POPULATION ESTIMATION IN AFRICAN ELEPHANTS WITH HIERARCHICAL BAYESIAN SPATIAL CAPTURE-RECAPTURE MODELS

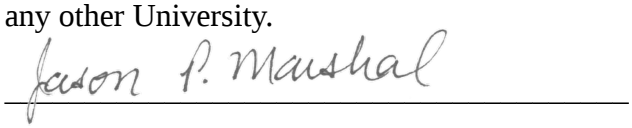
Jason P. Marshal

A Dissertation submitted to the Faculty of Science, University of the Witwatersrand, in fulfilment of the requirements for the degree of Master of Science

Johannesburg, 2017

DECLARATION

I declare that this Dissertation is my own, unaided work. It is being submitted for the Degree of Master of Science at the University of the Witwatersrand, Johannesburg. It has not been submitted before for any degree or examination at any other University.



(Signature of candidate)

29th day of May 2017 in Johannesburg.

Contents

ABSTRACT

With an increase in opportunistically-collected data, statistical methods that can accommodate unstructured designs are increasingly useful. Spatial capture-recapture (SCR) has such potential, but its applicability for species that are strongly gregarious is uncertain. It assumes that average animal locations are spatially random and independent, which is violated for gregarious species. I used a data set for African elephants (*Loxodonta africana*) and data simulation to assess bias and precision of SCR population density estimates given violations in location independence. I found that estimates were negatively biased and likely too precise if non-independence was ignored. Encounter heterogeneity models produced more realistic precision but density estimates were positively biased. Lowest bias was achieved by estimating density of groups, group size, and then multiplying to estimate overall population density. Such findings have important implications for the reliability of population density estimates where data are collected by unstructured means.

DEDICATION

To my parents, Paulette and Layne Marshal.

ACKNOWLEDGEMENTS

This MSc began as a sabbatical project to improve my quantitative skills and align my research focus more toward quantitative population ecology. I am grateful to Profs Kevin Balkwill and Helder Marques for supporting this project as a part of my sabbatical plan. I am also grateful to the Division of Human Resources, University of the Witwatersrand, which provided a staff bursary to cover the tuition fees for the MSc programme.

All statisticians rely on the generosity of others to supply data so that they may ply their trade. In my case those generous suppliers were Dr Michelle Henley and others of Elephants Alive, who put endless hours of toil into collecting and organizing vast amounts of field data. Without that contribution, this project would have ended before it began. Shunmuga Pillay provided invaluable assistance to get me up and running on the Mathematical Sciences computing cluster at Wits University.

Discussions with colleagues and students at the School of Animal, Plant & Environmental Sciences helped with problem solving and keeping me sane during difficult analysis and computing challenges. Francesca Parrini was always available to lend a listening ear and provide support. The researchers and students at the Centre for African Ecology and its regular Friday meetings did much to vet my ideas, and provide discussion and criticism. This dissertation is much better as a consequence.

Fitsum Abadi Gebreselassie was my post-graduate supervisor, but he was also considerably more. In his short time at Wits, he became a trusted colleague and friend. He was an important influence in my learning Bayesian statistics and hierarchical modelling. He also was the primary avenue by which I grew to know the quantitative ecology community of scientists and its literature. Those influences have left an indelible mark on my trajectory as a scientist.

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE – INTRODUCTION

Opportunistic collection of photographically-sampled large mammal data is increasingly common, given the relative inexpense and logistical ease with which such data can be gathered (e.g. (Auger-Méthé et al., 2010); Morrison and Bolger, 2014; Grange et al., 2015; (Marshal, 2017). Wildlife managers in South African game reserves commonly collect photographic data as part of their regular monitoring activities. For species that have individually-recognizable body markings, individuals can be monitored over time. This should permit estimation of population parameters, based on capture-recapture (CR) analysis (Kendall et al., 2004) after accounting for imperfect detection (Lebreton et al., 1992; MacKenzie et al., 2005). Where location data are also available, the spatial nature of such encounter data can be incorporated explicitly by means of spatial capture-recapture (SCR) analysis (Efford, 2004; Borchers and Efford, 2008; Royle and Young, 2008). Whereas most CR and SCR analysis methods assume a structured sampling regime (i.e. collection at fixed sampling locations and times), however, opportunistic collection often does not adhere to a structured regime. Thus, analysis of spatially and temporally unstructured data can present particular challenges when attempting estimation of population parameters.

1.1 Basic spatial capture-recapture models

The analysis methods of SCR were developed to take advantage of spatially explicit information about encounter locations to estimate population density (Efford, 2004; Borchers and Efford, 2008; Royle and Young, 2008). Before the

development of SCR, density estimation consisted of non-spatial CR estimation of abundance across an area containing an array or grid of detection devices (e.g. traps, cameras, hair snares), linked to an ad hoc estimation of the area the sampled population might occupy (Efford, 2004; Royle et al., 2014). That estimation entails buffering the area (i.e. adding an extra area of constant size around the outside of the area) of the sampling array by an amount based on quantities such as the average proportion of time spent on the study site (White and Shenk, 2001), one-half the mean maximum distance moved (MMDM; Wilson and Anderson, 1985), or the full MMDM (Parmenter et al., 2003). Uncertainty in which approach produces the most accurate study area size, combined with uncertainty about which statistical model best represents the CR data, can produce wide-ranging estimates and substantial doubt about the true density of the population (e.g. 0.178-0.432 individuals/km^2; Royle et al., 2014:Table 1.1).

The solution to ad hoc estimation of the study area size is to incorporate information about the locations of encounter to estimate density directly. Efford's (2004) formulation models density of locations of animals in a study area as the outcome of a spatial point process. The points represent the activity centres of individual animals in the study area (Royle and Young, 2008; Borchers, 2012). These activity centres could have some biological importance, or they might merely be the average location (i.e. centroid) of all the locations an individual animal exhibits through its movements. These activity centres, however, are unobserved, and must be estimated through the locations of the detectors that record an individual's encounters (Royle et al., 2014). The SCR approach also

incorporates an explicit connection between space and an individual by way of a model where population size scales with the size of the study area (Royle and Young, 2008). This model is based on a distance sampling approach (Royle and Dorazio, 2008), where the probability of an individual encountering a detector is a decreasing function of the distance from the animal's activity centre (Borchers and Efford, 2008; Royle and Young, 2008). Those distances are treated as individual covariates in a model relating encounter probability to distance. Because centre of activity is unobserved, distance to centre is treated as a random effect, and either integrated from a conditional-on-activity-centre likelihood within a classical framework (Borchers and Efford, 2008; Dawson and Efford, 2009), or assigned a prior distribution and modelled explicitly within a hierarchical Bayesian framework (Royle and Young, 2008). As with distance sampling, a number of models can be used to represent the change in encounter probability with distance from activity centre (e.g. half-normal, logit, exponential, hazard; Buckland et al., 2001).

1.2 Gregarious animals and non-independent encounters

A central assumption of SCR analysis is that encounters of individuals are independent of those of other individuals. This independence is a function of a model that treats locations of activity centres as spatially random (i.e. homogeneous point process; Efford, 2004; Borchers and Efford, 2008; Royle and Young, 2008). In cases where study species occur in aggregations or social groups, this assumption is violated. Where group membership is known, analysis methods can accommodate aggregated data. For example, Russell et al. (2012)

treated each animal aggregation as the unit of observation, estimated average aggregation size, and multiplied the two to estimate total density of animals. For encounters collected opportunistically, however, there could be missing information about group membership. Groups might be inferred from coincidental encounters of individuals at the same time and place. For species that occur in large herds, accurate identification of individuals within groups and accurate assignment of individuals to groups can be difficult, even for experienced researchers. Thus, for animals that commonly occur in groups, the validity of SCR analysis and the possible effects of violating its assumptions might affect the applicability of such methods to social or gregarious animals.

1.3 Sources of data

Data for this project were generated by the elephant monitoring programme in the Associated Private Nature Reserves (APNR; Elephants Alive, http://elephantsalive.org/; accessed 5 December 2016) to the west of Kruger National Park, South Africa (E 30.8-31.5º, S 24.1-24.5º). Data collection spanned the years 2002-2014, and provided spatially- and temporally-explicit photographic documentation of every individual encountered in the APNR. Photographs were collected by research teams in field vehicles during the course of monitoring in the reserve. A location consisted of a GPS coordinate recorded by the research team while in a research vehicle. The true location of the elephant was <50 m from the vehicle (M. Henley, pers. comm.). Individual animals were recognized by the characteristics of the ears, the pattern of the front of the trunk, and other unique markings (e.g. scars, missing tail). The core of the data set consists of

date-, time-, and location-specific records of recognized individuals.

Many individuals in the database had one encounter in 13 years of data. An encounter was defined as an instance where a photographed elephant was recognized as an individual already in the encounter database. I assumed that solo-encounter animals were in the encounter database because they were transient (i.e. moving through the study area, but not a member of the resident population; Pradel et al., 1997) or they were already in the database but mistakenly assigned a new identity. In both cases, such individuals are not expected to contribute information to the population under study (Morrison et al., 2011). Deleting the first encounter for each individual in the database effectively removed all solo-encounter animals, leaving animals with at least two encounters having a confirmed identity (Morrison et al., 2011). Subsequent analysis procedures involved this reduced data set.

For this dissertation, I used a subset of the 13-year data set consisting of one dry season, April-September 2008. Processing of these data for analysis consisted of converting encounter frequency data into binary capture histories (0, 1), with a GPS location associated with each encounter. For the data set used in this analysis, there were 316 encounter records from 135 individuals, with the number of records per individual ranging from 1 to 9.

1.4 Aims and objectives

Photographic collection of opportunistic data has the potential to provide a rich source of information about wildlife populations, particularly in places where there are limited resources or expertise for structured surveys. However, data

coming from unstructured sampling is useful only if statistical methods that accommodate that unstructured nature can be applied. Given this problem, my aim is to assess the reliability of SCR-based analysis to estimate population density from elephant photographic data from the APNR. Specifically, my objective is to investigate the effects of non-independent encounters that arise through spatially clustered observations of animals on bias and precision of density estimates derived from SCR methods.

1.5 Dissertation structure

This dissertation consists of three chapters and two appendices. The main data chapter, and main treatment of the research project, is in chapter 2. It is written in the format of a scientific journal article. Because it is intended to be published, it contains sufficient detail on the research project to be understandable outside this dissertation, and it contains an Introduction, Methods, Results, Discussion and References with details relevant to that project.

Chapter 1 is a general introduction to the dissertation. It contains information to develop the context of the problem presented in the main data chapter. This includes background on CR and SCR analysis methods, the problem of gregarious animals and non-independent data in the estimation of population density, and details data sources, methods of collection, and methods of data management. Chapters 3 presents the overall conclusions and their application to SCR estimation in wild populations. Two appendices contain the analysis scripts that I used to conduct the investigation in R (R Core Team, 2015) and JAGS (Plummer, 2003).

