Package: rema (via r-universe)

May 16, 2026

Type Package

Title A generalized framework to fit the random effects (RE) model, a state-space random walk model developed at the Alaska Fisheries Science Center (AFSC) for apportionment and biomass estimation of groundfish and crab stocks.

Version 1.2.0

Date 2022-12-15

Description This package provides a generalized framework to fit the random effects (RE) model, a state-space random walk model developed at the Alaska Fisheries Science Center (AFSC) for smoothing survey biomass estimates and apportioning catch among management areas. REMA is a multivariate extension of the original single-survey, single-strata RE model that allows the use of multiple strata within a survey and an additional survey (e.g. CPUE or relative population numbers) to inform the biomass trend (Hulson et al. 2021). If multi-survey mode is turned off, REMA runs the same as the univariate (RE) and multivariate (i.e. multiple area or depth strata; REM) versions of the model. REMA was developed in Template Model Builder (TMB; Kristensen et al. 2016).

Depends R (>= 3.6.0)

Imports TMB (>= 1.7.20), tidyr (>= 1.1.2), dplyr (>= 1.0.2), ggplot2 (>= 3.3.2), zoo (>= 1.7), sessioninfo (>= 1.1.1), abind (>= 1.4-5), rmarkdown (>= 2.11), knitr (>= 1.37), magrittr (>= 2.0.0)

LinkingTo TMB (>= 1.8.1), RcppEigen (>= 0.3.3.9.1)

URL https://github.com/JaneSullivan-NOAA/rema, https://afsc-assessments.github.io/rema/

BugReports https://github.com/JaneSullivan-NOAA/rema/issues

License GPL-3

LazyData true

VignetteBuilder knitr

RoxygenNote 7.3.1

Encoding UTF-8

Suggests testthat (>= 3.0.0), cowplot (>= 1.1.1), tmbstan, bayesplot, rstan

Remotes kaskr/tmbstan/tmbstan

Config/testthat/edition 3

Config/pak/sysreqs cmake make libicu-dev libuv1-dev

Repository https://jimianelli.r-universe.dev

Date/Publication 2025-01-11 00:47:12 UTC

RemoteUrl https://github.com/afsc-assessments/rema

RemoteRef HEAD

RemoteSha 9d801300099041f7f54c1f94dd6de42ab3d85eb1

Contents

`%>%` *Pipe function*

Description

Allows use of the pipe function, `%>%`

check_convergence *Check convergence of REMA model*

Description

Access quick convergence checks from ‘TMB‘ and ‘nlminb‘. Function modified from wham::check_convergence.

Usage

```
check_convergence(mod, ret = FALSE, f = "")
```

Arguments

mod	output from fit_rema
ret	T/F, return list? Default = FALSE, just prints to console

Value

a list with at least the first three of these components:

$convergence From stats::nlminb, "0 indicates successful convergence for nlminb"

$maxgr Max absolute gradient value, from ‘max(abs(mod$gr(mod$opt$par)))‘

$maxgr_par Name of parameter with max gradient

$is_sdrep If TMB::sdreport was performed for this model, this indicates whether it performed without error

$na_sdrep If TMB::sdreport was performed without error for this model, this indicates which (if any) components of the diagonal of the inverted hessian were returned as NA

See Also

fit_rema, fit_tmb, stats::nlminb

Examples

```
## Not run:
# placeholder for example

## End(Not run)
```

`check_estimability` *Check for identifiability of fixed effects Originally provided by https://github.com/kaskr/TMB_contrib_R/TMBhelper Internal function called by* `fit_tmb`.

Description

`check_estimability` calculates the matrix of second-derivatives of the marginal likelihood w.r.t. fixed effects, to see if any linear combinations are not estimable (i.e. cannot be uniquely estimated conditional upon model structure and available data, e.g., resulting in a likelihood ridge and singular, non-invertable Hessian matrix)

Usage

```
check_estimability(obj, h)
```

Arguments

`obj`	The compiled object
`h`	optional argument containing pre-computed Hessian matrix

Value

A tagged list of the hessian and the message

`compare_rema_models` *Plot REMA model comparisons and return AIC values when appropriate*

Description

Takes list of REMA models from from `fit_rema`, and returns a list of `ggplot2` objects to be plotted or saved, a list of `tidy_rema` data.frames, and AIC values.

Usage

```
compare_rema_models(
  rema_models,
  admb_re = NULL,
  save = FALSE,
  filetype = "png",
  path = NULL,
  xlab = NULL,
  biomass_ylab = "Biomass",
  cpue_ylab = "CPUE"
)
```

Arguments

rema_models	list of REMA models to be compared. Each REMA model in the list should be a list object output from fit_rema
admb_re	list of ADMB RE model input/output from read_admb_re. Accepts a single list, not list of multiple ADMB RE models. If admb_re is provided, no AIC calculations will be conducted.
save	(optional) logical (T/F) save figures as filetype in path. Default = FALSE. NOT YET IMPLEMENTED.
filetype	(optional) character string; type of figure file. Default = 'png'. NOT YET IMPLEMENTED.
path	(optional) directory path to location where figure files are to be saved if save = TRUE. NOT YET IMPLEMENTED.
xlab	(optional) label for x-axis of biomass and CPUE plots (e.g. 'Year'). Default = NULL.
biomass_ylab	(optional) label for y-axis of biomass plots (e.g. 'Biomass (t)'). Default = 'Biomass'.
cpue_ylab	(optional) label for y-axis of CPUE plots (e.g. 'Relative Population Number'). Default = 'CPUE'.

