

# Mortality estimation from carcass searches using the R-package carcass – a tutorial

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

*F. Korner-Nievergelt (fraenzi.korner@oikostat.ch) and P. Korner-Nievergelt, oikostat GmbH, Ausserdorf 43, CH-6218 Ettiswil, Switzerland and Swiss Ornithological Institute, Sempach, Switzerland. – O. Behr, Chair of Sensor Technology, Univ. of Erlangen, DE-91054 Erlangen, Germany. – R. Brinkmann, Freiburg Inst. for Applied Animal Ecology, FrInaT, DE-79106 Freiburg, Germany. – M. A. Etterson, US Environmental Protection Agency, Duluth, MN 55804-2595, USA. – M. Huso and D. Dalthorp, US Geological Survey Forest and Rangeland Ecosystem Science Center, Corvallis, OR 97331, USA. – T. Roth, Hintermann & Weber AG, Austrasse 2A, CH-4153 Reinach, Switzerland. – I. Niermann, Inst. of Environmental Planning, Leibniz Universität Hannover, DE-30419 Hannover, Germany.*

This article is a tutorial for the R-package *carcass*. It starts with a short overview of common methods used to estimate mortality based on carcass searches. Then, it guides step by step through a simple example. First, the proportion of animals that fall into the search area is estimated. Second, carcass persistence time is estimated based on experimental data. Third, searcher efficiency is estimated. Fourth, these three estimated parameters are combined to obtain the probability that an animal killed is found by an observer. Finally, this probability is used together with the observed number of carcasses found to obtain an estimate for the total number of killed animals together with a credible interval.

## What is carcass used for?

Collisions of bats and birds with man-made structures such as large buildings, towers, wind turbines or traffic are of ecological concern (Schaub and Lebreton 2004, Kunz et al. 2007). To assess mortality rates, systematic searches for carcasses are often performed. The number of carcasses found, however, largely underestimates mortality because only a subset of the animals killed fall into an area that is searched (the ‘search area’). Furthermore, many carcasses may be removed by scavengers before the searches. Finally, of the dead animals that have not been removed by scavengers some may remain undetected because the probability of finding a carcass during a search usually is less than one. In order to obtain unbiased estimates of mortality from systematic carcass searches the following three parameters must be measured or estimated: 1) proportion of animals killed that fall into the search area  $a$ , 2) carcass persistence probability  $s$ , and 3) searcher efficiency  $f$ . The proportion of animals killed that fall into the search area  $a$  can be obtained from the spatial distribution of the search area in relation to the spatial distribution of the carcasses. The spatial distribution of the carcasses mainly depends on the type of obstacle they collide with. The package *carcass* includes a table with empirical and theoretical distributions of bat carcasses beneath wind turbines of different sizes. However, the spatial distribution of the carcasses is likely to depend on further parameters, e.g. the wind speed, mass of the animal and direction.

Some studies are designed in such a way that carcass persistence probability  $s$  and searcher efficiency  $f$  can be estimated from the carcass search data alone using mark-recapture models (Lebreton et al. 1992), see examples in Guinard et al. (2012) or Péron et al. (2013). The package *carcass*, however, is designed for studies where carcass persistence probability and searcher efficiency are estimated based on additional experiments, such as laying out trial carcasses to monitor their removal over time or to test naive observers. For such data, *carcass* may be used to estimate persistence probability and searcher efficiency, to subsequently combine these parameters into an estimate of the probability that an animal that has died during the study period is found by the observer  $p$ , and to finally estimate the mortality  $N$  given  $p$  and the number of carcasses found  $c$ .

The function `persistence.prob` estimates site-specific persistence probabilities based on experimental data. The function `search.efficiency` estimates the probability that a carcass that persisted is found by an observer. These two functions are wrappers of standard statistical methods. They are written for people with little experience in applying statistical models. However, before the functions are applied, please make sure that the data meet the model assumptions described below.

Estimates for carcass persistence probability and for searcher efficiency can be combined taking into account the design of the searches (how often and at what intervals the searches have been performed) to get an estimate of the

probability that a carcass that has fallen into the search area will be found. This has been done in different ways by different authors (Erickson et al. 2004, Shoenfeld 2004, Kerns et al. 2005, Jain et al. 2007, Pollock 2007, Huso 2010, Korner-Nievergelt et al. 2011, Etterson 2013, Péron and Hines 2014). The R-package *carcass* provides the methods presented by Korner-Nievergelt et al. (2011) and generalizations of these methods (Etterson 2013) that allow relaxing some of the assumptions made by the original method and Eq. 14 from Etterson (2013). *Carcass* also includes the method of Huso (2010). We do not give the theoretical background and do not test these methods via simulations here because this has been done in the specific studies.

There is no standard method that can be used in every case. Whether a method produces biased estimates or not depends on how well the data meet the implicit assumptions of the method chosen. Data sets can differ in a variety of aspects, such as number of carcasses found, length and regularity of search intervals, carcass persistence probability and its variability over time, searcher efficiency and its variability over time, design of the experiments to assess searcher efficiency and persistence probability, topography, spatial distribution of carcasses in relation to the search area etc. Such differences in methodology and experimental conditions induce that a statistical method that performs well in one study may produce a serious bias in another. *Carcass* is a collection of tools that help to individually design the analyses of carcass search data. To help to pick the most appropriate tools, we expanded the overview over the currently used methods from the review of Bernardino et al. (2013) and give some guidelines for when to choose which method below.

In addition to methods for the estimation of the detection probability  $p$  (that a killed animal is eventually found), *carcass* also provides functions to estimate the number of animals that have died ( $N$ ) given an estimate for  $p$  and the number of carcasses found  $c$ . To do so, we use Bayes' theorem. *Carcass* provides two functions for the estimation of  $N$ : 1) `posteriorN` calculates the posterior distribution of the number of fatalities given a detection probability estimate without taking into account the precision of the estimated detection probability. 2) `estimateN` does the same as `posteriorN` but accounts for uncertainty of the estimate of detection probability, using Monte Carlo simulations.

The methods provided by *carcass* allow users to estimate the total number of animals that have died, that is, summed over a specific (long) time span or spatial range assuming that mortality rate is constant. This is because the estimation of mortality based on a low number of carcasses, e.g. zero or one, is associated with very high stochastic uncertainty (see Discussion). Therefore, if more detailed estimates for the mortality rate are needed, e.g. daily or weekly mortality rates for single sites rather than the total number of animals killed over a long time period, we either have to collect the carcasses using a mark-recapture study design (Péron et al. 2013), or we could include additional information about the number of animals exposed to the hazard and use model based approaches as has been done by Bellebaum et al. (2013) and Korner-Nievergelt et al. (2013). The R-package *fatalityCMR* (Péron and Hines 2014) provides a model based approach to obtain mortality estimates.

To work with *carcass* basic knowledge of the R-language is required. If you are unfamiliar with the R-language, you may check out the internet tools <[www.kollisionsopfersuche.uni-hannover.de](http://www.kollisionsopfersuche.uni-hannover.de)> (in German; requires that the study design exactly follows Niermann et al. 2011), or <[www.bio3.pt/en/services-and-projects/wildlife-fatality-estimator](http://www.bio3.pt/en/services-and-projects/wildlife-fatality-estimator)>, or the software provided by the USGS (Huso et al. 2012, Dalthorp et al. 2014).

## How is carcass used?

### Principal process

The following steps are necessary to estimate the number of fatalities from carcass searches.

- 1) Estimate the proportion of killed or injured animals that fall into the search area ( $a$ ).
- 2) Estimate carcass persistence probability ( $s$ ), if necessary for different ages of the carcass and different vegetation classes.
- 3) Estimate the probability that a carcass is found by an observer given it persists in the search area to the time of a search (searcher efficiency  $f$ ), if necessary dependent on vegetation classes and dependent on the number of searches a carcass has already been exposed to.
- 4) Combine the parameters  $a$ ,  $s$  and  $f$  with the search interval ( $d$ ) or search schedule ( $J$ ) to get the probability that an animal that has died is found (detection probability  $p = f(a, s, f, d)$ ).
- 5) Estimate the number of animals that have died  $N$  from the number of carcasses found ( $c$ ) and the estimate for detection probability ( $p$ ).