1.6 References

Auger-Méthé, M., Marcoux, M., Whitehead, H., 2010. Nicks and notches of the dorsal ridge: promising mark types for the photo-identification of narwhals. Marine Mammal Science 26, 663–678. doi:10.1111/j.1748-7692.2009.00369.x

Borchers, D., 2012. A non-technical overview of spatially explicit capture–recapture models. Journal of Ornithology 152, 435–444. doi:10.1007/s10336-010-0583-z

Borchers, D.L., Efford, M.G., 2008. Spatially explicit maximum likelihood methods for capture–recapture studies. Biometrics 64, 377–385. doi:10.1111/j.1541-0420.2007.00927.x

Buckland, S.T., Anderson, D.R., Burnham, K.P., Laake, J.L., Borchers, D.L., Thomas, L., 2001. Introduction to distance sampling: estimating abundance of biological populations. Oxford University Press, Oxford, United Kingdom.

Dawson, D.K., Efford, M.G., 2009. Bird population density estimated from acoustic signals. Journal of Applied Ecology 46, 1201–1209. doi:10.1111/j.1365-2664.2009.01731.x

Efford, M.G., 2004. Density estimation in live-trapping studies. Oikos 106, 598–610. doi:10.1111/j.0030-1299.2004.13043.x

Grange, S., Barnier, F., Duncan, P., Gaillard, J.-M., Valeix, M., Ncube, H., Périquet, S., Fritz, H., 2015. Demography of plains zebras (Equus quagga) under heavy predation. Population Ecology 57, 201–214.

doi:10.1007/s10144-014-0469-7

Kendall, W.L., Langtimm, C.A., Beck, C.A., Runge, M.C., 2004. Capture-recapture analysis for estimating manatee reproductive rates. Marine Mammal Science 20, 424–437. doi:10.1111/j.1748-7692.2004.tb01170.x

Lebreton, J., Burnham, K., Clobert, J., Anderson, D., 1992. Modeling survival and testing biological hypotheses using marked animals: a unified approach with case studies. Ecological Monographs 62, 67–118.

MacKenzie, D.I., Nichols, J.D., Sutton, N., Kawanishi, K., Bailey, L.L., 2005. Improving inferences in population studies of rare species that are detected imperfectly. Ecology 86, 1101–1113. doi:10.1890/04-1060

Marshal, J.P., 2017. Survival estimation of a cryptic antelope via photographic capture–recapture. African Journal of Ecology 55, in press. doi:10.1111/aje.12304

Morrison, T.A., Bolger, D.T., 2014. Connectivity and bottlenecks in a migratory wildebeest Connochaetes taurinus population. Oryx 48, 613–621. doi:10.1017/S0030605313000537

Morrison, T.A., Yoshizaki, J., Nichols, J.D., Bolger, D.T., 2011. Estimating survival in photographic capture-recapture studies: overcoming misidentification error. Methods in Ecology and Evolution 2, 454–463. doi:10.1111/j.2041-210X.2011.00106.x

Parmenter, R.R., Yates, T.L., Anderson, D.R., Burnham, K.P., Dunnum, J.L., Franklin, A.B., Friggens, M.T., Lubow, B.C., Miller, M., Olson, G.S., Parmenter, C.A., Pollard, J., Rexstad, E., Shenk, T.M., Stanley, T.R.,

White, G.C., 2003. Small-mammal density estimation: a field comparison of grid-based vs. web-based density estimators. Ecological Monographs 73, 1–26. doi:10.1890/0012-9615(2003)073[0001:SMDEAF]2.0.CO;2

Plummer, M., 2003. JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling, in: Hornik, K., Leisch, F., Zeileis, A. (Eds.), Proceedings of the 3rd International Workshop on Distributed Statistical Computing. Vienna, Austria.

Pradel, R., Hines, J.E., Lebreton, J.-D., Nichols, J.D., 1997. Capture-recapture survival models taking account of transients. Biometrics 53, 60–72. doi:10.2307/2533097

R Core Team, 2015. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Royle, J.A., Chandler, R.B., Sollmann, R., Gardner, B., 2014. Spatial capture-recapture. Academic Press, Waltham, USA.

Royle, J.A., Dorazio, R.M., 2008. Hierarchical modelling and inference in ecology. Academic Press, Waltham, USA.

Royle, J.A., Young, K.V., 2008. A hierarchical model for spatial capture-recapture data. Ecology 89, 2281–2289.

Russell, R.E., Royle, J.A., Desimone, R., Schwartz, M.K., Edwards, V.L., Pilgrim, K.P., Mckelvey, K.S., 2012. Estimating abundance of mountain lions from unstructured spatial sampling. Journal of Wildlife Management 76, 1551–1561. doi:10.1002/jwmg.412

White, G.C., Shenk, T.M., 2001. Population estimation with radio-marked

animals, in: Millspaugh, J.J., Marzluff, J.M. (Eds.), Radio Tracking and Animal Populations. Academic Press, Waltham, USA, pp. 329–350.

Wilson, K.R., Anderson, D.R., 1985. Evaluation of two density estimators of small mammal population size. Journal of Mammalogy 66, 13–21. doi:10.2307/1380951

CHAPTER TWO – GREGARIOUS ANIMALS AND NON-INDEPENDENT ENCOUNTERS IN SPATIAL CAPTURE-RECAPTURE ANALYSIS

2.1 Abstract

The gregarious nature of some animal species can affect estimation of population density when using methods where observations of individuals are assumed to be independent of each other, such as for spatial capture-recapture (SCR) analysis. In SCR locations of activity or home-range centres for individuals are assumed to show spatial randomness (e.g. homogeneous point process). I used data simulation, case-study data for African elephants (*Loxodonta africana*), and analysis methods based on SCR, to investigate the effect of non-independent observations on bias and precision of population density estimates. Data simulation demonstrated that methods that assume spatially random data produced estimates with negative bias that were likely too precise when locations were clustered. Attempting to correct with encounter heterogeneity models reduced precision but produced estimates that were nearly three times the true density. Bias was lowest when estimating density of groups and multiplying by average group size. Incomplete information on elephant herds precluded the use of the lowest-bias method on the case study data. Otherwise, estimates of elephant density were likely too low if groups were ignored or too high if models with encounter heterogeneity were used. In cases where group membership is incompletely known, more accurate density estimation might be achieved with methods that assume a cluster point process model for locations of activity centres.

Keywords: African elephant, animal aggregation, hierarchical Bayesian analysis,

individual heterogeneity, *Loxodonta africana*, point process model, social group, spatial capture-recapture

2.2 Introduction

Population estimation for several animal species can be complicated by their occurring in social aggregations, because observations of one individual are not independent of observations of others in the same group (Williams et al., 2002; Conroy and Carroll, 2009). The consequence is a potential effect on estimation of standard errors such that the level of the precision is an overestimate of the true precision (Williams et al., 2002). Moreover, aggregations can cause bias if individuals are detected more or less than if they were alone (Caughley, 1977; Sinclair et al., 2005). With highly aggregated animals (e.g. African buffalo, *Syncerus caffer*; African elephant, *Loxodonta africana*), researchers often forgo sampling in favour of total counts, because the large animal aggregations separated by large distances produce estimates with high variability and large standard errors (Sinclair et al., 2005). Researchers aware of the effect of aggregation on estimate accuracy can compensate by, among other things, adjusting the standard errors of the estimates through adjustments for serial correlation (Ramsey and Schafer, 2002), or by accommodating non-independence explicitly in model structure used for analysis (e.g. random effects in a generalized linear mixed model; Breslow and Clayton, 1993; Natarajan and McCulloch, 1999; Burnham and White, 2002; Gillies et al., 2006). Because varying degrees of aggregation within a population can influence an individual's encounter probability in a capture-recapture context, use of encounter models that

incorporate individual heterogeneity (Dorazio and Royle, 2003; Royle, 2008; Abadi et al., 2013) might account for differences in encounter that arise as a consequence of living in groups. The effect of aggregation on several methods of estimation and the best approaches to deal with such non-independence, however, remain poorly understood.

Among the approaches requiring a careful assessment of the effect of aggregation on estimation is spatial capture-recapture (SCR). This is an increasingly common approach to estimate density of wildlife populations, where standard encounter history data from individually-recognizable study animals are combined with spatial information from locations of encounter (Efford, 2004; Borchers and Efford, 2008; Royle and Young, 2008). The approach assumes that each individual has an activity centre (e.g. an average location or home-range centre) that might or might not have some biological relevance (e.g. a den or nesting site). The SCR approach uses distance to activity centre as an encounter-specific covariate, such that as distance increases, probability of encounter decreases. Because activity centres are not known (i.e. latent variables), they are treated as random effects and removed from the distribution of observations with integrated likelihood in a classical frequentist framework (Efford, 2004; Borchers and Efford, 2008), or assigned a prior distribution and estimated by Markov chain Monte Carlo (MCMC) simulation in a Bayesian framework (Royle and Young, 2008; Royle and Dorazio, 2008). Estimation of density is then an estimation of the number of activity centres within an area of interest. An important assumption of this method is that the activity centres can be described as a Poisson spatial

point process (i.e. that activity centres are randomly distributed and independent of each other; Royle et al., 2014). This assumption is violated in cases where animals aggregate.

If information about animal aggregations is collected in the field along with usual encounter information, then the effect of aggregations on density estimation can be assessed. But this is not always possible. If encounters are collected by camera traps or DNA sampling, the sampling mechanism might not record group members. This could also be a problem with data collected opportunistically by volunteers (i.e. citizen scientists), where information associated with each observation might be incomplete. Datasets collected by volunteers are increasingly common and increasingly easy for non-scientists to collect, raising the potential to provide information about wildlife populations in resource-poor regions that might otherwise go unstudied (Roberts et al., 2007; Kéry et al., 2010). Sampling of animals by photographic methods, where geographic coordinates for each observation are also recorded, can be combined with animal characteristics that provide individual recognition (e.g. pellage patterns, scars) to generate spatially-explicit encounters of known individuals (Royle et al., 2009; Auger-Méthé et al., 2010; Morrison and Bolger, 2014). Such data collected opportunistically, however, might lack the completeness associated with a formal study design, thus making data quality unknown.

Unknown effects of gregariousness and non-independent observations on SCR estimation, and the growing availability of data of uncertain quality gathered via opportunistic methods, make development of statistical methods to

accommodate uncertain quality increasingly necessary. My objective was to investigate the effect of aggregated observations on bias and precision of density estimates generated from SCR models. I applied SCR analysis methods in a case study to data collected opportunistically from a population of African elephants where aggregations were known to occur but were inconsistently recorded. I then assessed the effects of aggregation on density estimation by simulating encounter data with known aggregation patterns. I used simulated data to compare analysis methods that accommodated aggregation in different ways: ignoring aggregation altogether, estimating density of aggregations and multiplying by aggregation size, and incorporating random effects in encounter heterogeneity models. I demonstrate that bias and precision of density estimates from simulated data, and thus the reliability of density estimates from field data, depend on the manner in which non-independence is accommodated in SCR models.

