

Contents

1	Contexto	2
2	Tareas a desarrollar.	2
3	Temporalización y dinámica de las actividades propuestas.	2
4	Análisis de datos de los stocks de merluza europea (<i>Merluccius merluccius</i>) de la GSA1 (mar de Alborán) y GSA6 (Levante-Norte de España).	2
4.1	Leer datos formato a4a	2
5	Escribir archivo dat modificado con la función SS_write	15

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).

3 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.

4 Análisis de datos de los stocks de merluza europea (*Merluccius merluccius*) de la GSA1 (mar de Alborán) y GSA6 (Levante-Norte de España).

4.1 Leer datos formato a4a

El modelo SCAA a4a está implementado en FLR (https://flr-project.org/doc/Loading_your_data_into_FLR.html), por lo tanto la arquitectura de sus ficheros input sigue su 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 imput, 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 imputs:

- 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.
- PROP: 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 imput 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.

4.1.1 GSA1 (mar de Alborán)

4.1.1.1 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...

4.1.1.2 Descripción del modelo

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 ?

4.1.1.3 Archivos utilizado para enfoque de modelación a4a

```
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"
```

4.1.1.4 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
```

4.1.2 Información general del modelo

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)

#Especificaciones iniciales
dat1$styr <-1971 #_StartYr
dat1$endyr <-2001 #_EndYr
dat1$nseas <-1 #_Nseas
dat1$months_per_seas<-12 #_months/season
dat1$Nsubseasons <-2 #_Nsubseasons (even number, minimum is 2)
dat1$spawn_month <-1 #_spawn_month
dat1$Ngenders <-2 #_Ngenders: 1, 2, -1 (use -1 for 1 sex setup with SSB multiplied by
#female_frac parameter)
dat1$Nages <-40 #_Nages=accumulator age, first age is always age 0
dat1$N_areas <-1 #_Nareas
dat1$Nfleets <-3 #_Nfleets (including surveys)
```

4.1.2.1 Capturas anuales del stock (toneladas)

1. formato a4a

```
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
```

2. formato SS3

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

```
#_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

# Modificar con los datos propios
fleetnames1 <-c("FISHERY", "SURVEY1", "SURVEY2")      #character vector
surveytiming1 <-c(-1,1,1)#numeric vector
units_of_catch1<-c(1,2,2) #numeric vector
areas1 <-c(1,1,1) #numeric vector

# crear data.frame para fleetinfo
fleetinfo1<-data.frame(type = c(1,3,3),
                        surveytiming =surveytiming1,
                        area=areas1,
                        units=units_of_catch1,
                        need_catch_mult =c(0,0,0),
                        fleetname=fleetnames1)

dat1$fleetinfo <- fleetinfo1      #data.frame
dat1$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
```

A continuación, se especifica la captura.

Consulte la Guía de usuario de SS3: Sección 7.9 “Catch”.

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

# Datos de modelo simple
year<-1971:2001
nyear<-length(year)
catch_year<-c(-999,year)
catch_seas<-rep(1,nyear+1)
catch_fleet<-rep(1,nyear+1)
catch_catch<-dat$catch$catch #datos ss3 reemplazar por datos gsa1
catch_catch_se<-rep(0.01,nyear+1)

# Modificar con los datos propios
catch1<-data.frame(year=catch_year,
                    seas=catch_seas,
                    fleet=catch_fleet,
                    catch=catch_catch,
                    catch_se=catch_catch_se)
dat1$catch<-catch1 # data.frame
```

```

dat1$catch
##      year seas fleet catch catch_se
## 1  -999     1     1     0    0.01
## 2  1971     1     1     0    0.01
## 3  1972     1     1    200    0.01
## 4  1973     1     1   1000    0.01
## 5  1974     1     1   1000    0.01
## 6  1975     1     1   2000    0.01
## 7  1976     1     1   3000    0.01
## 8  1977     1     1   4000    0.01
## 9  1978     1     1   5000    0.01
## 10 1979     1     1   6000    0.01
## 11 1980     1     1   8000    0.01
## 12 1981     1     1  10000    0.01
## 13 1982     1     1  10000    0.01
## 14 1983     1     1  10000    0.01
## 15 1984     1     1  10000    0.01
## 16 1985     1     1  10000    0.01
## 17 1986     1     1  10000    0.01
## 18 1987     1     1  10000    0.01
## 19 1988     1     1   9000    0.01
## 20 1989     1     1   8000    0.01
## 21 1990     1     1   7000    0.01
## 22 1991     1     1   6000    0.01
## 23 1992     1     1   4000    0.01
## 24 1993     1     1   4000    0.01
## 25 1994     1     1   4000    0.01
## 26 1995     1     1   4000    0.01
## 27 1996     1     1   4000    0.01
## 28 1997     1     1   3000    0.01
## 29 1998     1     1   3000    0.01
## 30 1999     1     1   3000    0.01
## 31 2000     1     1   3000    0.01
## 32 2001     1     1   3000    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.

