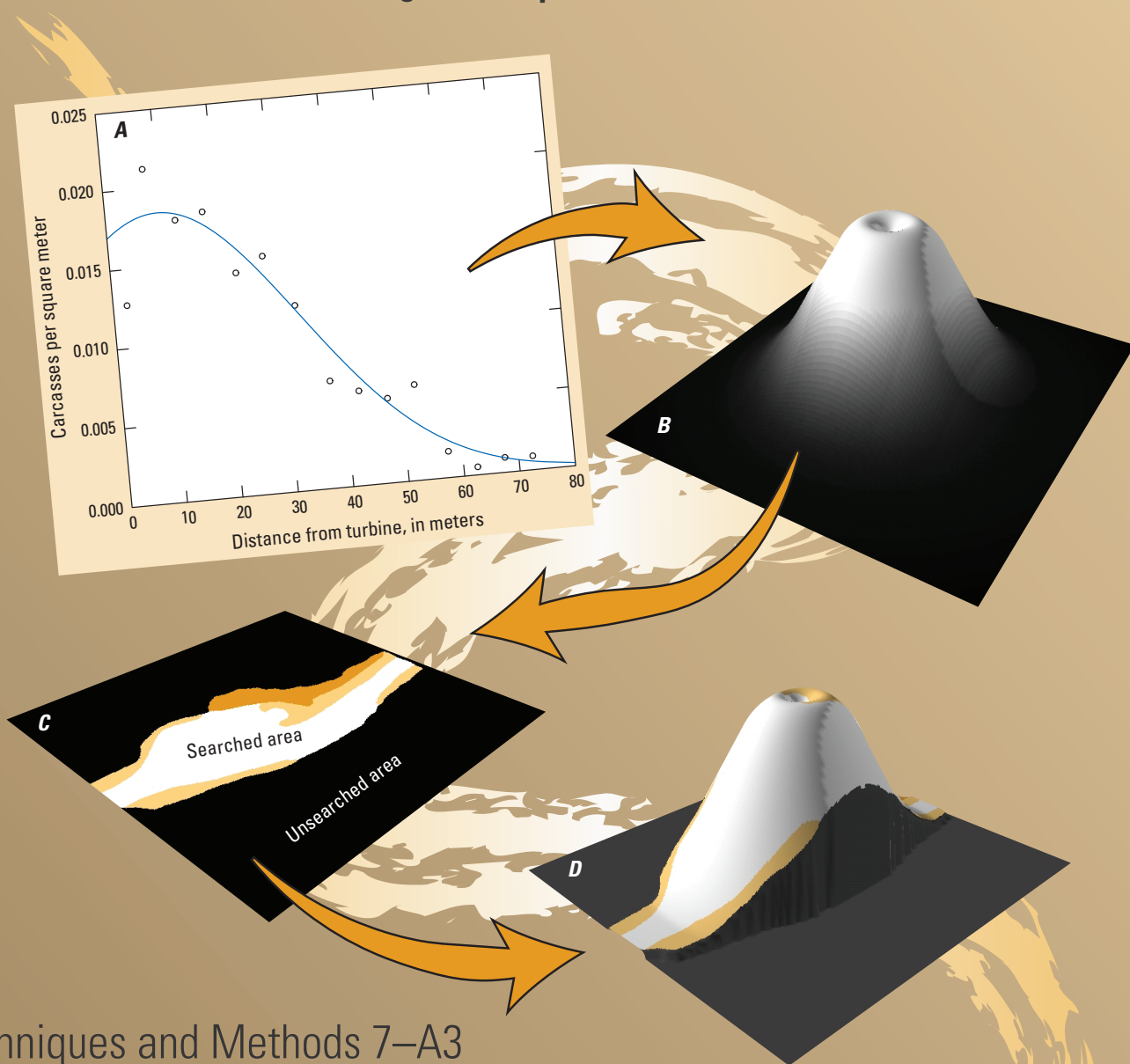# Accounting for the Fraction of Carcasses Outside the Searched Area in the Estimation of Bird and Bat Fatalities at Wind Energy Facilities

Chapter 3 of
Section A, Algorithms
Book 7, Automated Data Processing and Computations

Techniques and Methods 7–A3

Cover (from top left, following arrows). Graphic representation of (*A*) the process of modeling carcass density as a function of distance, (*B*) applying that modeling in all directions around the turbine, (*C*) selecting only those areas that were searched, and (*D*) integrating to give a density-weighted proportion.

Accounting for the Fraction of Carcasses Outside the Searched Area in the Estimation of Bird and Bat Fatalities at Wind Energy Facilities

By Daniel Dalthorp, Manuela Huso, Mark Dalthorp, and Jeffrey Mintz

Techniques and Methods 7–A3

U.S. Geological Survey, Reston, Virginia: 2024

Acknowledgments

Contents

Figures

Tables

Conversion Factors

International System of Units to U.S. customary units

Multiply	By	To obtain
Length		
meter (m)	3.281	foot (ft)
Area		
square meter (m²)	0.0002471	acre
square meter (m²)	10.76	square foot (ft²)
Flow rate		
meter per second (m/s)	3.281	foot per second (ft/s)

Abbreviations and Definitions

ψ probability of a carcass lying inside the searched area

AICc   Akaike information criterion corrected for finite sample size

ΔAICc   difference between AICc value for a given model and the lowest AICc value among the models tested

CDF   cumulative distribution function

dwp density-weighted proportion, fraction of carcasses lying inside the searched area

g carcass detection probability inside the searched area

GenEst   Generalized Estimator of Mortality software and statistical model

GLM   generalized linear model

IQR   interquartile range, which equals 25th quantile to 75th quantile of a dataset

M mortality, number of fatalities

PDF   probability density function

SE   searcher efficiency, probability of finding a carcass that is present at time of search

xepx   specification of carcass distribution model (app. 1)

NOTE: A hat (^) over a parameter signifies an estimate or estimator of the parameter rather than its true (but unknown) value. For example, the actual number of fatalities, M, is unknown, and $\widehat{M}$ would denote the estimated number of fatalities.

Accounting for the Fraction of Carcasses Outside the Searched Area in the Estimation of Bird and Bat Fatalities at Wind Energy Facilities

By Daniel Dalthorp,[1] Manuela Huso,[1] Mark Dalthorp,[2] and Jeffrey Mintz,[1]

Abstract

Accurate estimation of bird and bat mortality at wind energy facilities requires accounting for carcasses that lie outside the search plots because they lie beyond the search radius or in areas within the search radius that remain unsearched due to sub-optimal search conditions such as thick vegetation, rough or dangerous ground, water, or restricted access to the land. However, carcass density is not constant around a turbine and the fraction of carcasses within the unsearched area can vary greatly depending on where the area lies relative to the turbine. The *density-weighted proportion* approach takes into account the changing density of carcasses around turbines to estimate the fraction of carcasses lying in unsearched areas (*dwp*). It involves tallying the carcasses found in concentric rings centered at the turbine, fitting a curve to the carcass densities in the rings, and dividing the integral of the curve over the area searched by the integral over the total area. Accounting for unsearched area presents special difficulties such as extrapolation beyond the search radius, spatial prediction, and model selection, which are frequently ignored or under-appreciated, potentially resulting in substantial estimation errors.

A powerful new R software package (dwp) is available to perform the calculations, given the distances at which carcasses were found from turbines and a map of the searched area used to discern the fraction of the ground searched at each distance. If all ground within a given search radius has been searched, the map is simply the search radius. For more complicated search plots, other kinds of maps may be used: R polygons for plots that can be readily delineated into searched and not-searched areas (for example, searches restricted to access roads and turbine pads), GIS shape files for complicated search patterns (for example, non-uniform vegetation or ground texture resulting in spatially varying search conditions), or raster files for complicated search patterns coupled with carcass spatial distribution that depends on both distance and direction from turbines.

This study discusses estimation and interpretation of *dwp* in the context of several realistic examples; provides guidance for use of the dwp software for doing the analyses; and addresses questions of extrapolation, spatial prediction, and model selection.

Overview

Estimation of bird and bat mortality at wind energy facilities typically involves periodic searches for carcasses on the ground near turbines and adjustment of the carcass count to account for the estimated fraction of carcasses missed in the searches. In order for a carcass to be observed, it must lie inside the area searched, persist until the time of search, and be found by searchers. Conceptually, the total mortality (*M*) is estimated as $\widehat{M} = \dfrac{x}{dw \times pr \times SE}$, where *x* is the number of carcasses observed, *dwp* is the fraction of carcasses lying inside the searched area, *r* is the probability a carcass persists until the search, and *SE* is the searcher efficiency or probability that a carcass is found given that it is present in the searched area at the time of the search. Searcher efficiency and carcass persistence typically are estimated from data collected in field trials and combined into an estimated detection probability, $\hat{g}$. The number of carcasses lying inside the searched area is then estimated as $\widehat{M}_{\text{inside}} = x/\hat{g}$.

---

[1]U.S. Geological Survey

[2]Cornell University

The techniques and models for estimating g tend to be complicated, but there are several convenient R packages available to aid in the estimation of SE, r, g, and $M \mid \{x, \hat{g}\}$. The most flexible and powerful is GenEst (Dalthorp, Madsen, and others, 2018; Dalthorp, Simonis, and others, 2018; Rabie and others, 2021), whereas eoa (Dalthorp and others, 2017) is specially designed for accuracy and utility when carcass counts are very small or even 0. Several other packages are available as well (Huso and others, 2012; Wolpert and Coleman, 2015; Korner-Nievergelt and others, 2015). However, in mortality estimation, dwp is as influential as SE and r, but estimation of dwp has received relatively little attention in the literature and in mortality estimation protocols despite its importance. Mortality estimates that are derived from careful and accurate estimation of SE and r and from careful and accurate modeling of g can still be grossly inaccurate unless dwp also is carefully and accurately accounted for. Shortcuts in dwp estimation are likely to result in unreliable mortality estimates. Estimation of dwp presents special difficulties such as extrapolation outside the search radius, spatial prediction and model selection, which are discussed in this report and addressed by the dwp software package for R (Dalthorp and Huso, 2022).

We refer to the fraction of carcasses lying inside the searched area as the density-weighted proportion (dwp) after Huso and others (2012). It is sometimes also referred to as the area correction factor (AC) (Studyvin and Rabie, 2022, or spatial coverage (a) (Dalthorp and others, 2017). Estimates of dwp (or AC or a) can be used with GenEst (as the dwp parameter) or eoa (as the a parameter) in estimating mortality. Additionally, the dwp package provides an extensive set of tools for analyzing carcass density as a function of distance from turbine which may be of interest beyond estimating mortality at a particular site and may be used in more general studies of carcass dispersion at wind energy sites under a variety of conditions. Here we present (1) a set of tutorials on the use of dwp software and a set of appendixes that provide greater detail on the models used in the dwp software, the performance of the models and difficulties of model selection under a wide variety of conditions; (2) the theoretical foundations of the models used in the dwp software; and (3) extensive simulation studies to investigate estimator performance and model selection under a wide variety of conditions.

We begin with a brief discussion of terminology, general concepts, difficulties, special techniques used in dwp analysis, and the array of command line tools that dwp (1.0.1) provides. Section, "General Principles" gives a brief introduction to the main tasks and a discussion of some critical issues to consider in estimating dwp. More detailed discussion of data requirements and formatting is included in sections, "Site Layout—Raw Data," "Site Layout—Formatted Data," and "Examples." The second half of the document is a set of technical appendixes, discussing details about the models used in dwp (app. 1); simulated carcass distributions used in testing the models (app. 2); model performance in the simulation scenarios (app.3); and additional technical details about dwp and the accuracy of confidence intervals for ψ (probability that a carcass lies in the searched area), dwp, and M (app. 4 and 5).

General Principles

Carcass Dispersion

After being struck and killed by a turbine blade, where a carcass will land is a function of many factors, such as wind velocity, turbine blade speed, strike angle on the blade, carcass size, carcass cross-sectional area presented to the wind, wind shear, initial velocity of the carcass after being struck, and others. After landing, the carcass may be susceptible to being moved by scavengers, gravity, or wind, or—in cases where the animal is not immediately killed upon impact—it may hobble some distance seeking shelter.

The factors affecting carcass location are numerous and complex. Regardless of which combination of factors are in operation and in what ways, a general principle of carcass dispersion remains the same, namely, that carcass densities on the ground decrease with distance because (1) fewer carcasses fall at great distances from turbines than near turbines, and (2) carcasses are spread over greater areas at greater distances from turbines. Although the change in density with distance is not always monotone, and near the turbine density may actually increase with distance over a short range, there is a limit to how far carcasses will fall from a turbine, and the density of carcasses generated by a turbine will inevitably decrease to zero at great distances from the turbine.

Because carcass density varies, a simple area correction that divides the carcass count by the fraction of the area sampled to get, for example $\widehat{M} = x/0.75$ because 75 percent of the area was sampled, is highly prone to error (Huso and others, 2011; Huso and Dalthorp, 2014). For example, locations of 100 carcasses with a gradual decrease in carcass density with increasing distance from the turbine are shown in figure 1. The designated search plot area is a square, 200 meters (m) on a side, centered at the turbine. If 25 percent of the area is not searchable, the area correction depends strongly on which 25 percent is not searched. For example, each shade of gray constitutes 25 percent of the total area in the 200 m × 200 m square. If the gray area near the outer edge of the plot is the 25 percent that is not searched, then the simple area correction would estimate M
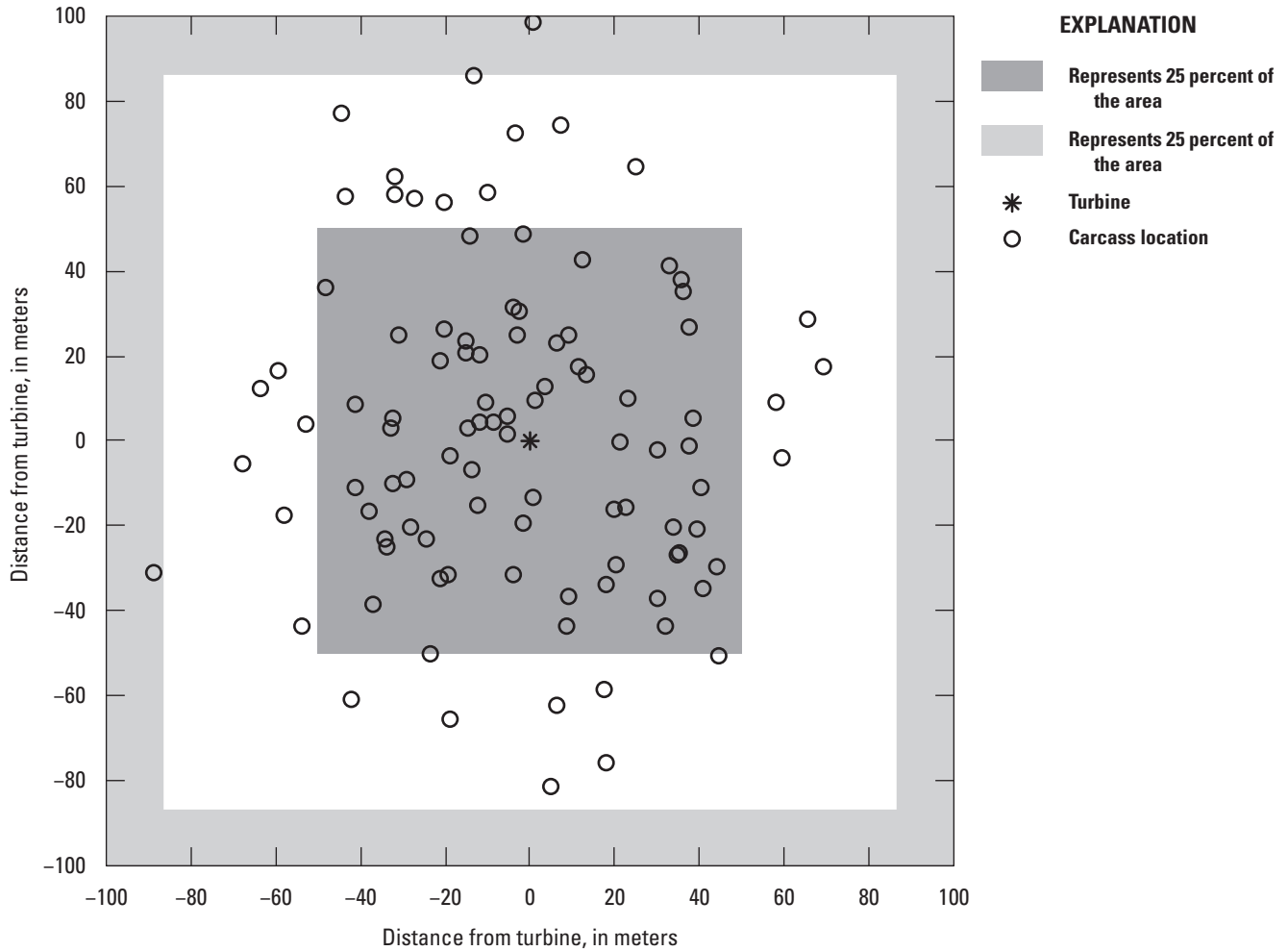
Figure 1. Carcass locations relative to turbine. Areas with shades of gray each constitute 25 percent of the area in a 200-meter square plot centered at the turbine.

as $\widehat{M} = 98/0.75 = 130.7$, which substantially overestimates the actual mortality of $M = 100$. If, instead, the gray area in the center is the 25 percent not searched, then $\widehat{M} = 31/0.75 = 41.3$, which vastly underestimates M. Clearly, a more reliable solution than simple area correction is needed.

Accounting for Unsearched Area

The dwp package fits statistical models of the patterns of change in carcass density with distance from turbines and uses the models to predict what fraction of carcasses lie inside the searched area. Given a list of the locations at which carcasses were discovered and a map of the searched area, the `prepRing` function[3] tallies the carcasses in concentric 1-meter (m) rings and estimates the carcass density in each ring by dividing the carcass count by the area sampled at that distance (fig. 2A).

Then, `ddFit` is used to fit a curve to the carcass densities (fig. 2A, solid curved line) and rotate the curve around the turbine to create a volcano-like surface of relative carcass densities (fig. 2B)[4]. After proper normalization (which is done automatically by `ddFit`), the volcano surface is a probability density function (PDF) that shows the relative carcass density at each point. The total volume of the volcano represents 100 percent of the carcasses, and the volume under a specific area represents the probability that a carcass will lie in that area.

---

[3]This introductory section is limited to a simple outline of the main processes involved in a *dwp* analysis. Details for formatting and importing data and using the functions in the dwp package are discussed in detailed examples in later sections. The figures shown in this section are meant as simple heuristics, and the dwp package does not produce similar pictures.

[4]The rotation is done mathematically, and there is no dwp function for creating such "volcano plots."
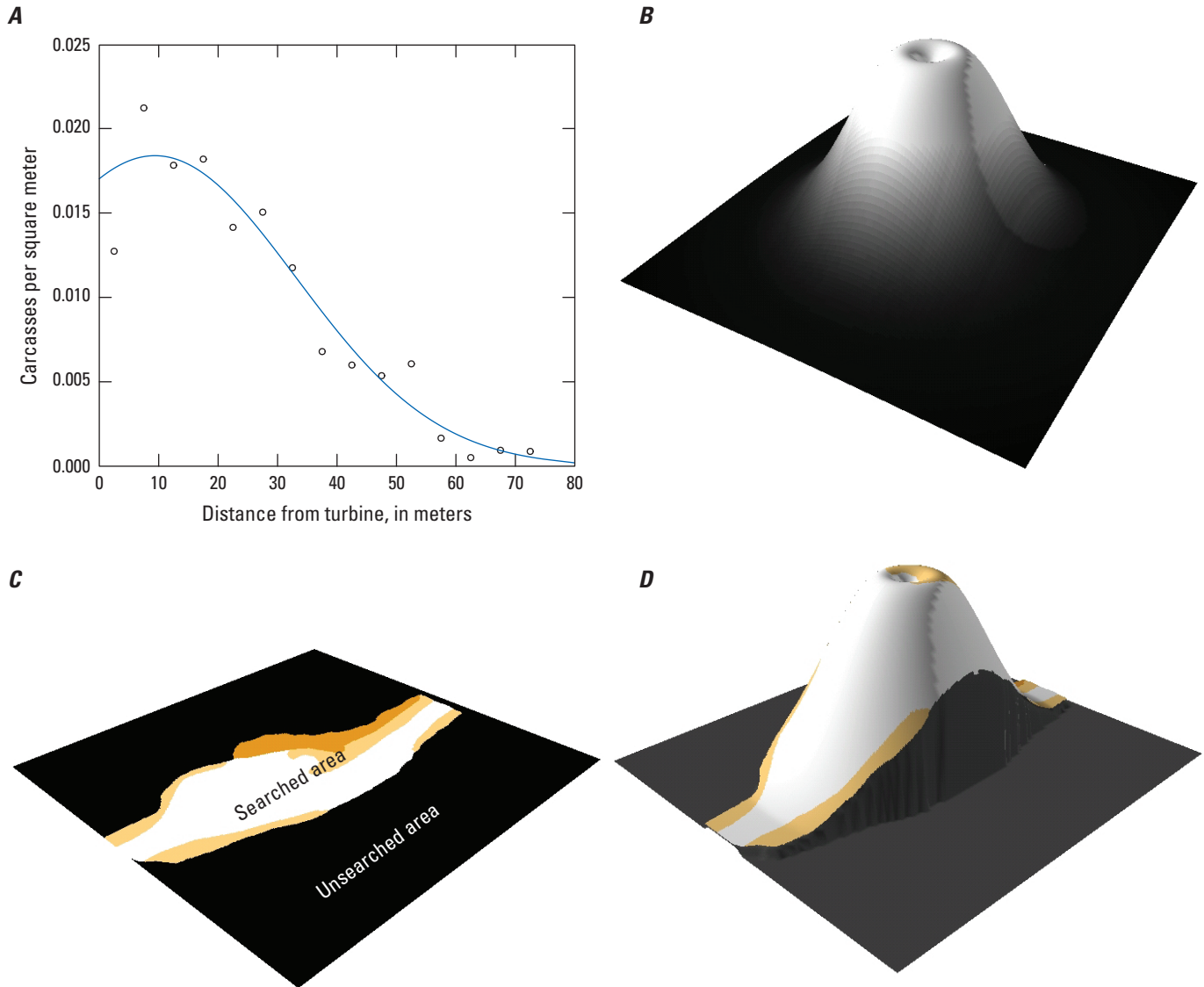
Figure 2.   Estimations of proportion of carcasses lying inside the searched area. *A*, Carcass density as a function of distance from the turbine. *B,* Relative carcass density in the area around the turbine, normalized so that it integrates to 1. *C*, Area searched. *D*, Relative density for the area searched. The fraction of carcasses lying inside the searched area is the normalized relative carcass density integrated over the searched area.

For example, the searched area at the turbine in figure 2 is an irregularly shaped stripe down the middle. The probability of a carcass lying inside the searched area (ψ) is the volume of the volcano above that area (fig. 2*D*) and is calculated using estpsi.

Although the basic idea is fairly straightforward, there are some nuances that are easy to overlook or whose importance is easy to discount. In particular, estimating how many carcasses lie outside the search radius requires extrapolation well beyond the range of the data, which is always hazardous. Furthermore, constructing proper confidence intervals for the fraction of carcasses inside the searched area (*dwp*) requires accounting for the (1) uncertainty in the probability that a carcass lies inside the searched area (ψ) and (2) uncertainty in the actual number of carcasses inside the searched area given ψ. The complications involved in extrapolation are discussed in general in section, "Extrapolation Outside the Searched Area." Examples and discussion of tools for resolving the complications are given in section, "Examples," and an extended theoretical discussion is presented in appendix 3. For the reader who is curious about the technical details of how the estimation uncertainties are accounted for and accurate confidence intervals are constructed by estpsi and estdwp, the theory is presented and tested in appendix 5.

Model of Carcass Density

In the dwp package, estimation of the fraction of carcasses that lie inside the searched area involves three general tasks:

1. Modeling carcass density as a function of distance from the turbine and integrating over the searched area,

2. Extrapolating carcass density models to areas outside the search radius, and

3. Incorporating the uncertainties that are inherent to the estimation process into the final estimates of *dwp*.

The choice of models to use is influenced by the need to prevent over- or under-estimation of tail probabilities during extrapolation. dwp provides several checks to help avoid selection of unrealistic models—extensibility, tail plausibility, and high influence filters (discussed in section, "Tools for Model Selection"). The models remaining after filtering can be further compared using AICc; however, AICc must be used with discretion as it is an unreliable measure of model performance outside the search radius, as demonstrated in appendix 3.

The model selection and filtering process are guided by several principles or assumptions:

1. Distribution of distances of carcasses from turbines belongs to an exponential family (app. 1);

2. Number of animals killed by a turbine is finite, and consequentially there exists a radius beyond which very few of those animals will be found; and

3. Models should not be overly sensitive to any single observation or data point.

Together, these principles guide us toward a collection of distributions and rules by which to select from the collection for further comparison.

A set of homogeneity assumptions about carcass dispersion ensures accuracy of the estimates and inferences:

1. Fatality rates do not vary among turbines,

2. Carcass distributions do not vary among turbines, and

3. Carcass distributions do not depend on direction (isotropy).

These assumptions allow for straightforward pooling of the carcasses across turbines to fit a single carcass distance model that can be applied to an entire site. Once a model for a site is selected, *dwp* can be estimated for each turbine. Even if the same distance model is used for the entire site, *dwp* will typically vary among turbines, depending on the size and shape of the search plots. If the homogeneity assumptions are not met, the analysis may require additional steps to ensure accuracy.

Homogeneity of Fatality Rates Among Turbines

Homogeneity of rates does not mean that the actual number of fatalities must be identical at all turbines but that the numbers do not vary significantly. In this context, it is difficult to precisely define "significantly" because the effect that variation in rates has on the model depends on the sizes, shapes, and characteristics of the search plots, and tolerance for potential bias depends on the objectives of the research. In theory, variation in rates can adversely affect accuracy. In practice, however, even substantial variation in mortality rates among turbines will rarely cause significant inaccuracies in estimating *dwp*.

- If the sizes and shapes of the search plots are identical among the turbines and the detection probabilities are the same, then homogeneity of rates is not necessary. The resulting fitted model will be the same regardless of the degree of variation in rates among turbines.

- If search plot shapes vary widely among turbines and plot shapes are correlated with fatality rates, then there is potential for modest bias due to heterogeneity of fatality rates. An example of how this might occur would be if turbines that are known or strongly suspected to have lower than average fatality rates are searched on roads and pads only, whereas turbines with relatively high fatality rates are searched on cleared plots with a large search radius.

- If search plot size and shape does not correlate with mortality rate, then the potential for bias due to lack of homogeneity fatality rates is greatly reduced.

Homogeneity of Carcass Distributions Among Turbines

Variation in wind patterns, turbine types, turbine operations (for example, curtailed versus uncurtailed), carcass type, and other factors may result in variation in carcass dispersion patterns. We refer to factors that vary among turbines or carcasses and that influence the actual carcass distributions as interacting covariates. These factors can be explicitly accounted for in the carcass distance models when the search plot layouts are imported from shapefiles or as *x-y* coordinates on a grid. With the simpler data types (list of distances, simple geometry, or R polygons), interacting covariates can be accounted for by fitting separate models for different levels of the covariate. For example, bats and small, medium, and large birds may each require their own models; likewise for curtailed and uncurtailed turbines, large turbines and small turbines, site A and site B, etc.

Homogeneity of Carcass Distributions in Each Direction

Variation in carcass distributions by direction from a turbine is referred to as anisotropy. If not accounted for, anisotropy can be a significant source of bias when the shapes of the search plots also vary by direction. Accounting for anisotropy introduces a considerable degree of statistical complexity along with more extensive data requirements. Perhaps the most straightforward and flexible way to account for anisotropy would be via Poisson regression of carcass counts on a coordinate grid, as discussed in Maurer and others (2020). The dwp package (version 1.0.1) does not include special functions for handling anisotropy.

Extrapolation Outside the Searched Area

There is a wide variety of possible approaches that could be used to model carcass density as a function of distance from turbine. Any approach to fitting a distribution will need to account for carcasses that lie outside the search radius and beyond the scope of data and will thus require extrapolation. All approaches will be faced with the inherent difficulty that model performance within the range of data may tell little to nothing about how well it will perform outside the range of the data. Traditional model selection criteria such as AIC (Akaike, 1974; Burnham and Anderson, 2002), likelihood ratio tests (Vuong, 1989), stepwise regression (Efroymson, 1960), cross-validation (Allen, 1974) and others have limited utility when a significant fraction of carcasses lies outside the search radius. In fact, two models, A and B, that are equally plausible in theory and are virtually indistinguishable within the range of data may be radically different outside the range of data, giving wildly different estimates of the fraction of carcasses inside the searched area.

The dwp package fits a large number of probability distributions in the exponential family (Andersen, 1970), including the well-known and named exponential, gamma, normal, lognormal, Rayleigh, Maxwell-Boltzmann, inverse gamma, inverse Gaussian, normal-gamma, Pareto, and chi-squared models and several unnamed models, which we designate with names like "xep1" and "xep012." Detailed knowledge of the models is not required for understanding the difficulties inherent in extrapolation outside the search radius, but for the curious reader, further details about the models and our techniques for fitting them are discussed in appendix 1.

The example discussed in this section illustrates some important general concepts and issues involved in determining the fraction of carcasses inside the searched area, and by extension, the fraction outside that will be missed. Most notably,

1. Predicting the number or fraction of carcasses lying outside the search radius requires extrapolation or assumption;

2. Two (or more) models that are in close agreement inside the range of the data may have radically different behaviors outside the search radius;

3. Criteria that draw on knowledge about carcass ecology (for example, the number of carcasses is finite and a very small fraction of carcasses will lie beyond 150 or 200 m from their turbine) are a necessary complement to traditional criteria (like AICc, high-influence points) for assessing model suitability; and

4. The `modelFilter` function performs a suite of diagnostic tests and ranks the models by suitability.

Example Data

Problems with extrapolating fitted models outside the search radius are illustrated clearly in an example dataset in which 100 carcass locations were simulated. Distances were drawn from a gamma distribution in which one-half the carcasses are expected to fall within 41 m of the turbine and 82 percent within 80 m. Searches were conducted in a circular plot centered at the turbine and with radius 80 m. All 81 carcasses inside the search radius were found and none of the 19 carcasses outside the search radius were found (fig. 3). The dwp package was used to fit 17 models from the exponential family of distributions.

Slightly Outside the Search Radius—Radical Divergence Among Models

All but a few (constant density, chi squared, Maxwell-Boltzmann, and inverse Gaussian) of the 17 models pictured in figure 4, appear to provide fairly close fits to the carcass data inside the 80-m search radius (fig. 4, gray bars), but just outside the search radius there is wide divergence among the models (fig. 4, lines). Although the vast majority (81 percent) of the carcasses lie inside the search radius, there is no guarantee that the models can accurately predict the final 19 percent of carcasses that lie outside the search radius. Indeed, the actual fraction of carcasses lying inside the search radius is 81 percent, but the models predict anywhere from 0 to 100 percent ($\hat{\psi}$ in table 1), depending on the model.
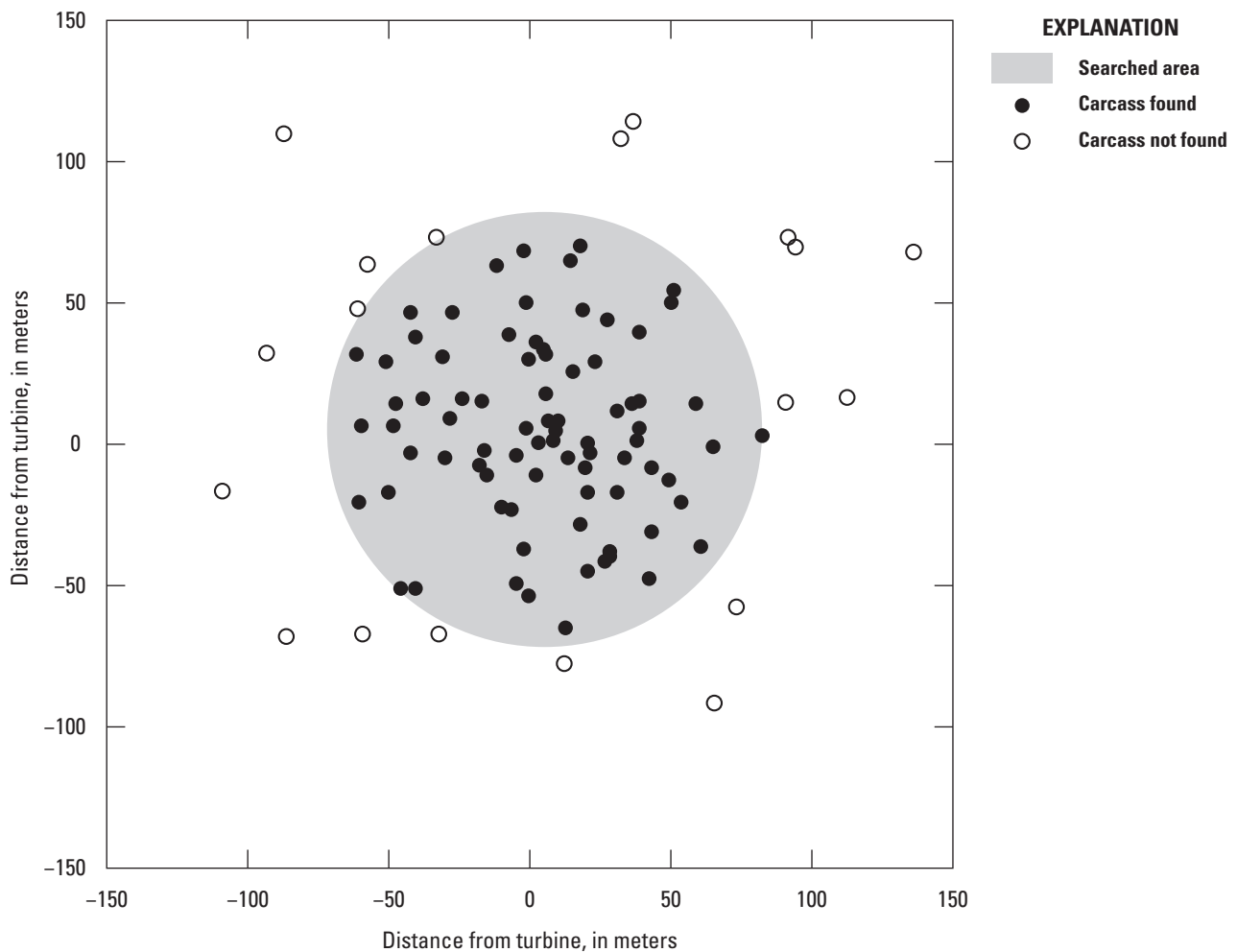
Figure 3.   Simulated carcass locations relative to the nearest turbine. The turbine is at the center of a circular search plot with a radius of 80 meters. There are 81 carcasses inside the search radius (all were found) and 19 outside the search radius (none were found).
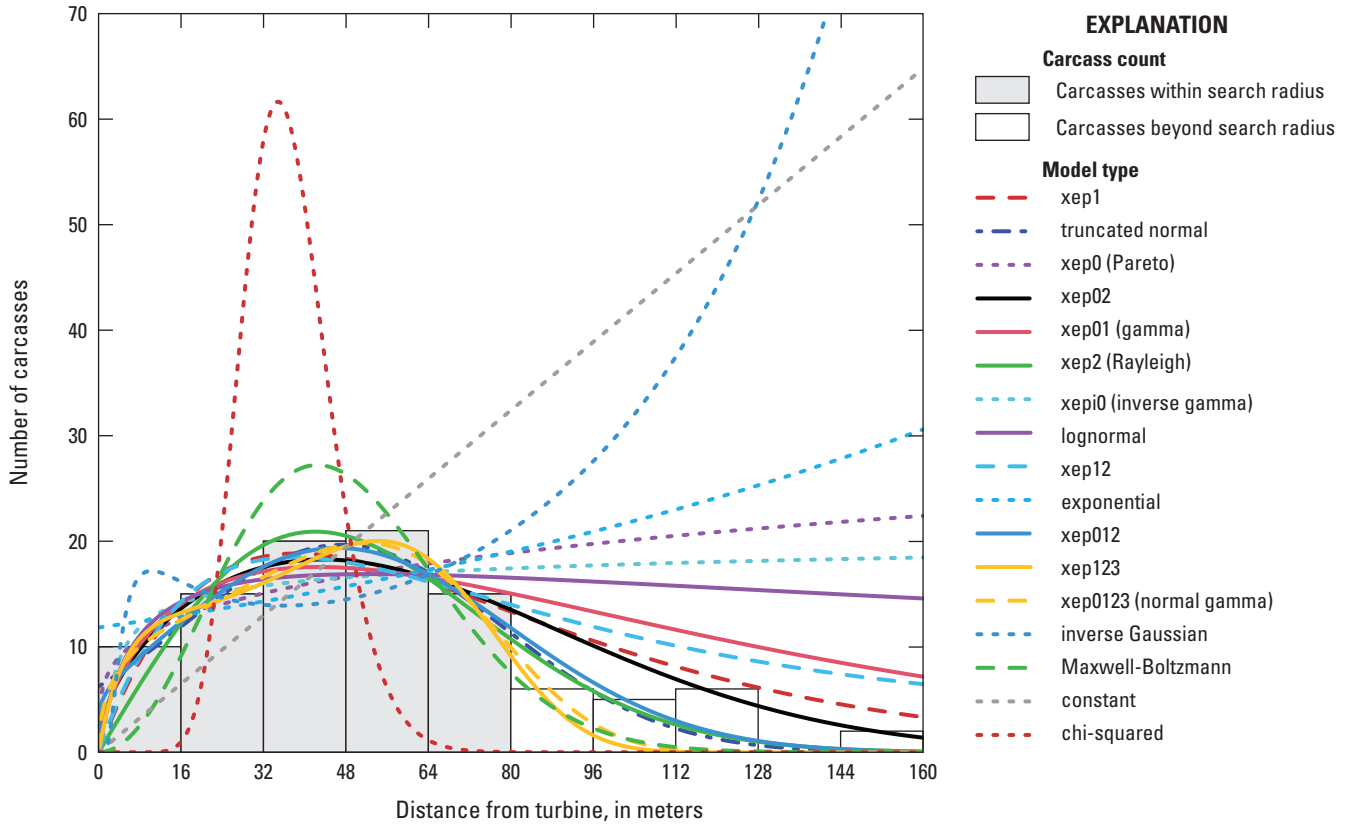
Figure 4. Histogram of simulated data and model spatial predictions of relative carcass density.

Highly Limited Utility of the Akaike Information Criterion for Model Selection

A traditional staple in model selection is the Akaike information criterion or AIC (Burnham and Anderson, 2002), which is a measure of how well a model fits the data given the number of parameters in the model. We use AICc, which is a slightly modified version of AIC that may be more accurate than AIC when sample sizes are small (Cavanaugh, 1997). To facilitate easy comparison of models, we define ΔAICc as the difference between a model's AICc score and the minimum of the AICc scores of all models to be compared. The model with the most parsimonious fit has ΔAICc=0 and serves as the standard for comparison.

Thus, ΔAICc confirms the visual impression that the constant, chi-squared, and Maxwell-Boltzmann models give relatively poor fits to the data (fig. 4). Among the remaining models, it is not clear which provide the best fits to the data inside the search radius. AICc does capture some of the subtle distinctions among the model fits inside the range of data. However, outside the range of the data, the models diverge radically from each other, with the estimated fraction of carcasses lying inside the search radius ranging from 0 to 93 percent for the models with ΔAICc < 3. AICc provides no guidance for assessing model performance beyond the search radius.

As a rule of thumb, for two models that differ by more than about 7 in AICc, the one with the smaller AICc is preferable, and models with scores differing by less than about 2 are often considered statistically indistinguishable by this measure (Richards, 2005). If we only consider models with ΔAICc < 2, the range of estimated proportions of carcasses inside the search area narrows only slightly, to 0–84.3 percent. It is evident that AICc has extremely limited utility[5] in distinguishing among models and their ability to extrapolate accurately. AICc measures only the quality of the fit of a model within the range of data and offers no guidance on performance outside the range of the data. Additional selection criteria are essential (see section, "Model Filter—Extensibility").

---

[5]That utility may even be exaggerated in this example because the ΔAICc<2 is far too restrictive, eliminating from consideration some models that appear to provide only marginally poorer quality fits within the range of data than the "best" models but without regard to the shapes of the curves outside the range of data.

Table 1. Diagnostic statistics for 17 models shown in figure 4.

[*k*: Number of parameters in the model. **ΔAICc:** Difference between a model's AICc and the lowest AICc among the models tested. (AICc is Akaike information criterion corrected for finite sample size.) **Extensible**: Extensibility test. $\hat{\psi}$: Estimated probability that a carcass lies inside the searched area. $\widehat{M}$: Estimated total number of fatalities. Inf, the model's density prediction did not converge to 0. **Tail weight:** Indicator of whether the right (R) and left (L) tail probabilities of the distribution are plausible (1) or not (0). **HI:** Indicator of whether the model is free of points with high influence (HI = 1). Data points with high influence on a model (HI = 0) cast doubt on the reliability of the model]

Model (fig. 4)	k	ΔAICc	Extensible	$\hat{\psi}$	$\widehat{M}$	Tail weight		
						R	L	HI
xep1	2	0.00	1	0.614	131.9	0	1	1
truncated normal	3	0.54	1	0.843	96.1	1	1	1
xep0 (Pareto)	2	0.56	0	0.000	Inf	0	1	0
xep02	3	0.87	1	0.695	116.5	1	1	0
xep01 (gamma)	3	1.33	1	0.451	179.6	0	1	0
xep2 (Rayleigh)	2	1.49	1	0.836	96.9	1	1	1
xepi0 (inverse gamma)	3	1.61	0	0.000	Inf	0	1	0
lognormal	3	1.70	1	0.033	2,454.5	0	1	0
xep12	3	2.13	0	0.000	Inf	0	1	1
exponential	2	2.64	0	0.000	Inf	0	1	1
xep012	4	2.67	1	0.820	98.8	1	1	0
xep123	4	2.68	1	0.934	86.7	1	1	0
xep0123 (normal-gamma)	5	4.69	1	0.913	88.7	1	1	0
inverse Gaussian	3	6.52	0	0.000	Inf	0	1	0
Maxwell-Boltzmann	2	18.22	1	0.931	87.0	1	1	1
constant	1	19.82	0	0.000	Inf	0	1	1
chi-squared	2	380.85	1	1.000	81.0	1	0	1

Tools for Model Selection

With a closer look at the model predictions in figure 4 and closer consideration of what is known about carcass dispersion around turbines, we can filter out most of the obviously poor choices, but we should do so without reference to the white bars in the histogram because they are from carcasses outside the search radius, data that we normally would not have access to.

The `modelFilter` function performs a series of tests to aid in model selection and returns the results in a convenient format, greatly reducing or even eliminating the guesswork. The tests are intuitive, easy to understand, and easy to implement.

Model Filter—Extensibility

If we assume that the number of carcasses is finite, we can eliminate from consideration all models that do not converge to zero or do not converge fast enough to guarantee that the estimated number of carcasses outside the search area is finite. We refer to models that predict a finite number of carcasses outside the searched area as extensible because they can be extended outside the search radius to give properly defined probability distributions[6]. Typically, models that are still increasing outside the search radius are non-extensible, and models that are decreasing are extensible (figs. 4 and 5*A*).[7]

The estimated fraction of carcasses inside the search radius for non-extensible models is an impossible 0 percent. Among the extensible models—which, for this data, include the truncated normal, gamma, Rayleigh, log- normal, Maxwell-Boltzmann, normal-gamma, chi-squared, xep1, xep02, xep012, and xep123—the estimated fraction ranges from a merely implausible 3.3 to 100 percent (table 1).

---

[6]This is discussed further in sections, "Extrapolation Outside the Searched Area" and "Examples," and in appendix 1.

[7]The rule is not hard and fast. Sometimes a non-extensible model decreases outside the search radius but at too slow a rate; sometimes an extensible model may still be increasing at the search radius but is decreasing rapidly shortly thereafter. These are exceptional situations. They may be rare but they are not impossible.

Figure 5.   Criteria for filtering out poor model choices for extrapolating beyond the range of the data and predicting proportion of carcasses outside the search radius. *A*, Extensible models. *B*, Extensible models, plausible tails. *C*, Extensible models, plausible tails, and no high influence points. *D,* Extensible models, plausible tails, no high influence points, and ΔAICc less than 10. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between a model's AICc and the lowest AICc among the models tested.

Extensibility is a relatively permissive criterion that excludes only the most grossly inadequate models. Several dwp package functions automatically test for extensibility. Most notably, `ddFit` tags all fitted models as either extensible (`$extensible = 1`) or not extensible (`$extensible = 0`), and `modelFilter` returns a table of test results (including extensibility) for arrays of models to be compared.

Model Filter—Tail Plausibility

Because gravity acts so inexorably on carcasses after turbine strikes, few carcasses are expected beyond 150 or 200 m (although this number will likely increase as turbines increase in height and rotor diameter.) However, because the search plots are inevitably finite and the models are fit with spatially limited data, sometimes a fitted model will perform well inside the search radius but have implausible tails, predicting far too many carcasses at great or short distances. A rough filter for tail plausibility identifies obviously inappropriate models. By default, models that predict that more than 1percent of carcasses lie beyond 200 m or more than 5 percent beyond 150 m fail the `modelFilter` for a plausible right tail.[8] For this dataset with the default settings, the filter eliminates the gamma, lognormal, and xep1 models (table 1), which were extensible but predict too many carcasses at great distances from the turbine to be plausible. For example, the lognormal predicts that 93.8 percent of the carcasses fall beyond 150 m, whereas the gamma and xep1 models predict 25.3 and 90 percent, respectively.

Because of the great size and height of turbines, we would expect a substantial fraction of carcasses to lie beyond 20 m and a significant fraction beyond 50 m. By default, `modelFilter` identifies models that predict that more than 50 percent of the carcasses lie within 20 m or more than 90 percent within 50 m as having implausibly heavy left tails. By this criterion, the chi-squared model is eliminated from further consideration for this dataset (table 1).

The tests of tail plausibility further narrow the range of the estimated proportion of carcasses inside the search radius from 3.3–100 percent to 69.5–93.4 percent (table 1), which is still a wide range but a vast improvement over using the extensibility criterion by itself (fig. 5).

Model Filter—High Influence Points

In some cases, the apparently good fit of a model may strongly depend on one or a small set of data points, so that removing one or a small number of data points has a strong impact on the shape of the fitted model. The presence of these high influence points (Davison and Hinkley, 1997; Canty and Ripley, 2021) casts considerable doubt on the reliability of the model for the given dataset. Extrapolation would be especially worrisome for such models because their shapes are strongly dependent on one or a small number of fickle data points, and the effects would be felt well beyond the range of the data.

Several of the extensible models with plausible tails have points with high influence—the xep02, xep012, xep123, and normal-gamma models (table 1). Removing these from consideration further narrows the range of estimated proportions of carcasses inside the searched area to $\hat{\psi} \in (0.836, 0.931)$ (table 1) and eliminates much of the remaining clutter from the graphs of models still under consideration (fig. 5B–5C).

Model Filter—AICc

After models with implausible extrapolations or worrisome instabilities have been filtered out, criteria like AICc that measure the quality of a model's fit strictly within the range of the data are useful for distinguishing among plausible models. AICc and other internal criteria are especially valuable when there are substantial unsearched areas inside the search radius, as would be the case with road and pad searches (Maurer and others, 2020) or with sites with forest or other thick vegetation, water, cliffs, poisonous snakes, or other features that render much of the area around turbines practically unsearchable. Predicting the proportion of carcasses in unsearched areas inside the search radius is an interpolation problem rather than extrapolation, and the quality of the fit to the data inside the search radius is directly relevant in a way that it is not when extrapolating outside the search radius.

In an extensive simulation study of the accuracy of carcass distribution models (app. 2 and 3), AICc was generally a poor guide for gaging the relative adequacy of models for extrapolating outside the search radius. Under some of the conditions tested, models with low AICc correlated with more accurate extrapolations; under other conditions, low AICc was correlated with less accurate extrapolations. However, under most conditions, there was no apparent correlation between AICc and

---

[8]If desired, users may override these default settings.

suitability for extrapolation. One exception was that models with ΔAICc greater than (>) 10 frequently performed worse than models with ΔAICc < 10. For this reason, `modelFilter` compares model AICc scores and, by default[9], identifies and filters out models with ΔAICc > 10.