2.3 Methods

In this dissertation, I adopt a hierarchical Bayesian modelling approach because of its intuitive separation of an overall analysis model into observation and process models (i.e. state-space formulation) and because of its flexibility in accommodating alternative model structures to accomplish this separation (Royle and Dorazio, 2008; (Kéry and Schaub, 2012); Royle et al., 2014). The data design assumes that an individual *i* encounters one or more locations or detectors (indexed by *j*) over a sampling period lasting *K* occasions; thus, the encounter history for each individual consists of a matrix of each time and location of encounter (Royle et al., 2014). I use the term “detector” for any device by which

encounters are recorded (e.g. camera trap, DNA sampling device).

2.3.1 Process model

The ecological process component of the analysis models begins with a distribution of activity centres $\boldsymbol{s}_i$ across the study area (S). Activity centres are two-dimensional coordinates for the activity centre of individual i, which are unobservable and treated as random effects. Within the Bayesian framework, random effects are assigned a prior distribution. The basic SCR model assumes a uniform distribution of activity centres:

$$\boldsymbol{s}_i \sim \text{Uniform}(S) \quad (1)$$

with S defined to be large enough to be uninformative (Royle and Young, 2008). This prior distribution assumes spatial randomness of activity centres, which is likely to be biologically unrealistic, but it might provide a reasonable approximation of activity centres over relatively short sampling periods or in situations with sparse data (Royle et al., 2011b).

2.3.2 Observation model

If an encounter of individual i at location j on occasion k is represented by $y_{ijk} = 1$, the y_{ijk} are distributed as independent Bernoulli trials,

$$y_{ijk}|\boldsymbol{s}_i \sim \text{Bernoulli}(p_{ijk}) \quad (2)$$

where p_{ijk} is the probability of encounter specific to individual i, location j and occasion k. When encounters are combined across K occasions, the resultant counts are distributed

$$y_{ij}|\boldsymbol{s}_i \sim \text{Binomial}(K, p_{ij}) \quad (3)$$

The observation model (eqn 3) is conditional on the process model (eqn 1) that

represents the ecological process (i.e. animal abundance). Let $\boldsymbol{x}_j$ be a two-dimensional coordinate for detector j. Encounter probability p_{ij} can be represented as a function of the Euclidean distance $||\boldsymbol{x}_j - \boldsymbol{s}_i||$ from each detector to each activity centre. Based on a half-normal model,

$$p_{ij} = p_0 \exp(-\alpha_1 ||\boldsymbol{x}_j - \boldsymbol{s}_i||^2) \quad (4)$$

where p_0 is the encounter probability at the activity centre and $\alpha_1 = 1/(2\sigma^2)$ is a scale parameter that describes how quickly probability of encounter decreases with distance from activity centre (Royle et al., 2014).

2.3.3 Estimating N

Estimation of population parameters in a Bayesian framework often involves data augmentation (Royle and Young, 2008; Royle, 2009; Gardner et al., 2009). Abundance estimation by SCR methods depends on inference from encounter histories for observed individuals, but also on accounting for an unknown number of individuals that are never encountered. Royle et al. (2007) address this "unknown index" problem by adding a large number of all-zero encounter histories to an encounter data set, reparameterizing the model of encounter from a multinomial to a zero-inflated binomial model, and estimating a binomial parameter that represents the probability that an all-zero encounter history is an unobserved individual in the population.

Thus, M is defined as the number of individuals encountered during the study (n) plus the augmented capture histories ($M - n$), some of which represent animals in the population available to be encountered but were not (Royle et al., 2014). Estimation of the abundance parameter (N) then consists of estimating ψ,

the probability that one of the M encounter histories represents an individual in the sampled population, based on the expectation $E[N] = \psi M$. Let z_i be a latent variable where $z_i = 1$ for an encounter history that represents a member of the population and $z_i = 0$ otherwise. An estimate of ψ is $\Pr(z_i = 1)$, and can be incorporated into the SCR model as $y_{ij}|z_i \sim \text{Binomial}(K, z_i\, p_{ij})$. Abundance becomes a derived parameter estimated by ψ and z_i:

$$N = \sum_{i=1}^{M} z_i \qquad (5)$$

This quantity represents the estimated number of activity centres within the study area S, and thus, an estimation of the number of individuals in that area (Royle and Young, 2008). Density D depends on the estimate of the population size and the area represented in the study by $D = N/\text{area}(S)$. The researcher defines the size of S; however, the density estimated by this approach should be invariant to study area size if S is chosen to be large enough, which can be confirmed through trial analyses (Royle et al., 2011b). Similarly, M should be substantially larger than the true population size, but otherwise it has little effect on resulting estimates (Royle et al., 2007).

Thus the basic SCR model (SCR0), based on a half-normal encounter probability function and data augmentation, consists of four components (Royle et al., 2014):

- ecological process: $\boldsymbol{s}_i \sim \text{Uniform}(S)$,
- observation process: $y_{ij}|\boldsymbol{s}_i \sim \text{Binomial}(K, z_i\, p_{ij})$,
- encounter probability: $p_{ij} = p_0 \exp(-\alpha_1 ||\boldsymbol{x}_j - \boldsymbol{s}_i||^2)$, where $\alpha_1 = 1/(2\sigma^2)$, and

- zero-inflation model to estimate N: $z_i \sim$ Bernoulli(ψ).

2.3.4 Case study

African elephants are recognized as occurring in aggregations (Estes, 1991). I applied SCR models for data from aggregated individuals sampled photographically at the Associated Private Nature Reserve (APNR) and neighbouring Kruger National Park, South Africa. Encounters were collected 1 April to 30 September 2008 by unstructured search-encounter methods (Russell et al., 2012; Thompson et al., 2012; Royle et al., 2014), and the data set consisted of encounter date, time and geographic coordinates for each encounter. To analyse these encounters, I overlaid the study area with grid of cells (5 × 5 km) and I assumed that the centre of each grid cell contained a detector so that the locations of encounters were the coordinates of each cell centroid (Russell et al., 2012; Thompson et al., 2012; Royle et al., 2014). I assessed the sensitivity of my choice of grid cell size by experimenting with a range of sizes (2, 4, 8 km) to investigate the effect on the resultant density estimate (Russell et al., 2012).

This data collection scenario produced a three-dimensional array of encounters for n = 135 individuals at J = 19 encounter locations over K = 83 sampling occasions, with locations having covariates in the form of (X, Y) coordinates. The area of observation consisted of a region 5 × 6 grid cells (25 × 30 km), to which I added a buffer. I experimented with buffer sizes ranging from 1 to 4 grid cells (5–20 km), and I chose the smallest buffer where additional increase did not change the resulting density estimate (Russell et al., 2012).

Density estimation occurred by way of a zero-inflated SCR model (Royle

and Young, 2008). To represent potential unobserved individuals, I added sufficient all-zero capture histories to bring the total histories to $M = 1000$, a value chosen to be large enough not to constrain estimates of N. I used eqns 1–5 as statistical models for analysis; however, I used a value for K in eqn 3 that varied by the number of sampling occasions at each encounter location (Royle et al., 2011b). Thus I modified eqn 3 to be $y_{ij}|\mathbf{s}_i \sim \text{Binomial}(K_j, z_i\, p_{ij})$. In addition to the basic SCR model (SCR0), I investigated three additional models. The first of these included the distance to the volunteers' research base during data collection (SCRd). I modified eqn 4 by replacing p_0 with p_{0j}, and calculating p_{0j} by

$$\text{logit}(p_{0j}) = \beta_0 + \beta_1 \times \text{distance}_j \qquad (6)$$

where distance_j was the Euclidean distance from the research base to each cell's centre point, β_0 and β_1 were regression coefficients to be estimated with prior distributions Normal(0, 0.01), and 0.01 was the value chosen for prior precision ($1/\sigma^2$) which is the convention in Bayesian analysis. Prior precision was chosen to be non-informative; however, in some cases extreme values for the priors can be generated during analysis that produce invalid values for $\text{logit}(p_{0j})$ (e.g. numbers that approach $+/-\infty$). To avoid such values a truncation can be chosen to omit such extreme values in the prior distributions. I set such a truncation for the prior distributions of β_0 and β_1 at (–10, 10). Thus, using truncation allowed for a balance between maintaining a non-informative prior distribution and producing extreme values in the prior distributions that resulted in invalid values for $\text{logit}(p_{0j})$.

The second model was an individual heterogeneity model without

covariates, to account for potential differences in detection probability based on animal sex, age and herd size (SCRh). In this case, I modified eqn 4 by replacing p_0 with p_{0i}, and treating p_{0i} as an individual random effect by

$$\text{logit}(p_{0i}) = \alpha_0 + \varepsilon_i \quad (7)$$

where α_0 was an estimate for each level (i.e. individual) of the random effect with prior distribution Normal(0, 0.01) also truncated at (-10, 10), and ε_i was distributed as Normal(0, τ_ε) with $\tau_\varepsilon = 1/\sigma_\varepsilon^2$ and $\sigma_\varepsilon \sim$ Uniform(0, 10). Again, the value for prior precision was chosen to be non-informative, which sometimes differed depending on the parameter being estimated.

The third model (SCRdh) included both random effects and the distance covariate:

$$\text{logit}(p_{0i,j}) = \alpha_0 + \beta_1 \times \text{distance}_j + \varepsilon_i \quad (8)$$

where β_1 was as in eqn 6, and α_0 and ε_i were as in eqn 7. Analysis by MCMC simulation was conducted in JAGS (Plummer, 2003) via R (R Core Team, 2015) by using runjags (Denwood, 2016), with three chains of 100 000 iterations following 100 000 burn-in iterations. To reduce the effects of autocorrelation, I used one value for every 100 iterations; thus, posterior distributions were based on a total of 3000 iterations. I assessed convergence using the Brooks-Gelman-Rubin diagnostic ($\hat{R} < 1.1$; Gelman and Shirley, 2011). The analysis required one or more weeks to complete on a four-core desktop computer running the MCMC chains in parallel. Scripts for R and JAGS programming are in Appendices 1–3.

