Spatial models for distance sampling data: recent developments and future directions

David L. Miller\textsuperscript{1*}, M. Louise Burt\textsuperscript{2}, Eric A. Rexstad\textsuperscript{2}, Len Thomas\textsuperscript{2}.

1. Department of Natural Resources Science, University of Rhode Island, Kingston, Rhode Island 02881, USA
2. Centre for Research into Ecological and Environmental Modelling, The Observatory, University of St. Andrews, St. Andrews KY16 9LZ, UK

*Correspondence author. dave@ninepointeightone.net

Date of submission: 18th July 2013
Summary

1. Our understanding of a biological population can be greatly enhanced by modelling their distribution in space and as a function of environmental covariates. Such models can be used to investigate the relationships between distribution and environmental covariates as well as reliably estimate abundances and create maps of animal/plant distribution.

2. Density surface models consist of a spatial model of the abundance of a biological population which has been corrected for uncertain detection via distance sampling methods.

3. We review recent developments in the field and consider the likely directions of future research before focussing on a popular approach based on generalized additive models. In particular, we consider spatial modelling techniques that may be advantageous to applied ecologists such as quantification of uncertainty in a two-stage model and smoothing in areas with complex boundaries.

4. The methods discussed are available in an R package developed by the authors (dsm) and are largely implemented in the popular Windows software Distance.

Keywords: abundance estimation, Distance software, generalized additive models, line transect sampling, point transect sampling, population density, spatial modelling, wildlife surveys
Introduction

When surveying biological populations it is increasingly common to record spatially referenced data, for example: coordinates of observations, habitat type, elevation or (if at sea) bathymetry. Spatial models allow for vast databases of spatially-referenced data (e.g. OBIS-SEAMAP, Halpin et al., 2009) to be harnessed, enabling investigation of interactions between environmental covariates and population densities. Mapping the spatial distribution of a population can be extremely useful, especially when communicating results to non-experts. Recent advances in both methodology and software have made spatial modelling readily available to the non-specialist (e.g., Wood, 2006; Rue et al., 2009). Here we use “spatial model” to refer to any model that includes any spatially referenced covariates, not only those models that include explicit location terms. This article is concerned with combining spatial modelling techniques with distance sampling (Buckland et al., 2001, 2004).

Distance sampling extends plot sampling to the case where detection is not certain. Observers move along lines or visit points and record the distance from the line or point to the object of interest \(y\). These distances are used to estimate the detection function, \(g(y)\) (for example, Fig. 1), by modelling the decrease in detectability with increasing distance from the line or point (conventional distance sampling, CDS). The detection function may also include covariates (multiple covariate distance sampling, MCDS; Marques et al., 2007) which affect the scale of the detection function. From the fitted detection function, the average probability of detection can be
estimated by integrating out distance. The estimated average probability
that an animal is detected given that it is in the area covered by the survey,
\( \hat{p}_i \), can then be used to estimate abundance as

\[
\hat{N} = \frac{A}{a} \sum_{i=1}^{n} s_i \hat{p}_i,
\]

where \( A \) is the area of the study region, \( a \) is the area covered by the survey
(i.e., the sum of the areas of all of the strips/circles) and the summation
takes place over the \( n \) observed clusters, each of size \( s_i \) (if individuals are
observed, \( s_i = 1 \forall i \)) (Buckland et al., 2001, Chapter 3). Often up to half
the observations in a plot sampling data set are discarded to ensure the
assumption of certain detection is met. In contrast, distance sampling uses
observations that would have been discarded to model detection (although
typically some detections are discarded beyond a given truncation distance
during analysis).

Estimators such as eqn (1) rely on the design of the study to ensure
that abundance estimates over the whole study area (scaling up from the
covered region) are valid. This article focusses on model-based inference
to extrapolate to a larger study area. Specifically, we consider the use of
spatially explicit models to investigate the response of biological populations
to biotic and abiotic covariates that vary over the study region. A spatially-
explicit model can explain the between-transect variation (which is often a
large component of the variance in design-based estimates) and so using a
model-based approach can lead to smaller variance in estimates of abundance
than design-based estimates. Model-based inference also enables the use of
data from opportunistic surveys, for example, incidental data arising from “ecotourism” cruises (Williams et al., 2006).

Our aims in creating a spatial model of a biological population are usually two-fold: (i) estimating overall abundance and (ii) investigating the relationship between abundance and environmental covariates. As with any predictions that are outside the range of the data, one should heed the usual warnings regarding extrapolation. For example, if a model contains elevation as a covariate, predictions at high, unsampled elevations are unlikely to be reliable. Frequently, maps of abundance or density are required and any spurious predictions can be visually assessed, as well as by plotting a histogram of the predicted values. A sensible definition of the region of interest avoids prediction outside the range of the data.

In this article we review the current state of spatial modelling of detection-corrected count data, illustrating some recent developments useful to applied ecologists. The methods discussed have been available in Distance software (Thomas et al., 2010) for some time but the recent advances covered here have been implemented in a new R package, dsm (Miller et al., 2013) and are to be incorporated into Distance.

Throughout this article a motivating data set is used to illustrate the methods. These data are sightings of pantropical spotted dolphins (Stenella attenuata) during April and May of 1996 in the Gulf of Mexico. Observers aboard the NOAA vessel Oregon II recorded sightings and environmental covariates (see http://seamap.env.duke.edu/dataset/25 for survey details). A complete example analysis is provided in Appendix A. The data used in the analysis are available in the dsm package and Distance.
The rest of the article reviews approaches for the spatial modelling of distance sampling data before focussing on the density surface modelling approach of Hedley & Buckland (2004) to estimate abundance and uncertainty. We then describe recent advances and provide practical advice regarding model fitting, formulation and checking. Finally we discuss future directions for research in spatially modelling detection-corrected count data.

Approaches to spatial modelling of distance sampling data

Modelling of spatially referenced distance sampling data is equivalent to modelling spatially-referenced count data, with the additional information provided by collecting distances to account for imperfect detection. We review recent efforts to model such data; some consist of two steps (correction for imperfect detection, then spatial modelling), while others jointly estimate the relevant parameters.

TWO-STAGE APPROACHES

The focus of this article is the “count model” of Hedley & Buckland (2004), we will henceforth refer to this approach as density surface modelling (DSM). Modelling proceeds in two steps: a detection function is fitted to the distance data to obtain detection probabilities for clusters (flocks, pods, etc.) or individuals. Counts are then summarised per segment (contiguous transect section). A generalised additive model (GAM; e.g. Wood, 2006) is then
constructed with the per-segment counts as the response with either counts or segment areas corrected for detectability (see *Density surface modelling*, below). GAMs provide a flexible class of models that include generalized linear models (GLMs; McCullagh & Nelder, 1989) but extend them with the possible addition of splines to create smooth functions of covariates, random effects terms or correlation structures. We cover advances using this approach in *Recent developments*.

As with the DSM approach, Niemi & Fernández (2010) used a two-step procedure: first fitting a detection function, then using a Bayesian point process to model spatial pattern (fitted using MCMC). Object density was described by an intensity function, which included spatially-referenced covariates. A possible disadvantage of their approach was that the distance function was assumed fixed once its parameters are estimated, and thus uncertainty may not be correctly propagated into final abundance estimates.

Ver Hoef *et al.* (2013) also included separate density and detection models for seals in the Bering sea. However, they were able to separate the detection process into three components: (i) incomplete detection on the transect line, (ii) declining detection probability as a function of distance, and (iii) availability bias (as seals could only be observed when hauled out on ice flows). After correcting counts for uncertain detection, they used a hierarchical, zero-inflated spatial regression model to estimate abundance, propagating variance associated with each stage of modelling into final estimates. The analysis shows that when extra information is available (such as telemetry data for the haul-out process) additional insight can be derived.

We note that there are many approaches to modelling spatially referenced
count data (Oppel et al., 2011, provides an overview of such methods for marine bird modelling). Also worthy of note is the approach of Barry & Welsh (2002) who used a two-stage approach to model presence/absence then spatial distribution (each via a separate GAM) to account for zero-inflation.

**ONE-Stage APPROACHES**

Rather than fitting two separate models, some authors have estimated parameters of the detection and spatial models simultaneously. Perhaps the first such example was Royle et al. (2004), who considered an integrated likelihood model for point and line transects. The approach views abundance as a nuisance variable which was integrated out of the likelihood, but inferences may still be made about factors affecting underlying density (including covariate effects). This approach was originally developed for binned distance data, but was extended by Chelgren et al. (2011) for continuous distance data.

Both Schmidt et al. (2011) and Conn et al. (2012) took data augmentation approaches to add unobserved clusters within their hierarchical Bayesian models. Schmidt et al. (2011) used a presence/absence-type model and a super-population approach (as in Royle & Dorazio, 2008). Conn et al. (2012) augmented observations only within the sampled transects using RJMCMC. Looking at the problem with at a coarser spatial resolution (stratum-level), Moore & Barlow (2011) separated the problem into observation and process components using a state-space model. The process component described the underlying population density as it changed over time and space, which was linked to the data via the detection function.
Another point process-based approach is that of Johnson et al. (2010), who used a Poisson process to model the locations of individuals in the survey area. Unlike Niemi & Fernández (2010), parameters of the intensity function were estimated jointly with detection function parameters via standard maximum likelihood methods for point processes (Baddeley & Turner, 2000) (allowing uncertainty from both the spatial pattern and detection function to be included in variance estimates). A post-hoc correction factor was used to address overdispersion unmodelled by spatial covariates (i.e. counts that do not follow a Poisson mean-variance relationship).

