Package 'IRATER'

October 24, 2016

Type Package

Title A R Interface for the Instantaneous RATEs (IRATE) Model

Version 0.0.1

Date 2016-10-24

Author Robert Bauer, Gary Nelson

Maintainer Robert Bauer <robert.bauer@ird.fr>

Description A R interface to setup, run and read IRATE model runs to assess band recovery (conventional tagging) data (i.e. age-dependent or independent fishing and natural mortality rates).

Depends R (>= 3.0.1)

Imports coda, lattice, R2admb, plyr, stats

SystemRequirements AD Model Builder <http://admb-project.org>

License GPL

LazyLoad yes

Repository CRAN

Date/Publication 2016-10-24 12:03:00

NeedsCompilation no

R topics documented:

---

clean.admb	*delete IRATE (ADMB) run files*

Description

delete IRATE (ADMB) run files (not model setup files).

Usage

```
clean.admb(setup.name)
clean.IRATE(setup.name)
```

Arguments

setup.name   Character string defining preinstalled IRATE example setups/datasets to be run.
                 If missing or not valid, the user will be asked to choose among valid examples.
                 Check IRATE.examples() for valid, preinstalled IRATE examples.

Author(s)

Robert K. Bauer

See Also

compile_admb, run_admb, IRATE.examples

Examples

```
example.setup <- IRATE.examples()
new.setup <- new.setup.path <- example.setup[1] # select old IRATE setup to reparameterize
print(new.setup) # print setup name to be run
old.setup.path <- system.file(paste0("IRATE.examples/",new.setup,".dat"), package = "IRATER")

system(paste("mkdir -p ",new.setup.path)) # create run folder for new setup
system(paste("cp",old.setup.path, new.setup.path)) # copy old setup in new run folder
setwd(new.setup.path)

run.IRATE(new.setup) # compile and run setup
clean.IRATE(new.setup) # deletes run files (leaving only the setup file for reruns)
```

IRATE.examples	*list preinstalled IRATE examples*

Description

list preinstalled IRATE examples

Usage

```
IRATE.examples()
```

Value

HoenigNonMixing Age-dependent, harvest only (no catch-&-release) conventional tagging dataset from 1960-1964 of lake trout (Salvelinus namaycush), assuming an incomplete mixing of tagged fish withe the wild population after being released. Based on: Hoenig et al. 1998 (incomplete mixing Table 4 for more details).

JiangADCR Age-dependent, harvest and catch-&-release conventional tagging dataset of Striped bass (Morone saxatilis) from 1991-2003, assuming an incomplete mixing of tagged fish withe the wild population after being released (Similar to the JiangADH and JiangAICR data sets). Based on: Jiang et al., 2007 (Table 2 Model b).

JiangADH Age-dependent, harvest only (no catch-&-release) conventional tagging dataset of Striped bass (Morone saxatilis) from 1991-2003 (Similar to the JiangADH and JiangAICR data sets), assuming an incomplete mixing of tagged fish withe the wild population after being released. Based on: Jiang et al., 2007b (Table 4 Model a).

JiangAICR Age-independent, harvest and catch-&-release conventional tagging dataset of Striped bass (Morone saxatilis) from 1991-2003 (Similar to the JiangADCR and JiangADH data sets). Based on Jiang Dissertation (Page 81 Model Fy,F'y,M91-99, M00-03).

Author(s)

Robert K. Bauer

References

Hoenig JM, Barrowman NJ, Pollock KH, Brooks EN, Hearn WS, Polacheck T. (1998) Models for tagging data that allow for incomplete mixing of newly tagged animals. Canadian Journal of Fisheries and Aquatic Sciences 55:1477-1483, doi: 10.1139/f97-258

Jiang H. (2005) Age-dependent tag return models for estimating fishing mortality, natural mortality, and selectivity. Ph.D. thesis, North Carolina State University, Raleigh

Jiang H, Pollock KH, Brownie C, Hightower JE, Hoenig JM, Hearn WS. (2007b) Age-dependent tag return models for estimating fishing mortality, natural mortality, and selectivity. Journal of Agricultural, Biological, and Environmental Statistics 12:177-194, doi: 10.1198/108571107X197382