2.3.5 Simulation study

To assess the reliability of estimates generated for the case study, I used data simulation to compare bias and precision of four SCR models. The data simulation represented aspects of the sampling scenario for the APNR elephant encounter data. I used 10 sampling occasions repeatedly to generate 50 datasets. To mimic the 25-km^2 grid cells from the case study, the simulated state-space consisted of a grid, five cells by six cells within which encounters could occur, and it was buffered on all sides by three units. I assumed a sampling scenario where the centre of each grid cell contained a detector so that the locations of encounters were the coordinates of each cell centroid (Russell et al., 2012; Thompson et al., 2012). Within this state-space I simulated activity centres for aggregations that were located according to a uniform distribution, a distribution that is sufficiently flexible to represent moderate degrees of clustering over a short period of time (e.g. during a wildlife survey; Royle et al., 2014). I simulated a total population of 450 individuals, distributed across approximately 40 aggregations ranging in size from 1 to 30 individuals, with average aggregation size being approximately 10 individuals. I simulated encounters with aggregations with a Bernoulli distribution, and I calculated encounter probability with eqn 4 (p_0 = 0.08 and σ = 1.3). The values in the simulation study were chosen to be similar to values estimated or observed in the case study. Encounters with individuals, conditional on encounter with its aggregation, were also calculated as Bernoulli trials with individual encounter probability equal to the reciprocal of the number of individuals in an aggregation.

The first model (SCR0) was based on a half-normal encounter probability

model (eqns 1–5), and I assumed that each encounter with an individual was independent of those with other individuals even if they were in the same group. The second model (SCRa) used the same set of equations (1–5), but estimated density of group and then multiplied by average group size to calculate density of individuals (Russell et al., 2012). The third model (SCRi) began with the SCR0 model, added the modification in eqn 7, and used individual identity as a random effect to produce an individual heterogeneity model. The fourth model (SCRg) assumed that each individual had a known group membership. I assigned a prior distribution for unobserved group membership by using a categorical distribution where the number of categories was the number of groups in the simulated population, and the probability of membership to a particular group was the population-level proportion of individuals in that aggregation. With that prior, I modified the encounter probability model so that

$$\text{logit}(p_{0i}) = \gamma_0 + \gamma_{1i} \quad (9)$$

where γ_{1i} were coefficients associated with herd membership. Priors for model SCRg were chosen to be non-informative: $\gamma_0 \sim \text{Normal}(0, 0.01)$, $\gamma_{1i} \sim \text{Normal}(0, \tau_{\gamma 1})$ with $\tau_{\gamma 1} = 1/(\sigma^2)$ and $\sigma^2 \sim \text{Uniform}(0, 10)$.

I conducted Bayesian analyses in JAGS (Plummer, 2003) through R (R Core Team, 2015) by using runjags (Denwood, 2016). I estimated posterior distributions of population density using MCMC. Numbers of iterations varied depending on model complexity; all models used three chains, and iterations ranged up to 150 000 following up to 50 000 burn-in iterations. I thinned to one in every 30–100 iterations for estimation, with the thinning rate chosen to reduce

the effects of autocorrelation among posterior estimates. Simulations and analysis took several days or weeks to complete, depending on model complexity and computing resources at my disposal. I judged that the chains had converged when $\hat{R} < 1.1$. Scripts for R and JAGS programming are in Appendices 4–7.

I compared relative bias in density estimates between estimation methods using

$$Bias(D) = \frac{\frac{1}{B}\sum_{b=1}^{B} D_b - D}{D} \quad ,$$

where D_b was the estimated density for the b^{th} simulated data set, D was the true density used in simulation, and B was the number of simulated data sets (Abadi et al., 2013). I compared the precision between estimation methods with the standard deviation (*SD*) of the posterior distribution for D_b. For fitted models from the case study and the data simulation, I calculated goodness of fit (GOF) statistics in the form of Bayesian *P*-values (Gelman et al., 1996), which is a measure of discrepancy between observed numbers of encounters and numbers of encounters predicted by a model. Because there is still no fixed approach to assessing GOF for SCR models, I used three variations on Bayesian *P*-value assessment suggested by Royle et al. (2014). The first aggregated counts over occasions and compared expected and observed frequencies across individuals and detectors, which might be appropriate if the model has a time-varying structure:

$$T_1(\boldsymbol{y}, \vartheta) = \sum_i \sum_j \left(\sqrt{y_{ij\cdot}} - \sqrt{E(y_{ij\cdot})}\right)^2$$

where $y_{ij\cdot} = \Sigma_k\, y_{ijk}$ and $E(y_{ij\cdot}) = \Sigma_k\, p_{ijk}$. The second assessed frequencies by

individual alone, which might be useful for assessing whether individual heterogeneity is adequately accommodated by a model:

$$T_2(\boldsymbol{y},\vartheta)=\sum_i\left(\sqrt{y_{i..}}-\sqrt{E\left(y_{i..}\right)}\right)^2$$

where $y_{i..} = \Sigma_j \Sigma_k y_{ijk}$ *and* $E(y_{i..}) = \Sigma_i \Sigma_j p_{ijk}$. The third is based on frequencies by detector alone, which should provide a useful assessment of spatial observations:

$$T_3(\boldsymbol{y},\vartheta)=\sum_j\left(\sqrt{y_{.j.}}-\sqrt{E\left(y_{.j.}\right)}\right)^2$$

where $y_{.j.} = \Sigma_i \Sigma_k y_{ijk}$ and $E(y_{.j.}) = \Sigma_i \Sigma_k p_{ijk}$. For each assessment, an adequate-fitting model was indicated by Bayesian *P*-values within the range 0.1–0.9 (Royle et al., 2014).

2.4 Results

2.4.1 Case study

To analyse the African elephant data from APNR, I began by varying the width by which to buffer the state-space. Beginning with 0 km, I increased the size of the buffer by 5-km increments and re-analysed the encounter data using the SCR0 model. The density estimate stabilized at 0.14 elephants/km^2 at a buffer of 15 km. This produced a 55 × 60-km state-space having a total area of 3300 km^2. Density estimate was robust to choice of cell size. With the state-space buffer set at 15 km, estimated 95% Bayesian credible intervals (CRIs) ranged from (0.13, 0.22) for 2-km cells, to (0.14, 0.23) for 8-km cells. I conducted subsequent analyses with a 5-km cell size, which was judged to provide a reasonable compromise between spatial resolution and computational efficiency.

With the extent and resolution of the state-space established, I proceeded to estimate the density of elephants. Assuming no herd structure (SCR0), the

estimate was 0.1382 animals/km^2 (95% CRI: 0.1027, 0.1770). Goodness of fit assessment suggested a model potentially with poor fit. Assessment of discrepancy across individuals and detectors produced low Bayesian *P*-values (*P*-value = 0.04). Discrepancy across individuals only (*P*-value = 0.39) or detectors only (*P*-value = 0.25) suggested reasonable fit of models to encounter data (Table 2.1). With the addition of the distance covariate (model SCRd), the density estimate (0.1379 elephants/km^2; 95% CRI: 0.1024, 0.1788) changed little and there was marginal improvement of the GOF of the model (*P*-value > 0.08) (Table 2.1). Moreover, the estimate for the parameter associated with distance to research base overlapped with zero (β_1 = 0.14; 95% CRI: –0.02, 0.31) suggesting no evidence of an effect of distance from research base on detection probability.

When I included individual heterogeneity into the model (SCRh), estimate of density increased to 0.2077 elephants/km^2 (95% CRI: 0.1342, 0.2921) and GOF showed further improvement (*P*-value > 0.11) (Table 2.1). With both individual heterogeneity and the distance covariate (SCRdh), density estimate changed little (0.2010 elephants/km^2; 95% CRI: 0.1330, 0.2842), but there was further minor improvement in GOF (*P*-value > 0.15) (Table 2.1). Similarly for SCRd, there was no evidence of an effect of distance to research base on encounter probability (β_1 = 0.12; 95% CRI: –0.07, 0.29). Lack of an effect of distance was further supported by the relatively small change in σ_ϵ (1.06 for SCRh to 0.99 for SCRdh) which indicated that little variability in detection was accounted for by the addition of the distance covariate. Thus, of the models I compared, the SCRh model appeared to be best-supported by the elephant encounter data, in that it had

both adequate GOF to the encounter data and parameter posterior distributions that did not overlap substantially with zero.

2.4.2 Simulation study

Data simulation demonstrated substantial differences in relative bias between models. If I ignored group structure, the average relative bias was –0.43 (Fig. 2.1a). Thus, with a true density of 0.14 animals/km^2, the SCR0 estimated a density of 0.08 animals/km^2 on average (Table 2.2). The range in relative bias estimates, however, was relatively narrow compared to other models I tested, with most relative bias estimates falling between –0.8 and 0 (Fig. 2.1a). Thus, the degree of relative bias was somewhat consistent and relatively narrow across simulated data sets.

The three approaches to accommodate aggregations produced very different outcomes. The SCRa model produced a small average relative bias (<0.01); thus, true and estimated densities were close to each other (Fig. 2.1b). Moreover, the relative bias estimates clustered tightly around zero (Fig. 2.1b), indicating that low-bias estimates were consistently estimated across simulated data sets. Using the group-heterogeneity model (SCRg), however, produced a relatively strong positive relative bias (0.53) on average (Fig. 2.1c). Thus, compared to a true density of 0.14 animals/km^2, estimated density was 0.21 on average (Table 2.2). The degree of relative bias for the SCRg model changed considerably between data sets, with estimates ranging from –0.27 to 1.03. The individual-heterogeneity model (SCRi) showed the poorest performance of the models I compared. The average relative bias was the largest of the four models

(1.59), such that model density was an overestimate of true density by 0.21 animals/km^2 (Table 2.2). Furthermore, the range in relative bias (0.21–3.70) was also the largest of the four models and the range across datasets was consistently positive (Fig 2.1d).

Estimated precision of density estimates decreased with increasingly sophisticated models (Fig. 2.2). I found the highest estimated precision was for model SCR0 ($\bar{SD}$ = 0.018), which decreased for models SCRa ($\bar{SD}$ = 0.024), SCRg ($\bar{SD}$ = 0.042) and then substantially for SCRi ($\bar{SD}$ = 0.130) (Table 2.2). However, the ranges of *SD* estimates were similar among estimation methods, which the exception of SCRi, where the range in *SD* estimates was noticeably wider than for the other model (Fig. 2.2d). Furthermore, the distribution of *SD* estimates overlapped noticeably among methods, again with the exception of those for SCRi, which overlapped only with the *SD* distribution for SCRg but not the other two models. It is noteworthy that all models that accommodated groups produced estimates with lower precision on average, as might be expected when non-independence within group encounters is included in the model structure.

2.5 Discussion

Several lessons are apparent from the outcomes from the preceding analyses. First, ignoring aggregation structure and using the SCR0 model for density estimation likely produce underestimates of true density that are too precise. Second, accounting for aggregation structure in the observation process with a heterogeneity encounter model, either with group (SCRg) or individual (SCRi)

heterogeneity, likely produces an overestimate of true density. Precision of those estimates, however, are likely more realistic. Third, if group membership of individuals can be accurately recorded, density estimation of aggregations, and multiplying by aggregation size (SCRa), produces estimates with relatively low bias. Finally, because the aggregation structure in the APNR elephant population was imperfectly known (precluding use of the SCRa model), the density estimates produced in the case study are likely biased low (in the case of models SCR0 and SCRd) or high (in the case of models SCRh and SCRdh).