After filtering out the lone extensible model with plausible tails, no high influence points, and ΔAICc > 10 (namely, the Maxwell-Boltzmann model), the two remaining models—truncated normal and Rayleigh—predict that, respectively, 84.3 and 83.6 percent of the carcasses lie inside the searched area. Both are in good agreement with the generating model in which 82 percent of the (simulated) carcasses are expected to lie inside the searched area. The fitted models closely match each other and appear to be good fits to the data (fig. 5). Also, the difference in their AICc scores is only 0.95, which is too small to make a statistically meaningful distinction between the models.

Standard Models

Several of the distributions are not well suited for modeling carcass distances. These include the inverse gamma (xepi0), inverse Gaussian, chi-squared, Pareto (xep0), and the exponential distributions. Because these distributions would rarely (if ever) be among the most appropriate models for carcasses, they are excluded from the dwp package's list of 12 "standard" distributions but can still be used if specifically invoked. The standard models, which include xep1, xep01 (gamma), xep2 (Rayleigh), xep02, xep12, xep02, xep123, xep0123 (normal-gamma), truncated normal, Maxwell-Boltzmann, lognormal and constant, are the primary focus of this report. All the models are defined and briefly discussed in appendix 1.

Summary

In general, when a significant fraction (for example, >10 percent) of carcasses are expected to lie outside the search radius and extrapolation is necessary to estimate the fraction of carcasses lying in the searched areas, it is essential to consider the suitability of models for extrapolation, using broad criteria like extensibility, tail plausibility, and stability (for example, the presence of high-influence points) before considering strictly internal criteria like AICc. If the search radius is long enough so that few carcasses are expected outside the search radius and a substantial fraction of the area inside the search radius is unsearched, then AICc and high-influence points become the critical criteria.

To determine which models are likely to perform well or poorly in predicting carcass numbers outside the search radius, it is necessary to step beyond traditional model selection criteria and make use of more general information about the distribution of carcasses. Most notably, the number of carcasses generated by a turbine will always be finite, and models that predict infinite carcasses outside the search radius should be eliminated from consideration. We refer to models that can be extended beyond the search radius to infinity and still predict finite numbers of carcasses as "extensible." Among the original 17 models, 11 are extensible for the example data. Although many of the remaining, non-extensible models (xep0 (Pareto), xepi0 (inverse gamma), xep12, exponential, inverse Gaussian, constant) perform well according to AICc (table 1) and provide good fits to the data, they are not suitable for extrapolation outside the search radius and should not be used in estimating the fraction of carcasses inside the searched area.

The example illustrates some of the power and utility of the diagnostics. However, it remains a single example, and results specific to this particular dataset may not be generalizable. For example, the lognormal model fits relatively well within the range of this data but was implausible for extrapolation because it estimated that 97 percent of the carcasses lay beyond the 80 m search radius. For this dataset, lognormal is implausible, but for other datasets it may be the best choice.

Uncertainties and Confidence Intervals

In estimating the total mortality (M), it is necessary to account for carcasses that lie outside the searched area. If the probability that a carcass lies inside the searched area is ψ, then a reasonable point estimate for M would be $\widehat{M} = m_{in}/\psi$, where m_{in} is the number of carcasses lying inside the searched area. However, even if we know precisely the probability that a carcass will lie inside the searched area and we know the number of carcasses that did lie inside the searched area, there will still be uncertainty about the total number of carcasses.

The problem is akin to coin tossing, where we know the probability of heads is $\psi = 0.5$. If someone tosses a coin some unknown number of times (M), tells us that there were $m_{in} = 5$ heads, and asks us to guess the total number of tosses, a good guess would be $\widehat{M} = 5/0.5 = 10$, which is the guess with the maximum likelihood. It is possible that the number of tosses

---

was 10, but it could easily have been 11 or 15 or 8 instead, but probably not as few as 5 or 6 or as many as 100 or even 25. That binomial uncertainty must be properly accounted for if we are to create accurate confidence intervals of either the fraction of carcasses lying inside the searched area or the total mortality.

The probability that a carcass will lie inside the searched area, ψ, is like the theoretical probability of H in a coin toss, whereas *dwp* is like the fraction of Hs in M actual coin tosses. The probability that a carcass will lie inside the searched area (ψ) is unknown. The dwp package estimates ψ by integrating (`estpsi`) a fitted carcass probability distribution (`ddFit` and `modelFilter`) over the searched area (`prepRing` and `addCarcass`). Once ψ is estimated, `estdwp` accounts for the uncertainty about actual fraction of carcasses lying in each turbine's searched area, given the number of carcasses observed at each turbine (X_i) and the estimated probability that a carcass lies inside the searched area ($\widehat{\psi}_i$).

The *dwp* is estimated in a way that is fully compatible with the GenEst mortality estimator. The distinction between *dwp* and ψ and their roles in estimation are discussed in more detail in appendixes 4 and 5. With the eoa estimator, a single number is entered as an aggregate *dwp* for the site as a whole. Typically, for eoa, a user would take the average "total" *dwp*, or `mean(dwp[, "total"])`, where `dwp` is the *dwp* array estimated from the `estdwp` function.

The dwp Package

Doing basic analyses with the dwp package does not require extensive prior experience with R, but basic familiarity with the workflow of entering/loading data, entering commands, and using R's "help" features at a basic level is assumed. This tutorial covers a number of examples and briefly discusses some of the main functions used in the data management and analysis. Additional information on formatting, usage, and additional options is available for any function by entering the function name (case sensitive) preceded by a question mark. A list of all functions, grouped by utility and linked to function-specific help files, can be seen by entering `?dwp` in the R command line. A complete reference manual with functions arranged alphabetically is available at https://cran.r-project.org/web/packages/dwp/index.html.

The general workflow involves several steps, and dwp provides easy-to-use functions to help accomplish each step:

1. Importing site layout data and formatting for analysis (`initLayout`),

2. Adding carcass data to the site layout (`readCarcass` and `addCarcass`),

3. Fitting carcass distance models (`ddFit`),

4. Selecting a carcass distance model to use in estimating *dwp* (`modelFilter`),

5. Estimating probability of carcass lying inside searched area (`estpsi`),

6. Estimating fraction of carcasses lying inside searched areas (`estdwp`), and

7. Exporting to GenEst (`exportGenEst`).

All data types require a quantitative description of the searched area. The description may be as simple as the search radius (with the assumption that all ground inside that search radius is searched and no ground outside that search radius is searched) or as complex as a detailed GIS-generated map. Altogether there are five options for site layout data format, including search radius, simple geometries, R polygons, GIS shapefiles, and rasters (that is, search plots laid out on an *xy*-grid). The data processing in the first two steps varies, depending on the data type. However, for many data types the calculations are all done automatically in the background by dwp functions, and the user interface for each data type is similar.

Site Layout—Raw Data

The dwp package can accommodate several different data formats for characterizing site layout, including GIS shapefiles, R polygons, simple geometric descriptions (square, circular, or road and pad searches with user-specified dimensions for each turbine), a list of carcass distances with a search radius, or coordinates on an *xy*-grid.

1. **Simple search radius** can accommodate a simple vector of distances from turbine to carcass. It assumes circular search plots with the same radius at each turbine with no unsearched area within the plots and the same detection probability in each plot; that is, the same searcher efficiency, carcass persistence and search interval.

2. **Simple geometry** can accommodate simple variation in search geometries as circular, square, or road and pad with specific parameters (shape, search radius, size of turbine pad, width of roads, number of roads) for each turbine individually. This option does not allow for multiple search classes. A *search class* is an area where the carcass detection probability is expected to be the same in all parts.

3. **R polygons** can accommodate moderately complex search geometries that do not involve multiple search classes with different detection probabilities. A *search class* is an area where the carcass detection probability is expected to be the same in all parts.

4. **GIS shapefiles** can accommodate highly complex search geometries, multiple carcass classes (for example, large, medium, small, bats) and multiple search classes with different detection probabilities corresponding to different ground types, search schedules (for example, 1-day and 14-day), or other variables that affect detection probability. A *search class* is an area where the carcass detection probability is expected to be the same in all parts.

5. ***xy*-grid** can accommodate highly complex search geometries and multiple covariates and is suitable for more sophisticated custom analyses, such as accounting for anisotropy or interacting covariates. Although it opens the door to more sophisticated analyses, gridded data are substantially more computationally demanding than other data formats, and the models run much more slowly. Additionally, the dwp package does not provide automated tools for doing complex, custom analyses.

More complete definitions of each data type are given in worked examples in section, "Examples."

Site Layout—Formatted Data

To estimate relative carcass density as a function of distance from a turbine, the package requires a description of the layout of the areas searched at a site and a list of locations where carcasses were discovered. The standard analysis divides the area around each turbine into 1-m rings and tallies the number of carcasses and the total area searched in each ring. The *xy*-grid data are processed and modeled on a grid rather than rings. The grid approach is ultimately more flexible and powerful than the rings but substantially more computationally intensive and slightly less accurate.

Ring Structure

With the exception of the *xy*-grid data, all the data types are converted to a common ring structure for analysis. A rings data structure is a list with six elements:

1. `$rdat` list of data frames—one for each turbine and one for the site as a whole—with 3 or 4 columns. `rdat` is used in fitting the distance models.

   - `r` = outer radius of 1-m ring.

   - `Class` = search class (may be `NULL` or a name other than Class).

   - `exposure` = area (m²) in the ring representing the given search class.

   - `ncarc` = number of carcasses in the given ring and search class.

2. `$rpA` list of data frames—one for each turbine and one for the site as a whole—giving the proportion of the area searched in each ring. `rpA` is used in calculating estimated *dwp*.

   - `r` = outer radius of 1-m ring.

   - `pinc` = proportion of area searched in the ring.

3. `$srad` = maximum search radius at any of the turbines searched.

4. `$ncarc` = vector of carcass counts at each turbine.

5. `$scVar` = name of the search class variable (or `NULL`).

6. `$tcenter` = array of turbine locations with columns *x* and *y* and row names are turbine IDs.

The analysis of rings data begins with the fitting of distance models. By default, the `ddFit` function fits 12 different generalized linear models (GLMs) to the data. Specifically, the models are Poisson regressions of carcass counts in 1-m rings, using the searched area (or the searched area in a given carcass class) as the exposure and the natural logarithm of the exposure as an offset. The fitted GLMs are then converted to parametric distributions. The models are discussed in greater detail in appendix 1.

The data structure for layouts defined on coordinate grids rather than shapes are somewhat different, but the analyses are similar.

Coordinate Grid (x,y)

A layout of the site on a grid of (x,y) coordinates marking the centers of 1 m^2 or 2 m $\times$2 m quadrats are the most flexible for modeling, but they are difficult to work with because of the computer memory requirements and slow processing speeds. A properly formatted, small (x,y) gridded layout dataset (`layout_xy`) is available in dwp. It is a standard data frame with x and y coordinates on 1- or 2-meter grids overlaying each turbine (`unitCol`) at the site. The coordinates may either be relative to the turbine location (that is, each turbine is assumed to be located at $(0, 0)$ according to the grid coordinates listed for the turbine), or the coordinates are UTMs with the turbine locations given in a separate file (`file_turbine`). The number of carcasses found in each grid cell also must be provided (`ncarcCol`).

The importing and formatting is accomplished via `initLayout`, and there is no need for further processing before proceeding to the analysis, as discussed in section, "The (x, y) Grid Data."

Examples

The remainder of the body of this guide (prior to the appendixes) is devoted to examples of increasingly complicated and detailed data analyses based on a variety of data formats seen in fatality monitoring programs.

Example 1—A Terse Analysis

The first example is a highly streamlined introduction to the main dwp functions and main tasks of a basic analysis. It serves strictly as an outline. More detailed and annotated analyses that address some of the difficulties that arise in a more thorough analysis are presented in sections, "Examples 2–6."

Import and Process the Data

In this example, the site layout and carcass location data are imported from shapefiles. Detection probability varies with search class on the ground, as specified in the `Class` column in the shapefile. It is not necessary to know or estimate the detection probabilities in each search class. Rather, `Class` can be specified as a covariate in the model, and the relative detection probabilities will be implicitly accounted for in the model.

The first steps are to import the data from shapefiles (`initLayout`), format the data into rings (`prepRing`), and tally the carcasses discovered in each ring (`addCarcass`).

```
# import site map
layout_shape <- initLayout(data_layout = "searchpoly.shp",
  file_turbine = "turbine_pt.shp", unitCol = "Turbine")

# format the site map for analysis
rings_shape <- prepRing(layout_shape, scVar = "Class", notSearched = "Out")

# import carcass locations
cod <- readCarcass("carcasses.shp", unitCol = "Turbine")

# add carcasses to the site layout
rings_shape <- addCarcass(cod, data_ring = rings_shape, plotLayout = layout_shape)
```

Fit Models and Select One

Once the site layout and carcass location data have been imported and formatted, use ddFit to fit a wide array of possible carcass distribution models to the data and use modelFilter as an aid in selecting an appropriate model for estimating *dwp*. Detailed help on any of the functions can be found by entering, for example, ?ddFit or ?modelFilter in R.

```
dmod_shape <- ddFit(rings_shape, scVar = "Class") # scVar is an optional covariate
models <- modelFilter(dmod_shape, quiet = TRUE)
best_mod <- models$filtered
```

Estimate Probability that a Carcass Lies Inside Searched Area

Now that a model has been selected, use estpsi to estimate ψ, the probability that a carcass lies inside the searched area. The Greek letter ψ is "psi" in English and short for "**p**robability that **s**earch area **i**ncludes carcass." estpsi estimates ψ at every turbine and accounts for the uncertainty in estimating the model parameters.

```
psihat <- estpsi(rings_shape, model = best_mod)
```

Estimate *dwp* and Export to File for Reading into GenEst

The "density-weighted proportion" (*dwp*) is the fraction of carcasses falling inside the searched area. The expected *dwp* is ψ, but the uncertainty in estimating *dwp* is greater than the uncertainty in ψ due to variation in the actual fraction of carcasses falling in the searched area given the probability of a carcass falling in the searched area. The problem is similar to flipping a coin 10 times. We know the probability of heads is 0.5, but the actual number of heads in 10 flips varies. estdwp accounts for that uncertainty.

When the *dwp* is estimated, it can be formatted[10], where nturb is the number of turbines and nsim is the number of simulation reps and used as a component in GenEst for estimating mortality.

```
dwphat <- estdwp(psihat, ncarc = getncarc(rings_shape))
exportGenEst(dwphat, file = "dwp.csv")
```

A series of more detailed examples that demonstrate how to run analyses on different data types and to highlight some of the difficulties that may be encountered in a *dwp* analysis are given in sections, "Examples 2–6."

Example 2—Vector of Distances

The simplest scenario is a vector of carcass distances coupled with a search radius. This framework assumes that all ground inside the search radius is searched at every turbine, the detection probability is the same around each turbine, and nothing else is searched. All that is required in this simplest scenario is a vector of distances and a known search radius.

Importing and Formatting Distance Vector Data

Bundled with the dwp package is an example dataset compiled from eagle carcass searches over several years at a large site. The searches were conducted with dogs out to a radius of approximately 100 m. We defined the search radius to be 100 m and assumed the plots were circular. The example dataset is formatted as a data frame, layout_eagle, which consists of three columns: DateFound, turbine, and r (which gives the distance from the carcass to the nearest turbine). The data frame structure is optional.

---

[10]For *dwp*, GenEst requires a .csv file with a column for turbine name and a column for *dwp* estimates for each turbine. The *dwp* estimates for each turbine may be either (1) point estimates (single number), or (2) simulated values that account for the uncertainty in the estimates. In the latter case, the turbine and *dwp* columns will be of length nturb×nsim.

```
# the first few lines of the eagle data set
head(layout_eagle)
#>      DateFound    turbine   r
#> 1    03-Aug-05    t21       23
#> 2    10-Oct-05    t43       46
#> 3    31-Oct-05    t14       25
#> 4    09-Apr-06    t25       30
#> 5    28-Apr-06    t16       65
#> 6    03-May-06    t21       42
```

- Required:

    1. Vector of distances, and

    2. Search radius.

- Assumptions:

    1. Circular search plots,

    2. No unsearched area inside search radius,

    3. Same search radius at all turbines, and

    4. Isotropic carcass distribution (same in all directions from the turbine).

After the raw data have been defined in a vector or a column in a data frame or array, they need to be formatted as a rings object (section, "Ring Structure").

```
# format by feeding the bare vector of distances to prepRing (with search radius)
rings_eagle <- prepRing(layout_eagle$r, srad = 100)
```

Fitting Carcass Distance Models

After the data have been formatted as a `rings` object, use `ddFit` to fit parametric distributions to the data.

```
eaglemod <- ddFit(rings_eagle)
#> Extensible models:
#>     xep1
#>     xep01
#>     xep2
#>     xep02
#>     xep12
#>     xep0123
#>     tnormal
#>     MaxwellBoltzmann
#> lognormal
#>
#> Non-extensible models:
#>     xep012
#>     xep123
#>     constant
```

Of the 12 standard models that are fit by default (app. 1), all but 3 are extensible for this dataset. With the exception of `xep1` and the `lognormal`, all the extensible models are in close agreement with each other and fall within a tight band (fig. 6). Because the search coverage within 100 m of a turbine is 100 percent and there is no unsearched area inside the search radius, the estimated *dwp* is the estimated fraction of carcasses falling within 100 m, which is the *y*-value of the cumulative distribution functions (CDFs) at $x = 100$ (fig. 6). Thus, it can be seen that all models except for `xep1` predict that more than 95 percent of the carcasses lie within 100 m, and thus there is little distinction among the models.
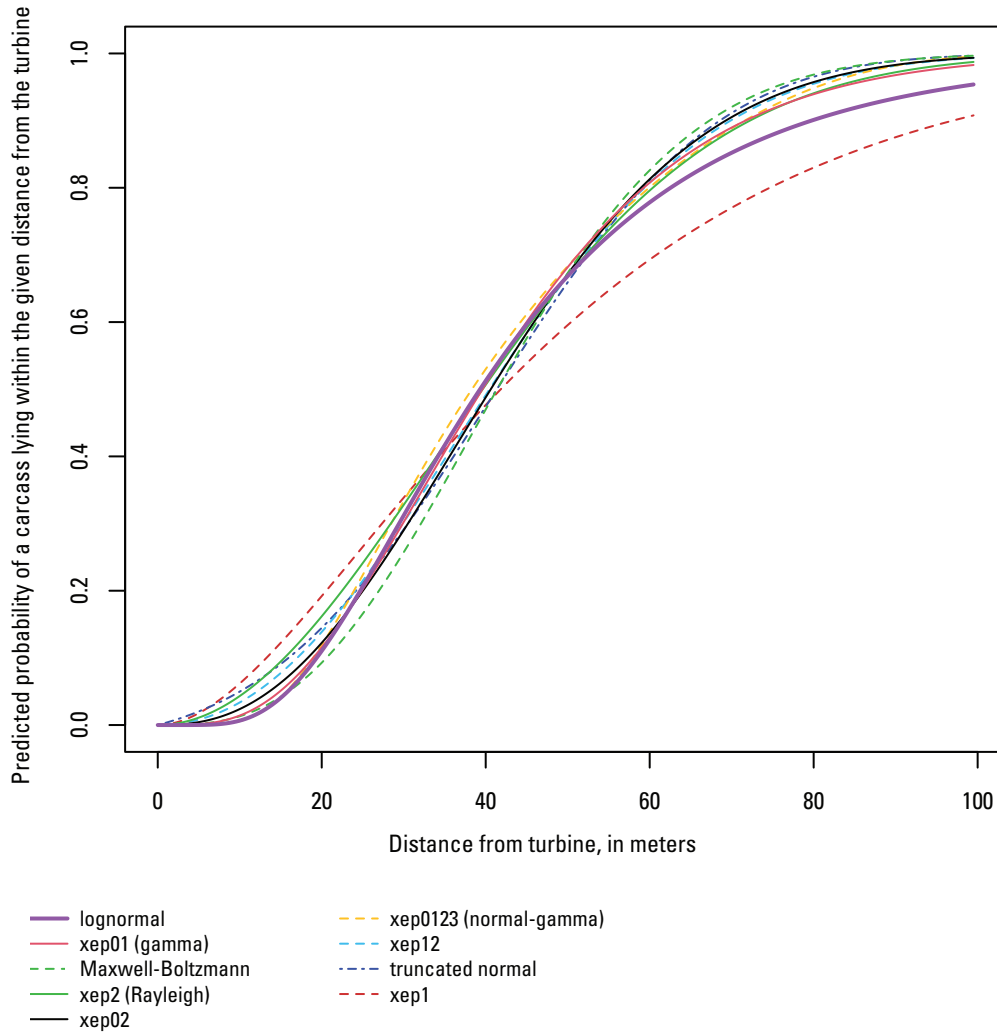
Figure 6.   Cumulative distribution functions for eagle carcass distributions. Figure drawn using the plot function from the dwp package with the fitted eagle models—`plot(eaglemod)`.

Model Selection

There appears to be little distinction among the fitted models for the eagle data beyond `xep1` and `lognormal` predicting somewhat lower ψ than the other distributions (fig. 6). The question is whether it is the `xep1`, `lognormal`, or one of the other seven models that is giving the most accurate prediction of ψ and *dwp*.

A primary aid in model selection is the function `modelFilter`, which gives a table of filtering test results, which are discussed in section, "Tools for Model Selection" and illustrated in section, "Examples." The filter gives a 0 (fail) or 1 (pass) for each test for each model, along with ΔAICc.

```
modelFilter(eaglemod)
#>                     extensible   rtail   ltail   aicc   hin   deltaAICc
#>    lognormal        1            1       1       1      1     0.000000
#>    xep01            1            1       1       1      1     1.118513
#>    MaxwellBoltzman  1            1       1       1      1     2.286072
#>    n xep2           1            1       1       1      1     2.638046
#>    xep02            1            1       1       1      1     3.208478
#>    xep0123          1            1       1       1      1     4.109567
#>    xep12            1            1       1       1      1     4.183636
#>    tnormal          1            1       1       1      1     8.123697
#>    xep1             1            1       1       1      1     8.681678
#>    xep012           1            1       1       1      1     2.863524
#>    xep123           1            1       1       1      1     4.859597
#>    constant         0            0       1       0      1     66.68849
```

All extensible models passed all the filtering tests (value = 1). The `lognormal` had the lowest AICc, but all the extensible models had ΔAICc < 10. When there is little to no unsearched area inside the search radius as with the eagle data, AICc is of little to no value for distinguishing among models with ΔAICc < 10 because the *dwp* prediction does not involve any interpolation but is strictly a matter of extrapolation to the area outside the search radius. In general, the heavy-tailed `lognormal` and `xep1` distributions tend to be less well-suited to eagle carcass distributions and should be used with caution (app. 3). The other fitted distributions are remarkably similar for the eagle data, with all predicting at least 98.3 percent of the carcasses inside the search radius, which is given as `p_win` in the summary `stats` table for the fitted models:

```
stats (eaglemod)
#>                      median   75%    90%    95%     mode    P_win   deltaAICc
#>    lognormal         39.3     57.0   79.7   97.4    29.0    0.954   0.00
#>    xep01             39.6     54.9   71.7   83.1    32.5    0.983   1.12
#>    MaxwellBoltzmann  41.4     54.5   67.3   75.2    38.1    0.997   2.29
#>    xep2              39.6     56.0   72.2   82.4    33.71   0.987   2.64
#>    xep02             40.6     55.1   69.3   78.1    36.3    0.994   3.21
#>    xep0123           38.3     54.4   71.6   80.3    26.7    0.995   4.11
#>    xep12             40.4     55.5   69.9   78.8    36.5    0.993   4.19
#>    tnormal           41.4     55.6   68.6   76.3    40.5    0.997   8.13
#>    xep1              41.8     67.1   97.0   118.3   24.9    0.908   8.68
```

Thus, regardless of which model we choose (excluding `xep1` and `lognormal`), $\widehat{dwp}$ will be practically the same, so we will select xep01 (`gamma`) because, among the non-heavy-tailed distributions, it is the top model in terms of AICc. The fitted parameters of this model are in the `parms` element of the corresponding model object. Obtaining confidence intervals for parameters is demonstrated in section, "Model Fitting and Analysis."

```
eaglemod$xep01$parms
#>    shape       rate
#>    4.06980861  0.09448663
```

Probability of a Carcass Lying Inside the Searched Area and Density-Weighted Proportion

The probability that a carcass will lie inside the searched area can be estimated using `estpsi`, which returns a vector of simulated $\hat{\psi}$ values that account for the uncertainty in estimation of ψ, and a confidence interval can be calculated using the R base function, `quantile`.

```
psihat <- estpsi(rings_eagle, model = eaglemod[["xep01"]])
# 90% confidence interval:
quantile(psihat, prob = c(0.05, 0.95))
#>    5%        95%
#>    0.943892  0.994859
```

Then, $\widehat{dwp}$ is calculated from the estimated `psihat` and the number of carcasses using `estdwp`. A confidence interval can once again be calculated using `quantile`.

```
dwphat <- estdwp(psihat, ncarc = getncarc(rings_eagle))
# 90% confidence interval:
quantile(dwphat, prob = c(0.05, 0.95))
#>      5%      95%
#>    0.923  1.000
```

As expected (app. 4), the confidence interval for *dwp* is wider than the confidence interval for ψ. The $\widehat{dwp}$ can now be formatted for GenEst and exported.

```
eagle_frm <- formatGenEst(dwphat)
exportGenEst(eagle_frm, file = "dwp_eagles.csv")
```

Example 3—Simple Geometry

If the search areas are fairly regular and are amenable to simple description as square, circular, or road and pad with a few defining parameters, the data can be imported and processed for analysis using `initLayout`, `prepRing`, and `addCarcass`. Unlike the vector of distances format, the simple geometry format can accommodate differences in search areas among turbines. However, it does not allow complex shapes or multiple search classes. A raw simple geometry dataset is simply a standard data frame with 3–6 columns, including the following:

- **turbine:** turbine IDs formatted as syntactically valid R names, which include only letters, numbers, underscores (_), and (or) periods (.) and do not begin with a number.

- **radius:** the search radius. If the search plot is an m × m square, the radius is its half-width, m. If the search plot is roads and pad, the radius is the maximum search distance from the turbine.

- **shape:** general descriptor of the shape of the search plot and must be one of "square", "circular", or "RP" (for road and pad searches).

- **padrad:** radius of the turbine pad, which is assumed to be circular; required when shape = "RP", otherwise optional (and ignored).

- **roadwidth:** width of the access road(s); required when shape = "RP", otherwise optional (and ignored).

- **n_road:** number of access roads; required when shape = "RP", otherwise optional (and ignored).

Importing and Formatting Site Layouts with Simple Geometry

Simple layouts can be read from .csv files using `initLayout` or can be entered by hand and then preformatted using `initLayout`. There is an example of a simple geometry site description (`layout_simple`) bundled into the package. The dataset has search plots with different shapes at each of four turbines (fig. 7).

```
data(layout_simple)
layout_simple
#>          turbine  radius   shape     padrad   roadwidth  n_road
#>    t1    t1        90      circular   NA       NA         NA
#>    t2    t2        65      square     NA       NA         NA
#>    t3    t3       120      RP         15       5          2
#>    t4    t4       100      RP         20       4          1
```

The `layout_simple` raw dataset is formatted as a data frame and needs to be preformatted using `initLayout` to convert it into a `simpleLayout` object which can be recognized by other dwp functions.

```
# initial format:
class(layout_simple)
#> [1] "data.frame"
```

```
# convert to simpleLayout:
layout_simple <- initLayout(layout_simple)
class(layout_simple)
#> [1] "simpleLayout" "data.frame"
```
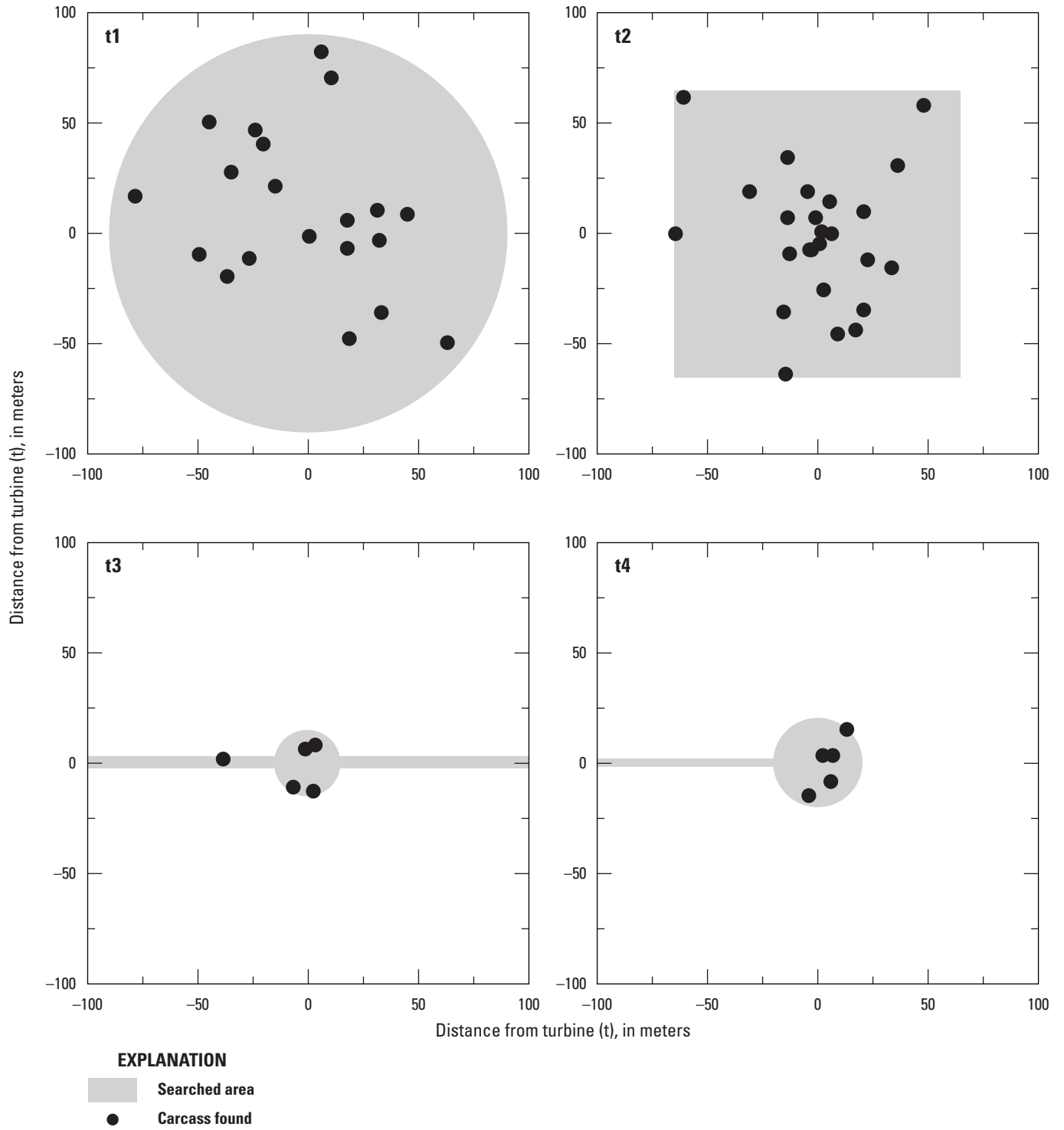
Figure 7. Site layout with simple geometry. Axes indicate coordinate grids in meters relative to turbine at (0, 0).

Once the site layout data have been preformatted using `initLayout`, use the `prepRing` function to process the layout data into 1 m concentric rings with the area (in square meters) searched in each ring at each turbine.

```
rings_simple <- prepRing(layout_simple)
```

After formatting the site layout data into rings, use `addCarcass` to incorporate carcass data into the `rings` data structure for analysis. The carcass data for simple geometry layouts should be organized in a data frame with each row representing a carcass and columns for the turbine ID and the distances from the turbine at which the carcasses were found.

```
# carcass data
head(carcass_simple)
#>      turbine    r
#> 1    t2         47.99
#> 2    t2         75.07
#> 3    t1         19.97
#> 4    t1         51.84
#> 5    t4         8.06
#> 6    t2         8.83

# adding carcasses to the formatted (but bare) site layout data
rings_simple <- addCarcass(carcass_simple, data_ring = rings_simple)
```

After adding carcasses to the site layout formatted as a `rings` object using `prepRing` and `add_carcass`, the carcass distribution models can be fit using `ddFit`.

```
dmod_simple <- ddFit(rings_simple)
#> Extensible models:
#>     xep1
#>     xep01
#>     xep2
#>     xep02
#>     xep123
#>     tnormal
#>     MaxwellBoltzmann
#>     lognormal
#>
#> Non-extensible models:
#>     xep12
#>     xep012
#>     xep0123
#>     constant
```

The lognormal distribution predicts only 30.6 percent of the carcasses to lie within the maximum search radius among the turbines (120 m), and the gamma predicts 82.8 percent. All other extensible models predict in excess of 95 percent (fig. 8) within 120 m. If there were very little unsearched area within 120 m, then the six extensible models other than lognormal and gamma would give the same >95 percent for $\widehat{dwp}$, and there would be no practical difference among them. However, there is a great deal of unsearched area within the maximum search radius at the site. The models are quite different from one another between about 40 and 100 m from the turbine, and the choice of model is likely to have a significant impact on the resulting *dwp* prediction.

Model Selection for the Simple Geometry Site

The first step in model selection is to use the `modelFilter` function.

```
modelFilter(dmod_simple)$scores
```

The fitted `lognormal` and gamma (`xep01`) distributions evidently have implausibly heavy right tails, as they both fail the `rtail` model filter test (table 2). There is at least one high-influence point for the `xep02` and `xep1` models, thus failing the hin test (table 2). Only three models—the truncated normal (`tnormal`), `xep123`, and Rayleigh (`xep2`)—pass all the diagnostic tests. The truncated normal and `xep123` CDFs are virtually indistinguishable, whereas the Rayleigh differs markedly (fig. 8). Because there is a great deal of unsearched area at the site, prediction of ψ and *dwp* relies strongly on interpolation, so a good fit within the range of the data is crucial, and models with large AICc scores—like the Rayleigh (`xep2`)—can be eliminated. The truncated normal has the lowest AICc score among the models that pass all the diagnostic tests, and that is the one we select for calculating $\widehat{dwp}$ for these example data.
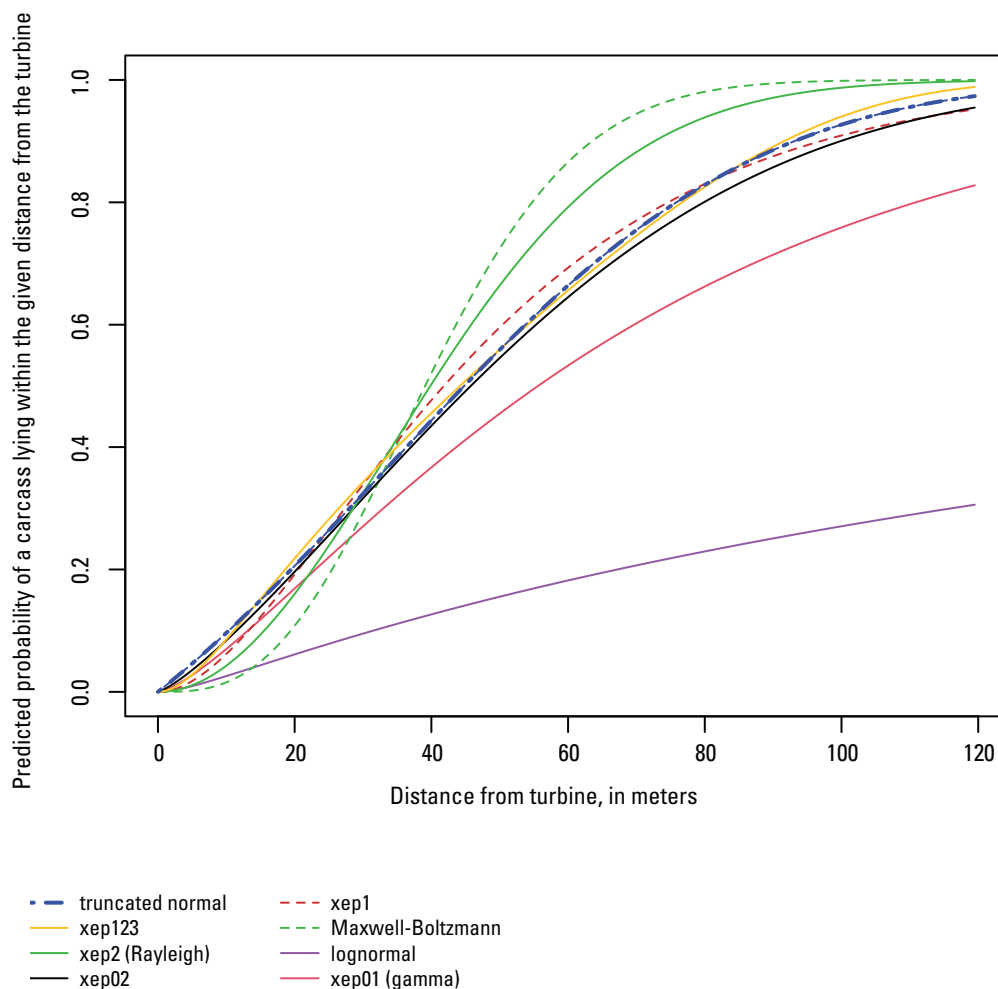
Figure 8. Fitted cumulative distributions of carcasses at the simple geometry site. Only the extensible models are shown. Figure drawn using the plot function from the dwp package with the fitted models: `plot(dmod_simple)`.

Table 2. Model filter for simple geometry site.

[Table shows model filter results where 1 = filter test passed and 0 = filter test failed. Tests are described in section, "Tools for Model Selection." **Extensible**: Extensibility test. **rtail** and **ltail** = heavy r(ight) and l(eft) tail tests. **aicc:** ΔAICc less than 10. **hin:** High influence test. **ΔAICc:** Difference between AICc value for a given model and the lowest AICc value among the models tested. (AICc is the Akaike information criterion corrected for finite sample size.)]

Model	Extensible	rtail	ltail	aicc	hin	ΔAICc
truncated normal	1	1	1	1	1	1.18
xep123	1	1	1	1	1	2.78
xep2	1	1	1	1	1	8.11
xep02	1	1	1	1	0	0.37
xep1	1	1	1	1	0	0.75
Maxwell-Boltzmann	1	1	1	0	1	38.47
lognormal	1	0	1	1	1	0.00
xep01	1	0	1	1	0	0.10
xep12	0	0	1	1	1	1.05
xep012	0	0	1	1	0	2.22
xep0123	0	0	1	1	0	4.40
constant	0	0	1	0	1	43.52

Estimated Probability of a Carcass Lying Inside the Searched Area and Estimated Fraction of Carcasses Lying Inside the Searched Area for the Simple Geometry Site

Having selected the truncated normal distribution (`tnormal`) for each turbine, we can predict the probability that a carcass lies inside the searched area using `estpsi`. Because `estpsi` integrates the selected model over the searched area (as with the "volcano" in fig. 2), both the searched area (formatted as `rings_simple`) and the selected model are required as arguments to `estpsi`.

```
psihat_simple <- estpsi(rings_simple, model = mod_simple["tnormal"])
```

The prediction of *dwp* depends not just on the probability of a carcass lying inside the searched area (ψ) but also on the number of carcasses discovered inside the searched area, so the `estdwp` function requires both $\hat{\psi}$ and the carcass counts.

```
dwphat_simple <- estdwp(psihat_simple, ncarc = getncarc(rings_simple))
```

The estimated *dwp*'s for each turbine can now be plotted (fig. 9) and exported to a .csv file for use in GenEst.

```
plot(dwphat_simple)
exportGenEst(dwphat_simple, file = "dwphat_simple.csv")
```

Example 4—R Polygons

R Polygon Site Layouts—Data

If the shape of the search plots is somewhat complex, but there is only one search class, you may import your site layout data in R polygon format. An R polygon is simply two columns of data giving the *x* and *y* coordinates of the vertices of a polygon. For *dwp* analysis, the polygons must be associated with turbines in a standard data frame with three columns:

- **turbine:** a turbine ID, which must be a syntactically valid name in R and include letters, numbers, dots (.), and underscores (_) only. Spaces, hyphens, commas, pound signs, and other special characters are not allowed. Syntactically valid names must not begin with a number or with a dot followed by a number.

- **x:** x coordinates of the vertices of the polygons delineating search areas at each turbine. All coordinates are assumed to be in meters with the turbine referenced in the 'turbine' column assumed to be at (0, 0).

- **y:** y coordinates of the vertices of the polygons.

There must be one polygon for each turbine searched, and polygons must not be self-intersecting. A polygon site layout dataset with R polygons is included as part of the dwp package and can readily be loaded and viewed:

```
data(layout_polygon)
layout_polygon
#>       turbine   x            y
#> 1     t1        4.814164     69.209213
#> 2     t1        -12.278842   17.930197
#> 3     t1        -64.255532   15.139502
#> 4     t1        -54.836937   -54.627868
#> 5     t1        22.953680    -27.767431
#> 6     t1        70.395491    14.092991
#> 7     t1        20.162985    19.674381
#> 8     t2        -19.604416   -3.000014
#> 9     t2        -6.348615    -17.302325
#> 10    t2        14.581596    -2.302341
#> 11    t2        6.907185     10.255786
#> 12    t2        2.372306     25.953444
#> 13    t2        -1.116063    52.465045
#> 14    t2        -5.302105    58.744108
#> 15    t2        -17.860231   68.511540
#> 16    t2        -23.441621   81.767340
#> 17    t2        -35.650911   76.185950
#> 18    t2        -25.883479   33.627855
#> 19    t2        -35.999748   9.906949
```
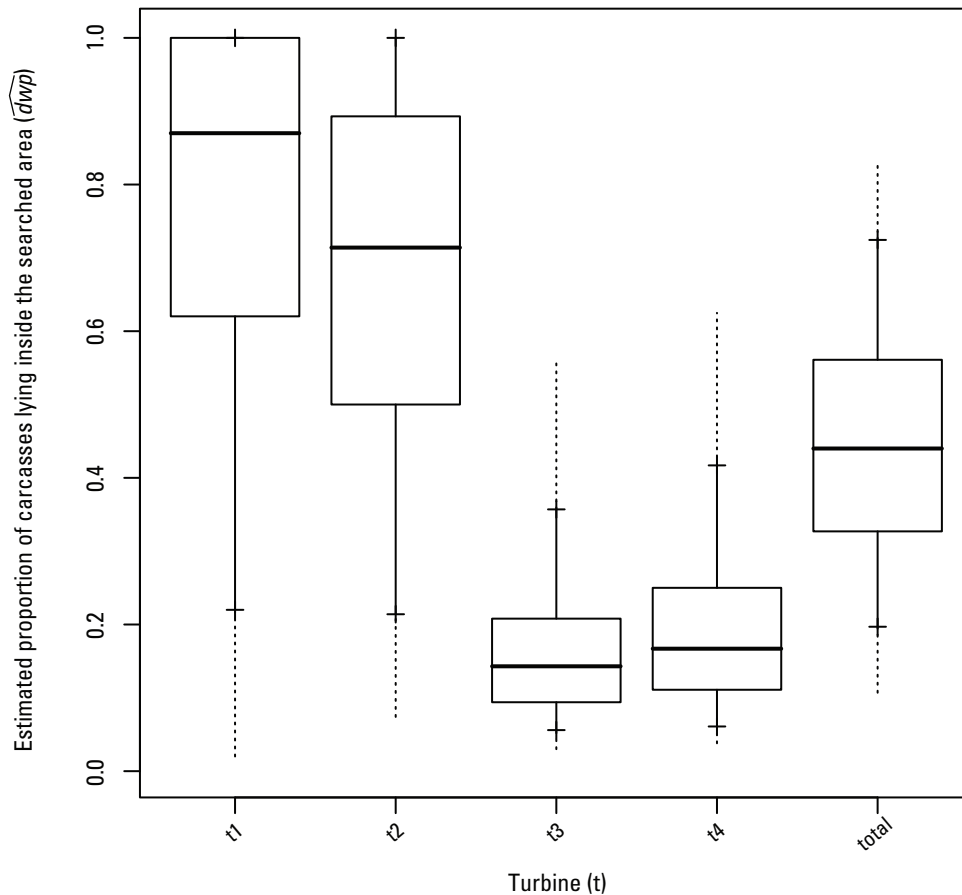
Figure 9.   Estimated proportion of carcasses lying inside the searched area for all turbines and the site as a whole for the simple geometry site. Figure drawn using the `plot` function from the dwp package with the estimated *dwp* values: `plot(dwphat_simple)`.

To begin the analysis, the polygons must be converted to a `polygonLayout` object which has been error-checked and converted to a properly formatted list. This can be done using the `initLayout` function with argument `dataType = "polygon"`. Once the polygon layout has been formatted, it can be plotted (fig. 10) and prepared for GLM analysis by tallying the searched area in concentric 1m rings around each turbine using the function `prepRing`.

```
# initial formatting
playout <- initLayout(layout_polygon, dataType = "polygon")

# plot
plot(playout, las = 2)

# create a ring structure
rings_polygon <- prepRing(playout)
```

A simulated carcass dataset for this polygon site (`carcass_polygon`) is included in the package. It is a simple, two-column data frame with each row representing a carcass and columns for turbine ID (`turbine`) and distances from the turbine (`r`, in meters) at which carcasses were discovered in the carcass surveys. The carcass data are added to the ring structure via `addCarcass` to complete the data preparations for analysis.

```
# the first few lines of the carcass_polygon data
head(carcass_polygon)
#>      turbine    r
#> 1    t1         32.5
#> 2    t1         38.2
#> 3    t1         45.1
#> 4    t2         5.6
#> 5    t1         12.4
#> 6    t1         29.5
```
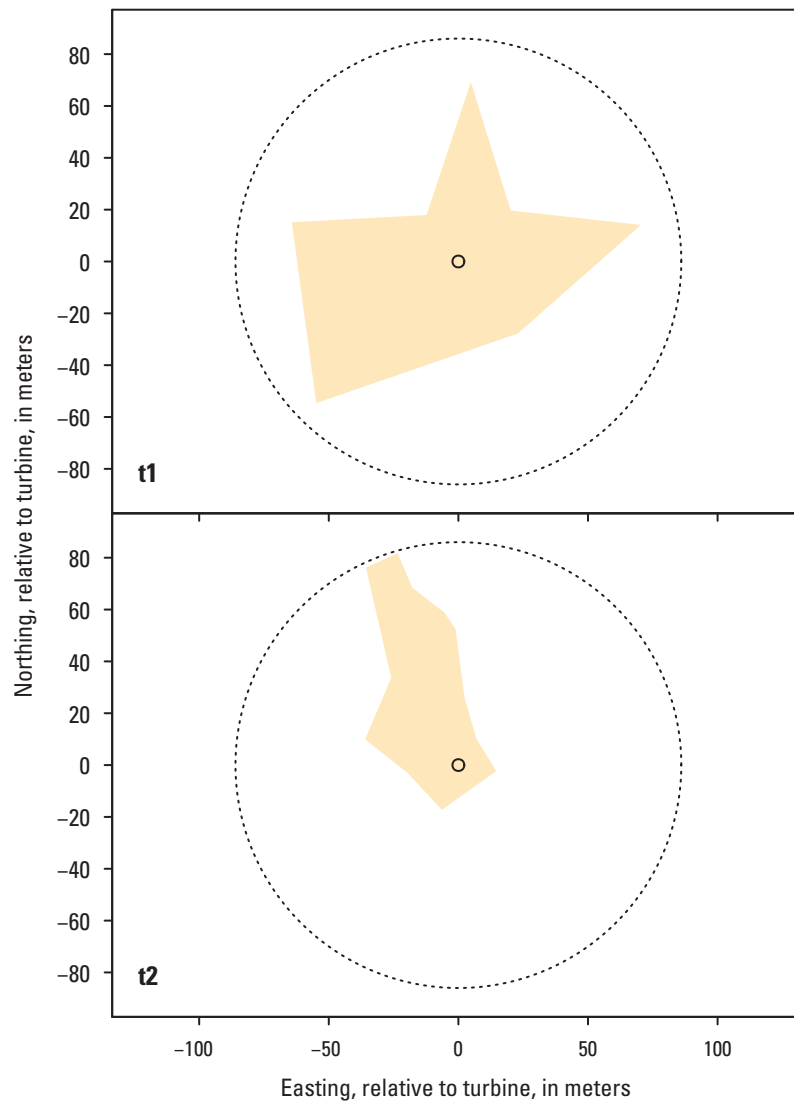

Figure 10.   Search plots for R polygon dataset. Figure drawn using the plot function from the dwp package with the formatted site layout data: `plot(playout)`. t1 and t2, turbines 1 and 2.