Jiang H, Pollock KH, Brownie C, Hoenig JM, Latour RJ, Wells BK, Hightower JE. (2007) Tag return models allowing for harvest and catch and release: Evidence of environmental and management impacts on striped bass fishing and natural mortality rates. North American Journal of Fisheries Management 27:387-396, doi: 10.1577/M06-089.1

See Also

run.IRATE.example and remake.dat to change the model configuration of an existing IRATE-setup

Examples

```
IRATE.examples()
```

IRATER   *IRATER - A R Interface for the Instantaneous RATEs (IRATE) Model*

Description

IRATE is a software tool, developed by Dr. Gary Nelson, to analyze band recovery (conventional tagging) data. It can estimate fishing and natural mortality rates based on likewise estimated exploitation and survival rates of (continuous) fisheries. IRATE can handle both age-independent and age-dependent instantaneous rates models (Hoenig et al., 1998; Jiang et al., 2007) to be fitted to multi-year fish tag return data. IRATE allows model development with either age-dependent harvest-only or harvest and catch-release tag returns or similar age independent models. It also allows estimation of harvest reporting rates, catch and release reporting rates, and tag retention of harvested and/or released fish. However, not all parameters in the model can be estimated simultaneously with tag data alone. Some parameters must be fixed and assumed known (usually reporting rate and tag loss) to obtain good estimates of remaining parameters. Additionally, the model can account for non-mixing of the tagged fish in the first release year and adjust for harvest and M selectivity in the age-based models. The negative log likelihood is used as the objective function to obtain maximum likelihood estimates of parameters. Several model fit statistics are provided that can be used to select the best model formulation; these include the Akaike Information Criterion (AIC), c-hat (a measure of overdispersion) and standard residuals.
Source: http://nft.nefsc.noaa.gov/IRATE.html

IRATER is an interface to setup IRATE model runs and execute them from within R, using the ADMB engine of IRATE. It further provides functions to read and access the model results back into R.

Major functions:

- make.dat: writes and sets up an input data set (setup file) for IRATE model runs
- run.IRATE: compile and run IRATE model from within R
- read.par & read.rep: read run results

Getting Started
Check out the help files of the principle functions, listed above.

Author(s)

Robert K. Bauer

References

http://nft.nefsc.noaa.gov/IRATE.html

Jiang, H.H., K.H. Pollock, C. Brownie, J.M. Hoenig, R.J. Latour, B.K. Wells, and J.E. Hightower. 2007. Tag return models allowing for harvest and catch and release: evidence of environmental and management impacts on striped bass fishing and natural mortality rates. N. Amer. J. Fish. Man. 27: 387-396. doi: 10.1577/M06-089.1

Hoenig, J.M., N.J. Barrowman, W.S. Hearn, and K.H. Pollock. 1998. Multiyear tagging studies incorporating fishing effort data. Can. J. Fish. Aquat. Sci. 55: 1466-1476. doi: 10.1139/f97-256

Brownie, C., D.R. Anderson, K.P. Burnham, and D.S. Robson. 1985. Statistical inference from band recovery data: a handbook. Resource publication 156 (2nd edition), US Fish and Wildlife Service. 305 p.

---

make.dat	*writes and sets up an input data set (setup file) for IRATE model runs*

Description

writes and sets up an input data set (setup file) for IRATE model runs. Obligatory arguments are: `setup.file.name`, `releases` and `recapture`