2.5.1 Goodness of fit: encounter model

An exploration of model assumptions, and an assessment of agreement between model assumptions and data, is a necessary part of any statistical investigation. The GOF assessment conducted here, however, likely provided only a partial picture of model fit to the data and investigation of assumptions. Because an SCR model is hierarchical, overall GOF depends on the fit of its separate components: the observation model to the observed encounter data and the point process model to the underlying ecological process (Royle et al., 2011a; 2014).

The fit of the observation model is a function of the discrepancy between the observed encounters and the expected encounters based on the chosen encounter model (Royle et al., 2014). For both the case and simulation studies, I modelled encounters with a half-normal function that represented a decrease in encounter probability with distance from the activity centre. Encounter probability could also be a function of covariates, such as distance from a central research base in the case study analysis. I also considered models that

accommodated heterogeneity at the aggregation (SCRg) and individual (SCRh, SCRdh, SCRi) levels, on the presumption that differences in encounter probability might vary among individuals or groups as a function of group membership and group characteristics. Based on the three discrepancy measures I used, adequate fit of the encounter data was marginally achieved with the individual heterogeneity model, likely because the effect of group size or membership on individual encounter could be absorbed by a random effect in the heterogeneity model. Fit was improved somewhat further in the case study analysis with the distance-to-research-base covariate, but its 95% CRI contained zero.

The assessment of encounter model fit requires aggregating binary encounter data from a three-dimensional encounter array to produce counts of encounters (Royle et al., 2014). Because there is no established method to assess fit of SCR models (Royle et al., 2011a), different approaches to aggregation over one or two dimensions can focus an assessment on variation in encounters across detectors, occasions and individuals, giving different views of model fit. For example, aggregating over detectors and occasions provides a summary of encounters per individual that would allow for an assessment of model fit and individual heterogeneity. Royle et al. (2014) argue that such an summary of encounters would provide a test of extra-binomial variation as a consequence of variable encounter probability among individuals, something strongly affected by aggregations in social species. For all assessments, I summarized encounters across time because I did not expect changes in animal behavioural that would affect encounter probability within the sampling period.

2.5.2 Goodness of fit: point process model

The second component of model fit depends on the degree to which the underlying ecological process matches the process model (Royle et al., 2011a; Royle et al., 2014). At the core of the SCR analysis is a spatial point process model that represents the activity centres of the encountered individuals (Efford, 2004; Borchers and Efford, 2008; Royle and Young, 2008). For a standard SCR model (i.e. SCR0), the point process is assumed to show spatial randomness (i.e. point locations are independent and distributed according to a uniform distribution; Royle et al., 2011a; Royle et al., 2014). The uniform distribution is sufficiently flexible to accommodate moderate degrees of aggregation under most circumstances (Royle et al., 2014), but the degree of aggregation demonstrated by highly social animals could be too extreme for the spatial randomness assumption.

In principle, performing a GOF assessment for the process model is relatively straight-forward. The approach involves dividing the study area into bins, and at each MCMC iteration, calculating counts of estimated activity centres in each bin and comparing to bin counts under the spatial randomness assumption (Royle et al., 2014). The assessment, however, is highly sensitive to bin size and to the extent of the state-space (Illian et al., 2008; Royle et al., 2014). Given the computational challenges that I had already encountered in this study, assessment of process model GOF and exploration of bin and extent size was excluded from the present analysis.

Although an SCR process model with spatial randomness should provide a reasonable short-term approximation of the distribution of individual activity centres, more extreme situations such as highly social or territorial animals, might

require an alternative process model to produce accurate estimates of animal density. For example, density of activity centres could be modelled as a function of environmental covariates, using an inhomogenous point process model (Borchers and Efford, 2008; Efford et al., 2009). For species that have territories, locations of activity centres are generally more evenly distributed than are uniformly distributed locations; that is, the average distance between activity centres is substantially larger than the standard deviation in distances, compared to a uniform distribution where the average and standard deviation are more similar. To model activity centres for territorial field voles (*Microtus agrestis*), Reich and Gardner (2014) modelled activity centres with a Strauss process (Strauss, 1975), which incorporates a parameter that represents the strength of repulsion between activity centres.

For gregarious species, a spatial cluster process model might be more appropriate for representing grouped individuals (Borchers and Efford, 2008; Royle et al., 2014). The Neyman-Scott process is a commonly used method to represent a spatial cluster process (Illian et al., 2008). Parent points are uniformly distributed in the state-space, and each parent is then replaced by a cluster of daughter points with the degree of clustering defined by model parameters. Each cluster is uniformly distributed and independent of other clusters but points within clusters are highly aggregated, a pattern that might be more representative for gregarious species such as elephants.

Of the models considered in the simulation study, the SCRa model would most closely accommodate a spatial cluster process. This is also likely to be why

estimates based on the SCRa approach produced the lowest bias on average and had the narrowest range of bias estimates of the approaches used in the simulation study. Estimates were based on density of simulated herds and assumed known herd sizes. Thus, herd locations met the assumptions of spatial randomness, despite the locations of individuals being highly aggregated. The disadvantage of this model for real data, particularly data collected opportunistically, is a requirement for complete knowledge of aggregation size and membership. For species that occur in small groups, group membership is relatively easy to assess. Russell et al. (2012) used SCR to estimate density of mountain lions (*Puma concolor*) in the Blackfoot Mountains, USA, by treating each individual as an independent observation and by estimating density of family groups and average group size. Estimates based on family groups were low relative to density estimates from other models, and they suspect they missed some family groups when surveying animals in their study area. Compared to elephants in the APNR, observed group sizes in the Blackfoot Mountains were at most two individuals, making it easier to account for all individuals in a group.

2.5.3 Case study

Depending on the model, SCR analysis produced estimates in the range of 0.14–0.21 animals/km^2 (Table 2.1). For the analyses that did not attempt to account for aggregations, the simulation study suggests that those estimates might be negatively biased. For the analyses that included individual heterogeneity, in contrast, modelled density estimates could be higher than true density. The SCR models I used included a covariate to account for the likely search intensity of the

study area and individual-level heterogeneity in the detection probability model. The addition of individual heterogeneity at least partially compensated for irregularities in encounters associated with opportunistic collection. However, there was no evidence from the analysis that encounters occurred closer to the camp where researchers were based. Moreover, encounter probability might be expected to vary with sex and age of an animal, and with the size of the group it is associated with. Unfortunately, these attributes were inconsistently recorded in the field. Including individual heterogeneity, however, assisted with accounting for the variability in encounter detection associated with the unique attributes of each animal. Although an individual heterogeneity model is a useful ad hoc approach to deal with variation in encounter probability, a planned sampling design that accounts for expected variability among different segments of a population will produce stronger inferences about population parameters (Abadi et al., 2013). Without a representative process model, however, density estimates from SCR models that assume spatial randomness can be expected to show substantial bias.

2.6 References

Abadi, F., Botha, A., Altwegg, R., 2013. Revisiting the effect of capture heterogeneity on survival estimates in capture-mark-recapture studies: does it matter? PLoS ONE 8, e62636. doi:10.1371/journal.pone.0062636

Auger-Méthé, M., Marcoux, M., Whitehead, H., 2010. Nicks and notches of the dorsal ridge: promising mark types for the photo-identification of narwhals. Marine Mammal Science 26, 663–678. doi:10.1111/j.1748-7692.2009.00369.x

Borchers, D.L., Efford, M.G., 2008. Spatially explicit maximum likelihood methods for capture–recapture studies. Biometrics 64, 377–385. doi:10.1111/j.1541-0420.2007.00927.x

Breslow, N.E., Clayton, D.G., 1993. Approximate inference in generalized linear mixed models. Journal of the American Statistical Association 88, 9–25. doi:10.2307/2290687

Buckland, S.T., Anderson, D.R., Burnham, K.P., Laake, J.L., Borchers, D.L., Thomas, L., 2001. Introduction to distance sampling: estimating abundance of biological populations. Oxford University Press, Oxford, United Kingdom.

Burnham, K.P., White, G.C., 2002. Evaluation of some random effects methodology applicable to bird ringing data. Journal of Applied Statistics 29, 245–264. doi:10.1080/02664760120108755

Caughley, G., 1977. Analysis of vertebrate populations. John Wiley & Sons, Hoboken, USA.

Conroy, M.J., Carroll, J.P., 2009. Quantitative conservation of vertebrates. Wiley-Blackwell, Chichester, United Kingdom.

Denwood, M.J., 2016. runjags: an R package providing interface utilities, model templates, parallel computing methods and additional distributions for MCMC models in JAGS. Journal of Statistical Software 71, 1–25.

Dorazio, R.M., Royle, J.A., 2003. Mixture models for estimating the size of a closed population when capture rates vary among individuals. Biometrics 59, 351–364.

Efford, M.G., 2004. Density estimation in live-trapping studies. Oikos 106, 598–610. doi:10.1111/j.0030-1299.2004.13043.x

Efford, M.G., Dawson, D.K., Borchers, D.L., 2009. Population density estimated from locations of individuals on a passive detector array. Ecology 90, 2676–2682. doi:10.1890/08-1735.1

Estes, R.D., 1991. The behaviour guide to African mammals. University of California Press, Berkeley, USA.

Gardner, B., Royle, J.A., Wegan, M.T., 2009. Hierarchical models for estimating density from DNA mark–recapture studies. Ecology 90, 1106–1115. doi:10.1890/07-2112.1

Gelman, A., Meng, X.-L., Stern, H., 1996. Posterior predictive assessment of model fitness via realized discrepancies. Statistica Sinica 6, 733–807.

Gelman, A., Shirley, K., 2011. Inference from simulations and monitoring convergence, in: Brooks, S., Gelman, A., Jones, G.L., Meng, X.-L. (Eds.), Handbook of Markov Chain Monte Carlo. Chapman & Hall/CRC, Boca

Raton, Florida, USA, pp. 163–174.

Gillies, C., Hebblewhite, M., Nielsen, S., Krawchuk, M., Aldridge, C., Friar, J., Saher, D., Stevens, C., Jerde, C., 2006. Application of random effects to the study of resource selection by animals. Journal of Animal Ecology 75, 887–898.

Illian, J., Penttinen, A., Stoyan, H., Stoyan, D., 2008. Statistical analysis and modelling of spatial point patterns. John Wiley & Sons, Chichester, United Kingdom.