### Load the package

```
library(carcass)
```

### Proportion of killed or injured animals that fall into the search area

The proportion of killed or injured animals that fall into the search area (and do not move away) is determined by the size and distribution of the area that can be searched, the spatial distribution of the carcasses, and the proportion of injured animals that move away from the study area before they die. If you can assume that all animals that have been killed have fallen into the area that has been searched (and that no additional carcasses dead from another cause lay in the search area), you can skip this chapter and use  $a = 1$  later. However, often only parts of the area where carcasses fall into can be searched due to inaccessibility, time and funding constraints, or because of dangers such as traffic. In many studies, the proportion of carcasses that have fallen into the search area is far lower than the proportion of carcasses removed by scavengers prior to the search or the proportion of carcasses overlooked by an observer (Warren-Hicks et al. 2013, Huso and Dalthorp 2014). Therefore,  $a$  is often among the most important parameters in terms of its influence on the resulting estimate of numbers of animals killed.

To obtain  $a$  the spatial distribution of carcasses has to be combined with the spatial distribution of the search area. In many studies, carcass density decreases with distance to the obstacle (the cause of the hazard). Carcass density may,

therefore, be approximated by a function of distance from the obstacle. The density function of distance to the obstacle can be different in different directions depending on the shape of the obstacle, its height, prevailing wind directions and other factors. *Carcass* does not provide a general method that allows estimating  $a$ . Here, we only present an example of bat collisions at wind turbines to illustrate the principle of calculating  $a$  using information about the spatial distribution of carcasses from literature.

The data frame *batdist* contains examples of empirical and theoretical carcass distributions of bats at wind turbines (Table 1). The expected proportions of bats in each 10 m distance ring are given. Hull and Muir (2010) use a ballistic model to describe the spatial distribution below three different sizes of wind turbines for different carcass types and sizes, but the model does not seem to fit empirical data well (Arnett et al. 2005, Niermann et al. 2011, Huso and Dalthorp 2014). In addition, variation among sites and years may be substantial (Table 1). An empirical estimate of carcass distribution may be preferred (Dalthorp and Huso 2013).

To obtain the proportion of killed or injured animals that fall into the search area ( $a$ ), we can, e.g. multiply the proportion that was searched in each of the 10 m distance rings with the proportion of carcasses found in the corresponding ring and sum over all rings. As an example, we use the bat distribution averaged over the first three studies in *batdist* and multiply these proportions with the proportions of the search area per ring. The latter proportions may be obtained using a geographic information system (GIS). When sites are heterogeneous regarding visibility class, it is preferable to use the proportions of carcasses falling into the different sub-units (e.g. distance rings) as weights when averaging detection probability  $p$  or any other parameter, such as  $f$  or  $s$ , over the whole site.

```
# load data
data(batdist)

# proportion of carcasses per 10 m distance ring
sel <- c("prop10m", "prop20m", "prop30m", "prop40m", "prop50m")
propcarc.usnc <- apply(batdist[1:3, sel], 2, mean)
propcarcasses <- propcarc.usnc/sum(propcarc.usnc)

# proportion of searched area per 10 m distance ring
searchable.area <- c(0.6, 0.8, 0.75, 0.5, 0.5)

# proportion of carcasses that have fallen into the search area
a <- sum(propcarcasses * searchable.area)
a
[1] 0.6687713
```

If the study species may move away from the wind turbine after collision, but before dying, we have to multiply the proportion of carcasses lying in the search area (obtained as described above) by 1 minus the proportion of animals that move out of the search area. For red kites *Milvus milvus* the

proportion of individuals that move away from the turbine after collision may be up to 15 % (Bellebaum et al. 2013). We are not aware of estimates for other taxa.

### Carcass persistence time

Carcass persistence time is the duration a carcass stays on the ground before it is removed by scavengers or before it has decomposed (to a state that makes it essentially undetectable by the observers). Carcass persistence time depends primarily on the density and activity of scavengers, temperature and humidity. Thus, persistence time likely varies between sites. *carcass* allows the estimation of carcass persistence probabilities for multiple study sites simultaneously.

To estimate carcass persistence time, individual carcasses have to be monitored over time to record the time until they disappear. Such time-to-event data are commonly analysed using survival models to estimate (daily) persistence probability.

Survival models differ in their assumption about how removal probability changes over time. Several studies found that persistence probability of carcasses increases over time, that is, fresh carcasses are removed with a higher probability than old carcasses (Van Pelt and Piatt 1995, Erickson et al. 2004, Ward et al. 2006, Warren-Hicks et al. 2013). Bispo et al. (2013) explore the fit of different parametric survival models to carcass removal data.

At present, two different models to estimate carcass persistence time are implemented in *carcass*:

- 1) the exponential model that assumes constant persistence probability over time,
- 2) the Cox proportional hazard model that allows for a changing persistence probability over time without making any assumption about the shape of the change.

But other survival models may be preferred (Bispo et al. 2013). From any survival analyses, either daily persistence probabilities or the proportion of carcasses that persist until every day after death ('survivor function') can later be used to estimate carcass detection probability.

Here, we first estimate daily carcass persistence probability  $s$  assuming a constant persistence probability. Second, we estimate separate daily persistence probabilities for each day after death. For both analyses, we will use the data set 'persistence'. It contains persistence times in days for 636 brown mice and a few bat carcasses from 30 wind turbines. For each carcass, we know the persistence time in days (variable 'perstime') if the variable 'status' is 1, or we know the minimal persistence time if 'status' is 0 for carcasses that have not

Table 1. Estimated spatial distribution of bat carcasses below wind turbines in different studies (1 = Niermann et al. 2011, 2a,b = Arnett et al. 2005, 3 = Hull and Muir 2010). Rotor = rotor diameter in m, Nacelle = nacelle height in m.

Study	Rotor	Nacelle	0–10m	10–20m	20–30m	30–40m	40–50m	> 50m
1	70	98	0.21	0.39	0.27	0.11	0.02	0.00
2a	72	80	0.10	0.26	0.28	0.26	0.07	0.03
2b	72	80	0.04	0.19	0.28	0.31	0.14	0.04
3	66	65	0.30	0.26	0.20	0.15	0.07	0.02
3	90	80	0.24	0.23	0.19	0.15	0.12	0.07
3	110	94	0.21	0.20	0.18	0.15	0.12	0.14

disappeared until the last search (i.e. right-censored scavenging events). For example, the first brown mouse tested at turbine 5 (see first line in example data) was present at day 1 and day 2 after its placement (at day 0) but disappeared between day 2 and 3, i.e. the disappearance was observed (status = 1) at day 3. Another mouse, on line 7, persisted until the last control on day 14. Therefore, its persistence time is 14 and the status is 0 ('disappearance not observed'). Note that the persistence times are not known exactly because a scavenging event can hardly be observed directly, but it can be recorded that the carcass has disappeared between the last and present control. The models presently implemented in *carcass* assume that status = 1 means that the disappearance is exactly known. Therefore, when the time between two controls is large, other methods (e.g. allowing for interval-censoring) may be preferred.

```
data(persistence)
head(persistence, 7) # print first 7 lines
  turbineID perstime status
1         5         3     1
2         6         1     1
3         5         1     1
4         6         2     1
5         5         2     1
6         6         1     1
7         5        14     0
```

When we assume a constant persistence probability, we fit an exponential survival model `persistence.prob` and obtain one estimate for persistence probability per turbine (or site). The argument `pers.const = TRUE` is used to tell R that we assume a constant persistence probability.

```
consts <- persistence.prob(persistence$turbineID,
  persistence$perstime, persistence$status, pers.const=TRUE)
head(consts)
  turbineID persistence.prob      lower      upper mean.persistence.time
1         1         0.8007374 0.6621167 0.8867505         4.500000
2         2         0.7083425 0.3599893 0.8866514         2.900000
3         3         0.8232919 0.5393822 0.9406873         5.142857
4         4         0.8539397 0.5978285 0.9530950         6.333333
5         5         0.7852460 0.5301777 0.9134652         4.136364
6         6         0.5857374 0.2523696 0.8154391         1.869565
```

The function returns a data frame with one row per site, the estimated daily persistence probability with the lower and upper limit of the 95% confidence interval, as well as the mean persistence time in days.

