

Package ‘retmort’

February 10, 2025

Title Estimate User-Based Tagging Mortality and Tag Loss in Mark-Recapture Studies

Version 1.0.0

Description We provide several avenues to predict and account for user-based mortality and tag loss during mark-recapture studies. When planning a study on a target species, the `retentionmort_generation()` function can be used to produce multiple synthetic mark-recapture datasets to anticipate the error associated with a planned field study to guide method development to reduce error. Similarly, if field data was already collected, the `retentionmort()` function can be used to predict the error from already generated data to adjust for user-based mortality and tag loss. The `test_dataset_retentionmort()` function will provide an example dataset of how data should be inputted into the function to run properly. Lastly, the `retentionmort_figure()` function can be used on any dataset generated from either model function to produce an 'rmarkdown' printout of preliminary analysis associated with the model, including summary statistics and figures. Methods and results pertaining to the formation of this package can be found in McCutcheon et al. (in review, ``Predicting tagging-related mortality and tag loss during mark-recapture studies").

License MIT + file LICENSE

Imports dplyr, ggplot2, gridExtra, patchwork, readr, rmarkdown

Encoding UTF-8

RoxygenNote 7.3.2

NeedsCompilation no

Author Brendan Campbell [aut, cre] (<<https://orcid.org/0000-0001-5604-7658>>),
Jasper McCutcheon [aut] (<<https://orcid.org/0009-0004-4138-532X>>),
Raleigh Hudock [aut] (<<https://orcid.org/0009-0009-2769-2880>>),
Noah Motz [aut] (<<https://orcid.org/0009-0002-2480-9712>>),
Madison Windsor [aut] (<<https://orcid.org/0009-0000-4517-0677>>),
Aaron Carlisle [aut] (<<https://orcid.org/0000-0003-0796-6564>>),
Edward Hale [aut] (<<https://orcid.org/0000-0002-7176-6420>>)

Maintainer Brendan Campbell <bpc@udel.edu>

Repository CRAN

Date/Publication 2025-02-10 09:40:02 UTC

Contents

retentionmort	2
retentionmort_figure	7
retentionmort_generation	8
test_dataset_retentionmort	13

Index	15
--------------	-----------

retentionmort	<i>Estimating User-Based Tagging Mortality and Tag Shedding in Field Mark-Recapture Studies</i>
---------------	---

Description

This model estimates the percent loss in tagged animals at large for field-based recapture studies based on a linear decrease in survival and tag retention (including lost tags and missidentified tags) for five weeks per tagging cohort based on laboratory retention/survival studies. The `retentionmort()` function can be used following a recapture field study to estimate user-based tag loss in animals at large. The model is changed by linear regression coefficients of weekly tag loss rate, weekly mortality rate, and their respective intercepts. The coefficients used can be selected from the currently included list using the `err` input or be customized. This function is also capable of working with a cofactor with two conditions (e.g. `class1` individuals and large individuals) to improve resolution for more specified studies.

Usage

```
retentionmort(
  nT,
  n_c1 = nT,
  TaL,
  c,
  R,
  err = 2,
  m_mort_c1 = NA,
  b_mort_c1 = NA,
  m_ret_c1 = NA,
  b_ret_c1 = NA,
  m_mort_c2 = NA,
  b_mort_c2 = NA,
  m_ret_c2 = NA,
  b_ret_c2 = NA
)
```

Arguments

`nT` A vector of the number of tagged individuals for each tagging effort.

n_c1	(optional) A vector of the number of tagged individuals in one of two categorical variables. If this is not being used then n_c1 will be equal to nT.
TaL	A vector of the cumulative number of tagged individuals following each effort.
c	One value for the total number of tagging efforts. This value must be greater than or equal to 6.
R	A vector of the number of recaptured individuals per effort.
err	A value (between 1 and 26) that represents the weekly mortality rate and weekly tag loss rate from a preloaded case study listed in the metadata. Alternatively, model coefficients can be manually included using a combination of the preceding parameters. While the preloaded data are based on weekly time stamps, customized model coefficients can reflect any time period specified and the projection will predict loss at 5 times the time interval.

