

Package: AgeingError (via r-universe)

October 8, 2026

Type Package

Title Estimating ageing error with 'TMB' from double reads

Version 2.2.0

Date 2026-06-16

Description Implements Generalized Linear Mixed Effect Models (GLMMs) using TMB to estimate ageing error from double reads of otoliths. The original analysis (Punt et al. 2008; <doi:10.1139/F08-111>) was written in ADMB and focused on estimating imprecision and bias. The newer version of the software allows for random effects using TMB.

License GPL-3

URL <https://pfmc-assessments.github.io/AgeingError/>

BugReports <https://github.com/pfmc-assessments/AgeingError/issues>

Depends R (>= 3.5.0)

Imports cli, DiagrammeR, dplyr, fs, lifecycle, readr, stats, TMB (>= 1.8.0)

Suggests knitr, litedown, remotes, spelling, testthat (>= 3.0.0)

LinkingTo RcppEigen, TMB

VignetteBuilder litedown

ByteCompile true

Config/roxygen2/version 8.1.0

Config/testthat/edition 3

Config/testthat/parallel true

Copyright inst/COPYRIGHTS

Encoding UTF-8

Language en-US

LazyData true

Roxygen list(markdown = TRUE)

SystemRequirements GNU make, C++17

Config/pak/sysreqs cmake libglpk-dev make libicu-dev libuv1-dev libxml2-dev libx11-dev
Repository https://noaa-fisheries-integrated-toolbox.r-universe.dev
Date/Publication 2026-10-08 15:39:35 UTC
RemoteUrl https://github.com/pfmc-assessments/AgeingError
RemoteRef HEAD
RemoteSha 4589e7354988d9b5133c56c9188cc303e8195ace

Contents

ageing_comparison	2
ageing_error_matrix_to_fims	4
cMx	4
CreateData	5
CreateSpecs	5
determine_n_sets	6
DoApplyAgeError	6
estgrowth.vb	7
fit	8
load_data	9
load_specs	9
minimizer	10
plot_output	11
PlotOutputFn	12
ProcessResults	12
rMx	13
run	14
RunFn	15
SimulatorFn	16
StepwiseFn	18
tally_repeats	21
write_data_file	22
write_files	23
write_specs_file	26
Index	29

ageing_comparison	<i>Plot comparison of double age readings</i>
-------------------	---

Description

Plot with circles proportional to how many double readings fell in each pair of coordinates

Usage

```
ageing_comparison(
  xvec,
  yvec,
  scale.pts = 2,
  col.pts = grDevices::grey(0.1, alpha = 0.5),
  col.hist = grDevices::rgb(0, 0, 0.5, alpha = 0.7),
  counts = TRUE,
  maxage = NULL,
  hist = TRUE,
  hist.frac = 0.1,
  xlab = "Age reader A",
  ylab = "Age reader B",
  title = NULL,
  png = FALSE,
  filename = "ageing_comparison.png",
  SaveFile = NULL,
  verbose = TRUE
)
```

Arguments

xvec	vector of values from reader A
yvec	vector of values from reader B
scale.pts	Documentation needed.
col.pts	color for points
col.hist	color for histograms
counts	include text within each bubble showing count of values?
maxage	maximum age to include in the plot (doesn't yet work well)
hist	include a histogram along each axis?
hist.frac	maximum value of histograms as fraction of maxage
xlab	label for xvec
ylab	label for yvec
title	Optional title to add at top of plot
png	Save plot to PNG file?
filename	File name for PNG file.
SaveFile	directory where plot will be saved. NULL value will make it go to working directory.
verbose	Report messages as function runs.

Author(s)

Ian G. Taylor

 ageing_error_matrix_to_fims

Format an ageing-error probability matrix for FIMS

Description

Convert a matrix with true ages in rows and observed ages in columns to the long format expected for ageing_error input by FIMS.

Usage

```
ageing_error_matrix_to_fims(
  probabilities,
  ages,
  fleet = NA_character_,
  timing = NA_real_
)
```

Arguments

probabilities	Numeric square matrix of probabilities. Rows represent true ages and columns represent observed ages.
ages	Age values corresponding to the rows and columns of probabilities.
fleet	Fleet name, or NA to apply to every fleet with age-composition data.
timing	Model year, or NA to use as the default for every year.

Value

A data frame with columns required for FIMS ageing_error input.

 cMx

Make a column matrix

Description

The function is currently defined as:

```
function (Input)
as.matrix(Input)
```

Usage

```
cMx(Input)
```

Arguments

Input input to be converted to a matrix

Author(s)

James T. Thorson

CreateData	<i>Deprecated function renamed to load_data()</i>
------------	---

Description

[Deprecated] CreateData() has been renamed as [load_data\(\)](#) to better clarify its purpose

Usage

```
CreateData(...)
```

Arguments

... Any arguments associated with the deprecated function

Author(s)

Ian G. Taylor

See Also

[load_data\(\)](#)

CreateSpecs	<i>Deprecated function renamed to load_specs()</i>
-------------	--

Description

[Deprecated] CreateSpecs() has been renamed as [load_specs\(\)](#) to better clarify its purpose

Usage

```
CreateSpecs(...)
```

Arguments

... Any arguments associated with the deprecated function

Author(s)

Ian G. Taylor

See Also

[load_specs\(\)](#)

determine_n_sets	<i>Determine the number of data sets in a data file</i>
------------------	---

Description

Determine the number of data sets in a data file

Usage

```
determine_n_sets(file)
```

Arguments

file	A file path to a data file.
------	-----------------------------

Value

An integer giving the number of data sets in the file.