R Polygon Site Layouts—Carcass Distribution Modeling and Density-Weighted Proportion

After adding carcasses to the formatted `rings_polygon`, carcass distance models can be fit using `ddFit` and assessed using these functions:

- `stats`, which gives descriptive statistics for carcass distances (table 3),

- `modelFilter`, which performs diagnostic tests on the fitted models and ranks them by suitability,

- `plot`, which creates graphs for fitted PDFs or CDFs for fitted carcass distributions, and

- `aic`, which calculates AICc scores for fitted distributions.

For details on the use of these and other useful functions in the package, enter `?dwp`.

Table 3.   Descriptive statistics for fitted carcass distributions at the R polygon site.

[Statistics are for the estimated quantiles of carcass distances. For example, **Median** gives the estimated distance within which 50% of the carcasses are expected to lie, according to the model; and **75%**, **90%**, **95%** give the estimated distance within which 75%, 90%, and 95% of the carcasses are expected to lie, respectively. **Mode** gives the estimated distance from the turbine at which the most carcasses are likely to lie, and **p_win** gives the estimated probability that a carcass will lie inside the search radius. **ΔAICc:** Difference between AICc value for a given model and the lowest AICc value among the models tested. (AICc is the Akaike information criterion corrected for finite sample size.) %, percent]

Model	Median	75%	90%	95%	Mode	p_win	ΔAICc
xep02	38.5	59.7	81.1	94.7	26.1	0.919	0.00
xep01	53.6	93.3	142.1	177.7	22.4	0.712	0.31
tnormal	35.3	51.8	67.2	76.6	31.3	0.977	0.61
xep1	36.1	58.0	83.8	102.2	21.5	0.906	1.26
lognormal	11,626.2	61,620.5	276,438.8	678,767.5	25.7	0.023	1.75
xep012	37.5	57.3	77.0	89.3	27.2	0.938	2.18
xep123	36.9	56.2	72.2	80.9	17.9	0.968	4.30
xep2	31.7	44.8	57.7	65.9	26.9	0.994	7.87
MaxwellBoltzmann	29.7	39.1	48.2	53.9	27.3	1.000	46.10

```
# fit the distance models
dmod_polygon <-ddFit(rings_polygon)
#> Extensible models:
#>     xep1
#>     xep01
#>     xep2
#>     xep02
#>     xep012
#>     xep123
#>     tnormal
#>     MaxwellBoltzmann
#>     lognormal
#>
#> Non-extensible models:
#>     xep12
#>     xep0123
#>     constant
stats(dmod_polygon)
#>                    median    75%      90%       95%        mode    P_win   deltaAICc
#>   xep02            38.5      59.7     81.1      94.7       26.1    0.919   0.00
#>   xep01            53.6      93.3     142.1     177.7      22.4    0.712   0.31
#>   tnormal          35.3      51.8     67.2      76.6       31.3    0.977   0.61
#>   xep1             36.1      58.0     83.8      102.2      21.5    0.906   1.26
#>   lognormal        11626.2   61620.5  276438.8  678767.5   25.7    0.023   1.75
#>   xep012           37.5      57.3     77.0      89.3       27.2    0.938   2.18
#>   xep123           36.9      56.2     72.2      80.9       17.9    0.968   4.30
#>   xep2             31.7      44.8     57.7      65.9       26.9    0.994   7.87
#>   MaxwellBoltzmann 29.7      39.1     48.2      53.9       27.3    1.000   46.10
```

The extensible models give a broad range of predictions of the probability of a carcass lying inside the search radius, from a low of 2.3 percent for the lognormal distribution to a high of 100 percent for the Maxwell-Boltzmann. The `modelFilter` provides useful guidance on model selection (table 4).

```
modelFilter(dmod_polygon)$scores
```

The lognormal model failed the right tail test because it predicts implausibly low probabilities of carcasses lying within 80, 120, 150, and 200 m, whereas the Maxwell-Boltzmann model failed the left tail test because it predicts implausibly high probability of carcasses lying within 50 m.

Table 4.   Model filter for simple geometry site.

[Table shows model filter results where 1 = filter test passed and 0 = filter test failed. Tests are described in section, "Tools for Model Selection." **Extensible**: Extensibility test. **rtail** and **ltail** = heavy r(ight) and l(eft) tail tests. **aicc:** Δ**AICc** less than 10. **hin:** High influence test. Δ**AICc:** Difference between AICc value for a given model and the lowest AICc value among the models tested. (AICc is the Akaike information criterion corrected for finite sample size.)]

Model	Extensible	rtail	ltail	aicc	hin	ΔAICc
tnormal	1	1	1	1	1	0.61
xep1	1	1	1	1	1	1.27
xep123	1	1	1	1	1	4.31
xep2	1	1	1	1	1	7.87
xep02	1	1	1	1	0	0.00
xep012	1	1	1	1	0	2.18
Maxwell-Boltzmann	1	1	0	0	1	46.10
xep01	1	0	1	1	0	0.31
lognormal	1	0	1	1	0	1.76
xep12	0	0	1	1	1	2.45
xep0123	0	0	1	1	0	4.01
constant	0	0	1	0	1	42.47

```
pdd(c(80, 120, 150, 200), model = dmod_polygon["lognormal"])
#> [1] 0.02202108 0.03217810 0.03924999 0.05018037
pdd(c(20, 50), model = dmod_polygon["MaxwellBoltzmann"])
#> [1] 0.2167363 0.9184358
```

The `xep02` and `xep012` models fail the `hin` test (table 4) because they have high influence points, that is, points that are out of line with the others according to the model but that have a significant impact on the shape of the fitted curve. This is a red flag that signals an instability in the model for the given data.

The remaining fitted distributions (`tnormal`, `xep1`, `xep123`, `xep2`) are broadly comparable according to the diagnostic criteria tested. However, the distributions differ markedly between 30 m and the maximum search radius of 85 m (fig. 11), so other considerations come into play. In particular, there is a tendency for the light-tailed distributions to overestimate ψ when the search radius is short or moderate and there is substantial unsearched area inside the search radius as with this example (app. 3), so the Rayleigh would not be recommended here. Also, when there is substantial unsearched area, it is important to have a low AICc score, so `tnormal` and `xep1` are to be commended over `xep123`.
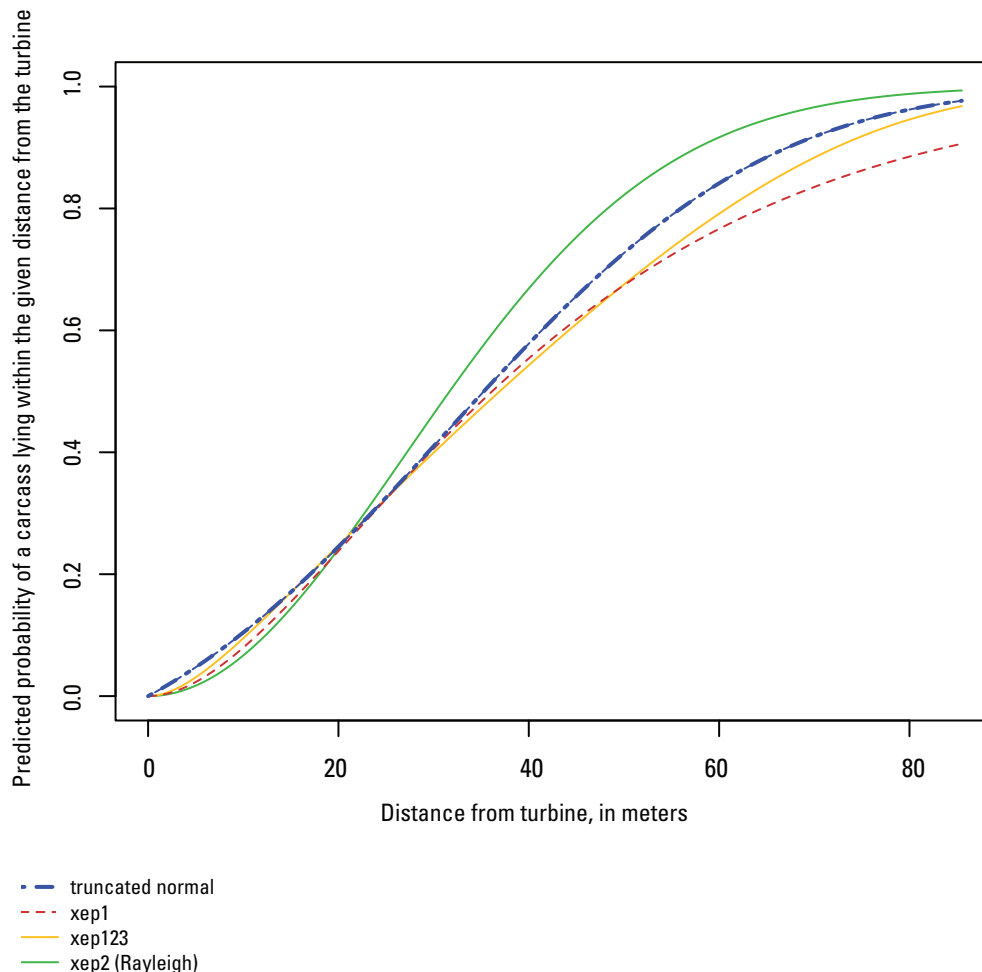

Figure 11.   Cumulative distribution functions for the truncated normal, xep1, xep123, xep2 (Rayleigh) models fit to the R polygon dataset. Figure drawn using the plot function from the dwp package with a subset of the fitted models—`plot(dmod_polygon[passind])`, where passind is a vector of names of the models to be plotted.

The two remaining models (`tnormal` and `xep1`) have AICc scores differing by only 0.65, well below a minimal threshold to be statistically meaningful. The `xep1` distribution has a heavier tail (fig. 11) but the truncated normal has much more frequency of very small $\hat{\psi}$ values than does `xep1` (fig. 12) and therefore a less stable fit.

After selecting the `xep1` model, we can calculate $\widehat{dwp}$ and export for use in GenEst or other programs using this parameter estimate.

```
# estimate probability of carcass lying in searched area
psipoly <- estpsi(rings_polygon, model = dmod_polygon["xep1"])
dwp <- estdwp(psipoly, ncarc = getncarc(rings_polygon))
exportGenEst(dwp, file = "dwp_poly.csv")
```
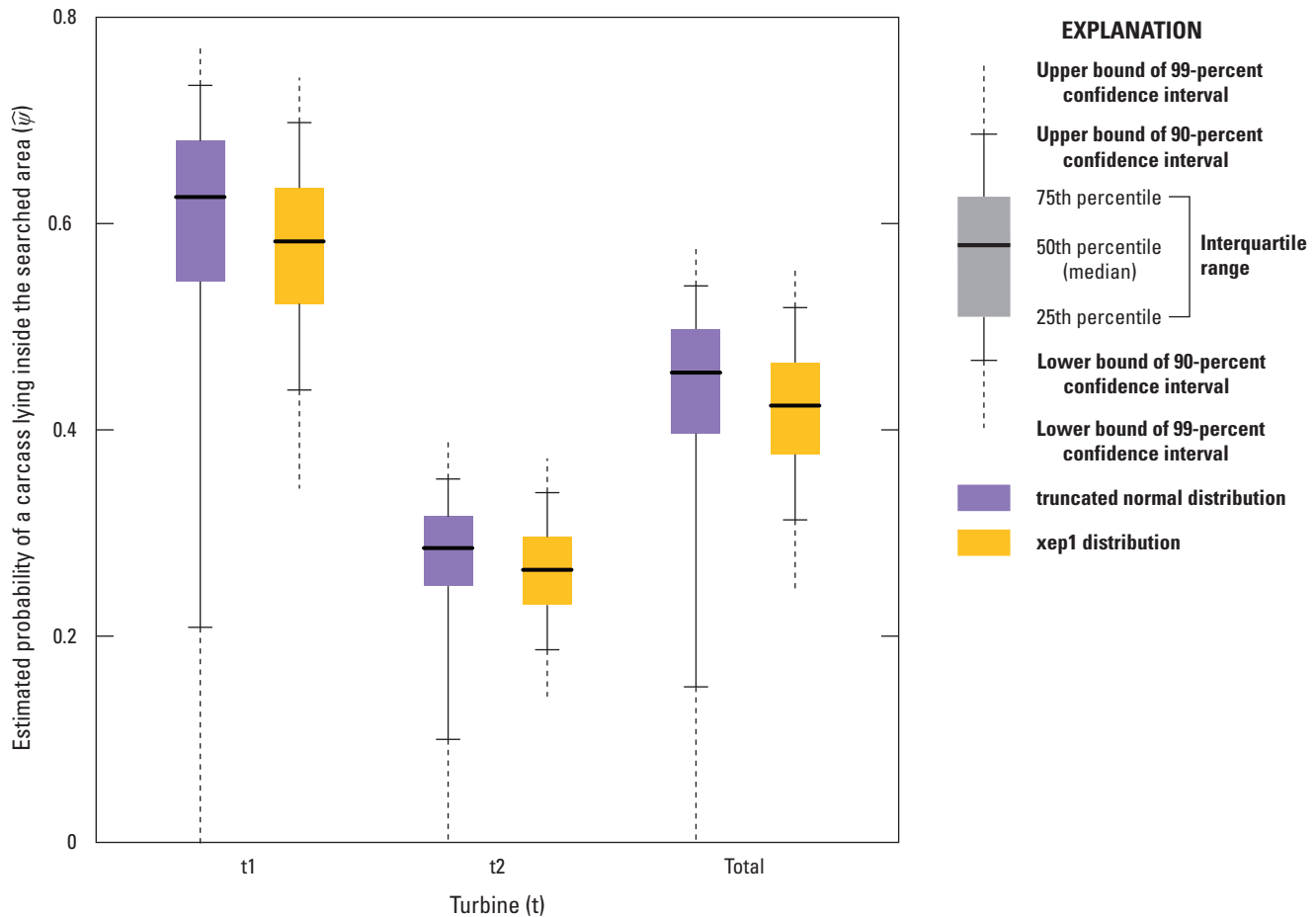

Figure 12.   Estimated probability of a carcass lying inside the searched area, for all turbines and the polygon site as a whole for the truncated normal and xep1 distributions.

Example 5—The (`x`, `y`) Grid Data

The dwp package includes an example dataset with site layout from a fictitious site with one turbine and a road and pad search area. The site layout data are formatted as an (x, y) grid, which gives the coordinates (in meters relative to the turbine at (0, 0)) of each square meter of searched ground at the turbine. The data are stored in a data frame, `layout_xy`, with columns for x and y coordinates, the number of carcasses (`ncarc`) in the square meter cell centered at each grid node, the distance from the turbine (`r`), and the turbine name.

```
# the first few lines of the xy data layout that is bundled with dwp package
head(layout_xy)
#>     x     y     ncarc   r            turbine
#> 1   0    -15    0       15.00000     t1
#> 2  -5    -14    0       14.86607     t1
#> 3  -4    -14    0       14.56022     t1
#> 4  -3    -14    0       14.31782     t1
#> 5  -2    -14    0       14.14214     t1
#> 6  -1    -14    0       14.03567     t1
xy_formatted <- initLayout(layout_xy, dataType = "xy")
```

Given gridded site layout data, `initLayout` returns an `xyLayout` object, which is a list with components:

1. `xydat` data frame with turbine, (*x, y*) grid coordinates relative to turbine, number of carcasses in each grid cell (`ncarc`), distance from turbine to center of grid cell (`r`), and (optional) columns for covariates or other identifiers;

2. `tcenter` array of turbine locations (*x, y*), with row names = turbine IDs;

3. `ncarc` vector of carcass counts at each turbine, with element names = turbine IDs;

4. `unitCol` name of the column in `xydat` that has turbine IDs; and

5. `tset` vector of names of turbines that were searched.

The analysis of gridded data differs slightly from the ring data. The models are fit on 1 m² cells rather than rings. Compared to the rings format, the grid format is more cumbersome to work with, runs more slowly, and does not offer as wide of a selection of pre-formatted models. The advantage of the grid structure is that allows for more sophisticated modeling options. Of particular interest would be accounting for anisotropy, which is variability in carcass distribution that depends on direction, as might happen if a prevailing wind direction strongly affects carcass dispersion patterns. These anisotropic models are discussed by Maurer and others (2020) and not addressed here or in the software.

The *xy*-grid dataset poses special problems because few carcasses (n = 15) were found at a single turbine searched on road and pad only (fig. 13). Although 65 percent of the carcasses fell beyond 30 meters (m), the maximum distance at which any carcass was found was 29.1 m. With such a low probability of carcasses lying on roads at distances beyond 30 m or so, it is difficult to get a representative sample with road and pad searches unless the number of carcasses is large. If the sample includes no carcasses well beyond the pad radius, then the fitted models will tend to overestimate *dwp* because they have difficulty "seeing" the distant carcasses. If, however, the sample does include carcasses well beyond the turbine pad, the fitted models will tend to underestimate *dwp* because they over-estimate the number of carcasses at great distances. On average, the tendencies to over- or under-estimate balance out, but on any particular dataset, one of the two is likely to be acting. If there are carcasses at great distance, then the light-tailed models are preferred because they provide some degree of insurance that the predicted densities decrease to zero rapidly beyond the range of the data. If there are no carcasses at great distance, then the heavy-tailed models are preferred in order to prevent the distribution from collapsing to zero too rapidly and not properly accounting for carcasses falling beyond the farthest carcass observed.

The fitted models:

```
dmod_xy <- ddFit(xy_formatted)
#> Extensible models:
#>     xep1
#>     xep01
#>     xep2
#>     xep02
#>     xep12
#>     xep012
#>     xep123
#>     xep0123
#>
#> Non-extensible models:
#> none
plot(dmod_xy)
```
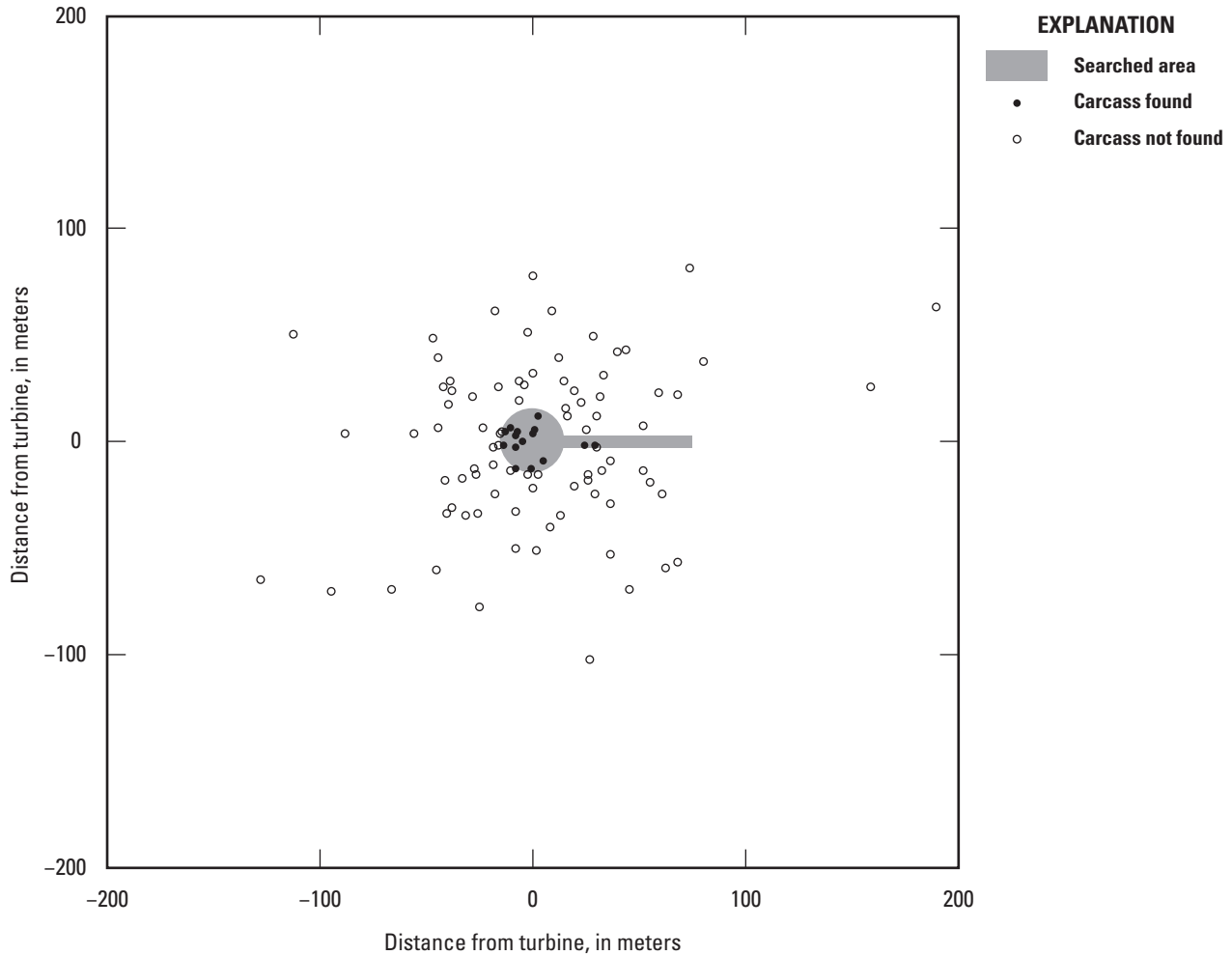
Figure 13.   Layout of *xy* grid turbine data with carcasses, showing the distribution of all carcasses in the simulation, both those that were found in the searched area and those that were not found.

With the gridded dataset, there was enormous discrepancy among the fitted models (fig. 14), but the `modelFilter` winnows out all but two of the models (table 5):

```
modelFilter(dmod_xy)$scores
```

The `xep1` and `xep01` distributions are the only ones that do not predict that more than 90 percent of the carcasses lie within 50 m of the turbine and thus pass the `ltail` filter (table 5). The estimated 90th percentile of carcasses distances according to the `xep01` model is 56.6 m[11], which is just beyond the cutoff of 50 m for the left-tail test and largely in line with the other, rejected models. By contrast, the `xep1` model has a slightly better AICc, which, all else being equal, is a reasonable tie-breaker for model selection.

As a postscript, note that the actual distribution that the carcasses were generated from (namely, gamma($\alpha = 1.774$, $\beta = 28.17$), where β is the scale parameter) has known average proportion of 79.5 percent of carcasses falling inside the search radius of 75 m. The `xep1` model estimated[12] 80 percent, which is extremely close to the expected fraction. None of the other models was even close. All were estimated in excess of 96.5 percent within 75 m.

Selecting the `xep1` model for estimating ψ and *dwp*, we get:

```
psi_xy <- estpsi(xy_formatted, model = dmod_xy["xep1"])
dwp_xy <- estdwp(psi_xy, ncarc = getncarc(xy_formatted), forGenEst = TRUE)
```

---

[11]This can be calculated as `qdd(0.9, dmod_xy["xep01"])`.

[12]The estimated proportion falling within a given radius, x, can be calculated as `pdd(x, dmod_xy["xep1"])`.
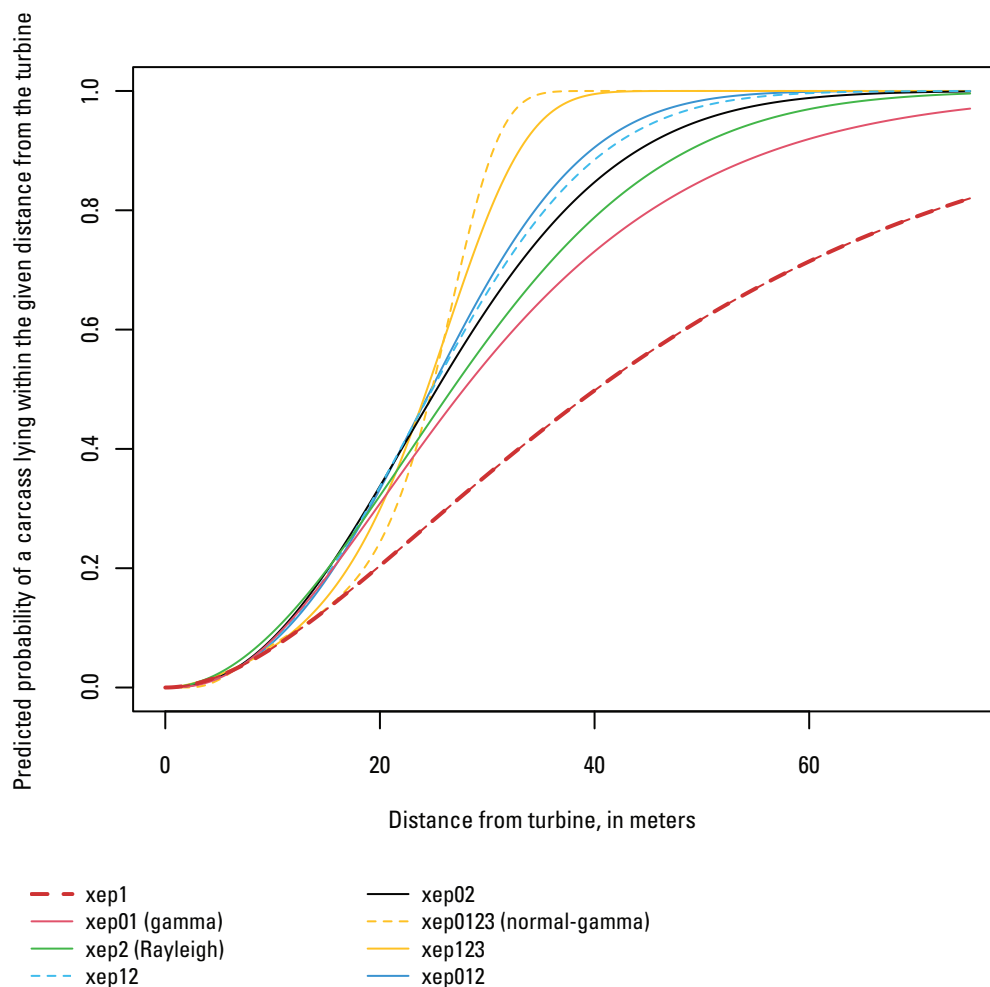
Figure 14. Cumulative distribution functions of fitted models for the *xy* grid data. Figure drawn using the plot function from the dwp package with the fitted models—`plot(dmod_xy)`, where `passind` is a vector of names of the models to be plotted.

Table 5. Model filter for site with gridded layout data.

[Table shows model filter results where 1 = filter test passed and 0 = filter test failed. Tests are described in section, "Tools for Model Selection." **Extensible**: Extensibility test. **rtail** and **ltail** = heavy r(ight) and l(eft) tail tests. **aicc:** ΔAICc less than 10. **hin:** High influence test. **ΔAICc:** Difference between AICc value for a given model and the lowest AICc value among the models tested. (AICc is the Akaike information criterion corrected for finite sample size.)]

Model	extensible	rtail	ltail	aicc	hin	ΔAICc
xep1	1	1	1	1	1	0.90
xep01	1	1	1	1	1	2.11
xep012	1	1	0	1	0	3.58

Example 6—Site Layout and Carcass Data from ShapeFiles—The Casselman Data

Compared to the simpler data structures (vector, simple geometry, and polygons), shapefiles allow for greater power to account for (1) variation in detection probabilities among search classes and (2) interacting covariates that affect the distances that carcasses fly but do require more sophisticated mapping and data management procedures. However, unlike the *xy*-grid format, the shapefiles do not lend themselves well to accounting for anisotropy.

We use the Casselman data to illustrate some of the analyses that can be performed using the dwp package with data from shapefiles. The Casselman dataset was collected at the Casselman Wind Project in Pennsylvania in 2008 (Arnett and others, 2009a, 2009b). It includes carcass counts for several bat species found at 22 turbines. Search areas and locations of carcass discoveries were mapped by a global positioning system and stored in Geographic Information System (GIS) shapefiles. Search areas were classed by the search conditions (`Easy`, `Moderate`, `Difficult`, `Very Difficult`). Unsearched areas also are delineated in the shapefiles and classed as Out. For this analysis, it is assumed that the distances that carcasses fly after being struck by a turbine blade are not affected by the search conditions on the ground (Huso and Dalthorp, 2014). However, the probabilities of observing carcasses that fall at a given distance will differ depending on search class, so the search class must be included in the models as a non-interacting covariate.

The dataset also includes interacting covariates that, unlike search class, affect the distances that carcasses fly. We consider two of those—turbine operations (curtailed at low wind speeds or not) and carcass size—in sections, "Additional Analysis for an Interacting Covariate" and "Carcass Size."

Shapefiles—Data

The `initLayout` function reads GIS shapefiles using the `st_read` function from the sf (simple features) R package for importing and managing GIS data. The importing of the shapefiles and subsequent formatting of the data for dwp is done behind the scenes in the package and does not assume that the user has any familiarity with the sf package. However, users with a good working knowledge of sf may be able to use that to their advantage in managing the data and displaying results.

The importing and formatting of shapefile data involves four steps in R:

1. `initLayout` reads the site layout data into a `shapeLayout` structure,

2. `prepRing` pre-processes the site data into a `rings` for analysis,

3. `readCarcass` imports carcass observations data from a shapefile, and

4. `addCarcass` adds carcass location data to the `rings` object.

To read shapefiles, the `initLayout` function requires "simple features" files, including a polygon or multi-polygon shapefile that delineates the area searched at each turbine and a points file that gives the locations of the turbines on the same coordinate system as the polygons. The shapefiles must have their three standard component files (.shp, .shx, and .dbf) stored in the same folder.

The polygons and turbine point files must have a column with turbine IDs, which must be syntactically valid R names, that is, contain combinations of letters, numbers, dots (.), and underscores (_) only and not begin with a number or a dot followed by a number. Other characters are not allowed: no spaces, hyphens, parentheses, etc. are allowed in turbine IDs.

Search areas may be defined for any number of search classes that affect detection probabilities on the ground but are not expected to affect the distances that carcasses fly after being struck, for example, vegetation type or ground texture. Interacting covariates that affect the actual distances that carcasses fly—like wind speed, turbine height, carcass size—present special difficulties that are difficult to model using the kinds of automated procedures used in most of the convenient dwp functions and are not implemented in the basic package functions. Savvy R users may be able to use the dwp functions as a basis for more elaborate analyses, including accounting for anisotropy (or carcass dispersion patterns that vary with direction from turbines) (Maurer and others, 2020) or analyzing dependence of dispersion on other interacting variables.

The shapefiles for site layout should include at a minimum a turbine ID and a "geometry" that gives the coordinates of the vertices for polygons that delineate the search areas. Other covariates (for example, search class, species, size, turbine operational constraints) are optional. When a shapefile is read into R using `sf::st_read` (as with dwp's `initLayout`), it should look something like the following, showing necessary geographic information in a header and a summary of the first 10 polygons or "features":

```
Simple feature collection with 178 features and 3 fields
geometry type: MULTIPOLYGON
dimension:  XY
bbox: xmin: 658608.2 ymin: 4411682 xmax: 662782 ymax: 4415250
projected CRS: NAD83 / UTM zone 17N
First 10 features:
OBJECTID  Turbine      Class      geometry
1         1            1          Out MULTIPOLYGON (((658822.4 44...
2         2            2          Out MULTIPOLYGON (((658747.7 44...
3         3            3          Out MULTIPOLYGON (((658717 4414...
4         4            4          Out MULTIPOLYGON (((658789.2 44...
5         5            5          Out MULTIPOLYGON (((658847.6 44...
6         6            6          Out MULTIPOLYGON (((658835.4 44...
7         7            7          Out MULTIPOLYGON (((658873 4413...
8         8            8          Out MULTIPOLYGON (((658771.2 44...
9         9            9          Out MULTIPOLYGON (((659276.5 44...
10        10           10         Out MULTIPOLYGON (((659249.2 44...
```

Shapefiles are read into R using the `initLayout` function with the names of the search files (including the file path if they are not stored in your working directory in R) along with the name of the column that contains turbine IDs. Other arguments for `initLayout` are for reading other file and data formats and are ignored when reading shapefiles.

Notice that the turbine IDs (1, 2, …) in the raw data file (shown above) are not syntactically valid R names. `initLayout` will convert these to simple, syntactically valid names, and alert the user that conversion has taken place. In general, users should use the `initLayout` function rather than `sf::st_read` directly for importing the shapefiles because `initLayout` employs some additional error-checks and formats the data for further processing and analysis:

```
# import site layout and carcass observation distances from shapefiles
layout_shape <- initLayout(data_layout = "searchpoly.shp",
  file_turbine = "turbine_pts.shp", unitCol = "Turbine")

cod <- readCarcass("carcasses.shp", unitCol = "Turbine")
```

The polygons in the `searchpoly.shp` shapefile are expressed as collections of (x, y) coordinates in meters relative to a reference point. In this example, there are a total of 178 polygons (or "features") that delineate distinct, non-intersecting search areas at 23 turbines. Search classes include `Easy`, `Moderate`, `Difficult`, `Very Difficult`, and `Out`. Maps of the searched areas at three of the turbines are shown in figure 15.

After the data are imported into R, they must still be formatted for analysis:

```
rings_shape <- prepRing(layout_shape, scVar = "Class", notSearched = "Out")
```

For a large site with multiple search classes, data formatting may take several minutes. The result is an S3 object of class `rings`, which features a characterization of the searched areas at each turbine, including the total area and number of carcasses in each search class in each concentric, 1-m ring from the turbine to the search radius (stored in `rdat`, which is used in fitting the distance models) along with some other data. The ring structure is efficient and convenient for analysis. A full description of the data structure is provided in the section, "Rings Structure."

After formatting the search plot configurations into rings for analysis, carcasses must be added to the rings. If the search plots are delineated into separate search classes, carcasses must be added by a two-step process of (1) importing the shape data, and (2) adding the data to the rings. If there are no search class distinctions, then carcasses may be added via a data frame with columns for turbine ID (`unitCol`) and for the carcass distances from turbine (`r`).
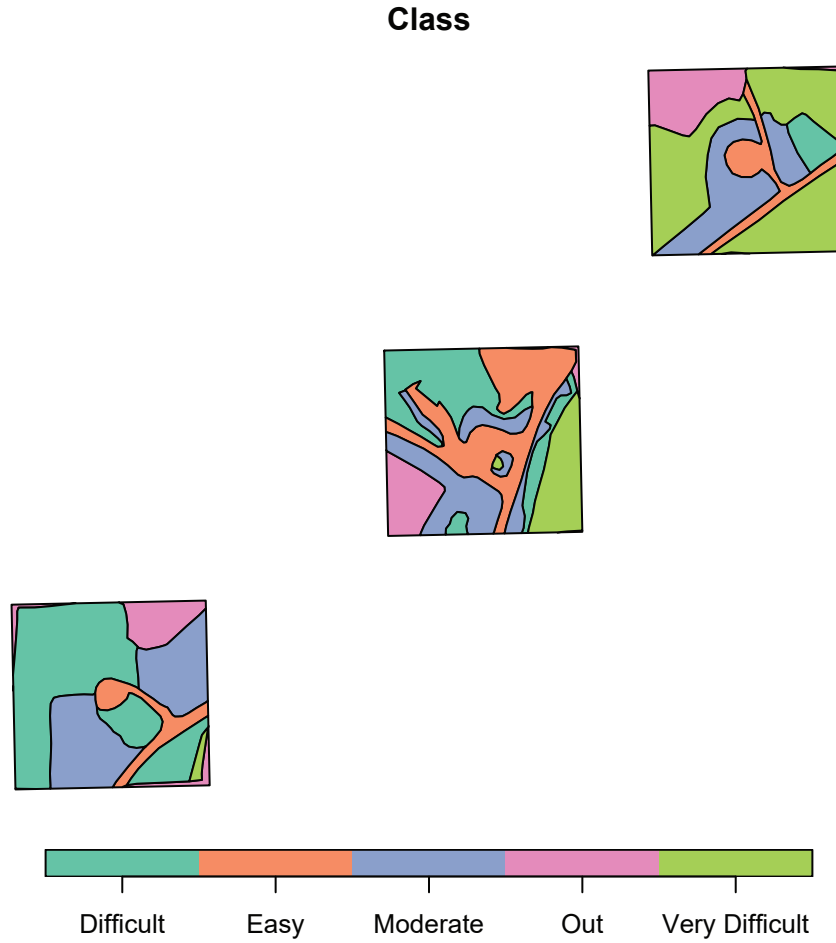
Class

Figure 15.   Software-generated map showing search areas for three turbines with five search classes—Easy, Moderate, Difficult, Very Difficult, and Out. "Easy" is represented primarily by roads and turbine pads, the other classes represent progressively more challenging search conditions, and "Out" areas were not searched due to rough terrain or thick vegetation. Figure drawn using the plot function from the dwp package with formatted site layout data from shapefiles.

```
# read carcass data
cod <- readCarcass(file_cod = "carcasses.shp", unitCol = "Turbine")

# add carcasses
rings_shape <- addCarcass(cod, data_ring = rings_shape, plotLayout = layout_shape)

head(cod$carcasses)
#> Simple feature collection with 6 features and 4 fields
#> Geometry type: POINT
#> Bounding box:  xmin: 658875.9 ymin: 4413105 xmax: 659242.4 ymax: 4413770
#> Projected CRS: NAD83 / UTM zone 17N
#>    Date_       Turbine  Species   Visibility   geometry
#> 1  2009-08-31  t7       PESU      Easy         POINT (658875.9 4413770)
#> 2  2009-08-03  t9       LACI      Mod          POINT (659242.4 4413290)
#> 3  2009-10-07  t9       LANO      Diff         POINT (659228.5 4413308)
#> 4  2009-08-16  t9       PESU      Mod          POINT (659195.4 4413340)
#> 5  2009-08-30  t10      LABO      Easy         POINT (659221.9 4413108)
#> 6  2009-08-24  t10      LACI      Mod          POINT (659223.4 4413105)
```

When reading carcass observation data from a shapefile into a `rings` structure with search class distinctions, dwp assigns each carcass to the proper ring and search class. The `rings` data are required, but there are two ways to insert the carcass data into the `rings`. The recommended approach is to provide the `plotLayout` data. Then, `addCarcass` determines the proper ring and search class to assign each carcass to by calculating the distances from the carcass (*x*, *y*) coordinates in the carcass observation data to the turbine locations stored in the `plotLayout` data, which has been previously read and formatted via `initLayout` and `prepRing`. Search class assignment is determined by the search class in `plotLayout` that is associated with the carcass (*x*, *y*) coordinates. If a carcass location falls in an unsearched area, the `addCarcass` function aborts with an error. The calculations are all done safely "under the hood" and directly from the imported shapefile data without needing further user input.

An alternative approach would be leave out the `plotLayout` data (`plotLayout = NULL`) to add carcasses directly from the carcass observation data into the `rings` data. Carcasses are assigned to rings based on distances calculated from the carcass locations (*x*, *y*) coordinates in the carcass observation data to the turbines (*x*, *y*) coordinates in the `$tcenter` array in the `data_ring`. Carcasses are assigned to search classes as indicated by the `scVar` in the carcass observation file. This can introduce errors in interpretation, because the model assumes that the "search classes" apply to the characteristics of the general area where a carcass is found (fig. 15), but often the search classes that are tagged to individual carcasses in carcass observation data apply to the search conditions in the immediate vicinity of carcasses. For example, if a carcass is found in a small `Difficult` patch in a large `Moderate` area, and the site map (`plotLayout`) includes only that `Moderate` area but not the `Difficult` patch, the carcass should be tallied as `Moderate`. This potential problem in data interpretation is not an issue if the `plotLayout` is included in the `addCarcass` function call.

If there are no class distinctions among the search areas, carcasses may be added to `rings` from a data frame with columns for turbine ID and carcass distance. For example, the `cod` argument in the `addCarcass` function call may be a data frame rather than an imported carcass location shapefile:

```
cod <- data.frame(r = c(31.2, 63.9, 18.4, 4.7), turbine = c("t1", "t2", "t2", "t1"))
```

Now that the site layout data have been formatted into rings and the carcass data have been inserted into the ring structure, the model fitting and analysis can proceed. For most of the input formats (excepting the grid data), the analyses follow the same path regardless of the initial data format. For the Casselman data, we do a basic analysis first but then indulge some more sophisticated analyses as an introduction to some of the more advanced questions that can be addressed using the software.

Model Fitting and Analysis

The basic modeling and analysis is similar, regardless of the initial data format. Several tools for fitting, interpreting, and evaluating models are available. However, the shapefiles allow covariates to be included in the models, enabling greater power to address more difficult and varied questions and to account for important factors that the other data formats cannot easily address.

The analysis begins with the fitting of distance models using `ddFit` with ring data or grid data. The basic function call requires only a formatted dataset as returned by the `prepRing` function (or `initLayout` for grid data). With the shapes data (`rings_shape`, as imported and formatted earlier) in the example discussed in this section, the search plots are divided into areas with different detection probabilities or search classes, which are stored in the `Class` column in the data and should be included in the model as `scVar = "Class"`.

Search class variables (like vegetative cover, for example) affect the detection probability of carcasses but are not expected to affect how far carcasses will lie from the turbines. There is no need to estimate the detection probabilities for the various search classes; the model automatically estimates relative detection probabilities among search classes and accounts for the differences. Although the relative detection probabilities are estimated (for example, the odds of detection in class A may be 2.52 times as great as in class B), the absolute detection probabilities (*g* in GenEst and eoa) are not. A simple example of fitting distance models using `ddFit` is given below. A number of options to `ddFit` are available; enter `?ddFit` in R for a more detailed information on arguments and return values.

```
dmod_shape <- ddFit(x = rings_shape, scVar = "Class")
#> Extensible models:
#>     xep1
#>     xep01
#>     xep2
#>     xep02
#>     xep12
#>     xep012
#>     tnormal
#>     MaxwellBoltzmann
#>     lognormal
#>
#> Non-extensible models:
#>     xep123
#>     xep0123
#>     constant
```

As the models are fit, their names are listed on the console as either "extensible" or "non-extensible," depending on whether or not the model can be extended outside the search radius and converted to a probability distribution. Non-extensible models generally are not appropriate for *dwp* analysis but may be interesting in their own right. The plot function can be used to graph the fitted models (fig. 16).

```
plot(dmod_shape)
```

The default for `plot` is to graph the cumulative distribution function (CDF) for each extensible, fitted model, although any subset of models (for example, `distr = c("xep01", "xep1")`) also may be plotted (fig. 17). By default, the best model according to a default model filter (`?modelFilter`) is highlighted, but any model can be highlighted at the user's discretion by setting the `mod_highlight` argument equal to the name of the model. Alternatively , the probability density functions (`type = "PDF"`) rather than the CDF can be plotted, but the probability density is limited strictly to inside the search radius (`extent = "win"`). For more details on plotting options, enter `?plot` in R.

```
plot(dmod_shape, type = "PDF")
```

To estimate ψ, the probability of a carcass lying inside the searched area and *dwp,* the fraction of carcasses lying inside the searched area, a model must be selected. If it is desired to account for all carcasses lying outside the areas searched—including areas outside the search radius—an extensible model is required.

The graph of the CDF shows the estimated probability that a carcass lies within a given distance from the turbine. In this example, the two heaviest-tailed models (`lognormal` and `xep1`) predict about 20 percent of the carcasses lie outside the search radius of around 88 m, whereas the other extensible models predict virtually zero carcasses outside the search radius. In some datasets, choice of model can have a profound impact on the estimated fraction of carcasses falling inside the search radius, $\widehat{dwp}$. Some of the important issues involved in model selection are discussed in the appendixes. The `modelFilter` function is designed to address many of the issues.

The probability that a carcass lies inside the search radius (`srad`)—or any other specified distance—can be estimated for any of the extensible, fitted distributions using the `pdd` function, which is an analog to `pnorm` for distance distributions.
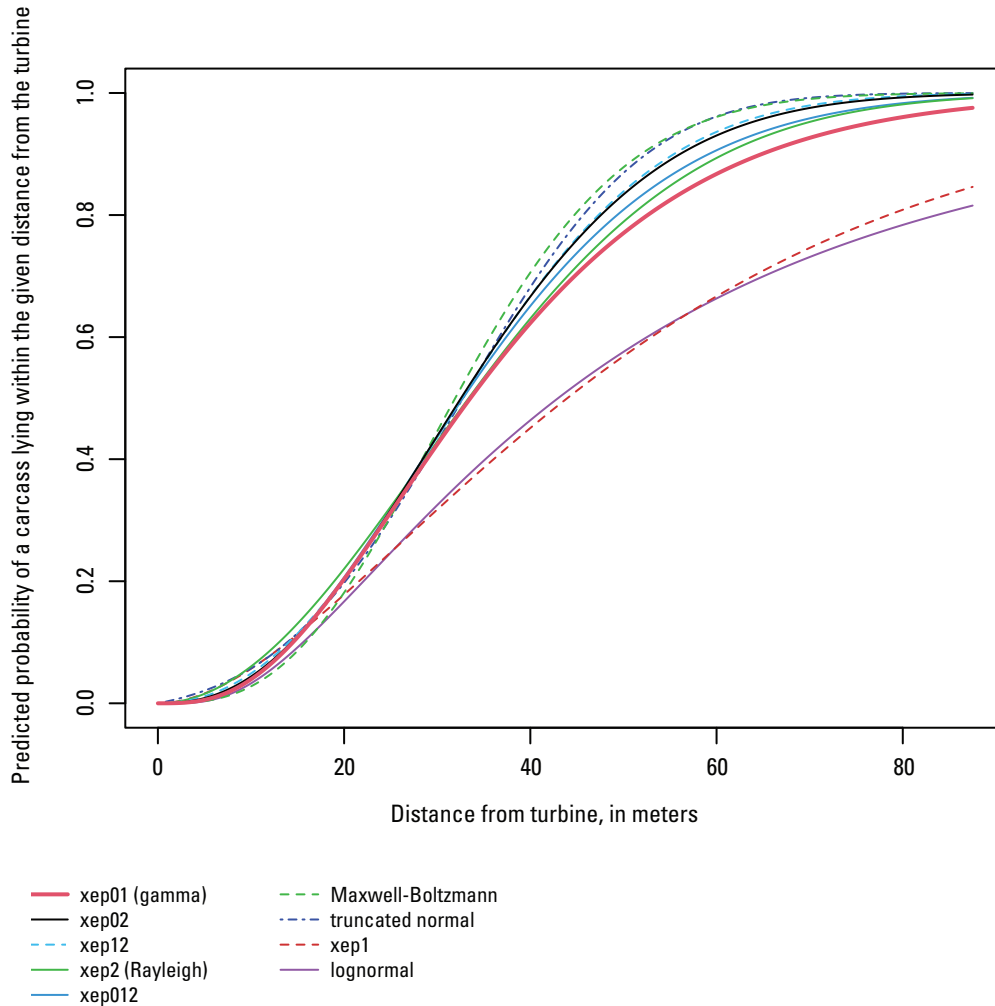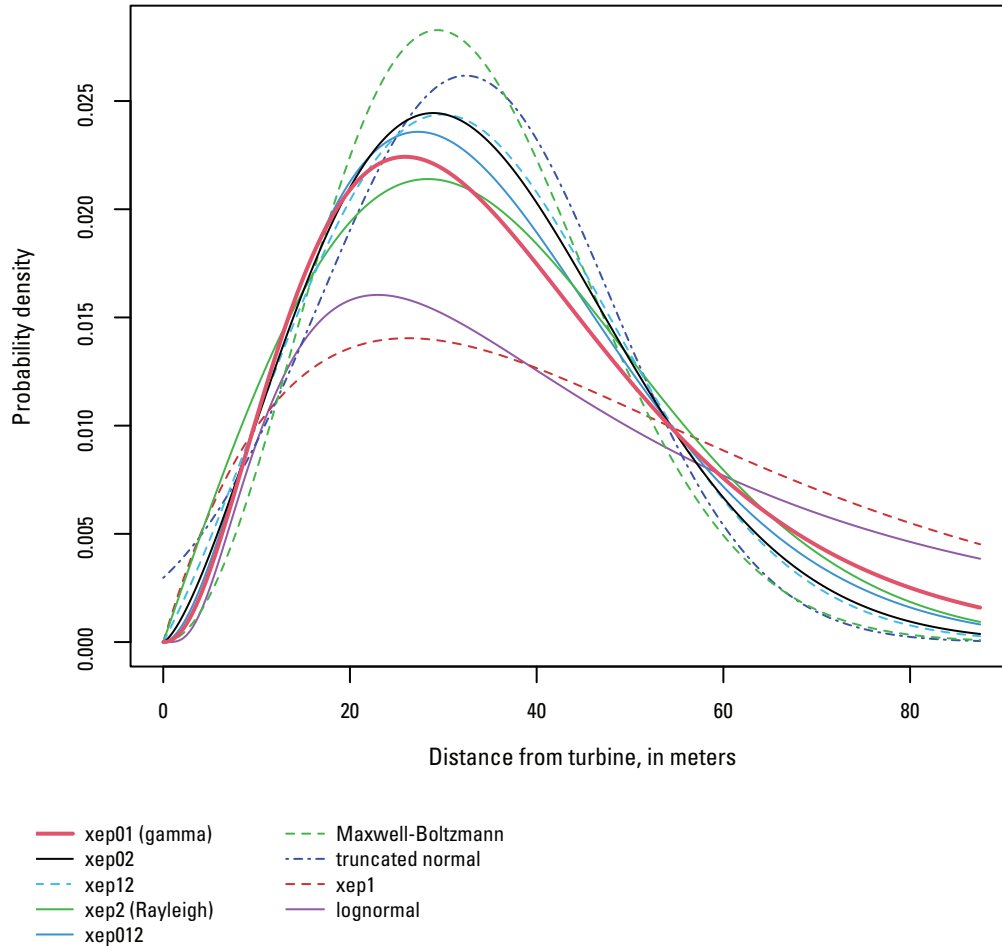
Figure 16. Cumulative distribution functions for extensible, standard models for the full "shape" dataset in section, "Model Fitting and Analysis." Figure drawn using the `plot` function from the `dwp` package with the fitted models.

```
pdd(rings_shape$srad, model = dmod_shape["xep01"])
#> [1] 0.9764346
pdd(rings_shape$srad, model = dmod_shape["xep02"])
#> [1] 0.997577
pdd(rings_shape$srad, model = dmod_shape["lognormal"])
#> [1] 0.8174281
```

The lines shown in figures 16 and 17 are for the maximum likelihood estimators of the models, which represent the most likely or "best" estimate for the given distribution. However, due to random variation, the distributions are estimated with uncertainty. That uncertainty can be accounted for by simulating from the asymptotic distribution of the parameter estimators using the `ddSim` (distance distribution simulation) function, and confidence intervals can be constructed (fig. 18).