1	= Large (> 61mm TL) and class1 (< 61mm TL) Mummichogs tagged with VIE in caudal peduncle (avg + 95% CI) - McCutcheon et al. in prep
2	= Large (> 61mm TL) and class1 (< 61mm TL) Mummichogs tagged with VIE in caudal peduncle (avg) - McCutcheon et al. in prep
3	= Large (> 61mm TL) and class1 (< 61mm TL) Mummichogs tagged with VIE in caudal peduncle (avg - 95% CI) - McCutcheon et al. in prep
4	= American Eel elvers (80 - 149 mm TL) tagged with 2 VIE tags in anterior, posterior, central of body - Eissenhauer et al. 2024 https://doi.org/10.1002/nafm.11016
5	= Mummichogs (45 - 82 mm TL) tagged with 8mm PIT tags in abdominal cavity - Kimball & Mace 2020 https://doi.org/10.1007/s12237-019-00657-4
6	= Mummichogs (45 - 82 mm TL) tagged with 12mm PIT tags in abdominal cavity - Kimball & Mace 2020 https://doi.org/10.1007/s12237-019-00657-4
7	= Pinfish (45 - 82 mm TL) tagged with 8mm or 12mm PIT tags in abdominal cavity - Kimball & Mace 2020 https://doi.org/10.1007/s12237-019-00657-4
8	= Cichlids (29 - 59 mm TL) tagged with VIE in various locations on body - Jungwirth et al. 2019 https://doi.org/10.1007/s00265-019-2659-y
9	= River Shiners (36 - 49 mm TL) tagged with VIE using anesthesia in various locations - Moore & Brewer 2021 https://doi.org/10.1002/nafm.10607
10	= River Shiners (50 - 56 mm TL) tagged with 8 mm PIT using anesthesia in various locations - Moore & Brewer 2021 https://doi.org/10.1002/nafm.10607
11	= River Shiners (40 - 51 mm TL) tagged with VIE using no anesthesia in various locations - Moore & Brewer 2021 https://doi.org/10.1002/nafm.10607
12	= River Shiners (50 - 55 mm TL) tagged with 8 mm PIT using no anesthesia in various locations - Moore & Brewer 2021 https://doi.org/10.1002/nafm.10607
13	= Delta Smelt (> 70 mm FL) tagged with injected acoustic tag -

- Wilder et al. 2016
<https://doi.org/10.1080/02755947.2016.1198287>
- 14 = Delta Smelt (> 70 mm FL) surgically tagged with acoustic tag -
 Wilder et al. 2016
<https://doi.org/10.1080/02755947.2016.1198287>
- 15 = Rohu Carp tagged with floy tags under dorsal fin - Hadiuzzaman
 et al. 2015
https://www.researchgate.net/publication/289460932_Feasibility_study_of_using_1
- 16 = Silver Carp tagged with floy tags under dorsal fin - Hadiuzzaman
 et al. 2015
https://www.researchgate.net/publication/289460932_Feasibility_study_of_using_1
- 17 = Black Bullhead (mean TL = 153.3 mm) tagged with VIE near dorsal
 fin - Schumann et al. 2013
<https://benthamopen.com/contents/pdf/TOFISHSJ/TOFISHSJ-6-41.pdf>
- 18 = Bluegill (mean TL = 75.8 mm) tagged with VIE near dorsal fin -
 Schumann et al. 2013
<https://benthamopen.com/contents/pdf/TOFISHSJ/TOFISHSJ-6-41.pdf>
- 19 = Channel Catfish (mean TL = 127.9 mm) tagged with VIE near dorsal
 fin - Schumann et al. 2013
<https://benthamopen.com/contents/pdf/TOFISHSJ/TOFISHSJ-6-41.pdf>
- 20 = Juvenile Burbot (88 - 144 mm TL) tagged with coded wire tag on
 snout, periocular region, nape, pectoral fin base, dorsal fin
 base, and anal fin base - Ashton et al. 2013
<https://doi.org/10.1080/02755947.2014.882458>
- 21 = Delta Smelt adults (45 - 77 mm FL) and juveniles (20 - 40 mm FL)
 tagged with calcein markers - Castillo et al. 2014
<https://doi.org/10.1080/02755947.2013.839970>
- 22 = Juvenile Seabass (mean 173 g) tagged with dummy acoustic
 transmitters in external or intraperitoneal cavity -
 Begout Anras et al. 2003
[https://doi.org/10.1016/S1054-3139\(03\)00135-8](https://doi.org/10.1016/S1054-3139(03)00135-8)
- 23 = Juvenile American Eels (113 - 175 mm TL) tagged with
 micro-acoustic transmitter in body cavity - Mueller et al. 2017
<https://doi.org/10.1016/j.fishres.2017.06.017>
- 24 = Juvenile European Eels (7 - 25 g) tagged with 12mm PIT tags -
 Jepsen et al. 2022
<https://doi.org/10.1111/jfb.15183>
- 25 = Adult Atlantic Croaker (147 - 380 mm TL) tagged with VIE tags in
 caudal fin - Torre et al. 2017
<https://doi.org/10.1080/00028487.2017.1360391>
- 26 = Adult Spot (65 - 222 mm FL) tagged with VIE tags in caudal fin -
 Torre et al. 2017
<https://doi.org/10.1080/00028487.2017.1360391>