**ONE- VS. TWO-STAGE APPROACHES**

Generally very little information is lost by taking a two-stage approach. This is because transects are typically very narrow compared with the width of the study area so, provided no significant density variation takes place “across” the width of the lines or within the point, there is no information in the distances about the spatial distribution of animals (this is an assumption of two-stage approaches).

Two-stage approaches are effectively “divide and conquer” techniques: concentrating on the detection function first, and then, given the detection function, fitting the spatial model. One-stage models are more difficult to both estimate and check as both steps occur at once; models are potentially simpler from the perspective of the user and perhaps more mathematically elegant.

Two-stage models have the disadvantage that to accurately quantify model uncertainty one must appropriately combine uncertainty from the detection
function and spatial models. This can be challenging; however, the alternative of ignoring uncertainty from the detection process (e.g., Niemi & Fernández, 2010) can produce confidence or credible intervals for abundance estimates that have coverage below the nominal level. More information regarding how variance estimation is addressed for DSMs is given in Recent developments.

Density surface modelling

This section focuses on modelling the density/abundance estimation stage of the DSM approach introduced previously. Both line and point transects can be used, but if lines are used then they are are split into contiguous segments (indexed by $j$), which are of length $l_j$. Segments should be small enough such that neither density of objects nor covariate values vary appreciably within a segment (making the segments approximately square is usually sufficient; $2w \times 2w$, where $w$ is the truncation distance). The area of each segment enters the model as (or as part of) an offset: the area of segment $j$ is $A_j = 2wl_j$ and for point $j$ is $A_j = \pi w^2$.

Count or estimated abundance (per segment or point) is then modelled as a sum of smooth functions of covariates ($z_{jk}$ with $k$ indexing the covariates, e.g., location, sea surface temperature, weather conditions; measured at the segment/point level) using a generalized additive model. Smooth functions are modelled as splines, providing flexible unidimensional (and higher-dimensional) curves (and surfaces, etc) that describe the relationship between the covariates and response. Wood (2006) and Ruppert et al. (2003) provide
more in-depth introductions to smoothing and generalized additive models.

We begin by describing a formulation where only covariates measured
per-segment (e.g. habitat, Beaufort sea state) are included in the detection
function. We later expand this simple formulation to include observation
level covariates (e.g., cluster size, species)

COUNT AS RESPONSE

The model for the count per segment is:

\[ E(n_j) = \hat{p}_j A_j \exp \left( \beta_0 + \sum_k f_k(z_{jk}) \right), \]

where the \( f_k \)'s are smooth functions of the covariates and \( \beta_0 \) is an intercept
term. Multiplying the segment area (\( A_j \)) by the probability of detection (\( \hat{p}_j \))
gives the effective area for segment \( j \). If there are no covariates other than
distance in the detection function then the probability of detection is constant
for all segments (i.e., \( \hat{p}_j = \hat{p}, \forall j \)). The distribution of \( n_j \) can be modelled
as an overdispersed Poisson, negative binomial, or Tweedie distribution (see
Recent developments).

Fig. 2 shows the raw observations of the dolphin data, along with the
transect lines, overlaid on the depth data. A half-normal detection function
was fitted to the distances and is shown in Fig. 1. Fig. 3 shows a DSM fitted
to the dolphin data. The top panel shows predictions from a model where
depth was the only covariate, the bottom panel shows predictions where
a (bivariate) smooth of spatial location was also included. Comparing the
models using GCV score, the latter had a considerably lower score (39.12 vs
As well as simply calculating abundance estimates, relationships between covariates and abundance can be illustrated via plots of marginal smooths. The effect of depth on abundance (on the scale of the link function) for the dolphin data can be seen in Fig. 4.

An alternative to modelling counts is to use the per-segment/circle abundance using distance sampling estimates as the response. In this case we replace \( n_j \) by:

\[
\hat{N}_j = \sum_{r=1}^{R_j} \frac{s_{jr}}{p_j},
\]

where \( R_j \) is the number observations in segment \( j \) and \( s_{jr} \) is the size of the \( r^{th} \) cluster in segment \( j \) (if the animals occur individually then \( s_{jr} = 1, \forall j, r \)).

The following model is then fitted:

\[
E(\hat{N}_j) = A_j \exp \left[ \beta_0 + \sum_{k} f_k (z_{jk}) \right],
\]

where \( \hat{N}_j \), as with \( n_j \), is assumed to follow an overdispersed Poisson, negative binomial, or Tweedie distribution (see Recent developments, below). Note that the offset (\( A_j \)) is now the area of segment/point rather than effective area of the segment/point. Although \( \hat{N}_j \) can always be modelled instead of \( n_j \), it seems preferable to use \( n_j \) when possible, as one is then modelling actual (integer) counts as the response rather than estimates. Note that although \( \hat{N}_j \) may take non-integer values, this does not present an estimation problem for the response distributions covered here.
DSM with covariates at the observation level

The above models consider the case where the covariates are measured at the segment/point level. Often covariates \((z_{ij}, \text{for individual/cluster } i \text{ and segment/point } j)\) are collected on the level of observations; for example sex or cluster size of the observed object or identity of the observer. In this case the probability of detection is a function of the object (individual or cluster) level covariates \(\hat{p}(z_i)\). Object level covariates can be incorporated into the model by adopting the following estimator of the per-segment/point abundance:

\[
\hat{N}_j = \sum_{r=1}^{R_j} \frac{s_{jr}}{\hat{p}(z_{rj})}.
\]

Density, rather than abundance, can be modelled by excluding the offset and instead dividing the count (or estimated abundance) by the area of the segment/point (and weighting observations by the segment/point areas). We concentrate on abundance here; see Hedley & Buckland (2004) for further details on modelling density.

Prediction

A DSM can be used to predict abundance over a larger/different area than was originally surveyed. In that case the investigator must create a series of prediction cells over the prediction region. For each cell the covariates included in the DSM must be available; the area of each cell is also required. Having made predictions for each cell, these can be plotted as an abundance map (as in Fig. 3) and, by summing over cells, an overall estimate of abund-
ance can be calculated. It is worth noting that using prediction grid cells that are smaller than the resolution of the spatially referenced data has no effect on abundance/density estimates.

Variance estimation

Estimating the variance of abundances calculated using a DSM is not straightforward: uncertainty from the estimated parameters of the detection function must be incorporated into the spatial model. A second consideration is that in a line transect survey, abundances in adjacent segments are likely to be correlated; failure to account for this spatial autocorrelation will lead to artificially low variance estimates and hence misleadingly narrow confidence intervals.

Hedley & Buckland (2004) describe a method of calculating the variance in the abundance estimates using a parametric bootstrap, resampling from the residuals of the fitted model. The bootstrap procedure is as follows.

Denote the fitted values for the model to be $\hat{\eta}$. For $b = 1, \ldots, B$ (where $B$ is the number of resamples required).

1. Resample (with replacement) the per-segment/point residuals, store the values in $r_b$.

2. Refit the model but with the response set to $\hat{\eta} + r_b$ (where $\hat{\eta}$ are the fitted values from the original model).

3. Take the predicted values for the new model and store them.

From the predicted values stored in the last step the variance originating in the spatial part of the model can be calculated. The total variance of the
abundance estimate (over the whole region of interest or sub-areas) can then be found by combining the variance estimate from the bootstrap procedure with the variance of the probability of detection from the detection function model using the delta method (which assumes that the two components of the variance are independent; Ver Hoef, 2012).

The above procedure assumes that there is no correlation in space between segments, which are usually contiguous along transects. If many animals are observed in a particular segment then we might expect there to be high numbers in the adjacent segments. A moving block bootstrap (MBB; Efron & Tibshirani, 1993, Section 8.6) can account for some of this spatial autocorrelation in the variance estimation. The segments are grouped together into overlapping blocks (so if the block size is 5, block one is segments 1,...,5, block two is segments 2,...,6, and so on). Then, at step (2) above, resamples are taken at the block level (rather than individual segments within a transect). Using MMB will account for correlation between the segments at scales smaller than the block size, inflating the variances accordingly. Block size can be selected by plotting an autocorrelogram of the residuals from the DSM.

Both bootstrap procedures can also be modified to take detection function uncertainty into account. Distances are simulated from the fitted detection function and then the offset is re-calculated by fitting a detection function to the simulated distances.

Uncertainty can be estimated for a given prediction region by calculating the appropriate quantiles of the resulting abundance estimates (outlier removal may be required before quantile calculation). DSM uncertainty can
be visualised via a plot of per-cell coefficient of variation obtained by dividing
the standard error for each cell by its predicted abundance (as in Fig. 5).

Recent developments

**GAM uncertainty and variance propagation**

Rather than using a bootstrap, one can use GAM theory to construct un-
certainty estimates for DSM abundance estimates. This requires that we use
the distribution of the parameters in the GAM to simulate model coefficients,
using them to generate replicate abundance estimates (further information
found in Wood, 2006, page 245). Such an approach removes the need to
refit the model many times, making variance estimation much faster.

