iSAM

Process Description:

Observations\_at\_length

Author:

*I Doonan, S Ramsussan*

# Process Overview

What is the point of this process? What does it offer that the other processes don't?

This process is based on the proportions\_at\_age (numbers\_at\_age) observation process, so that length frequency observations can be fitted in an age based model. The selected categories’ population-by-age are converted into population-by-lengths using the distribution of lengths-at-age and the mean length by age growth curve. The results, along with the data, are used to calculate the log-likelihood.

# Example Configuration File Syntax

Please put in a list of all parameters you expect to be able to use in the configuration file, including the type of parameter, is it a list or single value, is it optional or have a default value etc. The more information here the better.

Even adding a description is helpful as this is automatically picked up by the documentation generator and put in to the manual.

sub-commands time\_step, categories, likelihood, simulation\_likelihood, & selectivity already defined in @observation

For @observation <label; string>

#### **Common sub-commands**

type process\_proportions\_at\_length | time\_step\_proportions\_at\_length | process\_numbers\_at\_length | time\_step\_numbers\_at\_length

# this mirrors the types for age frequency data

~~years Define the years that have data.  
Type: Vector of integers or integer range  
Default: No default (may be specified in the table obs sub-command)  
Value: Valid model years~~

class\_mins <vector of length boundaries>

Type: Integer vector (length m) #we should be able to specify 3.0 3.5 4.0 4.5 since these are valid lengths?

Why: Define the minimum length for each length class and the optional cap for the last length class

Default: No default

Value: Valid lengths in the range 0 and upper limit found as the mean length at @model.max\_age plus 3 times the standard error. The ith class width is class\_min[i] up to, but not including class\_min[i+1]. If plus\_group = true, then width of the last group is class\_min[m] up to the upper limit above. If minus\_group = true, then width of the first group is 0 up to, but not including class\_min[1]. If plus\_group = false, then width of the last group is class\_min[m - 1] up to, but not including class\_min[m], i.e., we need an extra value to cap the last length class. Number of length classes is m – 1 + ( minus\_group + plus\_group).

plus\_group <bool>  
Type: bool  
Default: true

What: use length plus group

minus\_group <bool>  
Type: bool  
Default: false

Comment: Use length minus group

table <type>

Comment: Define a table of values for the type specified

Type: string, of value “obs” or “error\_values” or “process\_errors”?

Default: No default

Value: The first column is the year. Rest of row contains data in the same order specified in *class\_mins*, one block for each category\_groups (order of blocks is same as in sub-command *categories*. Length of row is 1 + number\_of\_category\_groups \* (m - 1 + ( minus\_group + plus\_group)). If type is “obs”, then data is numbers or proportions at length. If type is “error\_values”, then data is whatever is specified in distribution sub-command (e.g., cv or standard deviation or N). Values should all be non-negative. [Do we need the first row being column names? ]

For “error\_values” or “process\_errors”, do we need another format such as one value for each category group that applies to the year the data is collected in? I have not thought this thru yet.

end\_table

Comment: Defines the end of the table, with no value associated with it.

~~process\_errors #Specify any process error~~ Put into table above?

~~Type: constant vector ???~~

~~Default: true ???~~

tolerance Tolerance to something?

Type: constant

Default: Double(0.001

delta Delta for something?

Type: constant

Default: DELTA

#### **sub-commands for types like process\_xxx**

process <Process label>

Comment: process that data is collected from

Type: string

Default: No default

process\_proportion <type>

Comment: Process proportion when data applies to

Type: constant

Default: Double(0.5)

#### **sub-commands for types like xxx\_proportions\_xxx**

sum\_to\_one <bool>

Comment: Specify that each years’ data sums to one

Type: bool

Default: false

Value: If true, divide each year’s vector by its sum. If false, throw an error if the sum is not one.

#### **sub-commands for types like xxx\_numbers\_xxx**

catchability <catchability label>

Comment: Link data to a catchability so we can convert data into absolute

Type: string

Default: No default

Value: type must be valid label to a catchability parameter

# Step-by-step

A detailed list of steps the process goes **through in the following order (steps 1-3)** to achieve its purpose.