m_mort_c1

A value that represents the slope of the mortality rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, b_mort_c1, m_ret_c1, and b_ret_c1

	need to be used. The use of these coefficients will override the err term.
b_mort_c1	A value that represents the intercept of the mortality rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c1, m_ret_c1, and b_ret_c1 need to be used. The use of these coefficients will override the err term.
m_ret_c1	A value that represents the slope of the tag loss (represented as tag loss and misidentification) rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c1, b_mort_c1, and b_ret_c1 need to be used. The use of these coefficients will override the err term.
b_ret_c1	A value that represents the intercept of the tag loss (represented as tag loss and misidentification) rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c1, b_mort_c1, and m_ret_c1 need to be used. The use of these coefficients will override the err term.
m_mort_c2	A value that represents the slope of the mortality rate for the class2 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, b_mort_c2, m_ret_c2, and b_ret_c2 need to be used. The use of these coefficients will override the err term.
b_mort_c2	A value that represents the intercept of the mortality rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c2, m_ret_c2, and b_ret_c2 need to be used. The use of these coefficients will override the err term.
m_ret_c2	A value that represents the slope of the tag loss (represented as tag loss and misidentification) rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c2, b_mort_c2, and b_ret_c2 need to be used. The use of these coefficients will override the err term.
b_ret_c2	A value that represents the intercept of the tag loss (represented as tag loss and misidentification) rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c2, b_mort_c2, and m_ret_c2 need to be used. The use of these coefficients will override the err term.

Value

This returns a dataframe `datacomp` that contains summary information from each mark-recapture effort, several parameters used in the calculation of adjusted recaptures, and basic error values between the expected and observed number of recaptures. The `datacomp` dataframe can be used in the `retentionmort_figure()` function to generate some preliminary figures that can be used to assess model performance and factors that influence the error between expected and observed recaptures.

Values that will be returned include:

```

week =      The week in the study
nT =       The number of tagged individuals per tagging effort
n_c1 =     The number of tagged individuals in class1 per tagging
           effort
TaL =      The cumulative number of tagged individuals at large
TASum =    The weekly sum of adjusted number of tags at large
TDF =      Tag depreciation factor
YSs =      Resultant survival rate of class1
YSl =      Resultant survival rate of class2
YMs =      Resultant tag loss rate of class1
Yml =      Resultant tag loss rate of class2
TaLs =     The cumulative number of class1 individuals tagged at
           large
TaLl =     The cumulative number of class2 individuals tagged at
           large
R =        The number of recaptured individuals per effort
Rpercent = The proportion of recaptured individuals
RA =       The adjusted number of recaptured individuals
PSE =      The percent standard error between observed and
           estimated recaptured individuals