4.1.2.2 índices de abundancia para calibrar (CPUEs o Campañas, o ambas). Normalmente para índices de campaña empleamos número/KM2. 1. formato a4a

```
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
```

2. 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 “Indices”.

```
#_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

# Datos de modelo simple
CPUEinfo_Fleet <-c(1,2,3)
CPUEinfo_Units <-c(1,1,0)
CPUEinfo_Errtype <-c(0,0,0)
CPUEinfo_SD_Report<-c(0,1,0)
CPUEinfo_names <-c("FISHERY", "SURVEY1", "SURVEY2")

# Modificar con los datos propios
CPUEinfo1<-data.frame(Fleet=CPUEinfo_Fleet,
                      Units=CPUEinfo_Units,
                      Errtype=CPUEinfo_Errtype,
                      SD_Report=CPUEinfo_SD_Report)

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



```
dat1$CPUEinfo<-CPUEinfo1 # data.frame
dat1$CPUEinfo
##      Fleet Units Errtype SD_Report
## FISHERY      1      1      0      0
## SURVEY1      2      1      0      1
## SURVEY2      3      0      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 la campaña 1 tienen unidades de biomasa, mientras que la campaña 2 es en números. Se asume un error logarítmico normal para las 3 flotas.
- Inmediatamente después de su encabezado, se incluyen los datos de índices de abundancia:

```
# Datos de modelo simple
yearf1<-c(seq(1977,1998,3),2001)
yearf2<-seq(1990,2001,1)

CPUE_year <-c(yearf1,yearf2)
CPUE_seas <-rep(7,length(CPUE_year))
CPUE_index <-c(rep(2,length(yearf1)),rep(3,length(yearf2)))
CPUE_obs <-dat$CPUE$obs
CPUE_se_log <-c(rep(0.3,length(yearf1)),rep(0.7,length(yearf2)))

# Modificar con los datos propios
CPUE1<-data.frame(year=CPUE_year,
                  seas=CPUE_seas,
                  index=CPUE_index,
                  obs=dat$CPUE$obs,
                  se_log=CPUE_se_log)

dat1$CPUE<-CPUE1 # data.frame
dat1$CPUE
##   year seas index      obs se_log
## 1  1977    7     2 3.39689e+05  0.3
## 2  1980    7     2 1.93353e+05  0.3
## 3  1983    7     2 1.51984e+05  0.3
## 4  1986    7     2 5.52218e+04  0.3
## 5  1989    7     2 5.92323e+04  0.3
## 6  1992    7     2 3.11375e+04  0.3
## 7  1995    7     2 3.58454e+04  0.3
## 8  1998    7     2 2.74926e+04  0.3
## 9  2001    7     2 3.73383e+04  0.3
## 10 1990    7     3 5.19333e+00  0.7
## 11 1991    7     3 1.17840e+00  0.7
## 12 1992    7     3 5.94383e+00  0.7
## 13 1993    7     3 7.70106e-01  0.7
## 14 1994    7     3 1.63180e+01  0.7
## 15 1995    7     3 1.36339e+00  0.7
## 16 1996    7     3 4.76482e+00  0.7
## 17 1997    7     3 5.10707e+01  0.7
```

```
## 18 1998      7      3 1.36095e+00    0.7
## 19 1999      7      3 8.62531e-01    0.7
## 20 2000      7      3 5.97125e+00    0.7
## 21 2001      7      3 1.69379e+00    0.7
```

4.1.3 Descartes y tallas medias

A continuación, se podrían especificar los datos de descartes y tallas media, pero son 0 en este ejemplo.

Consulte la Guía de usuario de SS3: Sección 7.11 “Discard”.

```
# Modificar con los datos propios
dat1$N_discard_fleets <-0 #numeric
dat1$use_meanbodywt  <-0 #numeric
```

4.1.4 Composición de tallas

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”.

```
# set up population length bin structure (note - irrelevant if not using size
#data and using empirical wtatage

# Modificar con los datos propios
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 <-94 # maximum size in the population (lower edge of last bin)
dat1$use_lencompz <-1 # 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):

```
len_info_mintailcomp <-rep(0,3)
len_info_addtocomp   <-rep(1e-07,3)
len_info_combine_M_F <-rep(0,3)
len_info_CompressBins <-rep(0,3)
len_info_CompError    <-rep(0,3)
len_info_ParmSelect   <-rep(0,3)
len_info_minsamplesize <-rep(1,3)