Value

a list with the following items:

$output A list of tidied dataframes that include parameter estimates, biomass and optional CPUE data, and REMA model predictions for each model to be compared. Results for a given variable are only included if they are applicable to all comparison models. For example, if CPUE is fit in one model but not another, compare$output$cpue_by_strata) will return an informational message instead of a dataframe. See tidy_rema for more information.

$plots ggplot2 figure objects of compare$output data.

$aic A dataframe of Akaike Information Criteria (AIC) values. Only output if the underlying models are fit to the same data.

See Also

tidy_rema, plot_rema

Examples

```
## Not run:
# placeholder for example

## End(Not run)
```

`extract_fixed` *Extract fixed effects Originally provided by https://github.com/kaskr/TMB_contrib_R/TMBhelper Internal function called by* `check_estimability`.

Description

`extract_fixed` extracts the best previous value of fixed effects, in a way that works for both mixed and fixed effect models

Usage

```
extract_fixed(obj)
```

Arguments

`obj`	The compiled object

Value

A vector of fixed-effect estimates

`fit_rema` *Fit REMA model*

Description

Fits the compiled REMA model using `TMB::MakeADFun` and `stats::nlminb`. Source code and documentation modified from wham::fit_wham.

Usage

```
fit_rema(
  input,
  n.newton = 0,
  do.sdrep = TRUE,
  model = NULL,
  do.check = FALSE,
  MakeADFun.silent = TRUE,
  do.fit = TRUE,
  save.sdrep = TRUE
)
```

Arguments

input	Named list output from prepare_rema_input, which includes the following components needed to fit model using TMB::MakeADFun: $data Data, a list of data objects for model fitting or specification (e.g., user-defined pentalties, index pointers, etc.). A required input to MakeADFun. $par Parameters, a list of all random and fixed effects parameter objects. A required input to MakeADFun. $map Map, a mechanism for collecting and fixing parameters in TMB. An input to MakeADFun. $random Character vector defining the parameters to treat as random effects. An input to MakeADFun. $model_name Character, name of the model, e.g. "GOA shortraker with LLS by depth strata". Useful for model comparison.
n.newton	integer, number of additional Newton steps after optimization. Not an option that is currently needed, but is passed to fit_tmb. Default = 0.
do.sdrep	T/F, calculate standard deviations of model parameters? See sdreport. Default = TRUE.
model	(optional), a previously fit rema model.
do.check	T/F, check if model parameters are identifiable? Passed to fit_tmb. Runs internal function check_estimability, originally provided by https://github.com/kaskr/TMB_contrib_R/TM Default = TRUE.
MakeADFun.silent	T/F, Passed to silent argument of TMB::MakeADFun. Default = TRUE.
do.fit	T/F, fit the model using fit_tmb. Default = TRUE.
save.sdrep	T/F, save the full TMB::sdreport object? If FALSE, only save summary.sdreport to reduce model object file size. Default = TRUE.

Details

Future development: Implement one-step-ahead (OSA) residuals for evaluating model goodness-of-fit TMB::oneStepPredict). OSA residuals are more appropriate than standard residuals for models with random effects (Thygeson et al. (2017). See wham for an example of OSA implementation and additional OSA residual options (e.g. full Gaussian approximation instead of the (default) generic method using osa.opts=list(method="fullGaussian").

Value

a fit TMB model with additional output if specified:

$rep List of derived quantity estimates (e.g. estimated biomass)

$sdrep Parameter estimates (and standard errors if do.sdrep = TRUE)

See Also

fit_tmb, TMB::oneStepPredict

Examples

```
## Not run:
# place holder for example code

## End(Not run)
```

`fit_tmb` *Fit TMB model using nlminb*

Description

Runs optimization on the TMB model using `stats::nlminb`. If specified, takes additional Newton steps and calculates standard deviations. Internal function called by `fit_rema`. Source code and documentation modified from wham::fit_tmb.

Usage

```
fit_tmb(
  model,
  n.newton = 0,
  do.sdrep = TRUE,
  do.check = FALSE,
  save.sdrep = FALSE
)
```

Arguments

`model`	Output from `TMB::MakeADFun`.
`n.newton`	Integer, number of additional Newton steps after optimization. Default = `0`.
`do.sdrep`	T/F, calculate standard deviations of model parameters? See `TMB::sdreport`. Default = `TRUE`.
`do.check`	T/F, check if model parameters are identifiable? Runs internal `check_estimability`, originally provided by https://github.com/kaskr/TMB_contrib_R/TMBhelper. Default = `TRUE`.
`save.sdrep`	T/F, save the full `TMB::sdreport` object? If `FALSE`, only save `summary.sdreport)` to reduce model object file size. Default = `FALSE`.