Let *years* hold the years when observations data were made

If model year is not in *years*, do nothing.

If model year is in *years* then:

If needed (i.e., *plus\_group*), calculate the max length to consider (this may be done once in the set up by looking at the distribution of length\_at\_age over all growth curves). We should have a vector of class boundaries that is m2 + 1 long if there are m2 length classes (possibly a different number from the number of *min\_lengths*).

Get the population by age that applies to this observation taking into account selectivities and catchability, if needed, and category groups. The categories will have to be kept separate until joining then at the end since each component category may have different growth curves and distributions.

Conceptually, we build an age-length conversion matrix. Each column is assigned an age value in the same order as in the population-by-age vector. The columns sum to one so that all of fish at a certain age are accounted for in the resultant vector of population-by-length\_class. All values in the matrix must be >= 0 and <=1. The rows are labelled by the corresponding length class.

For each component category (it may be worth combining categories that have the same growth curve), build an age-length conversion matrix. For each age, populate its column by calculating the proportion of the length-at-age distribution that is in each length class.

In what follows, the value *x* to be used in the normal cumulative probability evaluation is given by (*li* – mean\_length)/ standard\_deviation. Standard\_deviation is given by cv (in decimal form) \* mean. If the distribution is lognormal, then *x* has to be in log space. This gives the probability than lengths are less than or equal to *li* (P[*li*]). The fraction of fish in the length class is then P[*li+1*] - P[*li*].

Approximation for the cumulative normal curve, cnf(x) or Φ(*x*), with the absolute error in Φ(*x*) is |*ε*(*x*)| < 7.5·10−8 (algorithm [26.2.17](http://www.math.sfu.ca/%7Ecbm/aands/page_932.htm)):

For *x > 0,* cnf (x) is given by:


    \Phi(x) = 1 - \phi(x)\left(b_1t + b_2t^2 + b_3t^3 + b_4t^4 + b_5t^5\right) + \varepsilon(x), \qquad t = \frac{1}{1+b_0x},
  

where *ϕ*(*x*) is the standard normal PDF (distribution), and *b*0 = 0.2316419, *b*1 = 0.319381530, *b*2 = −0.356563782, *b*3 = 1.781477937, *b*4 = −1.821255978, *b*5 = 1.330274429.

For x < 0, use x2 = abs(x) in the approx. above and return 1-ncf(x2).

Standard normal PDF is 1.0/(power (2\*pi, 0.5)) \*exp(-0.5\*power(x,2)), since x has mean 0 and standard deviation 1.

CASAL uses:

// Abramowitz & Stegun eqn 26.2.18

  // Equations: z = fabs((x-mu)/sigma);

  //            p = 1-0.5\*pow((1+0.196854\*z+0.115194\*z\*z+0.000344\*z\*z\*z+0.019527\*z\*z\*z\*z),-4);

  //            if (x<mu) p=1-p;

but this is not recommended by Copyright © Prentice Hall Inc. 1999. Author: Nick Bagley.

Reference: Equ 26.2.17 *In* Zelen, Marvin; Severo, Norman C. (1964). [*Probability Functions (chapter 26)*](http://www.math.sfu.ca/%7Ecbm/aands/page_931.htm). [*Handbook of mathematical functions with formulas, graphs, and mathematical tables*](http://en.wikipedia.org/wiki/Abramowitz_and_Stegun), by [Abramowitz, M.](http://en.wikipedia.org/wiki/Milton_Abramowitz); and [Stegun, I. A.](http://en.wikipedia.org/wiki/Irene_A._Stegun): National Bureau of Standards. New York, NY: Dover. [ISBN](http://en.wikipedia.org/wiki/International_Standard_Book_Number) [0-486-61272-4](http://en.wikipedia.org/wiki/Special:BookSources/0-486-61272-4).

#### **Expected population-by-length**

The population for a length class is given by multiplying its row in the age-length conversion matrix by the population-by-age vector and summing. If the observations are proportions, then vector is divided by its sum. The log-likelihood is then calculated.

**Scott's Implementation**

Before first execution (Build):

During actual execution (Execute):