When we cannot assume a constant persistence probability, we may use the Cox proportional hazard model by setting the argument `pers.const = FALSE` (i.e. the default).

```
vars <- persistence.prob(persistence$turbineID,
  persistence$perstime, persistence$status, pers.const=FALSE)
str(vars)
list of 4
 $ time          : num [1:14] 1 2 3 4 5 6 7 8 9 10 ...
 $ persistence.prob: num [1:14, 1:30] 0.769 0.584 0.475 0.367 0.286 ...
 .. attr(*, "dimnames")=List of 2
 .. ..$ : chr [1:14] "1" "2" "3" "4" ...
 .. ..$ : chr [1:30] "1" "2" "3" "4" ...
 $ lower         : num [1:14, 1:30] 0.65 0.417 0.299 0.197 0.131 ...
 .. attr(*, "dimnames")=List of 2
 .. ..$ : chr [1:14] "1" "2" "3" "4" ...
 .. ..$ : chr [1:30] "1" "2" "3" "4" ...
 $ upper         : num [1:14, 1:30] 0.909 0.818 0.756 0.684 0.622 ...
 .. attr(*, "dimnames")=List of 2
 .. ..$ : chr [1:14] "1" "2" "3" "4" ...
 .. ..$ : chr [1:30] "1" "2" "3" "4" ...
```

In this case, the function returns a list with four elements. The first element ('time') contains the time points for which the Cox-regression has estimated the proportion of remaining carcasses. Each of the other three elements is a matrix where rows correspond to time (one row for each time point given in the 'time' element) and columns correspond to sites. The values are the estimates, the lower and the upper limits of the 95% confidence intervals for the probability that a carcass persists up to the time of the corresponding row.

Both models fit reasonably well to the data (Fig. 1), and the difference between the two models is small (obviously, this may not always be the case).

When data from several sites are available, the Cox proportional hazard model estimates the shape of the disappearance probability over time (hazard function) based on the pooled data set, that is, by assuming that the proportions of disappearance probabilities (= 1 - persistence probability) between the different sites is constant over time. Therefore, the results differ depending on whether separate analyses for each site or one analysis including all sites are done. If data for only one site is available the result from the Cox proportional hazard model equals the data (Kaplan-Meier curves in Fig. 1). Such a result may be reliable only if sample size is very large. Therefore, if data for only one site is available, or sample size is small, we recommend a parametric model such as an exponential or a Weibull model. The latter is not yet implemented in *carcass*. Santos et al. (2011) list carcass persistence times for different taxonomic groups. However, be aware that persistence time can strongly vary between locations and date.

The persistence probabilities estimated above are probabilities that a carcass persists for a given time interval, e.g. 24 hours. However, to estimate carcass detection probability, the relevant parameter is the proportion of persisting carcasses from all carcasses that arrived during a specific time interval. Thus, the estimates above can be used only if all carcasses arrive at the beginning of each time interval. Such



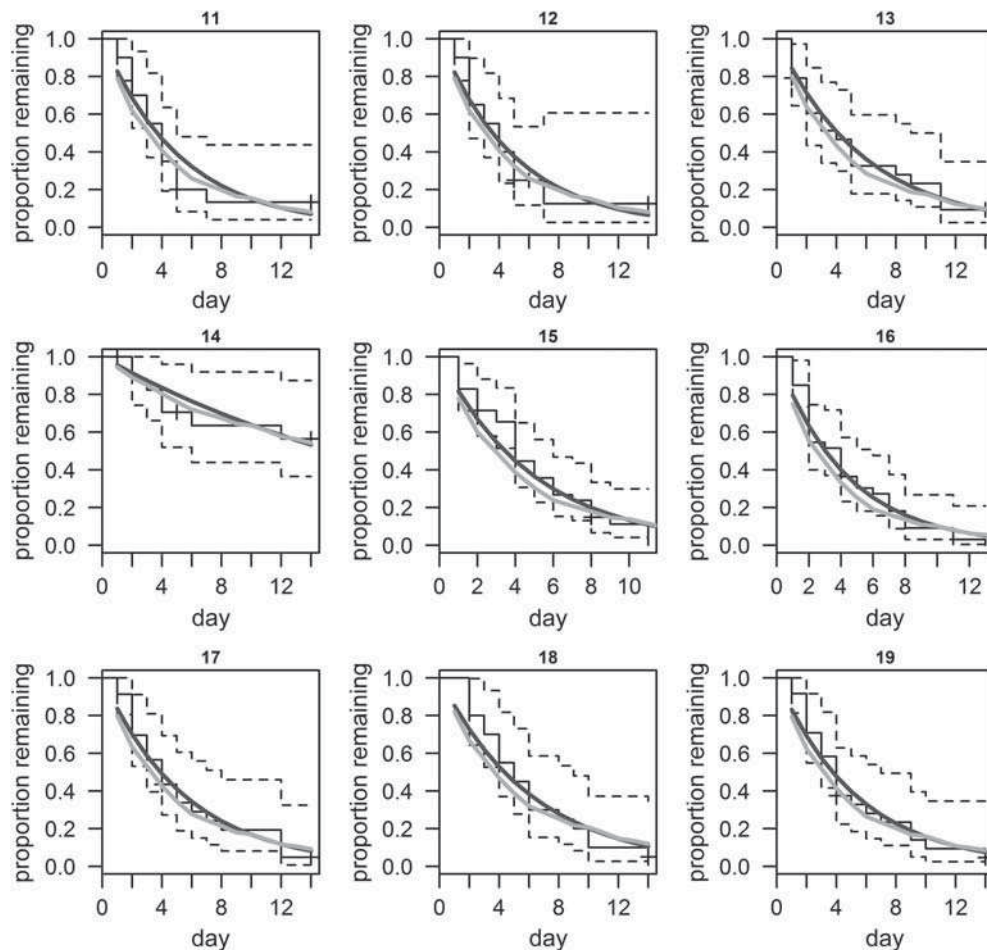


Figure 1. Proportion of persisting carcasses over time at nine different wind turbines (nr. 11 – 19). Black = data (Kaplan–Meier curve) with 95% confidence interval (dotted lines), dark grey = estimate from exponential model assuming constant persistence probability, light grey = estimate from Cox proportional hazard model allowing for a time dependent persistence probability.

an assumption may be realistic for bat carcasses since often the majority of the bat carcasses arrive in the hour after sunset (Niermann et al. 2014). However, in other situations, carcasses may arrive continuously. If we assume a constant arrival rate, the persistence probabilities can be integrated over the last time interval to obtain the proportion of carcasses persisting until the end of the time interval.

Searcher efficiency depends on experience of the observer, characteristics of the ground (particularly vegetation density influencing visibility), and type of object that is searched for. Furthermore, the average searcher efficiency often decreases with the number of searches the objects have been exposed to, because decaying carcasses are harder to find and because highly detectable objects will be found earlier than hidden

```
# probability to persist 1,2,...,14 days at turbine 1
round(vars$persistence.prob[,1], 2)
1 2 3 4 5 6 7 8 9 10 11 12 13 14
0.77 0.58 0.48 0.37 0.29 0.22 0.19 0.16 0.13 0.12 0.10 0.08 0.07 0.06

# proportion of persisting carcasses at days 1,2,...,14 when carcasses arrive
# continuously over the whole day at turbine 1
round(integrate.persistence(vars$persistence.prob[,1]), 2)
0.88 0.67 0.53 0.42 0.32 0.25 0.21 0.18 0.15 0.13 0.11 0.09 0.08 0.07
```

Note that this integration is done automatically in the function `estimateN` when the argument ‘arrival’ is set to ‘uniform’.

### Searcher efficiency

Searcher efficiency is the probability that an observer finds a carcass that is in the search area at the time of the search.

objects. For example objects that have fallen into a hole will remain longer on the search area (Huso 2010).

Usually, experimental trials are performed to assess searcher efficiency. In such trials, carcasses or carcass-like objects are distributed by a second person and the observer whose efficiency is being estimated must search for them.