```

Examples

```

#To formulate a dataset for each example
ret_env <- new.env()
data<- test_dataset_retentionmort()
list2env(data, envir = ret_env)

#Using a preloaded set of model parameters
datacomp = retentionmort(n_c1=ret_env$n_c1, nT=ret_env$nT, TaL=ret_env$TaL,
                        c=ret_env$c, R=ret_env$R, err=ret_env$err
                        )

#Using custom model parameters for one class type
datacomp = retentionmort(n_c1=ret_env$n_c1, nT=ret_env$nT, TaL=ret_env$TaL,
                        c=ret_env$c, R=ret_env$R, m_mort_c1=-0.0625,
                        b_mort_c1=1.06, m_ret_c1=-0.113, b_ret_c1=1.05
                        )
#or
datacomp = retentionmort(n_c1=ret_env$n_c1, nT=ret_env$nT, TaL=ret_env$TaL,
                        c=ret_env$c, R=ret_env$R, m_mort_c2=-0.0203,

```

```

        b_mort_c2=1.03, m_ret_c2=-0.0541, b_ret_c2=0.993
    )

#Using custom model parameters for two class types
datacomp = retentionmort(n_c1=ret_env$n_c1, nT=ret_env$nT, TaL=ret_env$TaL,
    c=ret_env$c, R=ret_env$R, m_mort_c1=-0.0625,
    b_mort_c1=1.06, m_ret_c1=-0.113, b_ret_c1=1.05,
    m_mort_c2=-0.0203, b_mort_c2=1.03, m_ret_c2=-0.0541,
    b_ret_c2=0.993
    )

```

retentionmort_figure *Generate a .html markdown file of preliminary figures pertaining from either the retentionmort() or retentionmort_generation() function.*

Description

By inputing the datacomp dataframe, this function will save a markdown file in the working directory named retentionmort.html that provides helpful information on the error associated with the number of recaptured individuals compared to the expected number provided by the model. Some figures will be less applicable for the field data application using the retentionmort() function due to a low sample size, specifically figures 5 and 6.

Usage

```
retentionmort_figure(datacomp)
```

Arguments

datacomp The file generated from either the retentionmort() or retentionmort_generation() functions.

Value

This function will return one markdown file named retentionmort.html in your current working directory listing some helpful information for analyzing model data generated from the retentionmort() or retentionmort_generation() functions

Examples

```

#Using retentionmort_generation() to produce multiple iterations of data to
#run the model through

datacomp = retentionmort_generation()
Rmark = file.path(tempdir(),retentionmort_figure(datacomp))
unlink("retentionmort.Rmd")
unlink("retentionmort.html")

```

```

#Creating a dataset with test_dataset_retentionmort() and running the
#retentionmort() function
  ret_env <- new.env()
  data<- test_dataset_retentionmort()
  list2env(data, envir = ret_env)
  datacomp = retentionmort(n_c1=ret_env$n_c1, nT=ret_env$nT,
                          TaL=ret_env$TaL, c=ret_env$c, R=ret_env$R, err=ret_env$err)
#Creating the markdown on datacomp
  Rmark = file.path(tempdir(),retentionmort_figure(datacomp))
  unlink("retentionmort.Rmd")
  unlink("retentionmort.html")

```

retentionmort_generation

Estimating User-Based Tagging Mortality and Tag Shedding Error Using Artificial Mark-Recapture Data

Description

This model estimates the percent loss in tagged animals at large for field-based recapture studies based on a linear decrease in survival and tag retention (including lost tags and missidentified tags) that gets projected for five weeks per tagging cohort based on laboratory retention/survival studies. This `retentionmort_generation()` differs from `retentionmort()` because it generates a mark-recapture dataset instead of relying on field data, making it possible to estimate the expected error associated with an upcoming field effort to provide insight on methods development. The model is changed by linear regression coefficients of weekly tag loss rate, weekly mortality rate, and their respective intercepts. The coefficients used can be selected from the currently included list using the `err` input or be customized. This function is also capable of working with a cofactor with two conditions (e.g. `class1` individuals and large individuals) to improve resolution for more specified studies.

