

Package ‘carcass’

August 22, 2025

Type Package

Title Estimation of the Number of Fatalities from Carcass Searches

Version 1.9

Date 2025-08-22

Depends R (>= 3.1.0), stats, lme4, survival

Imports expm, arm, MASS

Description The number of bird or bat fatalities from collisions with buildings, towers or wind energy turbines can be estimated based on carcass searches and experimentally assessed carcass persistence times and searcher efficiency. Functions for estimating the probability that a bird or bat that died is found by a searcher are provided. Further functions calculate the posterior distribution of the number of fatalities based on the number of carcasses found and the estimated detection probability.

License GPL-2

LazyLoad yes

NeedsCompilation no

Author Fraenzi Korner-Nievergelt [aut, cre],
Ivo Niermann [aut],
Oliver Behr [aut],
Matthew A. Etterson [aut],
Robert Brinkmann [aut],
Pius Korner [aut],
Barbara Hellriegel [aut],
Tobias Roth [aut],
Manuela M. P. Huso [aut],
Dan Dalthorp [aut]

Maintainer Fraenzi Korner-Nievergelt <fraenzi.korner@oikostat.ch>

Repository CRAN

Date/Publication 2025-08-22 08:00:02 UTC

Contents

carcass-package	2
---------------------------	---

batdist	3
CIetterson	4
estimateN	6
ettersonEq14	8
ettersonEq14v1	10
ettersonEq14v2	11
integrate.persistence	12
perickson	14
persistence	15
persistence.prob	16
phuso	18
pkorner	19
posteriorN	21
search.efficiency	22
searches	24
shapeparameter	25
Index	27

carcass-package	<i>Estimation of the Number of Fatalities from Carcass Searches</i>
-----------------	---

Description

The number of bird or bat fatalities from collisions with buildings, towers or wind energy turbines can be estimated based on carcass searches and experimentally assessed carcass persistence times and searcher efficiency. This package provides diverse functions for estimating the probability that a bird or bat that died is found by a searcher. It further provides a function to obtain a posterior distribution of the number of fatalities based on the number of carcasses found and the estimated detection probability. see worked example in Korner-Nievergelt et al 2011, and 2015.

Author(s)

Fraenzi Korner-Nievergelt, Ivo Niermann, Oliver Behr, Matthew A. Etterson, Robert Brinkmann, Pius Korner, Barbara Hellriegel, Tobias Roth, Manuela M. P. Huso, Dan Dalthorp

Maintainer: Fraenzi Korner-Nievergelt <fraenzi.korner@oikostat.ch>

The authors thank Helga Garcia and Joana Bernardino for reporting bugs.

References

Erickson WP, Jeffrey J, Kronner K, Bay K (2004) Stateline Wind Project Wildlife. Monitoring Final Report, July 2001 - December 2003. - Technical report peer-reviewed by and submitted to FPL Energy, the Oregon Energy Facility Siting Council, and the Stateline Technical Advisory Committee, Oregon, USA, 105 pp

Etterson, M.A. (2013) Hidden Markov models for estimating animal mortality from antropogenic hazards. *Ecological Applications*, 23, 1915-1925.

Huso M (2010) An estimator of wildlife fatality from observed carcasses. *Environmetrics* 22: 318-329

Korner-Nievergelt F, Korner-Nievergelt P, Behr O, Niermann I, Brinkmann R, Hellriegel B (2011) A new method to determine bird and bat fatality at wind energy turbines from carcass searches. *Wildlife Biology* 17: 350-363

Korner-Nievergelt F, Behr O, Brinkmann R, Etterson MA, Huso MM, Dalthorp D, Korner-Nievergelt P, Roth T, Niermann I (2015) Mortality estimation from carcass searches using the R-package carcass ? a tutorial. *Wildlife Biology* 21: 30-43.

batdist

Distribution of bat carcasses below wind turbines

Description

The data contains the proportion of bats killed at wind turbines falling into different 10m-distance rings from the turbine. It is a collection of empirical and theoretical estimates from different publications.

Usage

```
data(batdist)
```

Format

A data frame with 6 observations on the following 16 variables.

`studyname` a factor with the name of the study

`study` a factor with the code of the study, see tutorial

`rotor.diameter.m` a numeric vector giving the average rotor diameter (in m) for the turbines investigated

`nacelle.height.m` a numeric vector giving the average nacelle height (in m) for the turbines investigated

`n` a numeric vector indicating the sample size (i.e. numbers of carcasses found)

`method` a factor indicating the method used to obtain the carcass distribution. `empirical`: based on field data, `physical_model`: according to a theoretical ballistic model

`species` a factor with only one level at the moment `bat`

`prop10m` a numeric vector with the proportion of killed bats lying in between 0 and 10 m from the turbine

`prop20m` a numeric vector with the proportion of killed bats lying in between 10 and 20 m from the turbine

`prop30m` a numeric vector with the proportion of killed bats lying in between 20 and 30 m from the turbine

prop40m a numeric vector with the proportion of killed bats lying in between 30 and 40 m from the turbine

prop50m a numeric vector with the proportion of killed bats lying in between 40 and 50 m from the turbine

prop60m a numeric vector with the proportion of killed bats lying in between 50 and 60 m from the turbine

prop70m a numeric vector with the proportion of killed bats lying in between 60 and 70 m from the turbine

prop80m a numeric vector with the proportion of killed bats lying in between 70 and 80 m from the turbine

prop50mplus a numeric vector: the sum of prop60m, prop70m and prop80m

References

Arnett, E. B., W. P. Erickson, J. Kerns, and J. Horn. 2005. Relationships between Bats and Wind Turbines in Pennsylvania and West Virginia: An Assessment of Fatality Search Protocols, Patterns of Fatality, and Behavioral Interactions with Wind Turbines. Bat Conservation International, Austin, Texas, USA.