Value

`model`, appends the following:

`model$opt` Output from `stats::nlminb`

`model$date` System date

`model$dir` Current working directory

`model$rep` model$report()

`model$TMB_version` Version of TMB installed

`model$parList` List of parameters, `model$env$parList()`

`model$final_gradient` Final gradient, `model$gr()`

`model$sdrep` Estimated standard deviations for model parameters, `TMB::sdreport` or `summary.sdreport)`

See Also

`fit_rema`, `TMBhelper::check_estimability`

`get_osa_residuals` *Get one-step-head (OSA)*

Description

Takes the rema model output from `fit_rema` and returns OSA residuals calculated using `TMB::oneStepPredict` with accompanying residual analysis plots. IMPORTANT: OSA residuals do not work for users implementing the Tweedie distribution.

Usage

```
get_osa_residuals(
  rema_model,
  options = list(method = "fullGaussian", parallel = TRUE)
)
```

Arguments

`rema_model`	list out output from `fit_rema`, which includes model results but also inputs. Of note to OSA residual calculations is the `rema_model$input$osa` object, which is a data.frame containing all the data or observations fit in the model that will have a residuals associated with them.
`options`	list of options for calculating OSA residuals, passed to `TMB::oneStepPredict`. Default: `options = list(method = "fullGaussian", parallel = TRUE)`. Alternative methods include "cdf", "oneStepGeneric", "oneStepGaussianOffMode", and "oneStepGaussian".

Value

a list of tidied data.frames containing the biomass and CPUE survey residuals with accompanying data, as well as a QQ-plot, histogram of residuals, and plots of residuals~year and residuals~fitted values by strata for the biomass and CPUE survey.

See Also

`tidy_rema`

Examples

```
## Not run:
# placeholder for example

## End(Not run)
```

`plot_extra_cv` *Plot the additional estimated observation error for biomass by strata and/or cpue by strata*

Description

Takes list output from `tidy_rema` and returns a list of ggplot2 objects to be plotted or saved.

Usage

```
plot_extra_cv(
  tidy_rema,
  save = FALSE,
  filetype = "png",
  path = NULL,
  xlab = NULL,
  biomass_ylab = "Biomass",
  cpue_ylab = "CPUE"
)
```

Arguments

`tidy_rema`	list out output from `tidy_extra_cv`, which includes inputs, model results, and confidence intervals for the total observation error (fixed + estimated)
`save`	(optional) logical (T/F) save figures as `filetype` in `path`. Default = FALSE. NOT YET IMPLEMENTED.
`filetype`	(optional) character string; type of figure file. Default = 'png'. NOT YET IMPLEMENTED.
`path`	(optional) directory path to location where figure files are to be saved if `save = TRUE`. NOT YET IMPLEMENTED.
`xlab`	(optional) label for x-axis of biomass and CPUE plots (e.g. 'Year'). Default = NULL.
`biomass_ylab`	(optional) label for y-axis of biomass plots (e.g. 'Biomass (t)'). Default = 'Biomass'.
`cpue_ylab`	(optional) label for y-axis of CPUE plots (e.g. 'Relative Population Number'). Default = 'CPUE'.

Value

a list of ggplot2 plots or character string messages about the data. Except for parameter estimates, the objects output from `tidy_rema` are the same outputted from this function.

See Also

tidy_rema

Examples

```
## Not run:
# placeholder for example

## End(Not run)
```

---

plot_rema *Plot survey data and model output*

Description

Takes list output from tidy_rema and returns a list of ggplot2 objects to be plotted or saved.

Usage

```
plot_rema(
  tidy_rema,
  save = FALSE,
  filetype = "png",
  path = NULL,
  xlab = NULL,
  biomass_ylab = "Biomass",
  cpue_ylab = "CPUE"
)
```

Arguments

tidy_rema	list out output from tidy_rema, which includes model results but also inputs
save	(optional) logical (T/F) save figures as filetype in path. Default = FALSE. NOT YET IMPLEMENTED.
filetype	(optional) character string; type of figure file. Default = 'png'. NOT YET IMPLEMENTED.
path	(optional) directory path to location where figure files are to be saved if save = TRUE. NOT YET IMPLEMENTED.
xlab	(optional) label for x-axis of biomass and CPUE plots (e.g. 'Year'). Default = NULL.
biomass_ylab	(optional) label for y-axis of biomass plots (e.g. 'Biomass (t)'). Default = 'Biomass'.
cpue_ylab	(optional) label for y-axis of CPUE plots (e.g. 'Relative Population Number'). Default = 'CPUE'.

Value

a list of ggplot2 plots or character string messages about the data. Except for parameter estimates, the objects output from tidy_rema are the same outputted from this function.

See Also

tidy_rema

Examples

```
## Not run:
# placeholder for example

## End(Not run)
```

---

prepare_rema_input *Prepare input data and parameters for REMA model*

Description

After the data is read into R (either manually from a .csv or other data file or by using read_admb_re), this function prepares the data and parameter settings for fit_rema. The model can be set up to run in single survey mode with one or more strata, or in multi-survey mode, which uses an additional relative abundance index (i.e. cpue) to inform predicted biomass. The optional inputs described below related to the CPUE survey data or scaling parameter q, such as cpue_dat and options_q are only used when multi_survey = 1. The function structure and documentation is modeled after wham::prepare_wham_input.