Such trials yield data similar to the data in the data frame searches:

data(searches)				
searches				
	person	visibility	detected	notdetected
1	NameA	open	4	0
2	NameB	open	1	0
3	NameC	open	4	1
4	NameD	open	2	0
5	NameE	open	5	1
6	NameF	open	1	1
7	NameA	overgrown	2	0
8	NameB	overgrown	12	0
9	NameG	overgrown	1	0
10	NameH	overgrown	3	2
11	NameI	overgrown	0	1
12	NameE	overgrown	2	1
13	NameF	overgrown	1	1
14	NameB	heavily_overgrown	2	1
15	NameG	heavily_overgrown	4	0
16	NameI	heavily_overgrown	1	1
17	NameF	heavily_overgrown	7	2

The example data was collected by Niermann et al. (2011). It contains the number of detected and undetected items for each of nine observers (variable 'person') and three visibility classes of the ground. The function `search_efficiency` estimates searcher efficiency  $f$  for each person and visibility class, as well as for each visibility class averaged over all persons. The average may be useful, if trial data on searcher efficiency are missing for some persons in the study. The function `search_efficiency` uses the function `glmer` from the package *lme4* (Bates 2005) to fit a generalized linear mixed model with binomial error distribution and logit link function to model the number of detected and not detected items. From this model the fitted values (search efficiency) are extracted for each person and visibility class as well as for every visibility class averaged over all the persons. Visibility is used as a fixed effect and person as a random effect. When the data are limited to only one or two observers, observer is also treated as a fixed effect, and a generalised linear model is fitted by the function `glm` from the *stats*-package. Treating person as a random factor in the model has the advantage that for persons with a low number of experimental data (i.e. with weak information about their searcher efficiency) the estimate is shrunk towards the overall mean. In this way we avoid large over- or underestimation of person-specific searcher efficiencies that are due to the stochasticity of the experimental data (Gelman and Pardoe 2006).

To estimate searcher efficiency using `search_efficiency` one must provide the data (i.e. observer id, visibility for each trial, number of detected and not detected items) either bundled in a data frame or as separate vectors. If the function is applied to a data frame, this data frame must contain the following four variables with the exact names: 'person', 'visibility', 'detected', 'notdetected' (spelling and capitalization is important!). If data for only one visibility class is available (or visibility class has not been recorded), the variable 'visibility' has to contain the same value on each line. The function allows two (and only two) categorical explanatory variables (visibility class and observer id), thus it is not possible to include other explanatory variables or covariates.

The precision of the estimated search efficiency  $f^{\hat{}}$  is obtained using Bayesian methods, because these methods are more reliable in generalised linear mixed models than frequentist methods (Bolker et al. 2009). The function `sim` from the package *arm* is used to sample from the joint poste-

rior distribution of the model parameters (Gelman and Hill 2007). The 2.5% and 97.5% quantiles of the random draws from the posterior distributions are used as lower and upper limits of the 95% credible intervals. The number of random draws can be specified in the argument `nsim`. Default is 1000, but higher numbers are recommended for reliable credible intervals.

The models used by `search_efficiency` assume that the number of detected items is binomially distributed within each observer and visibility class. If the data are aggregated so that there is one row for each combination of observer and visibility class and visibility class has the same effect on searcher efficiency for each observer (no interaction between observer and visibility class), there is little danger of violating this assumption. However, if the observers have been tested in different trials so that there are two or more rows per observer and visibility class, the variance in the data may be higher than assumed by the binomial model (overdispersion). If overdispersion is likely to be present, the models have to be fitted 'by hand' using the functions `glm` or `glmer` and checked for overdispersion as, e.g. described in Gelman and Hill (2007).

In real data, different sub-units (areas) will have different visibility classes with different corresponding searcher efficiencies and, often, also different persistence probabilities (Warren-Hicks et al. 2013). In such cases, it may be best to calculate the detection probability  $p^*_k$  for every sub-unit ( $1, \dots, K$ ) separately. The weighted average of the detection probabilities  $p^*_k$  with weights equal to the proportion of carcasses falling into the different sub-units ( $a_k$ ) can serve as an overall detection probability  $p$ . Dalthorp et al. (2014) explain how to calculate the variance for this weighted average. For simplicity, we here assume that only one visibility class was present.

### **Combining the proportion of carcasses in the search area $a$ , persistence probability $s$ and searcher efficiency $f$ to get the detection probability $p$**

The probability that an animal that has died is eventually found by an observer ( $p$ ) is related to the proportion of carcasses that have fallen into the search area ( $a$ ), the persistence time ( $s$ ) and the searcher efficiency ( $f$ ). These three parameters have been combined in various different ways to get  $p$ . Table 2 gives an overview over the commonly used methods.

First, some methods assume that the searches have been done regularly (constant search interval). If the searches have been done at irregular time intervals, an average search interval can be used. However, if the search interval is large compared to the carcass persistence time, mortality will be underestimated because during longer than average search intervals some carcasses will disappear without having the chance of being detected during any search. In such cases, it is prudent to use a method that allows for irregular search intervals.

Second, many studies on carcasses found that fresh carcasses are removed by scavengers with higher probability than old carcasses (Van Pelt and Piatt 1995, Ward et al. 2006, Bispo et al. 2013, Warren-Hicks et al. 2013). Therefore, it is better to use methods that allow for age-dependent persistence probabilities, if sample size is sufficient to estimate age-dependent persistence probabilities. It may not be important to account for changing persistence times with carcass age when search intervals are short compared to persistence

```

search.f <- search. efficiency(searches)
search.f
$f.perperson
  visibility person      f      lwr      upr
1 heavily_overgrown NameA 0.8185871 0.5088642 0.9523684
2              open  NameA 0.8770466 0.6264570 0.9674160
3              overgrown NameA 0.8275182 0.5628363 0.9481587
4 heavily_overgrown NameB 0.8365675 0.5530922 0.9532575
5              open  NameB 0.8900113 0.6393168 0.9720503
6              overgrown NameB 0.8447812 0.6386728 0.9436316
7 heavily_overgrown NameC 0.7835797 0.4663199 0.9421976
8              open  NameC 0.8512704 0.5825056 0.9592465
9              overgrown NameC 0.7938000 0.4934589 0.9374071
10 heavily_overgrown NameD 0.8010652 0.4714172 0.9478942
11              open  NameD 0.8642345 0.5838319 0.9662468
12              overgrown NameD 0.8106593 0.5117921 0.9451198
13 heavily_overgrown NameE 0.7771807 0.4464743 0.9381668
14              open  NameE 0.8464808 0.5755811 0.9559271
15              overgrown NameE 0.7876214 0.5085431 0.9288058
16 heavily_overgrown NameF 0.7567780 0.4806568 0.9099537
17              open  NameF 0.8310441 0.5259102 0.9533614
18              overgrown NameF 0.7678891 0.4795424 0.9195665
19 heavily_overgrown NameG 0.8201803 0.5384621 0.9479579
20              open  NameG 0.8782027 0.6079234 0.9708404
21              overgrown NameG 0.8290493 0.5618396 0.9473118
22 heavily_overgrown NameH 0.7620694 0.4130943 0.9329548
23              open  NameH 0.8350719 0.5288516 0.9565154
24              overgrown NameH 0.7730113 0.5021893 0.9199497
25 heavily_overgrown NameI 0.7475718 0.4156456 0.9233062
26              open  NameI 0.8239952 0.4925924 0.9549342
27              overgrown NameI 0.7589695 0.4479066 0.9220849

$f.average
  visibility      f      lwr      upr
1 heavily_overgrown 0.7922068 0.5292800 0.9293065
2              open  0.8576893 0.6193702 0.9564306
3              overgrown 0.8021226 0.5749819 0.9209872

```

```

f <- search.f$f.average$f[1]
f.lower <- search.f$f.average$lwr[1]
f.upper <- search.f$f.average$upr[1]

```

times and searcher efficiency is high. In some studies change in persistence time with calendar date may be more important than with carcass age, e.g. when temperature or humidity substantially changed or scavengers arrived. Then the formula developed by Etterson (2013) will be a good choice.

Third, some methods do not take into account that carcasses that have been overlooked during one search can be found during subsequent searches ('bleed-through'). Therefore, for data sets with long persistence times (relative to the search interval) and low searcher efficiency (where a high proportion of carcasses can be expected to be overlooked during the first search and found during subsequent searches) it is recommended to use a method that accounts for this situation (which may often be the case, thus bleed-through should mostly be accounted for) such as Kerns et al. (2005), Korner-Nievergelt et al. (2011) or Etterson (2013).