Hull, C. L. and S. Muir. 2010. Search area for monitoring bird and bat carcasses at wind farms using a Monte-Carlo model. Australian Journal of Environmental Management 17:77-87.

Niermann, I., R. Brinkmann, F. Korner-Nievergelt, and O. Behr. 2011. Systematische Schlagopfer-suche - Methodische Rahmenbedingungen, statistische Analyseverfahren und Ergebnisse. Pages 40-115 in R. Brinkmann, O. Behr, I. Niermann, and M. Reich, editors. Entwicklung von Methoden zur Untersuchung und Reduktion des Kollisionsrisikos von Fledermaeusen an Onshore-Windenergieanlagen. Cuvillier Verlag, Goettingen.

Examples

```
data(batdist)
batdist
```

CIetterson	<i>Confidence interval for the functions ettersonEq14, ettersonEq14v1 and ettersonEq14v2</i>
------------	--

Description

Calculates the Monte Carlo confidence intervals for the estimated carcass detection probability when persistence probability and searcher efficiency are uncertain.

Usage

```
CIetterson(s, s.lwr, s.upr, f, f.lwr, f.upr, J, s.time.variance = "carcass age",
f.time.variance = "number of searches", nsim = 1000, ci = 0.95)
```

Arguments

s	point estimate for persistence probability (see help file for functions etterson14, ettersonEq14v1 or ettersonEq14v2)
s.lwr	lower limit of the 95% confidence interval of persistence probability
s.upr	upper limit of the 95% confidence interval of persistence probability
f	point estimate for the searcher efficiency (see help file for functions etterson14, ettersonEq14v1 or ettersonEq14v2)
f.lwr	lower limit of the 95% confidence interval of searcher efficiency
f.upr	upper limit of the 95% confidence interval of searcher efficiency
J	vector of search intervals
s.time.variance	character, one of "date" or "carcass age"
f.time.variance	character, one of "date" or "number of searches"
nsim	number of Monte Carlo simulations
ci	size of the confidence interval, default is 0.95

Details

The time variance in s and f is either both with date or both with carcass age and number of searches, respectively. In case of constant s and f, the function uses ettersonEq14 independent of the arguments s.time.variance or f.time.variance, when only one value is given for both parameters.

Value

a list	
p.lower	lower limit of the confidence interval
p.upper	upper limit of the confidence interval

Author(s)

F. Korner

Examples

```
J <- c(2,3,2,4,3,5,3,2,3,4)
s <- plogis(seq(0.2, 2, length=sum(J)))
f <- plogis(seq(1.5, 0.9, length=length(J)))

s.lwr<- plogis(seq(0.2, 2, length=sum(J))-0.5)
f.lwr <- plogis(seq(1.5, 0.9, length=length(J))-0.3)

s.upr <- plogis(seq(0.2, 2, length=sum(J))+0.5)
f.upr <- plogis(seq(1.5, 0.9, length=length(J))+0.3)

CIetterson(s=s, s.lwr=s.lwr, s.upr=s.upr, f=f, f.lwr=f.lwr, f.upr=f.upr, J=J, nsim=100)
# nsim is too low, please, increase!
```

estimateN	<i>Estimation of number of killed animals based on carcass searches and estimates for detection probability</i>
-----------	---

Description

Estimates the number of killed animals from carcass searches taking into account the uncertainty in the estimates for carcass detection probability (or alternatively for searcher efficiency and carcass persistence probability).

Usage

```
estimateN(count, p = NA, p.lower = NA, p.upper = NA,
          f = NA, f.lower = NA, f.upper = NA, s = NA,
          s.lower = NA, s.upper = NA, arrival="discrete", a=1, a.lower=1, a.upper=1,
          pform = "korner", d = 1, n = NA, J=NA, maxn = 1000,
          nsim = 1000, plot = TRUE, postdist=FALSE, k=1, x=c(1:10))
```

Arguments

count	number of carcasses found
p	estimate for detection probability (alternatively give f, s, d and n)
p.lower	lower limit of 95% CI of detection probability
p.upper	upper limit of 95% CI of detection probability
f	estimate for searcher efficiency
f.lower	lower limit of 95% CI of searcher efficiency
f.upper	upper limit of 95% CI of searcher efficiency
s	estimate for persistence probability: this is a scalar in case of constant persistence probability. In case of non-constant persistence probability, it is a vector. ATTENTION: If 'pform="korner"', the proportion of remaining carcasses until each day after death (survivor function) have to be given. If 'pform="etterson"', the daily persistence probabilities for each day after death are needed.
s.lower	lower limit of 95% CI of persistence probability
s.upper	upper limit of 95% CI of persistence probability
arrival	either "uniform" or "discrete". assumption of the distribution of carcass arrival times: if "uniform" it is assumed that carcasses arrive and are removed at a constant rate over time. if "discrete" (default) it is assumed that all carcasses arrive every day at the same time and and persistence probability given for the first day corresponds to the time from death to the time when carcass searches normally take place on the first day (the search does not necessary need to take place on the first day).
a	estimate for the proportion of killed animals that falled into the searched area. default is 1 (complete area searched).