# Modificar con los datos propios
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")

dat1$len_info  <- len_info1 #data.frame
dat1$len_info
##          mintailcomp addtocomp combine_M_F CompressBins CompError ParmSelect
## FISHERY           0    1e-07           0           0           0           0
## SURVEY1           0    1e-07           0           0           0           0
## SURVEY2           0    1e-07           0           0           0           0
##          minsamplesize
## FISHERY              1
## SURVEY1              1
## SURVEY2              1

```

La especificación del vector de tallas:

```

# Modificar con los datos propios
dat1$N_lbins  <-25 #numeric
dat1$lbins_vector <-c(seq(26,64,2),seq(68,80,4),90) #numeric vector

```

Y luego los datos de composición de tallas:

```

# separar por sexo !!!!!

# Modificar con los datos propios
new_lencomp_flt_1 <- data.frame(Yr = 2002:2021,
                                Seas = 7,
                                FltSvy = -1,
                                Gender = 3,
                                Part = 0,
                                Nsamp = 125)

new_lencomp_flt_2 <- data.frame(Yr = seq(2002, 2021, by = 3),
                                Seas = 7,
                                FltSvy = -2,
                                Gender = 3,
                                Part = 0,
                                Nsamp = 125)

dat_rows_names <- colnames(dat$lencomp)[- (1:6)]
dat_rows <- as.data.frame(matrix(data = 1, nrow = nrow(new_lencomp_flt_1)+nrow(new_lencomp_flt_2),
                                ncol = length(dat_rows_names)))
names(dat_rows) <- dat_rows_names

new_lencomp <- rbind(new_lencomp_flt_1, new_lencomp_flt_2)
new_lencomp <- cbind(new_lencomp, dat_rows)

dat1$lencomp <- rbind(dat$lencomp, new_lencomp) #data.frame

```

En este modelo de 2 sexos, los datos de composición de tallas se introducen para las hembras, con 1 número por intervalo de tallas especificado previamente, seguido de los datos de composición de tallas para los machos, también con 1 número por intervalo de tallas.

4.1.4.1 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
```

4.1.5 Composiciones de edad

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”.

Datos de error edad

```
# datos de modelo simple
dat$ageerror
##   age0 age1 age2 age3 age4 age5 age6 age7 age8 age9 age10 age11
## 1 0.500 1.500 2.500 3.500 4.500 5.500 6.500 7.500 8.500 9.500 10.500 11.500
## 2 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001 0.001
## 3 0.500 1.500 2.500 3.500 4.500 5.500 6.500 7.500 8.500 9.500 10.500 11.500
## 4 0.500 0.650 0.670 0.700 0.730 0.760 0.800 0.840 0.880 0.920 0.970 1.030
##   age12 age13 age14 age15 age16 age17 age18 age19 age20 age21 age22
## 1 12.500 13.500 14.500 15.500 16.500 17.500 18.500 19.500 20.500 21.500 22.500
## 2  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001
## 3 12.500 13.500 14.500 15.500 16.500 17.500 18.500 19.500 20.500 21.500 22.500
## 4  1.090  1.160  1.230  1.320  1.410  1.510  1.620  1.750  1.890  2.050  2.230
##   age23 age24 age25 age26 age27 age28 age29 age30 age31 age32 age33
## 1 23.500 24.500 25.500 26.500 27.500 28.500 29.500 30.500 31.500 32.500 33.500
## 2  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001  0.001
## 3 23.500 24.500 25.500 26.500 27.500 28.500 29.500 30.500 31.500 32.500 33.500
## 4  2.450  2.710  3.000  3.000  3.000  3.000  3.000  3.000  3.000  3.000  3.000
##   age34 age35 age36 age37 age38 age39 age40
## 1 34.500 35.500 36.500 37.500 38.500 39.500 40.500
```

```
## 2 0.001 0.001 0.001 0.001 0.001 0.001 0.001
## 3 34.500 35.500 36.500 37.500 38.500 39.500 40.500
## 4 3.000 3.000 3.000 3.000 3.000 3.000 3.000
```

```
# Modificar con los datos propios
dat1$ageerror <- dat$ageerror #data.frame
```