Author(s)

Kelli F. Johnson

DoApplyAgeError	<i>Run the ageing error optimization routine</i>
-----------------	--

Description

Run the ageing error optimization routine

Usage

```
DoApplyAgeError(
  Species = "AgeingError",
  DataSpecs,
  ModelSpecsInp,
  AprobWght = 1e-06,
  SlopeWght = 0.01,
  SaveDir = getwd(),
  verbose = FALSE
)
```

Arguments

Species	A string that will be used to create file names. Typically, users will use the common name for the species of interest, especially if you are saving files from multiple species in a single directory. Though, the default is "AgeingError".
DataSpecs	A data object returned from <code>load_data()</code> .
ModelSpecsInp	A specification object returned from <code>load_specs()</code> .
AprobWght, SlopeWght	Numeric values passed to the model. The defaults are 1e-06 and 0.01. Andre originally had these hard coded from his workspace. TODO: decide if they should be passed in the specifications or data files.
SaveDir	A path, relative or absolute, to a directory where the results will be saved. The directory need not exist currently as it will be created dynamically.
verbose	A logical specifying if messages should be printed. The default is to NOT print, i.e., verbose = FALSE.

Author(s)

Andre E. Punt

estgrowth.vb

Calculate von Bertalanffy Growth Parameters from Lengths and Ages

Description

Calculate von Bertalanffy growth parameters from length and age data or predicted lengths given ages and input parameters.

Usage

```
estgrowth.vb(Par, Ages, Lengths, ReturnType = c("NLL", "Pred"), sdFactor = 1)
```

Arguments

Par	A list of von Bertalanffy growth parameters in log space ordered as follows: K, Linf, L0, CV0, and CV1. Names will be assigned if they are not provided.
Ages	A vector of ages in years. Values of NA are accepted.
Lengths	A vector of Lengths in cm. Lengths can be NULL if ReturnType == "Pred" because you are only predicting using ages, where the lengths are just needed for estimation purposes. If not NULL, ensure that there is one length measurement for every age measurement. Values of NA are accepted.
ReturnType	A single character value with "NLL" being the default, which leads to the negative log-likelihood value being returned. If "Pred", then three values are returned for each combination of length and age, low, prediction, and high based on the input parameters and standard deviation factor, i.e., sdFactor.
sdFactor	The number of standard deviations to include in the low and high calculations. The default is 1.0.

Value

Depending on `ReturnType`, either the negative log likelihood is returned based on fits to the data or a matrix of three columns with low, predicted, and high values for each combination of length and age. Distance of the low and high from the predicted value depends on the `sdFactor`, allowing confidence intervals based on normal theory or other theories to be created.

Examples

```
## Not run:
bio_dat <- data.frame(
  Age = rep(0:30, each = 20),
  Length_cm = rnorm(n = 31 * 20, mean = 50, sd = 5)
)
pars_in <- lapply(FUN = log, X = list(
  "K" = 0.13,
  "Linf" = 55,
  "L0" = 5,
  "CV0" = 0.1,
  "CV1" = 0.1
))
solve <- optim(
  fn = estgrowth.vb, par = unlist(pars_in), hessian = FALSE,
  Ages = bio_dat[, "Age"],
  Lengths = bio_dat[, "Length_cm"]
)
predictions <- estgrowth.vb(
  Par = solve$par, ReturnType = "Pred",
  sdFactor = 1,
  Ages = bio_dat[, "Age"],
  Lengths = bio_dat[, "Length_cm"]
)
plot(bio_dat$Age, predictions[, "Lhat_pred"],
  xlab = "Age (years)", ylab = "Predicted length (cm)"
)
exp(solve$par)

## End(Not run)
```

fit

Fit an ageing error model using TMB

Description

Fit an ageing error model using TMB

Usage

```
fit()
```

Author(s)

Kelli F. Johnson

load_data*Load the formatted ageing error data*

Description

Load the formatted ageing error data

Usage

```
load_data(DataFile = "data.dat", NDataSet = 1, verbose = FALSE, EchoFile = "")
```

Arguments

DataFile	Filename for input data
NDataSet	Number of data sets within DataFile
verbose	Return messages to the console (in addition to any output to EchoFile)
EchoFile	A file path to a file that will be created or appended to if it already exists to store information about your data inputs. The default is '', which leads to output being printed to the screen rather than saved in a file. An example of a user-defined input would be 'EchoTMB.out'.

Author(s)

Andre E. Punt

load_specs*Read the ageing error specifications*

Description

Read the ageing error specifications

Usage

```
load_specs(SpecsFile = "data.spc", DataSpecs, verbose = FALSE)
```

Arguments

SpecsFile	Filename for input specifications.
DataSpecs	The output from load_data()
verbose	Return messages to the console (TRUE/FALSE)

Author(s)

Andre E. Punt

minimizer*Minimize the negative log likelihood*

Description

Minimize the negative log likelihood using "nlmimb" and/or "optim".