a.lower	lower limit of 95% CI of the proportion of killed animals that falled into the searched area. default is 1 (complete area searched).
a.upper	upper limit of 95% CI of the proportion of killed animals that falled into the searched area. default is 1 (complete area searched).
pform	formula used to estimate p, one of "korner", "huso", "erickson", "etterson"
d	search interval, the number days (or any other time unit) between two searches
n	number of searches
J	vector of length of search intervals. This argument can only be used with pform="etterson". For the other methods regular search intervals are used and their length is given in the argument d.
maxn	maximal possible number of animals killed for which the posterior probability is estimated (should not be too high but also not be too low!)
nsim	number of Monte Carlo simulations
plot	logical, TRUE = posterior probability is plotted
postdist	logical, TRUE = the posterior probability distribution of the number of fatalities is given in the output. See details.
k	numeric, factor of decrease of searcher efficiency with number of searches, see help to pkorner.
x	vector of integers. The posterior probability of the number of killed animals being larger than x will be calculated.

Details

Either p (with p.lower and p.upper) OR f (with f.lower and f.upper), s (with s.lower and s.upper), n and d are needed. In the latter case, p is calculated from f, s, n and d using the formula specified in the argument 'pform'.

A documented code of this function is given in the Appendix of Korner-Nievergelt et al. (2011) J. Wildlife. Biol.

If postdist is TRUE, the output is a list, if postdist is FALSE, the output is a named numeric vector.

Value

If postdist=FALSE, the function produces a vector with the median of the posterior distribution ("estimate"), the 2.5% and the 97.5% quantiles ("lower", and "upper") and the Horvitz-Thompson estimate ("HT.estimate", which equals count/p). If postdist=TRUE, the output is a list with the following elements:

estimate	median of the posterior distribution
lower	2.5% quantile of the posterior distribution
upper	97.5% quantile of the posterior distribution
HT.estimate	Horvitz-Thompson estimate, i.e. count/p
P.true.larger.x	Posterior probability that the true mortality is larger than x.

Author(s)

F. Korner-Nievergelt

References

Erickson WP, Jeffrey J, Kronner K, Bay K (2004) Stateline Wind Project Wildlife. Monitoring Final Report, July 2001 - December 2003. - Technical report peer-reviewed by and submitted to FPL Energy, the Oregon Energy Facility Siting Council, and the Stateline Technical Advisory Committee, Oregon, USA, 105 pp

Etterson, MA (2013) Hidden Markov models for estimating animal mortality from antropogenic hazards. *Ecological Applications* 23: 1915-1925

Huso M (2010) An estimator of wildlife fatality from observed carcasses. *Environmetrics* 22: 318-329

Korner-Nievergelt F, Korner-Nievergelt P, Behr O, Niermann I, Brinkmann R, Hellriegel B (2011) A new method to determine bird and bat fatality at wind energy turbines from carcass searches. *Wildlife Biology* 17: 350-363 Huso M, Dathorp D in prep.

See Also

[posteriorN](#)

Examples

```
estimateN(count=3, f=0.72, f.lower=0.62, f.upper=0.81, s=0.84, s.lower=0.64,
          s.upper=0.94, d=2, pform="korner", n=100, maxn=500, nsim=1000,
          plot=TRUE)
```

```
estimateN(count=3, f=0.72, f.lower=0.62, f.upper=0.81, s=0.84, s.lower=0.64,
          s.upper=0.94, d=2, pform="huso", maxn=500, nsim=1000, plot=TRUE)
```

```
res.p <- pkorner(f=0.72, f.lower=0.62, f.upper=0.81, s=0.84, s.lower=0.64, s.upper=0.94,
               d=2, n=100, CI=TRUE)
estimateN(count=3, p=res.p["p"], p.lower=res.p["2.5%"], p.upper=res.p["97.5%"])
```

ettersonEq14

Equation 14 of Etterson (2013) Ecological Applications 23, 1915-1925

Description

Calculates the probability that a carcass that has fallen in the search area during n regular or irregular search intervals is found by a searcher.

Usage

```
ettersonEq14(s, f, J)
```


Arguments

s	persistence probability, probability that a carcass persists a specific time interval (normally a day), i.e. daily persistence probability. Note that the same time unit as for the length of the search interval must be used.
f	searcher efficiency, i.e. the probability that a carcass present in the search area is found during one search.
J	a vector containing the lengths of the search intervals.

Details

This formula can handle irregular search intervals. It assumes constant persistence time and constant searcher efficiency. Confidence intervals are not yet implemented, but a Monte Carlo confidence intervals will be calculated by the function estimateN with the argument form="etterson".

Value

the probability that a carcass that has fallen into the search area during the study is found by the searcher.

Note

In case of regular search intervals the functions ettersonEq14 and pkorner are equivalent (see first example). A mathematical proof for this equivalence is given in the tutorial to carcass (Korner-Nievergelt et al., re-submitted to Wildlife Biology June 2014). When search intervals are irregular, pkorner overestimates the detecton probability (i.e. underestimates fatality rates) as you see in the second example below.

Author(s)

M. A. Etterson

References

Etterson, M.A. (2013) Hidden Markov models for estimating animal mortality from antropogenic hazards. Ecological Applications, 23, 1915-1925.