Usage

```
prepare_rema_input(
  model_name = "REMA for unnamed stock",
  multi_survey = 0,
  admb_re = NULL,
  biomass_dat = NULL,
  cpue_dat = NULL,
  sum_cpue_index = FALSE,
  start_year = NULL,
  end_year = NULL,
  wt_biomass = NULL,
  wt_cpue = NULL,
  PE_options = NULL,
  q_options = NULL,
  zeros = NULL,
  extra_biomass_cv = NULL,
  extra_cpue_cv = NULL
)
```

Arguments

model_name	name of stock or other identifier for REMA model
multi_survey	switch to run model in single or multi-survey mode. 0 (default) = single survey, 1 = multi-survey.
admb_re	list object returned from read_admb_re.R, which includes biomass survey data (admb_re$biomass_dat), optional cpue survey data (admb_re$cpue_dat), years for model predictions (admb_re$model_yrs), and model predictions of log biomass by strata in the correct format for input into REMA (admb_re$init_log_biomass_pred). If supplied, the user does not need enter biomass_dat or cpue_dat.
biomass_dat	data.frame of biomass survey data in long format with the following columns: strata character; the survey name, survey region, management unit, or depth strata. Note that the user must include this column even if there is only one survey strata year integer; survey year. Note that the user only needs to include years for which there are observations (i.e. there is no need to supply NULL or NA values for missing survey years) biomass numeric; the biomass estimate/observation (e.g. bottom trawl survey biomass in mt). By default, if biomass == 0, this value will be treated as an NA (i.e., a failed survey). If the user wants to make other assumptions about zeros (e.g. adding a small constant), they must define it in the data manually. cv numeric; the coefficient of variation (CV) of the biomass estimate (i.e. sd(biomass)/biomass)
cpue_dat	(optional) data.frame of relative abundance index (i.e. cpue) data in long format with the following columns: strata character; the survey name, survey region, management unit, or depth strata (note that the user must include this column even if there is only one survey strata) year integer; survey year. Note that the user only needs to include years for which there are observations (i.e. there is no need to supply NULL or NA values for missing survey years) cpue numeric; the cpue estimate/observation (e.g. longline survey cpue or relative population number); By default, if cpue == 0, this value will be treated as an NA (i.e., a failed survey). If the user wants to make other assumptions about zeros (e.g. adding a small constant), they must define it in the data manually. cv numeric; the coefficient of variation (CV) of the cpue estimate (i.e. sd(cpue)/cpue)
sum_cpue_index	T/F or 1/0, is the CPUE survey index able to be summed across strata to get a total CPUE survey index? For example, Longline survey relative population numbers (RPNs) are summable but longline survey numbers per hachi (CPUE) are not. Default = FALSE.
start_year	(optional) integer value specifying the start year for estimation in the model; if admb_re is supplied, this value defaults to start_year = min(admb_re$model_yrs); if admb_re is not supplied, this value defaults to the first year in either biomass_dat or cpue_dat

end_year	(optional) integer value specifying the last year for estimation in the model; if admb_re is supplied, this value defaults to end_year = max(admb_re$model_yrs); if admb_re is not supplied, this value defaults to the last year in either biomass_dat or cpue_dat
wt_biomass	(optional) a multiplier on the biomass survey data component of the negative log likelihood. For example, nll = wt_biomass * nll. Defaults to wt_biomass = 1
wt_cpue	(optional) a multiplier on the CPUE survey data component of the negative log likelihood. For example, nll = wt_cpue * nll. Defaults to wt_cpue = 1
PE_options	(optional) customize implementation of process error (PE) parameters, including options to share PE across biomass survey strata, change starting values, fix parameters, and add penalties or priors (see details)
q_options	(optional) customize implementation of scaling parameters (q), including options to define q by biomass or cpue survey cpue strata, change starting values, fix parameters, and add penalties or priors (see details). only used when multi_survey = 1
zeros	(optional) define assumptions about how to treat zero biomass or CPUE observations, including treating zeros as NAs, changing the zeros to small constants with fixed CVs, or modeling the zeros using a Tweedie distribution (see details).
extra_biomass_cv	(optional) estimate additional observation error for the biomass survey data (see details). By default, assumption = "extra_cv" will estimate one extra CV parameter, regardless of the number of biomass survey strata.
extra_cpue_cv	(optional) estimate additional observation error for the CPUE survey data (see details). By default, assumption = "extra_cv" will estimate one extra CV parameter, regardless of the number of CPUE survey strata.

Details

PE_options allows the user to specify options for process error (PE) parameters. If NULL, default PE specifications are used: one PE parameter is estimated for each biomass survey strata, initial values for log_PE are set to 1, and no penalties or priors are added. The user can modify the default PE_options using the following list of entries:

$pointer_PE_biomass An index to customize the assignment of PE parameters to individual biomass strata. Vector with length = number of biomass strata, starting with an index of 1 and ending with the number of unique PE estimated. For example, if there are three biomass survey strata and the user wants to estimate only one PE, they would specify pointer_PE_biomass = c(1, 1, 1). By default there is one unique log_PE estimated for each unique biomass survey stratum

$initial_pars A vector of initial values for log_PE. The default initial value for each log_PE is 1.