Usage

```
minimizer(  
  model,  
  method = c("optim", "nlmimb", "both"),  
  lower,  
  upper,  
  verbose = FALSE  
)
```

Arguments

model	A model to be optimized.
method	A string specifying the desired method to be used for the optimization routine. The options are listed in the function call, where the default is to use "optim". Using both routines is an option, via "both", and will lead to first optimizing the model using "nlminb" and then re-optimization of the model with "optim". Note that when using <code>stats::optim()</code> , the "L-BFGS-B" method is used rather than the default method of "Nelder-Mead".
lower, upper	Vectors of parameter bounds of the same length as the number of parameters in the model.
verbose	A logical specifying if messages should be printed. The default is to NOT print, i.e., verbose = FALSE.

Author(s)

Andre E. Punt

plot_output	<i>Plot output</i>
-------------	--------------------

Description

Plots age comparisons and results from the fitted Ageing Error model

Usage

```
plot_output(
  Data,
  IDataset,
  MaxAge,
  Report,
  subplot = 1:3,
  Nparameters = 0,
  LogLike = 0,
  ReaderNames = NULL,
  Species = "AgeingError",
  SaveDir = getwd(),
  verbose = FALSE,
  ...
)
```

Arguments

Data	Input data matrix
IDataset	Index of the data set used in creating the filename
MaxAge	Maximum estimated age
Report	Results from fitting the model
subplot	Vector of which plots to create.
Nparameters	Number of parameters
LogLike	Negative log likelihood from fitting the model
ReaderNames	Vector with names of each reader, defaults to 'Reader1', 'Reader2', etc. if left at the default argument of NULL. If you pass a vector of strings, the vector must be the same length as <code>NCOL(Data) - 1</code> .
Species	String used at beginning of the output files
SaveDir	Directory for fitted model
verbose	Report messages as function runs.
...	Additional arguments passed to <code>ageing_comparison()</code> .

Value

A list containing AIC, AICc, and BIC; the SS3-format ErrorAndBiasArray; and ageing_error_fims, a tibble of FIMS-format probability matrices with one column per reader (reader1, reader2, etc.).

Author(s)

James T. Thorson, Ian G. Taylor

References

Punt, A.E., Smith, D.C., KrusicGolub, K., and Robertson, S. 2008. Quantifying age-reading error for use in fisheries stock assessments, with application to species in Australias southern and eastern scalefish and shark fishery. Can. J. Fish. Aquat. Sci. 65: 1991-2005.

PlotOutputFn	<i>Deprecated function replaced by <code>plot_output()</code></i>
--------------	---

Description

[Deprecated] PlotOutputFn() has been replaced by `plot_output()`

Usage

```
PlotOutputFn(...)
```

Arguments

... Any arguments associated with the deprecated function

Author(s)

James T. Thorson, Ian G. Taylor

See Also

`plot_output()`

ProcessResults	<i>Process results of the ageing error estimation</i>
----------------	---

Description

Process results of the ageing error estimation

Usage

```
ProcessResults(
  Species = "AgeingError",
  SaveDir = getwd(),
  CalcEff = FALSE,
  verbose = FALSE
)
```

Arguments

Species	A string that will be used to create file names. Typically, users will use the common name for the species of interest, especially if you are saving files from multiple species in a single directory. Though, the default is "AgeingError".
SaveDir	A path, relative or absolute, to a directory where the results will be saved. The directory need not exist currently as it will be created dynamically.
CalcEff	Calculate effective sample sizes (TRUE/FALSE)
verbose	A logical specifying if messages should be printed. The default is to NOT print, i.e., verbose = FALSE.

Author(s)

Andre E. Punt

rMx

Make a row matrix

Description

The function is currently defined as:

```
function (Input)
if (is.vector(Input)) Output <- t(as.matrix(Input))
if (!is.vector(Input)) Output <- as.matrix(Input)
Output
```

Usage

rMx(Input)

Arguments

Input	input to be converted into a row matrix
-------	---

Author(s)

James T. Thorson

run	<i>Run ageing error routine</i>
-----	---------------------------------

Description

A wrapper for running a TMB model to estimate ageing error for a given data set and specification file.

Usage

```
run(directory, file_data = "data.dat", file_specs = "data.spc")
```

Arguments

directory	A string specifying a file path to a directory where you would like to save the results.
file_data	A string specifying the data file within 'directory'.
file_specs	A string specifying the specifications file within 'directory'.

Value

Invisibly return model output.