See Also

[ettersonEq14v1](#) [ettersonEq14v2](#) [pkorner](#) [phuso](#) [perickson](#)

Examples

```
# in case of regular search intervals, the calculations below give the same results
ettersonEq14(s=0.8, f=0.8, J=c(3,3,3,3,3))
pkorner(s=0.8, f=0.8, d=3, n=5)
```

```
# in case of irregular search intervals the function ettersonEq14 is more appropriate
ettersonEq14(s=0.8, f=0.8, J=c(3,5,1,4,2))
```

```
pkorner(s=0.8, f=0.8, d=mean(c(3,5,1,4,2)), n=5)
```

ettersonEq14v1 *Equation 14 of Etterson (2013) Ecological Applications 23, 1915-1925, adapted so that persistence probability and searcher efficiency can vary with calendar date*

Description

Calculates the probability that a carcass that has fallen in the search area during n regular or irregular search intervals is found by a searcher.

Usage

```
ettersonEq14v1(s, f, J)
```

Arguments

s vector of persistence probabilities with N elements, where N is the number of days in the full study period (i.e, $N = \text{sum}(J)$).

f vector of searcher efficiency values (probabilities that a carcass present in the search area is found during one search) with n elements, where n is the number of searches performed.

J a vector containing the lengths of the search intervals.

Details

This function assumes that persistence probability and searcher efficiency change over time and that this change depends upon serial date within your study period. In other words, it assumes that all carcasses in the environment on a given date experience the same scavenging probability and detectability, regardless of how old the carcasses is. However it does allow persistence probability and searcher efficiency to change with the calendar date.

Value

the probability that a carcass that has fallen into the search area during the study is found by the searcher.

Author(s)

M. A. Etterson

References

Etterson, M.A. (2013) Hidden Markov models for estimating animal mortality from antropogenic hazards. *Ecological Applications*, 23, 1915-1925.

See Also

[ettersonEq14](#) [ettersonEq14v2](#) [pkorner](#) [phuso](#) [perickson](#)

Examples

```
J <- c(2,3,2,4,3,5,3,2,3,4)
s <- plogis(seq(0.2, 2, length=sum(J)))
f <- plogis(seq(1.5, 0.9, length=length(J)))
ettersonEq14v1(s, f, J)
```

ettersonEq14v2

Equation 14 of Etterson (2013) Ecological Applications 23, 1915-1925, adapted so that persistence probability and searcher efficiency can vary with age of the carcass

Description

Calculates the probability that a carcass that has fallen in the search area during n regular or irregular search intervals is found by a searcher.

Usage

```
ettersonEq14v2(s, f, J)
```

Arguments

s vector of persistence probabilities with N elements, where N is the number of days in the full study period (i.e, $N = \text{sum}(J)$) (so N is the maximum number of days that a carcass could be present in the environment).

f vector of discovery probabilities with n elements, where n is the number of searches performed.

J a vector containing the lengths of the search intervals.

Details

This function assumes that persistence probability and searcher efficiency change over time and that this change depends upon the time elapsed since the carcass was killed (carcass age). If you assume that the change in detectability of a carcass is due to its age, then this function may not be reliable because it indexes the searcher efficiency to the number of searches the carcass has been exposed to. However, if you assume that searcher efficiency changes with the number of searches a carcass was exposed to because the easy carcasses are found first, then this function should perform well.

Value

the probability that a carcass that has fallen into the search area during the study is found by the searcher.

Author(s)

M. A. Etterson

References

Etterson, M.A. (2013) Hidden Markov models for estimating animal mortality from antropogenic hazards. *Ecological Applications*, 23, 1915-1925.

See Also

[ettersonEq14](#) [ettersonEq14v1](#) [pkorner](#) [phuso](#) [perickson](#)

Examples

```
J <- c(2,3,2,4,3,5,3,2,3,4)
s <- plogis(seq(0.2, 2, length=sum(J)))
f <- plogis(seq(1.5, 0.9, length=length(J)))
ettersonEq14v2(s,f,J)
```

`integrate.persistence` *Integrate persistence probability over the discrete time intervals (e.g. days) to account for constant arrival of carcasses*

Description

The function integrates the proportion of carcasses remaining on the ground (measured at discrete time intervals after death) over the last time interval. This gives for each discrete time point the proportion of carcasses that have arrived continuously (uniformly distributed) during the first time interval. This takes into account that carcasses of the same cohort (arrived during the same time interval) can be of different age (max. age difference is length of time interval, e.g. one day in most studies).

Usage

```
integrate.persistence(s, n = 20, d = 1)
```

Arguments

s Either a number or a numeric vector. If a number is given (which should be the probability that a carcass persists one time interval, e.g. 24 hours), an exponential survivor function (constant persistence probability) is assumed and the proportion of carcasses persisting until the end of every time interval is calculated. Thus, the function returns a vector. The first number is the proportion of carcasses remaining until the end of the first time interval, which is the time interval during which the carcasses have continuously arrived at a constant rate. The second number gives the proportion of carcasses that have arrived during the first time interval and that are still persisting until the end of the second time

interval, a.s.o. This proportion is calculated up to the n-th time interval. If a vector is given, this vector can be either the proportion of carcasses remaining after every time interval after death ("survivor function") or the age dependent probability to persist one time interval (e.g. daily persistence probability). In the first case, the function returns the proportion of persisting carcasses after each time interval assuming that they arrived at a constant rate during the first time interval. In the second case the same integration leads to a average of the persistence probability at time t and t-1 and can thus be seen as a correction for the different ages of the carcasses belonging to the same cohort. **ATTENTION:** if you use an exponential survival model (constant persistence probability) together with the function `ettersonEqv2` that requires daily persistence probabilities rather than survivor functions, you must not provide `s` as a single number here. Use `rep(s, n)` in this case!