$fix_pars Option to fix PE parameters, where the user specifies the index value of the PE parameter they would like to fix at the initial value. For example, if there are three biomass survey strata, and the user wants to fix the log_PE for the second stratum but estimate the log_PE for the first and third strata they would specify fix_pars = c(2) Note that this option is not recommended.

$penalty_options Warning: the following options are experimental and not well-tested. Options for penalizing the PE likelihood or adding a prior on log_PE include the following:

"none" (default) no penalty or prior used

"wt" a multiplier on the PE and random effects component of the negative log likelihood. For example, nll = wt * nll, where wt = 1.5 is specified as a single value in the penalty_values argument

"squared_penalty" As implemented in an earlier version of the RE.tpl, this penalty prevents the PE from shrinking to zero. For example, `nll = nll + (log_PE + squared_penalty)^2`, where `squared_penalty = 1.5`. A vector of `squared_penalty` values is specified for each PE in the `penalty_values` argument

"normal_prior" Normal prior in log space, where `nll = nll - dnorm(log_PE, pmu_log_PE, psig_log_PE, 1)` and `pmu_log_PE` and `psig_log_PE` are specified for each PE parameter in the `penalty_values` argument

penalty_values user-defined values for the `penalty_options`. Each penalty type will is entered as follows:

"none" (default) NULL For example, `penalty_values = NULL`

"wt" a single numeric value. For example, `penalty_values = 1.5`

"squared_penalty" a vector of numeric values with length = number of PE parameters. For example, if three PE parameters are being estimated and the user wants them to have the same penalty for each one, they would use `penalty_values = c(1.5, 1.5, 1.5)`

"normal_prior" a vector of paired values for each PE parameter, where each vector pair is the prior mean of log_PE `pmu_log_PE` and the associated standard deviation `psig_log_PE`. For example, if three PE parameters are being estimated and the user wants them to have the same normal prior of log_PE ~ N(1.0, 0.08), `penalty_values = c(c(1.0, 0.08), c(1.0, 0.08), c(1.0, 0.08))`

`q_options` allows the user to specify options for the CPUE survey scaling parameters (q). If `multi_survey = 0` (default), no q parameters are estimated regardless of what the user defines in `q_options`. `multi_survey = 0` and `q_options = NULL`, default q specifications are used: one q parameter is estimated for each CPUE survey strata, biomass and CPUE surveys are assumed to share strata definitions (i.e., `biomass_dat` and `cpue_dat` have the same number of columns and the columns represent the same strata), initial values for `log_q` are set to 1, and no penalties or priors are added. The user can modify the default `q_options` using the following list of entries:

$pointer_q_cpue An index to customize the assignment of q parameters to individual CPUE survey strata. Vector with length = number of CPUE strata, starting with an index of 1 and ending with the number of unique q parameters estimated. For example, if there are three CPUE survey strata and the user wanted to estimate only one q, they would specify `pointer_q_cpue = c(1, 1, 1)`. The recommended model configuration is to estimate one log_q for each CPUE survey stratum.

$pointer_biomass_cpue_strata An index to customize the assignment of biomass predictions to individual CPUE survey strata. Vector with length = the number of biomass survey strata, starting with an index of 1 and ending with the number of unique CPUE survey strata. This pointer only needs to be defined if the number of biomass and CPUE strata are not equal. The `pointer_biomass_cpue_strata` option allows the user to calculate predicted biomass at the CPUE survey strata level under the scenario where the biomass survey strata is at a higher resolution than the CPUE survey strata. For example, if there are 3 biomass survey strata that are represented by only 2 CPUE survey strata, the user may specify `pointer_biomass_cpue_strata = c(1, 1, 2)`. This specification would assign the first 2 biomass strata to the first CPUE

strata, and the third biomass stratum to the second CPUE stratum. If there is no CPUE data to compliment a specific biomass stratum, the user can populate these with NAs. For example if pointer_biomass_cpue_strata = c(1, NA, 3), it means there is CPUE data for biomass strata 1 and 3 but not 2. NOTE: there cannot be a scenario where there are more CPUE survey strata than biomass survey strata because the CPUE survey is used to inform the biomass survey trend. An error will be thrown if `q_options$pointer_biomass_cpue_strata` is not defined and the biomass and CPUE survey strata definitions are not the same.

$initial_pars A vector of initial values for `log_q`. The default initial value for each log_q is 1.

$fix_pars Option to fix q parameters, where the user specifies the index value of the q parameter they would like to fix at the initial value. For example, if there are three CPUE survey strata, and the user wants to fix the `log_q` for the second stratum but estimate the `log_q` for the first and third strata they would specify `fix_pars = c(2)`

$penalty_options Options for penalizing the q likelihood or adding a prior on `log_q` include the following:

"none" (default) no penalty or prior used

"normal_prior" Warning, experimental and not well-tested. Normal prior in log space, where `nll = nll - dnorm(log_q, pmu_log_q, psig_log_q, 1)` and `pmu_log_q` and `psig_log_q` are specified for each q parameter in the `penalty_values` argument

penalty_values user-defined values for the `penalty_options`. Each penalty type will is entered as follows:

"none" (default) NULL For example, `penalty_values = NULL`

"normal_prior" a vector of paired values for each q parameter, where each vector pair is the prior mean of log_q `pmu_log_q` and the associated standard deviation `psig_log_q`. For example, if 2 q parameters are being estimated and the user wants them to have the same normal prior of log_q ~ N(1.0, 0.05), `penalty_values = c(c(1.0, 0.05), c(1.0, 0.05))`

`zeros` allows the user to specify options for how to treat zero biomass or CPUE survey observations. By default zero observations are treated as NAs and a warning msg to that effect is returned to the console. `zeros` allows the user to specify non-default zero assumptions using the following list of entries:

$assumption character, name of assumption using. Only three alternatives are currently implemented, `zeros = list(assumption = c("NA", "small_constant", "tweedie"))`. `"NA"` is the default; this option assumes the zero estimates are failed surveys and removes them. `"small_constant"` is an ad hoc method where a fixed value is added to the zero with an assumed CV. By default, the small constant = 0.0001 and the CV is the value entered by the user in the data. The user can change the assumed value and CV using `options_small_constant`. `"tweedie"` uses the Tweedie as the assumed error distribution of the survey data, which allows zeros. This alternative estimates one additional power parameter. The assumed CV for zero biomass or zero cpue survey observations defaults to 1.5. The user can change this assumed CV, change initial values for the inverse logit transformed power parameter, or fix it at initial values using `options_tweedie`.

$options_small_constant a vector length of two numeric values. The first value is the small constant to add to the zero observation, the second is the user-defined coefficient for this value. The user can specify the small value but use the input CV by specifying an NA for the second value. E.g., 'options_small_constant = c(0.0001, NA)'.

$options_tweedie a list of entries to control initial or fixed values for Tweedie parameters. Currently, this argument accepts the following entries:

$zeros_cv Change the assumed CV of zero biomass or cpue survey observations. Default CV = 1.5. This input accepts a positive, non-zero numeric value.

$initial_pars Input to change initial values. In single-survey mode, `zeros$options_tweedie$initial_pars` must be a vector of numeric values with length = 1 `c(logit_tweedie_p)`. In multi-survey mode, `zeros$options_tweedie$initial_pars` must be a vector of numeric values with length = 2 `c(biomass survey logit_tweedie_p, cpue survey logit_tweedie_p)`. Initial values for log_tweedie_dispersion should be in log space. Initial values for logit_tweedie_p < -10 approach tweedie_p = 1 (zero-inflated Poisson), logit_tweedie_p > 10 approach tweedie_p = 2 (gamma).

$fix_pars `zeros$options_tweedie$fix_pars` must be a vector of integer value(s) with the index value (starting at 1) of logit_tweedie_p parameters to be fixed. For example, in single survey mode, if the user wants to fix the biomass survey logit_tweedie_p, they should enter `zeros = list(assumption = 'tweedie', options_tweedie = list(fix_pars = c(1)))`. In multi-survey, if they want to fix only the cpue survey log_tweedie_p but estimate the biomass survey log_tweedie_p, they should enter `zeros = list(assumption = 'tweedie', options_tweedie = list(fix_pars = c(2)))`.

`extra_biomass_cv` allows the user to specify options for estimating an additional CV parameter (`log_tau_biomass` in the source code, estimated in log-space) for the biomass survey observations. If `extra_biomass_cv = NULL` (default), no extra CV is estimated. The user can modify the default `extra_biomass_cv` options using the following list of entries:

$assumption A string identifying what assumption is used for the biomass survey observations. Options include "none" (default in which no extra CV is estimated) or "extra_cv". If assumption = "extra_cv", by default only one extra CV will be estimated, regardless of how many biomass strata are defined. If `extra_biomass_cv` is not NULL, user must define appropriate assumption.

$pointer_extra_biomass_cv An index to customize the assignment of extra CV parameters to individual biomass survey strata. Vector with length = number of biomass strata, starting with an index of 1 and ending with the number of unique extra CV parameters estimated. If there are three biomass survey strata and user wanted to estimate an extra CV per stratum, they would specify `pointer_extra_biomass_cv = c(1, 2, 3)`. By default, only one additional parameter is estimated, regardless of how many strata are defined (i.e. `pointer_extra_biomass_cv = c(1, 1, 1)`).

$initial_pars A vector of initial values for the extra biomass `log_tau_biomass`. The default initial value for each log_tau_biomass is log(1e-7) (approximately 0 on the arithmetic scale).

$fix_pars Option to fix extra biomass CV parameters, where the user specifies the index value of the parameter they would like to fix at the initial value. For example, if there are three biomass survey strata defined in `pointer_extra_biomass_cv`, and the user wants to fix the `log_tau_biomass` for the second stratum but estimate the `log_tau_biomass` for the first and third strata they would specify `fix_pars = c(2)`.

`extra_cpue_cv` allows the user to specify options for estimating an additional CV parameter (`log_tau_cpue` in the source code, estimated in log-space) for the cpue survey observations. If `extra_cpue_cv = NULL` (default), no extra CV is estimated. The user can modify the default `extra_cpue_cv` options using the following list of entries:

$assumption A string identifying what assumption is used for the cpue survey observations. Options include "none" (default in which no extra CV is estimated) or "extra_cv". If assumption = "extra_cv", by default only one extra CV will be estimated, regardless of how many cpue strata are defined. If `extra_cpue_cv` is not NULL, user must define appropriate assumption.