Kéry, M., Royle, J.A., Schmid, H., Schaub, M., Volet, B., Häfliger, G., Zbinden, N., 2010. Site-occupancy distribution modeling to correct population-trend estimates derived from opportunistic observations. Conservation Biology 24, 1388–1397. doi:10.1111/j.1523-1739.2010.01479.x

Kéry, M., Schaub, M., 2012. Bayesian population analysis using WinBUGS: a hierarchical perspective. Academic Press, Waltham, USA.

Morrison, T.A., Bolger, D.T., 2014. Connectivity and bottlenecks in a migratory wildebeest *Connochaetes taurinus* population. Oryx 48, 613–621. doi:10.1017/S0030605313000537

Natarajan, R., McCulloch, C.E., 1999. Modeling heterogeneity in nest survival data. Biometrics 55, 553–559. doi:10.1111/j.0006-341X.1999.00553.x

Plummer, M., 2003. JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling, in: Hornik, K., Leisch, F., Zeileis, A. (Eds.), Proceedings of the 3rd International Workshop on Distributed Statistical Computing. Vienna, Austria.

R Core Team, 2015. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria.

Ramsey, F.L., Schafer, D.W., 2002. The statistical sleuth: a course in methods of data analysis, Second. ed. Duxbury, Pacific Grove, USA.

Reich, B.J., Gardner, B., 2014. A spatial capture-recapture model for territorial species. Environmetrics 25, 630–637. doi:10.1002/env.2317

Roberts, R.L., Donald, P.F., Green, R.E., 2007. Using simple species lists to monitor trends in animal populations: new methods and a comparison with independent data. Animal Conservation 10, 332–339. doi:10.1111/j.1469-1795.2007.00117.x

Royle, J.A., 2009. Analysis of capture-recapture models with individual covariates using data augmentation. Biometrics 65, 267–274.

Royle, J.A., 2008. Modeling individual effects in the Cormack-Jolly-Seber model: a state-space formulation. Biometrics 64, 364–370.

Royle, J.A., Chandler, R.B., Sollmann, R., Gardner, B., 2014. Spatial capture-recapture. Academic Press, Waltham, USA.

Royle, J.A., Dorazio, R.M., 2008. Hierarchical modelling and inference in ecology. Academic Press, Waltham, USA.

Royle, J.A., Dorazio, R.M., Link, W.A., 2007. Analysis of multinomial models with unknown index using data augmentation. Journal of Computational and Graphical Statistics 16, 67–85. doi:10.1198/106186007X181425

Royle, J.A., Karanth, K.U., Gopalaswamy, A.M., Kumar, N.S., 2009. Bayesian inference in camera trapping studies for a class of spatial capture-recapture

models. Ecology 90, 3233–3244.

Royle, J.A., Kéry, M., Guélat, J., 2011a. Spatial capture-recapture models for search-encounter data. Methods in Ecology and Evolution 2, 602–611. doi:10.1111/j.2041-210X.2011.00116.x

Royle, J.A., Magoun, A.J., Gardner, B., Valkenburg, P., Lowell, R.E., 2011b. Density estimation in a wolverine population using spatial capture–recapture models. Journal of Wildlife Management 75, 604–611. doi:10.1002/jwmg.79

Royle, J.A., Young, K.V., 2008. A hierarchical model for spatial capture-recapture data. Ecology 89, 2281–2289.

Russell, R.E., Royle, J.A., Desimone, R., Schwartz, M.K., Edwards, V.L., Pilgrim, K.P., Mckelvey, K.S., 2012. Estimating abundance of mountain lions from unstructured spatial sampling. Journal of Wildlife Management 76, 1551–1561. doi:10.1002/jwmg.412

Sinclair, A.R.E., Fryxell, J.M., Caughley, G., 2005. Wildlife ecology, conservation, and management, Second edition. ed. Blackwell Publishing, Oxford, United Kingdom.

Strauss, D.J., 1975. A model for clustering. Biometrika 62, 467–475. doi:10.1093/biomet/62.2.467

Thompson, C.M., Royle, J.A., Garner, J.D., 2012. A framework for inference about carnivore density from unstructured spatial sampling of scat using detector dogs. Journal of Wildlife Management 76, 863–871. doi:10.1002/jwmg.317

Williams, B.K., Nichols, J.D., Conroy, M.J., 2002. Analysis and management of animal populations. Academic Press, San Diego, USA.

Table 2.1. Case study: analysis of photographic capture-recapture data for African elephants at the Associated Private Nature Reserves, South Africa, 2008. Summaries of posterior distributions for model parameters, including elephant density (*D*).

							[a]Discrepancy between expected and observed encounters across...		
Model[a]	Parameter	Mean	SD	2.5%	50%	97.5%	detectors and individuals	individuals only	detectors only
SCR0	ψ	0.4562	0.0657	0.3332	0.4537	0.5918	0.0390	0.3923	0.2463
	σ	1.2811	0.1087	1.0971	1.2734	1.5212			
	D	0.1382	0.0191	0.1027	0.1373	0.1770			
SCRd	ψ	0.4549	0.0665	0.3334	0.4527	0.5935	0.0797	0.3787	0.2450
	σ	1.2851	0.1154	1.0948	1.2752	1.5335			
	D	0.1379	0.0197	0.1024	0.1370	0.1788			
	β_0	-2.4713	0.1227	-2.7161	-2.4695	-2.2382			
	β_1	0.1428	0.0816	-0.0161	0.1421	0.3047			
SCRh	ψ	0.6853	0.1378	0.4429	0.6738	0.9624	0.1133	0.5710	0.2597
	σ	1.2627	0.1092	1.0751	1.2528	1.4975			
	D	0.2077	0.0417	0.1342	0.2038	0.2921			
	α_0	-3.3739	0.4171	-4.2928	-3.3382	-2.6701			
	α_1	0.3205	0.0539	0.2230	0.3186	0.4326			
	σ_ε	1.0644	0.2810	0.5242	1.0616	1.6349			
SCRdh	ψ	0.6630	0.1312	0.4382	0.6523	0.9347	0.1477	0.5757	0.2443
	σ	1.2636	0.1075	1.0802	1.2538	1.4944			

D	0.2010	0.0396	0.1330	0.1973	0.2842
α_0	-3.2616	0.4014	-4.1551	-3.2164	-2.6325
α_1	0.3198	0.0529	0.2239	0.3181	0.4285
β_1	0.1156	0.0932	-0.0668	0.1170	0.2948
σ_ε	0.9858	0.2790	0.5148	0.9663	1.5564

[a] Bayesian *P*-values to assess encounter model goodness of fit; adequate fit indicated by values 0.1–0.9.

[b] Spatial capture-recapture (SCR) models: SCR0, basic model with no covariate or heterogeneity; SCRd, basic model with "distance to research base" covariate; SCRh, individual heterogeneity model; SCRdh, individual heterogeneity model with distance covariate.

Table 2.2. Simulation study: comparison of four spatial capture-recapture models fitted to simulated encounter data to estimate density (*D*). Reported values were averages across 50 simulated data sets.

				[a]Discrepancy between expected and observed encounters across...		
Model[b]	*D*	Mean $\hat{D}$	Mean *SD*	detectors and individuals	individuals only	detectors only
SCR0	0.1364	0.0784	0.0181	0.4744	0.3681	0.1037
SCRa	0.1364	0.1412	0.0243	0.4931	0.4928	0.5414
SCRg	0.1364	0.2081	0.0417	0.4653	0.5605	0.1027
SCRi	0.1364	0.3533	0.1296	0.4930	0.4794	0.1040

[a] Bayesian *P*-values to assess encounter model goodness of fit; adequate fit indicated by values 0.1–0.9.

[b] Spatial capture-recapture (SCR) models: SCR0, basic model to estimate density of individuals; SCRa, basic model to estimate density of groups and multiply by average group size; SCRg, heterogeneity model with group identity as random effect; SCRi, heterogeneity model with individual identity as random effect.

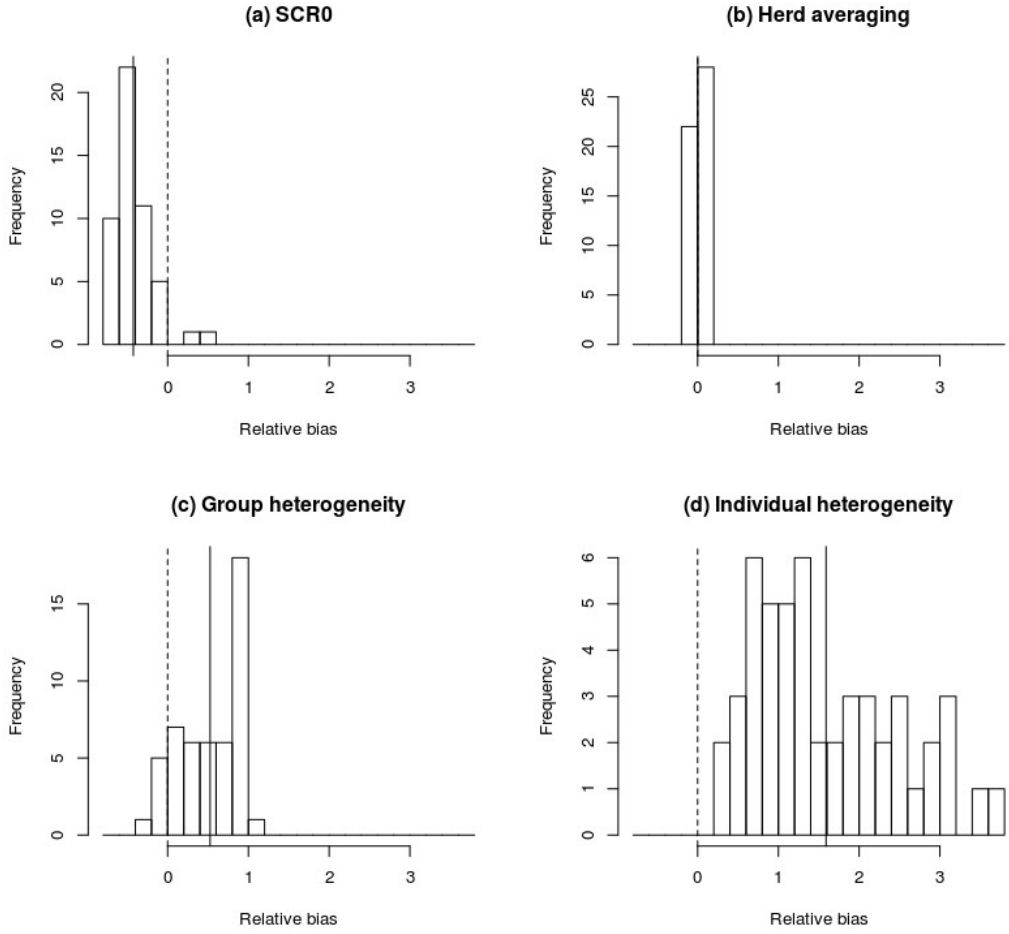


Figure 2.1. Data simulation: relative bias for spatial capture-recapture (SCR) estimation of density. Plotted are ranges of relative bias for 50 simulated data sets representing encounters with aggregated study animals, each data set analysed by four model: (a) basic SCR model estimating density of individuals, with no accounting for aggregations; (b) basic SCR model estimating density of aggregations, multiplied by average aggregation size; (c) SCR model with encounter heterogeneity, random effect represented by group identity; (d) SCR model with encounter heterogeneity, random effect represented by individual identity. The dashed line (-----) represents no bias; the solid line (——) represents the average bias for that model.