Usage

```
make.dat(setup.file.name, releases, recaptures, age.dependent=F,only.harvested=F,HM
         ,eM=T,pM,sM,Mb # natural mortality settings
         ,eF=T,pF,sF,Fb # fishing mortality settings
         ,eT=T,pT,sT,Tb # tagging mortality settings
         ,combine.Hrr_and_Rrr=F # combine Harvest & Release reporting rates
         ,eHrr=F,pHrr,sHrr,Hrrb # Harvest reporting rates
         ,eRrr=F,pRrr,sRrr,Rrrb # Release reporting rates
         ,combine.HRS_and_RRS=F # combine Harvest & Release Retention Survival
         ,eHRS=F,pHRS,sHRS,HRSb #  Harvest Retention Survival
         ,eRRS=F,pRRS,sRRS,RRSb #  Release Retention Survival
         ,incomplete.mix=F,pNonMix,sNonMix,NonMixb #  Non-Mixing
         ,combine.HS_and_RS=T # combine Harvest & Release Selectivity
         ,eHS=T,sHS,HSb # age-depdendent Harvest Selectivity
         ,eRS=F,sRS,RSb # age-depdendent Release Selectivity
         ,eMS=T,aMS,sMS # age-depdendent M Selectivity
         ,use.L.const=T # Likelihood Constant

)
```

Arguments

`setup.file.name`

        character string defining the name of the IRATE-model setup file (`.dat`) to be created.

releases	raw `data.frame` with release dates (`rel.date` in `"Date"`-format) as well as the age (numeric) of released individuals (if age dependent).
recaptures	raw `data.frame` with release & recapture dates (`rel.date`, `rec.date` in `"Date"`-format) as well as the age (numeric) of recaptured individuals and a vector `harvested` to indicate whether recaptured individuals were harvested or released (if age dependent).
age.dependent	whether setup file should be created for an age dependent model (default = TRUE).
only.harvested	optional, whether actual releases of recaptures should be treated as harvested or a release-recapture model should be defined. (default = TRUE).
HM	hooking-mortality value for each recapture year (default = 0.09), only required for release-recapture model, otherwise being ignored.
eM,pM,sM,Mb	eM: whether to estimate natural mortality (default = TRUE), if set TRUE: pM: beginning years of different natural mortality periods, sM: starting values of different natural mortality periods, Mb: natural mortality lower and upper boundary values for model runs.
eF,pF,sF,Fb	eF: whether to estimate fishing mortality (default = TRUE), if set TRUE: pF: beginning years of different fishing mortality periods, sF: starting values of different fishing mortality periods, Fb: fishing mortality lower and upper boundary values for model runs.
eT,pT,sT,Tb	eT: whether to estimate tagging mortality (default = TRUE), if set TRUE: pT: beginning years of different tagging mortality periods, sT: starting values of different tagging mortality periods, Tb: tagging mortality lower and upper boundary values for model runs.
combine.Hrr_and_Rrr	
	whether to combine Harvest & Release reporting rates (default = FALSE)
eHrr,pHrr,sHrr,Hrrb	
	eHrr: whether to estimate harvest reporting rates (default = FALSE), if set TRUE: pHrr: beginning years of different harvest reporting periods, sHrr: starting values of different harvest reporting periods, Hrrb: harvest reporting lower and upper boundary values for model runs.
eRrr,pRrr,sRrr,Rrrb	
	eRrr: whether to estimate release reporting rates (default = FALSE), if set TRUE: pRrr: beginning years of different release reporting periods, sRrr: starting values of different release reporting periods, Rrrb: release reporting lower and upper boundary values for model runs.
combine.HRS_and_RRS	
	whether to combine Harvest & Release Retention Survival (default = FALSE)
eHRS,pHRS,sHRS,HRSb	
	eHRS: whether to estimate harvest-retention survival rates (default = FALSE), if set TRUE: pHRS: beginning years of harvest-retention survival periods, sHRS: starting values of different-harvest retention survival periods, HRSb: lower and upper boundary values of harvest-retention survival rates for model runs.

eRRS,pRRS,sRRS,RRSb

> eRRS: whether to estimate release-retention survival rates (default = FALSE), if set TRUE:
>
> pRRS: beginning years of release-retention survival periods,
>
> sRRS: starting values of different release-retention survival periods,
>
> RRSb: lower and upper boundary values of release-retention survival rates for model runs.

incomplete.mix,pNonMix,sNonMix,NonMixb

> incomplete.mix: whether to assume incomplete mixing of tagged and wild population (default = FALSE), if set TRUE:
>
> pNonMix: beginning years of incomplete mixing periods,
>
> sNonMix: starting values of incomplete mixing periods,
>
> NonMixb: of release-retention survival rates lower and upper boundary values for model runs.