Author(s)

Kelli F. Johnson

See Also

[write_files\(\)](#)

Examples

```
## Not run:
data_test <- data.frame(
  reader1 = c(7, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, 5, 8, 5),
  reader2 = c(8, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, NA, NA, NA),
  reader3 = c(7, 10, 7, 6, 6, 8, 7, 9, 8, 10, 10, 5, 6, 7, NA, NA, NA, NA, 5, 8, 5)
)
write_files(dat = data_test, dir = tempdir())
out <- run(dir = tempdir())
# see estimated parameters
out$model$par
# see model selection results
out$output$ModelSelection

# see ageing error in SS3 format
out$output$errorAndBiasArray
```

```

# add to an SS3 model (assumes the model already has a single
# ageing error matrix and a maxage <= maxage in the ageing error model)
ss3_inputs <- r4ss::SS_read()
maxage <- ss3_inputs$dat$Nages
ss3_inputs$dat$ageerror <-
  out$output$errorAndBiasArray[c("Expected_age", "SD"), 1 + 0:maxage, "Reader 1"] |>
  as.data.frame()
r4ss::SS_write(inputlist = ss3_inputs)

# see ageing error in FIMS format (one separate matrix per reader)
out$output$ageing_error_fims[["reader1"]]
# add to a FIMS model (depends on matching the range of ages)
data_4_fims <- rbind(data_4_fims, out$output$ageing_error_fims[["reader1"]])

## End(Not run)

```

RunFn

*Deprecated function replaced by run()***Description**

[Deprecated] RunFn() has been replaced by [run\(\)](#)

Usage

```
RunFn(...)
```

Arguments

... Any arguments associated with the deprecated function

Author(s)

James T. Thorson, Ian J. Stewart, Andre E. Punt, Ian G. Taylor

See Also

[run\(\)](#)

 SimulatorFn

Simulate double reading data

Description

A function to generate simulated double reading data with given properties

Usage

```

SimulatorFn(
  Nreaders,
  M,
  SelexForm,
  ErrorParams,
  BiasParams,
  SelexParams,
  ReadsMat,
  RecCv = 0.6,
  RecAr1 = 0.8,
  Amax = 100
)

```

Arguments

Nreaders	The number of ageing readers
M	True natural mortality
SelexForm	Form of selectivity-at-age (logistic selex-at-age is the only one that is implemented).
ErrorParams	Error type CV in the following equation: $\text{Var}[\text{AgeRead}] = (\text{CV} \times \text{TrueAge})^2$
BiasParams	Bias type b in the following equation: $E[\text{AgeRead}] = b \times \text{TrueAge}$
SelexParams	Selectivity parameters, which are standard to the logistic equation.
ReadsMat	Matrix describing number of reads per reader combination. Where each row specifies how many reads (in the first column) have a particular pattern of double reads (in the second through Nreaders+1 columns).
RecCv	CV of recruitment, and it should be noted that recruitment is assumed to be stationary over time.
RecAr1	First-order autoregressive coefficient for recruitment
Amax	True maximum age

Value

Returns a simulated double read matrix

Author(s)

James T. Thorson

References

Punt, A.E., Smith, D.C., KrusicGolub, K., and Robertson, S. 2008. Quantifying age-reading error for use in fisheries stock assessments, with application to species in Australia's southern and eastern scalefish and shark fishery. *Can. J. Fish. Aquat. Sci.* 65: 1991-2005.

Examples

```
# Parameters for generating data
# This represents 2 unique readers
# Row 1 -- Otoliths read only once by reader
# Row 2 -- Otoliths read twice by reader 1
# Row 3 -- Otoliths read only once by reader 2
# Row 4 -- Otoliths read twice by reader 2
# Row 5 -- Otoliths read once by reader 1 and once by reader 2
ReadsMat <- structure(matrix(
  nrow = 5, ncol = 5,
  c(
    rep(25, 5),
    1, 1, 0, 0, 1,
    0, 1, 0, 0, 0,
    0, 0, 1, 1, 1,
    0, 0, 0, 1, 0
  )
), dimnames = list(
  c(
    "Reader1_Only", "Reader1_DoubleReads",
    "Reader2_Only", "Reader2_DoubleReads",
    "Reader1_&_Reader2"
  ),
  c(
    "NumberOfReads",
    "Reader1", "Reader1_DoubleReads",
    "Reader2", "Reader2_DoubleReads"
  )
))

# Generate data
set.seed(2)
AgeReads <- SimulatorFn(
  Nreaders = 4, M = 0.2,
  SelexForm = "Logistic",
  SelexParams = c(5, 0.2), BiasParams = c(1, 1, 1.1, 1.1),
  ErrorParams = c(0.2, 0.2, 0.2, 0.2), ReadsMat = ReadsMat,
  RecCv = 0.6, RecAr1 = 0.8, Amax = 100
)
```

StepwiseFn

*Step-wise model selection***Description**

Run step-wise model selection to facilitate the exploration of several modelling configurations using Akaike information criterion (AIC).