Usage

```

retentionmort_generation(
  n = 100,
  min_weeks = 6,
  max_weeks = 100,
  max_tags = 500,
  prop_class1 = 0,
  max_recap = 0.5,
  err = 2,
  m_mort_c1 = NA,
  b_mort_c1 = NA,
  m_ret_c1 = NA,
  b_ret_c1 = NA,
  m_mort_c2 = NA,
  b_mort_c2 = NA,

```

```

    m_ret_c2 = NA,
    b_ret_c2 = NA
  )

```

Arguments

n	The number of iterations (i.e. generated mark-recapture datasets) the model will run through (default = 100).
min_weeks	The minimum number of efforts that each generated mark-recapture dataset will operate for (default = 6), must be at least 6.
max_weeks	The maximum number of efforts that each generated mark-recapture dataset will run for (default = 100).
max_tags	The maximum number of individuals that will be tagged per effort (default = 500).
prop_class1	(Optional) The estimated proportion of tagged individuals that belong to the first of two classifications (use 1 or 0 if none; default = 0).
max_recap	The maximum proportion of recaptured individuals per effort (default = 0.5).
err	A value (between 1 and 26) that represents the weekly mortality rate and weekly tag loss rate from a preloaded case study listed in the metadata (default = 2). Alternatively, model coefficients can be manually included using a combination of the preceding parameters. While the preloaded data are based on weekly time stamps, customized model coefficients can reflect any time period specified and the projection will predict loss at 5 times the time interval. <ul style="list-style-type: none"> 1 = Large (> 61mm TL) and class1 (< 61mm TL) Mummichogs tagged with VIE in caudal peduncle (avg + 95% CI) - McCutcheon et al. in prep 2 = Large (> 61mm TL) and class1 (< 61mm TL) Mummichogs tagged with VIE in caudal peduncle (avg) - McCutcheon et al. in prep 3 = Large (> 61mm TL) and class1 (< 61mm TL) Mummichogs tagged with VIE in caudal peduncle (avg - 95% CI) - McCutcheon et al. in prep 4 = American Eel elvers (80 - 149 mm TL) tagged with 2 VIE tags in anterior, posterior, central of body - Eissenhauer et al. 2024 https://doi.org/10.1002/nafm.11016 5 = Mummichogs (45 - 82 mm TL) tagged with 8mm PIT tags in abdominal cavity - Kimball & Mace 2020 https://doi.org/10.1007/s12237-019-00657-4 6 = Mummichogs (45 - 82 mm TL) tagged with 12mm PIT tags in abdominal cavity - Kimball & Mace 2020 https://doi.org/10.1007/s12237-019-00657-4 7 = Pinfish (45 - 82 mm TL) tagged with 8mm or 12mm PIT tags in abdominal cavity - Kimball & Mace 2020 https://doi.org/10.1007/s12237-019-00657-4 8 = Cichlids (29 - 59 mm TL) tagged with VIE in various locations on body - Jungwirth et al. 2019 https://doi.org/10.1007/s00265-019-2659-y 9 = River Shiners (36 - 49 mm TL) tagged with VIE using anesthesia in