Figure 17.   Probability density functions within the search radius for extensible, standard models for the full "shape" dataset in section, "Model Fitting and Analysis." Figure drawn using the `plot` function from the dwp package with the fitted models.

```
# simulate parameters to account for uncertainty:
simparm01 <- ddSim(dmod_shape["xep01"])
simparm1 <- ddSim(dmod_shape["xep1"])
simparm02 <- ddSim(dmod_shape["xep02"])
simparmln <- ddSim(dmod_shape["lognormal"])

# calculate confidence intervals:
srad <- rings_shape$srad
(CI01 <- round(quantile(pdd(srad, simparm01), prob = c(0.05, 0.95)), 3))
#>    5%      95%
#>    0.938   0.991
(CI02 <- round(quantile(pdd(srad, simparm02), prob = c(0.05, 0.95)), 3))
#>    5%      95%
#>    0.989   0.999
(CI1 <- round(quantile(pdd(srad, simparm1), prob = c(0.05, 0.95)), 3))
#>    5%      95%
#>    0.741   0.912
(CIln <- round(quantile(pdd(srad, simparmln), prob = c(0.05, 0.95)), 3))
#>    5%      95%
#>    0.629   0.917
```
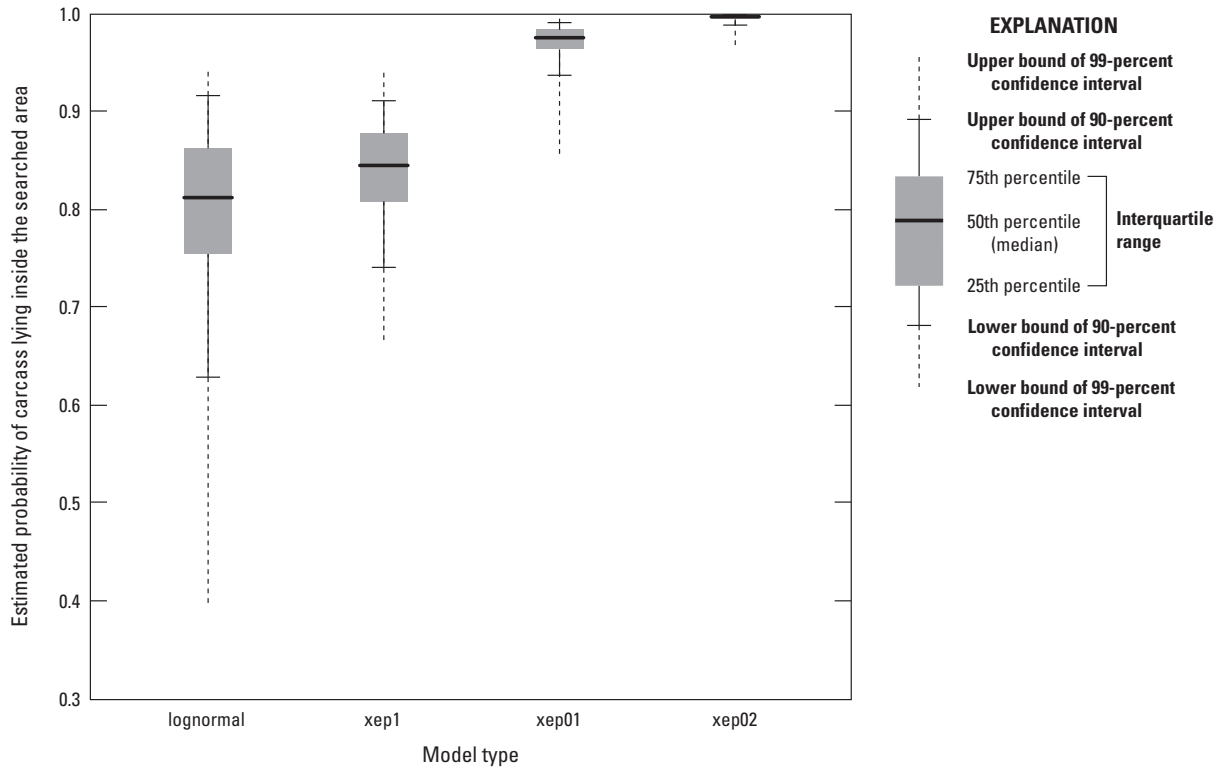
Figure 18.   Estimated probability of carcass lying inside the searched area according to four distributions fit to the full "shape" dataset.

It is also straightforward to calculate confidence intervals for quantiles of distance distributions using the qdd function, which is the analog of qnorm but for distance distributions. For example, to calculate 90 percent confidence intervals for the 90th percentile of distances at which carcasses lie, we can use the same sets of simulated parameters used in estimating the probabilities that carcasses lie inside the search radius, and calculate the 90th percentiles as qdd(0.9, ...):

```
# simulate 90th percentiles:
qxep01 <- qdd(0.9, model = simparm01)
qxep1 <- qdd(0.9, model = simparm1)
qxep02 <- qdd(0.9, model = simparm02)
qxepln <- qdd(0.9, model = simparmln)
# calculate confidence intervals:
(CI01 <- round(quantile(qxep01, prob = c(0.05, 0.95)), 1))
#>    5%       95%
#>    56.8     78.0
(CI02 <- round(quantile(qxep02, prob = c(0.05, 0.95)), 1))
#>    5%       95%
#>    51.4     63.5
(CI1 <- round(quantile(qxep1, prob = c(0.05, 0.95)), 1))
#>    5%       95%
#>    84.7     129.4
(CIln <- round(quantile(qxepln, prob = c(0.05, 0.95), na.rm = T), 1))
#>    5%       95%
#>    82.1     221.1
```

Thus, according to the `xep02` model, carcasses had an estimated 90-percent probability of falling within 56.1 m of the turbine with a 90-percent confidence interval of [51.4, 63.5] m (`qdd(0.9, dmod_shape["xep02"])`), whereas the lognormal model estimates a substantially greater distance of 118.5 m with a confidence interval of [82.1, 221.1] m (`qdd(0.9, dmod_shape["lognormal"])`). The `xep01` model was similar to the `xep02`, and the `xep1` more like the lognormal (fig. 19).
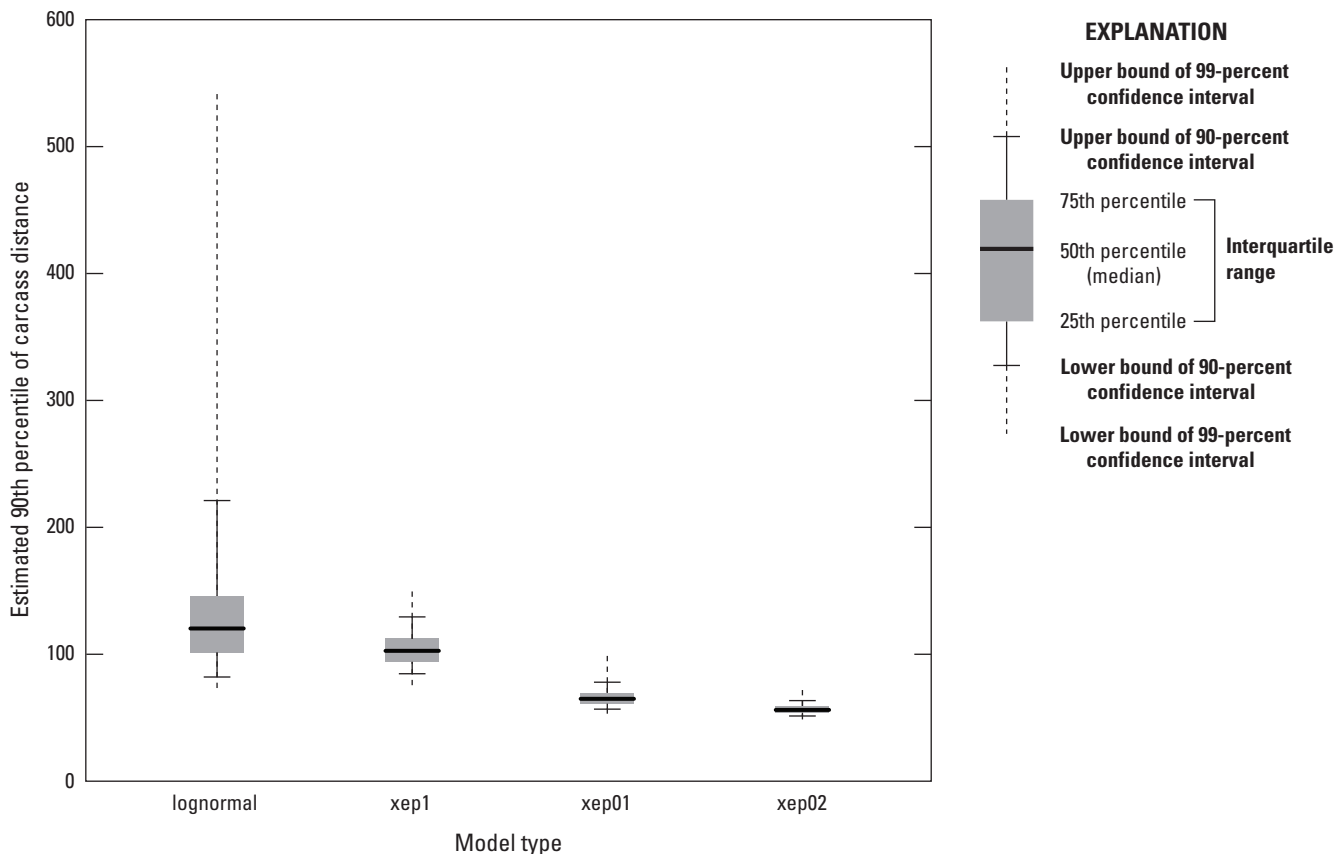
Probability of a Carcass Lying Inside the Searched Area

The `pdd` function can be used to estimate the CDF of carcass distribution as a function of distance from a turbine, which is related to but distinct from the probability, ψ, that a carcass lies inside the searched area. ψ is estimated by integrating the PDF of the fitted carcass distribution over the areas searched at each turbine. This can be accomplished for any fitted model and site layout using the `estpsi` function, and a summary figure can be viewed using `plot` (fig. 20).

```
psi01 <- estpsi(rings_shape, model = dmod_shape["xep01"])
plot(estpsi(psi01), main = "xep01")
```

The probabilities of lying in the searched area ranged from around 55 percent for turbines t11 and t15 to 85–90 percent for turbines t17–t23. The combined total was around 75 percent. More precise statistics can be extracted from the `psiHat` object returned from the `estpsi` function. In this case, that is `psi01`, which is a matrix with a column of `nsim` simulated values, $\hat{\psi}$, for each turbine, with the uncertainty in the estimation of ψ at each turbine captured by the variation in values in the turbine's column.

The $\hat{\psi}$ values for `xep01` and `xep02` varied by a few percent (fig. 21), although the 0.005 quantiles sometimes differed from one another by more than 10 percent.

Figure 19.   Estimated 90th percentile of carcass distances for four distributions fit to the full "shape" dataset.
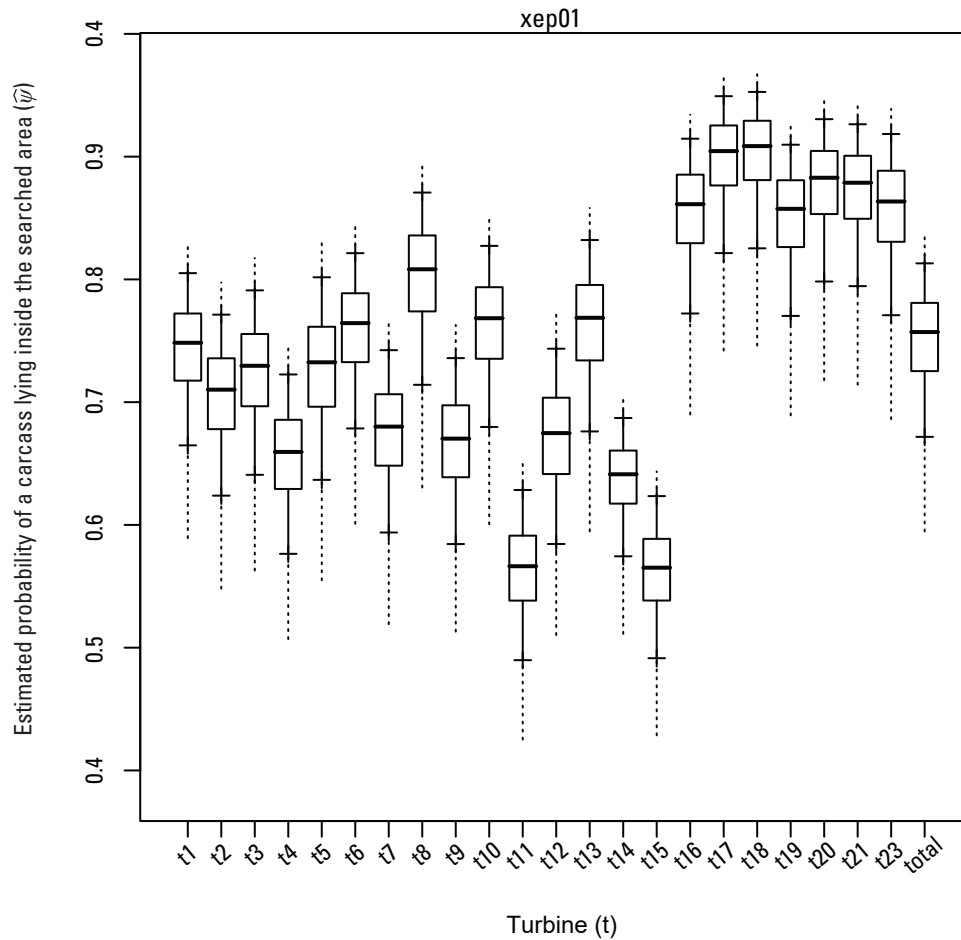
Figure 20. Estimated probability of a carcass lying inside the searched area. Figure drawn using the `plot` function from the dwp package with the estimated probability of a carcass lying inside the searched area.

## Estimated Fraction of Carcasses Lying Inside the Searched Area

The *expected* fraction of carcasses lying inside the searched area (that is, the mean *dwp*) is equal to the probability of a carcass lying inside the searched area (ψ), but the uncertainty in the estimate of the actual fraction of carcasses lying inside the searched area ($\widehat{dwp}$) is greater than the uncertainty in the estimated underlying probability ($\hat{\psi}$). Recall, the situation is analogous to a coin toss. We know the probability of a head (equivalent to ψ) is 50 percent, and the expected fraction of times a coin comes up heads (equivalent to *dwp*) in *n* coin tosses is *n*/2. However, the actual fraction of heads in a sequence of *n* flips will usually not be exactly *n*/2 but a little more or a little less. That additional, binomial uncertainty, in addition to the uncertainty in estimating ψ, needs to be captured in our estimate of *dwp* (Maurer and others, 2020). The distinction between ψ and *dwp* is discussed further in appendix 4.

The uncertainties in estimating ψ are captured in `estpsi`, and the uncertainties in estimating *dwp* given ψ are captured using the `estdwp` function. The function requires the estimated ψ and the number of carcasses observed at each turbine. The number of carcasses is required because the fewer the carcasses, the greater the relative uncertainty. Again, the coin toss analogy is illustrative. It would not be surprising to get anything from 0 to 100 percent heads in 4 coin tosses, but it would be highly unusual to get less than 35 percent or more than 65 percent heads in 40 coin tosses.

The fraction of carcasses lying inside the searched area at each turbine is calculated using `estdwp`:

```
dwp01 <- estdwp(psi01, ncarc = getncarc(rings_shape))
```

The structure of the *dwp* estimate is the same as that of $\hat{\psi}$, namely, a matrix with a column for each turbine, incorporating the uncertainties in the estimated *dwp*'s for each turbine and for the total at the site.
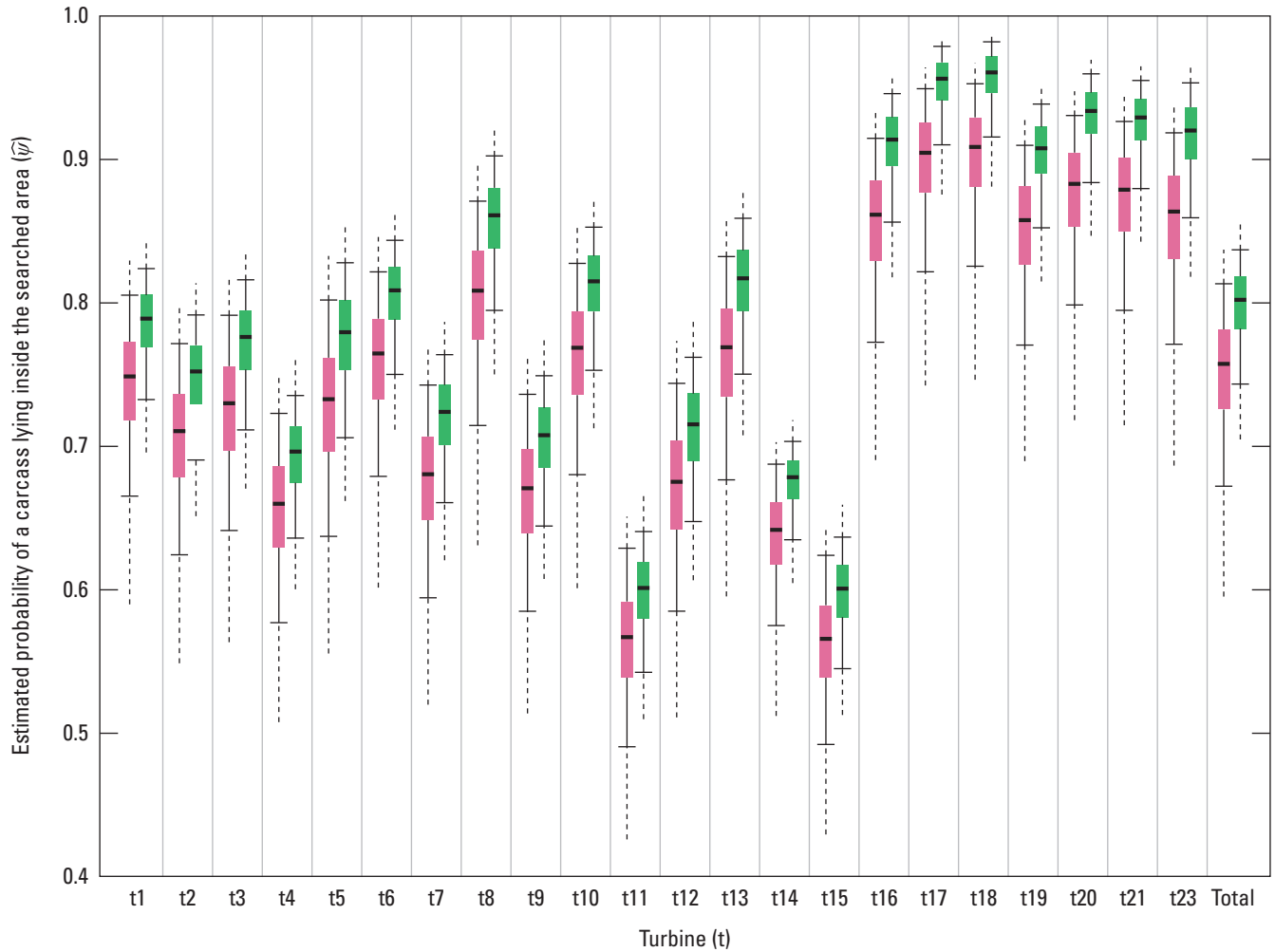
Figure 21.   Comparison of estimated probabilities of a carcass lying inside the searched area according to the `xep01` and `xep02` models.

Paired boxplots of $\hat{\psi}$ and estimated *dwp* clearly reveal the greater uncertainty in estimating *dwp* compared to ψ (fig. 22).

The estimated *dwp* can be formatted for use in GenEst either by setting `forGenEst = TRUE` in the argument list for `estdwp`, or by using the `formatGenEst` function on the estimated *dwp*.

```
# Option 1: format estimated dwp for GenEst in two
steps set.seed(1942) # for repeatability
dwp01 <- estdwp(psi01, ncarc = getncarc(rings_shape))
dwp01 <- formatGenEst(dwp01)
head(dwp01)
#>      Turbine   dwp
#> 1    t1        0.750
#> 2    t2        0.600
#> 3    t3        0.500
#> 4    t4        0.692
#> 5    t5        1.000
#> 6    t6        1.000


# Option 2: format estimated dwp for GenEst in one step
set.seed(1942)
dwp01 <- estdwp(psi01, ncarc = getncarc(rings_shape), forGenEst = TRUE)
head(dwp01)
#>      turbine   dwp
#> 1    t1        0.750
#> 2    t2        0.600
#> 3    t3        0.500
#> 4    t4        0.692
#> 5    t5        1.000
#> 6    t6        1.000
```

The formatted `dwp01` can now be used as `data_dwp = dwp01` in the argument list in a command line call to the `estM` function in GenEst (assuming that the other required arguments can be supplied as well) or can be exported to a .csv file for use in the GenEst GUI using the `exportGenEst` function.

```
exportGenEst(dwp01, file = "my_dwp.csv")
```

Additional Analysis for an Interacting Covariate

In the previous example, we included a covariate to account for variability in carcass detection probabilities on the ground. Interacting covariates—that is, variables that affect the actual distributions of carcasses rather than simply affecting detection probabilities on the ground—present challenges that are difficult to overcome in a generic way that would be appropriate for automation in dwp.

The Casselman data provide an example of an interacting covariate, which can be analyzed by fitting separate sets of models for different covariate levels. In particular, the site represented in the "shapes" files has turbines that were under two different operating protocols, freely operating or curtailed at low windspeeds. We would expect that the carcasses at the curtailed turbines would be distributed farther from the turbines than carcasses at the non-curtailed turbines because the curtailed turbines were operating only under relatively high wind speeds. We can analyze the data from the two turbine sets separately to see if there are differences in the distance distributions and consequent $\hat{\psi}$.

The `subset` function from the dwp package can be used to split the `layout_shape` dataset into two parts—freely operating turbines (`layout_free`) and turbines curtailed at low wind speeds (`layout_curtailed`).
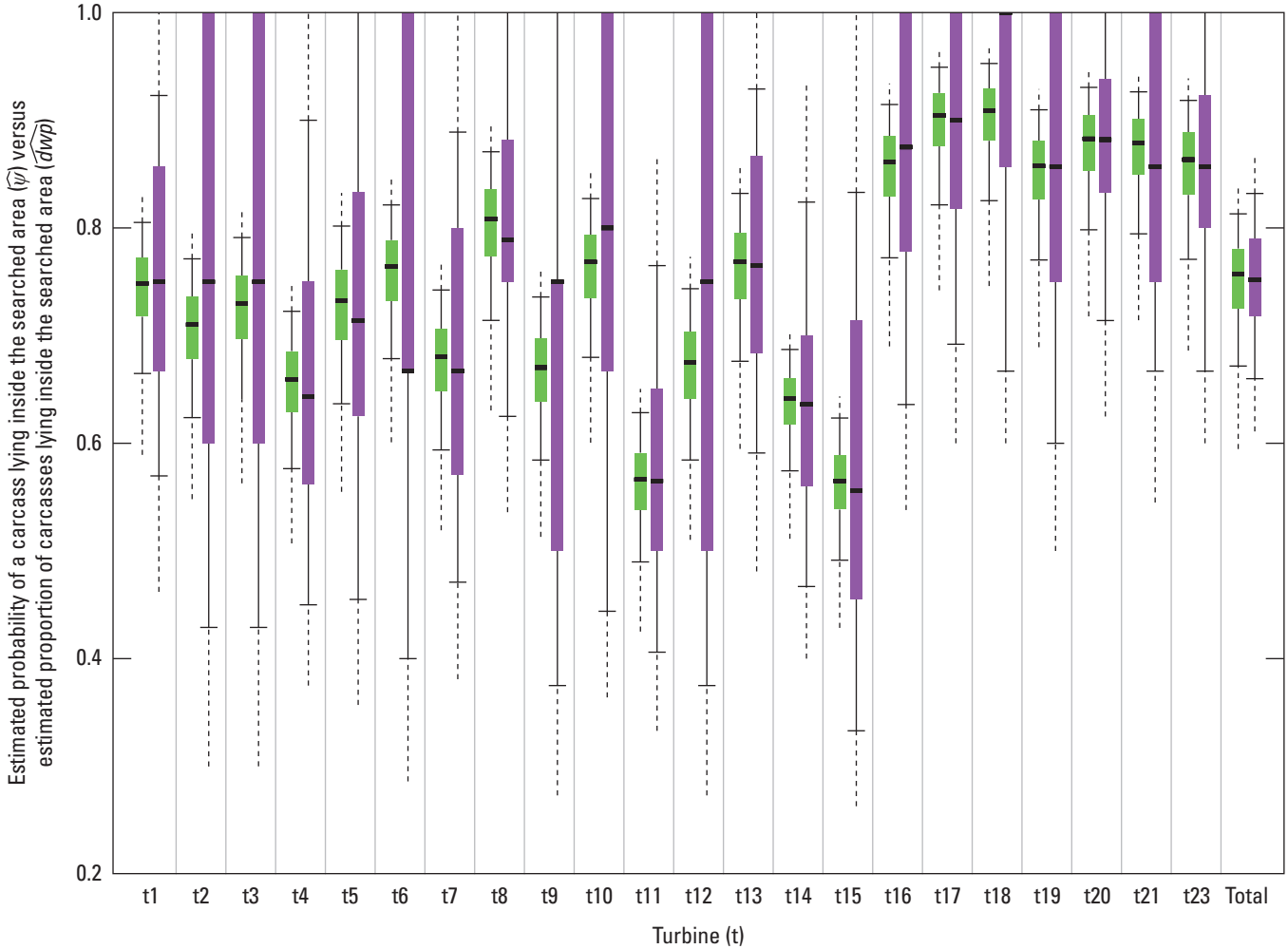
Figure 22. Comparison of estimated probability of a carcass lying inside the searched area versus estimated proportion of carcasses lying inside the searched area for each turbine according to the `xep01` model.

The two datasets can then be analyzed separately and the results compared:

```
######## basic setup for freely operating turbines:
# data formatting
layout_free <- subset(layout_shape, select = "Turbine",
  subset = paste0("t", c(1, 3, 4, 8, 11, 13, 14, 16, 20, 23)))
cod_free <- subset(cod, select = "Turbine",
  subset = paste0("t", c(1, 3, 4, 8, 11, 13, 14, 16, 20, 23)))
rings_free <- prepRing(layout_free, scVar = "Class", notSearched = "Out", silent = T)
rings_free <- addCarcass(cod_free, data_ring = rings_free, plotLayout = layout_free)

# fitting the models
mod_free <- ddFit(rings_free, scVar = "Class")
#> Extensible models:
#>    xep1
#>    xep01
#>    xep2
#>    xep02
#>    xep12
#>    tnormal
#>    MaxwellBoltzmann
#>    lognormal
#>
#> Non-extensible models:
#>    xep012
#>    xep123
#>    xep0123
#>    constant

######## basic setup for curtailed turbines:
layout_curtailed <- subset(layout_shape, select = "Turbine",
  subset = paste0("t", c(2, 5, 6, 7, 9, 10, 12, 15, 17, 18, 19, 21)))
cod_curtailed <- subset(cod, select = "Turbine",
  subset = paste0("t", c(2, 5, 6, 7, 9, 10, 12, 15, 17, 18, 19, 21)))
rings_curtailed <- prepRing(layout_curtailed, scVar = "Class", notSearched = "Out",
  silent = TRUE) # "silent" cancels updates on the status of the calculations
rings_curtailed <- addCarcass(cod_curtailed, data_ring = rings_curtailed,
  plotLayout = layout_curtailed)
mod_curtailed <- ddFit(rings_curtailed, scVar = "Class")
#> Extensible models:

#>    xep1
#>    xep01
#>    xep2
#>    xep02
#>    xep12
#>    xep012 tnormal
#>    MaxwellBoltzmann
#>    lognormal
#>
#> Non-extensible models:
#>    xep123
#>    xep0123
#>     constant
```

The descriptive statistics for the fitted models can be summarized using the function stats, which calculates estimated medians, modes, and the 75th, 90th, and 95th percentiles according to models, and sorts the results by ΔAICc.

```
stats(mod_free)
#>                     median    75%    90%    95%     mode   P_win  deltaAICc
#>      xep01          30.6      44.3   59.5   70.1    23.5   0.985  0.00
#>      xep2           30.6      43.3   55.9   63.7    26.0   0.996  0.64
#>      xep02          30.4      41.9   53.2   60.2    26.89  0.999  1.51
#>      xep12          30.5      42.3   53.6   60.7    27.0   0.998  2.35
#>      lognormal      35.6      59.1   93.2   122.5   20.3   0.884  2.51
#>      MaxwellBoltzmann  30.1   39.7   49.0   54.8    27.7   1.000  5.10
#>      xep1           36.8      59.1   85.4   104.1   21.9   0.907  6.41
#>      tnormal        30.8      50.8   50.8   56.5    30.3   1.000  9.61
stats(mod_curtailed)
#>                     median    75%    90%    95%     mode   p_win  deltaAICc
#>      MaxwellBoltzmann  35.0   46.1   56.9   63.6    32.2   0.998  0.00
#>      xep12          35.7      47.2   58.1   64.7    33.6   0.997  1.02
#>      xep02          36.0      48.6   60.9   68.6    32.3   0.993  1.06
#>      xep2           38.6      54.5   70.3   80.2    32.8   0.969  1.09
#>      xep01          38.8      55.9   74.9   87.9    30.1   0.946  1.95
#>      tnormal        35.5      45.7   55.0   60.5    35.3   1.000  2.66
#>      xep012         35.7      47.6   59.0   66.0    33.1   0.996  3.06
#>      xep1           60.1      96.5   139.4  170.0   35.8   0.695  4.47
#>      lognormal      64.2      116.5  199.2  274.6   29.4   0.632  4.93
```

In general, the carcasses at the curtailed turbines did tend to be somewhat more distant from turbines than the carcasses at the freely operating turbines as expected (fig. 23). The magnitude of the estimated differences strongly depended on the selected model (fig. 24A). The differences among the models largely vanish when the models are restricted solely to the range of the data and not extrapolated outside the search radius (fig. 24B). This indicates that the discrepancies among models primarily are due to the extrapolation outside the search radius and the models' respective assumptions about the distribution outside the search radius, that is, in the tails of the distributions. The heavy-tailed distributions (lognormal and xep1), predict a relatively large fraction of carcasses outside the search radius, whereas the light-tailed distributions (truncated normal and Maxwell-Boltzmann) predict relatively few carcasses outside the search radius.

Because the search radius is not long (square with half-width of 60 m and diagonal radius of 82 m), one of the heavier-tailed distributions with degree less than 2 is preferred (xep1, xep01, or lognormal). The light-tailed distributions (Maxwell-Boltzmann and the xep models with degree 2 or 3) tend to converge rapidly to zero shortly beyond the most distant carcass and underestimate the total outside the searched area. In both the curtailed and free datasets, the xep01 (gamma distribution) was the highest-ranking, relatively heavy-tailed distribution and had ΔAICc < 2. Thus, the xep01 distribution is an appropriate choice for both types of turbine.

The estimated probabilities of carcasses lying inside the searched areas can be calculated separately for the two sets of turbines (freely operating or curtailed):

```
# estimated spatial coverage for curtailed turbines...
# ...using distance model fit to data from free turbines
psi_free <- estpsi(rings_free, model = mod_free["xep01"])

# ...using distance models fit to data from curtailed turbines
psi_curtailed <- estpsi(rings_curtailed, model = mod_curtailed["xep01"])
```

The estimated ψ and dwp for the free and curtailed turbines depend on the configuration of the search plots, which vary by turbine, so the values are not strictly comparable. However, a few meaningful patterns can be discerned (fig. 25). First, the lower tails of the distributions for $\hat{\psi}$ extend much lower for the curtailed turbines than for the free turbines. This is due to the presence of more carcasses at greater distances at the curtailed turbines and a smaller probability that the tail of the distribution is resolved enough to avoid the possibility that many carcasses lie outside the search radius. Next, the uncertainty introduced in estimating dwp erases the clarity of the distinction between curtailed and free turbines.

After calculating the $\widehat{dwp}$'s for curtailed and free turbines separately, the estimates combined into one data frame for export for GenEst.

```
ge_combined <- rbind(formatGenEst(dwp_curtailed), formatGenEst(dwp_free))
exportGenEst(ge_combined, file = "dwp_all.csv")
```
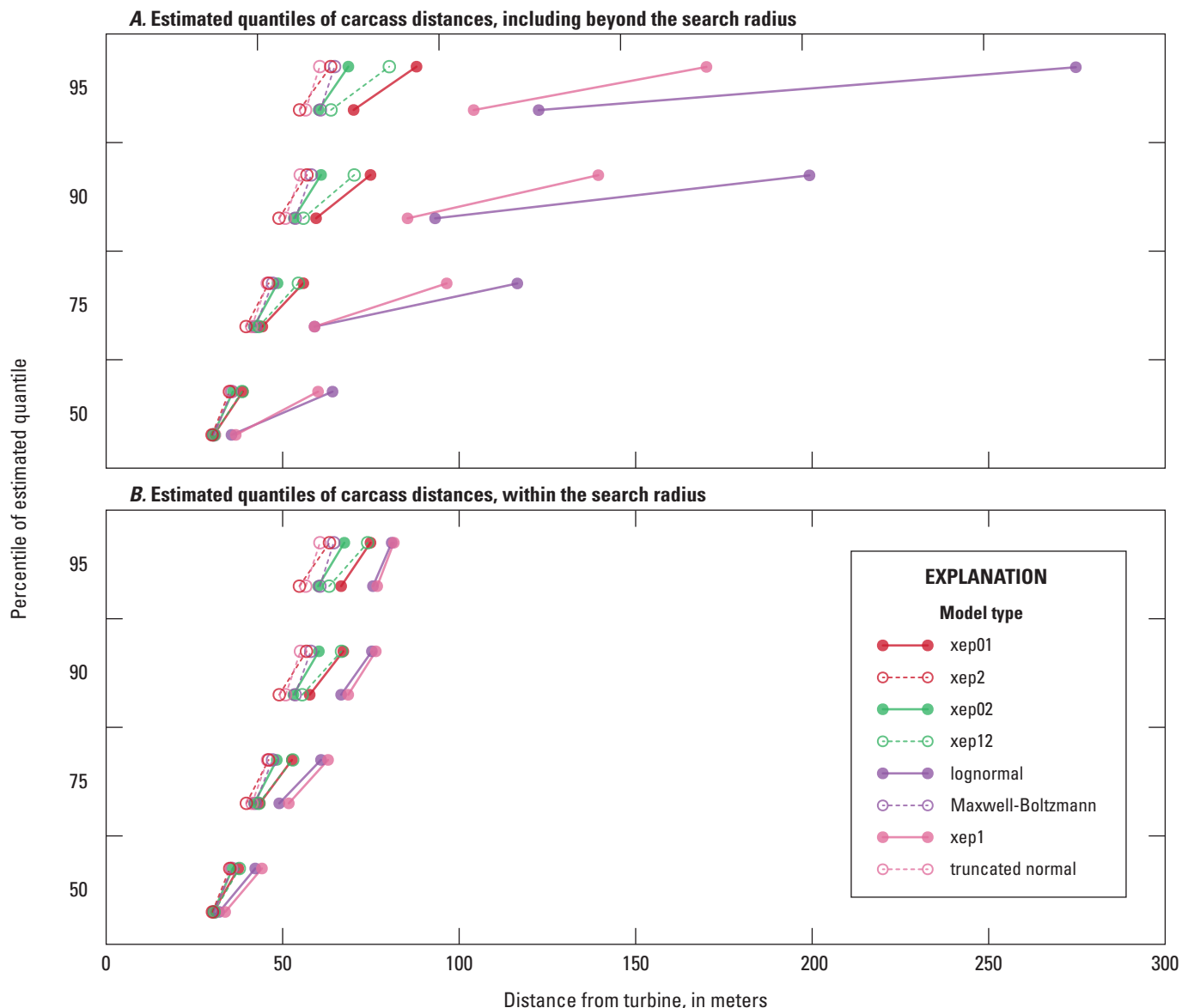
Figure 23. Comparison of descriptive statistics for the distance distributions for the curtailed turbines (top row of dots in each quantile) and the freely operating turbines (bottom row of dots in each quantile). The lines show the difference between curtailed and free turbines. *A.* Estimated quantiles of carcass distances (including extrapolations to account for carcasses outside the search radius). *B.* estimated quantiles of carcass distances solely inside the search radius.

Although we choose the gamma distribution (xep01) for estimating *dwp* for export to GenEst, the Rayleigh distribution (xep2) provides an equally high-quality fit and has the interesting and convenient property that the ratio of the quantiles of two Rayleighs is equal to the ratio of their parameters ($\sigma 1/\sigma 2$), regardless of the quantile. Thus, measuring by the fitted Rayleigh distribution, carcasses at the curtailed turbines were $\sigma_{\text{curtailed}}/\sigma_{\text{free}} = 1.258$ times more distant from the turbines than were the carcasses at the free turbines. That is, the median, 90th percentile, 99th percentile, and so on were all 25.8 percent greater at the curtailed turbines than at the free turbines.

```
# estimated spatial coverage for curtailed turbines...
# ...using distance model fit to data from free turbines
psi_free <- estpsi(rings_curtailed, model = mod_free["xep01"])

# ...using distance models fit to data from curtailed turbines
psi curtailed <- estpsi(rings curtailed, model = mod curtailed["xep01"])
```
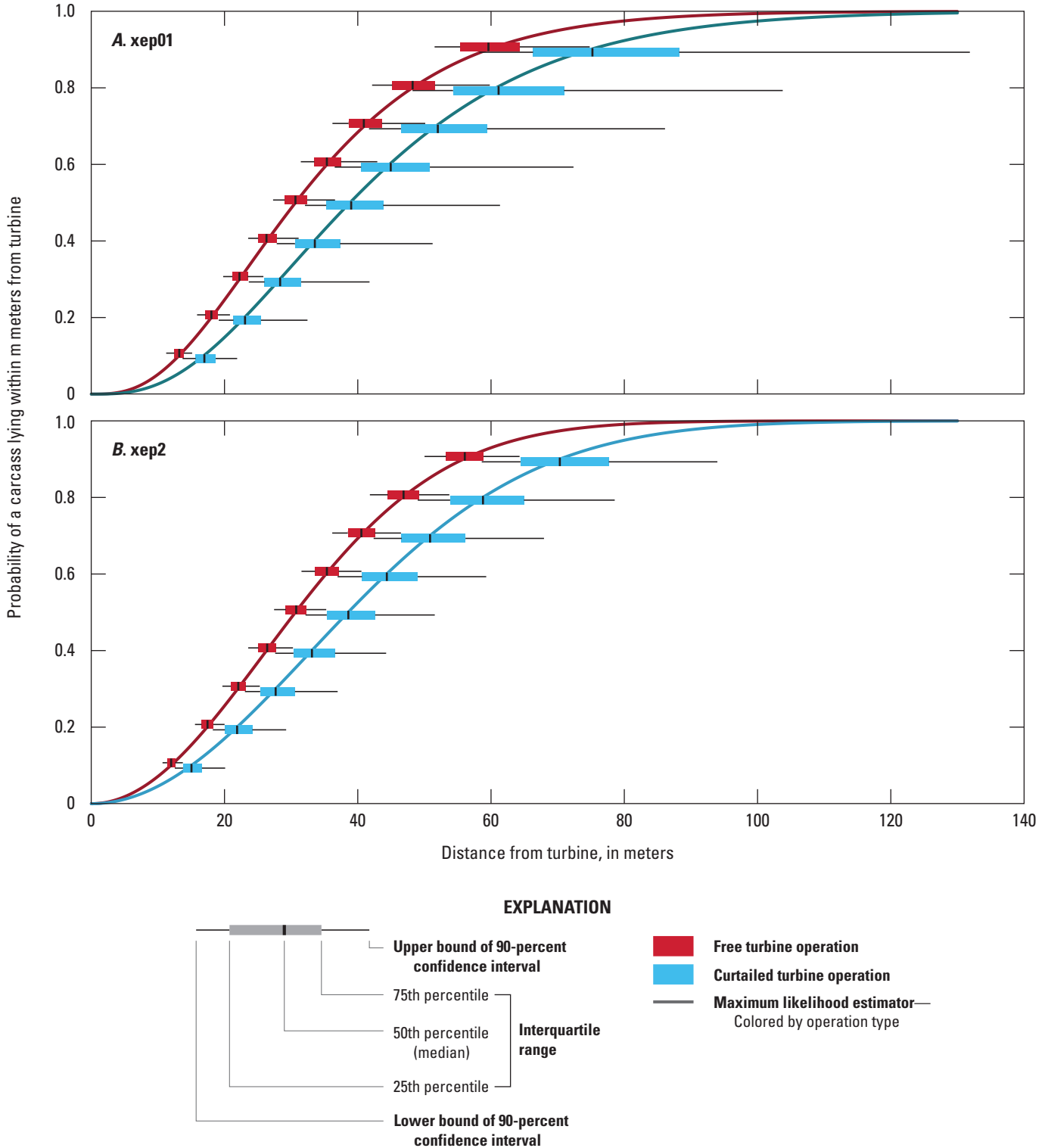
Figure 24. Cumulative (*A*) `xep01` and (*B*) `xep2` distribution functions of carcass distances at curtailed and free turbines at the Casselman site. Curves show the maximum likelihood estimator.
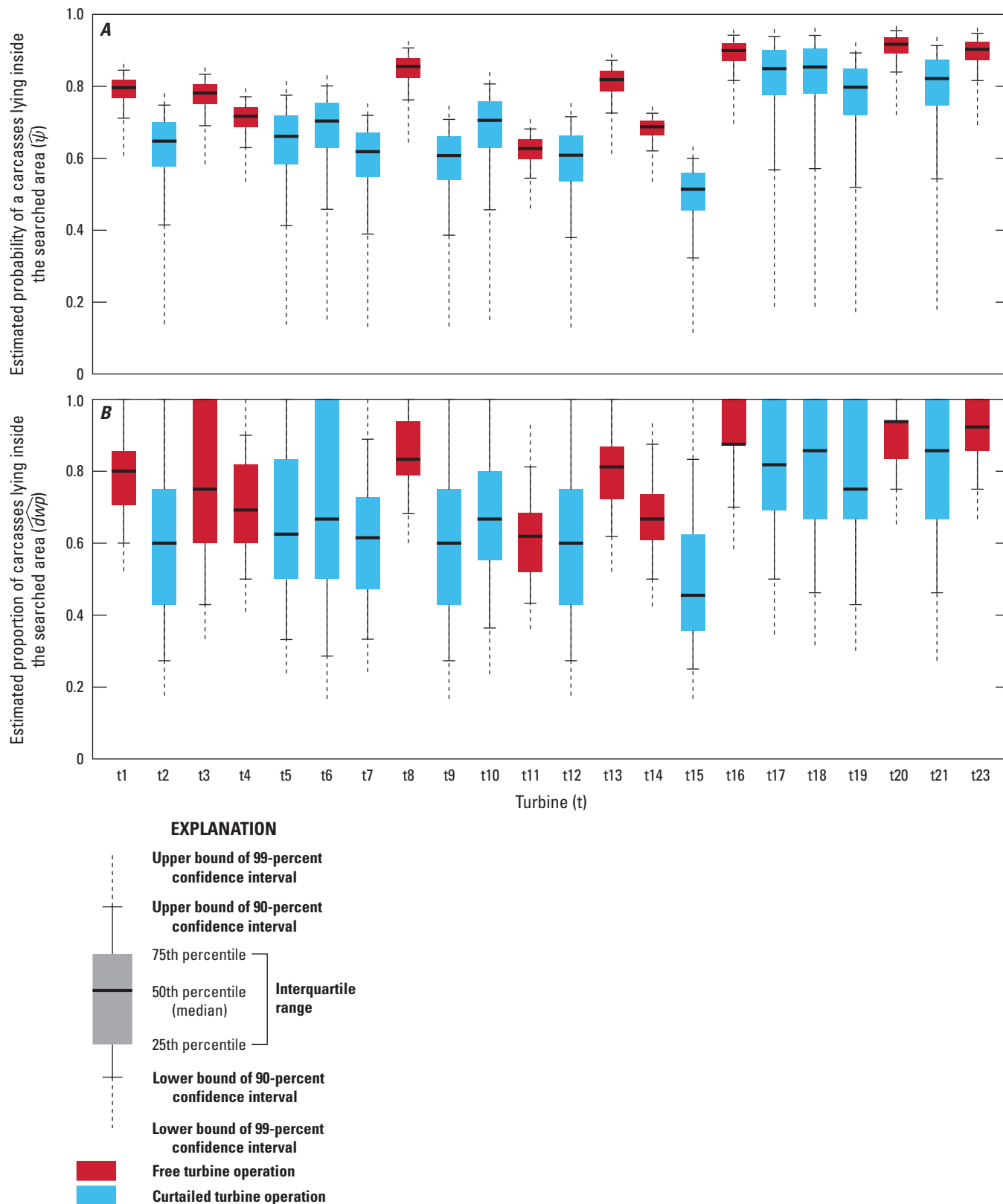
Figure 25. *A.* Estimated probability of a carcass lying inside the searched area, and *B.* estimated proportion of carcasses lying inside the searched area according to the `xep01` distribution for curtailed and free turbines at the Casselman site.