Fourth, if searcher efficiency is likely to change with the number of searches a carcass was exposed to or with its age the choice should be one of the methods that allow for variable searcher efficiency. This may be particularly important

on heterogeneous terrain, where some carcasses fall into holes and others not, or when some animals are injured and seek cover prior to dying. In such cases, the average detectability of a carcass in a cohort decreases with the number of searches they were exposed to because easy to detect carcasses are found first. Carcass detectability may also decrease due to decay of the carcasses, i.e. dependent on carcass age, or change with calendar date, e.g. due to vegetation growth.

At present, the package *carcass* provides the following different methods to calculate  $p$ : Huso's estimator (Huso 2010), the formula we developed earlier (Korner-Nievergelt et al. 2011) including some extensions, and the recently published formulae by Etterson (2013). The methods currently provided by *carcass* are summarized in Table 3. The formula presented by Huso (2010) and implemented in the software Huso et al. (2012) require a mean persistence time estimates from a parametric model such as an exponential or Weibull model. The formula Korner-Nievergelt et al. (2011) assumes constant persistence probability and constant or decreasing searcher efficiency. In the package *carcass* we extend the formulas published in Korner-Nievergelt et al. (2011) to allow for time variation (based on parametric or non-parametric models) of persistence probabilities (Supplementary material Appendix 1). The formulas by Etterson (2013) are based on the same model as the one in Korner-Nievergelt et al. (2011).

Table 2. Overview over the most commonly used methods to estimate mortality from carcass searches (using unmarked carcasses). For every method, assumptions regarding the regularity of the search interval, the carcass persistence probability, and searcher efficiency are given. The 5th column indicates whether the method accounts for carcasses that are not found during one search but found in subsequent searches (termed 'bleed-through' by Warren-Hicks 2013). The last column names software with the specific method implemented. The first part of the table ("Formulas") contains methods that first estimate the probability that a killed animal is found and, subsequently, use the total number of carcasses found together with the estimated detection probability to get a mortality estimate, e.g. a Horvitz-Thompson estimate or a posterior distribution using Bayes' theorem. The second part of the tables ("Models") contains statistical models that are fitted to the raw search data (number of carcasses found for every single search) taking detection probability into account. The table is adapted from Bernardino et al. (2013).

Estimator	Search interval	Persistence probability	Searcher efficiency	Carcass not found during a search can be found in a subsequent search	Software
<b>Formulas</b>					
Shoenfeld 2004 (Eq. 4/5) = Erickson et al. 2004 (p. 8)	search intervals are Poisson-distributed with mean search interval $\bar{d}$	removal times follow a Poisson or an Exponential distribution	?	yes but assumes 100% bleed-through (Warren-Hicks 2013)	- R-package carcass
Kerns et al. 2005 (Eq. 6/7)	regular	?	constant	yes	
Jain et al. 2007 (Eq. 8)	no explicit assumptions	constant	?	no	- Wildlife Fatality Estimator (<www.bio3.pt>)
Pollock 2007 (Eq. 8/9)	regular	number of verifications before removal follows a geometric model	?	no	
Huso 2010 (Eq. 10)	no explicit assumption	one of several potential parametric survival models: Weibull, loglogistic, lognormal, exponential	constant	no	- Huso et al. (2012) (<http://pubs.er.usgs.gov/publication/ds729>) - R-package carcass - Wildlife Fatality Estimator (<www.bio3.pt>)
Huso et al. 2012	regular or irregular	constant or depending on carcass age	constant or time variable	no	- Huso et al. (2012)
Korner-Nievergelt et al. 2011 (Eq. 11/12)	regular	constant	constant or depending on the number of searches per carcass (i.e. carcass age)	yes	- R-package carcass - Wildlife Fatality Estimator (<www.bio3.pt>) - <www.kollision-sopfersuche.uni-hannover.de/>
this article	regular	constant or depending on carcass age (any survival function)	constant or decreasing exponentially with the number of searches per carcass (i.e. carcass age)	yes	- R-package carcass
Etterson 2013	regular or irregular	constant or dependent on carcass age or dependent on calendar date	constant or dependent on number of searches per carcass or dependent on calendar date	yes	- R and Matlab code in electronic supplement - R-package carcass
Dalthorp et al. 2014	regular or irregular	one of several potential parametric survival models: Weibull, loglogistic, lognormal, exponential	constant or depending on carcass age	Yes	<http://pubs.er.usgs.gov/publication/ds881>
<b>Models</b>					
Korner-Nievergelt et al. 2013	regular or irregular	constant*	constant*	yes	- Bugs code in electronic supplement
Péron et al. 2013 Péron and Hines 2014	regular or irregular (but for all sites the same sampling protocol has to be used)	constant or different between two classes (e.g. 'fresh' and 'old') of the carcasses	constant or different between age classes (e.g. 'fresh' and 'old') of the carcasses	yes	- R-package fatality CMR
Etterson 2013	regular or irregular requires double searches and marking of carcasses	constant or time dependent*	constant or depending on carcass age*	yes	- R and Matlab code in electronic supplement
Wolpert in Warren-Hicks et al. 2013	regular or irregular	Weibull (including exponential)	constant or depending on carcass age and number of searches per carcass	yes	- R-package in preparation

\*The model code can be adapted to allow for any survival function or for any relationship of searcher efficiency with carcass age or the number of searches a carcass has experienced. The model based approach allows including covariates for any model parameters.



Table 3. Overview of the functions currently provided by carcass for calculating the carcass detection probability  $p$  from persistence probability  $s$ , searcher efficiency  $f$ , and optional additional information such as search interval  $d$  or a vector of lengths of search intervals of different lengths  $J$ . All functions, except `phuso` account for bleed-through (Table 2).

Search interval	Searcher efficiency	Persistence probability	Function in carcass
Regular	constant	constant	<code>phuso</code> <code>pkorner</code> <code>ettersonEq14</code>
Regular	decreasing with number of searches	constant	<code>phuso</code> <code>pkorner</code> <code>ettersonEqv2</code>
Regular	constant	dependent on age of carcass (input: 'survivor curve')	<code>pkorner</code>
Regular	decreasing with number of searches	dependent on age of carcass (input: 'survivor curve')	<code>pkorner</code>
Irregular	constant	constant	<code>ettersonEq14</code>
Irregular	constant or dependent on calendar date	constant or dependent on calendar date	<code>ettersonEqv1</code>
Irregular	constant or dependent on the number of searches	constant or dependent on age of carcass (input: daily persistence probability)	<code>ettersonEqv2</code>

A mathematical proof that Eq. 15 of Etterson (2013) is equivalent to the formula Korner-Nievergelt et al. (2013) is given in Supplementary material Appendix 2. Etterson (2013) provided a generalization of this formula that allows for irregular search intervals. He further provides two different variants of time dependence in searcher efficiency and persistence probability (on carcass age or calendar date).

Detection probability  $p$  is a function  $f(s, f, a, d, n)$ , where  $s$  is the persistence probability,  $f$  the searcher efficiency,  $a$  the proportion of carcasses that have fallen into the search area,  $d$  the search interval (usually in days), and  $n$  the number of searches. In case of irregular search intervals, a vector with the length of the search intervals  $J$  is used instead of  $d$  and  $n$ . The factor  $a$  affects  $p$  in a simple multiplicative way, therefore, we can write  $p^* = f(s, f, d, n)$ , or  $p^* = f(s, f, J)$  respectively, and  $p = ap^*$ .

#### Calculation of $p^*$ assuming constant persistence and constant searcher efficiency

Let's assume that searcher efficiency  $f$  and persistence probability  $s$  are constant. Then we use either of the functions `phuso`, or `pkorner` for regular search intervals or `ettersonEq14` for irregular search intervals.