- n number of time intervals for which the proportion of persisting carcasses is calculated, default is 20
- d This argument is only needed by the function `estimateN`, where `n` is the number of searches and `d` the search interval. Therefore, the number of days of the study is `n*d` and it is reasonable to calculate the persistence probabilities for a maximum time span that equals the study length.

Details

IMPORTANT: if you set the argument "`arrival='uniform'`" in the `estimateN` function, you must not use the function `integrate.persistence` prior to use the `estimateN` function. Since `estimateN` does this integration when "`arrival='uniform'`". But the `integrate.persistence` function is useful in combination with the function `pkorner`, `phuso`, `perikson`, and all `etterson`-formulas.

Value

A vector containing the proportion of carcasses of one cohort of carcasses (that have arrived at a constant rate during the first time interval) that persist until the end of `n` time intervals.

Author(s)

F. Korner

References

Software in prep by Dan Dalthorp and Manuela Huso

See Also

[estimateN](#)

Examples

```
integrate.persistence(0.5)
integrate.persistence(c(0.8,0.7,0.6,0.55))
```

perickson

Carcass detection probability according to Erickson et al. 2004

Description

Calculates carcass detection probability based on persistence time, searcher efficiency, search interval and duration of study.

Usage

```
perickson(t.bar=NULL, s, f, d)
```

Arguments

t.bar	mean carcass persistence time in days (or another time unit)
s	mean (e.g. daily) carcass persistence probability.
f	searcher efficiency: probability that a carcass present on the study plot is detected by a searcher during a search.
d	search interval: time (in days) between two searches

Details

Time measurements (search interval and carcass persistence time or persistence probability) should be given in the same unit. Either t.bar or s has to be provided.

Value

a proportion: the probability that an animal that dies during the study period on the study plot is detected by a searcher

Author(s)

Fraenzi Korner-Nievergelt

References

Erickson WP, Jeffrey J, Kronner K, Bay K (2004) Stateline Wind Project Wildlife. Monitoring Final Report, July 2001 - December 2003. - Technical report peer-reviewed by and submitted to FPL Energy, the Oregon Energy Facility Siting Council, and the Stateline Technical Advisory Committee, Oregon, USA, 105 pp

See Also

[phuso pkorner](#)

Examples

```
perickson(t.bar=30, f=0.8, d=1)
```

persistence

Times until removal for brown mice carcasses

Description

Experimental data to measure persistence time of bat-sized mammal carcasses. In total, 595 brown mice and 41 bat carcasses have been put on the ground at 30 different sites (turbineID). During 14 days, the carcasses have been observed and the number of days until their disappearance was noted. All carcasses that did not disappear until day 14 were marked as censored (status=0) in the data.

Usage

```
data(persistence)
```

Format

A data frame with 636 observations on the following 3 variables.

turbineID a factor with site or turbine names

perstime a numeric vector containing the days until disappearance (persistence times)

status a numeric vector 1 = disappearance is observed, 0 = no disappearance until end of study (censored)

Details

Due to conservation reasons, experiments to estimate carcass persistence times have been conducted with brown mice instead of bat carcasses. There has been concern that bats have longer persistence times than mice. In our data, during the first four days removal probability of mice and bats were identical. After four days, bats seem to be removed less likely. However, this difference was based on such low sample size that we do not dare to claim any difference at the moment. For the moment, we can say that it is probably save to use mice when the search interval is less than four days. But for longer search intervals, it may be better to use bats to measure carcass persistence probability.

Source

Niermann, I., R. Brinkmann, F. Korner-Nievergelt, and O. Behr. 2011. Systematische Schlagopfer-suche - Methodische Rahmenbedingungen, statistische Analyseverfahren und Ergebnisse. Pages 40-115 in R. Brinkmann, O. Behr, I. Niermann, and M. Reich, editors. Entwicklung von Methoden zur Untersuchung und Reduktion des Kollisionsrisikos von Fledermaeusen an Onshore-Windenergieanlagen. Cuvillier Verlag, Goettingen.

References

Niermann, I., R. Brinkmann, F. Korner-Nievergelt, and O. Behr. 2011. Systematische Schlagopfer-suche - Methodische Rahmenbedingungen, statistische Analyseverfahren und Ergebnisse. Pages 40-115 in R. Brinkmann, O. Behr, I. Niermann, and M. Reich, editors. Entwicklung von Methoden zur Untersuchung und Reduktion des Kollisionsrisikos von Fledermaeusen an Onshore-Windenergieanlagen. Cuvillier Verlag, Goettingen.

Examples

```
data(persistence)
head(persistence)
```

persistence.prob	<i>Estimates carcass persistence probability based on carcass removal experiment data</i>
------------------	---

Description

This function either uses a Cox proportional hazard model or an exponential model (if persistence probability can assumed to be constant over time) to estimate daily persistence probabilities of carcasses.

Usage

```
persistence.prob(turbineID, perstime, status, pers.const = FALSE, R = 10000)
```

Arguments

turbineID	factor, character or numeric vector with name of the turbines or sites for which the carcass persistence probability should be estimated separately
perstime	numeric vector with the persistence times for each object (e.g. in days)
status	indicator variable of observed removal (1= removal has been observed, 0 = object was still there at the end of the observation period)
pers.const	logical value indicating whether a constant persistence probability over time can be assumed. default is FALSE. If FALSE, a Cox proportional hazard model is used and for each turbine/site the estimated proportion of remaining objects is given for each day. If TRUE, an exponential model is fitted and the estimated daily persistence probability is give for each turbine/site.
R	number of Monte Carlo simulations used to obtain the 95 percent confidence intervals of the estimated persistence probabilities from the exponential model

Details

Note that there is increasing evidence in the literature that carcass persistence probability increases with the age of a carcass. Thus you are saver to use non-constant persistence probabilities unless you have tested, how seriously an assumption of constant persistence probability influences your results.