If not accounted for, the tendency for the carcasses to be more distant at the curtailed turbines can result in substantial bias when estimating total mortality. For example, using the distribution that was fit for the freely operating turbines to estimate mortality at the curtailed turbines, we would underestimate the mortality by some 55 percent on average at the curtailed turbines due to systematic bias in the estimates of ψ (fig. 26).
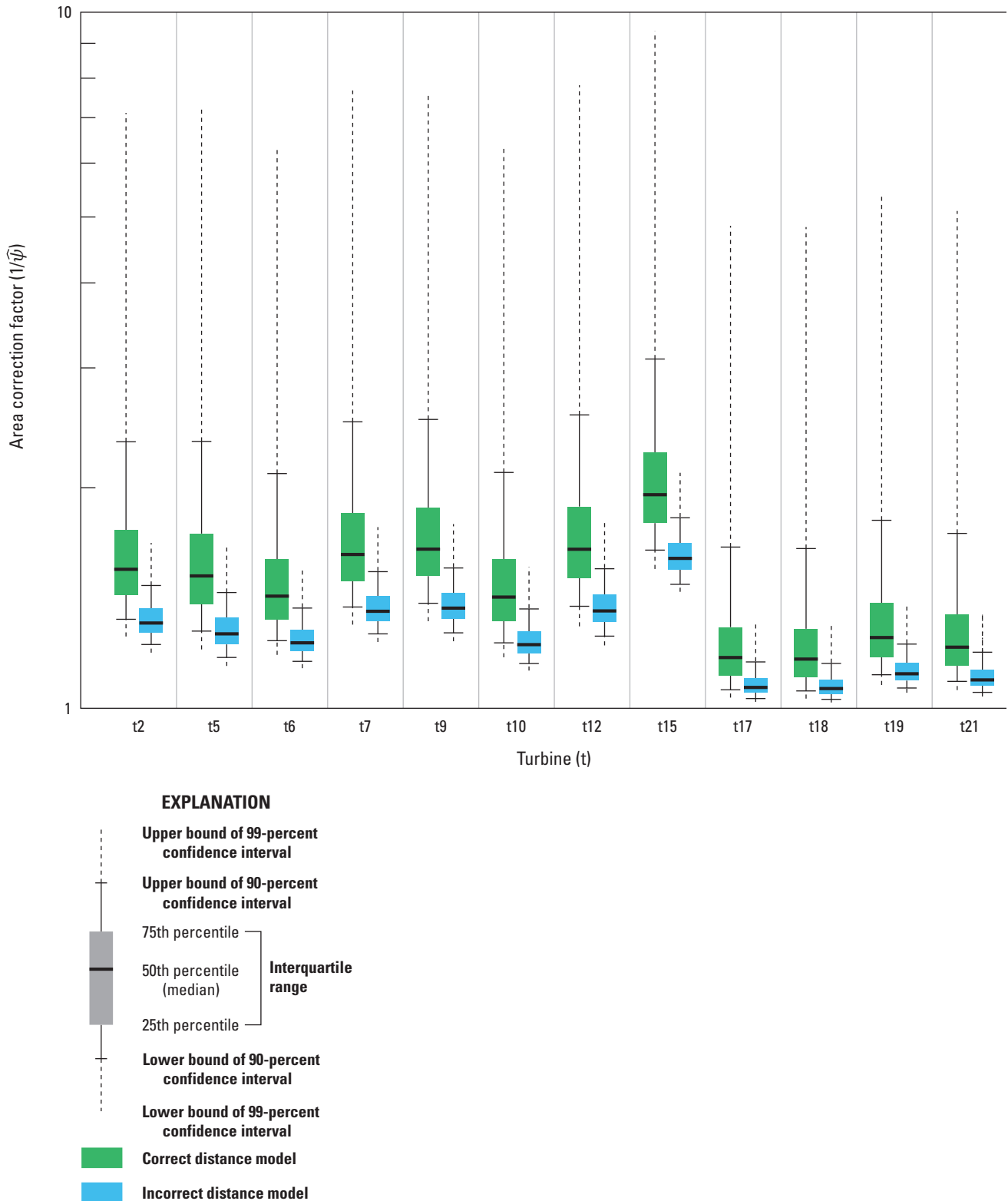
Figure 26.   Comparison of estimates of the area correction factor at curtailed turbines using the correct distance model based on data from curtailed turbines versus using an incorrect distance model derived from data from freely operating turbines.

Carcass Size

Carcass size also may influence the distances that carcasses may fly, perhaps even size differences among bats can influence the carcass dispersion patterns around turbines. The Casselman data include a range of bat species of different sizes, which we can class as either large (`L`) or small (`S`). The dwp package provides tools for processing datasets with size class distinctions and exporting the resulting *dwp* estimates to GenEst.

The following sections of code show how to split a dataset on a carcass class variable, fit separate distance models for each size class using a single command, and create GenEst-ready *dwp* with carcass class distinctions.

The first task with this dataset is to add a "Size" variable to the carcass data from the freely operating turbines, with LACI, LASE, and EPFU classed as "L" and the other species as small.

```
cod_free$carcasses$Size <- ifelse(
  cod_free$carcasses$Species %in% c("LACI", "LASE", "EPFU"), "L", "S")
```

It is always useful to check whether your command has the desired effect. Tabulate the carcasses by size and species to verify that the split was done correctly and that there are sufficient carcasses in each subset to do the estimation.

```
table(cod_free$carcasses[, c("Species", "Size"), drop = TRUE])
#>                 Size
#>     Species     L       S
#>     EPFU        3       0
#>     LABO        0       15
#>     LACI        35      0
#>     LANO        0       28
#>     LASE        2       0
#>     MYLU        0       18
#>     PESU        0       12
```

NOTE: The `drop = TRUE` is necessary here because `cod_free` is an sf object, which is a specially formatted data frame with GIS information attached. When that extra information is attached, R does not recognize the sf object as a data frame, so functions like `table` will not work. The GIS information is dropped by using `drop = TRUE` in the subsetting. The result is a standard R data frame that can be manipulated like any other data frame in R.

To estimate ψ and calculate $\widehat{dwp}$ for use in GenEst when there are distinctions among carcass classes, the *dwp*'s must be estimated for each size class separately, which the dwp functions, `estpsi` and `estdwp`, can do automatically if the data are properly formatted. Formatting involves adding carcasses (feeding a carcass dataset into `addCarcass`) to the description of the site by rings (`data_ring`) with a directive to split the datasets by carcass class (`ccCol`).

When a `ccCol` (carcass class column) is included in the argument list, `addCarcass` creates separate ring structures for each carcass class, which in turn are interpreted by dwp functions to estimate ψ and *dwp*, and format the results for exporting to GenEst.

```
# split the ring data into separate structures for each size class
cod_free_size <- addCarcass(cod_free, data_ring = rings_free,
  plotLayout = layout_free, ccCol = "Size")

# fit separate distance models to the different size classes:
mod_size <- ddFit(cod_free_size, scCol = "Class")
#> Extensible models:
#>    xep1
#>    xep01
#>    xep2
#>    xep02
#>    xep12
#>    xep012
#>    xep123
#>    xep0123
#>    tnormal
#>    MaxwellBoltzmann
#>    lognormal
#>
#> Non-extensible models:
#>    constant
#> Extensible models:
#>    xep1
#>    xep01
#>    xep2
#>    xep02
#>    xep12
#>    tnormal
#>    MaxwellBoltzmann
#>    lognormal
#>
#> Non-extensible models:
#>    xep012
#>    xep123
#>    xep0123
#>    constant

# view summary statistics for the resulting models:
stats(mod_size)
#> $L
#>                          median   75%    90%    95%    mode   p_win  deltaAICc
#>    xep2                  28.4     40.1   51.7   59.0   24.1   0.999  0.00
#>    xep1                  30.8     49.4   71.4   87.1   18.4   0.951  0.22
#>    xep12                 28.4     41.8   55.5   64.4   21.8   0.994  1.66
#>    xep01                 29.0     44.4   62.1   74.6   19.7   0.976  1.72
#>    xep02                 28.4     41.0   53.5   61.4   23.3   0.997  1.78
#>    lognormal             34.7     62.8   107.0  147.3  16.0   0.854  1.98
#>    tnormal               28.7     40.4   51.2   57.7   27.1   1.000  3.26
#>    xep123                28.3     40.7   51.6   57.9   23.9   1.000  3.44
#>    xep012                28.4     42.2   56.7   66.2   21.1   0.992  3.69
#>    xep0123               28.1     41.0   50.7   55.6   13.5   1.000  4.51
#>    MaxwellBoltzmann      28.7     37.8   46.7   52.2   26.4   1.000  6.82
#>
#> $S
#>                          median   75%    90%    95%    mode   p_win  deltaAICc
#>    xep2                  28.9     40.8   52.6   60.0   24.5   0.998  0.00
#>    xep12                 28.8     41.1   53.3   60.9   24.1   0.998  2.01
#>    lognormal             32.1     53.2   83.9   110.1  18.3   0.910  2.04
#>    xep1                  32.3     51.8   74.9   91.3   19.2   0.941  2.57
#>    MaxwellBoltzmann      29.1     38.3   47.3   52.9   26.8   1.000  6.15
#>    tnormal               29.2     40.2   50.3   56.3   28.2   1.000  6.63
```

Selection of models to use in the calculations of $\widehat{dwp}$ must be done for each size class separately. As with the earlier examples in the Casselman data (sections, "Modeling Fitting and Analyses" and "Additional Analyses for an Interacting Covariate"), the xep01 model scores well in the modelFilter for both size classes:

```
# Model filter scores for small carcasses
modelFilter(mod_size[["S"]])$scores
#>                  extensible  rtail  ltail  aicc  hin    deltaAICc
#> xep2             1           1      1      1     0      0.0000000
#> xep01            1           1      1      1     0      0.5211667
#> xep02            1           1      1      1     0      1.9695832
#> xep12            1           1      1      1     0      2.0102368
#> lognormal        1           1      1      1     0      2.0444269
#> xep1             1           1      1      1     0      2.5680237
#> tnormal          1           1      1      1     0      6.6354677
#> MaxwellBoltzmann 1           1      0      1     0      6.1499831
#> xep123           0           0      1      1     0      0.1523155
#> xep0123          0           0      1      1     0      2.0836871
#> xep012           0           0      1      1     0      2.4650216
#> constant         0           0      1      0     0      54.2732536


# Model filter scores for large carcasses
modelFilter(mod_size[["L"]])$scores
#>                  extensible  rtail  ltail  aicc  hin    deltaAICc
#> xep2             1           1      1      1     0      0.0000000
#> xep1             1           1      1      1     0      0.2207812
#> xep12            1           1      1      1     0      1.6626334
#> xep01            1           1      1      1     0      1.7142410
#> xep02            1           1      1      1     0      1.7826877
#> tnormal          1           1      1      1     0      3.2572103
#> xep123           1           1      1      1     0      3.4374762
#> xep012           1           1      1      1     0      3.6869378
#> xep123           1           1      1      1     0      4.5038122
#> MaxwellBoltzmann 1           1      0      1     0      6.8218704
#> lognormal        1           0      1      1     0      1.9813879
#> constant         0           0      1      0     0      30.2262282
```

The same caveat about the tendency of light-tailed models to overestimate the ψ—especially with bats—when the search radius is not large enough to encompass all the carcasses (and with some cushion) that is mentioned in sections, "Example 3— Simple Geometry," "Example 5— The (x, y) Grid Data," "Additional Analysis for an Interacting Covariate," and appendix 3 applies here, so we again select the xep01 model for both size classes.

There are two ways to call estpsi with data and models that include carcass class distinctions. They both accomplish the same thing, namely, to estimate ψ for each carcass class.

1. Pass the whole array of fitted models (mod_size, in this example) to the estpsi function along with a vector of names of which models to select (modnames):

```
psi_size <- estpsi(cod_free_size,
  model = mod_size, # list of models for each carcass class
  modnames = c(L = "xep01", S = "xep01")) # specify 1 model for each carcass class
```

2. Pass a list of models with exactly one model selected for each carcass class:

```
psi_size <- estpsi(cod_free_size,
  model = list(L = mod_size[["L"]]["xep01"], S = mod_size[["S"]]["xep01"]))
```

Note that it is entirely possible (and easy) to select different model forms for each class. For example, there would be some justification for choosing xep01 for S but xep1 for L based on AICc scores. However, we opted to use the same model form for each size class because the xep01 model, which was selected as the "best" model for the pooled data, also worked well for each of the size classes individually.

Once ψ has been estimated, $\widehat{dwp}$ can be calculated and formatted for GenEst:

```
data_dwp <- estdwp(psi_size, ncarc = getncarc(cod_free_size), forGenEst = T)
dim(data_dwp) # nsim = number of simulated dwp's for each turbine and carcass class
#> [1] 10000    3
head(data_dwp) # the first few rows of the data frame
#>      turbine L          S
#> 1    t1       0.714    0.538
#> 2    t3       1.000    0.400
#> 3    t4       0.833    0.800
#> 4    t8       0.750    1.000
#> 5    t11      0.667    0.500
#> 6    t13      1.000    0.833


# data_dwp can be used directly from the command line as the data_dwp argument
# in GenEst::estM, or it can be exported to a .csv file for importing into the
# GenEst GUI:
exportGenEst(data_dwp, file = "mydwp.csv")
```

References Cited

Akaike, H., 1974, A new look at the statistical model identification: IEEE Transactions on Automatic Control, v. 19, no. 6, p. 716–723. [Also available at https://doi.org/10.1109/TAC.1974.1100705.]

Allen, D.M., 1974, The relationship between variable selection and data augmentation and a method for prediction: Technometrics, v. 16, no. 1, p. 125–127.

Andersen, E.B., 1970, Sufficiency and exponential families for discrete sample spaces: Journal of the American Statistical Association, v. 65, no. 331, p. 1248–1255.

Arnett, E.B., Schirmacher, M., Huso, M.M.P., and Hayes, J.P., 2009a, Effectiveness of changing wind turbine cut-in speed to reduce bat fatalities at wind facilities: Annual report submitted to the Bats and Wind Energy Cooperative, Bat Conservation International, Austin, Texas.

Arnett, E.B., Schirmacher, M.R., Huso, M.M.P., and Hayes, J.P., 2009b, Patterns of bat fatality at the Casselman Wind Project in south-central Pennsylvania—2008 annual report: Annual report submitted to the Bats and Wind Energy Cooperative and the Pennsylvania Game Commission, Bat Conservation International, Austin, Texas.

Bañuelos-Ruedas, F., Angeles-Camacho, C., and Rios-Marcuello, S., 2011, Methodologies used in the extrapolation of wind speed data at different heights and its impact in the wind energy resource assessment in a region, *in* Suvire, G.O., ed., Wind farm—Technical regulations, potential estimation and siting assessment: Rijeka, Croatia, InTech, p. 97–114.

Berger, J.O., Bernardo, J.M., and Sun, D., 2012, Objective priors for discrete parameter spaces: Journal of the American Statistical Association, v. 107, no. 498, p. 636–648. [Also available at https://doi.org/10.1080/01621459.2012.682538.]

Burnham, K.P., and Anderson, D.R., 2002, Model selection and multimodel inference—A practical information-theoretic approach: New York, Springer-Verlag, 488 p.

Canty, A.J., and Ripley, B.D., 2021, boot—Bootstrap R (S-Plus) functions—R package ver. 1.3-27: Cran R Project web page, accessed February 12, 2021, at https://CRAN.R-project.org/package=boot.

Cavanaugh, J.E., 1997, Unifying the derivations of the Akaike and corrected Akaike information criteria: Statistics & Probability Letters, v. 33, no. 2, p. 201–208. [Also available at https://doi.org/10.1016/S0167-7152(96)00128-9.]

Counihan, J., 1975, Adiabatic atmospheric boundary layers—A review and analysis of data from the period 1880–1972: Atmospheric Environment, v. 9, no. 10, p. 871–905.

Dalthorp, D., and Huso, M.M., 2022, dwp R package (ver. 1): Cran R Project web page, accessed January 27, 2022, at https://CRAN.R-project.org/package=dwp.

Dalthorp, D., Huso, M.M., and Dail, D., 2017, Evidence of absence (ver. 2.0) software and user guide: U.S. Geological Survey Data Series 1055, 109 p. [Also available at https://doi.org/10.3133/ds1055.]

Dalthorp, D., Madsen, L., Huso, M.M., Rabie, P.A., Wolpert, R., Studyvin, J., Simonis, J., and Mintz, J., 2018, GenEst statistical models—A generalized estimator of mortality: U.S. Geological Survey Techniques and Methods, book 7, chap. A2, 13 p. [Also available at https://doi.org/10.3133/tm7A2.]

Dalthorp, D., Simonis, J., Madsen, L., Huso, M., Rabie, P., Mintz, J., Wolpert, R., Studyvin, J., and Korner- Nievergelt, F., 2018, GenEst—Generalized mortality estimator—R package ver. 1.4.4: Cran R Project web page, accessed June 8, 2020, at https://CRAN.R-project.org/package=GenEst.

Davison, A.C., and Hinkley, D.V., 1997, Bootstrap methods and their applications: Cambridge, United Kingdom, Cambridge University Press, 582 p.https://doi.org/10.1017/CBO9780511802843

Efroymson, M.A., 1960, Multiple regression analysis, in Ralston, A., and Wilf, H.S., eds., Mathematical methods for digital computers: New York, Wiley, 293 p.

Grodsky, S.M., Behr, M.J., Gendler, A., Drake, D., Dieterle, B.D., Rudd, R.J., and Walrath, N.L., 2011, Investigating the causes of death for wind turbine-associated bat fatalities: Journal of Mammalogy, v. 92, no. 5, p. 917–925.

He, Y., Monahan, A.H., and McFarlane, N.A., 2013, Diurnal variations of land surface wind speed probability distributions under clear-sky and low-cloud conditions: Geophysical Research Letters, v. 40, no. 12, p. 3308–3314. [Also available at https://doi.org/10.1002/grl.50575.]

Hull, C.L., and Muir, S., 2010, Search areas for monitoring bird and bat carcasses at wind farms using a Monte-Carlo model: Australasian Journal of Environmental Management, v. 17, no. 2, p. 77–87.

Huso, M.M.P., and Dalthorp, D., 2014, Accounting for unsearched areas in estimating wind turbine-caused fatality: The Journal of Wildlife Management, v. 78, no. 2, p. 347–358. [Also available at https://doi.org/10.1002/jwmg.663.]

Huso, M.M.P., Som, N., and Ladd, L., 2012, Fatality estimator user's guide (ver. 1.2, December 2018): U.S. Geological Survey Data Series 729, 22 p. [Also available at https://doi.org/10.3133/ds729.]

Huso, M.M.P., Wilson, Z., and Arnett, E.B., 2011, Density-weighted area estimation: Proceedings of the Conference on Wind Energy and Wildlife Impacts, May 2–11, 2011, Trondheim, Norway.

Kaltschmitt, M., and Wiese, A., 2007, Wind energy, in Kaltschmitt, M., Streicher, W., and Wiese, A., eds., Renewable energy—Technology, economics, and environment: Berlin, Springer, p. 49–65.

Korner-Nievergelt, F., Behr, O., Brinkmann, R., Etterson, M.A., Huso, M.M.P., Dalthorp, D., Korner-Nievergelt, P., Roth, T., and Niermann, I., 2015, Mortality estimation from carcass searches using the R-package carcass—A tutorial: Wildlife Biology, v. 21, no. 1, p. 30–43.

Madsen, L., Dalthorp, D., Huso, M.M.P., and Aderman, A., 2020, Estimating population size with imperfect detection using a parametric bootstrap: Environmetrics, v. 31, no. 3, p. e2603. [Also available at https://doi.org/10.1002/env.2603.]

Maurer, J.D., Huso, M., Dalthorp, D., Madsen, L., and Fuentes, C., 2020, Comparing methods to estimate the proportion of turbine-induced bird and bat mortality in the search area under a road and pad search protocol: Environmental and Ecological Statistics, v. 27, p. 769–801.

McCullagh, P., and Nelder, J.A., 1983, Generalized linear models (2d ed.): Boca Raton, Florida, Chapman & Hall CRC, 532 p.

Nelder, J.A., and Wedderburn, R.W.M., 1972, Generalized linear models: Journal of the Royal Statistical Society. Series A (General), v. 135, no. 3, p. 370–384.

Prakash, S., and Markfort, C.D., 2020, Experimental investigation of aerodynamic characteristics of bat carcasses after collision with a wind turbine: Wind Energy Science, v. 5, no. 2, p. 745–758.

Prakash, S., and Markfort, C.D., 2021, Development and testing of a three-dimensional ballistics model for bat strikes on wind turbines: Wind Energy, v. 24, no. 12, p. 1407–1425. [Also available at https://doi.org/10.1002/we.2638.]

Rabie, P.D., Riser-Espinoza, D., Studyvin, J., Dalthorp, D., and Huso, M., 2021, Performance of the GenEst fatality mortality estimator compared to the Huso and Shoenfeld estimators: Prepared for American Wind Wildlife Institute, Washington, D.C., accessed June 12, 2023, at https://pubs.er.usgs.gov/publication/70217783.

Richards, S.A., 2005, Testing ecological theory using the information-theoretic approach—Examples and cautionary results: Ecology, v. 86, no. 10, p. 2805–2814. [Also available at https://doi.org/10.1890/05-0074.]

Studyvin, J., and Rabie, P., 2022, windAC—Area correction methods—R package ver. 1.2.10: Cran R Project web page, accessed October 17, 2023, at https://CRAN.R-project.org/package=windAC.

Vuong, Q.H., 1989, Likelihood ratio tests for model selection and non-nested hypotheses: Econometrica, v. 57, no. 2, p. 307–333.

Whitney, W.O., and Mehlhaff, C.J., 1987, High-rise syndrome in cats: Journal of the American Veterinary Medical Association, v. 191, no. 11, p. 1399–1403.

Wolpert, R., and Coleman, J., 2015, acmeR—Implements ACME estimator of bird and bat mortality by wind turbines, accessed March 2022, at https://cran.r-project.org/package=acmeR.

Appendix 1. Brief Introduction to the Carcass Distributions

Model Anatomy

Estimating the expected proportion of carcasses lying inside the searched area involves three steps (fig. 2):

1. Modeling carcass density (carcasses per square meter) as a function of location or distance from turbine;

2. Normalizing the fitted carcass densities to create a probability density surface; and

3. Integrating the probability density over the searched area.

Each model is a Poisson regression, which is a generalized linear model (GLM) with log link (McCullagh and Nelder, 1983) such that

$$y_i \sim \text{Poisson}(\lambda_i)$$

$$\log(\lambda_i) = XB$$

so that

$$\lambda_i = e^{XB} \tag{1.1}$$

where

y_i is the carcass count within measurement unit i at an average distance of x_i m from the turbine[*], and

λ_i is the expected number of carcasses in that unit.

In each model (dropping the index, i), XB is of the form

$$p(x) + c + \log(\varepsilon(x, c)) \tag{1.2}$$

where

c is an indicator variable for the search class,

$p(x)$ is a polynomial with coefficients to be estimated in the regression, and

$\log(\varepsilon(x, c))$ is the offset term, that is, the exposure or area (square meters) in search class c of the measurement unit.

Models of this form are distinguished by the form of the polynomial, whether, for example, it is linear, quadratic, or some other form. It is not necessary to know or estimate the detection probabilities in each search class. Rather, specifying c as a covariate in the model accounts for the relative detection probabilities among search classes. Because the search class term serves only to modify the intercept, we can ignore the search class term, c, in this discussion and consider the simple case where all the area inside the search radius is searched. Then, the exposure would be proportional to x, and the form simplifies, to

$$\lambda(x) \propto e^{p(x) + \log(x)} = x e^{p(x)} \tag{1.3}$$

We abbreviate such models as xep followed by digits to indicate the form of their polynomial. For example, xep12 would have a polynomial with linear and quadratic terms and would expand to

$$e^{a + \beta_1 x + \beta_2 x^2 + \log(\varepsilon(x))} \tag{1.4}$$

We represent a $\log(x)$ term by "0," and a $1/x$ term by "i." Thus xep01 would correspond to $p(x) = \beta_0 \log(x) + \beta_1 x$ and xepi0 to $p(x) = \beta_i 1/x + \beta_0 \log(x)$. In practice, the constant term, a, is a nuisance parameter that is lost when the model is normalized to integrate to 1.

---

[*]The measurement unit for xy-grid data is a 1 square meter patch of ground. For the other data types, it is a 1-meter (m) annulus with inner and outer radii at $x \pm 0.5$ m.

Normally, if the leading coefficient[**] of $p(x)$ is negative, the model corresponds to a probability distribution, which can be extended beyond the search radius, and the model can be extrapolated to estimate the fraction of carcasses falling outside the search radius. However, if the model also has a $\log(x)$ term, that coefficient (β_0) must be greater than -2 for the model to yield a proper probability distribution. Similarly, a $1/x$ coefficient (β_i) must be negative. In all cases, if the leading coefficient is positive, the model is not extensible outside the search radius and cannot be used to estimate the fraction falling outside the search radius.

All the extensible fitted Poisson regression models can be converted into probability distributions in the exponential family simply by normalizing the fitted model by dividing by its integral from 0 to ∞. In this way, some of the models give rise to familiar distributions, like the normal, lognormal, gamma, and exponential. For example, a normalized xep01 model is a gamma distribution, and a normalized xep model with polynomial $p(x) = \beta_0 \log(x) + \beta_1 \log(x)^2$ is a lognormal distribution. However, although all the models do give rise to distributions in the exponential family, most do not have familiar names. In all cases, the ddFit function automatically normalizes the fitted distance functions whenever possible, thereby converting them to probability distributions. The package also includes functions for calculating the PDF, the CDF, quantiles, and random deviates for all the distance distributions in much the same format as the d/p/q/r family of R functions for other distributions. For the fitted distance distributions in the dwp package, the functions ddd, pdd, qdd, and rdd serve as analogs to R's dnorm, pnorm, qnorm, and rnorm for the normal distribution.

In addition to the xep models that have offset = log(exposure) to account for the area searched in each ring, ddFit will fit a few models that have a modified offset that serves dual purposes. In addition to accounting for area sampled in each ring, the modified offset alters the shape of the distribution in a standard way as the distance changes. These xep-like distributions with modified offset that are fit by ddFit include the normal, exponential, x^2, Maxwell-Boltzmann, and inverse Gaussian. Finally, a model that assumes the carcass density is constant throughout the searched area is fit (table 1.1).

By default, a standard set of 12 distributions is fit, although others can be fit as well (table 1.2). Enter ?ddFit for details on the function, including how to specify which distributions to fit.

[**]The leading coefficient is the β associated with the highest power of x in the polynomial.

Table 1.1.   Poisson regressions, standard distributions.

[GLM, generalized linear model; NA, not applicable]

Distribution	GLM form and density	Extensibility
Constant	$\log y \sim \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x$	NA
xep1	$\log y \sim x + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_1 x}$	$\beta_1 < 0$
xep01 (gamma)	$\log y \sim \log x + x + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_0 \log x + \beta_1 x}$	$\beta_0 > -2, \beta_1 < 0$
xep2 (Rayleigh)	$\log y \sim x^2 + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_2 x^2}$	$\beta_2 < 0$
xep02	$\log y \sim \log x + x^2 + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_0 \log x + \beta_2 x^2}$	$\beta_0 > -2, \beta_2 < 0$
xep12	$\log y \sim x + x^2 + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_1 x + \beta_2 x^2}$	$\beta_2 < 0$
xep012	$\log y \sim \log x + x + x^2 + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_0 \log x + \beta_1 x + \beta_2 x^2}$	$\beta_0 > -2, \beta_2 < 0$
xep123	$\log y \sim x + x^2 + x^3 + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_1 x + \beta_2 x^2 \beta_3 x^3}$	$\beta_3 < 0$
xep0123	$\log y \sim \log x + x + x^2 + x^3 + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_0 \log x + \beta_1 x + \beta_2 x^2 + \beta_3 x^3}$	$\beta_0 > -2, \beta_3 < 0$
lognormal	$\log y \sim \log x + \log^2 x + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x e^{\beta_0 \log(x) + \beta_1 \log(x)^2}$	$\beta_1 < 0$
truncated normal	$\log y \sim x + x^2 + \text{offset}(\log(\varepsilon(x)) - \log x)$ $f(x) \propto x e^{\beta_1 x + \beta_2 x^2}$	$\beta_2 < 0$
Maxwell-Boltzmann	$\log y \sim x^2 + \text{offset}(\log(\varepsilon(x)) + \log x)$ $f(x) \propto x^2 e^{\beta_2 x^2}$	$\beta_2 < 0$

Table 1.2.   Poisson regressions, supplementary distributions.

[GLM, generalized linear model]

Distribution	GLM form and density	Extensibility
xep0 (Pareto)	$\log y \sim \log x + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x^{\beta_0 + 1}$	$\beta_0 < -2$ and $x\varepsilon[1, \infty]$
xepi0 (inverse gamma)	$\log y \sim 1/x + \log x + \text{offset}(\log(\varepsilon(x)))$ $f(x) \propto x^{\beta_0 + 1} e^{\beta_1/x}$	$\beta_0 < -2$ and $\beta_i < 0$
Chi-squared	$\log y \sim \log x + \text{offset}(\log(\varepsilon(x)) - x/2)$ $f(x) \propto x^{\beta_0 + 1} e^{-x/2}$	$\beta_0 < -2$
Exponential	$\log y \sim x + \text{offset}(\log(\varepsilon(x)) - \log(x))$ $f(x) \propto e^{\beta_1 x}$	$\beta_1 < 0$
Inverse Gaussian	$\log y \sim 1/x + x + \text{offset}((\log(\varepsilon(x)) - 5/2 \log(x))$ $f(x) \propto \sqrt{1/x^3} e^{\beta_1 x + \beta_i 1/x}$	$\beta_i, \beta_1 < 0$

Model Characteristics

There are no universally accepted criteria for determining which model is definitively the "best." When the model must be extrapolated well beyond the range of data, the problem becomes much more difficult. How well a model fits within the range of the data may give very little indication of how well it will fit outside the range of data. Conventional tools like Akaike Information Criterion (AIC) can be very useful for comparing how well various models fit the data, but two models that are virtually indistinguishable within the range of data can diverge substantially outside the range of data. However, there are some useful guidelines for model selection that are discussed briefly here and in greater detail in the discussions in section, "Examples" and in appendix 3.

1. Models with lower Akaike information criterion corrected for finite sample size (AICc) scores give better fits within the range of the data. Models with AICc scores within about 4 are considered indistinguishable by this measure. AICc scores are less informative about model reliability for extrapolating outside the search radius. Models with ΔAICc scores greater than 10 do not seem to fare as well as models with ΔAICc <5, where ΔAICc is the difference between AICc value for a given model and the lowest AICc value among the models tested.

2. Models with heavy tails tend to overestimate the fraction of carcasses falling outside the search radius when the search radius is long. The weight of the tail is primarily a function of the degree of the polynomial. For example, the tail of an xep distribution with degree 2 will have a lighter tail than one with degree 1. The weight of the tail also depends somewhat on the other terms in the polynomial, with lower order terms contributing to lighter tails.

3. Road and pad searches (RP) or other search patterns in which the proportion of area searched per ring decreases with distance from turbine and the search coverage is low but non-zero for a broad range of distances greater than about 40 m can lead to reduced stability in the models. In this situation:

   - There will be a tendency to overestimate the fraction of carcasses inside the search radius if no carcasses are found at great distances, and we recommend using a relatively heavy-tailed distribution in these situations to account for the possibility of significant numbers of carcasses outside the search radius; and

   - There will be a tendency to strongly underestimate the fraction of carcasses inside the search radius if carcasses are found at great distances, and we recommend using a relatively light-tailed distribution in these situations.

4. Distributions of small carcasses tend to have heavier tails than distributions of large carcasses because the small carcasses are more strongly affected by wind and are blown farther under high wind conditions and are not flung as far under light wind conditions, as shown in ballistic simulations in appendixes 2 and 3.

Appendix 2.   Mechanistic and Ballistics Models of Carcass Deposition

To explore properties of carcass density models and their performances on known data, we generated carcass distributions according to a ballistics model similar to those used by Hull and Muir (2010) and by Prakash and Markfort (2020). Models are fit to simulated carcass dispersions under several different scenarios with different carcass types (eagle and bat), wind regimes, and carcass search protocols. Because the actual (simulated) carcass distributions are known, models can be compared with each other based on how well they are able to predict the true data under various conditions.

Mechanistic Model

Physics puts a fairly strict limit on how far carcasses may fall from the turbine. Gravity inexorably pulls them to the ground after a brief time. Excluding cases where (1) a bird or bat is not killed immediately but is able to fly or walk a great distance after getting struck by a turbine blade; (2) a carcass is carried a great distance by a scavenger; (3) a carcass is swept upward or downward by a wind with a strong vertical component; or (4) a carcass is blown or rolls a great distance after landing, we can use well-known differential equations from ballistics modeling to get a qualitative understanding of carcass spatial distributions around a turbine. The general shape of the tails of the carcass distribution is of particular interest because it informs the choice of parametric distribution for empirical modeling of carcass distributions and estimation of the proportion of carcasses landing in the searched areas.

Mechanistic differential equation models can be useful in addressing simple, qualitative questions about carcass dispersion. Which general shapes of distributions are best in which types of situations? How well does Akaike information criterion corrected for finite sample size (AICc) perform for identifying models that will most accurately predict the fraction of carcasses landing in the searched area? How do carcass size and wind speed broadly affect the distribution of carcasses on the ground?

Ballistics Model

Unlike the qualitative understanding achievable through mechanistic models, a quantitative understanding that accurately predicts the actual distributions of carcasses in the field as a function of environmental covariates such as wind profile and turbine size is much more difficult and may not be feasible. The precise distribution critically depends on a number of parameters and assumptions that are difficult to validate in the field. The following factors all have significant impacts on the resulting carcass distribution on the ground, but none of them are known with any practical, quantitative degree of reliability:

- Wind speed and direction at the time of strike,

- Strike location on the blade,

- Carcass velocity at the time of strike,

- Change in carcass velocity upon strike (angle of the blade, transfer of momentum from blade to carcass),

- Drag coefficient and its change during flight as carcass changes shape and direction,

- Rotor speed at time of collision,

- Carcass fluttering or spinning after collision,

- Dependence of wind velocity on position (including wind shear and wake effect),

- Turbulence behind the blade,

- Displacement by scavengers after landing, and

- Even the type of ground cover (which affects the wind gradient with height).

If we make a few simplifying assumptions, the ballistics equations are not difficult. We start by assuming that the animal is killed upon impact with the turbine blade and that the wind is blowing horizontally (with no vertical component) and normal to the plane of the blades. Like Prakash and Markfort (2021) and Hull and Muir (2010), we ignore the wake effect behind the turbine, but we do include wind shear (Bañuelos-Ruedas and others, 2011).

To calculate a carcass, landing position after being struck by a turbine, we first define a coordinate system centered at the base of the turbine to track carcass position in three dimensions, $s = (x, y, w)$, where x is the horizontal position in the plane of the turbine blades, y is height above the ground, and w is the horizontal distance normal to the plane of the turbine blades in the downwind direction. We then use ballistics differential equations to describe the trajectory of the carcass as a function of time.

Define x and y to be the horizontal and vertical displacements of a carcass from the turbine base, and w, the distance downwind from the turbine. If $s(t)$ is the position vector of the carcass as a function of time and $v(t) = s'(t)$ is the velocity, then the pair $(s(t), v(t))$ is a 6-dimensional vector which satisfies a set of first order differential equations:

$$s'(t) = v(t) \tag{2.1}$$

$$v'(t) = \eta(v(t) - w(s(t))) \parallel v(t) - w(s(t)) \parallel -G \tag{2.2}$$

where

$w(s(t))$	is the wind velocity at $s(t)$,
$n = 9.807/y'_T{}^2$ (with y'_T . = terminal velocity of the carcass)	is an aerodynamic constant associated with each carcass type, and
$G = [0, 9.807, 0]$	is the acceleration due to gravity.

Given an initial carcass strike point on the blade and an initial carcass velocity, we track the trajectory of the carcass through space using the method of lines with $\Delta t = 0.01$ seconds. The final position of the carcass on the ground is (x, w) when $y = 0$.

This set of governing equations is similar to those used by Prakash and Markfort (2021) but allows wind speed to vary with carcass position, thereby enabling the accommodation of wind shear and wake effect. Like Hull and Muir (2010) and Prakash and Markfort (2021), we ignore the wake effect. We do account for wind shear through the Wind Power Law, or

$$w(y) = w(y_n)y/y_n^a \tag{2.3}$$

where

$w(y)$	is the wind speed at y meters above the ground,
y_n	is the height of the nacelle, and
a	is the Hellman coefficient (Counihan, 1975).

Additionally, we assume that the cross-sectional area of the carcass presented to the wind (A) and the drag coefficient (C_d) are constant, so the term $0.5\,C_d A \rho /m$ (Hull and Muir, 2010; Prakash and Markfort, 2 021) simplifies to $9.807/y'_T{}^2$ after noting that the terminal velocity is the vertical velocity at which the wind resistance equals the force of gravity so there is no longer any vertical acceleration. Therefore, terminal velocity, $\{y', T\}$, is the velocity such that $\rho/2\,C_d A/m \times y'_T{}^2 - g = 0$, from Prakash and Markfort's (2021) equations 3 and 4, with no acceleration in the horizontal directions, giving $\rho/2\,C_d A/m = G/y'_T{}^2 = \eta$. Thus, the terminal velocity, which is relatively easy to measure and is relatively well known, is used in lieu of the product of parameters with largely unknown values (C_d and A).

Table 2.1.   Parameters used in the ballistics model and simulations.

[m, meter; m/s, meter per second; m/s², meter per second squared]

Parameter	Value
Fixed across scenarios	
nacelle height	80 m
blade length	$r_0 = 45$ m
tip speed ratio	6
acceleration due to gravity	$G = 9.807$ m/s²
Hellman coefficient	$\alpha = 0.22$
strike position on the blade	random uniform
Varying among scenarios	
terminal velocity	$V_t = 8.8$ m/s for bats, and $V_t = 25$ m/s for eagles
wind speed	low, moderate, high
wind profile	constant, varying
carcass initial horizontal velocity	0, varying (up to 8 m/s)
search plot type	cleared, roads and pads

Simulation Parameters

We used the ballistics model described in the previous section to explore the effects of carcass type, wind regime, strike position, and flight speed on carcass distributions on the ground (section, "Distributions of Carcass Distances"). Random carcass locations were generated from the carcass distributions and then sampled to create simulated datasets. Distributions were fit to the simulated datasets using ddFit. Statistical properties of the models are explored and discussed in appendix 3 to provide guidance on *dwp* model selection.

Parameter values used in the carcass dispersion models and simulations are a mix of fixed constants (such as turbine specifications) that are constant across the simulation scenarios and variable parameters that may differ among carcasses and simulation scenarios (table 2.1).

The Hellman coefficient governs the wind shear or the tendency for wind speed to increase with height. It is a measure of the coarseness of the landscape texture (Kaltschmitt and Wiese, 2007), with smaller values corresponding to smoother surfaces and smaller changes in wind speed with height. Values typically range from about 0.05 for very smooth ground to 0.5 or 0.6 for cities with high-rise buildings.

The terminal velocity is a measure of an object's air resistance and is an amalgam of the drag coefficient (C_d), the cross-sectional area presented to the wind, the air density—all of which are unknown—and the object's mass. The terminal velocity is used in lieu of the product of parameters with largely unknown values. We assume a value of 8.8 meters per second (m/s) for the terminal velocity for bats, which we calculated from data given by Grodsky and others (2011).[1] We used a terminal velocity of 25 m/s for eagles, which is slightly less than it is for live cats (Whitney and Mehlhaff, 1987).

Turbine nacelle height and blade length match those of the 1.8-megawatt Vestas V90 turbine. We assume the turbine is operating at a tip speed ratio of 6; that is, that the tips of turbine blades are moving at six times the wind speed, while the wind varies.

Simulation scenarios assume that strike position is uniformly distributed in the plane of the blades and that the initial velocity of the carcass after being struck is equal to the sum of the blade and carcass velocities at the point of impact.

---

[1]Grodsky and others (2011) reported that bats took an average of 10–12 seconds to fall 91.4 meters in an experiment. Terminal velocity is the velocity, y', at which the carcass is no longer accelerating as it falls but is falling at a constant rate, that is $y'' = -G + A\left(y'_T\right)^2 = 0$, where y'_T is the terminal velocity, A is the projected area, and G is the acceleration due to gravity. Solving for y, we get $y(t) = y_0 - \log\left(\cosh\left(3.1305t\sqrt{A}\right)\right)/A$. Then, taking the initial height as $y_0 = 91.4$ and the final height after 11 seconds as $y(11) = 0$, we get $A = 0.1263$ and a terminal velocity of $y'_T = 8.8$ m/s for bats, which is in good agreement with the findings of Prakash and Markfort (2020).

Wind speeds behind the turbine and at the height of the nacelle were either constant at 4, 8, or 12 m/s or varying with low, moderate, or high average. Winds in the low wind speed regime were assumed to be Weibull distributed with mean of 5.32 m/s and standard deviation of 2.78 m/s. The high wind speed regime also was assumed to be Weibull distributed, with mean of 9.18 m/s and standard deviation of 2.1 m/s. The moderate wind speed regime mimics the winds measured at a site in the Netherlands, with a mean of 7.5 m/s, standard deviation of 1.25 m/s, and having a PDF of wind speeds of $f(w) = 10^{-0.04(w-7)^2 - 0.8}$, which approximately fits the moderate wind profile (row W_{80}, column JJA) of He and others (2013, fig. 2). In all cases, no carcasses were produced when wind speeds were <3.5 m/s. Wind direction was assumed to be uniformly distributed, with no prevailing direction.

Animal flight velocity was assumed to be constant in one-half the scenarios and variable in one-half. In the constant scenarios, carcasses attain the same velocity as the turbine blade at the point of impact upon being struck, implying a coefficient of restitution of $e = 1$, following Hull and Muir (2010) and Prakash and Markfort (2020). In the variable scenarios, flight speed relative to the ground was assumed to be 8 m/s with 0 vertical component but horizontal direction varied uniformly between 0 and 2π, with the component of carcass initial velocity in the plane of the blades (x) erased at the moment of impact and the component normal to the plane of the blades (w) added to the wind velocity.

The zone where carcasses are likely to fall would be expected to extend to a radius at least as great as the turbine blades (r_0). Carcasses may land outside that range if the impact from the blade knocks them beyond r_0 in the plane of the blades or if the wind carries them beyond r_0 in the windward direction. Figure 2.1 compares the carcass fall zones for eagles and bats at wind speeds of 4, 8, and 12 m/s, showing images of the disk spanned by the turbine blades projected onto carcass dispersion patterns on the ground. The contours are mappings of concentric rings around the turbine hub to the position on the ground where an eagle or bat carcass would be expected to fall under the given wind speed and carcass type under the conditions of the simulation scenario (table 2.1).

Eagle carcasses are large, and, compared to small carcasses, their trajectories are less affected by the wind. Like a baseball getting struck by a bat, an eagle directly hit by a turbine blade is flung in the direction dictated by the blade, with relatively little modification of its subsequent trajectory by the wind. In the simulations, under low wind speed conditions (4 m/s) eagle carcasses fell near the line of the blades. With stronger winds, carcasses landed significantly downwind (fig. 2.1).

By contrast, bat carcasses are small and may be carried aloft by the wind for great distances, like dandelion seed thrown into the air on a windy day. In the simulations, even at the relatively low wind speed of 4 m/s, carcasses struck near the top of the swept area were blown more than 50 m, significantly beyond the $r_0 = 45$ m blade length (fig. 2.1). Greater windspeeds had little effect on the distribution of carcasses in the plane of the blades (left-right in fig. 2.1) but a substantial effect on the distribution of carcasses in the downwind direction (bottom to top in fig. 2.1). With a wind of 12 m/s, carcasses near the top of the swept area were blown close to 200 m.
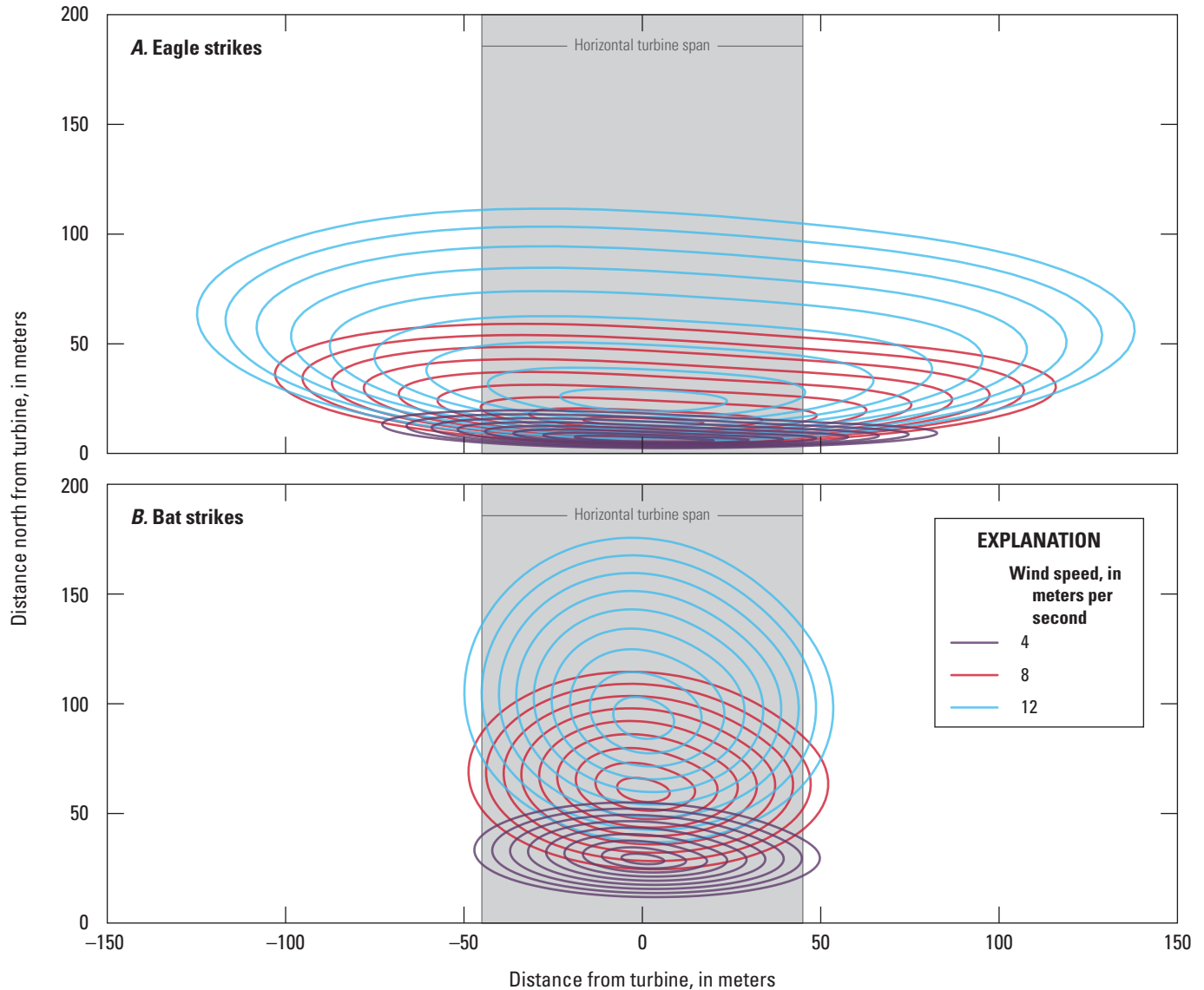
Figure 2.1.   Projection of (*A*) eagle and (*B*) bat turbine-carcass strike points onto the ground. Turbine is located at (0, 0), with wind blowing north (bottom to top) at a constant velocity. Lines are projected images of turbine-carcass strike points onto the ground. Contours correspond to strike points on the blades in concentric rings with radii of 5, 10, ..., 45 meters. Shading marks the span of the turbine blades in the *x* direction.

Distributions of Carcass Distances

Because bats are more heavily influenced by the wind than eagles and are blown farther, a smaller proportion of bats is expected to fall very close to the turbine, as reflected in the histograms of simulated carcass distances shown in figures 2.2 and 2.3.

