_flt_4
## 82 1.000 #fecundity
## 101 3.083 #popwt_beg
## 120 3.083 #popwt_mid
## 7 2.637 #wt_flt_1
## 26 2.637 #wt_flt_2
## 45 2.637 #wt_flt_3
## 64 2.637 #wt_flt_4
## 83 1.000 #fecundity
## 102 2.637 #popwt_beg
## 121 2.637 #popwt_mid
## 8 3.045 #wt_flt_1
## 27 3.045 #wt_flt_2
## 46 3.045 #wt_flt_3
## 65 3.045 #wt_flt_4
## 84 1.000 #fecundity
## 103 3.045 #popwt_beg
## 122 3.045 #popwt_mid
## 9 2.875 #wt_flt_1
## 28 2.875 #wt_flt_2
## 47 2.875 #wt_flt_3
## 66 2.875 #wt_flt_4
## 85 1.000 #fecundity
## 104 2.875 #popwt_beg
## 123 2.875 #popwt_mid
## 10 2.875 #wt_flt_1
## 29 2.875 #wt_flt_2
## 48 2.875 #wt_flt_3
## 67 2.875 #wt_flt_4
## 86 1.000 #fecundity
## 105 2.875 #popwt_beg
## 124 2.875 #popwt_mid
## 11 3.257 #wt_flt_1
## 30 3.257 #wt_flt_2
## 49 3.257 #wt_flt_3
## 68 3.257 #wt_flt_4
## 87 1.000 #fecundity
## 106 3.257 #popwt_beg
## 125 3.257 #popwt_mid
## 12 3.529 #wt_flt_1
## 31 3.529 #wt_flt_2
## 50 3.529 #wt_flt_3
## 69 3.529 #wt_flt_4
## 88 1.000 #fecundity
## 107 3.529 #popwt_beg
## 126 3.529 #popwt_mid
## 13 3.460 #wt_flt_1
## 32 3.460 #wt_flt_2
## 51 3.460 #wt_flt_3
## 70 3.460 #wt_flt_4
## 89 1.000 #fecundity

```

```

## 108 3.460 #popwt_beg
## 127 3.460 #popwt_mid
## 14 2.875 #wt_flt_1
## 33 2.875 #wt_flt_2
## 52 2.875 #wt_flt_3
## 71 2.875 #wt_flt_4
## 90 1.000 #fecundity
## 109 2.875 #popwt_beg
## 128 2.875 #popwt_mid
## 15 2.875 #wt_flt_1
## 34 2.875 #wt_flt_2
## 53 2.875 #wt_flt_3
## 72 2.875 #wt_flt_4
## 91 1.000 #fecundity
## 110 2.875 #popwt_beg
## 129 2.875 #popwt_mid
## 16 2.875 #wt_flt_1
## 35 2.875 #wt_flt_2
## 54 2.875 #wt_flt_3
## 73 2.875 #wt_flt_4
## 92 1.000 #fecundity
## 111 2.875 #popwt_beg
## 130 2.875 #popwt_mid
## 17 2.754 #wt_flt_1
## 36 2.754 #wt_flt_2
## 55 2.754 #wt_flt_3
## 74 2.754 #wt_flt_4
## 93 1.000 #fecundity
## 112 2.754 #popwt_beg
## 131 2.754 #popwt_mid
## 18 2.754 #wt_flt_1
## 37 2.754 #wt_flt_2
## 56 2.754 #wt_flt_3
## 75 2.754 #wt_flt_4
## 94 1.000 #fecundity
## 113 2.754 #popwt_beg
## 132 2.754 #popwt_mid
## 19 2.754 #wt_flt_1
## 38 2.754 #wt_flt_2
## 57 2.754 #wt_flt_3
## 76 2.754 #wt_flt_4
## 95 1.000 #fecundity
## 114 2.754 #popwt_beg
## 133 2.754 #popwt_mid

```

#### 5.5.2.1 Vector de mortalidad Natural por edad.

```
dir(dir_GSA1)[7]
## [1] "NATMOR.DAT"
NATMOR.DAT <- read.table(paste(dir_GSA1,dir(dir_GSA1)[7],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)
NATMOR.DAT
##      V1 V2 V3 V4 V5 V6
## 1 1.9 0.7 0.39 0.29 0.23 0.2
```

## 6 GSA6 (Levante-Norte de España).

La conceptualización del modelo biológico de **GSA6** considera los siguientes componentes de la dinámica poblacional:

- Estructura geográfica
- Reproducción
- Reclutamiento
- Mortalidad natural
- Crecimiento
- otros...

### 6.1 Descripción del modelo

El modelo de evaluación de stock de **GSA6** se basa en un análisis estadístico de la dinámica de estructuras de edad anual que incorpora información biológica y pesquera. La información que ingresa al modelo consiste en ...

- Años de evaluación de stock = 2002\_2021
- número de edades = 6 años ?

## 6.2 Archivos utilizado para enfoque de modelación a4a