Especificaciones de los datos de composicion de edad

```
# datos de modelo simple
dat$age_info
##          mintailcomp addtocomp combine_M_F CompressBins CompError ParmSelect
## FISHERY           0      1e-07           1           0           0           0
## SURVEY1           0      1e-07           1           0           0           0
## SURVEY2           0      1e-07           1           0           0           0
##          minsamplesize
## FISHERY              1
## SURVEY1              1
## SURVEY2              1

# Modificar con los datos propios
age_info1<-data.frame(mintailcomp=dat$age_info$mintailcomp,
                      addtocomp=dat$age_info$addtocomp,
                      combine_M_F=dat$age_info$combine_M_F,
                      CompressBins=dat$age_info$CompressBins,
                      CompError=dat$age_info$CompError,
                      ParmSelect=dat$age_info$ParmSelect,
                      minsamplesize=dat$age_info$minsamplesize)

row.names(age_info1)<-c("FISHERY", "SURVEY1", "SURVEY2")

dat1$age_info <- age_info1 #data.frame
```

Datos de composicion de edad

```
# Modificar con los datos propios
new_agecomp_flt_1 <- data.frame(Yr=2002:2021,
                               Seas=7,
                               FltSvy=-1,
                               Gender=3,
                               Part=0,
                               Ageerr=2,
                               Lbin_lo=-1,
                               Lbin_hi=-1,
                               Nsamp=75)

new_agecomp_flt_2 <- data.frame(Yr=seq(2002, 2021, by = 3),
                               Seas=7,
                               FltSvy=-2,
                               Gender=3,
                               Part=0,
                               Ageerr=2,
                               Lbin_lo=-1,
                               Lbin_hi=-1,
                               Nsamp=75)
```

```

dat_rows_names <- colnames(dat$agecomp)[- (1:9)]

dat_rows <- as.data.frame(matrix(data = 1, nrow = nrow(new_agecomp_flt_1)+nrow(new_agecomp_flt_2),
                                ncol = length(dat_rows_names)))

names(dat_rows) <- dat_rows_names

new_agecomp <- rbind(new_agecomp_flt_1, new_agecomp_flt_2)
new_agecomp <- cbind(new_agecomp, dat_rows)
dat1$agecomp <- rbind(dat$agecomp, new_agecomp) #data.frame

```

Otros datos

```

# Modificar con los datos propios
dat1$use_MeanSize_at_Age_obs <- 1 #numeric

MeanSize_at_Age_obs_esp<-data.frame(Yr=dat$MeanSize_at_Age_obs$Yr,
                                     Seas=dat$MeanSize_at_Age_obs$Seas,
                                     FltSvy=dat$MeanSize_at_Age_obs$FltSvy,
                                     Gender=dat$MeanSize_at_Age_obs$Gender,
                                     Part=dat$MeanSize_at_Age_obs$Part,
                                     AgeErr=dat$MeanSize_at_Age_obs$AgeErr,
                                     Ignore=dat$MeanSize_at_Age_obs$Ignore)

MeanSize_at_Age_obs_f <-dat$MeanSize_at_Age_obs[,8:24]
MeanSize_at_Age_obs_m <-dat$MeanSize_at_Age_obs[,25:41]
MeanSize_at_Age_obs_N_f <-dat$MeanSize_at_Age_obs[,42:58]

dat_rows_names <-c(colnames(MeanSize_at_Age_obs_esp),colnames(dat$MeanSize_at_Age_obs[1,8:58]))

MeanSize_at_Age_obs1 <-cbind(MeanSize_at_Age_obs_esp,
                             MeanSize_at_Age_obs_f,
                             MeanSize_at_Age_obs_m,
                             MeanSize_at_Age_obs_N_f)

names(MeanSize_at_Age_obs1) <-dat_rows_names

dat1$MeanSize_at_Age_obs<-MeanSize_at_Age_obs1 #data.frame

# Modificar con los datos propios
dat1$N_environ_variables <-0 #numeric
dat1$N_sizefreq_methods <-0 #numeric
dat1$do_tags <-0 #numeric
dat1$morphcomp_data <-0 #numeric
dat1$use_selectivity_priors <-0 #numeric
dat1$eof <-TRUE #logical

```

5 Escribir archivo dat modificado con la función SS_write

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

5.0.0.1 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
```


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

```
dir(dir_GSA1)[11]
## [1] "STOCWT.DAT"
STOCWT.DAT <- read.table(paste(dir_GSA1,dir(dir_GSA1)[11],sep="/"),
                        sep=" ",na="NA",fill=T,skip = 5)
STOCWT.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

5.0.0.3 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
```

5.0.0.4 Ogiva 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)
PROPMAT.DAT
##      V1      V2      V3 V4 V5 V6
## 1  0 0.2965 0.9855 0.99  1  1
```

5.0.1 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...

5.0.1.1 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 ?

5.0.1.2 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"
```

5.0.1.3 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
```

5.0.1.4 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
```

5.0.1.5 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
```

5.0.1.6 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

```

5.0.1.7 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
```

5.0.1.8 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
```

5.0.1.9 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
```