The function `phuso` is based on the average persistence time ( $\hat{i}$ ) which is  $1/-\log(s)$  in the case of a constant persistence probability. We can either provide  $s$  or  $\hat{i}$ . The search interval  $d$  has to be provided in addition to  $s$  and  $f$ . For `pkorner` we have to provide the total number of searches  $n$  in addition ( $dn$  is then the length of the study period), and `ettersonEq14` requires a vector with the lengths of the search intervals.

```
s <- consts$persistence.prob[1]
s.lower <- consts$lower[1]
s.upper <- consts$upper[1]
ph <- phuso(s=s, f=f, d=2)
pk <- pkorner(s=s, f=f, d=2, n=100)
J <- rep(2, 100)
pet <- ettersonEq14(s=s, f=f, J=J)
ph; pk; pet

[1] 0.6395835
[1] 0.6579283
      [,1]
[1,] 0.6579283
```

The function `pkorner` provides 95% confidence intervals of the detection probability if the argument `CI` is set to `TRUE` and lower and upper limits of  $s$  and  $f$  are given. These confidence intervals are obtained by Monte Carlo simulations.

```
pkci <- pkorner(s=s, s.lower=s.lower,
s.upper=s.upper, f=f, f.lower=f.lower,
f.upper=f.upper, d=2, n=100, CI=TRUE)
pkci
      p      2.5%      97.5%
0.6579533 0.4700964 0.8136018
```

For the result of `ettersonEq14` a Monte Carlo confidence interval can be calculated by the function `CIetterson`.

```
CIetterson(s=s, s.lwr=s.lower,
s.upr=s.upper, f=f, f.lwr=f.lower,
f.upr=f.upper, J=J)
$p.lower
      2.5%
0.473692

$p.upper
      97.5%
0.817388
```

The function `ettersonEq14` can handle irregular search intervals.

```
J <- c(3,5,4,2,1,4,4,5,6,2,3,4,5,3,2)
ettersonEq14(s=s, f=f, J=J)

      [,1]
[1,] 0.5192987
```

Note that when defining a constant  $s$ , it is assumed that carcasses arrive only at the beginning of each time interval. To account for continuous carcass arrival, the survivor function (proportion of persisting carcasses for each day after death) has to be integrated over each time interval and the formulas allowing for time variable persistence probabilities used (use 'sca' instead of 'vars' below).

```
# proportion of persisting carcasses when carcasses arrive continuously
sca <- integrate.persistence(s^c(1:20))
sca
[1] 0.897 0.718 0.575 0.460 0.369 0.295 0.236 0.189 0.152 0.121 0.097 0.078
[13] 0.062 0.050 0.040 0.032 0.026 0.021 0.016 0.013
```

#### Calculation of $p^*$ assuming constant persistence probability and time-variable searcher efficiency

If the detectability differs between the carcasses, e.g. because some carcasses have fallen into a hole whereas others are exposed, the average detectability of a cohort of carcasses will decrease with the number of searches they were exposed to. This is because easily detectable carcasses will be found during earlier searches whereas hard to detect carcasses remain on the ground. Huso (2010) suggests that the decrease in average detectability with each search may be as high as 75%. The function `pkorner` permits modelling such a decrease by setting the argument `search.efficiency.constant=FALSE`. The argument `k` allows one to specify the factor by which the searcher efficiency  $f$  (of a cohort of carcasses) is multiplied at each search (that the cohort is exposed to). The mathematical derivation of this formula is given in the Supplementary material Appendix 1.

If the argument `CI` is set to `TRUE`, a 95% confidence interval is calculated using Monte Carlo simulations.

```
pk <- pkorner(s=s, f=f, d=2, n=100, search.efficiency.constant=FALSE, k=0.25)
pk
[1] 0.5928075

pkci <- pkorner(s=s, s.lower=s.lower, s.upper=s.upper, f=f, f.lower=f.lower,
f.upper=f.upper, d=2, n=100, search.efficiency.constant=FALSE, k=0.25,
CI=TRUE, nsim=100)
pkci # a higher number of Monte Carlo simulations (nsim) is recommended
      p          2.5%        97.5%
0.5928075 0.404781 0.7644495
```

The function `ettersonEq14v2` allows specifying different searcher efficiencies for all possible number of searches a carcass has been exposed to. For example, when detectability of a carcass is high during the first search and then drops markedly, because the easy to detect carcasses are found during the first search but later the change in average detectability between two searches may vanish, this pattern can be specified by a vector that contains the searcher efficiency for every search separately. Note that persistence probability needs to be specified for every possible age of a carcass in the study.

Searcher efficiency can also decrease with age of the carcass. To account for age-dependent searcher efficiency is not yet implemented. However, in case of regular (and short) search intervals, carcass age corresponds to the number of searches a carcass has been exposed to, and `ettersonEq14` or `pkorner` may be applied. The R-package `carcass-`

`CMR` (Péron and Hines 2014) can estimate searcher efficiency dependent on two age classes (old versus fresh). The R-package that is currently under development by R. Wolpert (Table 2) will be able to account for searcher efficiency that depends on age and the number of searches a carcass has been exposed to.

```
fperdate <- c(0.4,0.4,0.4,0.4,0.9,0.9,0.9,0.9)
J <- c(3,2,5,2,4,4,5,2)
ettersonEq14v1(s=rep(s, sum(J)), f=fperdate, J=J)
      [,1]
[1,] 0.4772668
```

The function `ettersonEq14v1` allows for searcher efficiency that depends on calendar date. For example, the searches started in high and dense crop where searcher efficiency was low and after half-time of the study the crop was cut so that searcher efficiency suddenly increased.

For the confidence intervals, use `CIetterson`.

The function `phuso` does not allow one to specify the factor of the decrease in searcher efficiency with each search, but it seems to be fairly robust with respect to a decrease in searcher efficiency (Huso 2010, Korner-Nievergelt et al. 2011), and the new software Huso (2012) and Dalthorp et al. (2014) also allows for time dependence in searcher efficiency.

#### Calculation of $p^*$ assuming time-variable persistence and constant searcher efficiency

The functions `pkorner` and `ettersonEq14v2` can deal with persistence probabilities that vary with the age of the carcass. `pkorner` takes the proportion of remaining carcasses for each day after death in a vector as input (i.e. the 'survivor curve'; output of `persistence.prob` with non-constant persistence), whereas `ettersonEq14v2` takes a vector with daily persistence probabilities for every possible

```
fpersearch <- c(0.9, 0.2, 0.18, 0.15, 0.15, 0.15, 0.15, 0.15)
J <- c(3,2,5,2,4,4,5,2)
ettersonEq14v2(s=rep(s, sum(J)), f=fpersearch, J=J)
      [,1]
[1,] 0.5542363
```

```

nonconstant.s <- vars$persistence.prob[,1]
nonconstant.s
      1      2      3      4      5      6      7
0.76865162 0.58393492 0.47539667 0.36678317 0.28594559 0.22125794 0.1912215
      8      9     10     11     12     13     14
0.16330202 0.13298724 0.12482560 0.10244159 0.08000630 0.07153021 0.0605884
# check whether proportion of remaining carcasses is estimated for every
consecutive day (whether there are missing values):
vars$time
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14
nonconstant.s.lower <- vars$lower[,1]
nonconstant.s.upper <- vars$upper[,1]

pk <- pkorner(s=nonconstant.s, f=f, d=2, n=100)
pk
[1] 0.6143254

pkci <- pkorner(s=nonconstant.s, s.lower=nonconstant.s.lower,
s.upper=nonconstant.s.upper, f=f, f.lower=f.lower, f.upper=f.upper, d=2, n=100,
CI=TRUE, nsim=100) # higher values for nsim are recommended
pkci
      p      2.5%      97.5%
0. 6143254 0.49007620.7278280

```

carcass age. Both functions cannot handle missing values, thus these need to be imputed.

If the argument CI is set to TRUE, a 95% confidence interval is calculated using Monte Carlo simulations.

To use the function `ettersonEq14v2`, e.g. because search intervals were irregular, the daily persistence probabilities instead of the survivor curve has to be given. Here, we extract the daily persistence probabilities from the survivor curve and impute the missing values for the carcasses with age higher than 14 days with the estimate for day 14.

```

J <- c(3,2,5,2,4,4,5,2)
dailys <- numeric(sum(J))
dailys[1] <- nonconstant.s[1] # day 1
for(i in 2:length(nonconstant.s)) dailys[i] <-
nonconstant.s[i]/nonconstant.s[i-1]
# use the last estimate for persistence probability for the missing values
dailys[(length(nonconstant.s)+1):sum(J)] <- dailys[length(nonconstant.s)]

ettersonEq14v2(s=dailys, f=rep(f,length(J)), J=J)
      [,1]
[1,] 0.4915348

```

If daily persistence probability depends on calendar date rather than carcass age, the function `ettersonEq14v1` is used.

#### **Calculation of $p^*$ assuming time-variable persistence and time-variable searcher efficiency**

We can allow for both decreasing searcher efficiency with the number of searches a carcass was exposed to and time-variable persistence for regular (`pkorner` or `ettersonEq14v2`) and irregular (`ettersonEq14v2`) search intervals.