```
dir_GSA6 <- here("hke_GSA6_a4a_format")
dir(dir_GSA6)
## [1] "CATCH.DAT"
## [2] "CATNUM.DAT"
## [3] "CATWT.DAT"
## [4] "Definiciones_Ficheros imput_a4a.docx"
## [5] "INCHECK.TXT"
## [6] "LOWIND.DAT"
## [7] "NATMOR.DAT"
## [8] "PROPF.DAT"
## [9] "PROPM.DAT"
## [10] "PROPMAT.DAT"
## [11] "STOCWT.DAT"
## [12] "TUNEFF.DAT"
```

### 6.2.0.1 Capturas anuales del stock (toneladas)

```
dir(dir_GSA6)[1]
## [1] "CATCH.DAT"
CATCH.DAT <- read.table(paste(dir_GSA6,dir(dir_GSA6)[1],sep="/"),
                        header=T,sep=" ",na="NA",fill=T,skip = 4)
CATCH.DAT
##      X5
## 1 2834.9
## 2 4632.7
## 3 3150.5
## 4 3473.4
## 5 3627.2
## 6 2539.8
## 7 3340.9
## 8 3847.2
## 9 2821.6
## 10 3181.7
## 11 2641.4
## 12 2949.9
## 13 2489.3
## 14 1726.1
## 15 1809.8
## 16 1728.2
## 17 2442.5
## 18 1628.1
## 19 1099.3
## 20 1531.0
```

**6.2.0.2 índices de abundancia para calibrar (CPUEs o Campañas, o ambas).** Normalmente para índices de campaña empleamos número/KM2.

```
dir(dir_GSA6)[12]
## [1] "TUNEFF.DAT"
TUNEFF.DAT <- read.table(paste(dir_GSA6,dir(dir_GSA6)[12],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 6)
TUNEFF.DAT
```

```
##      V1      V2      V3      V4      V5      V6      V7
## 1  1 2119.2 101.1  7.4  1.7  NA  NA
## 2  1 2051.7 132.6 10.8  1.0  0.1  0.2
## 3  1 4189.4  90.6 23.5  1.8  NA  NA
## 4  1 3454.8  70.4  5.9  0.5  NA  NA
## 5  1 6054.1 168.2 27.2  0.7  0.7  NA
## 6  1 1565.0  56.4  6.3  1.5  0.1  NA
## 7  1 1799.5  74.8  2.7  1.7  0.2  NA
## 8  1 2188.2 114.0  7.6  1.0  0.4  NA
## 9  1 2824.6  62.9  5.7  1.0  NA  NA
## 10 1  917.2 132.3  5.9  0.3  NA  NA
## 11 1 1333.2  55.1  4.3  0.9  0.2  NA
## 12 1 1263.2 137.6  8.7  0.3  NA  NA
## 13 1  701.9  73.3  9.7  0.9  0.1  0.2
## 14 1  935.1  31.3  7.7  0.8  0.2  NA
## 15 1 1242.1  52.0  5.6  0.7  0.2  NA
## 16 1  931.3  72.7 10.1  0.5  0.3  NA
## 17 1  602.6  89.0  6.1  NA  0.2  NA
## 18 1  530.9  42.2 13.3  0.3  0.1  0.2
## 19 1  623.8  56.8  8.8  0.3  0.1  0.1
## 20 1 1875.8  79.5  2.8  0.7  0.3  NA
```

### 6.2.0.3 Matriz de número de individuos por edad/año de las capturas (Miles de individuos)

```
dir(dir_GSA6)[2]
## [1] "CATNUM.DAT"
CATNUM.DAT <- read.table(paste(dir_GSA6,dir(dir_GSA6)[2],sep="/"),
                          sep=" ",na="NA",fill=T,skip = 5)
CATNUM.DAT
##      V1      V2      V3      V4      V5      V6
## 1 109196.9 6665.5 1248.3 277.6 132.8 33.2
## 2  72755.2 29484.8 1855.7 179.8  58.3  7.7
## 3  61686.9 14773.6 1219.5 173.2  27.0 17.9
## 4  39457.7 16612.0 2048.1 349.0  37.5  3.0
## 5  53949.5 16097.8 2050.5 380.7  32.4  6.4
## 6  34983.3  8333.9 1804.8 310.0  54.4 12.8
## 7  61690.5 17293.3 1430.4 171.1  62.2  3.9
## 8  68385.7 18710.1 1619.0 264.6  60.0  1.7
## 9  11268.7 14311.1 2250.0 215.1  46.2  2.1
## 10  7536.9 18339.0 2469.5 227.8  16.7  5.1
## 11  9210.2 18153.2 1641.6 158.1  23.9  1.3
## 12  6636.3 19209.9 1886.4 152.7   9.5  2.1
## 13  7376.4 11887.3 2232.3 137.7   9.5  2.0
## 14  4831.9  9390.1 1350.5 105.8  13.2  0.1
## 15  7948.0 12312.3  989.8  91.4  13.4  0.6
## 16  2752.8 10938.7 1219.6 115.8  16.7  2.0
## 17  3727.2 16840.1 1518.9 156.6   9.5  0.5
## 18   593.7  7251.4 1513.8 186.3  12.3  2.6
## 19 1894.6  7109.8  770.7  67.8   4.5  0.6
## 20   875.9 10995.6  842.9  81.4  27.2  3.9
```

### 6.2.0.4 Peso medio por edad y año de tu matriz de captura (Kilos).