combine.HS_and_RS

> whether to combine Harvest & Release Selectivity (default = TRUE)

eHS,sHS,HSb   eHS: whether to estimate age-dependent harvest selectivity rates (default = FALSE), if set TRUE:

> sHS: starting values of age-depdendent harvest selectivity periods,
>
> HSb: lower and upper boundary values of age-depdendent harvest selectivity rates for model runs.

eRS,sRS,RSb   eRS: whether to estimate age-dependent release selectivity rates (default = FALSE), if set TRUE:

> sRS: starting values of age-depdendent release selectivity periods,
>
> RSb: lower and upper boundary values of age-depdendent release selectivity rates for model runs.

eMS,aMS,sMS   eMS: whether to estimate age-dependent natural mortality rates (default = TRUE; only considered if age.dependent is set TRUE), if set TRUE:

> aMS: beginning age classes of age-dependent natural mortality rates (age classes are counted from 1 onwards irrespective of their age-values),
>
> sMS: starting values of (grouped) age-depdendent natural mortality rates,

use.L.const   whether to set likelihood function constant (default = TRUE)

Author(s)

Robert K. Bauer

References

<http://nft.nefsc.noaa.gov/IRATE.html>

See Also

To check model setup files in detail see: read.dat.
To alter existent model setup files see: remake.dat.
To read model run results see: read.rep and read.par.
For preinstalled example runs and setup data see: run.IRATE.example.
To delete run files see: clean.IRATE.

For more information on admb model compiliation and run prodcudres see: compile_admb and run_admb from the R2admb-package

---

read.dat   *loads existing IRATE-model setup (incl. tag release & recapture data)*

Description

loads an existing IRATE-model setup (incl. tag release and recapture data) as a list

Usage

```
read.dat(setup.file.name,skip=0)
```

Arguments

`setup.file.name`

character string defining the name of the existing IRATE-model input file (*.dat*) to be loaded.

`skip`   Number of lines to skip in setup file (default = 0).

Author(s)

Robert K. Bauer

See Also

IRATE.examples, make.dat, remake.dat

Examples

```
### read a preinstalled IRATE setup example:
examples.folder <- system.file("IRATE.examples", package = "IRATER")
setwd(examples.folder)
fnames <- Sys.glob('*.dat')
read.dat(fnames[1])
```

---

read.par   *Read in parameters from an IRATE run*

Description

Reads in par-file from an IRATE run

Usage

```
read.par(fn)
```

Arguments

fn   Character string indicating the name of the .par-file to be read.

Value

a list containing the parameters from an IRATE run:

napram number of parameters estimated

ofv objective function value

max_grad_comp maximum gradient component

e_F fishing mortality estimates

e_M natural mortality estimates

e_FA tag mortality

LRR1, LRR2 reporting rate

PHI1R, PHI2R phi

e_nmixh, e_nmixr non-mixing

sel1, sel2 catch-at-age selectivity

Author(s)

Robert K. Bauer

See Also

read.rep

---

read.rep   *Read report-file from an IRATE run*

Description

Reads in the report-file from an IRATE run

Usage

```
read.rep(fn, skip=0, show.sm=F, short=F)
read.short.rep(fn, skip=0, show.sm=F, short=T)
```

Arguments

fn	Character string indicating the name of the .rep-file to be read.
skip	Number of lines to skip in report file (default = 0).
show.sm	Show model report summary while reading the file.
short	read short report or long report.