- various locations - Moore & Brewer 2021
<https://doi.org/10.1002/nafm.10607>
- 10 = River Shiners (50 - 56 mm TL) tagged with 8 mm PIT using anesthesia in various locations - Moore & Brewer 2021
<https://doi.org/10.1002/nafm.10607>
- 11 = River Shiners (40 - 51 mm TL) tagged with VIE using no anesthesia in various locations - Moore & Brewer 2021
<https://doi.org/10.1002/nafm.10607>
- 12 = River Shiners (50 - 55 mm TL) tagged with 8 mm PIT using no anesthesia in various locations - Moore & Brewer 2021
<https://doi.org/10.1002/nafm.10607>
- 13 = Delta Smelt (> 70 mm FL) tagged with injected acoustic tag - Wilder et al. 2016
<https://doi.org/10.1080/02755947.2016.1198287>
- 14 = Delta Smelt (> 70 mm FL) surgically tagged with acoustic tag - Wilder et al. 2016
<https://doi.org/10.1080/02755947.2016.1198287>
- 15 = Rohu Carp tagged with floy tags under dorsal fin - Hadiuzzaman et al. 2015
https://www.researchgate.net/publication/289460932_Feasibility_study_of_using_1
- 16 = Silver Carp tagged with floy tags under dorsal fin - Hadiuzzaman et al. 2015
https://www.researchgate.net/publication/289460932_Feasibility_study_of_using_1
- 17 = Black Bullhead (mean TL = 153.3 mm) tagged with VIE near dorsal fin - Schumann et al. 2013
<https://benthamopen.com/contents/pdf/TOFISHSJ/TOFISHSJ-6-41.pdf>
- 18 = Bluegill (mean TL = 75.8 mm) tagged with VIE near dorsal fin - Schumann et al. 2013
<https://benthamopen.com/contents/pdf/TOFISHSJ/TOFISHSJ-6-41.pdf>
- 19 = Channel Catfish (mean TL = 127.9 mm) tagged with VIE near dorsal fin - Schumann et al. 2013
<https://benthamopen.com/contents/pdf/TOFISHSJ/TOFISHSJ-6-41.pdf>
- 20 = Juvenile Burbot (88 - 144 mm TL) tagged with coded wire tag on snout, periocular region, nape, pectoral fin base, dorsal fin base, and anal fin base - Ashton et al. 2013
<https://doi.org/10.1080/02755947.2014.882458>
- 21 = Delta Smelt adults (45 - 77 mm FL) and juveniles (20 - 40 mm FL) tagged with calcein markers - Castillo et al. 2014
<https://doi.org/10.1080/02755947.2013.839970>
- 22 = Juvenile Seabass (mean 173 g) tagged with dummy acoustic transmitters in external or intraperitoneal cavity - Begout Anras et al. 2003
[https://doi.org/10.1016/S1054-3139\(03\)00135-8](https://doi.org/10.1016/S1054-3139(03)00135-8)
- 23 = Juvenile American Eels (113 - 175 mm TL) tagged with micro-acoustic transmitter in body cavity - Mueller et al. 2017
<https://doi.org/10.1016/j.fishres.2017.06.017>
- 24 = Juvenile European Eels (7 - 25 g) tagged with 12mm PIT tags - Jepsen et al. 2022

<https://doi.org/10.1111/jfb.15183>

25 = Adult Atlantic Croaker (147 - 380 mm TL) tagged with VIE tags in caudal fin - Torre et al. 2017

<https://doi.org/10.1080/00028487.2017.1360391>

26 = Adult Spot (65 - 222 mm FL) tagged with VIE tags in caudal fin - Torre et al. 2017

<https://doi.org/10.1080/00028487.2017.1360391>

m_mort_c1	A value that represents the slope of the mortality rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, b_mort_c1, m_ret_c1, and b_ret_c1 need to be used. The use of these coefficients will override the err term.
b_mort_c1	A value that represents the intercept of the mortality rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c1, m_ret_c1, and b_ret_c1 need to be used. The use of these coefficients will override the err term.
m_ret_c1	A value that represents the slope of the tag loss (represented as tag loss and misidentification) rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c1, b_mort_c1, and b_ret_c1 need to be used. The use of these coefficients will override the err term.
b_ret_c1	A value that represents the intercept of the tag loss (represented as tag loss and misidentification) rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c1, b_mort_c1, and m_ret_c1 need to be used. The use of these coefficients will override the err term.
m_mort_c2	A value that represents the slope of the mortality rate for the class2 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, b_mort_c2, m_ret_c2, and b_ret_c2 need to be used. The use of these coefficients will override the err term.
b_mort_c2	A value that represents the intercept of the mortality rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, m_mort_c2, m_ret_c2, and b_ret_c2 need to be used. The use of these coefficients will override the err term.
m_ret_c2	A value that represents the slope of the tag loss (represented as tag loss and misidentification) rate for the class1 individuals. While all the preloaded datasets

work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, `m_mort_c2`, `b_mort_c2`, and `b_ret_c2` need to be used. The use of these coefficients will override the `err` term.

`b_ret_c2` A value that represents the intercept of the tag loss (represented as tag loss and missidentification) rate for the class1 individuals. While all the preloaded datasets work in weekly time intervals, these can be customized to any time interval to match the sampling interval. The resulting model will then project mortality and tag loss for 5 times the time interval. If this value is added, then, at minimum, `m_mort_c2`, `b_mort_c2`, and `m_ret_c2` need to be used. The use of these coefficients will override the `err` term.