Williams *et al.* (2011) go a step further and incorporate the uncertainty in
the estimation of the detection function into the variance of the spatial model,
albeit only when segment level covariates are in the DSM. Their procedure
is to fit the density surface model with an additional random effect term
that characterises the uncertainty in the estimation of the detection function
(via the derivatives of the probability of detection, $\hat{p}$, with respect to their
parameters). Variance estimates of the abundance calculated using standard
GAM theory will include uncertainty from the estimation of the detection
function. A more complete mathematical explanation of this result is given
in Appendix B.

We consider that propagating the uncertainty in this manner to be prefer-
able to the MBB because it is more computationally efficient meaning invest-
igators can easily and quickly estimate variances of complex models. The
confidence intervals produced via variance propagation appear comparable
(if not narrower) than their bootstrap equivalents, while maintaining good
coverage (results of a small simulation study are given in Appendix C).

Fig. 5 shows a map of the coefficient of variation for the model which
includes both location and depth covariates. Variance has been calculated
using the variance propagation method.

Edge effects

Previous work (Ramsay, 2002; Wang & Ranalli, 2007; Wood et al., 2008;
Scott-Hayward et al., 2013; Miller & Wood, submitted) has highlighted the
need to take care when smoothing over areas with complicated boundaries,
e.g., those with rivers, peninsulae or islands. If two parts of the study area
(either side of a river or inlet, say) are inappropriately linked by the model
(i.e. if the distance between the points is measured as a straight line, rather
than taking into account obstacles) then the boundary feature (river, etc)
can be “smoothed across” so positive abundances are predicted in areas where
animals could not possibly occur. Ensuring that a realistic spatial model has
been fitted to the data is essential for valid inference. The soap film smoother
of Wood et al. (2008) is an appealing solution: a bivariate smooth function
of location that can be included in any GAM but that allows for boundary
conditions to be estimated and obeyed for a complex study area. Such an
approach can be helpful when uncertainty is estimated via a bootstrap as
edge effects can also cause large, unrealistic predictions which can plague
other smoothers (Bravington & Hedley, 2009).

Even if the study area does not have a complicated boundary, edge effects
can still be problematic. Miller (2012) notes that some smoothers have plane
components that tend to cause the fitted surface to increase unrealistically as
predictions are made further away from the locations of survey effort. This
problem can be alleviated by the using a different type of smoother (e.g. a
generalisation of thin plate regression splines called Duchon splines).

**Tweedie distribution**

The Tweedie distribution offers a flexible alternative to the quasi-Poisson and
negative binomial distributions as a response distribution when modelling
count data (Candy, 2004). In particular it is useful when there are a high
proportion of zeros in the data (Shono, 2008; Peel *et al.*, 2012) and avoids
multiple-stage modelling of zero-inflated data (as in Barry & Welsh, 2002).

The distribution has three parameters parameters: a mean, dispersion
and a third power parameter, which leads to additional flexibility. The dis-
tribution does not change appreciably when the power parameter is changed
by less than 0.1 and therefore a simple line search over the possible values
for the power parameter is usually a reasonable approach to estimating the
parameter. Mark Bravington (pers. comm.) suggested plotting the square
root of the absolute value of the residuals against fitted values; a “flatter”
plot (points forming a horizontal line) give an indication of a “good” value.

We additionally suggest using the metrics described in the next section for
model selection.

Appendix D gives further details about the Tweedie distribution (includ-
ing its probability density function and further references).
Practical advice

A flow diagram of the modelling process for creating a DSM is shown in Fig. 6. The diagram shows which methods are compatible with each other and what the options are for modelling a particular data set.

In our experience, it is sensible to obtain a detection function that fits the data as well as possible and only begin spatial modelling after a satisfactory detection function has been obtained. Model selection for the detection function can be performed using AIC and model checking using goodness-of-fit tests given in Burnham et al. (2004, Section 11.11). If animals occur in clusters rather than individually, bias can be incurred due to the higher visibility of larger clusters. It may then be necessary to include size as a covariate in the detection function (see Buckland et al., 2001, Section 4.8.2.4). For some species cluster size may change according to location, Ferguson et al. (2006) use two GAMs (one to model observed clusters and one to model the cluster size) to deal with spatially-varying cluster size amongst delphinids, though the authors do not present the variance of the resulting predictions.

Smooth terms can be selected using (approximate) p-values (Wood, 2006, Section 4.8.5). An additional useful technique for covariate selection is to use an extra penalty for each term in the GAM allowing smooth terms to be removed from the model during fitting (illustrated in Appendix A; Wood, 2011). Smoothness selection is performed by generalized cross validation (GCV) score, unbiased risk estimator (UBRE) or restricted maximum likelihood (REML) score. When model covariates are effectively functions of one another (e.g. depth could be written as a function of location) GCV and...
UBRE can suffer from optimisation problems (Wood, 2006, Section 4.5.3) which can lead to unstable models (Wood, 2011). REML provides a fitting criteria with a more pronounced optima which avoids some problems with parameter estimation, though caution should always be taken when dealing with highly correlated covariates. A significant drawback of REML is that scores cannot be used to compare models with different linear terms or offsets (Wood, 2011), though the $p$-value and additional penalty techniques described above can be used to select model terms. We highly recommend the use of standard GAM diagnostic plots; Wood (2006) provides further practical information on GAM model selection and fitting.

In the analysis of the dolphin data we included a smooth of location that nearly doubles the percentage deviance explained (27.3% to 52.7%). One can see this when comparing the two plots in Fig. 3 and the plot of the depth (Fig. 2), the plot of the model containing only a smooth of depth looks very similar to the raw plot of the depth data. Using a smooth of location can be a primitive way to account for spatial autocorrelation and/or as a proxy for other spatially varying covariates that are unavailable.

A more sophisticated way to account for spatial autocorrelation between segments (within transects) is to use an autocorrelation structure within the DSM (e.g. autoregressive models). Appendix A shows an example using generalized additive mixed model (GAMMs; Wood, 2006, Section 6.6, see Appendix A for an example) to construct an autoregressive (lag 1) correlation structure. This gives a significant reduction in variance, tightening the confidence interval around the abundance estimate.

In the analysis presented here, spatial location has been transformed from
latitude and longitude to kilometres north and east of the centre of the survey region at \((27.01^\circ, -88.3^\circ)\). This is because the bivariate smoother used (the thin plate spline; Wood, 2003) is isotropic: there is only one parameter controlling the smoothness in both directions. Moving one degree in latitude is not the same as moving one degree in longitude and so using kilometres from the centre of the study region makes the covariates isotropic. Using metric units rather than non-standard units of measure such as degrees or feet throughout makes analysis much easier.

A smooth of an environment-level covariate such as depth can be very useful for assessing the relationships between abundance and the covariate (as in Fig. 4). Caution should be employed when interpreting smooth relationships and abundance estimates, especially if there are gaps over the range of covariate values. Large counts may occur at large values of depth but if no further observations occur at such a large value, then investigators should be skeptical of any relationship.

**Discussion**

The use of model-based inference for determining abundance and spatial distribution from distance sampling data presents new opportunities in the field of population assessment. Spatial models can be particularly useful when it comes to prediction: making predictions for some subset of the study area relies on stratification in design-based methods and as such can be rather limited. Our models also allow inference from a sample of sightings to a population in a study area without depending upon a random sample design,
and therefore data collected from "platforms of opportunity" (Williams et al., 2006) can be used (although a well designed survey is always preferable).

Unbiased estimates are dependent upon either (i) distribution of sampling effort being random throughout the study area (for design-based inference) or (ii) model correctness (for model-based inference). It is easier to have confidence in the former rather than in the latter because our models are always wrong. Nevertheless model-based inference will play an increasing role in population assessment as the availability of spatially-referenced data increases.

The field is quickly evolving to allow modelling of more complex data building on the basic ideas of density surface modelling. We expect to see large advances in temporal inferences and the handling of zero-inflated data and spatial correlation. These should become more mainstream as modern spatio-temporal modelling techniques are adopted. Petersen et al. (2011) provided a very basic framework for temporal modelling; their model included “before” and “after” smooth terms to quantify the impact of the construction of an offshore windfarm. Zero-inflation in count data may be problematic and two-stage approaches such as Barry & Welsh (2002) as well as more flexible response distributions made possible by Rigby & Stasinopoulos (2005) have yet to be exploited by those using distance sampling data. Spatial autocorrelation can be accounted for via approaches that explicitly introduce correlations such as generalized estimating equations (GEEs; Hardin & Hilbe, 2003) or generalized additive mixed models or via mechanisms such as that of Skaug (2006), which allow observations to cluster according to one of several states (such as high vs low density patches, possibly in response to
temporary agglomerations of prey, although the mechanism is unimportant). These advances should assist both modellers and wildlife managers to make optimal conservation decisions.

Advances in Bayesian computation (INLA; Rue et al., 2009), make one-step, Bayesian, density surface models computationally feasible (as INLA is an alternative to MCMC). An important step toward such models will be incorporation of detection function estimation into the spatial model. We anticipate that such a direct modelling technique will dominate future developments in the field.