Again, it is possible to specify `CI=TRUE` to obtain a 95% confidence interval. And, again, we recommend

using more than the 100 simulations ( $> 2000$ ) given in the example code.

```

J <- c(3,2,5,2,4,4,5,2)
# For example, after the first 15 days of the study, a fox detected the sources
of carcasses:
sperdate <- c(rep(0.8, 15), rep(0.4, 12))
ettersonEq14v1(s= sperdate, f=rep(f,length(J)), J=J)
      [,1]
[1,] 0.3352823

```

Also, for the last two estimates, confidence intervals can be calculated using the function `CIetterson`.

Or, when both persistence probability and searcher efficiency depend on calendar date:

```
pk <- pkorner(s=nonconstant.s, f=f, d=2, n=100, search.
efficiency.const=FALSE, k=0.25)
pk
[1] 0.5553802
pkci <- pkorner(s=nonconstant.s, s.lower=nonconstant.s.lower,
s.upper=nonconstant.s.upper, f=f, f.lower=f.lower, f.upper=f.upper, d=2,
n=100, search.
efficiency.constant=FALSE, k=0.25, CI=TRUE, nsim=100)
pkci
      p      2.5%      97.5%
0.5553802 0.4047535 0.6998506
```

```
J <- c(3,2,5,2,4,4,5,2)
ettersonEq14v2(s=dailys, f=fpersearch, J=J)
      [,1]
[1,] 0.5148988
```

```
ettersonEq14v1(s=sperdate, f=fperdate, J=J)
      [,1]
[1,] 0.2596395
```

And confidence intervals for the estimates obtained by the formulas of Etterson (2013) are calculated by `CIetterson`.

### Last important step to obtain $p$

Finally, we have to combine the probability that a carcass that has fallen into the search area is found by an observer  $p^*$  with the proportion of carcasses that have fallen into the search area  $a$  to obtain the probability that an animal that has died during the study period is found by an observer  $p$ . In the simple example used here with one visibility class only ( $p^*$  is equal for all sub-units of the search area),  $p$  is the product  $p = ap^*$ . When  $p^*$  differs between sub-units (e.g. due to different visibility classes),  $p$  is a weighted average of the sub-unit-specific  $p^*_k$  with weights equal to the proportions of carcasses within the different sub-units ( $a_k$ , see section ‘Searcher efficiency’).

### Estimate the number of animals that have died ( $N$ )

Assuming that  $a$  is constant over time and that no carcass has been present at day one of the study (which is the day of the first search minus the search interval  $d$ ), the probability  $p$  is used to estimate the number of animals that have died  $\hat{N}$  from the number of carcasses found ( $c$ ). The method implemented in `carcass` is using Bayes’ Theorem with a uniform prior distribution (Korner-Nievergelt et al. 2011, Niermann et al. 2011). Thereby, for any  $x$ , the probability that  $N$  equals  $x$  is given by:

$$P(N = x) = \binom{x}{c} \frac{(1-p)^{(x-c)}}{\sum_{i=c}^x \binom{i}{c} (1-p)^{(i-c)}}$$

If we assume that the detection probability  $p$  is known (without error), we use the function `posteriorN`. The function returns the probability for each possible value of  $N$  given  $c$  (argument `nf`, number found) and  $p$ , if the argument `dist = TRUE`. If `dist = FALSE`, the function returns the median of the posterior distribution (`expected`) and the 2.5% and 97.5% quantiles (95% credible interval). The Horvitz–Thompson (Horvitz and Thompson 1952) estimate

(`HT.estimate`) equals the number of carcasses found divided by  $p$ .

```
pk <- pkorner(s=s, f=f, d=2, n=100)
posteriorN(p=pk*a, nf=3, plot=FALSE, dist=FALSE)
$interval
[1] 3 16
$expected
[1] 8
$HT.estimate
[1] 6.818126
```

To take into account the error in the estimate of  $p$  we can use the function `estimateN`. This function uses Monte Carlo simulation to propagate the uncertainty of  $p$  to the estimate of  $N$ , as presented in the worked example in Korner-Nievergelt et al. (2011).

The message tells us what assumptions are made concerning the arrival of the carcasses and the time dependence of the carcass persistence probability and how we could change this if needed. Further, it plots the posterior distribution of the number of dead animals if the argument `plot` is set to `TRUE` (Fig. 2). The argument `pform` is one of ‘`etterson`’, ‘`huso`’ or ‘`korner`’. In the argument ‘`x`’ it is specified which posterior probabilities  $P(\text{number of animals killed} > x)$  are given in the output.

Thus, for turbine 1 with three carcasses found by the observers, we conclude based on the estimates for the proportion of carcasses that have fallen into the search area  $\hat{a} = 0.67$ , the daily persistence probability  $\hat{s} = 0.80$ , and the searcher efficiency  $\hat{f} = 0.79$  that the true number of fatalities must be around 7 with a 95% credible interval of 3–18, taking into account the uncertainties in the estimates  $\hat{s}$  and

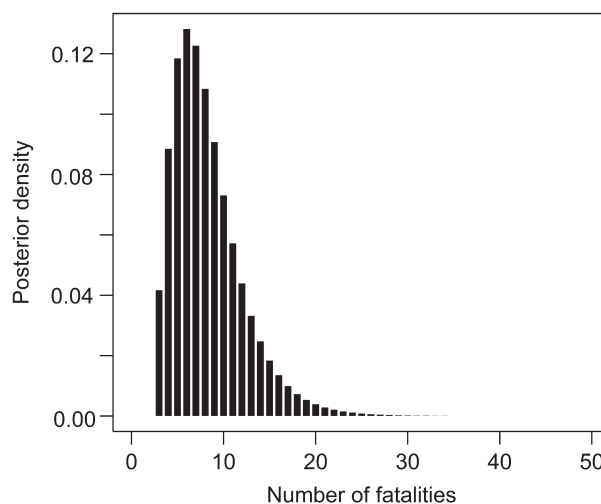


Figure 2. Posterior distribution of the number of fatalities.



```

estimateN(3, f = f, f.lower = f.lower,
          f.upper = f.upper,
          s = s, s.lower = s.lower, s.upper = s.upper, a = a,
          pform = "korner", d = 2, n = 100, plot=FALSE, x=c(5, 10, 20))
You assume that all carcasses arrive simultaneously
once a day right after the time the search takes place (if there is one)
and that the probability to persist from death to the first day
(the time the searches normally are performed) is equal to the probability
to persist from one day to the next. To change that,
use a different persistence probability for the first day
or choose arrival='uniform' to allow carcasses to arrive continuously.
$estimate
[1] 7

$lower
[1] 3

$upper
[1] 18

$HT.estimate
[1] 6.818126

$P.true.larger.x
      x=5      x=10      x=20
0.74998722 0.22799419 0.01075105

```

$\hat{f}$ . Or: there is only a small probability (of 0.01) that the true number of fatalities was larger than 20.

## Discussion

Unlike in approaches based on stochastic models, as provided in the R-package *fatalityCMR* (Péron et al. 2013) or presented in Etterson (2013) and Korner-Nievergelt et al. (2013), there is no formal method to assess whether the model assumptions are met or to do model selection, since the methods implemented in *carcass* are deterministic formulas rather than stochastic models that are fit to the data. No likelihood or other measure of model fit exists for such formulas. Therefore, it is important to decide whether time or age dependent persistence times or searcher efficiency are needed at the time when these parameters are estimated, e.g. by plotting the observed proportion of persisting carcasses over time together with the model estimates (as in Fig. 1). To assess whether searcher efficiency depends on carcass age, the search efficiency experiments have to be done with carcasses of different ages and its influence on searcher efficiency has to be analysed in separate analyses. Similarly, effects of calendar date need to be investigated for each single study.