Value

If you do not assume constant persistence probability, the function returns a list:

```
persistence.prob      matrix with estimated proportion of remaining carcasses after each time indicated on the rows for each turbine/site indicated on the column
estpers.lwr          the lower limits of the 95 percent confidence intervals
estpers.upr          the upper limits of the 95 percent confidence intervals
```

If you assume constant persistence probability, the function returns a data frame with the following variables:

```
turbineID: name of the turbine/site
persistence.prob: estimated daily persistence probability
lower: lower limit of the 95 percent confidence interval of the estimated persistence probability
upper: upper limit of the 95 percent confidence interval of the estimated persistence probability
mean.persistence.time: estimated mean persistence time
```

Note

Whether the models used in this function fits to your data is not in the responsibility of the author!

Author(s)

Fraenzi Korner-Nievergelt

References

Cox, D. R. 1972. Regression models and life-tables (with discussion). *Journal of the Royal Statistical Society B* 34:187-220.
Klein, J. P. and M. L. Moeschberber. 2003. *Survival Analysis, Techniques for Censored and Truncated Data*. Springer, New York.

See Also

You will find more information in the documentation of the packages `coxph` and `survreg`.

Examples

```
data(persistence)
persistence.prob(persistence$turbineID, persistence$perstime, persistence$status)
persistence.prob(persistence$turbineID, persistence$perstime, persistence$status,
  pers.const=TRUE)
```

phuso

*Carcass detection probability according to Huso 2010***Description**

Calculates carcass detection probability based on persistence time, searcher efficiency, search interval and duration of study. Persistence time obtained by any survival model (e.g. Exponential, Weibull, Log-Normal,...) can be given (but see below). Alternatively to persistence time, persistence probability can be given, In this case, the exponential model (constant persistence probability) is used to transform persistence probability into mean persistence time.

An argument shape and distribution for other than the Exponential persistence model will be provided soon.

Usage

phuso(s, t.bar, f, d)

Arguments

s	persistence probability: probability that a carcass remains on the study plot for one day. This parameter is only used when t.bar is not provided. Note that if s is given instead of t.bar, an exponential survival function (constant persistence probability) is used.
t.bar	mean persistence time (in days). Alternatively, s can be provided. Then, an exponential persistence model is assumed.
f	searcher efficiency: probability that a carcass present on the study plot is detected by a searcher during a search.
d	search interval: time (in days) between two searches

Details

Time measurements (search interval) and reference time units (persistence probability) should be given in the same unit.

Value

a proportion: the probability that an animal that dies during the study period on the study plot is detected by a searcher

Author(s)

Fraenzi Korner-Nievergelt

References

Huso M (2010) An estimator of wildlife fatality from observed carcasses. *Environmetrics* 22: 318-329

See Also[perickson pkorner](#)**Examples**

```
phuso(s=0.8, f=0.7, d=7)
```

pkorner	<i>Carcass detection probability according to Korner-Nievergelt et al. 2011</i>
---------	---

Description

Calculates carcass detection probability based on persistence probability, searcher efficiency, search interval and duration of study.

Usage

```
pkorner(s, s.lower=NA, s.upper=NA, f, f.lower=NA, f.upper=NA, d, n, k=0.25,
        search.likelihood.constant=TRUE, CI=FALSE, nsim=1000)
```

Arguments

s	persistence probability: Probability that a carcass remains on the study plot for one day (or another time unit). Scalar in case of constant persistence probability, vector with probabilities to persist until each day after death ('survivor function', i.e. proportion of remaining carcasses for each day after death) in case of non-constant persistence probabilities.
s.lower	lower limit of 95% CI of persistence probability
s.upper	upper limit of 95% CI of persistence probability
f	searcher efficiency: Probability that a carcass present on the study plot is detected by a searcher during a search.
f.lower	lower limit of 95% CI of searcher efficiency
f.upper	upper limit of 95% CI of searcher efficiency
d	search interval: time (in days) between two searches
n	number of searches: $n*d$ = duration of study
k	Factor by which the search efficiency is multiplied after each search for a specific carcass (see details).
search.likelihood.constant	TRUE if search efficiency is constant over time, FALSE if it decreases by a factor k with each search.
CI	TRUE if 95% CI should be calculated based on a beta distribution and Monte Carlo simulations, FALSE otherwise.
nsim	number of Monte Carlo simulations

Details

Time measurements (search interval) and reference time units (persistence probability) should be given in the same unit. The argument `k` together with `search.ency.constant=FALSE` can be given to account for the fact that during the first search carcasses with high detectability are found overproportionately, whereas carcasses hard to find remain on the study plot. Thus with each search, searcher efficiency decreases by the factor `k`. If `search.ency.constant=TRUE` the value given at the argument `k` is ignored (i.e. `k=1`). If one assumes that persistence probability is not constant, the function `persistence.prob` could be used to provide the data for `s` (see examples below). An estimate for `s` should be provided for every day for a sufficient time span, e.g. 14 days. Missing estimates for intermediate days have to be interpolated. After the last day with an `s` provided, the function assumes that the proportion of persisting carcasses is zero, e.g. `s` is assumed to be zero from day 15 and onwards if `s` is a vector of length 14.