$pointer_extra_cpue_cv An index to customize the assignment of extra CV parameters to individual cpue survey strata. Vector with length = number of cpue strata, starting with an index of 1 and ending with the number of unique extra CV parameters estimated. If there are three cpue survey strata and user wanted to estimate an extra CV per stratum, they would specify `pointer_extra_cpue_cv = c(1, 2, 3)`. By default, only one additional parameter is estimated, regardless of how many strata are defined (i.e. `pointer_extra_cpue_cv = c(1, 1, 1)`).

$initial_pars A vector of initial values for the extra cpue `log_tau_cpue`. The default initial value for each log_tau_cpue is log(1e-7) (approximately 0 on the arithmetic scale).

$fix_pars Option to fix extra cpue CV parameters, where the user specifies the index value of the parameter they would like to fix at the initial value. For example, if there are three cpue survey strata defined in `pointer_extra_cpue_cv`, and the user wants to fix the `log_tau_cpue` for the second stratum but estimate the `log_tau_cpue` for the first and third strata they would specify `fix_pars = c(2)`.

Value

This function returns a named list with the following components:

`data` Named list of data, passed to `TMB::MakeADFun`

`par` Named list of parameters, passed to `TMB::MakeADFun`

`map` Named list defining how to optionally collect and fix parameters, passed to `TMB::MakeADFun`

`random` Character vector of parameters to treat as random effects, passed to `TMB::MakeADFun`

`model_name` Name of stock or other identifier for REMA model

`biomass_dat` A tidied long format data.frame of the biomass survey observations and associated CVs by strata. This data.frame will be 'complete' in that it will include all modeled years, with missing values treated as NAs. Note that this data.frame could differ from the `admb_re$biomass_dat` or input `biomass` if assumptions about zero biomass observations are different between the ADMB model and what the user specifies for REMA. The user can change their assumptions about zeros using the `zeros` argument.

`cpue_dat` If optional CPUE survey data are provided and `multi_survey = 1`, this will be a tidied long-format data.frame of the CPUE survey observations and associated CVs by strata. This data.frame will be 'complete' in that it will include all modeled years, with missing values treated as NAs. Note that this data.frame could differ from the `admb_re$biomass_dat` or input `biomass` if assumptions about zero CPUE observations are different between the ADMB model and what the user specifies for REMA. The user can change their assumptions about zeros using the `zeros` argument. If optional CPUE survey data are not provided or `multi_survey = 0`, this object will be `NULL`.

Examples

```
## Not run:
# place holder for example code
```

```
## End(Not run)
```

`read_admb_re` *Convert ADMB version of the RE model data and output to REMA inputs*

Description

Read the report file from the ADMB version of the RE model (rwout.rep) and convert it into long format survey data estimates with CVs for input into REMA.

Usage

```
read_admb_re(
  filename,
  model_name = "Unnamed ADMB RE model",
  biomass_strata_names = NULL,
  cpue_strata_names = NULL
)
```

Arguments

`filename`	name of ADMB output file to be read (e.g. rwout.rep)
`model_name`	(optional) Name of stock and identifier for the ADMB version of the RE model. Defaults to 'ADMB RE'
`biomass_strata_names`	(optional) a vector of character names corresponding to the names of the biomass survey strata. Vector should be in the same order as the columns of srv_est in rwout.rep
`cpue_strata_names`	(optional) a vector of character names corresponding to the names of the CPUE survey strata. Vector should be in the same order as the columns of srv_est_LL in rwout.rep in the version of the ADMB RE model that accepts an additional survey index

Value

object of type "list" with biomass optional cpue survey data in long format, and initial parameter values for log_biomass_pred (the random effects matrix), ready for input into REMA

a list with the following items:

`$biomass_dat` A dataframe of biomass survey data with strata, year, biomass estimates, and CVs. Note that the CVs have been back-transformed to natural space.

`$cpue_dat` Optional dataframe of CPUE survey data with strata, year, CPUE estimates, and CVs. Note that the CVs have been back-transformed to natural space.

`$model_yrs` Vector of prediction years.

`$init_log_biomass_pred` Matrix of initial parameter values for log_biomass_pred (the random effects matrix), ready for input into REMA.

`$admb_re_results` A list of ADMB RE model results ready for comparison with REMA models using compare_rema_models(). User beware... there are many, many versions of the RE.tpl in existence and individual variances may cause errors in this output.

Examples

```
## Not run:
# place holder for example code

## End(Not run)
```

`read_rep` *Read ADMB .rep file and return an R object of type 'list'*

Description

Code modified from original function provided by Steve Martell, D'Arcy N. Webber called by `read_admb_re`

Usage

```
read_rep(fn)
```

Arguments

`fn`	full path and name of ADMB output file to be read

Value

object of type "list" with ADMB outputs therein

Examples

```
## Not run:
read_rep(fn = 'inst/example_data/goasr.rep')

## End(Not run)
```

tidy_extra_cv *Tidy estimates of extra biomass or CPUE index CV*

Description

Takes list output from tidy_rema and returns the same list with enhanced versions of the biomass_by_strata and cpue_by_strata when appropriate. These enhanced dataframes include three new columns, tot_log_obs_cv, tot_obs_lci, and tot_obs_uci, which represent combined log-space standard error and associated confidence intervals that include both assumed and estimated additional observation error.