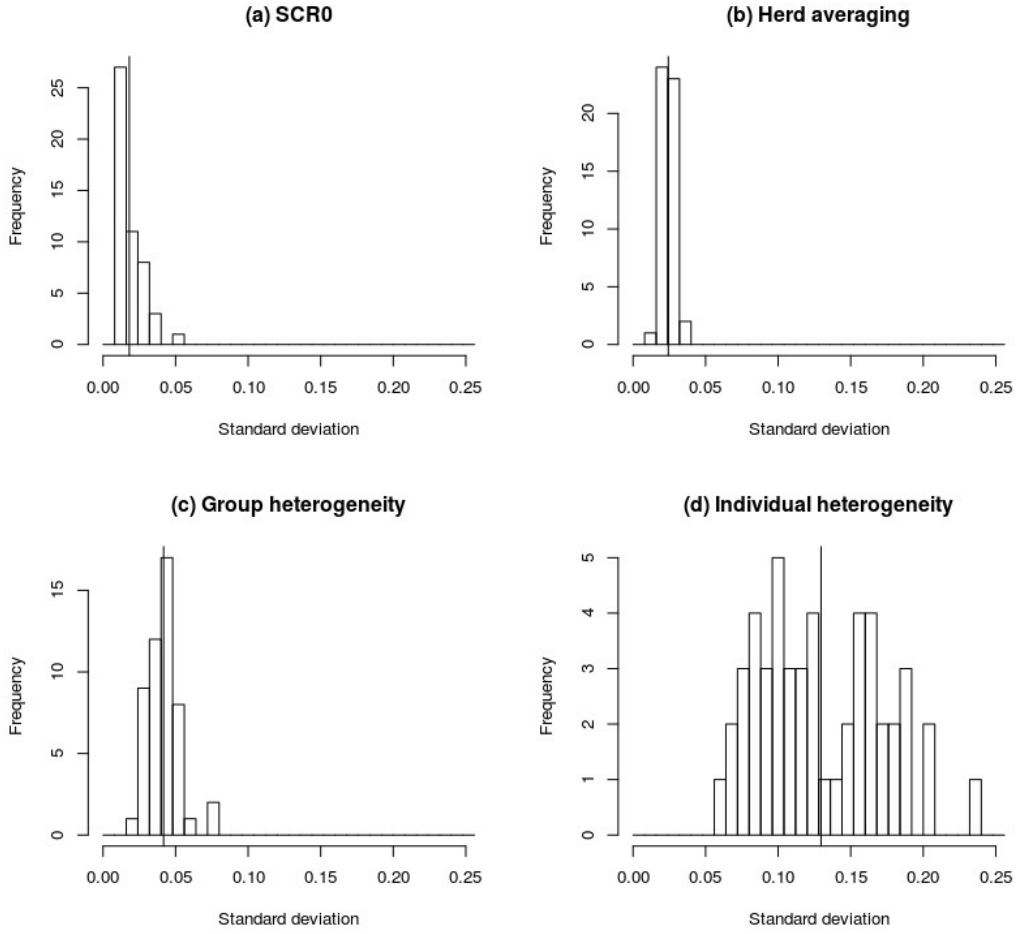


Figure 2.2. Data simulation: precision of spatial capture-recapture (SCR) density estimates, as measured by standard deviation of the density posterior distribution. Plotted are ranges of standard deviations for 50 simulated data sets representing encounters with aggregated study animals, each data set analysed by four model: (a) basic SCR model estimating density of individuals, with no accounting for aggregations; (b) basic SCR model estimating density of aggregations, multiplied by average aggregation size; (c) SCR model with encounter heterogeneity, random effect represented by group identity; (d) SCR model with encounter heterogeneity, random effect represented by individual identity. The solid line (——) represents the average standard deviation for that model.

CHAPTER THREE – CONCLUSION

The objective of this investigation was driven by a growing necessity to derive information useful for wildlife management and conservation from opportunistically-collected observations of wild animals. In the context of this study, those data are spatially-explicit photographs of individually recognizable animals, from which encounter histories could generate estimates of population density via spatial capture-recapture (SCR) models.

Two common problems can affect the utility of opportunistically-collected data (Kéry et al., 2010). One problem is the consequence of non-random sampling (Yoccoz et al., 2001). Because opportunistic collection is expected to occur in regions or at times when a species of interest is most likely to be seen, the resulting data set will be more representative of those high-visibility instances, and less so of a wider domain over which one might aim to draw inferences (Kéry et al., 2010). This results in a mismatch between the "target population" and the "sampled population" (Conroy and Carroll, 2009).

A second problem happens when detection of individuals varies over time in response to changes in survey methods (e.g. observers, survey intensity; MacKenzie and Kendall, 2002; Gu and Swihart, 2004; (Kéry et al., 2010). Changes in logistics can mask changes in detection probability or population parameters if they happen concomitantly, or if perceived changes in population parameters are the result of changes in survey logistics.

A third problem has been the focus of the present research project. That

problem relates to incomplete information collected during opportunistic observations. The consequence is missing data that might be used to apply appropriate models and assess assumptions to evaluate validity of estimates.

For example, in the present study, a more complete accounting of African elephant (*Loxodonta africana*) herds and herd membership might have made it possible to estimate densities of herds and average herd size, to produce a density estimate that has relatively low bias. However, incomplete information during field collection is often unavoidable. Animals that occur in large herds might not always be easily recorded and recognized if they are moving, obscured by herd-mates or hidden in vegetation.

Hierarchical Bayesian modelling provides a flexible tool that could be used to address the problem of incomplete information by taking a phenomenon of interest that cannot be directly observed and formulating it in terms of elements that can be observed (Link and Barker, 2010). For SCR analysis, this is accomplished by separating the overall analysis into an observation model and an ecological process model (i.e. a state-space model; Royle and Dorazio, 2008). Usually in SCR, the ecological process is a spatial point process model that represents randomly-located activity centres of individuals in the population (Efford, 2004; Borchers and Efford, 2008; Royle and Young, 2008). An alternative considered in Chapter 2 was to represent herds rather than individuals with the spatial point process model, an approach that produced estimates of population density with low bias (Fig. 2.2b). A further alternative not considered here might be to replace the spatial point process model in the standard SCR

analysis with a model that more realistically represents the activity centres of aggregated individuals (Royle et al., 2014). Such a process model could be based on a cluster point process (Illian et al., 2008; Royle et al., 2014).

The growing availability of data collected by opportunistic methods means an increasing need to assess the validity of estimation methods that are applied to such data. It also points to an increase in necessity to develop methods that are specifically designed to accommodate the irregular nature of opportunistically collected data.

3.1 References

Borchers, D.L., Efford, M.G., 2008. Spatially explicit maximum likelihood methods for capture–recapture studies. Biometrics 64, 377–385. doi:10.1111/j.1541-0420.2007.00927.x

Conroy, M.J., Carroll, J.P., 2009. Quantitative conservation of vertebrates. Wiley-Blackwell, Chichester, United Kingdom.

Efford, M.G., 2004. Density estimation in live-trapping studies. Oikos 106, 598–610. doi:10.1111/j.0030-1299.2004.13043.x

Gu, W., Swihart, R.K., 2004. Absent or undetected? Effects of non-detection of species occurrence on wildlife–habitat models. Biological Conservation 116, 195–203. doi:10.1016/S0006-3207(03)00190-3

Illian, J., Penttinen, A., Stoyan, H., Stoyan, D., 2008. Statistical analysis and modelling of spatial point patterns. John Wiley & Sons, Chichester, United Kingdom.

Kéry, M., Royle, J.A., Schmid, H., Schaub, M., Volet, B., Häfliger, G., Zbinden, N., 2010. Site-occupancy distribution modeling to correct population-trend estimates derived from opportunistic observations. Conservation Biology 24, 1388–1397. doi:10.1111/j.1523-1739.2010.01479.x

Link, W.A., Barker, R.J., 2010. Bayesian inference with ecological applications. Academic Press, London, United Kingdom.

MacKenzie, D., Kendall, W., 2002. How should detection probability be incorporated into estimates of relative abundance? Ecology 83, 2387–2393.

Royle, J.A., Chandler, R.B., Sollmann, R., Gardner, B., 2014. Spatial capture-recapture. Academic Press, Waltham, USA.

Royle, J.A., Dorazio, R.M., 2008. Hierarchical modelling and inference in ecology. Academic Press, Waltham, USA.

Yoccoz, N.G., Nichols, J.D., Boulinier, T., 2001. Monitoring of biological diversity in space and time. Trends in Ecology & Evolution 16, 446–453. doi:10.1016/S0169-5347(01)02205-4