Usage

```
StepwiseFn(
  SearchMat,
  Data,
  NDataSets,
  KnotAges,
  MinAge,
  MaxAge,
  RefAge,
  MaxSd,
  MaxExpectedAge,
  SaveFile,
  EffSampleSize = 0,
  Intern = TRUE,
  InformationCriterion = c("AIC", "AICc", "BIC"),
  SelectAges = TRUE
)
```

Arguments

- | | |
|----------------------|---|
| SearchMat | <p>A matrix explaining stepwise model selection options. One row for each readers error and one row for each readers bias + 2 rows, one for MinusAge, i.e., the age where the proportion at age begins to decrease exponentially with decreasing age, and one for PlusAge, i.e., the age where the proportion-at-age begins to decrease exponentially with increasing age.</p> <p>Each element of a given row is a possible value to search across for that reader. So, the number of columns of SearchMat will be the maximum number of options that you want to include. Think of it as several vectors stacked row-wise where shorter rows are filled in with NA values. If reader two only has two options that the analyst wants to search over the remainder of the columns should be filled with NA values for that row.</p> |
| InformationCriterion | <p>A string specifying the type of information criterion that should be used to choose the best model. The default is to use AIC, though AIC corrected for small sample sizes and BIC are also available.</p> |
| SelectAges | <p>A logical input specifying if the boundaries should be based on MinusAge and PlusAge. The default is TRUE.</p> |

Details

AIC seems like an appropriate method to select among possible values for PlusAge, i.e., the last row of SearchMat, because PlusAge determines the number of estimated fixed-effect hyperparameters that are used to define the true proportion-at-age hyperdistribution. This hyperdistribution is in turn used as a prior when integrating across a true age associated with each otolith. This true age, which is a latent effect, can be interpreted as a random effect with one for each observation. So, the use of AIC to select among parameterizations of the fixed effects defining this hyperdistribution is customary (Pinheiro and Bates, 2009). This was tested for sablefish, where AIC lead to a true proportion at age that was biologically plausible.

Author(s)

James T. Thorson

References

Punt, A.E., Smith, D.C., KrusicGolub, K., and Robertson, S. 2008. Quantifying age-reading error for use in fisheries stock assessments, with application to species in Australia's southern and eastern scalefish and shark fishery. *Can. J. Fish. Aquat. Sci.* 65: 1991-2005.

Pinheiro, J.C., and Bates, D. 2009. *Mixed-Effects Models in S and S-PLUS*. Springer, Germany.

See Also

- RunFn() will run a single model, where this function runs multiple models.
- PlotOutputFn() will help summarize the output from RunFn().

Examples

```
## Not run:
example(RunFn)
##### Run the model (MAY TAKE 5-10 MINUTES)
fileloc <- file.path(tempdir(), "age")
dir.create(fileloc, showWarnings = FALSE)
RunFn(
  Data = AgeReads2, SigOpt = SigOpt, KnotAges = KnotAges,
  BiasOpt = BiasOpt,
  NDataSets = 1, MinAge = MinAge, MaxAge = MaxAge, RefAge = 10,
  MinusAge = 1, PlusAge = 30, SaveFile = fileloc,
  AdmbFile = file.path(system.file("executables",
    package = "nwfsAgeingError"
  ), .Platform$file.sep),
  EffSampleSize = 0, Intern = FALSE, JustWrite = FALSE, CallType = "shell"
)
# Plot output
PlotOutputFn(
  Data = AgeReads2, MaxAge = MaxAge,
  SaveFile = fileloc, PlotType = "PDF"
)

##### Stepwise selection
```

```

# Parameters
MaxAge <- ceiling(max(AgeReads2) / 10) * 10
MinAge <- 1

##### Stepwise selection
StartMinusAge <- 1
StartPlusAge <- 30

# Define matrix explaining stepwise model selection options
# One row for each reader + 2 rows for
# PlusAge (age where the proportion-at-age begins to
# decrease exponentially with increasing age) and
# MinusAge (the age where the proportion-at-age begins to
# decrease exponentially with decreasing age)
# Each element of a given row is a possible value to search
# across for that reader
SearchMat <- array(NA,
  dim = c(Nreaders * 2 + 2, 7),
  dimnames = list(
    c(
      paste("Error_Reader", 1:Nreaders),
      paste("Bias_Reader", 1:Nreaders), "MinusAge", "PlusAge"
    ),
    paste("Option", 1:7)
  )
)
# Readers 1 and 3 search across options 1-3 for ERROR
SearchMat[c(1, 3), 1:3] <- rep(1, 2) %o% c(1, 2, 3)
# Reader 2 mirrors reader 1
SearchMat[2, 1] <- -1
# Reader 4 mirrors reader 3
SearchMat[4, 1] <- -3
# Reader 1 has no BIAS
SearchMat[5, 1] <- 0
# Reader 2 mirrors reader 1
SearchMat[6, 1] <- -1
# Reader 3 search across options 0-2 for BIAS
SearchMat[7, 1:3] <- c(1, 2, 0)
# Reader 4 mirrors reader 3
SearchMat[8, 1] <- -3
# MinusAge searches with a search kernel of -10,-4,-1,+0,+1,+4,+10
SearchMat[9, 1:7] <- c(
  StartMinusAge,
  StartMinusAge - 10,
  StartMinusAge - 4,
  StartMinusAge - 1,
  StartMinusAge + 1,
  StartMinusAge + 4,
  StartMinusAge + 10
)
SearchMat[9, 1:7] <- ifelse(SearchMat[9, 1:7] < MinAge,
  NA, SearchMat[9, 1:7])