Density surface modelling allows wildlife managers to make best use of the available spatial data to understand patterns of abundance, and hence make better conservation decisions (e.g., about reserve or development placement). The recent advances mentioned here increase the reliability of the outputs from a modelling exercise, and hence the efficacy of these decisions. Density surface modelling from survey data is an active area of research, and we look forward to further improvements and extensions in the near future.

**Acknowledgments**

We wish to thank Paul Conn, another anonymous reviewer, and the associate editor for their helpful comments. DLM wishes to thank Mark Brav- ington and Sharon Hedley for their detailed discussions and for providing code for their variance propagation method. Funding for the implementation of the recent advances into the *dsm* package and Distance software came from the US Navy, Chief of Naval Operations (Code N45), grant number
References


Halpin, P., Read, A., Fujioka, E., Best, B., Donnelly, B., Hazen, L., Kot, C.,
Urian, K., LaBrecque, E., Dimatteo, A., Cleary, J., Good, C., Crowder, L.
& Hyrenbach, K.D. (2009) OBIS-SEAMAP: The world data center for marine

Hall/CRC, London, UK.

Journal of Agricultural, Biological, and Environmental Statistics, 9, 181–199.

making ecological inference from distance sampling data. Biometrics, 66, 310–
318.


of bird density using multiple-covariate distance sampling. The Auk, 124, 1229–
1243.

Hall/CRC.

thesis, University of Bath.

dsm: Density surface modelling of distance sampling data.
URL http://github.com/dill/dsm

Miller, D.L. & Wood, S.N. (submitted) Finite area smoothing with generalized
distance splines.

Moore, J.E. & Barlow, J. (2011) Bayesian state-space model of fin whale abundance
trends from a 1991-2008 time series of line-transect surveys in the California

transect data. Journal of Agricultural, Biological, and Environmental Statistics,
15, 327–345.

Oppel, S., Meirinho, A., Ramírez, I., Gardner, B., O’Connell, A., Miller, P. &
Louzao, M. (2011) Comparison of five modelling techniques to predict the spatial
distribution and abundance of seabirds. Biological Conservation, 156, 94–104.


Figures

**Fig. 1** Estimated detection function for pantropical dolphin clusters overlaid onto the scaled histogram of observed distances. Distances are recorded in metres.
Fig. 2  The region, transect centrelines and location of detected pantropical dolphin clusters, where size of circle corresponds to the cluster size, overlaid onto depth data.
Fig. 3  Predicted abundance of dolphins from the DSM using only depth as an explanatory variable (top) and the model using both depth and location (bottom).
Fig. 4  Plot of the effect on the response of depth, given location (from the model with both depth and location smooths). Note that it is possible to draw a straight line between 750m and 3000m within the confidence band (between the dashed lines), so the wiggles in the smooth may not be indicative of any relationship. What is clear is that there the estimated number of dolphins increases up to a water depth of about 500m. The rug ticks at the bottom of the plot indicate we have good coverage of the range of depth values in the survey area. Note that the y axis in such plots is on the scale of the link function (log in this case), so care should be taken in their interpretation.
Fig. 5  Map of the coefficients of variation for the model with smooths of both depth and location. Uncertainty was estimated using the variance propagation method of Williams et al. (2011). As might be expected, there is high uncertainty where there is low sampling effort (Fig. 2).
Fig. 6  Flow diagram showing the modelling process for creating a density surface model.
1. INTRODUCTION

The analysis is based on a dataset of observations of pantropical dolphins in the Gulf of Mexico (shipped with Distance 6.0). For convenience the data are bundled in an R-friendly format, although all of the code necessary for creating the data from the Distance project files is available at the below URL. The OBIS-SEAMAP page for the data may be found at http://seamap.env.duke.edu/dataset/25.

The intention here is to highlight the features of the dsm package, rather than perform a full analysis of the data. For that reason, some important steps are not fully explored. Some familiarity with density surface modelling is assumed.

This is a knitr document. The source for this document contains everything you need to reproduce the analysis given here (aside from the data). The most recent version of this document can be found at github.com/dill/mexico-data.

In section 2, we show how the data and packages can be loaded, before moving on to describing the data and then producing some exploratory plots (in sections 3 and 4, respectively). In section 5 we fit a detection function, then a selection of DSMs in section 6. A detection function with covariates is fitted in section 7. An example using correlation structures is shown in section 8, before concluding.

2. PREAMBLE

Before we start, we load the dsm package (and its dependencies) and set some options:

```r
library(dsm)
```

```r
## Loading required package: mgcv
## This is mgcv 1.7-24.
## For overview type 'help("mgcv-package")'.
## Loading required package: mrds
## Loading required package: optimx
## Loading required package: numDeriv
## Loading required package: Rsolnp
## Loading required package: truncnorm
## Loading required package: parallel
## This is mrds 2.1.1 Built: R 3.0.1; ; 2013-06-25 16:32:18 UTC; unix
```
## Loading required package: ggplot2
## Loading required package: Distance
## This is dsm 2.0.5 Built: R 3.0.1; ; 2013-06-29 17:14:59 UTC; unix

# plotting options
gg.opts <- theme(
  panel.grid.major = element_blank(),
  panel.grid.minor = element_blank(),
  panel.background = element_blank())

# make the results reproducible
set.seed(11123)

3. The data

3.1. Observation and segment data. All of the data for this analysis has been nicely pre-formatted and is shipped with dsm. Loading up that data, we can see that we have four data frames, the first few lines of each are shown:

data(mexdolphins)
attach(mexdolphins)
head(segdata)

## latitude longitude Effort Transect.Label Sample.Label depth x
## 1 29.94 -86.93 13800 19960417 19960417-1 135.0 134159
## 2 29.84 -86.83 14000 19960417 19960417-2 147.7 143496
## 3 29.75 -86.74 14000 19960417 19960417-3 152.1 152050
## 4 29.66 -86.65 13900 19960417 19960417-4 163.8 161102
## 5 29.56 -86.57 13800 19960417 19960417-5 179.7 169553
## 6 29.49 -86.49 13800 19960417 19960417-6 188.5 176793
## y
## 1 325561
## 2 314055
## 3 304324
## 4 293475
## 5 282984
## 6 275103

head(distdata)

## object size distance Effort detected beaufort latitude longitude
## 45 45 21 3296.6 36300 1 4 27.73 -86.00
## 61 61 150 929.2 17800 1 4 26.00 -87.63
## 63 63 125 6051.0 21000 1 2 26.01 -87.95
## 85 85 75 5499.7 21800 1 1 27.50 -90.45
## 114 114 50 7259.0 13400 1 3 27.41 -94.99
## 120 120 45 1454.8 20900 1 5 26.02 -95.97
## x y
## 45 228139 79258
## 61 69199 -113083
## 63 37046 -112197  
## 85 -210016 54208  
## 114 -658878 43337  
## 120 -764824 -111005

**head**(obsdata)

## object Sample.Label size distance Effort  
## 45 45 19960421-9 21 3296.6 36300  
## 61 61 19960423-7 150 929.2 17800  
## 63 63 19960423-9 125 6051.0 21000  
## 85 85 19960427-1 75 5499.7 21800  
## 114 114 19960430-8 50 7259.0 13400  
## 120 120 19960501-5 45 1454.8 20900

**head**(preddata)

## latitude longitude depth x y width height  
## 1 30.08 -87.58 35 70832 341079 32072 37065  
## 2 30.08 -87.42 30 86868 341079 32072 37065  
## 3 30.08 -87.25 27 102904 341079 32072 37065  
## 4 30.08 -87.08 22 118940 341079 32072 37065  
## 5 30.08 -86.92 46 134976 341079 32072 37065  
## 6 29.92 -87.75 14 54888 322546 32126 37065

**distdata** holds the distance sampling data that will be used to fit the detection function.  
**segdata** holds the segment data: the transects have already been “chopped” into segments.  
**obsdata** holds the observations which have already been aggregated to the segments and  
**preddata** holds the prediction grid (which includes all the necessary covariates).

Typically (i.e. for other datasets) it will be necessary divide the transects into segments, and allocate observations to the correct segments using a GIS or other similar package, before starting an analysis using **dsm**.

### Converting units.

It is important to ensure that the measurements to be used in the analysis are in compatible units, otherwise the resulting estimates will be incorrect or hard to interpret. Having all of our measurements in SI units from the outset removes the need for conversion later, making life much easier. All of the data are already in the appropriate units (Northings and Eastings: kilometres from ~ -88.32 longitude, ~27.02 latitude, which is the centroid of the study region, multiplied up by 1000 to get the result in metres, for consistency).

We give an example of converting the survey area here to show that this is a simple process:

```r
# centroid
lon0 <- -88.31951
lat0 <- 27.01594

sa.tmp <-latlong2km(survey.area$longitude, survey.area$latitude, lon0 = lon0, lat0 = lat0)
```
survey.area <- data.frame(x = 1000 * sa.tmp$km.e, y = 1000 * sa.tmp$km.n)
rm(sa.tmp)

The function `latlong2km` uses the spherical law of cosines to convert latitude and longitude into Northings and Eastings (thanks to Simon N. Wood for providing code). There is extensive literature about when particular projections of latitude and longitude are appropriate and we highly recommend the reader review this for their particular study area. The other data frames have already had their measurements appropriately converted. By convention the directions are named x and y.