```

dir(dir_GSA6)[3]
## [1] "CATWT.DAT"
CATWT.DAT <- read.table(paste(dir_GSA6,dir(dir_GSA6)[3],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)

CATWT.DAT
##      V1      V2      V3      V4      V5      V6
## 1 0.011 0.111 0.436 0.996 1.623 2.391
## 2 0.016 0.094 0.407 0.967 1.627 2.646
## 3 0.018 0.103 0.400 0.929 1.630 4.050
## 4 0.019 0.104 0.398 0.928 1.656 2.291
## 5 0.018 0.101 0.411 0.926 1.619 3.827
## 6 0.019 0.104 0.415 0.967 1.631 3.046
## 7 0.017 0.097 0.391 0.980 1.656 2.496
## 8 0.022 0.084 0.449 0.983 1.488 5.082
## 9 0.018 0.109 0.414 0.925 1.712 2.424
## 10 0.026 0.105 0.405 0.903 1.747 2.646
## 11 0.026 0.095 0.402 0.902 1.672 2.596
## 12 0.028 0.106 0.387 0.933 1.598 2.664
## 13 0.022 0.120 0.401 0.911 1.622 2.634
## 14 0.024 0.112 0.400 0.908 1.579 2.188
## 15 0.026 0.098 0.401 0.959 1.611 2.309
## 16 0.030 0.103 0.391 0.945 1.585 2.652
## 17 0.030 0.101 0.402 0.925 1.618 2.335
## 18 0.031 0.123 0.399 0.915 1.705 2.578
## 19 0.029 0.103 0.398 0.888 1.639 2.624
## 20 0.034 0.102 0.389 0.975 1.673 2.545

```

6.2.0.5 Medio por edad y año asumida para el stock (Normalmente = CATWT) (Kilos).

```

dir(dir_GSA6)[11]
## [1] "STOCWT.DAT"
STOCWT.DAT <- read.table(paste(dir_GSA6,dir(dir_GSA6)[11],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)

STOCWT.DAT
##      V1      V2      V3      V4      V5      V6
## 1 0.011 0.111 0.436 0.996 1.623 2.391
## 2 0.016 0.094 0.407 0.967 1.627 2.646
## 3 0.018 0.103 0.400 0.929 1.630 4.050
## 4 0.019 0.104 0.398 0.928 1.656 2.291
## 5 0.018 0.101 0.411 0.926 1.619 3.827
## 6 0.019 0.104 0.415 0.967 1.631 3.046
## 7 0.017 0.097 0.391 0.980 1.656 2.496
## 8 0.022 0.084 0.449 0.983 1.488 5.082
## 9 0.018 0.109 0.414 0.925 1.712 2.424
## 10 0.026 0.105 0.405 0.903 1.747 2.646
## 11 0.026 0.095 0.402 0.902 1.672 2.596
## 12 0.028 0.106 0.387 0.933 1.598 2.664
## 13 0.022 0.120 0.401 0.911 1.622 2.634
## 14 0.024 0.112 0.400 0.908 1.579 2.188
## 15 0.026 0.098 0.401 0.959 1.611 2.309
## 16 0.030 0.103 0.391 0.945 1.585 2.652
## 17 0.030 0.101 0.402 0.925 1.618 2.335
## 18 0.031 0.123 0.399 0.915 1.705 2.578

```

```
## 19 0.029 0.103 0.398 0.888 1.639 2.624
## 20 0.034 0.102 0.389 0.975 1.673 2.545
```

#### 6.2.0.6 Vector de mortalidad Natural por edad.

```
dir(dir_GSA6)[7]
## [1] "NATMOR.DAT"
NATMOR.DAT <- read.table(paste(dir_GSA6,dir(dir_GSA6)[7],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)
NATMOR.DAT
##      V1      V2      V3      V4      V5      V6
## 1 1.8 0.72 0.41 0.3 0.24 0.18
```

#### 6.2.0.7 Ogiva de madurez por edad.

```
dir(dir_GSA6)[10]
## [1] "PROPMAT.DAT"
PROPMAT.DAT <- read.table(paste(dir_GSA6,dir(dir_GSA6)[10],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)
PROPMAT.DAT
##      V1      V2      V3      V4      V5      V6
## 1 0 0.2965 0.9855 0.99 1 1
```