Value

a proportion: the probability that an animal that dies during the study period on the study plot is detected by a searcher

Author(s)

Fraenzi Korner-Nievergelt

References

Korner-Nievergelt F, Korner-Nievergelt P, Behr O, Niermann I, Brinkmann R, Hellriegel B (2011) A new method to determine bird and bat fatality at wind energy turbines from carcass searches. *Wildlife Biology* 17: 350-363

See Also

[phuso](#), [perickson](#), [persistence.prob](#)

Examples

```
### Data
f <- 0.72
s <- 0.8
data(persistence)
attach(persistence)
sv <- persistence.prob(turbineID, perstime, status)$persistence.prob[,1]
sv.lower <- persistence.prob(turbineID, perstime, status)$lower[,1]
sv.upper <- persistence.prob(turbineID, perstime, status)$upper[,1]
n <- 4
d <- 3

### Constant search efficiency and constant persistence probability
pkorner(s=s, f=f, d=d, n=n)
pkorner(s=s, s.lower=0.6, s.upper=0.9, f=f, f.lower=0.6, f.upper=0.8,
        d=d, n=n, CI=TRUE)
```

```

### Decreasing search efficiency and constant persistence probability
pkorner(s=s, f=f, d=d, n=n, k=0.25, search.likelihood.constant=FALSE)

### Constant search efficiency and decreasing persistence probability
pkorner(s=sv, f=f, d=d, n=n)

### Decreasing search efficiency and decreasing persistence probability
pkorner(s=sv, f=f, d=d, n=n, search.likelihood.constant=FALSE)
pkorner(s=sv, s.lower=sv.lower, s.upper=sv.upper, f=f, f.lower=0.6,
        f.upper=0.8, d=d, n=n, search.likelihood.constant=FALSE, CI=TRUE)

```

posteriorN	<i>Posterior distribution of the number of fatalities based on the number of carcasses found and the probability of finding a carcass.</i>
------------	--

Description

This function gives the posterior distribution of the number of fatalities based on the number of carcasses counted during carcass searches and the probability of detecting a carcass. The function uses the theorem of Bayes.

Usage

```
posteriorN(p, nf = 0, maxN = 1000, ci.int = 0.95, plot = TRUE,
           dist = FALSE)
```

Arguments

p	probability of finding a carcass. This probability can be obtained by e.g. pkorner, phuso or perickson.
nf	the number of carcasses found
maxN	the maximal number of fatalities for which a posterior density should be given. It should be higher than the upper tail (density larger 0.0001) of the posterior distribution (there is a warning if not). If maxN is lower than the lower edge of the posterior distribution, an error is produced.
ci.int	length of the credible interval that should be given. Default is 0.95.
plot	if TRUE the posterior distribution is plotted
dist	if TRUE the posterior density is given for all elements in 0:maxN

Value

interval	vector of length 2 containing the lower and upper limit of the interval specified in the argument ci.int
expected	median of the posterior distribution of the number of fatalities
HT.estimate	Horvitz-Thompson estimate, i.e. nf/p
pN	posterior density for each element in 0:maxN number of fatalities

Note

The function does not take into account the uncertainty of the estimate for the detection probability. If you want do account for this uncertainty, use the function estimateN. For reporting bugs in the code, we would like to thank A. Marcia Barbosa.

Author(s)

Fraenzi Korner-Nievergelt

References

Korner-Nievergelt F, Korner-Nievergelt P, Behr O, Niermann I, Brinkmann R, Hellriegel B (2011) A new method to determine bird and bat fatality at wind energy turbines from carcass searches. *Wildlife Biology* 17: 350-363

See Also

[estimateN](#)

Examples

```
posteriorN(p=0.5, nf=3, dist=TRUE, maxN=15)
```

search. efficiency	<i>Estimates detection probability per person and visibility classes using a binomial model</i>
--------------------	---

Description

The function estimates detection probability per visibility class and person. A binomial mixed model is used with vegetation density as fixed effect and person as random factor, when data of more than 2 persons are available. The number of found and the number of overseen items have to be provided per person and visibility class (one line per observer and visibility class). When only one visibility class is available, the variable visibility must be provided with only one entry. It does not matter what. The detection probabilities are given per person with a 95% credible interval. An average vegetation density specific detection probability over all persons is given in addition.

Usage

```
search. efficiency(dat=NA, person=NA, visibility=NA, detected=NA,  
notdetected=NA, nsim = 1000)
```

Arguments

The search efficiency data may be provided as a `data.frame` containing all data or, alternatively, as separate vectors. If no visibility classes are available, the variable `visibility` should be a vector with the same length as the others containing a single value (e.g. "not_measured").

<code>dat</code>	Data.frame containing the following columns: <code>person</code> : names of the persons who searched <code>visibility</code> : visibility class <code>detected</code> : number of detected items <code>notdetected</code> : number of not detected items
<code>person</code>	vector with names of the persons who searched
<code>visibility</code>	vector with visibility classes
<code>detected</code>	numeric vector with number of detected items
<code>notdetected</code>	numeric vector with number of not detected items
<code>nsim</code>	the number of simulations from the posterior distributions of the model parameters used to construct the 95 percent credible intervals

Details

The function uses the function `glmer` of the package `lme4` and the function `sim` of the package `arm`.