Using latitude and longitude when performing spatial smoothing can be problematic when certain smoother bases are used. In particular when bivariate isotropic bases are used the non-isotropic nature of latitude and longitude is inconsistent (moving one degree in one direction is not the same as moving one degree in the other).

The below code generates Figure 1, which shows the survey area with the transect lines overlaid (using data from `segdata`).

```r
p <- qplot(data = survey.area, x = x, y = y, geom = "polygon",
            fill = I("lightblue"), ylab = "y", xlab = "x", alpha = I(0.7))
p <- p + coord_equal()
p <- p + gg.opts
p <- p + geom_line(aes(x, y, group = Transect.Label), data = segdata)
print(p)
```

### 4. Exploratory data analysis

#### 4.1. Distance data.

The top panels of Figure 2, below, show histograms of observed distances and cluster size, while the bottom panels show the relationship between observed distance and observed cluster size, and the relationship between observed distance and Beaufort sea state. The plots show that there is some relationship between cluster size and observed distance (fewer smaller clusters seem to be seen at larger distances).

The following code generates Figure 2:

```r
par(mfrow = c(2, 2))

# histograms
hist(distdata$distance, main = "", xlab = "Distance (m)"

hist(distdata$size, main = "", xlab = "Cluster size")

# plots of distance vs. cluster size
plot(distdata$distance, distdata$size, main = "", xlab = "Distance (m)",
     ylab = "Group size", pch = 19, cex = 0.5, col = rgb(0.74, 0.74, 0.74, 0.7))

# lm fit
l.dat <- data.frame(distance = seq(0, 8000, len = 1000))
lo <- lm(size ~ distance, data = distdata)
```
**Figure 1.** The survey area with transect lines.

\begin{verbatim}
lines(l.dat$distance, as.vector(predict(lo, l.dat)))
plot(distdata$distance, distdata$beaufort, main = "", xlab = "Distance (m)", ylab = "Beaufort sea state", pch = 19, cex = 0.5, col = rgb(0.74, 0.74, 0.74, 0.7))
\end{verbatim}

4.2. **Spatial data.** Looking separately at the spatial data without thinking about the distances, we can see the distribution of group size in space in Figure 3, below. Circle size indicates the size of the group in the observation. There are rather large areas with no observations, which might cause our variance estimates to be rather large. This plot shows that we don’t seem to have many observations in the very shallow areas near the shore. This should make us skeptical of predictions in those areas. We will use depth later as an explanatory covariate in our spatial model. Figure 3 also shows the raw depth data.

The following code generates Figure 3:

\begin{verbatim}
p <- ggplot(preddata)
p <- p + gg.opts
p <- p + coord_equal()
p <- p + labs(fill = "Depth", x = "x", y = "y", size = "Group size")
p <- p + geom_tile(aes(x = x, y = y, fill = depth, width = width, height = height))
p <- p + geom_line(aes(x, y, group = Transect.Label), data = segdata)
p <- p + geom_point(aes(x, y, size = size), data = distdata, colour = "red",
\end{verbatim}
5. ESTIMATING THE DETECTION FUNCTION

We use the `ds` function in the package `Distance` to fit the detection function. (The `Distance` package is intended to make standard distance sampling in R relatively straightforward. For a more flexible, but harder to use, alternative, see the function `ddf` in the `mrds` library.)

First, loading the `Distance` library:

```
library(Distance)
```
We can then fit a detection function with hazard-rate key with no adjustment terms:

```r
hr.model <- ds(distdata, max(distdata$distance), key = "hr", adjustment = NULL)
```

## Fitting hazard-rate key function AIC= 841.253
## No survey area information supplied, only estimating detection function.

```r
summary(hr.model)
```

## Summary for distance analysis
## Number of observations : 47
## Distance range : 0 - 7847
##
## Model : Hazard-rate key function
## AIC : 841.3

## Detection function parameters
## Scale Coefficients:
## (Intercept) 7.983 0.9532

```
The following code generates a plot of the fitted detection function (Figure 4):

```r
plot(hr.model)
```

**Figure 4.** Plot of the fitted detection function.

For brevity, detection function model selection has been omitted here. In practise we would fit many different forms for the detection function. Later in this document, we demonstrate fitting of a detection function with size as a covariate, but for now we stick to a simple model.

### 6. Fitting a DSM

Before fitting a `dsm` model, the data must be segmented; this consists of chopping up the transects and attributing counts to each of the segments. As mentioned above, these data have already been segmented.
6.1. **A simple model.** We begin with a very simple model. We assume that the number of individuals in each segment are quasi-Poisson distributed and that they are a smooth function of their spatial coordinates (note that the formula is exactly as one would specify to `gam` in `mgcv`). The abundance of clusters/groups rather than individuals can be estimated by setting `group=TRUE` (though we ignore this here).

Running the model:

```r
mod1 <- dsm(N ~ s(x, y), hr.model$ddf, segdata, obsdata)
summary(mod1)
```

```
##
## Family: quasipoisson
## Link function: log
##
## Formula: N ~ s(x, y) + offset(off.set)
## <environment: 0x10a71c2e8>
##
## Parametric coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## (Intercept) -18.409     0.394  -46.7  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
## edf Ref.df  F p-value
## s(x,y) 26.1 28.2 5.61 <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) = 0.113  Deviance explained = 44%
## GCV score = 42.985 Scale est. = 37.611  n = 387
```

We can then make the predictions over the the grid and calculate abundance. First we must create the offset (the area of each grid cell, which is 444km$^2$).

```r
off.set <- 444 * 1000 * 1000
modl.pred <- predict(mod1, preddata, off.set)
```

Figure 5 shows a map of the predicted abundance. Before plotting, we bind on the predictions to the data used to create them:

```r
pp <- cbind(preddata, modl.pred)
p <- ggplot(pp) + gg.opts
p <- p + geom_tile(aes(x = x, y = y, fill = modl.pred, width = width, height = height))
p <- p + coord_equal()
p <- p + geom_path(aes(x = x, y = y), data = survey.area)
p <- p + labs(fill = "Abundance")
print(p)
```
We can calculate abundance over the survey area by simply summing these predictions:

```r
sum(mod1.pred)
## [1] 47034
```

Figure 6 shows diagnostic plots for the model, generated with the following code:

```r
gam.check(mod1)
```

These show that there is some deviation in the Q-Q plot. The “line” of points in the plot of the residuals vs. linear predictor plot corresponds to the zeros in the data.
To check for residual autocorrelation we use the `dsm.cor` function:

```r
dsm.cor(mod1, max.lag = 10)
```

The plot is shown in Figure 7, and appears to have a spike at lag 6, but this is beyond the range of being interesting.

We can use the approach of Williams et al (2011), which accounts for uncertainty in detection function estimation in this situation where we have no covariates in the detection function.

```r
preddata.varprop <- split(preddata, 1:nrow(preddata))
offset.varprop <- as.list(rep(off.set, nrow(preddata)))
mod1.varprop <- dsm.var.prop(mod1, pred.data = preddata.varprop, 
                           off.set = offset.varprop)
```

Calling `summary` will give some information about uncertainty estimation:

```r
summary(mod1.varprop)
```

## Summary of uncertainty in a density surface model calculated 
## by variance propagation. 
##
## Quantiles of differences between fitted model and variance model 
## Min.  1st Qu.  Median     Mean  3rd Qu.     Max. 
## -6.57e-13 -1.00e-15  1.20e-14  1.08e-13  1.11e-13  3.67e-12
12 DAVID L. MILLER, M. LOUISE BURT, ERIC A. REXSTAD AND LEN THOMAS

Figure 7. Residual autocorrelation in mod1.

## Approximate asymptotic confidence interval:
## 5% Mean 95%
## 29962 47034 73832
## (Using delta method)
##
## Point estimate : 47034
## Standard error : 10966
## Coefficient of variation : 0.2331

The section titled Quantiles of differences between fitted model and variance model can be used to check the variance model does not have major problems (values much less than 1 indicate no issues).

We can also make a plot of the CVs using the following code (shown in Figure 8).

```r
plot(mod1.varprop, xlab = "Easting", ylab = "Northing")
```

6.2. Adding another covariate to the spatial model. The data set also contains a depth covariate (which we plotted above). We can include in the model very simply:

```r
mod2 <- dsm(N ~ s(x, y, k = 10) + s(depth, k = 20), hr.model, segdata, obsdata, select = TRUE)
summary(mod2)
```
FIGURE 8. Plot of the coefficient of variation for the study area with transect lines and observations overlaid.

```r
## Family: quasipoisson
## Link function: log
##
## Formula:
## N ~ s(x, y, k = 10) + s(depth, k = 20) + offset(off.set)
## <environment: 0x10b5839c8>
##
## Parametric coefficients:
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -18.823 0.734 -25.6 <2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
## edf Ref.df F p-value
## s(x,y) 3.88 9 2.06 0.00018 ***
## s(depth) 10.52 19 3.15 9.9e-10 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) = 0.0929 Deviance explained = 34%
## GCV score = 46.27 Scale est. = 42.967 n = 387
```
By setting the $k$ parameter we specify the largest complexity for that smooth term in the model; as long as this is high enough (i.e., the number in the edf column is not too close to that in the Ref.df column), we can be sure that there is enough flexibility. However, it may sometimes be necessary to set $k$ to be lower than this to limit the influence of (for example) spatial smoothers in the model.

Setting `select=TRUE` here imposes extra shrinkage terms on each smooth in the model (allowing smooth terms to be removed from the model during fitting; see `?gam` for more information). Although this is not particularly useful here, this can be a good way (along with looking at $p$-values) to perform term selection.

Again we can plot the predictions from this model (Figure 9, code below).