Usage

```
tidy_extra_cv(tidy_rema, save = FALSE, path = NULL, alpha_ci = 0.05)
```

Arguments

tidy_rema	list out output from tidy_rema, which includes model results but also inputs
save	(optional) logical (T/F) save figures as filetype in path. Default = FALSE. NOT YET IMPLEMENTED.
path	(optional) directory path to location where figure files are to be saved if save = TRUE. NOT YET IMPLEMENTED.
alpha_ci	(optional) the significance level for generating confidence intervals. Default = 0.05

Value

a list with the following items:

$parameter_estimates A data.frame of fixed effects parameters in REMA (e.g. log_PE and log_q) with standard errors and confidence intervals that have been transformed from log space to natural space for ease of interpretation.

$biomass_by_strata A tidy, long format data.frame of model predicted and observed biomass by biomass survey strata. This data.frame is now enhanced with new columns that include log-space standard error and associated confidence intervals that account for additional estimated observation error.

$cpue_by_strata A tidy, long format data.frame of model predicted and observed CPUE by CPUE survey strata. This data.frame is now enhanced with new columns that include log-space standard error and associated confidence intervals that account for additional estimated observation error. If REMA is not run in multi-survey mode, or if CPUE data are not provided, an explanatory character string with instructions for fitting to CPUE data is returned.

$biomass_by_cpue_strata A tidy, long format data.frame of model predicted biomass by CPUE survey strata. Note that observed/summed biomass observations are not returned in case there are missing values in one stratum but not another within a given year. This output is reserved for instances when the number of biomass strata exceeds that of CPUE survey strata, but the user wants to visualize predicted biomass at the same resolution as the CPUE predictions. In other scenarios, a character string is returned explaining the special use case for this object.

`$total_predicted_biomass` A tidy, long format data.frame of total model predicted biomass summed across all biomass survey strata. If only one stratum is used (i.e. the univariate RE), the predicted values will be the same as `output$biomass_by_strata`.

`$total_predicted_cpue` A tidy, long format data.frame of total model predicted CPUE summed across all CPUE survey strata. If only one stratum is used (i.e. the univariate RE), the predicted values will be the same as `output$cpue_by_strata`. If The CPUE survey index provided was defined as not summable in prepare_rema_input(), an character string will be returned explaining how to change this using the 'sum_cpue_index' in `?prepare_rema_input` if appropriate.

See Also

`tidy_rema`

Examples

```
## Not run:
# placeholder for example

## End(Not run)
```

`tidy_rema` *Tidy REMA model output*

Description

Takes outputs from `fit_rema`, and returns a named list of tidied data.frames that include parameter estimates and standard errors, and derived variables from the model. For more information on "tidy" data, please see Wickham 2014. Some code modified from `wham::par_tables_fun`.

Usage

```
tidy_rema(rema_model, save = FALSE, path = NULL, alpha_ci = 0.05)
```

Arguments

`rema_model`	list out output from `fit_rema`, which includes model results but also inputs
`save`	(optional) logical (T/F) save output data.frames as csvs in `path`. Default = FALSE. NOT YET IMPLEMENTED.
`path`	(optional) directory path to location where csvs are to be saved if `save = TRUE`. NOT YET IMPLEMENTED.
`alpha_ci`	(optional) the significance level for generating confidence intervals. Default = 0.05

Value

a list with the following items:

`$parameter_estimates` A data.frame of fixed effects parameters in REMA (e.g. log_PE and log_q) with standard errors and confidence intervals that have been transformed from log space to natural space for ease of interpretation.

`$biomass_by_strata` A tidy, long format data.frame of model predicted and observed biomass by biomass survey strata.

`$cpue_by_strata` A tidy, long format data.frame of model predicted and observed CPUE by CPUE survey strata. If REMA is not run in multi-survey mode, or if CPUE data are not provided, an explanatory character string with instructions for fitting to CPUE data is returned.

`$biomass_by_cpue_strata` A tidy, long format data.frame of model predicted biomass by CPUE survey strata. Note that observed/summed biomass observations are not returned in case there are missing values in one stratum but not another within a given year. This output is reserved for instances when the number of biomass strata exceeds that of CPUE survey strata, but the user wants to visualize predicted biomass at the same resolution as the CPUE predictions. In other scenarios, a character string is returned explaining the special use case for this object.

`$total_predicted_biomass` A tidy, long format data.frame of total model predicted biomass summed across all biomass survey strata. If only one stratum is used (i.e. the univariate RE), the predicted values will be the same as `output$biomass_by_strata`.

`$total_predicted_cpue` A tidy, long format data.frame of total model predicted CPUE summed across all CPUE survey strata. If only one stratum is used (i.e. the univariate RE), the predicted values will be the same as `output$cpue_by_strata`. If The CPUE survey index provided was defined as not summable in prepare_rema_input(), an character string will be returned explaining how to change this using the 'sum_cpue_index' in `?prepare_rema_input` if appropriate.

See Also

`fit_rema`

Examples

```
## Not run:
# placeholder for example

## End(Not run)
```

Index