The eagle histograms are qualitatively similar to those constructed for large birds by Hull and Muir (2010, fig. 2). Under most of the simulated conditions, the distributions have relatively flat tops over much of the range of distances but then decrease precipitously. The most consequential difference between our model and that of Hull and Muir (2010) is that the latter neglects the downwind direction. Because eagle carcasses are typically not blown far by the wind after being struck, the downwind direction plays a relatively minor role in determining eagle distance distributions, which explains the similarity between the patterns in figure 2.2 and those observed by Hull and Muir (2010).
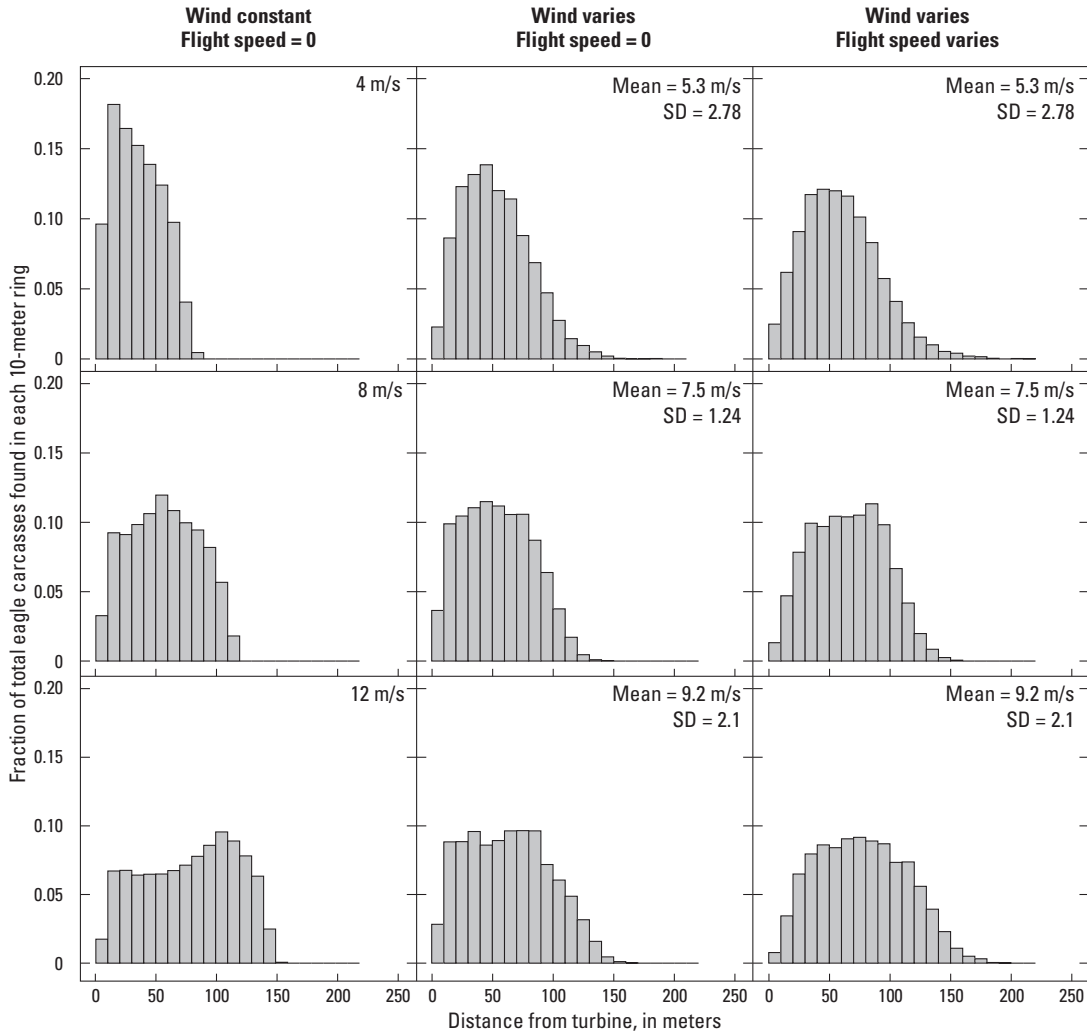
Figure 2.2.   Distances of eagle carcasses to turbine. Upon collision with a turbine blade, carcass velocity takes on the velocity of the blade plus the flight speed. Flight speed is either assumed to be 0, in which case initial carcass velocity equals turbine blade velocity at point of impact, or variable, in the range of wind speed plus or minus 18 meters per second (m/s). SD, standard deviation.

The bat histograms (fig. 2.3) show relatively few carcasses very near the turbines and more of a classic bell shape (albeit, somewhat right-skewed) than the eagle histograms. That these bat histograms bear little resemblance to those of Hull and Muir (2010, fig. 2) and Prakash and Markfort (2020, fig. 9) is due to the importance of the downwind direction, which is included in our model but neglected in the other two.
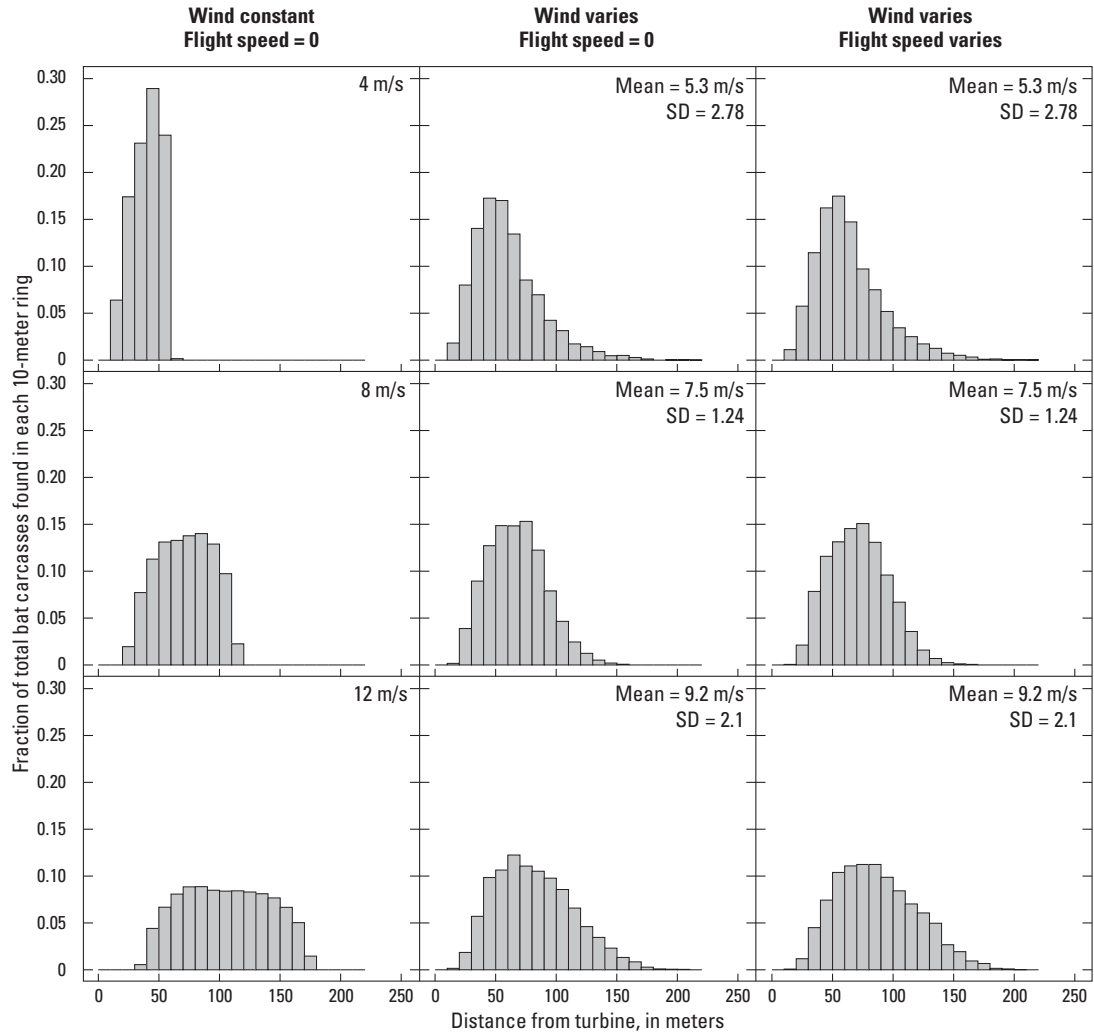
Figure 2.3.   Distances of bat carcasses to turbine. Upon collision with a turbine blade, carcass velocity takes on the velocity of the blade plus the flight speed. Flight speed is either assumed to be 0, in which case initial carcass velocity equals turbine blade velocity at point of impact, or variable, in the range of wind speed plus or minus 8 meters per second (m/s). SD, standard deviation.

Appendix 3.   Fitting Parametric Distributions to the Dispersion Patterns

The general shape of the actual distribution of carcasses varies with the ballistics parameters and environmental conditions. The performance of a fitted distribution depends on how well the distribution family's shape conforms to the actual distribution and, perhaps even more critically, how accurately the fitted model reflects the distribution of carcasses lying outside the searched area.

In practice, the search radius is finite, and a fitted model does not "know" what fraction of carcasses lie inside the search radius and has to "guess" based on its general shape. If the general shape happens to align well with the true distribution, the prediction of the fraction lying outside the search radius will be accurate; if not, the error may be substantial.

The ballistics modeling suggests that eagle carcass distributions are often likely to be relatively flat near the turbine but decrease precipitously with distance at some point (fig. 2.2), so that, under many conditions, eagle distributions would be expected to be light-tailed. Thus, relatively heavy-tailed distributions such as the lognormal and xep1 would usually not be a good match for eagle distributions. By contrast, the simulations suggest that bat carcass distributions will tend to have few carcasses very near the turbines, and, compared with the eagle distributions, a more gradual climb to a narrower peak (fig. 2.3) and have a different suite of distributions with the best fits.

Although some broad descriptions of model characteristics and their general applicability can be made, it may not be possible to determine *a priori* which particular models will provide the best predictions in a given situation. Factors like wind regime and species determine the actual distribution of carcasses, but other factors—like search radius and plot shape (for example, cleared plots or roads and pads)—can heavily influence the aptness of a model for predicting the fraction of carcasses outside the searched area. Selecting a specific model for a specific situation must be done on a case-by-case basis.

Model Accuracy in Estimating Probability of a Carcass Lying Inside the Searched Area

The accuracy of a carcass distribution model for predicting the probability that a carcass lies in the searched area depends on (1) how well the model conforms to the shape of the actual distribution and (2) how the fit inside the searched area can be extended outside the searched area. There are two types of unsearched areas—(1) unsearched areas outside a search radius, and (2) unsearched areas inside a search radius. Spatial prediction for carcasses outside a search radius involves extrapolation, and accuracy requires making effective use of additional information, apart from the data itself because a good or even perfect fit within the range of data gives no guarantee of a good or even plausible fit outside the range of the data. Prediction for unsearched areas inside a search radius involves interpolation, and the relative quality of a model's fit to the data inside a searched area (for example, Akaike information criterion corrected for finite sample size [AICc]) is a reasonable guide for model selection for interpolation. Thus, wind conditions, carcass type, search radius, and plot shape are all important factors in determining which models will provide the most accurate predictions.

We tested the model predictions for the standard *dwp* models (appendix 1) in a variety of simulation scenarios (table 2.1), with $n = 1,000$ replicates for each scenario. For each replicate, 200 random carcass distances were generated from the distance distributions arising from the ballistics models for eagles and bats under the wind regimes described in table 2.1 and appendix 2, section, "Distributions of Carcass Distances," and shown in figures 2.2 and 2.3. Random directions were generated as uniform $(0,2\pi)$ deviates. Carcasses that lay inside the searched area (fig. 3.1) were found and included in the data used for fitting the models. Carcasses that lay outside the searched area were not included. For replicates in which at least five carcasses were found, each of the standard *dwp* models was fit and $\hat{\psi}$ was calculated using the maximum likelihood estimator for the model parameters whenever the model was extensible. For each simulation scenario, boxplots of $\hat{\psi}$ are presented for each of the standard *dwp* models (figs. 3.2–3.13), ordered by degree of the polynomial in the model (table 1.1).

Accuracy of Estimated Probability of a Carcass Lying Inside the Searched Area—Eagles Under Constant Winds and Cleared Search Plots

For eagle locations simulated under constant wind conditions, the probability density functions (PDF) of the carcass-generating distributions was fairly flat out to a certain point but then decreased rapidly to zero (fig. 3.2). Under these conditions, a short search radius misses the precipitous decrease in the PDF, and the empirical models have great difficulty estimating the fraction of carcasses inside the search radius.
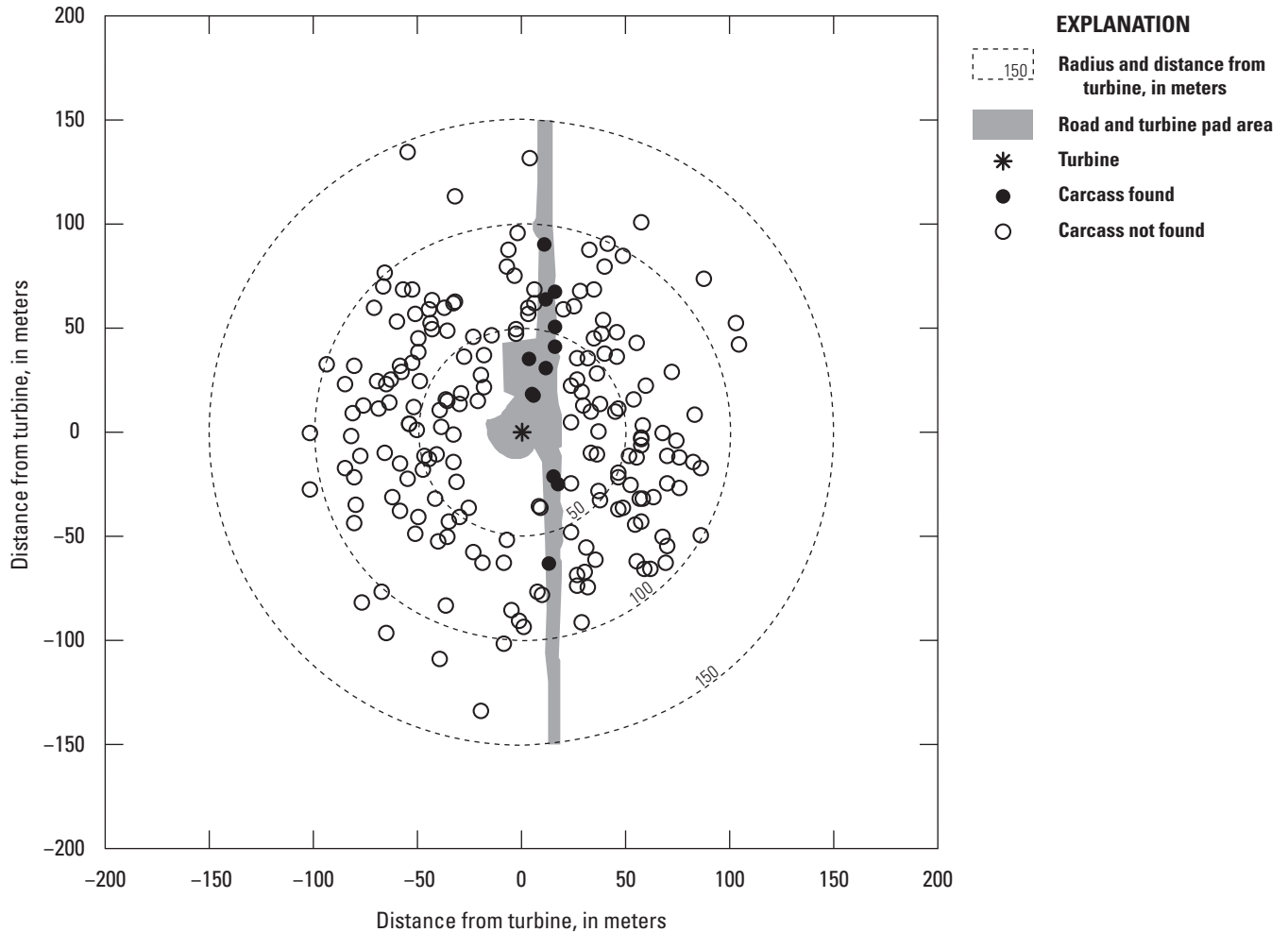
Figure 3.1.   Search areas used in the simulations. Searches were conducted out to radii of 50, 100, or 150 meters from the turbine and included either all carcasses inside the search radius (cleared plot) or only those carcasses on the roads and turbine pads (RP). Figure shows carcasses found on an RP search out to 150 meters.

At 4 meters per second (m/s), the PDF of carcasses (fig. 3.2, histogram on bottom left) begins a long decrease as distance increases beyond approximately 10 meters (m). The heavy-tailed lognormal continues the gradual decrease past the search radius, unable to accommodate the sharp decrease beginning at about 75 m. As a result, the lognormal regularly underestimates $\hat{\psi}$ under these conditions as the model implicitly assumes that there is a substantial number of carcasses past the search radius. By contrast, the distributions with degree greater than or equal to ($\geq$) 2 have an inherently greater acceleration in the rate of decrease in the PDF and tend to predict fewer carcasses inside the search radius than do the heavier-tailed distributions. With a search radius of 100 m, all carcasses fell inside the search plot, and all the fitted distributions except the lognormal routinely estimated $\hat{\psi} > 0.9$.

At 12 m/s (fig. 3.2, right column), the situation changes substantially. The PDF of carcasses is largely flat but gradually increasing with distance from the turbine until about 125 m, when the density decreases rapidly to zero. The short search radius ($r = 50$ m) entirely misses the decrease, and almost all the fitted distributions vastly overestimate the fraction of carcasses inside the search area as their implicit assumptions about how the density would converge to zero are in error, resulting in curves that underestimate the extent of the flat part of the density. The relatively heavy-tailed lognormal and xep1 distributions are forced to decrease gradually from their peak, and their degree of overestimation of $\hat{\psi}$ in this scenario is modest compared to the lighter-tailed distributions.

The distributions of carcasses within the first 100 m from a turbine were similar for winds of 8 and 12 m/s (fig. 3.2, histograms), so the $r = 50$ and 100 m search radii yield similar fitted distributions (fig. 3.2, comparing the 8 m/s panels with the 12 m/s panels for $r = 50$ and 100 m). However, the accuracy of the estimates depends on the target (fig. 3.2, true ψ).
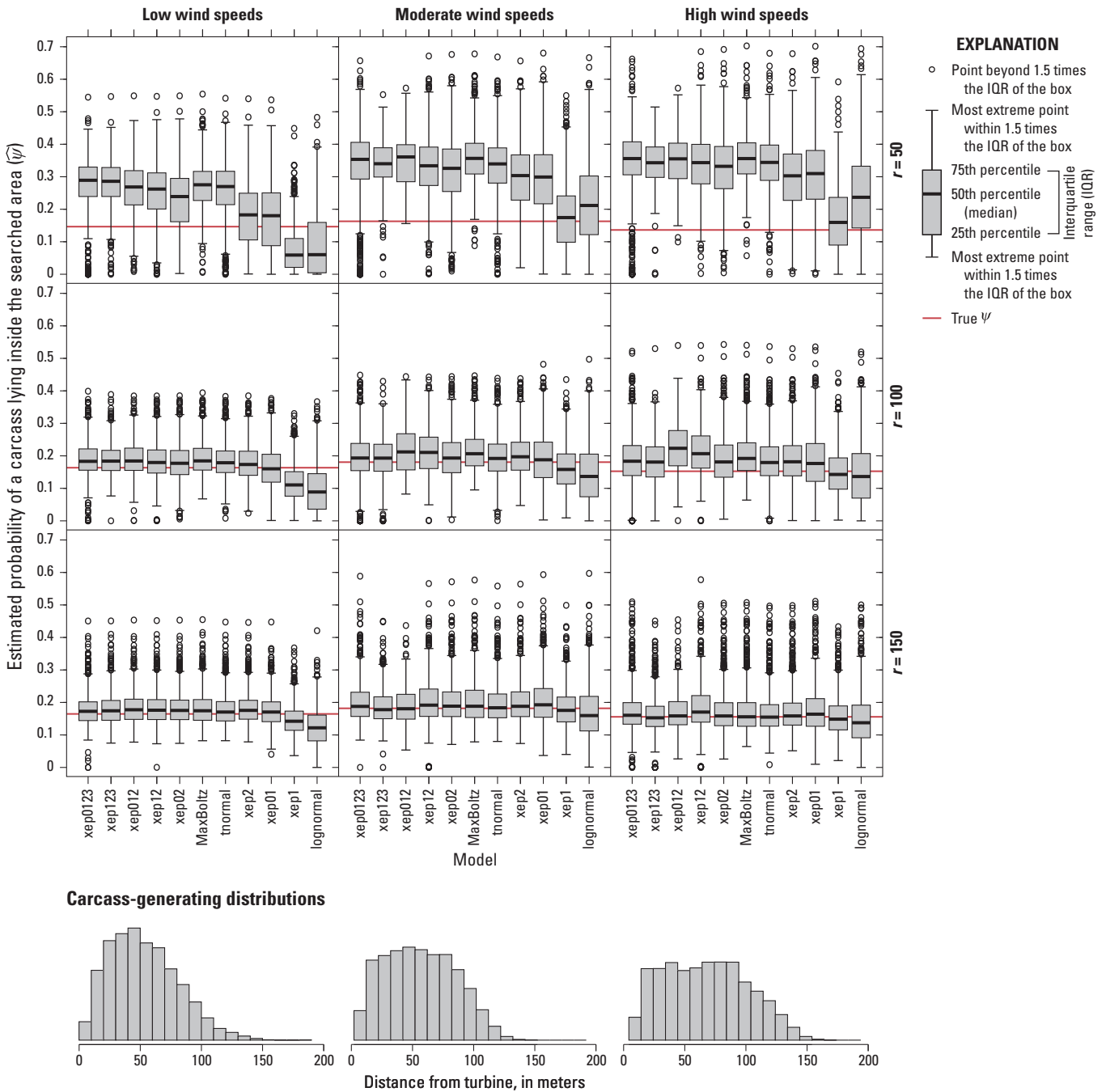
Accuracy of Estimated Probability of a Carcass Lying Inside the Searched Area—Eagles Under Constant Winds and Constant Flight Speeds for Road and Pad Searches

Under constant wind conditions, predictions across the fitted distributions were more consistent in the road and pad searches (fig. 3.3) than in the cleared plot searches (fig. 3.2). In each panel, the distinctions among the fitted distributions were slight. However, the lognormal almost always estimated a smaller fraction of carcasses inside the searched area than did all other models. The only exceptions were for a short search radius was short ($r = 50$ m) and the wind speed was 8 or 12 m/s, in which case model xep1 had the smallest (and most accurate) estimates for ψ. When the search radius was not short ($r = 100$ or 150 m), there was little distinction among the model fits.
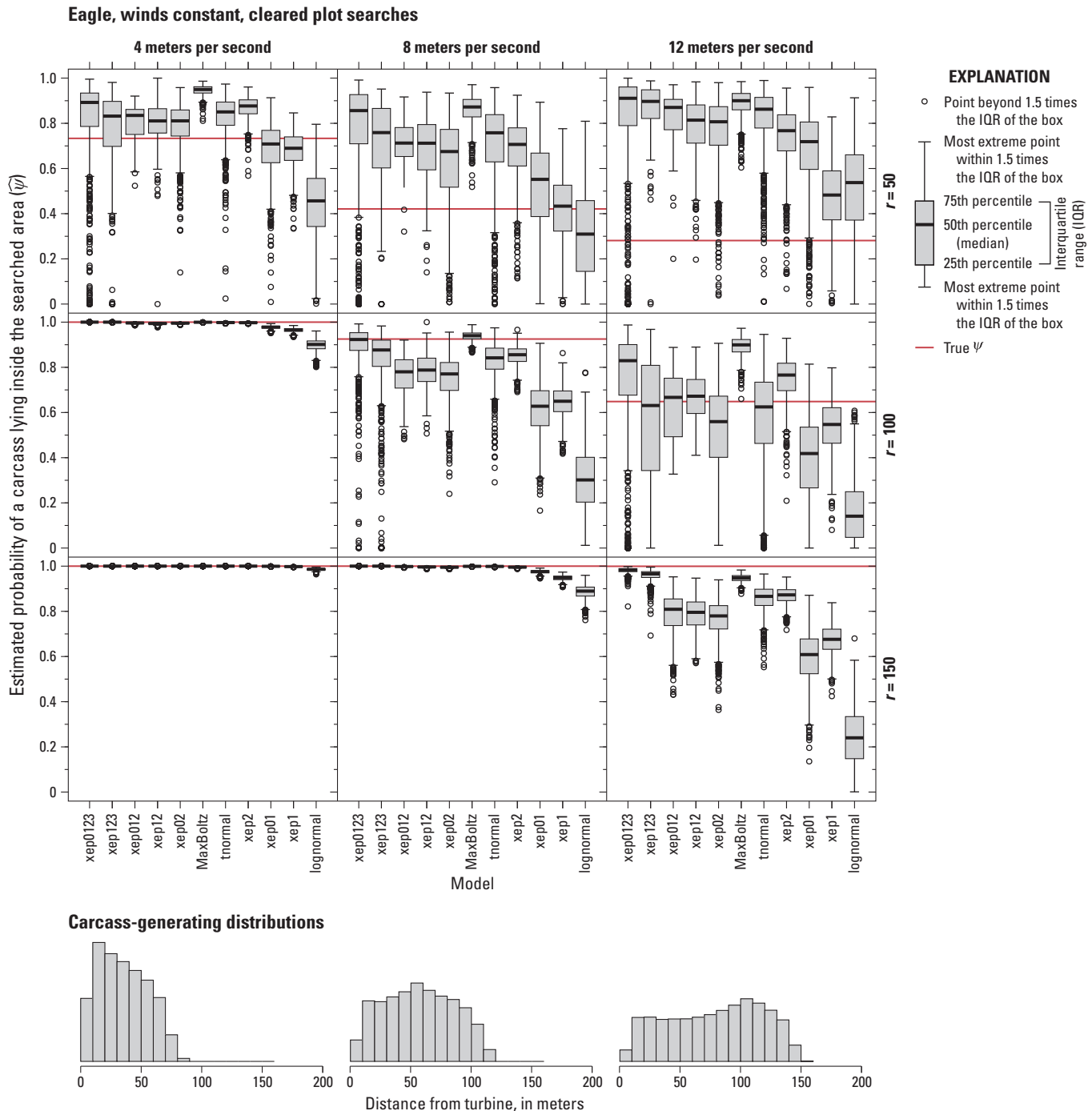
Figure 3.2. Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated eagle fatalities with winds constant, flight speed constant, and cleared-plot searches. r, search radius, in meters.

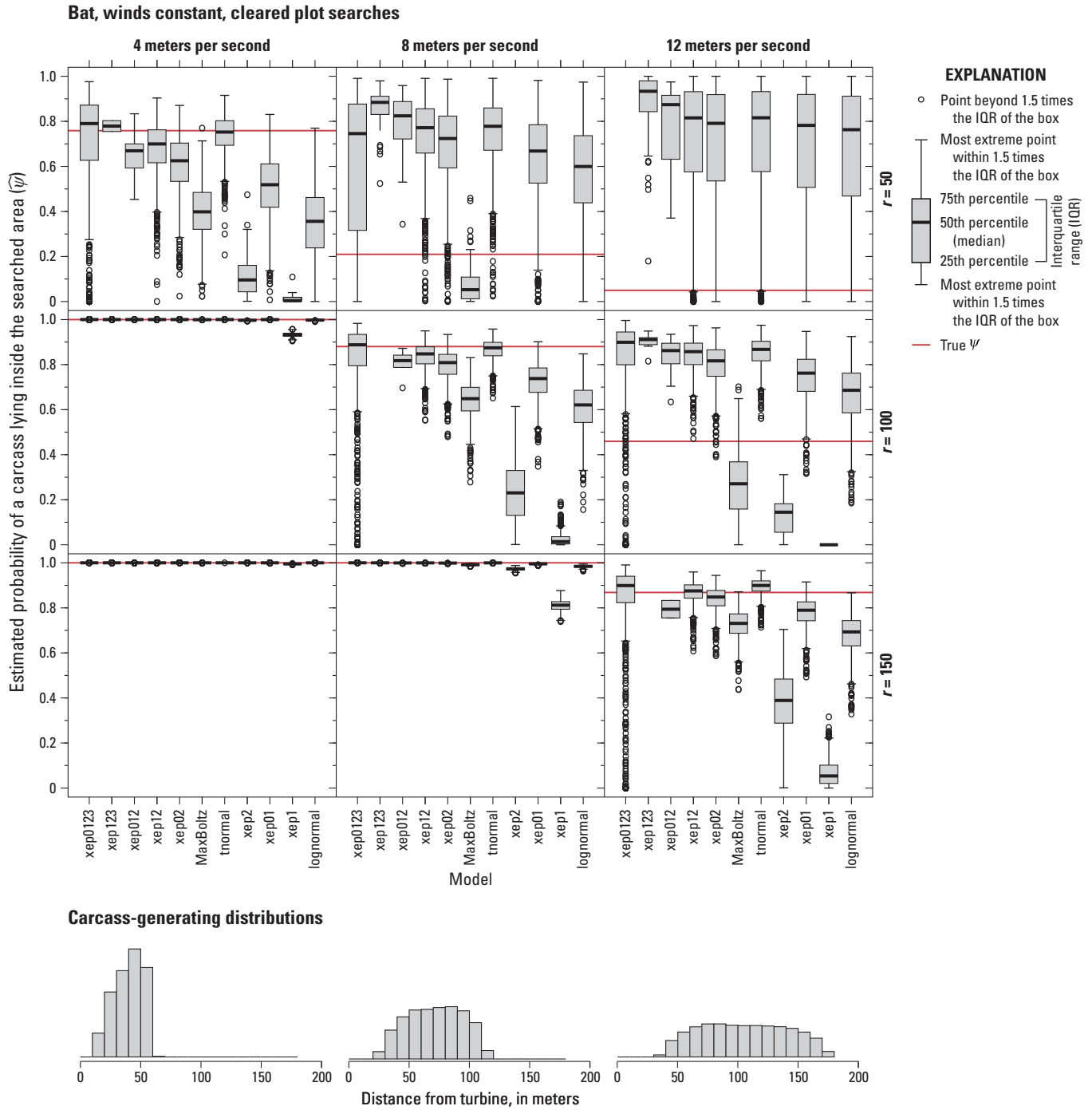**Eagle, winds constant, road and pad searches**
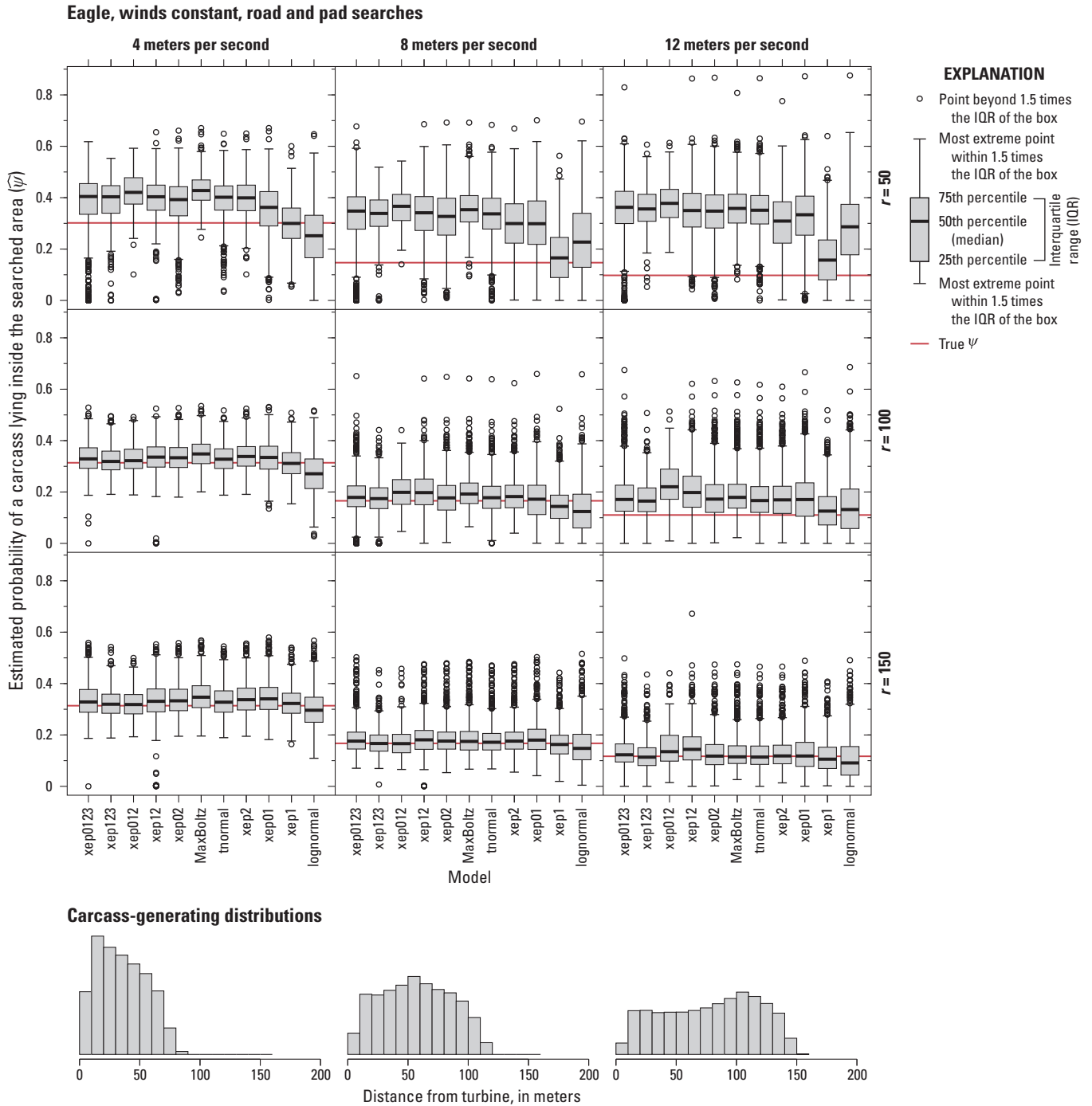
Figure 3.3.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated eagle fatalities with winds constant, flight speed constant, and road and pad searches. *r*, search radius, in meters.
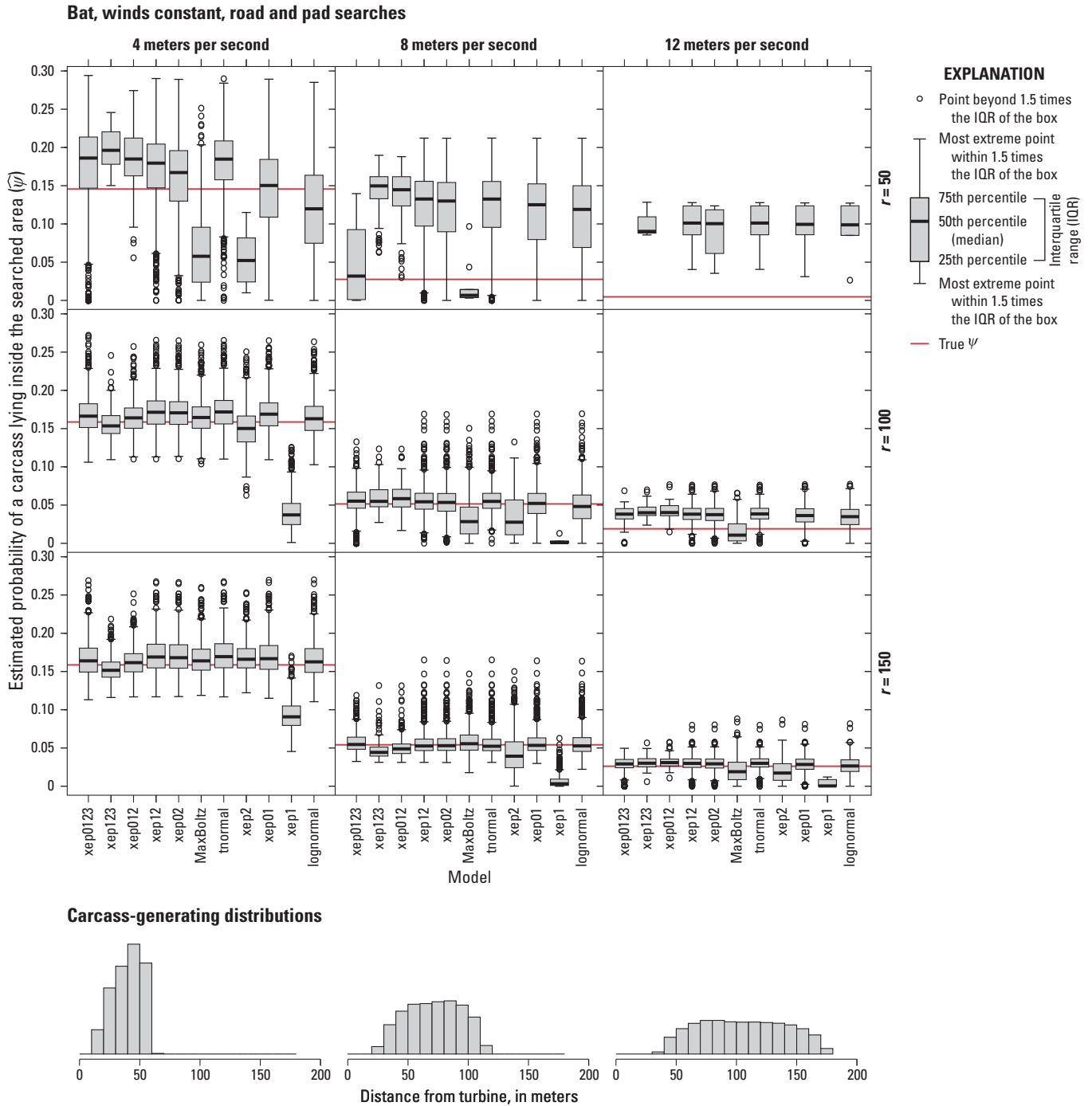
Accuracy of Estimated Probability of a Carcass Lying Inside the Searched Area—Eagles Under Varying Winds and Constant Flight Speed for Cleared Plot and Road and Pad Searches

The eagle distributions and fitted models under variable winds (figs. 3.4 and 3.5) were similar to those under constant winds (figs. 3.2 and 3.3). Refer to sections "Accuracy of Estimated Probability of a Carcass Lying Inside the Searched Area—Eagles Under Constant Winds and Cleared Search Plots" and "Accuracy of Estimated Probability of a Carcass Lying Inside the Searched Area—Eagles Under Constant Winds and Constant Flight Speeds for Road and Pad Searches" for discussion.
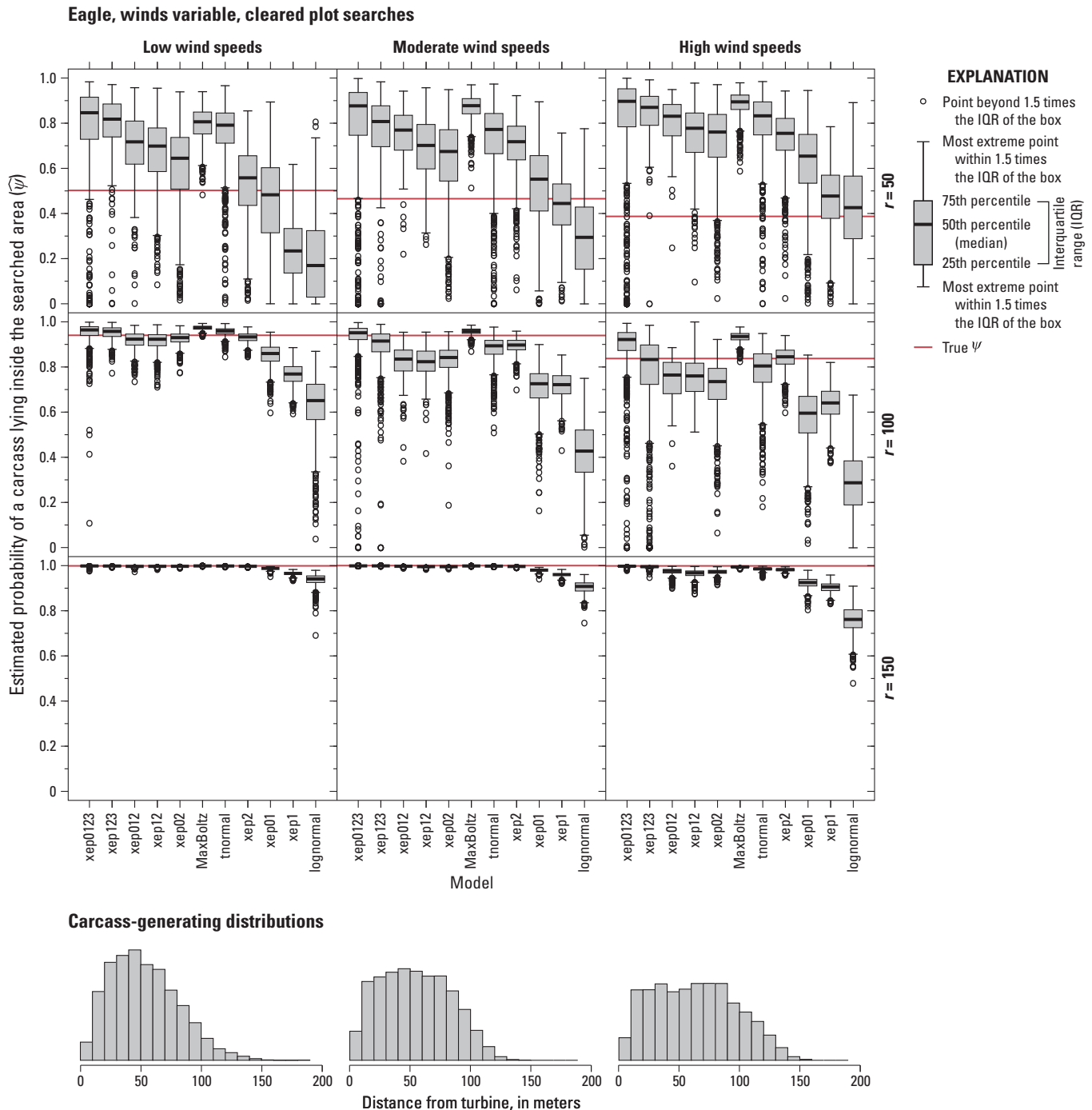
Figure 3.4. Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated eagle fatalities with winds variable, flight speed constant, and cleared plot searches. *r*, search radius, in meters.

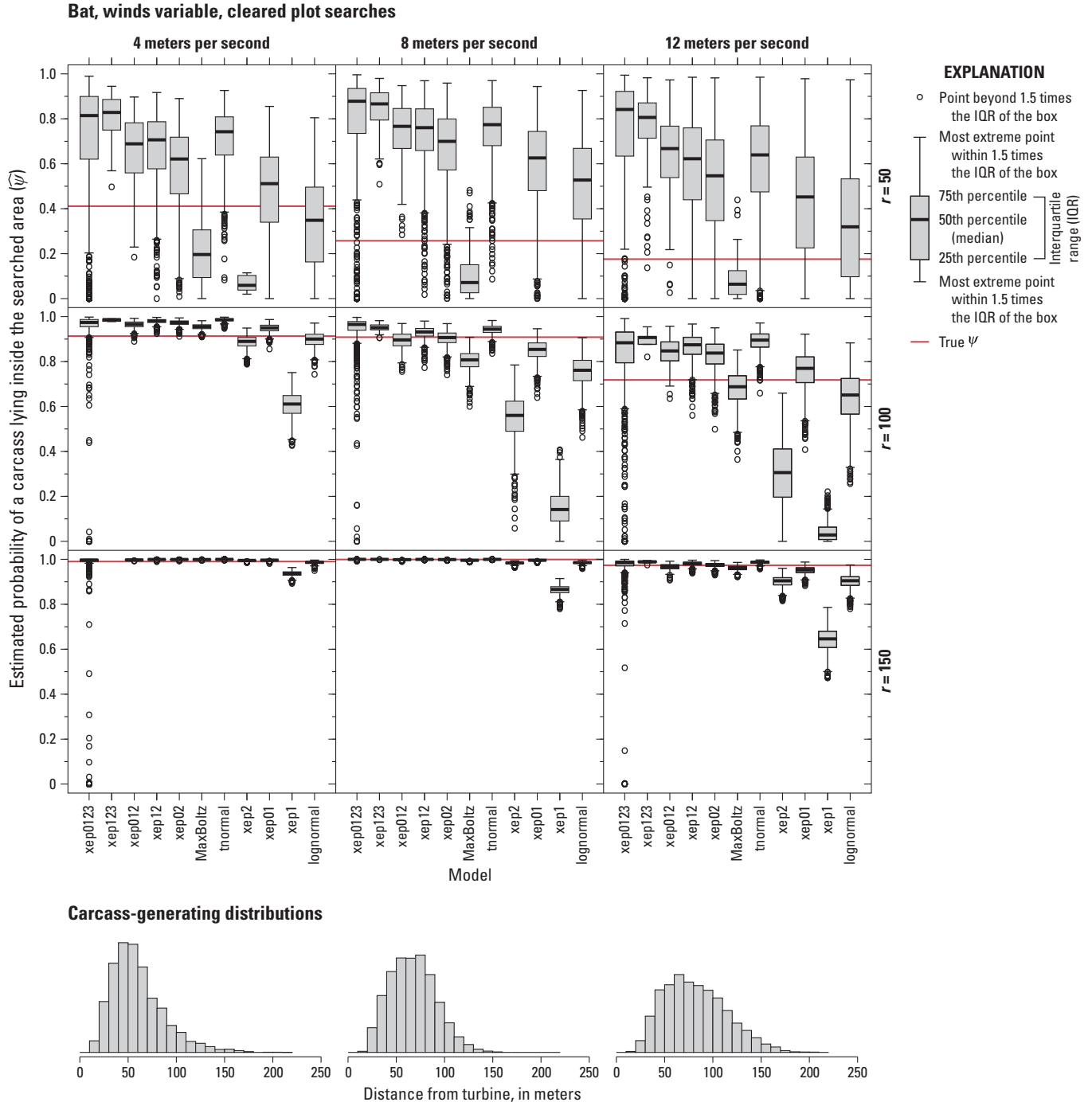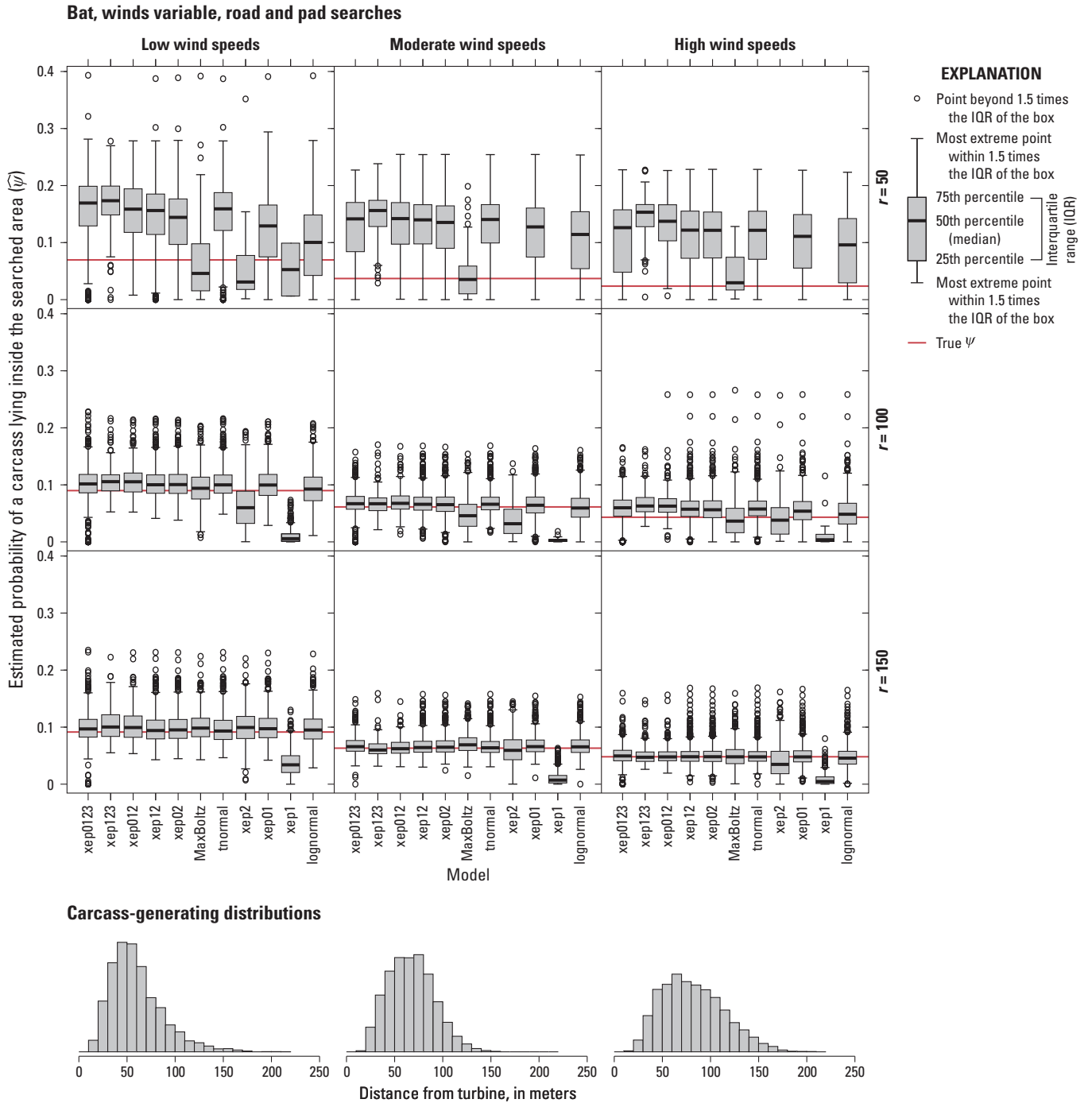**Eagle, winds variable, road and pad searches**

Figure 3.5.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated eagle fatalities with winds variable, flight speed constant and road and pad searches. *r*, search radius, in meters.