```

```

)
# PlusAge searches with a search kernal of -10,-4,-1,+0,+1,+4,+10
SearchMat[10, 1:7] <- c(
  StartPlusAge,
  StartPlusAge - 10,
  StartPlusAge - 4,
  StartPlusAge - 1,
  StartPlusAge + 1,
  StartPlusAge + 4,
  StartPlusAge + 10
)
SearchMat[10, 1:7] <- ifelse(SearchMat[10, 1:7] > MaxAge,
  NA, SearchMat[10, 1:7]
)

# Run model selection
# This outputs a series of files
# 1. "Stepwise - Model loop X.txt" --
#   Shows the AIC/BIC/AICc value for all different combinations
#   of parameters arising from changing one parameter at a time
#   according to SearchMat during loop X
# 2. "Stepwise - Record.txt" --
#   The Xth row of IcRecord shows the record of the
#   Information Criterion for all trials in loop X,
#   while the Xth row of StateRecord shows the current selected values
#   for all parameters at the end of loop X
# 3. Standard plots for each loop
# WARNING: One run of this stepwise model building example can take
# 8+ hours, and should be run overnight
StepwiseFn(
  SearchMat = SearchMat, Data = AgeReads2,
  NDataSets = 1, MinAge = MinAge, MaxAge = MaxAge,
  RefAge = 10, MaxSd = 40, MaxExpectedAge = MaxAge + 10,
  SaveFile = fileloc, InformationCriterion = c("AIC", "AICc", "BIC")[3]
)

## End(Not run)

```

tally_repeats

Tally repeated age reading combinations in a new count column

Description

AgeingError requires a table of data with columns for each reader and rows for each age reading combination, where a count column indicates how many times the age reading combination is repeated. This function tallies the repeated values and adds an initial count column with the tally.

Usage

```
tally_repeats(dat)
```

Arguments

dat	Input dataframe or tibble with columns for each reader and rows for each specimen
-----	---

Author(s)

Ian G. Taylor

write_data_file	<i>Write a data file for the AgeingError package</i>
-----------------	--

Description

The resulting file has the format required by the `load_data()` function. In the future, this step could be bypassed by creating the list output from `load_data()` directly. Only one data set is supported (`NDataSet = 1`). This function is based on a subset of the `RunFn()` function in the older `ADMB` version of this package.

Usage

```
write_data_file(
  dat,
  dir = getwd(),
  file_name = "data.dat",
  minage = 0,
  maxage = NULL,
  refage = NULL,
  minusage = NULL,
  plusage = NULL
)
```

Arguments

dat	Dataframe or tibble with columns for each reader and rows for each age reading combination. This could either have count as the first column or not. If not, <code>tally_repeats()</code> will be called to add the count column. Order your reader/lab columns such that similar readers/labs are located next to one another because columns to the right can mirror columns to their immediate left in terms of parameter estimates.
dir	Directory where the data file will be saved.
file_name	Name of the data file.
minage	An integer, specifying the minimum possible "true" age.
maxage	An integer, specifying the maximum possible "true" age. If <code>NULL</code> , this will be set to the multiple of 5 which is >120% of the observed maximum age.

refage An arbitrarily chosen age from which "true" age-composition fixed-effects are calculated as an offset. This has no effect on the answer but could potentially effect estimation speed. By default this will be set to the maxage / 4.

minusage The minimum age for which an age-specific age-composition is estimated. Ages below minusage have "true" proportion-at-age (P_a) estimated as

$$P_a = P_{minusage} * \exp^{\beta * (minusage - a)}$$

, where beta is an estimated log-linear trend in the "true" proportion-at-age. If minusage = minage, beta is not estimated.

plusage Identical to minusage except defining the age above with age-specific age composition is not estimated.

Value

Invisibly returns the path to the data file (file.path(dir, file_name)).

Author(s)

Ian G. Taylor, James T. Thorson, Ian J. Stewart, Andre E. Punt

See Also

[write_files\(\)](#), [write_specs_file\(\)](#), [load_data\(\)](#), [tally_repeats\(\)](#)

Examples

```
data_test <- data.frame(
  reader1 = c(7, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, 5, 8, 5),
  reader2 = c(8, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, NA, NA, NA),
  reader3 = c(7, 10, 7, 6, 6, 8, 7, 9, 8, 10, 10, 5, 6, 7, NA, NA, NA, 5, 8, 5)
)
data_file <- write_data_file(data_test, dir = tempdir(), file_name = "test.dat")
```

write_files	<i>Write the input files (data and specifications) for the AgeingError package</i>
-------------	--

Description

Write the input files (data and specifications) for the AgeingError package

Usage

```
write_files(
  dat,
  dir = getwd(),
  file_dat = "data.dat",
  file_specs = "data.spc",
  minage = 0,
  maxage = NULL,
  refage = NULL,
  minusage = NULL,
  plusage = NULL,
  biasopt = NULL,
  sigopt = NULL,
  knotages = NULL
)
```