Value

a list containing the model statistics and parameter estimates from an IRATE run:

for read.short.rep or read.rep(short=T):

Log_L, AIC, AICc, Effective_Sample_Size, Unpooled_Chi_square, Upooled_df, Unpooled_c_hat, Pooled_Chi_square, Pooled_df, Pooled_c_hat

othervise the full report file will be returned, containing the following

rep_info report file information

Log_L Log-Likelihood

K Number of parameters

AIC Akaike information criterion

AICc AIC corrected for finite sample sizes

Effective_Sample_Size

Unpooled_Chi_square

Upooled_df

Unpooled_c_hat

Pooled_Chi_square

Pooled_df

Pooled_c_hat

Harvest_Residuals, Harvest_Residuals_of_Age_Class_*

Release_Residuals, Release_Residuals_of_Age_Class_*

Not_seen_Residuals, Not_seen_Residuals_of_Age_Class_*

Obs_Recoveries_of_harvest_fish

Obs_Recoveries_of_release_fish

Total_Released

Total_Recovered_Tags

s_matrix matrix of survival rates of the tags (Sj,tags)

S_prob_matrix matrix of survival rates used in the calculation of expected cell probabilities

Exploitation_Rate_of_harvested_fish matrix of estimated exploitation rates for harvested
     fish (Uj,tags(kept))

Exploitation_Rate_of_released_fish matrix of estimated exploitation rates for released fish
     (Uj,tags(rels))

Expected_Probability_of_harvested_fish matrix of cell expected probabilities for harvested
     tag returns (Pij)

Expected_Probability_of_released_fish matrix of cell expected probabilities for released tag
     returns (P'ij)

Not_Seen_Probability matrix of cell expected probabilities for not seen tag returns (P-sum(Pij+P'ij))

Expected_Number_of_harvested_fish matrix of expected number of harvested tag returns (E(Rij)

Expected_Number_of_released_fish matrix of expected number of released tag returns (E(R'ij))

Expected_Number_of_not_seen matrix of expected number of not seen tag returns (N-sum(Rij+R'ij))

Cell_Likelihoods_of_harvested_fish individual cell likelihood values for harvested tag re-
     turns

Cell_Likelihoods_of_released_fish individual cell likelihood values for released tag returns

Cell_Likelihoods_of_not_seen individual cell likelihood values for not seen tag returns

Unpooled_Chi_squares_of_Harvested_Fish individual unpooled cell chi-square values for har-
     vested tag returns

Unpooled_Chi_squares_of_Released_Fish individual unpooled cell chi-square values for re-
     leased tag returns

Chi_squares_of_Not_Seen individual unpooled cell chi-square values for unseen tag returns

Pooled_Cells_of_Harvested_Fish matrix of pooled, observed harvested tag returns

Pooled_Cells_of_Released_Fish matrix of pooled, observed harvested tag returns

Pooled_Expected_Cells_of_Harvested_Fish matrix of pooled expected number of harvested
     tag returns

Pooled_Expected_Cells_of_Released_Fish matrix of pooled expected number of released tag
     returns

Pooled_Chi_squares_of_Harvested_Fish matrix of cell chi-square values for harvested tag re-
     turns

Pooled_Chi_squares_of_Released_Fish matrix of pooled cell chi-square values for released tag
     returns

Pearson_Residuals_for_harvested_fish matrix of Pearson residuals for harvested tag returns

Pearson_Residuals_for_released_fish matrix of Pearson residuals for released tag returns

Pearson_Residuals_for_not_seen matrix of Pearson residuals for not seen tag returns

Author(s)

Robert K. Bauer

See Also

read.par

---

remake.dat   *load and reparameterize an existing IRATE-model setup file*

Description

loads an existing IRATE-model setup file (incl. tag release and recapture data) and allows to reparameterize and save it as a new setup file for IRATE model runs. Note that reparameterization will depend on the tagging dataset inlcuded in the old setup. (e.g. a "harvest and catch-&-release"-serup is not feasable with a former harvest-only data set.)

Usage

```
remake.dat(old.setup.name, new.setup.name, age.dependent=T,
          only.harvested=T,...)
```

Arguments

old.setup.name	character string defining the name of the existing IRATE-model setup file (*.dat*) to be loaded.
new.setup.name	character string defining the name of the IRATE-model setup file (*.dat*) to be created.
age.dependent	whether setup file should be created for an age dependent model (default = *TRUE*).
only.harvested	optional, whether actual releases of recaptures should be treated as harvested or a release-recapture model should be defined. (default = *TRUE*).
...	Additional arguments, see Details

Details

HM
hooking-mortality value for each recapture year (default = 0.09), only required for release-recapture model, otherwise being ignored.

eM,pM,sM,Mb
eM: whether to estimate natural mortality (default = *TRUE*),
if set *TRUE*:
pM: beginning years of different natural mortality periods,
sM: starting values of different natural mortality periods,
Mb: natural mortality lower and upper boundary values for model runs.