Accuracy of Estimated Probability of a Carcass Lying Inside the Searched Area—Eagles Under Varying Winds and Flight Speeds (Cleared Plot Searches, Road and Pad Searches)

The eagle distributions under variable winds and variable flight speed (figs.3.6 and 3.7) were notably less flat than the eagle scenarios in which the effect of initial flight speed was nullified upon impact by the turbine blade. The estimated proportion of carcasses inside the searched area generally decreased with the order of the fitted distribution, with the accuracy increasing with search radius. There was a marked tendency for models to overpredict $\hat{\psi}$ with road and pad searches, especially when the search radius was short ($r = 50$ m). That tendency to overpredict was fully compensated for in the lognormal and xep1 models, which, as heavy-tailed distributions, naturally tend to underpredict.
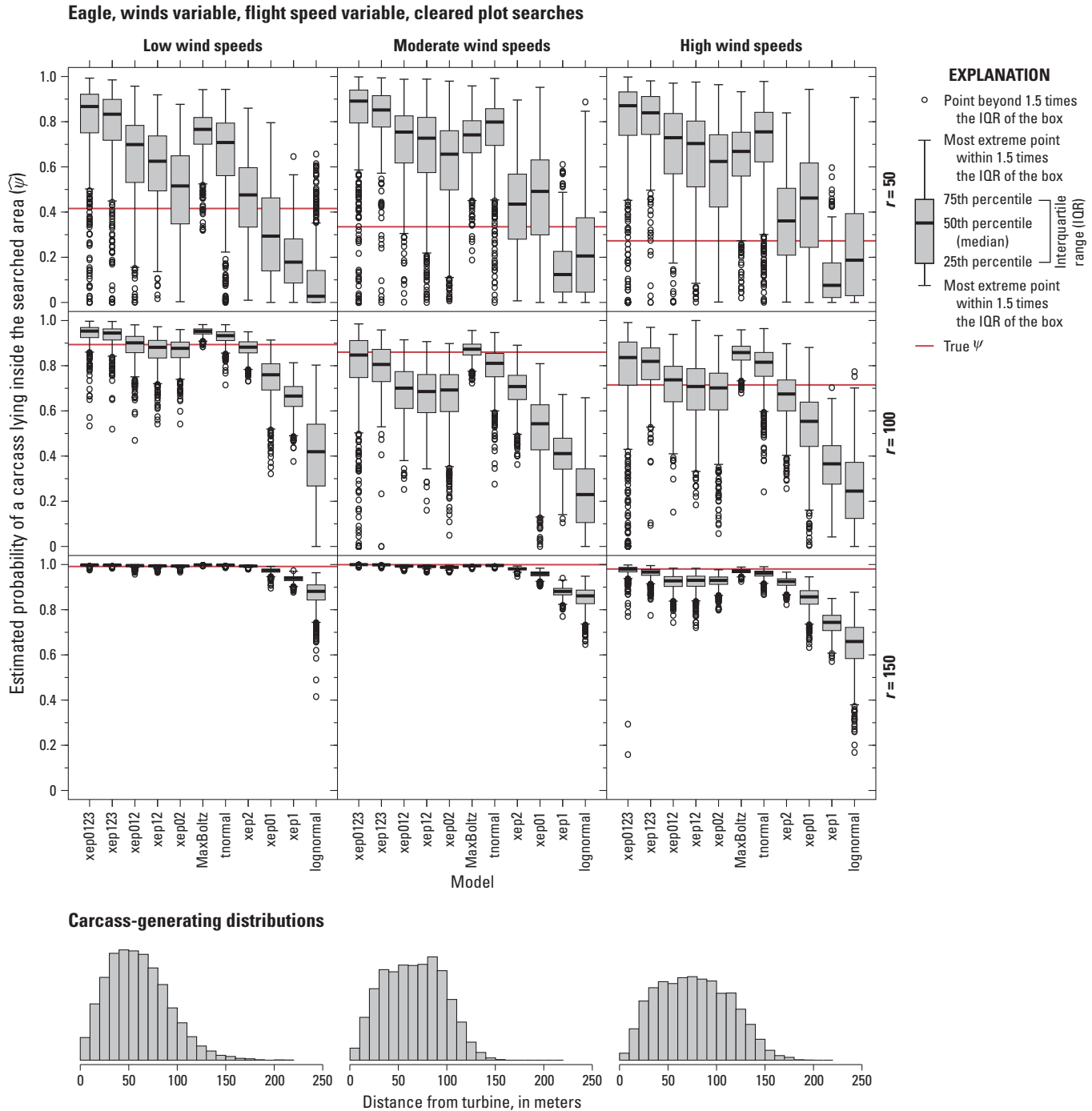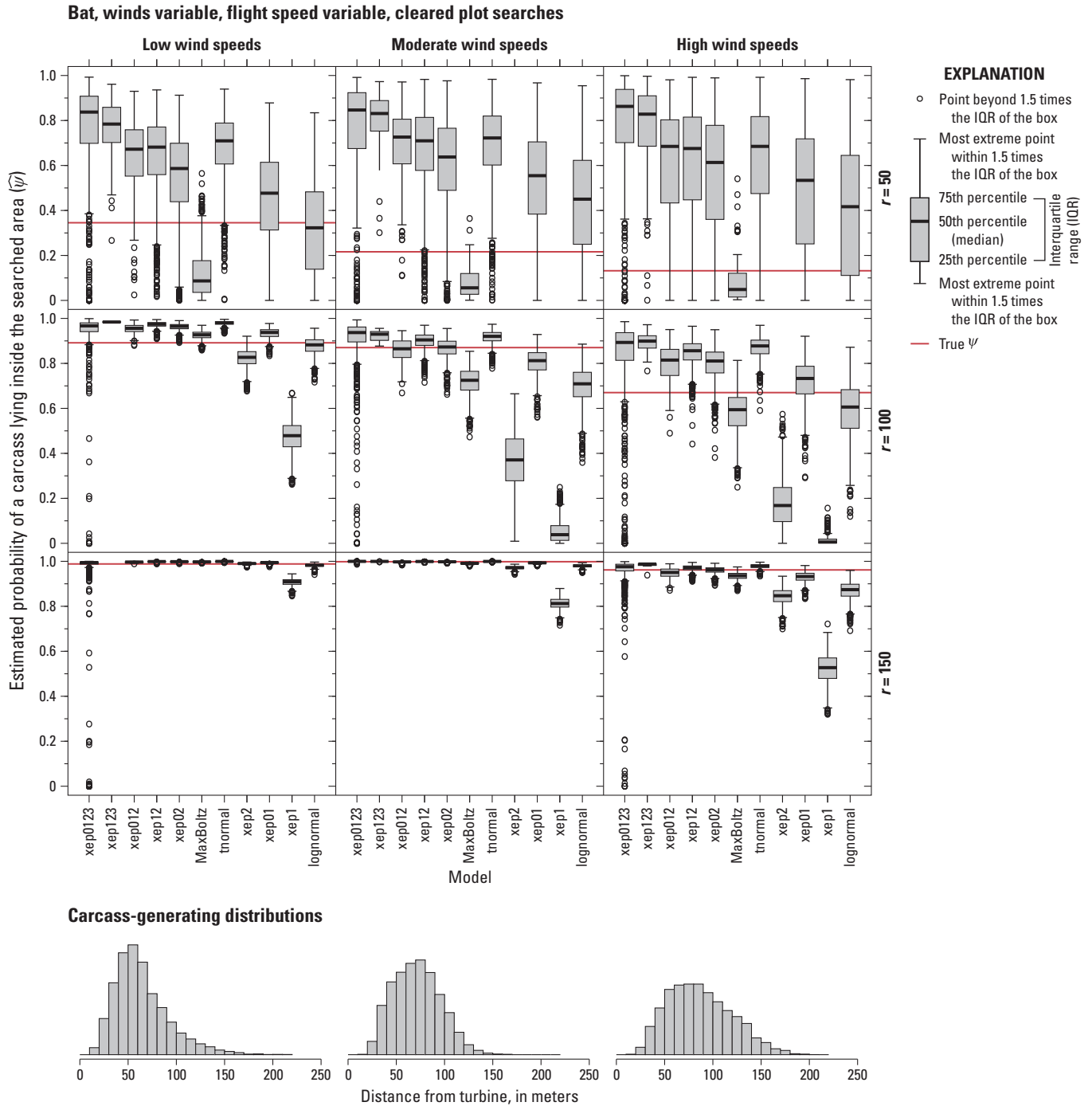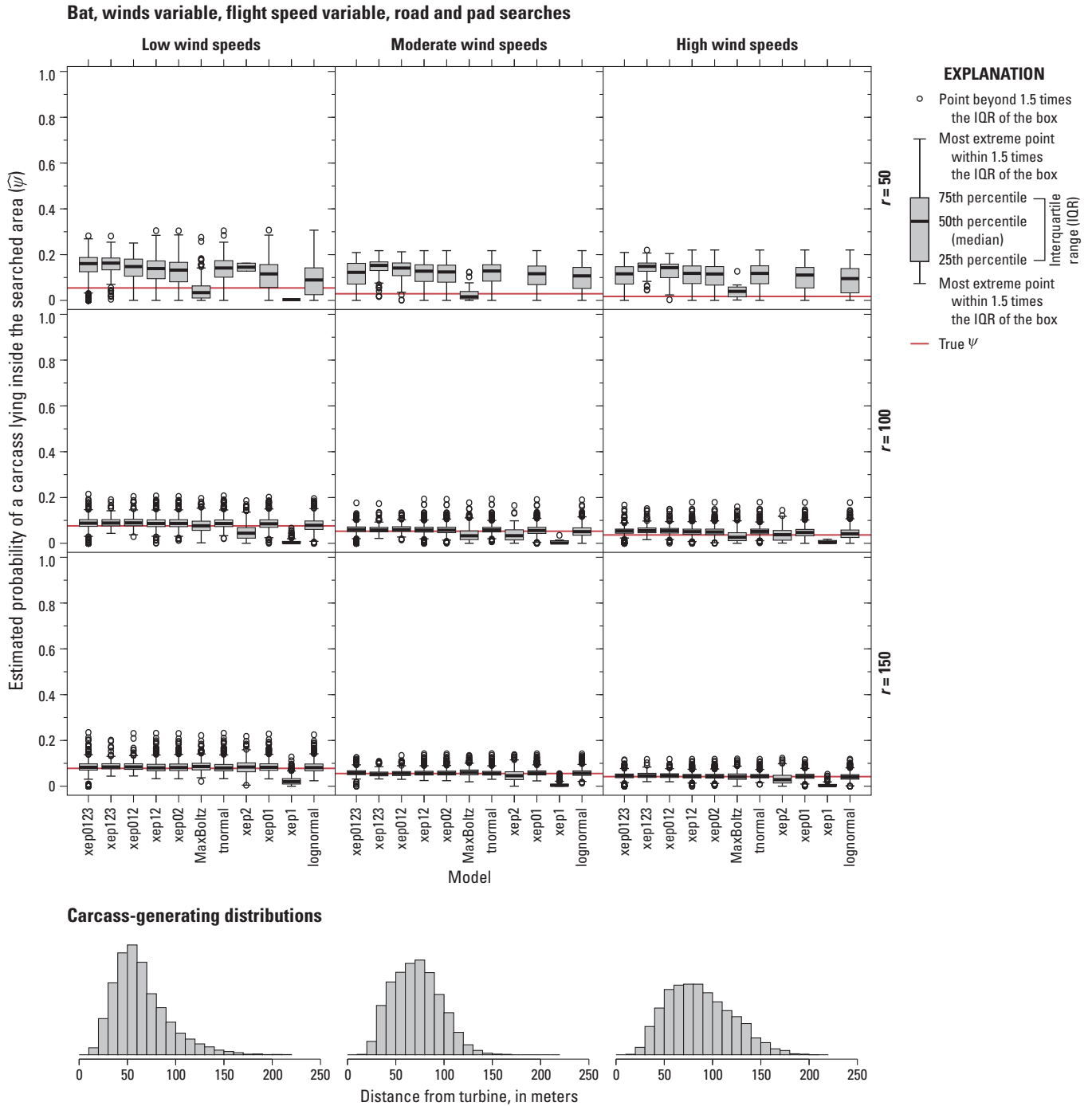

Figure 3.6.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated eagle fatalities with winds variable, flight speed variable, and cleared plot searches. *r*, search radius, in meters.
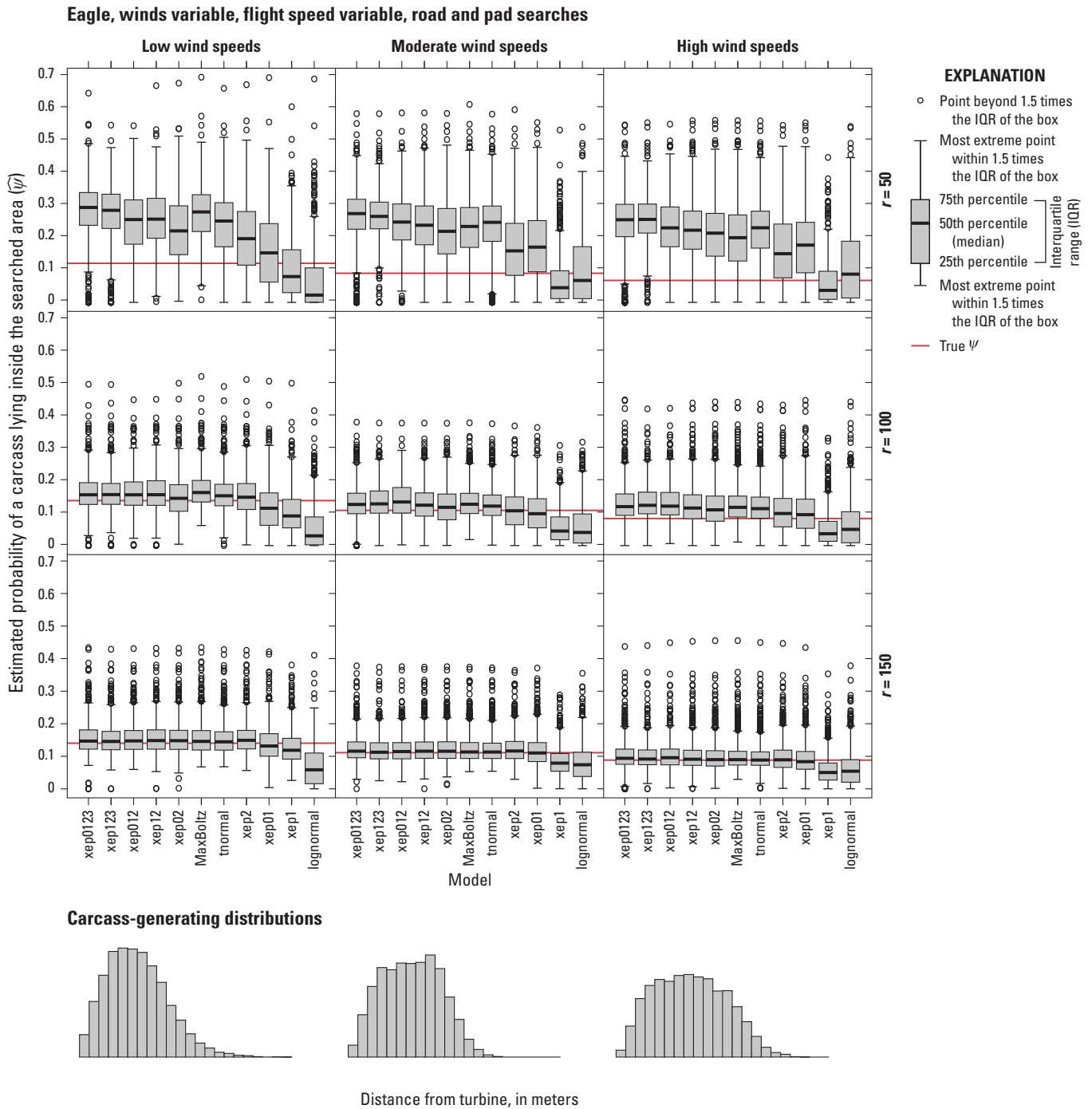
Figure 3.7.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated eagle fatalities with winds variable, flight speed variable, and road and pad searches. *r*, search radius, in meters.

Model Accuracy of Estimated Probability of a Carcass Lying Inside the Searched Area for Simulated Bat Locations

Bat carcass distributions tended to have fewer carcasses near the turbine and a somewhat more elongated right tail than the eagle distributions (figs. 3.8–3.13). As with the eagles, the models tended to overpredict by a substantial margin when the search radius was short (*r* = 50 m), variation among model predictions was much smaller for road and pad searches than for cleared plot searches, and $\hat{\psi}$ tended to decrease with the degree of the model (as reflected in the decreasing trend in the boxes in figs. 3.8–3.13).

Figure 3.8.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated bat fatalities with winds constant and cleared plot searches. *r*, search radius, in meters.

Bat, winds constant, road and pad searches

Figure 3.9.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated bat fatalities with winds constant and road and pad searches. *r*, search radius, in meters.

Figure 3.10. Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated bat fatalities with winds variable and cleared plot searches. *r*, search radius, in meters.

Bat, winds variable, road and pad searches

Figure 3.11.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated bat fatalities with winds variable and road and pad searches. *r*, search radius, in meters.

Figure 3.12. Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated bat fatalities with winds variable, flight speed variable, and cleared plot searches. *r*, search radius, in meters.

Bat, winds variable, flight speed variable, road and pad searches

Figure 3.13.   Estimated probability of a carcass lying inside the searched area for the standard models with 200 simulated bat fatalities with winds variable, flight speed variable, and road and pad searches. r, search radius, in meters.

The same pattern naturally arises with the road and pad searches. The sample is likely to not have any carcasses at great distances because the search coverage is too low there. The light-tailed distributions decrease rapidly, so there is not much probability in the tails and they overestimate the fraction of carcasses in the search radius. As the search radius increases, the heavy-tailed distributions tend to model a lot of carcasses farther out, and consequently they tend to underestimate the fraction falling in the searched area.

Akaike Information Criterion, Model Accuracy, and Probability of a Carcass Lying Inside the Searched Area

The Akaike information criterion (AICc) is a well-tested and commonly used tool for model selection (Burnham and Anderson, 2002) and has proven to be effective for distinguishing among models by the relative quality of the fit to the data. However, its validity is limited to the range of the data, and there is no guarantee that models that perform well within the range of the data will perform well—or will even be remotely plausible—when extended outside the range of the data, as must be done when predicting the proportion of carcasses that lie outside the searched area $(1 - \hat{\psi})$. To test the utility of AICc for prediction of ψ, we summarized the results from the simulations in section, "Model Accuracy in Estimating Probability of a Carcass Lying Inside the Searched Area" by ΔAICc (the difference between AICc value for a given model and the lowest AICc value among the models tested), binning the $\hat{\psi}$ values by the ΔAICc scores of the fitted model. For example, for eagles under constant wind speeds of 4 m/s and searched on cleared plots out to a radius of $r = 50$ m, we generated 1,000 simulated datasets, fit the 11 standard *dwp* models, and binned the $\hat{\psi}$ values according to their ΔAICc scores (fig. 3.14, upper left panel). The boxplot of $\hat{\psi}$ values for ΔAICc = 0 summarizes the collection of $\hat{\psi}$'s for the best-fitting model in each of the 1,000 replicate datasets. The boxplot for ΔAICc = 0–1 is a summary of the $\hat{\psi}$ values for models with ΔAICc $\in (0, 1)$ and the rightmost boxplot in each panel is for models with ΔAICc ≥ 10.

Eagles

For eagles under constant wind conditions and the initial carcass position and velocity are taken to be that of the rotor at the point of impact, the distribution (PDF) of carcasses was fairly flat out to a certain point but then decreased rapidly to zero (fig. 2.2, left panels). Under these conditions, a short search radius misses the precipitous decrease and empirical models have great difficulty estimating the fraction of carcasses inside the search radius. The key feature of the distribution—namely, the point marking the beginning of the steep decrease—is absent from the data, so measures of how well the models fit within the range of the data (AICc, for example) cannot make meaningful distinctions among the models for extrapolating outside the search radius.

In general, for cleared plots and constant wind speeds, ΔAICc was largely unrelated to accuracy in predicting ψ. For search radii of $r = 50, 100$ m, $\hat{\psi}$ tended to increase with ΔAICc (fig. 3.14), presumably because the heavier-tailed models tended to fit better within the range of data than the lighter-tailed models. However, the actual simulated carcass distributions are light-tailed, so ΔAICc gives an unreliable measure of the aptness of the model fits. For example, at 8 m/s, and a short search radius (fig. 3.14, top middle panel), the higher the ΔAICc, the poorer the prediction. However, with a longer search radius and a corresponding greater fraction of carcasses inside the search radius, the greater the ΔAICc, the better the prediction, the best-fitting models had by far the least accurate predictions (fig. 3.14, center panel).

When the search radius was close to the maximum carcass distance, as in the high winds, long search radius example, the ΔAICc tended to successfully select the model with the best estimate of ψ (bottom right panel). When windspeeds decreased (4 or 8 m/s) but search radius remained long (bottom row left and center, or middle row at left), few if any carcasses fell outside the search radius and all models did well at determining that $\psi = 1$. These scenarios are effectively interpolation rather than extrapolation given the length of the search radius and the shape of the generating distributions.
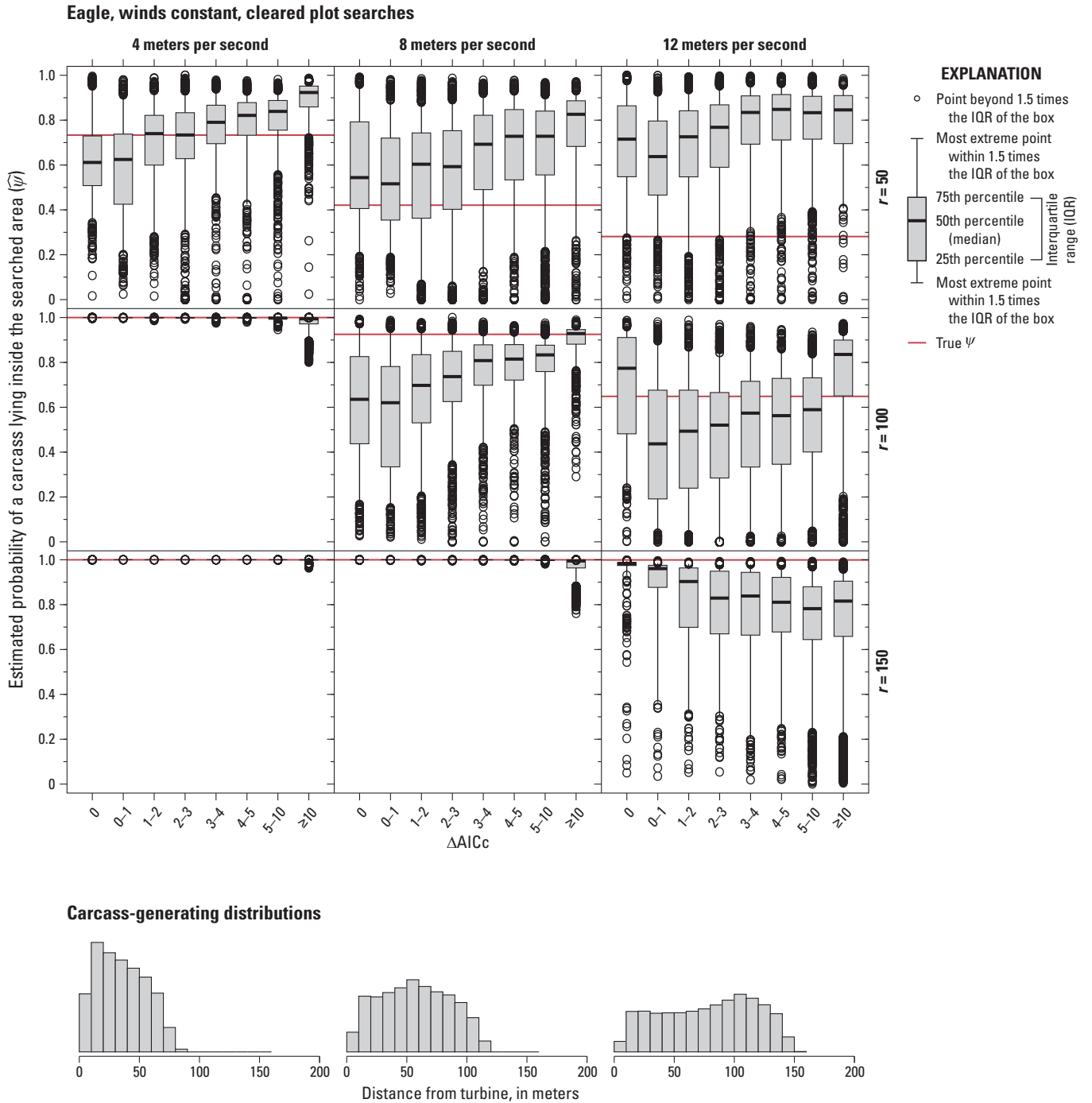
Figure 3.14.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated eagle fatalities with winds constant and cleared plot searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; r, search radius, in meters.

Because ΔAICc applies strictly to the fit of models within the range of data, we would expect ΔAICc to perform better when the fraction of carcasses inside the search radius is higher, as illustrated in figure 3.14 for r = 150 m, where close to 100 percent of the carcasses lie inside the search radius, and, in the case where there are distinctions among models (lower right panel), greater ΔAICc values are associated with poorer fits. However, in the center panel (8 m/s and r = 100 m), 90 percent of the carcasses lie inside the search plot but the best predictors tended to be the models with the poorest fits within the range of the data. In this simulation scenario, the distribution of carcasses was largely flat within the 100 m search radius but decreased rapidly to zero just outside the search radius, requiring a light-tailed distribution in order to capture the abrupt end of the distribution. However, it is the heavy-tailed distributions that fit best within that first 100 m from the turbine, hence the inverse relationship between the quality of the fit within the range of data and the adequacy for prediction outside the range of the data.

With road and pad sampling, much of the spatial prediction is in extending the models to the unsearched area inside the search radius. Because interpolation to unsearched areas within the range of the data plays a large role in prediction of ψ from road and pad data, we would expect ΔAICc to be a better indicator of model accuracy than it was for the cleared plot searches. Indeed, lower ΔAICc values do seem to be somewhat associated with more accurate prediction in the road and pad searches (figs. 3.15, 3.17, and 3.19) than in the corresponding cleared plot searches (figs. 3.14, 3.16, and 3.18, respectively).

Bats

Figure 3.20 shows bat carcass distributions and estimated probabilities that carcasses lie in the searched area for constant wind speeds of 4, 8, and 12 m/s and cleared plot search radii of 50, 100, and 150 m. As with eagles, AICc does not appear to be a useful predictor of model performance in estimating ψ except in cases where the search radius is large enough to encompass nearly the whole of the distribution.

For bat carcasses, ΔAICc appears to be even less reliable for selecting the most accurate models for ψ than it was for eagle carcasses. In some cases, the worse-fitting models (high ΔAICc) gave more accurate predictions (for example, fig. 3.20, upper right panel, with 12 m/s winds and search radius of r = 50 m); in some cases, the best-fitting models gave the most accurate predictions (for example, fig. 3.20, center panel, with 8 m/s winds and search radius of r = 100 m); in most cases, though, there was little discernible relationship between ΔAICc and accuracy (figs. 3.20–3.25).

The road and pad searches yielded accurate predictions for ψ for bats under constant or variable windspeeds (figs. 3.21, 3.23, and 3.24)[1], except when the search radius was too short to capture the distribution's decrease in the right tail, in which case models tended to vastly overestimate ψ. The ΔAICc had very little predictive value because ΔAICc measures relative fit within the range of the data but gives no information on how well the model extrapolates beyond the data or whether a given model happens to have a tail that matches that of the actual carcass distribution.

Under the conditions of figure 3.23, it is remarkable how performance appears to be almost entirely independent of AICc.

---

[1]The missing boxplots in the upper right panel of fig. 3.21 are due to so few models being able to fit viable distributions to the data because the carcass PDF did not start its decrease to zero until well outside the search radius of 50 m.
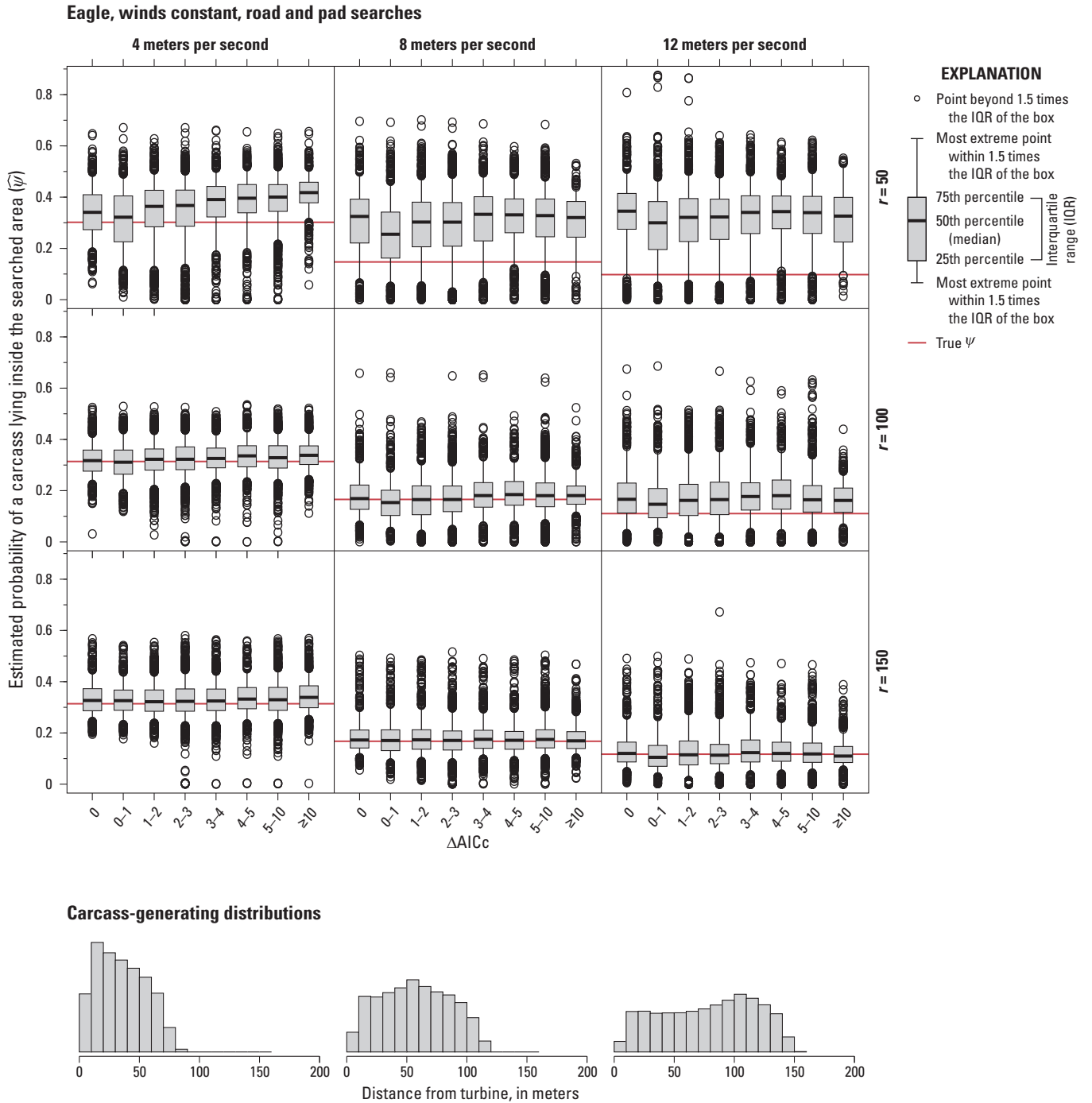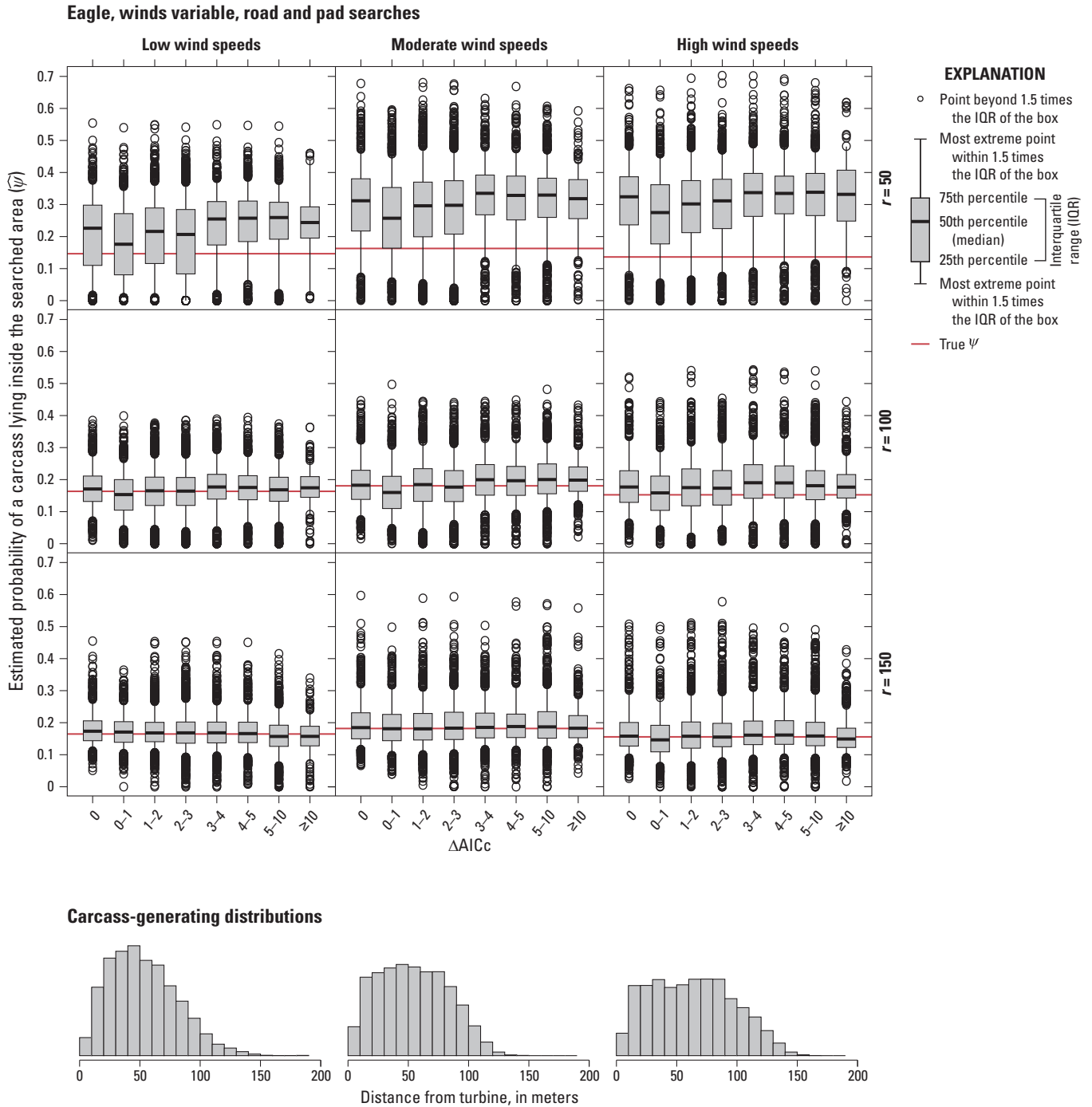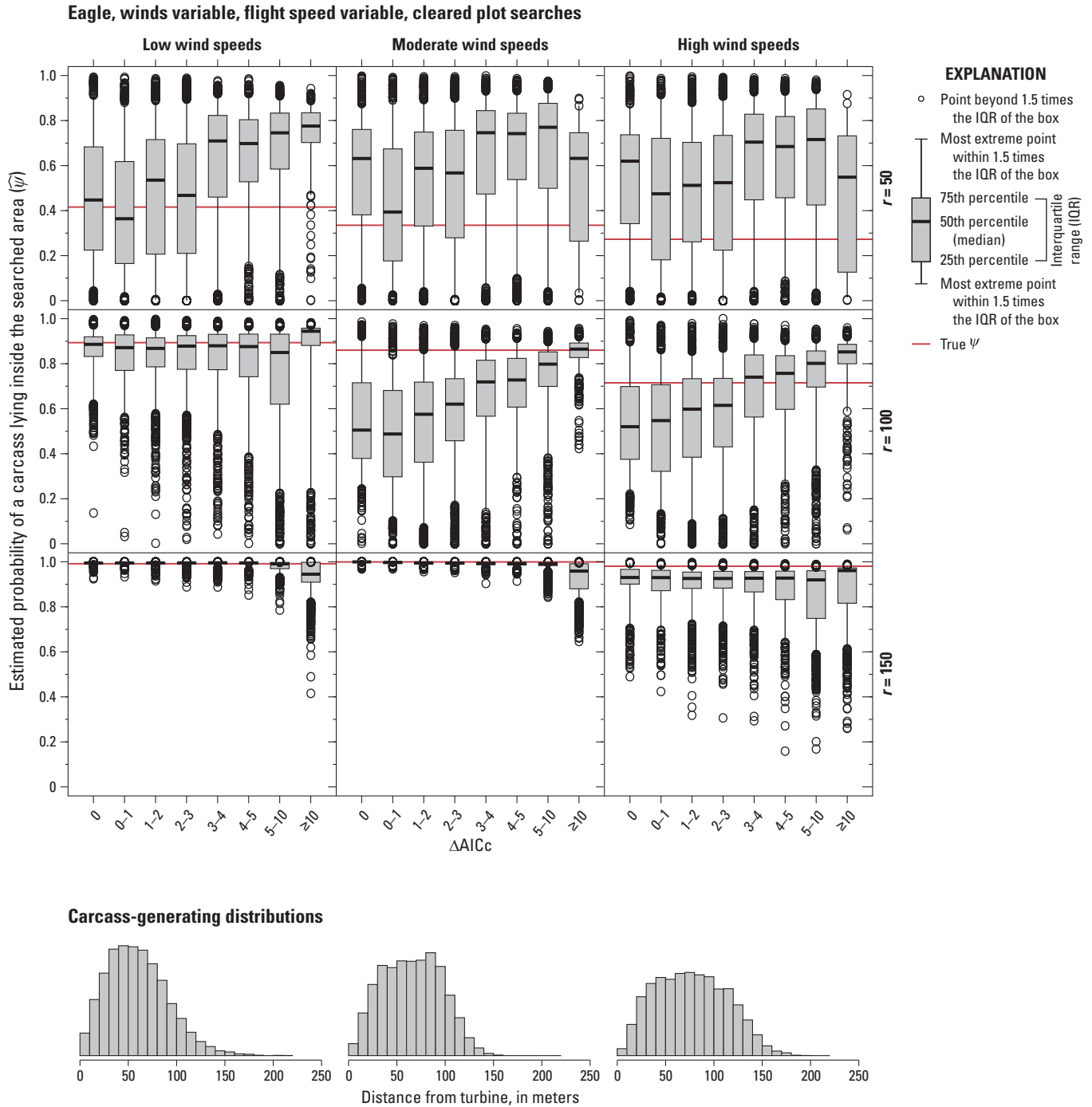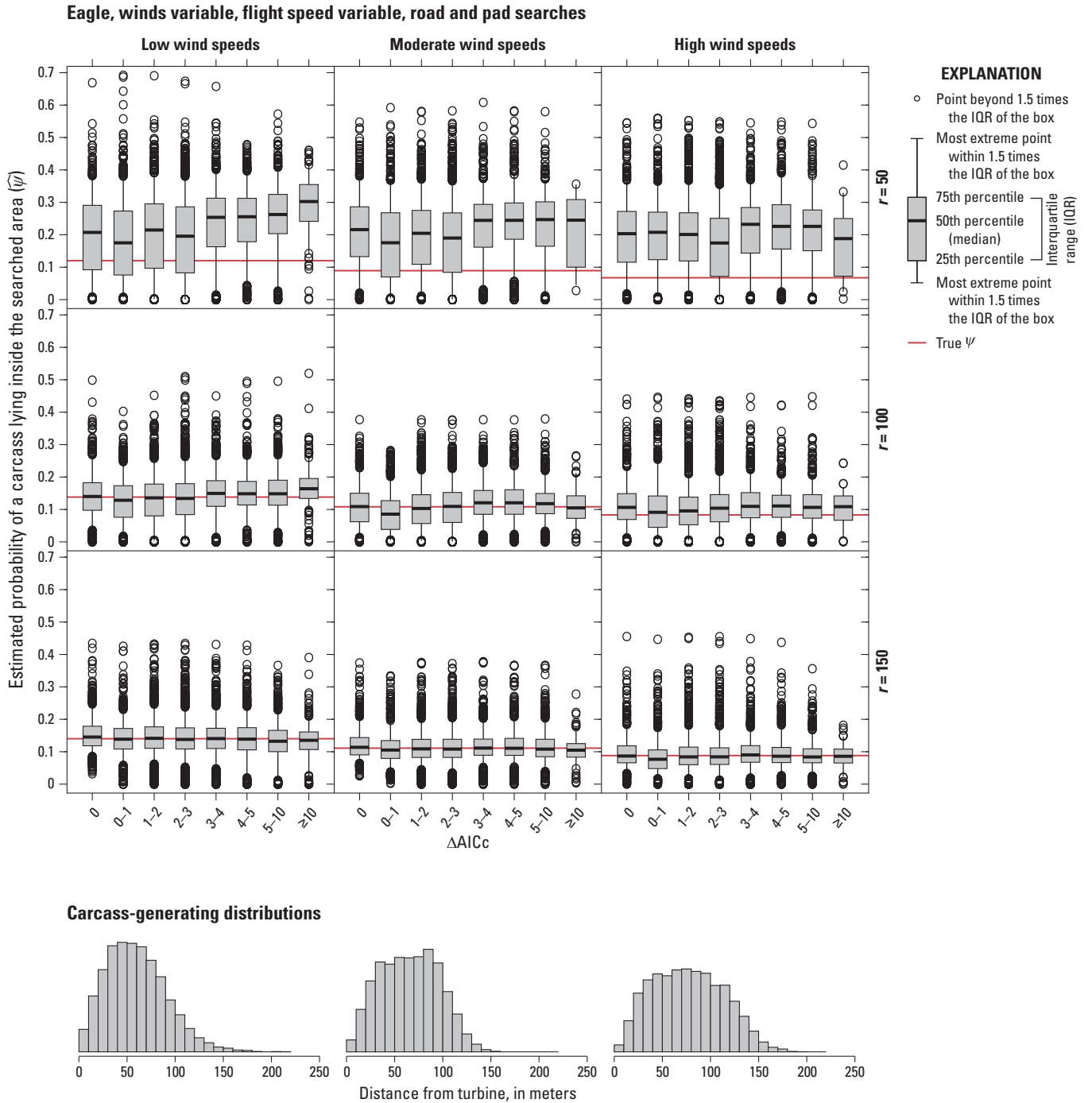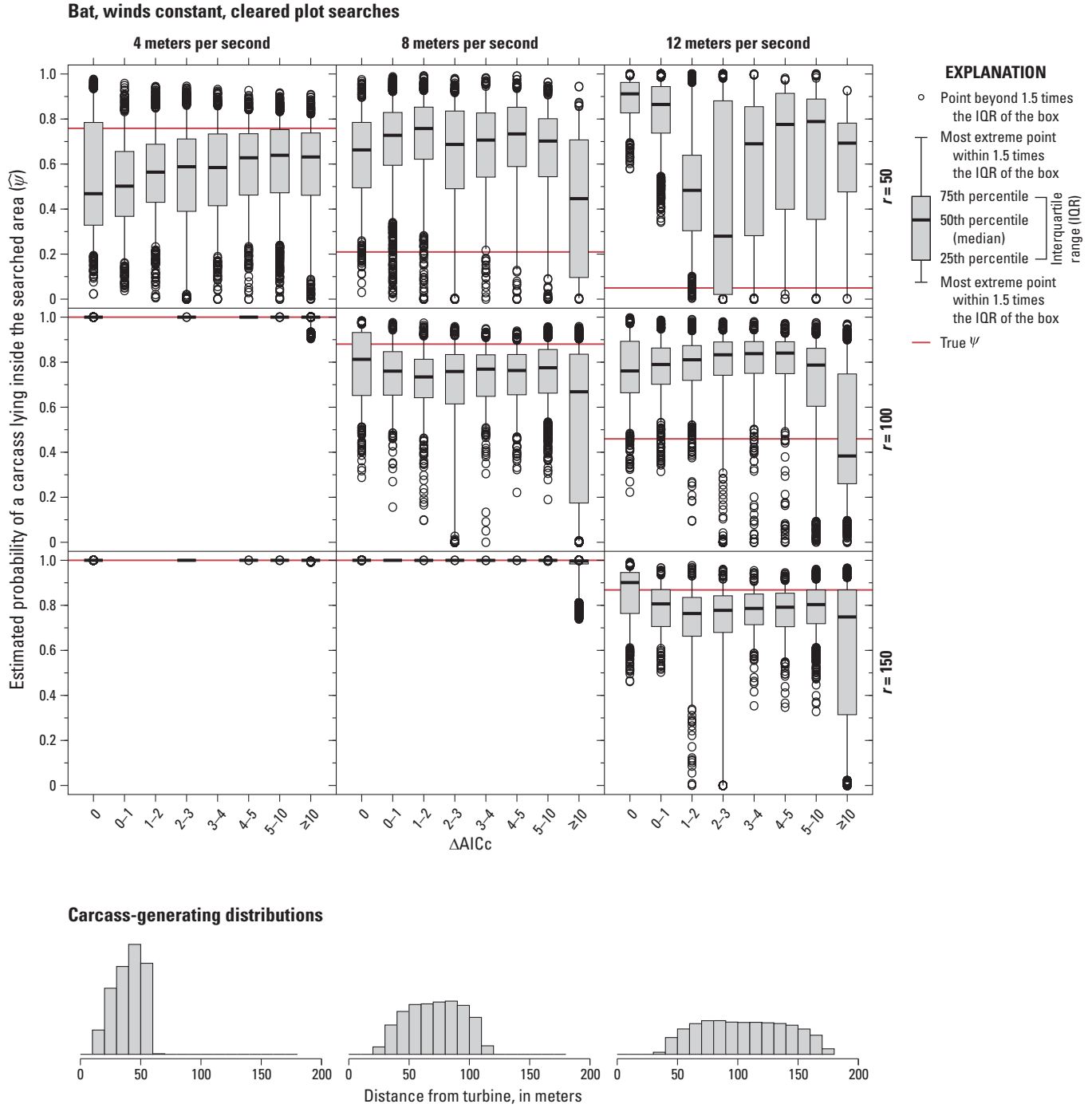
Figure 3.15.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated eagle fatalities with winds constant and road and pad searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.
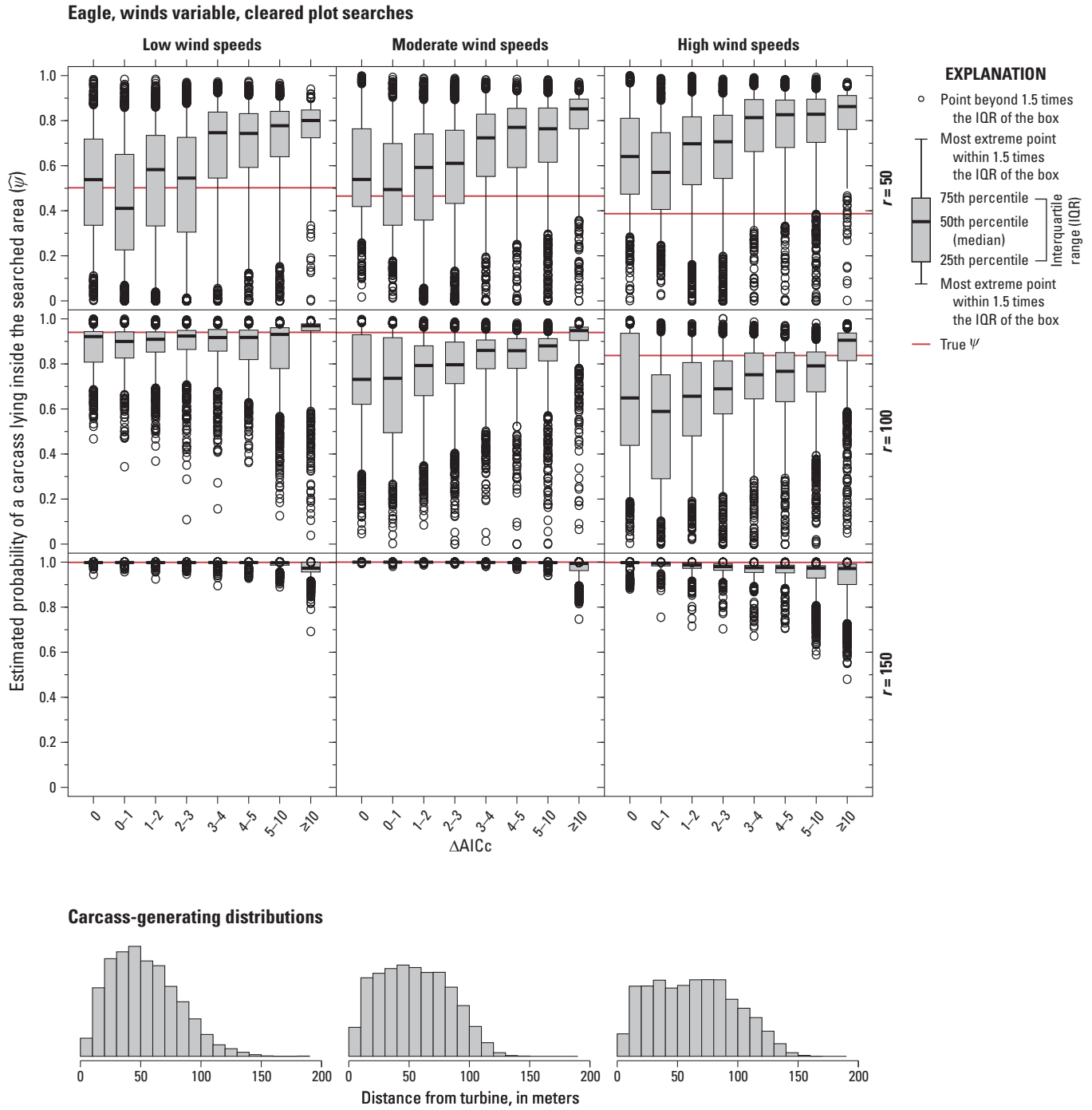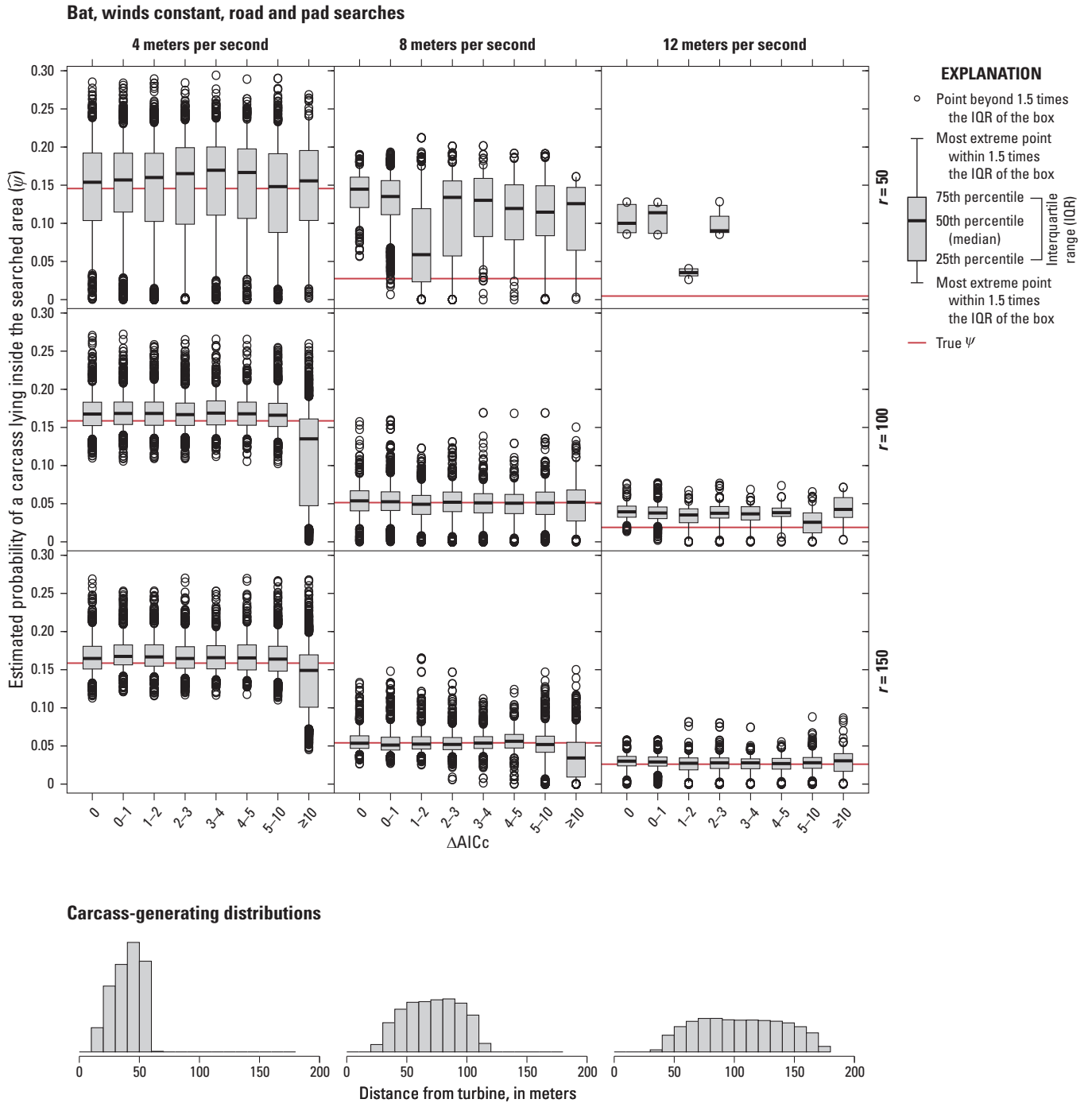
Figure 3.16.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated eagle fatalities with winds variable and cleared plot searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.
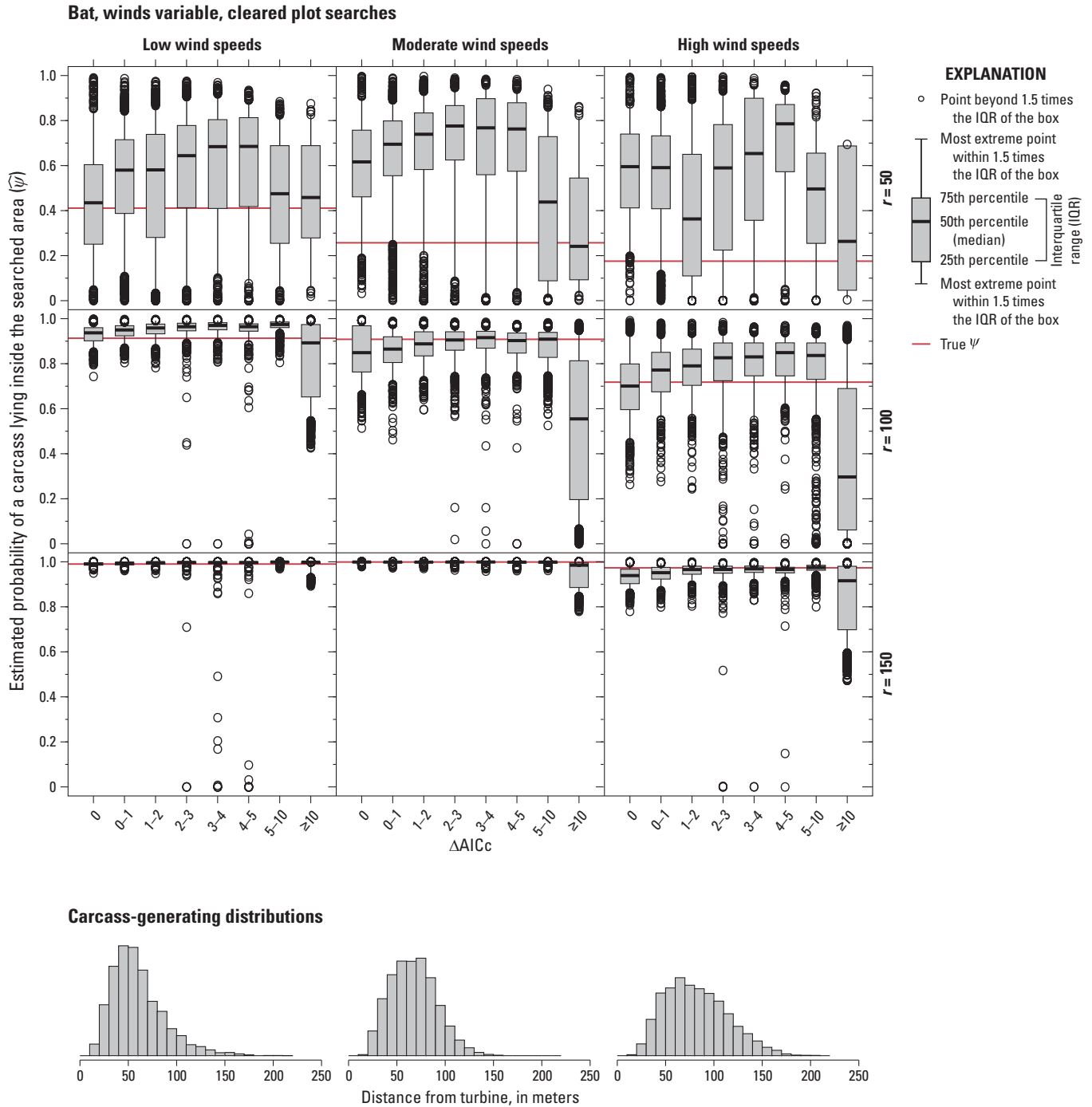
Figure 3.17. Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated eagle fatalities with winds variable and road and pad searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; r, search radius, in meters.