Arguments

dat	Dataframe or tibble with columns for each reader and rows for each age reading combination. This could either have count as the first column or not. If not, <code>tally_repeats()</code> will be called to add the count column. Order your reader/lab columns such that similar readers/labs are located next to one another because columns to the right can mirror columns to their immediate left in terms of parameter estimates.
dir	Directory where the data file will be saved.
minage	An integer, specifying the minimum possible "true" age.
maxage	An integer, specifying the maximum possible "true" age. If NULL, this will be set to the multiple of 5 which is >120% of the observed maximum age.
refage	An arbitrarily chosen age from which "true" age-composition fixed-effects are calculated as an offset. This has no effect on the answer but could potentially effect estimation speed. By default this will be set to the maxage / 4.
minusage	The minimum age for which an age-specific age-composition is estimated. Ages below minusage have "true" proportion-at-age (P_a) estimated as

$$P_a = P_{minusage} * \exp(\beta * (minusage - a))$$

, where beta is an estimated log-linear trend in the "true" proportion-at-age. If minusage = minage, beta is not estimated.

plusage	Identical to minusage except defining the age above with age-specific age composition is not estimated.
biasopt	A vector with one entry for each reader specifying the type of bias specific to each reader. Positive values lead to estimated parameters and negative values are used for shared parameters between readers. Parameter sharing (mirroring) is common when there is more than one reader in a lab working together to refine their methods such that they have matching techniques. If NULL is passed, the default is <code>rep(0, nreaders)</code> which specifies that all readers are unbiased. Possible entries include the following:

-[0-9]+ Mirror the bias of another reader, where the negative integer corresponds to the column of the reader that is being mirrored minus one, e.g., -1 causes it to mirror reader 1. Only lower-numbered readers can be mirrored.

0 Unbiased, where at least one reader has to be unbiased.

1 Constant coefficient of variation, i.e., a 1-parameter linear relationship of bias with true age.

2 Curvilinear, i.e., a 2-parameter Hollings-form relationship of bias with true age.

An example entry for the situation where you have seven readers and you assume that the first reader is unbiased, readers 2-7 have a curvilinear bias, reader 3 shares parameters with reader 2, reader 5 shares parameters with reader 4, and reader 7 shares parameters with reader 6 would look like `c(0, 2, -2, 2, -4, 2, -6)`.

sigopt

A vector with one entry for each reader. Each entry specifies the functional form of reading error as a function of true age. Positive values lead to estimated parameters and negative values are used for shared parameters between readers. If NULL is passed, the default is `c(1, rep(-1, nreaders - 1))` which specifies a constant CV in ageing error which is shared among all readers.

Possible entries include the following:

-[0-9]+ Mirror the standard deviation of another reader, where the negative integer corresponds to the column of the reader that is being mirrored minus one, e.g., -1 causes it to mirror reader 1, for which data is stored in the second column of Data. This number must be lower than -1 times the current position in the vector.

0 No error. But, there could be potential bias.

1 Constant coefficient of variation, i.e., a 1-parameter linear relationship of the standard deviation with the true age.

2 Curvilinear standard deviation, i.e., a 3-parameter Hollings-form relationship of standard deviation with true age.

3 Curvilinear coefficient of variation, i.e., a 3-parameter Hollings-form relationship of coefficient of variation with true age.

5 Spline with estimated slope at beginning and end where the number of parameters is 2 + number of knots. Supported when knotages is provided.

6 Linear interpolation with a first knot of 1 and a last knot of the maximum age, i.e., MaxAge. Supported when knotages is provided.

7 A linear change in the standard deviation of random age-reading error, σ_a , with age. This option has two parameters that need to be specified for each pair of independent readers in the specifications file.

8 A linear change in the coefficient of variation of random age-reading error, CV_a , with age. This option has two parameters that need to be specified for each pair of independent readers in the specifications file.

knotages

A list of knot ages for each reader. This is required when sigopt = 5 or sigopt = 6 and must have one element per reader.

Value

Invisibly returns the path to the data file (`file.path(dir, file_name)`).

Author(s)

Ian G. Taylor, James T. Thorson, Ian J. Stewart, Andre E. Punt

See Also

[load_data\(\)](#), [tally_repeats\(\)](#), [write_specs_file\(\)](#)

Examples

```
data_test <- data.frame(
  reader1 = c(7, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, 5, 8, 5),
  reader2 = c(8, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, NA, NA, NA),
  reader3 = c(7, 10, 7, 6, 6, 8, 7, 9, 8, 10, 10, 5, 6, 7, NA, NA, NA, 5, 8, 5)
)
write_files(dat = data_test, dir = tempdir())
run(dir = tempdir())
```

write_specs_file

Write a specifications file for the AgeingError package

Description

The resulting file has the format required by the [load_specs\(\)](#) function. In the future, this step could be bypassed by creating the list output from [load_specs\(\)](#) directly. This function is based on a subset of the `RunFn()` function in the older ADMB version of this package.