Mortality estimates based on carcass searches can be very uncertain especially when the number of carcasses found and detection probability are low. In a simulation study, we showed that the relative error in the point estimate for mortality strongly increases when the number of carcasses found is lower than 10 (Korner-Nievergelt et al. 2011). A consequence of the high uncertainty of estimates for  $\hat{N}$  based on a low number of carcasses found is that sums of  $\hat{N}$  (e.g. over many turbines within a wind park) will overestimate the total number of fatalities for the whole wind park. In such a case, we recommend estimating the total number of fatalities based on the sum of the turbine-specific counts using an average detec-

tion probability rather than summing the single turbine-specific estimates  $\hat{N}$ , or use the posterior probability that the true number of fatalities does exceed a specific number instead of the point estimate. Alternatively, a fully model based approach could allow estimation of turbine specific fatality rates based on a data set pooled over several turbines (Bellebaum et al. 2013, Korner-Nievergelt et al. 2013, Péron et al. 2013).

The aim of the functions provided by *carcass* (listed in Table 3) is to provide easy to apply mortality estimators while taking the most important variance in carcass persistence time and searcher efficiency into account. As often, there is a tradeoff between simplicity and flexibility. The methods will perform well as long as the data meets the underlying assumptions, explained in Table 3 and the text. However, if the data shows a higher complexity, e.g. when persistence probability simultaneously depends on calendar date and age of the carcass, mortality estimates will be more precise when using more complex models such as done by Bellebaum et al. (2013), Etterson (2013), Korner-Nievergelt et al. (2013) or Péron et al. (2013). Nevertheless, *carcass* provides a variety of different methods that can be combined in a flexible way. The user can choose in each step of the analysis the most appropriate model for the specific study to reduce bias in the mortality estimation.

*Acknowledgements* – We sincerely thank Barbara Hellriegel for assistance with the mathematical part of the work. Stefanie Kramer and Guillaume Péron gave valuable comments on the manuscript. Any use of trade, firm, or product names is for descriptive purposes only and does not imply endorsement by the U.S. Government.

## References

Arnett, E. B. et al. 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. – Tech. Rep., Bat Conserv. Int.
- Bates, D. 2005. Fitting linear mixed models in R using the lme4 package. – R News 5: 27–30.
- Bellebaum, J. et al. 2013. Wind turbine fatalities approach a level of concern in a raptor population. – J. Nat. Conserv. 21: 394–400.
- Bernardino, J. et al. 2013. Estimating bird and bat fatality at wind farms: a practical overview of estimators, their assumptions and limitations. – N. Z. J. Zool. 40: 63–74.
- Bispo, R. et al. 2013. Modeling carcass removal time for avian mortality assessment in wind farms using survival analysis. – Environ. Ecol. Stat. 20: 147–165.
- Bolker, B. M. et al. 2009. Generalized linear mixed models: a practical guide for ecology and evolution. – Trends. Ecol. Evol 24: 127–135.
- Dalthorp, D. et al. 2014. Evidence of Absence software. Data Series. – US Geological Survey.
- Erickson, W. P. et al. 2004. Stateline wind project wildlife monitoring final report, July 2001 – December 2003. – Tech. Rep. peer-reviewed by and submitted to FPL Energy, the Oregon Energy Facility Siting Council, and the Stateline Technical Advisory Committee.
- Etterson, M. A. 2013. Hidden Markov models for estimating animal mortality from antropogenic hazards. – Ecol. Appl. 23: 1915–1925.
- Gelman, A. and Pardoe, I. 2006. Bayesian measures of explained variance and pooling in multilevel (hierarchical) models. – Technometrics 48: 241–251.
- Gelman, A. and Hill, J. 2007. Data analysis using regression and multilevel/hierarchical models. – Cambridge Univ. Press.
- Guinard, E. et al. 2012. Motorways and bird traffic casualties: carcass surveys and scavenging bias. – Biol. Conserv. 147: 41–51.
- Horvitz, D. G. and Thompson, D. J. 1952. A generalization of sampling without replacement from a finite universe. – J. Am. Stat. Ass. 47: 663–685.
- Hull, C. L. and Muir, S. 2010. Search area for monitoring bird and bat carcasses at wind farms using a Monte Carlo model. – Aust. J. Environ. Manage. 17: 77–87.
- Huso, M. M. P. 2010. An estimator of wildlife fatality from observed carcasses. – Environmetrics 22: 318–329.
- Huso, M. and Dalthorp, D. 2014. Accounting for unsearched areas in estimating wind turbine-caused fatality. – J. Wildl. Manage. 78: 347–358.
- Huso, M. et al. 2012. Fatality estimator users guide. – US Geological Survey Data Series 729.
- Jain, A. et al. 2007. Annual report for the Maple Ridge wind power project: post-construction bird and bat fatality study 2006. Final report. – L. Curry & Kerlinger. Syracuse, NY, Curry & Kerlinger, LLC: 53.
- Kerns, J. et al. 2005. Bat and bird fatality at wind energy facilities in Pennsylvania and West Virginia. – In: Arnett, E. B. (ed.), Relationships between bats and wind turbines in Pennsylvania and West Virginia: an assessment of fatality search protocols, patterns of fatality, and behavioural interactions with wind turbines. A final report prepared for the bats and wind energy cooperative. – Bat Conserv. Int., Austin, TX, USA, pp. 24–95.
- Korner-Nievergelt, F. et al. 2011. A new method to determine bird and bat fatality at wind energy turbines. – J. Wildl. Biol. 17: 350–363.
- Korner-Nievergelt, F. et al. 2013. Estimating bat and bird mortality occurring at wind energy turbines from covariates and carcass searches using mixture models. – PloS ONE 8:e67997.
- Kunz, T. H. et al. 2007. Ecological impacts of wind energy development on bats: questions, research needs and hypotheses. – Front. Ecol. Environ. 5: 315–324.
- Lebreton, J.-D. et al. 1992. Modelling survival and testing biological hypotheses using marked animals: a unified approach with case studies. – Ecol. Monogr. 62: 67–118.
- Niermann, I. et al. 2011. Systematische Schlagopfersuche - Methodische Rahmenbedingungen, statistische und Ergebnisse. Cuvillier Verlag, Göttingen, pp. 40–115, in German with English summary.
- Niermann, I. et al. 2014. Kollisionsopfersuchen als Grundlage zur Überprüfung der Wirksamkeit von Abschaltalgorithmen. – In: Reduktion des Kollisionsrisikos von Fledermäusen an Onshore-Windenergieanlagen - Endbericht des Forschungsvorhabens gefördert durch das Bundesministerium für Umwelt, Naturschutz und Reaktorsicherheit (Förderkennzeichen 0327638C+D). – O. Behr et al. Erlangen/Hannover/ Freiburg.
- Péron, G. and Hines, J. E. 2014. fatalityCMR – Software to estimate fatalities at wind-farms. – USGS-PWRC. <[www.mbr-pwrc.usgs.gov/software/fatalityCMR.html](http://www.mbr-pwrc.usgs.gov/software/fatalityCMR.html)>.
- Péron, G. et al. 2013. Estimation of bird and bat mortality at wind-power farms with superpopulation models. – J. Appl. Ecol. 50: 902–911.
- Pollock, R. 2007. Recommended formulas for adjusting fatality rates. California guidelines for reducing impacts to birds and bats from wind energy development. Final commission report. C. E. C. a. C. D. o. F. a. Game. – California Energy Commission, Renewables Committee, and Energy Facilities Siting Divisions, and California Dept of Fish and Game, Resources Management and Policy Division, pp. 117–118.
- Santos, S. M. et al. 2011. How long do the dead survive on the road? Carcass persistence probability and implications for road-kill monitoring surveys. – PloS ONE 6:e25383.
- Schaub, M. and Lebreton, J.-D. 2004. Testing the additive versus the compensatory hypothesis of mortality from ring recovery data using a random effects model. – Anim. Biodiv. Conserv. 27: 73–85.
- Shoenfeld, P. 2004. Suggestions regarding avian mortality extrapolation. – Technical memo provided to FPL Energy. – West Virginia Highlands Conservancy, Davis, WV, p. 6.
- Van Pelt, T. and Piatt, J. 1995. Deposition and persistence of beachcast seabird carcasses. – Mar. Pollut. Bull. 30: 794–802.
- Ward, M. et al. 2006. Wild bird mortality and west Nile virus surveillance: biases associated with detection, reporting, and carcass persistence. – J. Wildl. Diseases 42: 92–106.
- Warren-Hicks, W. et al. 2013. Improving methods for estimating fatality of birds and bats at wind energy facilities. Public Interest Energy Research (PIER) Program, Final project report. – California Energy Commission. Berkeley, CA, California Wind Energy Association: p. 136.