```r
mod2.pred <- predict(mod2, preddata, off.set)
pp <- cbind(preddata, mod2.pred)
p <- ggplot(pp) + gg.opts
p <- p + labs(fill = "Abundance")
p <- p + geom_tile(aes(x = x, y = y, fill = mod2.pred, width = width, height = height))
p <- p + coord_equal()
p <- p + geom_path(aes(x = x, y = y), data = survey.area)
print(p)
```

![Figure 9. Predicted density surface for mod2.](image)

Simply calling `plot` on the model object allows us to look at the relationship between depth and abundance (shown in Figure 10):
plot(mod2, select = 2)

**Figure 10.** Plot of the smooth of depth in mod2.

Omitting the argument *select* in the call to *plot* gives plots of all the smooth terms, one at a time.

### 6.3. A more complicated model. Tweedie

Response distributions other than the quasi-Poisson can be used, for example the Tweedie distribution. If the Tweedie is used, then the p parameter must be specified. The choice of p is only sensitive to the first decimal place, so a quick search can be performed by simply comparing the score of the resulting models. In this simple example we only show the difference between two values for p.

```r
mod3.12 <- dsm(N ~ s(x, y), hr.model, segdata, obsdata, 
                family = Tweedie(p = 1.2))
summary(mod3.12)
```

```r
##
## Family: Tweedie(1.2)
## Link function: log
##
## Formula:
## N ~ s(x, y) + offset(off.set)
## <environment: 0x10bb3f518>
##
## Deviance Residuals:
##       Min        1Q    Median        3Q       Max
## -0.41781   -0.27255   -0.18900   -0.14063    1.08807
##
## Coefficients: (intercept) s(x) s(y) offset
##       Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.00000    0.00000     Inf    0.50
## s(x)        0.00000    0.00000     Inf    0.50
## s(y)        0.00000    0.00000     Inf    0.50
## offset      0.00000    0.00000     Inf    0.50
##
## Log-Likelihood: -2738.6
##
## Number of Smoothing Parameters: 3
## Basis Dimensions: 3 3
##
## Total effective degrees of freedom: 6
```
As well as looking at the GCV/UBRE/REML score of the model to assess the value of $p$ we can also look at a plot of the square root of the absolute value of the residuals versus the fitted values (Figure 11, left panel).

where as setting $p=1.7$:

```r
mod3.17 <- dsm(N ~ s(x, y), hr.model, segdata, obsdata,
               family = Tweedie(p = 1.7))
summary(mod3.17)
```

The plot (Figure 11, right panel; code below), appears to be much flatter.
par(mfrow = c(1, 2))
plot(sqrt(abs(residuals(mod3.12))), predict(mod3.12),
xlab = "Square root of the absolute value of the residuals",
ylab = "Fitted values", main = "p=1.2")
plot(sqrt(abs(residuals(mod3.17))), predict(mod3.17),
xlab = "Square root of the absolute value of the residuals",
ylab = "Fitted values", main = "p=1.7")

**Figure 11.** Plot of absolute value of the residuals versus the fitted values for the Tweedie model when $p=1.2$ (left) and $p=1.7$ (right). Note that the right plot is much flatter than the left.

Note also the improvement in GCV score.

In general, a “good” value can be found by simply plotting the above along with the GCV score for the model for values of $p$ between 1.1 and 1.9 and looking for the best GCV and “flattest” plot.

**Soap film smoothing**

To account for a complex region (e.g., a region that includes peninsulae) we can use the soap film smoother (Wood et al. 2008).

To use a soap film smoother for the spatial part of the model we must create a set of knots for the smoother to use. This is easily done using the `make.soapgrid()` function in dsm:
soap.knots <- make.soapgrid(survey.area, c(11, 6))  # knot 11 is not outside but is too close according to soap...
soap.knots <- soap.knots[-11,]

where the second argument specifies the number of points (in each direction) in the grid that will be used to create the knots (knots in the grid outside of survey.area are removed).

As we saw in the exploratory analysis, some of the transect lines are outside of the survey area. These will cause the soap film smoother to fail, so we remove them:

```r
x <- segdata$x
y <- segdata$y
onoff <- inside(x = x, y = y, bnd = as.list(survey.area))
rm(x, y)
segdata.soap <- segdata[onoff,]
```

We can run a model with both the depth covariate along with a spatial (soap film) smooth.

```r
mod4 <- dsm(N ~ s(x, y, bs = "so", k = 10, xt = list(bnd = list(survey.area)))) + s(depth), hr.model, segdata.soap, obsdata, knots = soap.knots)
```

```r
summary(mod4)
```

## Loading required package: Matrix
## Loading required package: lattice

# Summary

### Family: quasipoisson
### Link function: log

#### Formula:
#### N ~ s(x, y, bs = "so", k = 10, xt = list(bnd = list(survey.area))) + s(depth) + offset(off.set)
#### <environment: 0x10a3dfdf8>

#### Parametric coefficients:
#### Estimate Std. Error t value Pr(>|t|)
#### (Intercept) -19.40 0.62 -31.3 <2e-16 ***
#### ---
#### Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#### Approximate significance of smooth terms:
#### edf Ref.df  F p-value
#### s(x,y) 23.05 29.00 2.63 2e-07 ***
#### s(depth) 7.08 7.89 2.57 0.01 *
#### ---
#### Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

#### R-sq.(adj) = 0.183 Deviance explained = 45.9%
#### GCV score = 45.227 Scale est. = 38.341 n = 365
Comparing predictions from the model that included a smooth of depth, we can see that the soap film has prevented some of the extreme values (especially in the lower right corner of the survey area). This is shown in Figure 12 (figure created with the code below).

```r
mod4.pred <- predict(mod4, preddata, off.set)
pred <- cbind(preddata, mod4.pred)
p <- ggplot(pp) + gg.opts
p <- p + geom_tile(aes(x = x, y = y, fill = mod4.pred,
width = width, height = height))
p <- p + coord_equal()
p <- p + geom_path(aes(x = x, y = y), data = survey.area)
p <- p + labs(fill = "Abundance")
print(p)
```

![Figure 12. Predicted density surface for mod4.](image)

7. ADDING COVARIATES TO THE DETECTION FUNCTION

It is common to include covariates in the detection function (so-called Multiple Covariate Distance Sampling or MCDS). In this dataset there are two covariates that were collected on each individual: Beaufort sea state and size. For brevity we fit only a hazard-rate detection functions with the sea state included as a factor covariate as follows:
hr.beau.model <- ds(distdata, max(distdata$distance),
    formula = ~as.factor(beaufort), key = "hr", adjustment = NULL)
summary(hr.beau.model)

##
## Summary for distance analysis
## Number of observations : 47
## Distance range : 0 - 7847
##
## Model : Hazard-rate key function
## AIC : 843.7
##
## Detection function parameters
## Scale Coefficients:
## estimate se
## (Intercept) 7.66318 1.077
## as.factor(beaufort)2 2.27968 17.367
## as.factor(beaufort)3 0.28606 1.019
## as.factor(beaufort)4 0.07174 1.223
## as.factor(beaufort)5 -0.36399 1.537
##
## Shape parameters:
## estimate se
## (Intercept) 0.3004 0.518
##
## Estimate SE CV
## Average p 0.5421 0.1751 0.3229
## N in covered region 86.6957 29.4133 0.3393

In this example, the detection function with covariates does not give a lower AIC than the
model without covariates (hazard-rate model has AIC of 841.25 vs. 843.71 for this model).
Looking back to the bottom-right panel of Figure 2, we can see there is not a discernible
pattern in the plot of Beaufort vs distance.

The code to fit the model is similar to the other models, above. However since we are now
using a detection function with observation-level covariates, we change the response to be
Nhat so the abundances are estimated per segment and change the name of the detection
function model object:

mod5 <- dsm(Nhat ~ s(x, y), hr.beau.model, segdata, obsdata)
summary(mod5)

##
## Family: quasipoisson
## Link function: log
##
## Formula:
## Nhat ~ s(x, y) + offset(off.set)
## <environment: 0x107da5320>
##
## Parametric coefficients:

|                     | Estimate | Std. Error | t value | Pr(>|t|) |
|---------------------|----------|------------|---------|----------|
| (Intercept)         | -18.258  | 0.351      | -52     | <2e-16   *** |

---

## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

---

## Approximate significance of smooth terms:

<table>
<thead>
<tr>
<th></th>
<th>edf</th>
<th>Ref.df</th>
<th>F</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>s(x,y)</td>
<td>26</td>
<td>28.1</td>
<td>5.77</td>
<td>&lt;2e-16   ***</td>
</tr>
</tbody>
</table>

---

## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

## R-sq.(adj) = 0.118  Deviance explained = 42.9%
## GCV score = 75.24  Scale est. = 65.855  n = 387

Note that for models where there are covariates at the individual level we cannot calculate the variance via the variance propagation method (`dsm.var.prop`) of Williams et al (2011). Instead we can use a GAM uncertainty estimation and combine it with the detection function uncertainty via the delta method (`dsm.var.gam`), or use the moving block bootstrap. Other than this, all of the above functions can be used.

A plot of predictions from the covariate model:

```r
mod5.pred <- predict(mod5, preddata, off.set)
pp <- cbind(preddata, mod5.pred)

p <- ggplot(pp) + gg.opts
p <- p + geom_tile(aes(x = x, y = y, fill = mod5.pred, width = width, height = height))
p <- p + coord_equal()
p <- p + geom_path(aes(x = x, y = y), data = survey.area)
p <- p + labs(fill = "Abundance")
print(p)
```

### 8. Correlation structures

We can use a generalized mixed model (GAMM; Wood, 2006) to include correlation between the segments within each transect. First we re-code the sample labels and transect labels as numeric variables, then include them in the model as part of the `correlation` argument. For the sake of example we use an AR1 (lag 1 autocorrelation) correlation structure (though the correlogram did not indicate we had issues with residual autocorrelation, we show it here for illustrative purposes).