Usage

```
write_specs_file(
  dir = getwd(),
  file_name = "data.spc",
  nreaders,
  biasopt = NULL,
  sigopt = NULL,
  knotages = NULL,
  maxage
)
```

Arguments

<code>dir</code>	Directory where the specifications file will be saved.
<code>file_name</code>	Name of the specifications file.
<code>nreaders</code>	An integer, specifying the number of readers.
<code>biasopt</code>	A vector with one entry for each reader specifying the type of bias specific to each reader. Positive values lead to estimated parameters and negative values are used for shared parameters between readers. Parameter sharing (mirroring) is common when there is more than one reader in a lab working together to refine their methods such that they have matching techniques. If NULL is passed, the default is <code>rep(0, nreaders)</code> which specifies that all readers are unbiased. Possible entries include the following: -[0-9]+ Mirror the bias of another reader, where the negative integer corresponds to the column of the reader that is being mirrored minus one, e.g., -1 causes it to mirror reader 1. Only lower-numbered readers can be mirrored. 0 Unbiased, where at least one reader has to be unbiased. 1 Constant coefficient of variation, i.e., a 1-parameter linear relationship of bias with true age. 2 Curvilinear, i.e., a 2-parameter Hollings-form relationship of bias with true age. An example entry for the situation where you have seven readers and you assume that the first reader is unbiased, readers 2-7 have a curvilinear bias, reader 3 shares parameters with reader 2, reader 5 shares parameters with reader 4, and reader 7 shares parameters with reader 6 would look like <code>c(0, 2, -2, 2, -4, 2, -6)</code>.
<code>sigopt</code>	A vector with one entry for each reader. Each entry specifies the functional form of reading error as a function of true age. Positive values lead to estimated parameters and negative values are used for shared parameters between readers. If NULL is passed, the default is <code>c(1, rep(-1, nreaders - 1))</code> which specifies a constant CV in ageing error which is shared among all readers. Possible entries include the following: -[0-9]+ Mirror the standard deviation of another reader, where the negative integer corresponds to the column of the reader that is being mirrored minus one, e.g., -1 causes it to mirror reader 1, for which data is stored in the second column of Data. This number must be lower than -1 times the current position in the vector. 0 No error. But, there could be potential bias. 1 Constant coefficient of variation, i.e., a 1-parameter linear relationship of the standard deviation with the true age. 2 Curvilinear standard deviation, i.e., a 3-parameter Hollings-form relationship of standard deviation with true age. 3 Curvilinear coefficient of variation, i.e., a 3-parameter Hollings-form relationship of coefficient of variation with true age. 5 Spline with estimated slope at beginning and end where the number of parameters is 2 + number of knots. Supported when knotages is provided.

- 6** Linear interpolation with a first knot of 1 and a last knot of the maximum age, i.e., MaxAge. Supported when knotages is provided.
 - 7** A linear change in the standard deviation of random age-reading error, σ_a , with age. This option has two parameters that need to be specified for each pair of independent readers in the specifications file.
 - 8** A linear change in the coefficient of variation of random age-reading error, CV_a , with age. This option has two parameters that need to be specified for each pair of independent readers in the specifications file.
- knotages A list of knot ages for each reader. This is required when sigopt = 5 or sigopt = 6 and must have one element per reader.
- maxage The maximum possible "true" age. This is only required in the specifications file if biasopt = 2.

Value

Invisibly returns the path to the specifications file (`file.path(dir, file_name)`).

Author(s)

Ian G. Taylor, James T. Thorson, Ian J. Stewart, Andre E. Punt

See Also

[write_files\(\)](#), [write_specs_file\(\)](#), [load_specs\(\)](#)

Examples

```
data_test <- data.frame(
  reader1 = c(7, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, 5, 8, 5),
  reader2 = c(8, 10, 7, 6, 6, 10, 7, 9, 8, 10, 10, 5, 6, 7, 9, 7, 7, NA, NA, NA),
  reader3 = c(7, 10, 7, 6, 6, 8, 7, 9, 8, 10, 10, 5, 6, 7, NA, NA, NA, NA, 5, 8, 5)
)
data_file <- write_data_file(data_test, dir = tempdir(), file_name = "test.dat")
specs_file <- write_specs_file(dir = tempdir(), nreaders = 3, file_name = "test.spc")
data <- load_data(DataFile = data_file)
specs <- load_specs(SpecsFile = specs_file, DataSpecs = data)
```

Index

ageing_comparison, [2](#)
ageing_comparison(), [11](#)
ageing_error_matrix_to_fims, [4](#)

cMx, [4](#)
CreateData, [5](#)
CreateSpecs, [5](#)

determine_n_sets, [6](#)
DoApplyAgeError, [6](#)

estgrowth.vb, [7](#)

fit, [8](#)

load_data, [9](#)
load_data(), [5](#), [7](#), [22](#), [23](#), [26](#)
load_specs, [9](#)
load_specs(), [5–7](#), [26](#), [28](#)

minimizer, [10](#)

plot_output, [11](#)
plot_output(), [12](#)
PlotOutputFn, [12](#)
ProcessResults, [12](#)

rMx, [13](#)
run, [14](#)
run(), [15](#)
RunFn, [15](#)

SimulatorFn, [16](#)
stats::optim(), [10](#)
StepwiseFn, [18](#)

tally_repeats, [21](#)
tally_repeats(), [22–24](#), [26](#)

write_data_file, [22](#)
write_files, [23](#)
write_files(), [14](#), [23](#), [28](#)
write_specs_file, [26](#)
write_specs_file(), [23](#), [26](#), [28](#)