`eF,pF,sF,Fb`

eF: whether to estimate fishing mortality (default = *TRUE*),

if set *TRUE*:

pF: beginning years of different fishing mortality periods,

sF: starting values of different fishing mortality periods,

Fb: fishing mortality lower and upper boundary values for model runs.

`eT=T,pT,sT,Tb`

eT: whether to estimate tagging mortality (default = *TRUE*),

if set *TRUE*:

pT: beginning years of different tagging mortality periods,

sT: starting values of different tagging mortality periods,

Tb: tagging mortality lower and upper boundary values for model runs.

`combine.Hrr_and_Rrr` whether to combine Harvest & Release reporting rates (default = *FALSE*)

`eHrr=F,pHrr,sHrr,Hrrb`

eHrr: whether to estimate harvest reporting rates (default = *FALSE*),

if set *TRUE*:

pHrr: beginning years of different harvest reporting periods,

sHrr: starting values of different harvest reporting periods,

Hrrb: harvest reporting lower and upper boundary values for model runs.

`eRrr=F,pRrr,sRrr,Rrrb`

eRrr: whether to estimate release reporting rates (default = *FALSE*),

if set *TRUE*:

pRrr: beginning years of different release reporting periods,

sRrr: starting values of different release reporting periods,

Rrrb: release reporting lower and upper boundary values for model runs.

`combine.HRS_and_RRS` whether to combine Harvest & Release Retention Survival (default = *FALSE*)

`eHRS=F,pHRS,sHRS,HRSb`

eHRS: whether to estimate harvest-retention survival rates (default = *FALSE*),

if set *TRUE*:

pHRS: beginning years of harvest-retention survival periods,

sHRS: starting values of different-harvest retention survival periods,

HRSb: lower and upper boundary values of harvest-retention survival rates for model runs.

`eRRS=F,pRRS,sRRS,RRSb`

eRRS: whether to estimate release-retention survival rates (default = *FALSE*),

if set *TRUE*:

pRRS: beginning years of release-retention survival periods,

sRRS: starting values of different release-retention survival periods,

RRSb: lower and upper boundary values of release-retention survival rates for model runs.

`incomplete.mix=F,pNonMix,sNonMix,NonMixb`

incomplete.mix: whether to assume incomplete mixing of tagged and wild population (default = *FALSE*),

if set *TRUE*:

pNonMix: beginning years of incomplete mixing periods,

sNonMix: starting values of incomplete mixing periods,

NonMixb: of release-retention survival rates lower and upper boundary values for model runs.

`combine.HS_and_RS` whether to combine Harvest & Release Selectivity (default = *TRUE*)

```
eHS=F,pHS,sHS,HSb
```
eHS: whether to estimate age-depdendent harvest selectivity rates (default = *FALSE*),
if set *TRUE*:
pHS: beginning years of age-depdendent harvest selectivity periods,
sHS: starting values of age-depdendent harvest selectivity periods,
HSb: lower and upper boundary values of age-depdendent harvest selectivity rates for model runs.

```
eRS=F,pRS,sRS,RSb
```
eRS: whether to estimate age-depdendent release selectivity rates (default = *FALSE*),
if set *TRUE*:
pRS: beginning years of age-depdendent release selectivity periods,
sRS: starting values of age-depdendent release selectivity periods,
RSb: lower and upper boundary values of age-depdendent release selectivity rates for model runs.

```
eMS=F,aMS,sMS
```
eMS: whether to estimate age-depdendent natural mortality rates (default = *TRUE*; only considered
if *age.dependent* is set *TRUE*),
if set *TRUE*:
aMS: beginning age classes of age-dependent natural mortality rates (age classes are counted from
1 onwards irrespective of their age-values),
sMS: starting values of (grouped) age-depdendent natural mortality rates,

use.L.const whether to set likelihood function constant (default = *TRUE*)