```r
segdata$sg.id <- as.numeric(segdata$Sample.Label)
segdata$str.id <- as.numeric(segdata$Transect.Label)
mod1.gamm <- dsm(N ~ s(x, y), hr.model$ddf, segdata, obsdata, engine = "gamm", correlation = corAR1(form = ~sg.id), method = "REML")

## Loading required package: nlme
```
FIGURE 13. Predicted density surface for mod5.

## Maximum number of PQL iterations: 20

## iteration 1 iteration 2 iteration 3 iteration 4 iteration 5 iteration 6
## iteration 7 iteration 8 iteration 9 iteration 10

This example also includes using method="REML" for smoothing parameter selection.

GAMMs usually take considerably longer to fit than GAMs, so it’s usually start with a GAM first, select smooth terms and response distribution before starting to fit GAMMs.

The object returned is part lme (for the random effects) and part gam (for the smooth terms). Looking at the summary() for the gam part of the model:

```
summary(mod1.gamm$gam)
```

## Family: quasipoisson
## Link function: log
##
## Formula:
## \( N \sim s(x, y) + \text{offset}(\text{off.set}) \)
## <environment: 0x108fb1ed0>
##
## Parametric coefficients:
##
## Estimate Std. Error t value Pr(>|t|)
## (Intercept) -17.617 0.277 -63.5 <2e-16 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Approximate significance of smooth terms:
## edf Ref.df F  p-value
## s(x,y) 20.4 20.4 4.08 1.5e-08 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## R-sq.(adj) = 0.129 Scale est. = 68.686 n = 387

And run checks on the GAM part of the object (shown in Figure 14):

```r
gam.check(mod1.gamm$gam)
```

![Diagnostic plots for mod1.gamm.](image)

**Figure 14.** Diagnostic plots for `mod1.gamm`.

We can now make predictions and compare the abundance map produced by this model to the previous models (Figure 15):

```r
mod1.gamm.pred <- predict(mod1.gamm, preddata, off.set)
pp <- cbind(preddata, N = mod1.gamm.pred)
p <- ggplot(pp) + gg.opts
p <- p + geom_tile(aes(x = x, y = y, fill = N, width = width, height = height))
p <- p + coord_equal()
p <- p + geom_path(aes(x = x, y = y), data = survey.area)
```
p <- p + labs(fill = "Abundance")
print(p)

FIGURE 15. Predicted density surface for mod1.gamm.

Again, estimating variance is straightforward using the variance propagation method:

mod1.gamm.var <- dsm.var.prop(mod1.gamm, pred.data = preddata, off.set = off.set)

## Maximum number of PQL iterations: 20
## iteration 1 iteration 2 iteration 3 iteration 4 iteration 5 iteration 6
## iteration 7 iteration 8 iteration 9

summary(mod1.gamm.var)

## Summary of uncertainty in a density surface model calculated
## by variance propagation.
## Quantiles of differences between fitted model and variance model
## Min. 1st Qu. Median Mean 3rd Qu. Max.
## -4.120 -0.531 -0.244 -0.073 -0.019 19.500
## Approximate asymptotic confidence interval:
## 5% Mean 95%
## 35389 45701 59018
## (Using delta method)

### Point estimate : 45701
### Standard error : 5988
### Coefficient of variation : 0.131

Comparing this to the variance from mod1, we can see the GAMM offers a significant reduction in variance:

```r
mod1.varprop <- dsm.var.prop(mod1, pred.data = preddata.varprop, 
                            off.set = offset.varprop)
summary(mod1.varprop)
```

### Summary of uncertainty in a density surface model calculated by variance propagation.

### Quantiles of differences between fitted model and variance model

<table>
<thead>
<tr>
<th>Min.</th>
<th>1st Qu.</th>
<th>Median</th>
<th>Mean</th>
<th>3rd Qu.</th>
<th>Max.</th>
</tr>
</thead>
<tbody>
<tr>
<td>-6.57e-13</td>
<td>-1.00e-15</td>
<td>1.20e-14</td>
<td>1.08e-13</td>
<td>1.11e-13</td>
<td>3.67e-12</td>
</tr>
</tbody>
</table>

### Approximate asymptotic confidence interval:

<table>
<thead>
<tr>
<th>5% Mean</th>
<th>95%</th>
</tr>
</thead>
<tbody>
<tr>
<td>29962</td>
<td>47034 73832</td>
</tr>
</tbody>
</table>

## (Using delta method)

### Point estimate : 47034
### Standard error : 10966
### Coefficient of variation : 0.2331

More information on lme can be found in Pinheiro and Bates (2000) and Wood (2006).

9. CONCLUSIONS

This document has shown that the dsm package is a versatile and relatively easy-to-use package for the analysis of spatial distance sampling data. Note that there are many possible models that can be fitted using dsm and that the aim here was to show just a few of the options. Results from the models can be rather different, so care must be taken in performing model selection, discrimination and criticism.

NOTES

- **Distance** is available at [http://github.com/dill/Distance](http://github.com/dill/Distance) as well as on CRAN.
- **dsm** is available (along with some documentation and hints) at [http://github.com/dill/dsm](http://github.com/dill/dsm), as well as on CRAN.

REFERENCES

APPENDIX B: CALCULATION OF VARIANCE IN DENSITY SURFACE MODELS

DAVID L. MILLER, M. LOUISE BURT, ERIC A. REXSTAD AND LEN THOMAS

1. Introduction

This appendix gives a brief mathematical explanation of the method proposed in Williams et al. (2011) for the propagation of uncertainty from the detection function to the DSM, as well as how to calculate the variance of a non-linear function of a GAM (e.g. when calculating the variance of the predicted abundance).

2. Variance propagation

The formulation for a “count method” density surface model (DSM) is:

$$
E(n_j) = \exp \left[ \log (p_j(\theta)A_j) + \sum_{k=1}^{K} f_k(z_{jk}) \right],
$$

where we model the expected number of animals per segment ($n_j$). The $f_k$s are smooth functions of the covariates and $\beta_0$ is an intercept term. $A_j$ is the covered area and the probability of detection is given by ($\hat{p}_j$) and is estimated from the detection function.

Writing $\hat{p}_j$ explicitly as a function of the estimated detection function parameters $\hat{\theta}$ and exponentiating both sides yields:

$$
\exp[E(n_j)] = \log (A_j) + \log [p_j(\hat{\theta})] + \sum_{k=1}^{K} f_k(z_{jk}).
$$

At this point we add another term to the model. This new term is the derivative of $\log [\hat{p}(\hat{\theta})]$ multiplied by $\gamma = (\theta - \hat{\theta})$.

(1) $$
\exp[E(n_j)] = \log (A_j) + \log [p_j(\hat{\theta})] + \frac{d \log p(\theta)}{d\theta} \bigg|_{\theta=\hat{\theta}} \gamma + \sum_{k=1}^{K} f_k(z_{jk}).
$$

This term has basically no effect on the model, since, using the definition of a finite difference:

$$
\frac{d \log p(\theta)}{d\theta} \bigg|_{\theta=\hat{\theta}} = \left\{ \log \left[ p(\hat{\theta} + \delta) \right] - \log \left[ p(\hat{\theta}) \right] \right\} \delta^{-1},
$$

(if we assume that $\gamma$ is small enough such that $\gamma \approx \delta$).
We may then write (1) as:

\[
\exp [\mathbb{E}(n_j)] = \log (A_j) + \log \left[ p_j(\hat{\theta}) \right] + \frac{d \log p(\theta)}{d \theta} \bigg|_{\theta=\hat{\theta}} \gamma + \sum_{k=1}^{K} f_k(z_{jk}),
\]

\[
= \log (A_j) + \log \left[ p_j(\hat{\theta}) \right] + \left\{ \log \left[ p(\theta + \delta) \right] - \log \left[ p(\hat{\theta}) \right] \right\} \delta^{-1} \gamma + \sum_{k=1}^{K} f_k(z_{jk}).
\]

Assuming that \( \theta \) and \( \hat{\theta} \) are “close” we can say that \( \gamma \approx \delta \), so:

\[
\exp [\mathbb{E}(n_j)] \approx \log (A_j) + \log \left[ p_j(\hat{\theta}) \right] + \log \left[ p(\theta + (\theta - \hat{\theta})) \right] - \log \left[ p(\theta) \right] + \sum_{k=1}^{K} f_k(z_{jk})
\]

\[
\approx \log (A_j) + \log \left[ p_j(\hat{\theta}) \right] + \log \left[ p(\theta) \right] - \log \left[ p(\hat{\theta}) \right] + \sum_{k=1}^{K} f_k(z_{jk})
\]

\[
\approx \log (A_j) + \log \left[ p(\theta) \right] + \sum_{k=1}^{K} f_k(z_{jk}).
\]

So this extra term does not have a large effect on the resulting GAM. It does however have an effect on the variance estimates derived from the model. In practice, we can look at the difference between the model coefficients in the model with an without the extra term to check that there has been no large change in the model.

### 3. Calculating the Variance of the Abundance

To find the variance of the predicted abundance we are finding the variance of a function of the linear predictor (in the case of the abundance, this is simply the sum). We begin by revising some basic GAM theory before moving on to the specific case of DSMs.

When the identity link is used, finding the variance of some function of the model is relatively easy. The \texttt{lpmatrix} (Wood, 2006, page 245) is used, that is the matrix \( X_p \) such that:

\[
\hat{\eta}_p = X_p \hat{\beta}
\]

i.e. \( X_p \) maps the model parameters \( \hat{\beta} \) to the linear predictor \( \hat{\eta}_p \). We can then use \( X_p \) to find the covariance matrix for the linear predictor if we can estimate the parameter covariance matrix \( V_{\hat{\beta}} \):

\[
V_{\hat{\eta}_p} = X_p V_{\hat{\beta}} X_p^T.
\]

Only linear functions of the linear predictor can be calculated using this method but this just consists of changing the pre- and post-multiplying matrices. When the link function is not the identity calculations are not so straightforward, we now illustrate two ways of obtaining variance estimates when using a non-identity link function.

#### 3.1. Calculation by simulation.

First note that the distribution of the parameters (given the data) is multivariate normal with mean as the parameter estimates and the covariance matrix of the parameters. (i.e. \( \beta \sim N(\hat{\beta}, V_{\hat{\beta}}) \)).

The following algorithm is suggested by (Wood, 2006, page 246):

1. For \( b = 1, \ldots, N_b \) do the following:
(a) Simulate from $\beta \sim N(\hat{\beta}, V_{\hat{\beta}})$, to obtain $\beta_b$.
(b) Calculate $\hat{\eta}_b = \exp(X_p\beta_b)$ (e.g. if we are using the log-link)
(c) Sum over the survey area
(2) Calculate the appropriate summary statistics, e.g. median, 95% quantiles etc over $b$.

In practice $N_b$ does not have to be particularly large, Marra et al. (2011) achieve good results with $N_b = 100$.

3.2. Calculation by the delta method. Simulation may well be unnecessary and it may well be easier and more efficient to use the delta method:

$$
\left( \frac{\partial^2 \log e^{\eta}}{\partial \eta^2} \right)_{\eta = \hat{\eta}} \otimes X_p^T V_p \left( \frac{\partial^2 \log e^{\eta}}{\partial \eta^2} \right)_{\eta = \hat{\eta}} \otimes X_p^T ,
$$

where $\frac{\partial^2 \log e^{\eta}}{\partial \eta^2} \bigg|_{\eta = \hat{\eta}}$ is the vector of second derivatives of the link evaluated at the values of the linear predictor and $\otimes$ denotes $R$-style matrix-vector multiplication. The delta method inflates the variance based on the uncertainty in the linear predictor.

REFERENCES

APPENDIX C: RELATIVE PERFORMANCE OF GAM AND BOOTSTRAP UNCERTAINTY ESTIMATION IN DSM

DAVID L. MILLER, M. LOUISE BURT, ERIC A. REXSTAD, AND LEN THOMAS

1. INTRODUCTION

This appendix gives a summary of a series of very simple simulations that were conducted to investigate the properties of the four methods of uncertainty estimation described in the article. These are: moving block bootstrap (MBB; including detection function uncertainty via the delta method), moving block bootstrap with simulated detection function uncertainty (MBB+SDU; new distance generated from the fitted detection function), GAM uncertainty (GAMU; using standard GAM theory including detection function uncertainty via the delta method) and GAM uncertainty with variance propagation (VARPROP).

Three (admittedly very simple) scenarios were tested; each consisted of a density based on a simple pattern. Creating a “realistic” scenario (e.g. including many covariates, correlation and so on) is not easy to construct and such a scenario would never encompass all of the possible combinations. As such, we decided to create three simple situations where it would be easy to discover whether there were difficulties with any of the above methods. The R WiSP (available at http://www.ruwpa.st-and.ac.uk/estimating.abundance/WiSP/index.html) was used to generate samples and the packages dsm and Distance (available from CRAN) were used to calculate the abundance estimates and corresponding confidence intervals.

2. SCENARIOS

The underlying density surfaces are shown in Figure 1. For each of these scenarios, the population size was fixed (at 500 or 5000 individuals) and individuals were randomly placed in the survey area according to the density surface with group size generates from a Poisson distribution with mean 3. A systematic grid of 14 transect lines (parallel to the y axis) the “height” of the survey area were placed at a random starting point on the x axis. Points were then detected based on a half-normal detection function with a scale parameter of 0.7, the truncation distance was set to 1.

The observations were then allocated to 2 by 2 segments, and a DSM was fitted to the counts. Only a simple bivariate smooth of x and y was fitted to the per-segment counts and a quasi-Poisson response was used.

Once the model had been fitted, the variance of the predicted abundance was estimated using the four methods listed above. The two moving block bootstraps ran for 200 iterations with a block size of 5.
For each combination of population size and underlying density surface, 100 simulations were run. There were no convergence failures.

3. Results

3.1. Population size 5000. When the true population size in the survey area was 5000, the number of detections was around 900, which seems more than adequate to fit a GAM.

Figure 2 shows the confidence intervals calculated from each of the four methods for each simulation. The left column gives the simulations in order, the right column groups the same results by method. The larger population simulations show that the GAM and variance propagation methods give significantly smaller confidence intervals.

3.2. Population size 500. For the smaller population size the number of detections was around 90. In this situation, the GAM and variance propagation methods appear to have wider intervals than the bootstrap-based methods, however this does have the side effect that the confidence intervals cover truth more often and in general appear to have less erratic behaviour.

4. Conclusion

Admittedly the simulation is rather simplistic: there is no attempt to incorporate other simulated environmental covariates and only a bivariate smooth of location was used in modelling. However, this set up has shown that when there are a large number of observations the GAM-based methods perform well, with tighter confidence intervals and good coverage. In the situation where there are fewer observations, we see that (as one would expect) confidence intervals expand and appears more “noisy”. It is the moving block bootstrap with simulated detection function uncertainty that appear to suffer the most, having tighter confidence intervals than the other methods but with more erratic behaviour.

Given the significant difference in time taken to calculate the GAM-based intervals and the bootstraps, it seems highly prudent to use the former, especially if it is necessary to obtain uncertainty estimates for a wide variety of models in a short period of time.
**Figure 2.** Results of simulating populations of 5000 individuals in three scenarios (going down the page). Each coloured horizontal line indicates one simulation result and the black vertical line is truth. On the left side results are grouped by simulation and the blue lines give the point estimate of the population for a given simulation. On the right grouping is by method and the blue points indicate the corresponding point estimates.
Figure 3. Results of simulating populations of 5000 individuals in three scenarios (going down the page). Each colored horizontal line indicates one simulation result and the black vertical line is truth. On the left side results are grouped by simulation and the blue lines give the point estimate of the population for a given simulation. On the right grouping is by method and the blue points indicate the corresponding point estimates.
APPENDIX D: DETAILS OF THE TWEEDIE DISTRIBUTION

DAVID L. MILLER, M. LOUISE BURT, ERIC A. REXSTAD AND LEN THOMAS

1. INTRODUCTION

This appendix gives a brief mathematical explanation of the Tweedie distribution.

2. THE TWEEDIE DISTRIBUTION

The Tweedie distribution has three parameters: a mean (µ), dispersion (φ) and a third, power parameter (p), which leads to additional flexibility. The Tweedie distribution is characterised by the mean-variance relationship var(Y) = φµ^p. Setting p = 1 gives a quasi-Poisson distribution and p = 2 gives a gamma distribution. Tweedie random variables are a sum of M gamma variables where M is Poisson distributed (Jørgensen, 1987).

The Tweedie distribution has the following PDF (for 1 < p < 2):

\[
f(y; µ, φ, p) = a(y; φ) \exp \left[ \frac{1}{φ} \left\{ y^{\frac{µ-1}{1-p}} - \frac{µ^{2-p}}{2-p} \right\} \right],
\]

where

\[
a(y; φ) = \frac{1}{y} \sum_{j=1}^{∞} y^{-jα} (p - 1)^{αj} \frac{1}{φ^j (1-α) (2-p)^j j! Γ(-jα)}, \quad α = \frac{2-p}{1-p}
\]

Further technical information can be found in Jørgensen (1987); Dunn and Smyth (2005) and practical applications can be found in Candy (2004); Shono (2008); Peel et al. (2012).

REFERENCES