Value

A list with two elements of class `data.frame`

<code>f.perperson</code>	a <code>data.frame</code> with the estimated detection probabilities per person and visibility class, its lower and upper limit of the 95% credible interval and its standard error
<code>f.average</code>	a <code>data.frame</code> with the estimated detection probabilities per visibility class averaged over the persons

Author(s)

Fraenzi Korner-Nievergelt

References

Gelman A, Hill J (2007) Data Analysis Using Regression and Multilevel and Hierarchical Models. Cambridge: Cambridge University Press.

Niermann I, Brinkmann R, Korner-Nievergelt F, Behr O (2011) Systematische Schlagopfersuche - Methodische Rahmenbedingungen, statistische Analyseverfahren und Ergebnisse. In: Brinkmann R, Niermann I, Behr O, editors. Entwicklung von Methoden zur Untersuchung und Reduktion des Kollisionsrisikos von Fledermaeusen an Onshore-Windenergieanlagen Goettingen: Cuvillier Verlag. pp. 40-115.

Examples

```
data(searches)
searches

# Call to the function with data provided as data.frame:
## Not run: search.efficiency(searches)

# Alternative:
per <- searches$person
visi <- searches$visibility
det <- searches$detected
notdet <- searches$notdetected
## Not run: search.efficiency(person=per, visibility=visi, detected=det,
notdetected=notdet)
## End(Not run)
```

searches	<i>Data of a searcher efficiency trial</i>
----------	--

Description

Data containing the number of seen and not-seen items of a searcher efficiency trial performed in 3 different vegetation densities (visibilities) and for various persons.

Usage

```
data(searches)
```

Format

A data frame with 17 observations on the following 4 variables.

person a factor with the names of the persons
visibility a factor with visibility classes (vegetation densities)
detected a numeric vector indicating the number of found items
notdetected a numeric vector indicating the number of not found items

Source

Niermann I, Brinkmann R, Korner-Nievergelt F, Behr O (2011) Systematische Schlagopfersuche - Methodische Rahmenbedingungen, statistische Analyseverfahren und Ergebnisse. *In:* Brinkmann R, Niermann I, Behr O, editors. Entwicklung von Methoden zur Untersuchung und Reduktion des Kollisionsrisikos von Fledermaeusen an Onshore-Windenergieanlagen Goettingen: Cuvillier Verlag. pp. 40-115.

Examples

```
data(searches)
str(searches)
search.efficiency(searches)
```

shapeparameter	<i>Shapeparameters of a beta-distribution from the mean, the lower and upper limit of the 95% confidence or credible interval</i>
----------------	---

Description

The function returns the shapeparameters a and b of a beta-distribution from the mean, the lower and upper limit of the 95% confidence or credible interval. The derivation from mean and se to the shapeparameters of a beta-distribution is done by the method of moments.

Usage

```
shapeparameter(m, lwr=NA, upr=NA, se=NA)
```

Arguments

m	mean
lwr	lower limit of the 95percent confidence/credible interval
upr	upper limit of the 95percent confidence/credible interval
se	standard error of the proportion

Details

Either se or both lwr and upr has to be given. The result may be more reliable if se is given, since the calculation is based on se. If the lwr and upr limits of the 95% confidence interval are given, se is estimated as 1/4 times the width of the 95% confidence interval.

Value

a	parameter a of a beta-distribution
b	parameter b of a beta-distribution

Author(s)

Fraenzi Korner-Nievergelt

References

Gelman et al. (2004)

Examples

```
a <- shapeparameter(0.8, 0.72, 0.88)$a
b <- shapeparameter(0.8, 0.72, 0.88)$b
x <- seq(0, 1, by=0.01)
y <- dbeta(x, a, b)
plot(x, y, type="l")
```

Index

* datasets

batdist, [3](#)
persistence, [15](#)
searches, [24](#)

* methods

estimateN, [6](#)
ettersonEq14, [8](#)
ettersonEq14v1, [10](#)
ettersonEq14v2, [11](#)
perickson, [14](#)
persistence.prob, [16](#)
phuso, [18](#)
pkorner, [19](#)
posteriorN, [21](#)
shapeparameter, [25](#)

* misc

CIetterson, [4](#)
estimateN, [6](#)
ettersonEq14, [8](#)
ettersonEq14v1, [10](#)
ettersonEq14v2, [11](#)
integrate.persistence, [12](#)
perickson, [14](#)
persistence.prob, [16](#)
phuso, [18](#)
pkorner, [19](#)
posteriorN, [21](#)
search.efficiency, [22](#)
shapeparameter, [25](#)

* package

carcass-package, [2](#)

batdist, [3](#)

carcass (carcass-package), [2](#)
carcass-package, [2](#)
CIetterson, [4](#)

estimateN, [6](#), [13](#), [22](#)
ettersonEq14, [8](#), [11](#), [12](#)

ettersonEq14v1, [9](#), [10](#), [12](#)
ettersonEq14v2, [9](#), [11](#), [11](#)

integrate.persistence, [12](#)

perickson, [9](#), [11](#), [12](#), [14](#), [19](#), [20](#)
persistence, [15](#)
persistence.prob, [16](#), [20](#)
phuso, [9](#), [11](#), [12](#), [14](#), [18](#), [20](#)
pkorner, [9](#), [11](#), [12](#), [14](#), [19](#), [19](#)
posterior.N (posteriorN), [21](#)
posteriorN, [8](#), [21](#)

search.efficiency, [22](#)
searches, [24](#)
shapeparameter, [25](#)