Author(s)

Robert K. Bauer

See Also

IRATE.examples, make.dat, read.dat

Examples

```
# old.setup <- "JiangADCR" # select old IRATE setup to reparameterize
# old.setup.path <- system.file(paste0("IRATE.examples/",old.setup,".dat"), package = "IRATER")

# new.setup <- new.setup.path <- "JiangAICR_manual"
# system(paste("mkdir -p ",new.setup.path)) # create run folder for new setup
# system(paste("cp",old.setup.path, new.setup.path)) # copy old setup in new run folder

# setwd(new.setup.path)
# remake.dat(old.setup,new.setup,age.dependent = F) # reparameterize old setup
# run.IRATE(setup.new) # run new setup
```

---

run.IRATE   *compile and run NOAA-IRATE models from within R*

Description

compile and run NOAA-IRATE models from within R

Usage

```
run.IRATE(setup.file, safe=F, re=F, verbose=T,
          admb_errors=c("stop","warn","ignore"),
          mcmc=F, mcmc.opts=mcmc.control(),
          profile=F, extra.args="")
```

Arguments

setup.file	Character string defining IRATE setup to be run. Check IRATE.examples for preinstalled IRATE setup examples and make.dat to create new or remake.dat to alter existing setup files.
safe	(logical) Compile in safe mode? (default = *FALSE*)
re	(logical) Compile in random effects (ADMB-RE) mode? (default = *FALSE*)
verbose	(logical) Verbose output? (default = *TRUE*)
admb_errors	(character) how to handle compilation/linking errors?
mcmc	(logical) run post-hoc MCMC? (default = *FALSE*)
mcmc.opts	options for MCMC run (see mcmc.control)
profile	(logical) Run likelihood profiles? (default = *FALSE*)
extra.args	(character) extra (ADMB-) arguments for IRATE run

Author(s)

Robert K. Bauer

References

<http://nft.nefsc.noaa.gov/IRATE.html>

See Also

To check model setup files see: read.dat and make.dat.
To read run results see: read.rep and read.par.
For preinstalled example runs see: run.IRATE.example.
To delete run files see: clean.IRATE.
For more information on admb model compiliation and run prodcudres see: compile_admb and run_admb from the R2admb-package

Examples

```
example.setup <- IRATE.examples()
new.setup <- new.setup.path <- example.setup[1] # select old IRATE setup to reparameterize
print(new.setup) # print setup name to be run
old.setup.path <- system.file(paste0("IRATE.examples/",new.setup,".dat"), package = "IRATER")

system(paste("mkdir -p ",new.setup.path)) # create run folder for new setup
system(paste("cp",old.setup.path, new.setup.path)) # copy old setup in new run folder
setwd(new.setup.path)

run.IRATE(new.setup) # compile and run setup
```

---

run.IRATE.example   *compile and run preinstalled example IRATE-model setups*

Description

compile and run preinstalled example IRATE-model setups

Usage

```
run.IRATE.example(setup.name, run.folder, ...)
```

Arguments

setup.name	Character string defining preinstalled IRATE example setups/datasets to be run. If missing or not valid, the user will be asked to choose among valid examples. Check IRATE.examples() for valid, preinstalled IRATE examples.
run.folder	Character string defining the folder in which the selected example shall be run. If missing a folder, named after the example setup file, will be created in the current working directory.
...	Further arguments to be passed to run.IRATE

Author(s)

Robert K. Bauer

See Also

To check model setup files see: read.dat and make.dat.
For run results see: read.rep and read.par.
To delete run files see: clean.IRATE.
For more information on admb model compiliation and run prodcudres see: compile_admb and run_admb from the R2admb-package

Examples

```
### setup example not specified, user will be prompted to select among exisisting example setups:
# run.IRATE.example()

### example name specified
# run.IRATE.example("JiangADCR")
```

Index