Figure 3.18.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated eagle fatalities with winds variable, flight speed variable, and cleared plot searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.

Eagle, winds variable, flight speed variable, road and pad searches
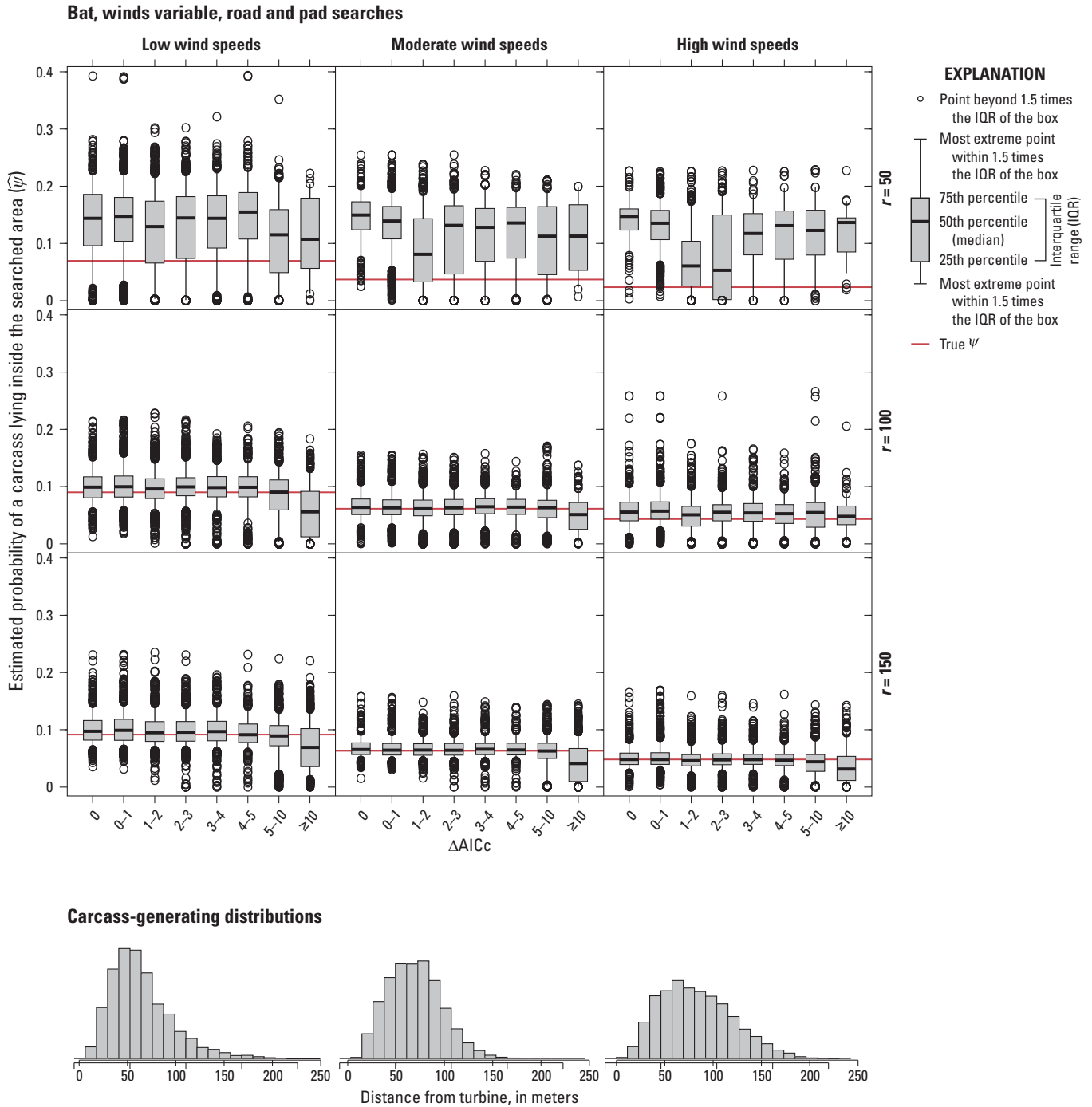

Figure 3.19.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated eagle fatalities with winds variable, flight speed variable, and road and pad searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.

Figure 3.20.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated bat fatalities with winds constant and cleared plot searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.

Bat, winds constant, road and pad searches
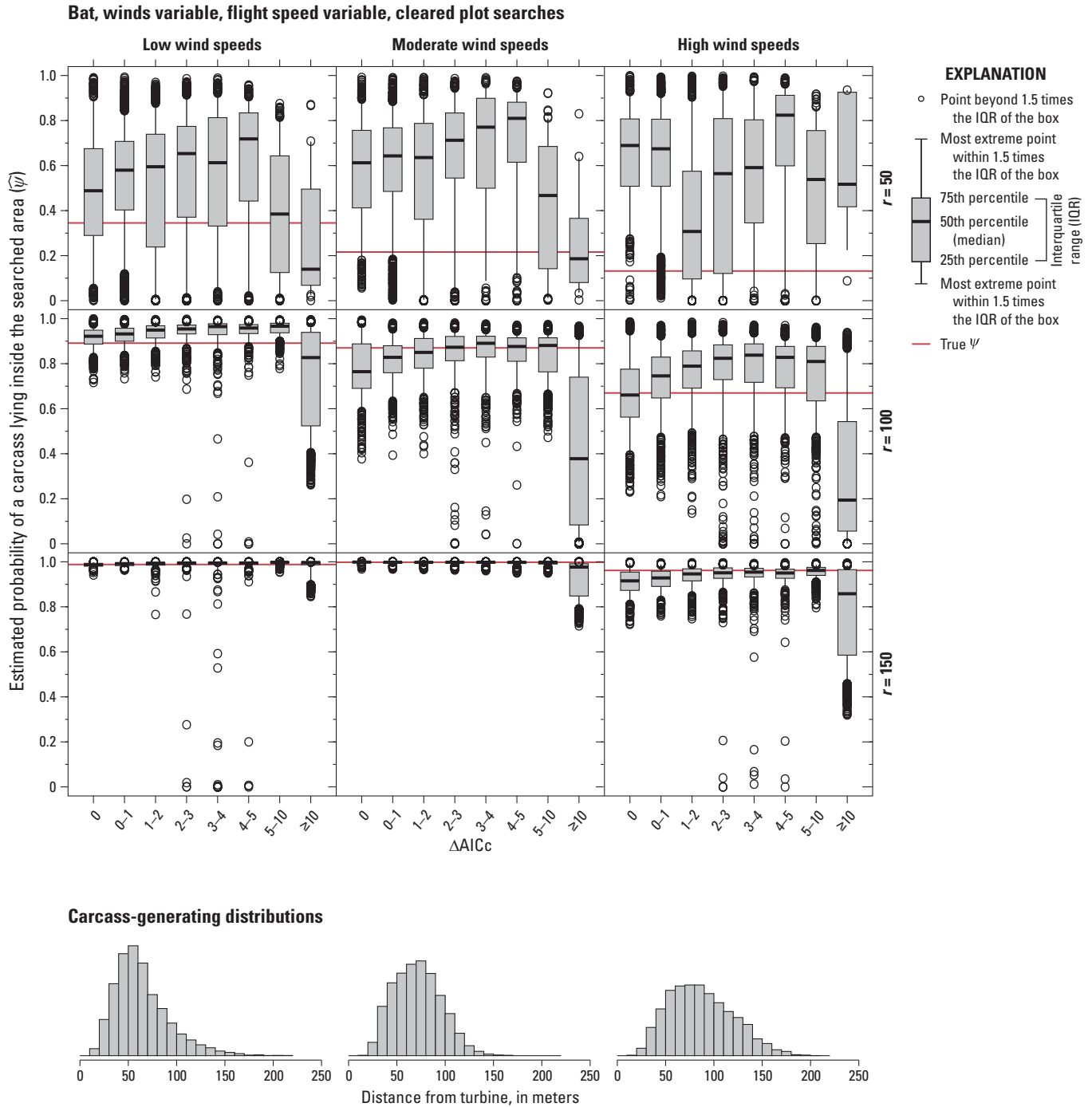

Figure 3.21.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated bat fatalities with winds constant and road and pad searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.
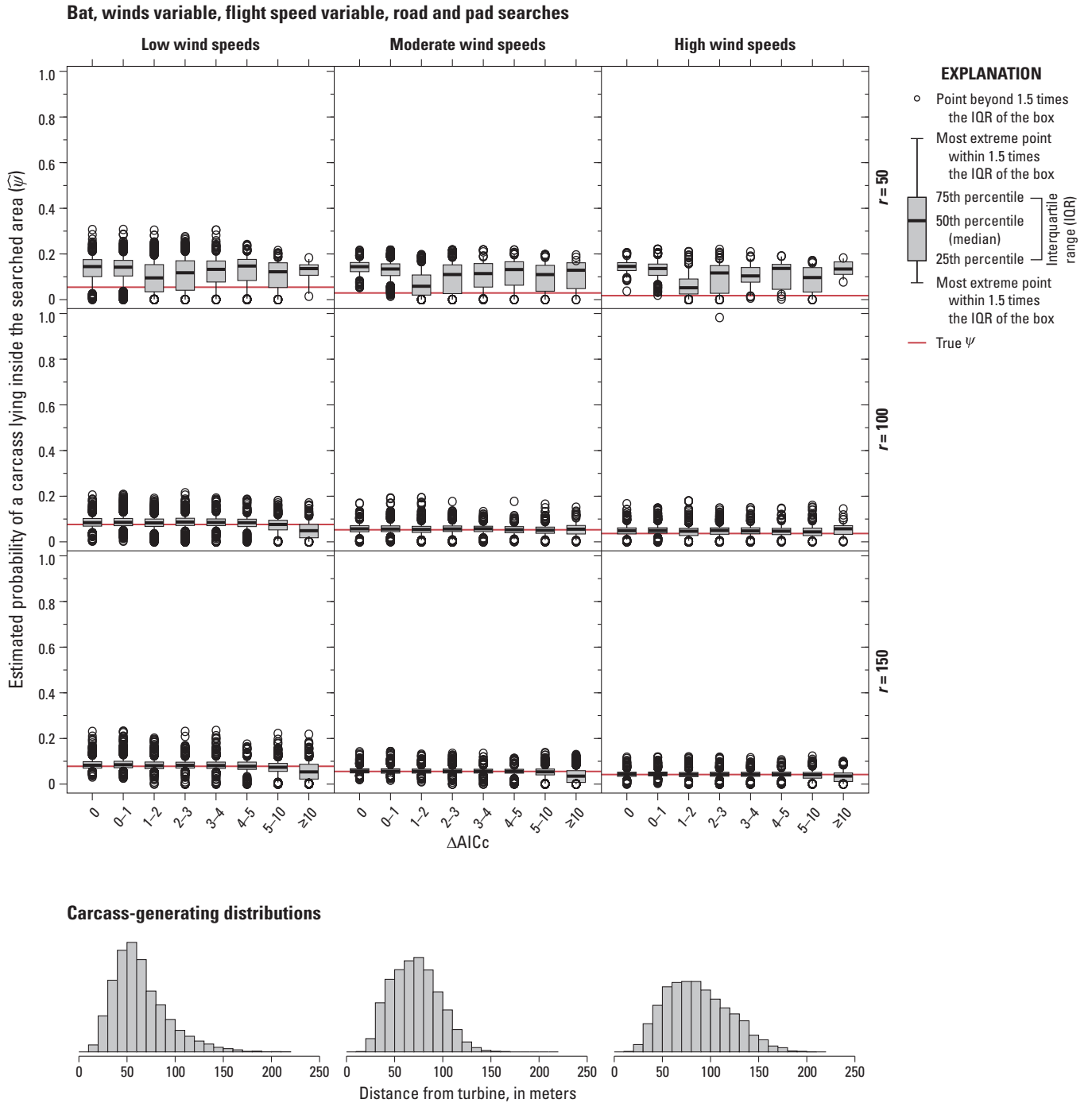
Figure 3.22. Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated bat fatalities with winds variable and cleared plot searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.

Bat, winds variable, road and pad searches

Figure 3.23. Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated bat fatalities with winds variable and roads and pads searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; r, search radius, in meters.

Figure 3.24.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated bat fatalities with winds variable, flight speed variable, and cleared plot searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; *r*, search radius, in meters.

Bat, winds variable, flight speed variable, road and pad searches

Figure 3.25.   Estimated probability of a carcass lying inside the searched area for the standard models binned by (ΔAICc), with 200 simulated bat fatalities with winds variable, flight speed variable, and roads and pads searches. AICc, Akaike information criterion corrected for finite sample size; ΔAICc, difference between AICc value for a given model and the lowest AICc value among the models tested; r, search radius, in meters.

Appendix 4.   Mortality Estimation and the Distinction Between the Probability of a Carcass Lying Inside the Searched Area and the Fraction of Carcasses Lying Inside that Area

Accurate estimation of the fraction of carcasses lying inside the searched area depends on both the probability that a carcass lies inside the searched area (ψ) and the number of carcasses that were found inside the searched area (X). For example, if ψ = 0.2 and X = 2 carcasses were found at turbine A, the most likely total number of carcasses would be M_A = 2/0.2 = 10 but M_A could readily be anywhere in its 90 percent credible interval[1]of [4,26], and the fraction of carcasses lying inside the searched area at the turbine (dwp_A) would likely be in the interval [2/26, 2/4] or 7.7–50 percent. By contrast, if ψ = 0.2 at turbine B too, but X_B = 20 rather than 2, then the credible interval for M_B would be [73, 109], and the fraction of carcasses lying inside the searched area at turbine B (dwp_A) would likely be in a much narrower interval—[20/139, 20/73] or 14.4–27.4 percent. The confidence interval for dwp properly reflects the greater relative information content of X = 20 compared to X = 2, but ψ does not.

To illustrate in another, more realistic way and validate the accuracy of the dwp estimator, we ran sets of 1,000 simulated collisions between bats and a turbine and the resulting dispersion of carcasses on the ground near the turbine. Carcass distances were generated according to a gamma distribution with 60 percent of the carcasses lying within 50 m of the turbine and 90 percent within 100 m (that is, a gamma distribution with `shape` = 1.7744 and `rate` = 0.0355). Carcass directions were uniformly distributed between 0° and 360°. The search area was confined to a fairly large turbine pad and a moderately wide access road, where the visibility was good and search conditions were easy. Carcasses were scavenged and removed from the field with persistence times randomly distributed as Weibull with `shape` = 0.64 and `scale` = 1.705, with median persistence time of 1 day and 90 percent of the carcasses removed within about 6 days. Searches were conducted on the road and pad area out to 150 m from the turbine every 5th day over a period spanning 150 days. Searcher efficiency for a carcass in the first search after arrival was 80 percent and $100 \times 0.8 \times 0.75^{k-1}$ percent on the kth search. Results for one set of 1,000 carcasses are shown in figure 4.1.

For each simulated dataset with 1,000 carcasses, `ddFit` from the dwp package was used to fit a gamma distribution to the dispersion of the recovered carcasses. Then, a 90-percent confidence interval for the probability that a carcass would lie in the searched area was calculated using `estpsi`, a 90-percent confidence interval for the fraction of carcasses lying in the searched area was calculated using `estdwp`, and the actual fraction of carcasses lying in the searched area was recorded. The process was repeated 1,000 times to create 1,000 confidence intervals for dwp calculated using `estdwp` and 1,000 for ψ using `estpsi`. Following this process, each simulated 90-percent confidence interval should have a 90 percent chance of including the actual fraction of carcasses in the searched area.

The proportion of intervals that do include (or "cover") the actual parameter (namely, the fraction of carcasses inside the searched area) is referred to as the coverage and is an essential test of estimator performance. Figure 4.2 displays 250 simulated confidence intervals for the fraction of carcasses lying in the searched area according to `estpsi` and `estdwp` estimators. The confidence intervals based on `estpsi` appear to be too small and routinely miss the true dwp. This is wholly expected because ψ is the probability of a carcass lying inside the searched area rather than the actual, realized fraction of carcasses lying inside the searched area and is inherently much less variable than dwp. Coverage for the nominal 90-percent confidence intervals from `estpsi` was only 26.8 percent. This is not a problem with `estpsi` or the calculation of $\hat{\psi}$; rather, it is a confusion of ψ as dwp and an attempt to use ψ to do dwp's job. By contrast, coverage for the nominal 90 percent confidence intervals based on `estdwp` was 90.5 percent, which is as close to 90 percent as can be expected for even a perfect estimator in such a small simulation.

---

[1]Credible interval is a Bayesian analog to the confidence interval of classical statistics. Credible intervals for M_A given X_A and Ψ can be calculated in the dwp package using, for example, MCI(postM(x = 2, g = 0.2)).
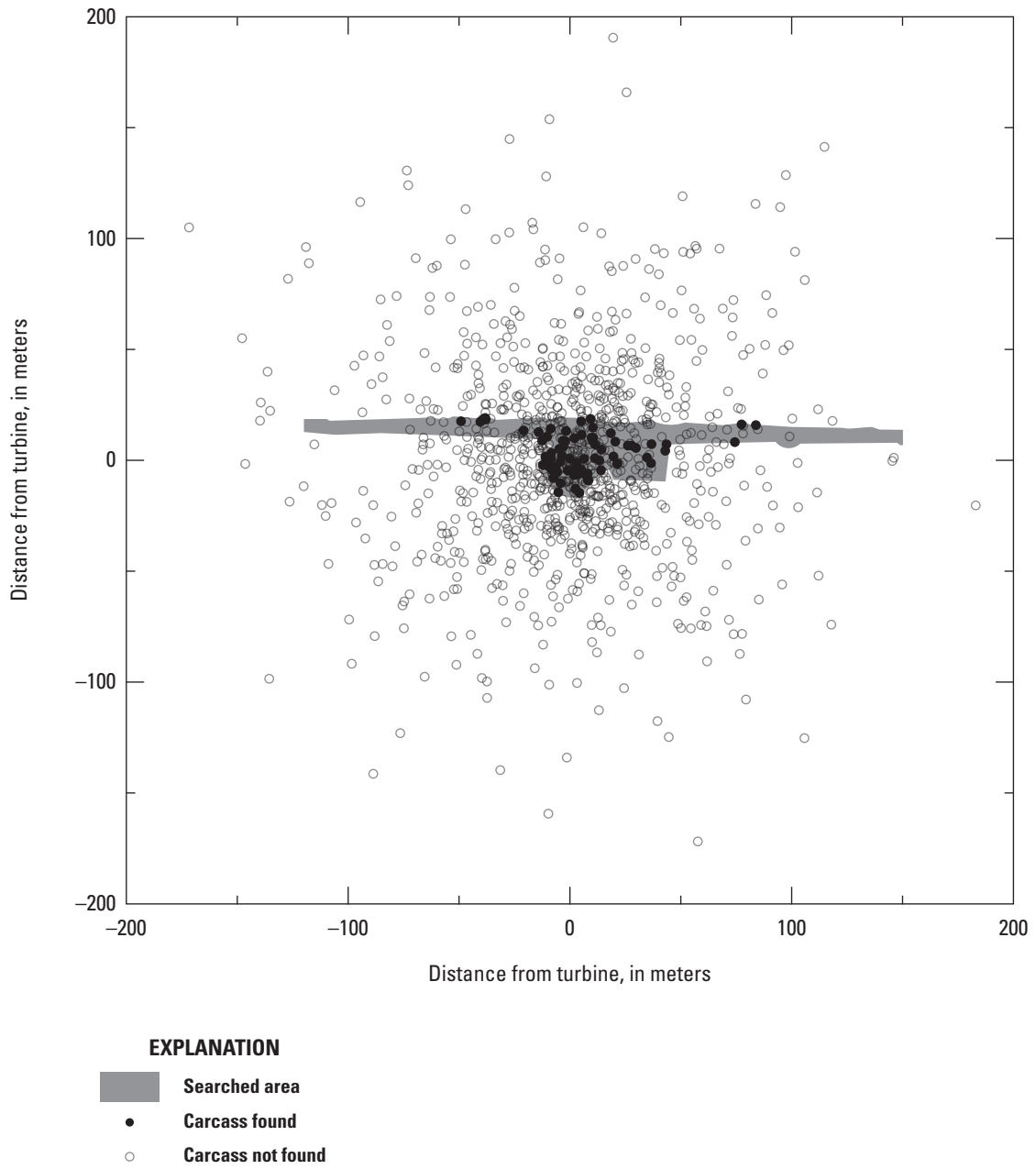
Figure 4.1. Simulated carcass dispersion and discovery for simulated dataset of 1,000 bat carcasses.
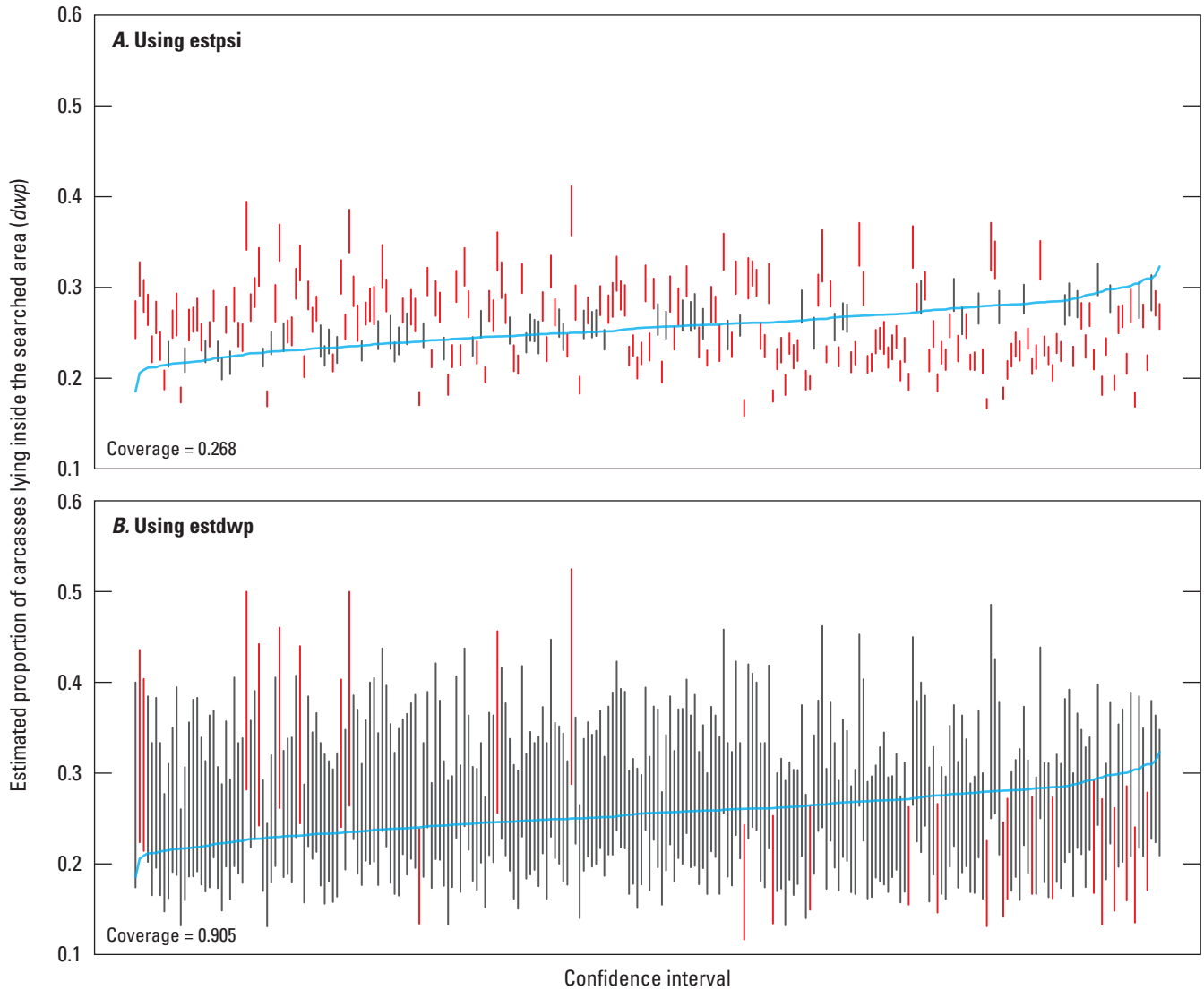
Figure 4.2.   The 250 confidence intervals (CIs) for *dwp*, the proportion of carcasses lying inside the searched area, using (*A*) `estpsi` or (*B*) `estdwp` from the dwp package as the estimator. The CIs are ordered by increasing target, actual *dwp*. Coverages are the proportion of 1,000 simulated CIs that cover the target *dwp*.

Appendix 5.   Accounting for the Uncertainty in Proportion of Carcasses Lying Inside the Searched Area—Technical Details

To be able to use the software effectively, it is not necessary to understand the details of how the dwp package accounts for the uncertainty in its estimate of proportion of carcasses lying inside the searched area (*dwp*). The details are offered here for completeness.

The uncertainty in estimated *dwp* ($\widehat{dwp}$) given the number of carcasses in the searched area (m_{in}) and the probability of a carcass lying in the searched area (ψ) is accounted for by sampling from the posterior distribution of the number of fatalities (M) given m_{in}, and ψ or m_{in}, more precisely,

$$\Pr(M = m | m_{in}, \psi) = \frac{\Pr(m_{in} | M = m, \psi)\Pr(M = m)}{\sum_m \Pr(m_{in} | M = m, \psi)\Pr(M = m)} \tag{5.1}$$

where
$m_{in} \sim$ binomial(M, ψ) and $\Pr(M)$ are the integrated reference prior, $M \propto \sqrt{(M+1)} - \sqrt{(M)}$ (Berger and others, 2012; Dalthorp and others, 2017).

In practice, ψ is not known but estimate using `ddFit` from the dwp package. The uncertainties in $\hat{\psi}$ and $M \mid (\psi, m_{in})$ are assessed through a three-step process:

1. Simulate the regression parameter estimates from the fitted generalized linear model as multivariate normals (Nelder and Wedderburn, 1972) to account for uncertainty in the fitted model (using `ddSim`, which is called by `estpsi` but can be called directly if desired);

2. Integrate the simulated models over the searched area (`estpsi`) to account for uncertainty in $\hat{\psi}$; and

3. Calculate a simulated $\widehat{dwp}_i = \dfrac{m_{in}}{M | (m_{in}, \hat{\psi})}$ for each simulated value of $\hat{\psi}$, in using a random draw from the posterior distribution of $M | (m_{in}, \hat{\psi})$.

The vector of estimated *dwp* values generated in this way incorporates the uncertainty in $M | (m_{in}, \hat{\psi})$, yielding accurate confidence intervals for *dwp*.

Proportion of Carcasses Lying Inside the Searched Area and the Estimation of the Number of Fatalities

After being properly formatted (using the function `exportGenEst`), the estimated *dwp* can be used with GenEst to estimate total mortality. At a site with a single search class with a single turbine, a simplified version of the model[1] for estimating M is $\widehat{M} = X / (\hat{g} \times \widehat{dwp})$, where X is the number of carcasses found, $\hat{g}$ is the estimated detection probability for carcasses lying in the searched area during the span of the monitoring field season, and $\widehat{dwp}$ is the estimated proportion of carcasses lying in the searched area.

Note that $\hat{M} = X / (\hat{g} \times \widehat{dwp}) = \dfrac{X / \hat{g}}{\widehat{dwp}} = \hat{m}_{in} / \widehat{dwp}$, which incorporates both the uncertainty in estimating the number of carcasses inside the searched area (m_{in}) and the uncertainty in the estimated proportion of carcasses lying in the searched area ($\widehat{dwp}$). Thus, $\widehat{M}$ accounts for the two major sources of uncertainty discussed previously, and in theory, it should give accurate confidence intervals, so that a 90-percent confidence interval should have a 90-percent chance of including M.

---

[1]The estimator for M that is used in GenEst is difficult to write down succinctly. Readers interested in finer detail are directed to the GenEst technical manual (Dalthorp, Madsen, and others, 2018).

To verify the accuracy, we ran a simulation of the processes of carcass deposition and discovery, field trials for estimating g, and distance modeling to estimate dwp. Then, we used GenEst to estimate mortality for each turbine and the total for the site. We tallied the number of 90 percent confidence intervals that included M, calculated the coverage as the fraction of the confidence intervals that included M, and compared the coverages with the target coverage of 90 percent. The parameters for the simulation scenarios:

Simulation:

1. Generate carcass distances for $M = 300$ or 1,000 carcasses as gamma($\alpha = 1.774, \beta = 28.17$).

2. Arrival at site with three turbines, with layouts as in figure 5.1.

3. Searches within squares with side = 240 meters on days = 0, 5, ..., 150.

4. Weibull persistence with parameters:

   - Easy visibility: $\alpha = 0.64, \beta = 1.705$

   - Moderate visibility: $\alpha = 0.64, \beta = 7.37$

   - Difficult visibility: $\alpha = 0.64, \beta = 9.47$

5. Searcher efficiency with parameters:

   - Easy visibility: $p = 0.8, k = 0.75$

   - Moderate visibility: $p = 0.3, k = 0.75$

   - Difficult visibility: $p = 0.15, k = 0.75$

6. Summary parameters

   - $\psi = 0.898, 0.254, 0.0967$ for turbines t1, t2, and t3, respectively.

   - $g = 0.283, 0.285, 0.179$ for search classes E, M, and D, respectively.

The estimated coverage probabilities range from 85 to 95 percent for all combinations of mortality ($M = 300, 1,000$) and field trial carcasses ($n = 10, 20, 100$) at all turbines and the site as a whole (fig. 5.2), with a single exception. Coverage for total mortality for the site as a whole when $n = 10$ and $M = 1,000$ was 84.8 percent, with 14.8 percent of the confidence intervals missing high (that is, $\widehat{M}_{lwr} > M$) and 1.8 percent missing low. This indicates a tendency to slightly overestimate the total under these conditions. With discrete random variables (M and n), coverage probabilities for 90 percent confidence intervals are guaranteed to differ from the nominal 90 percent except for in rare circumstances, so the small deviations from the target coverage is neither a surprise nor cause for alarm, especially when $n = 10$ (Madsen and others, 2020). When $M = 1,000$, the confidence intervals appear to be slightly too wide at turbine 3, as coverages approach 95 percent, with approximately equal probability of the confidence interval missing high or low (fig. 5.2).
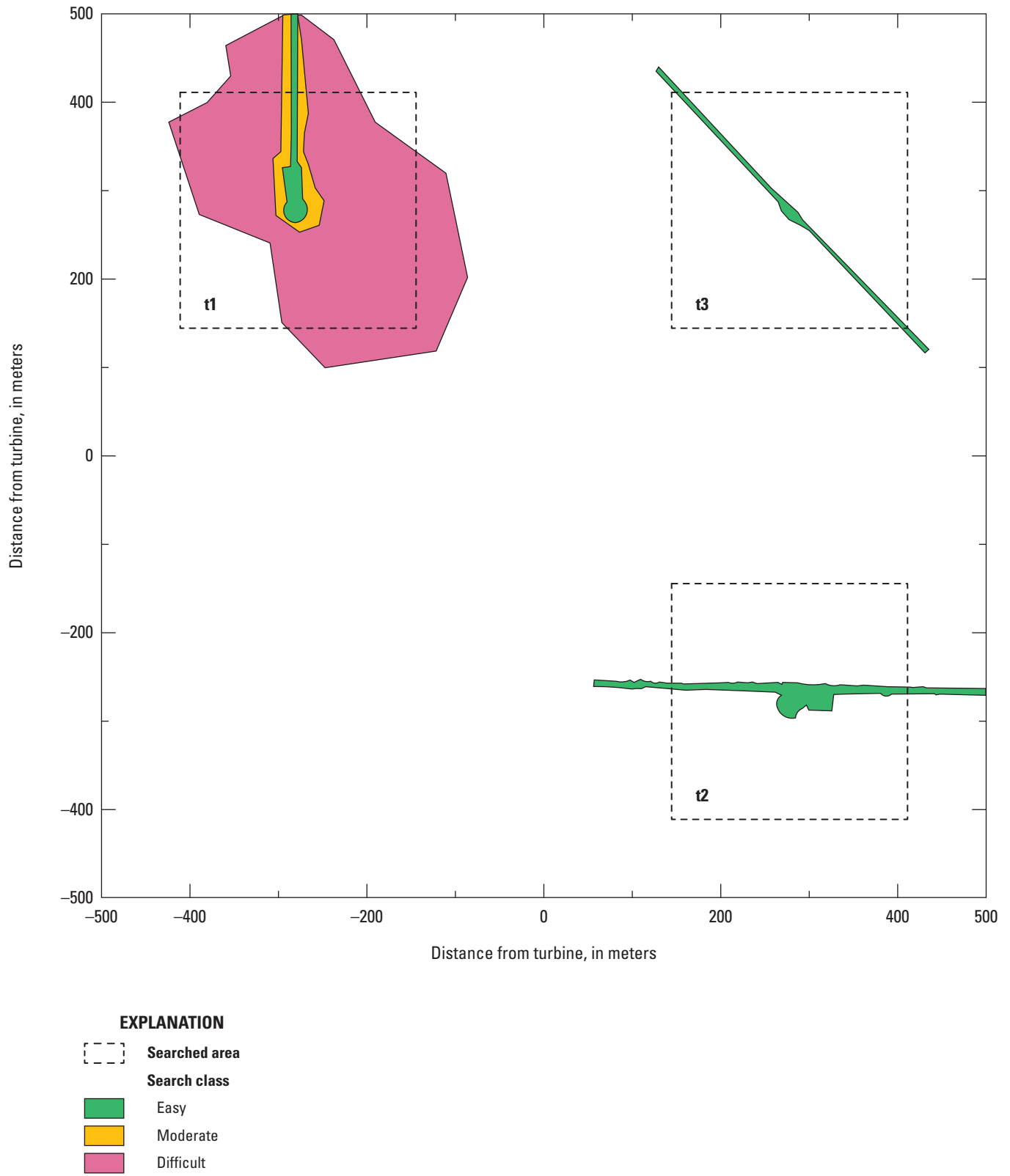
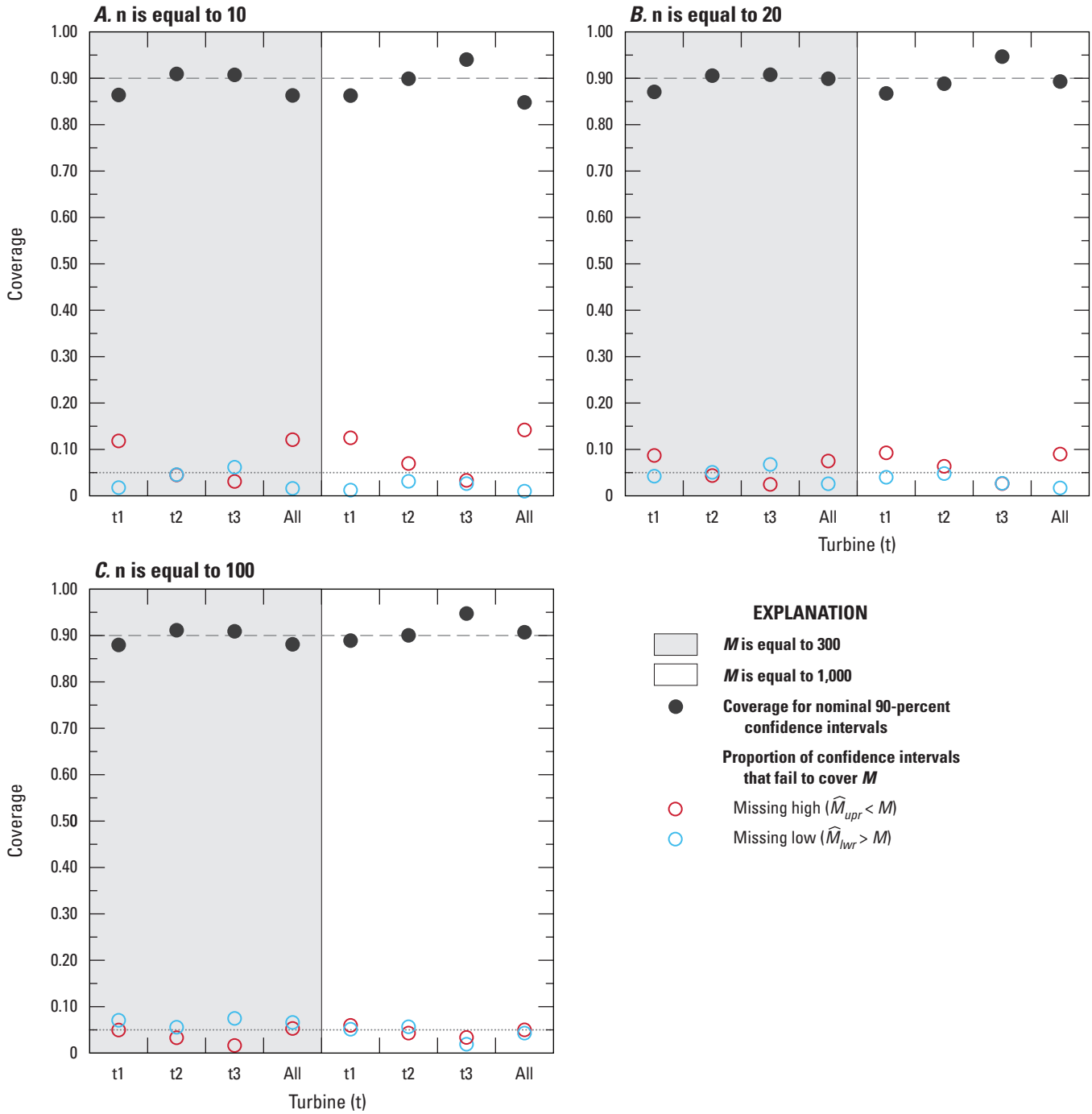**Figure 5.1.** Schematic showing search areas for three turbines.

Figure 5.2.   Coverages for nominal 90-percent confidence intervals for the three turbines at a site and for the total and the proportion of confidence intervals that fail to cover the number of fatalities (M) by missing high or missing low.

USGS