Value

This returns a dataframe `datacomp` that contains summary information from each mark-recapture effort, several parameters used in the calculation of adjusted recaptures, and basic error values between the expected and observed number of recaptures. The `datacomp` dataframe can be used in the `retentionmort_figure()` function to generate some preliminary figures that can be used to assess model performance and factors that influence the error between expected and observed recaptures.

Values that will be returned include:

```

week =      The week in the study
c =        The total number of efforts within the dataset
iteration = The iteration number of the dataset
nT =       The number of tagged individuals per tagging effort
nlc1 =     The number of tagged individuals in class1 per
           tagging effort
TaL =      The cumulative number of tagged individuals at large
TAsum =    The weekly sum of adjusted number of tags at large
TDF =      Tag depreciation factor
YSs =      Resultant survival rate of class1
YSl =      Resultant survival rate of class2
YMs =      Resultant tag loss rate of class1
YmL =      Resultant tag loss rate of class2
TaLs =     The cumulative number of class1 individuals tagged
           at large
TaLl =     The cumulative number of class2 individuals tagged
           at large
R =        The number of recaptured individuals per effort
Rpercent = The proportion of recaptured individuals
RA =       The adjusted number of recaptured individuals
PSE =      The percent standard error between observed and
           estimated recaptured individuals

```

Examples

```

#Using only default variables
datacomp = retentionmort_generation()

```

```

#Using custom model parameters for one class type
datacomp = retentionmort_generation(n = 100, max_weeks = 100, prop_class1 =
    0, max_recap = 0.5, err = NA, m_mort_c1=-0.0625,
    b_mort_c1=1.06, m_ret_c1=-0.113, b_ret_c1=1.05
)
#or
datacomp = retentionmort_generation(n = 100, max_weeks = 100, prop_class1 = 0,
    max_recap = 0.5, err = NA, m_mort_c2=-0.0203,
    b_mort_c2=1.03, m_ret_c2=-0.0541, b_ret_c2=0.993
)

#Using custom model parameters for two class types
datacomp = retentionmort_generation(n = 100, max_weeks = 100, prop_class1 =
    0, max_recap = 0.5, err = NA, m_mort_c1=-0.0625,
    b_mort_c1=1.06, m_ret_c1=-0.113, b_ret_c1=1.05,
    m_mort_c2=-0.0203, b_mort_c2=1.03, m_ret_c2=-0.0541,
    b_ret_c2=0.993
)
#or
datacomp = retentionmort_generation(n = 100, max_weeks = 100, prop_class1 =
    0, max_recap = 0.5, err = 2, m_mort_c1=-0.0625,
    b_mort_c1=1.06, m_ret_c1=-0.113, b_ret_c1=1.05,
    m_mort_c2=-0.0203, b_mort_c2=1.03, m_ret_c2=-0.0541,
    b_ret_c2=0.993
) #err gets overridden by customized coefficients

```

test_dataset_retentionmort

Generate a test dataset that can be input into the retentionmort() function

Description

Using the example code verbatim, this function will produce a series of outputs that can be directly input into the retentionmort() function.

Usage

```
test_dataset_retentionmort()
```

Value

c = One value for the total number of tagging efforts that is 6 or greater.

err = A value (between 1 and 26) that represents the weekly mortality rate and weekly tag loss rate from a preloaded case study listed in the metadata.

n_{c1} = A vector of the number of tagged individuals in one of two categorical variables.

nT = A vector of the number of tagged individuals for each tagging effort.

R = A vector of the number of recaptured individuals per effort.

TaL = A vector of the cumulative number of tagged individuals following each effort.

Examples

```
#Run this verbatim to produce a single mark-recapture dataset
ret_env <- new.env()
data<- test_dataset_retentionmort()
list2env(data, envir = ret_env)
```

Index

retentionmort, [2](#)

retentionmort_figure, [7](#)

retentionmort_generation, [8](#)

test_dataset_retentionmort, [13](#)