APPENDICES

APPENDIX 1. Scripts in R to conduct case study analysis with the basic spatial capture-recapture model with no covariate or heterogeneity (SCR0).

```
# Install and set up required packages
ipak <- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
}

packages <-
c("lattice","coda","R2OpenBUGS","R2jags","rjags",'mcmcplots','run
jags','jagsUI','mcmc')
ipak(packages)

library(sp)
library(rgdal)
library(scrbook)

nc.ele <- read.table('ElephantWorking_160115.csv', header = T,
sep = ',')

## GIS analysis for APNR elephants in R

# Set CRS (DD, WGS84)
# Transform to UTM
coordinates(nc.ele) <- c('xcoord', 'ycoord')
crsll <- CRS('+proj=longlat + ellps=WGS84')
eledata <- SpatialPoints(nc.ele, proj4string = crsll)
crsutm <- CRS('+proj=utm +zone=36 +south +datum=WGS84 +units=m
+no_defs +ellps=WGS84 +towgs84=0,0,0')
eledata <- spTransform(eledata, crsutm)

# read APNR study area shapefile
apnr <- readOGR('.', 'APNR_total_2014_UTM')

# create grid over study area
clsz <- 5000
bb <- bbox(apnr)
bb
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
apnr.grid <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
apnr.sg <- SpatialGrid(apnr.grid, proj4string = crsutm)
ele.cell <- over(eledata, apnr.sg)
ele.spt <- unique(coordinates(apnr.sg)[ele.cell,])
ele.spt <- SpatialPoints(ele.spt, proj4string = crsutm)
```

```
ele.spx <- SpatialPixels(ele.spt, proj4string = crsutm)
ele.sg <- as(ele.spx, 'SpatialGrid')

# set up encounter data file
ele.edf <- matrix(NA, nrow = dim(nc.ele)[1], ncol = 4)
colnames(ele.edf) <- c('sessID', 'indID', 'occID', 'trapID')
ele.edf[,1] <- rep(1, dim(nc.ele)[1])
ele.edf[,2] <- nc.ele$indID
ele.edf[,3] <- nc.ele$occID

# renumber grid cell numbers to trap ID integers
ele.edf[,4] <- over(eledata, ele.spx)

## Assemble trap data file (TDF)
nocc <- max(nc.ele$occID)
ele.tdf <- matrix(NA, nrow = dim(coordinates(ele.spx))[1], ncol =
3 + nocc)
colnames(ele.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
ele.tdf[,1] <- 1:dim(coordinates(ele.spx))[1]
ele.tdf[,2:3] <- coordinates(ele.spx)
trap.dates <- table(ele.edf[,4], ele.edf[,3])
trap.dates[trap.dates>1] <- 1 # trap x occasion -- information on
when 'traps' were 'operational'
ele.tdf[,4:(nocc+3)] <- trap.dates
trapOp <- ele.tdf[,4:ncol(ele.tdf)]

# create encounter arrays
y3d <- SCR23darray(ele.edf, ele.tdf) # ind x trap x occ
y <- apply(y3d, c(1, 2), sum) # ind x trap -- summed across
occasion

# set up data for analysis
traplocs <- as.matrix(ele.tdf[,2:3])
coord.scale <- 5000
traplocs[,1] <- traplocs[,1] - min(traplocs[,1])
traplocs[,2] <- traplocs[,2] - min(traplocs[,2])
traplocs <- traplocs/coord.scale
ntraps <- nrow(traplocs)

buffer <- 3
Xl <- min(traplocs[,1]) - clsz/(2*coord.scale) - buffer
Xu <- max(traplocs[,1]) + clsz/(2*coord.scale) + buffer
Yl <- min(traplocs[,2]) - clsz/(2*coord.scale) - buffer
Yu <- max(traplocs[,2]) + clsz/(2*coord.scale) + buffer
area <- (Xu - Xl) * (Yu - Yl)

M <- 1000
nind <- dim(y3d)[1]
nz <- M - nind

sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
```

```
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
}
zst <- c(rep(1, nind), rep(0, M - nind))

MASK <- ele.tdf[,4:ncol(ele.tdf)]
K <- dim(y3d)[3]
newy <- array(0, dim = c(nind + nz, ntraps, K))
for (j in 1:nind) {
  newy[j, 1:ntraps, 1:K] <- y3d[j, 1:ntraps, 1:K]
}

y3d <- newy
K <- apply(MASK, 1, sum)
y <- apply(y3d, c(1, 2), sum)

################################

# model in BUGS
cat('
model{

# priors
alpha1 <- 1/(2*sigma*sigma)
sigma ~ dunif(0, 10)
psi ~ dunif(0, 1)
p0 ~ dunif(0, 1)

# likelihood
for(i in 1:M){

  z[i] ~ dbern(psi)
  s[i,1] ~ dunif(Xl, Xu)
  s[i,2] ~ dunif(Yl, Yu)

  for(j in 1:ntraps){
    y[i,j] ~ dbin(mu[i,j], K[j])

    mu[i,j] <- z[i]*p[i,j]
    mu2[i,j] <- mu[i,j]*K[j]

    p[i,j] <- p0*exp(-alpha1*d[i,j]*d[i,j])
    d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

    # discrepancy for Bayesian p-value GOF
    ynew[i,j] ~ dbin(mu[i,j], K[j])
    err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
    errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
  }
  expected[i] <- sum(mu2[i,])
  nsum[i] <- sum(y[i,])
```

```
  nsumnew[i] <- sum(ynew[i,])

  err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
  err1new[i] <- pow(pow(nsumnew[i], 0.5) - pow(expected[i],0.5),
2)
}

for(j in 1:ntraps){
  traptotals[j] <- sum(y[1:M,j])
  traptotalsnew[j] <-sum(ynew[1:M,j])
  expectedtrap[j] <- sum(mu[,j])*K[j]
  err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
  err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
pow(expectedtrap[j], 0.5), 2)
}

X3obs<- sum(err3[])
X3new<- sum(err3new[])
X2obs<- sum(err1[])
X2new<- sum(err1new[])
X1obs <- sum(err[,])
X1new <- sum(errnew[,])

# derived values
N <- sum(z[1:M])
D <- N/(area*25) # convert dens/cell to dens/sq.km
}
', file = 'modelBinom.txt')

data <- list(y = y,
             traplocs = traplocs,
             K = K,
             M = M,
             ntraps = ntraps,
             Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
             area = area)

inits <- function() {list(sigma = runif(1, 0, 2),
                          psi = runif(1, 0, 1),
                          p0 = runif(1, 0, 1),
                          z = zst, s = sst)}

params <- c('psi', 'sigma', 'N', 'D', 'p0', 'alpha1',
            'X1obs', 'X1new', 'X2obs', 'X2new', 'X3obs', 'X3new')

ni <- 200000
nb <- 100000
nt <- 100
nc <- 3

#*****************************************
# Do the MCMC stuff calling WinBUGS from R
```

```
#****************************************

start <- as.POSIXlt(Sys.time())
print(start)
out <- jags(data = data, inits = inits, parameters.to.save =
params, model.file = "modelBinom.txt", n.chains = nc, n.thin =
nt, n.iter = ni, n.burnin = nb,parallel=T)
end <- as.POSIXlt(Sys.time())
print(end)
duration <- print(difftime(end,start,units='mins'))

print(out,3)
save(out, file='OutputSCR_0.out')
```

APPENDIX 2. Scripts in R to conduct case study analysis with the basic spatial capture-recapture model with “distance to research base” covariate (SCRd).

```
# Install and set up required packages
ipak <- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
}

packages <-
c("lattice","coda","R2OpenBUGS","R2jags","rjags",'mcmcplots','run
jags','jagsUI','mcmc')
ipak(packages)

library(sp)
library(rgdal)
library(scrbook)

nc.ele <- read.table('ElephantWorking_160115.csv', header = T,
sep = ',')

## GIS analysis for APNR elephants in R

# Set CRS (DD, WGS84)
# Transform to UTM
coordinates(nc.ele) <- c('xcoord', 'ycoord')
crsll <- CRS('+proj=longlat + ellps=WGS84')
eledata <- SpatialPoints(nc.ele, proj4string = crsll)
crsutm <- CRS('+proj=utm +zone=36 +south +datum=WGS84 +units=m
+no_defs +ellps=WGS84 +towgs84=0,0,0')
eledata <- spTransform(eledata, crsutm)

# read APNR study area shapefile
apnr <- readOGR('.', 'APNR_total_2014_UTM')

# create grid over study area
clsz <- 5000
bb <- bbox(apnr)
bb
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
apnr.grid <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
apnr.sg <- SpatialGrid(apnr.grid, proj4string = crsutm)
ele.cell <- over(eledata, apnr.sg)
ele.spt <- unique(coordinates(apnr.sg)[ele.cell,])
ele.spt <- SpatialPoints(ele.spt, proj4string = crsutm)
ele.spx <- SpatialPixels(ele.spt, proj4string = crsutm)
ele.sg <- as(ele.spx, 'SpatialGrid')
```

```
# distance to human settelments
anth <- readOGR('.', 'Anthropogenic_UTM')
proj4string(anth) <- crsutm
d2anth.rast <- distanceFromPoints(ele.sg, anth)
d2anth <- as(d2anth.rast, 'SpatialPixelsDataFrame')
d2anth.ele <- over(eledata, d2anth)

# distance to research base
mich <- subset(anth, anth$Property=='Tanda Tula')
d2mich.rast <- distanceFromPoints(ele.spx, mich)
d2mich <- as(d2mich.rast, 'SpatialPixelsDataFrame')
d2mich.ele <- over(eledata, d2mich)

# set up encounter data file
ele.edf <- matrix(NA, nrow = dim(nc.ele)[1], ncol = 4)
colnames(ele.edf) <- c('sessID', 'indID', 'occID', 'trapID')
ele.edf[,1] <- rep(1, dim(nc.ele)[1])
ele.edf[,2] <- nc.ele$indID
ele.edf[,3] <- nc.ele$occID

# renumber grid cell numbers to trap ID integers
ele.edf[,4] <- over(eledata, ele.spx)

## Assemble trap data file (TDF)
nocc <- max(nc.ele$occID)
ele.tdf <- matrix(NA, nrow = dim(coordinates(ele.spx))[1], ncol =
3 + nocc)
colnames(ele.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
ele.tdf[,1] <- 1:dim(coordinates(ele.spx))[1]
ele.tdf[,2:3] <- coordinates(ele.spx)
trap.dates <- table(ele.edf[,4], ele.edf[,3])
trap.dates[trap.dates>1] <- 1 # trap x occasion -- information on
when 'traps' were 'operational'
ele.tdf[,4:(nocc+3)] <- trap.dates
trapOp <- ele.tdf[,4:ncol(ele.tdf)]

# create encounter arrays
y3d <- SCR23darray(ele.edf, ele.tdf) # ind x trap x occ
y <- apply(y3d, c(1, 2), sum) # ind x trap -- summed across
occasion

# set up data for analysis
traplocs <- as.matrix(ele.tdf[,2:3])
coord.scale <- 5000
traplocs[,1] <- traplocs[,1] - min(traplocs[,1])
traplocs[,2] <- traplocs[,2] - min(traplocs[,2])
traplocs <- traplocs/coord.scale
ntraps <- nrow(traplocs)

buffer <- 3
Xl <- min(traplocs[,1]) - clsz/(2*coord.scale) - buffer
```

```
Xu <- max(traplocs[,1]) + clsz/(2*coord.scale) + buffer
Yl <- min(traplocs[,2]) - clsz/(2*coord.scale) - buffer
Yu <- max(traplocs[,2]) + clsz/(2*coord.scale) + buffer
area <- (Xu - Xl) * (Yu - Yl)

M <- 1000
nind <- dim(y3d)[1]
nz <- M - nind

sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
}
zst <- c(rep(1, nind), rep(0, M - nind))

MASK <- ele.tdf[,4:ncol(ele.tdf)]
K <- dim(y3d)[3]
newy <- array(0, dim = c(nind + nz, ntraps, K))
for (j in 1:nind) {
  newy[j, 1:ntraps, 1:K] <- y3d[j, 1:ntraps, 1:K]
}

y3d <- newy
K <- apply(MASK, 1, sum)
y <- apply(y3d, c(1, 2), sum)

# distance to Tunda Tula covariate
d2TT.rast <- distanceFromPoints(ele.spt, mich)
d2TT <- extract(d2TT.rast, ele.spt)

#################################

## Model using Binomial

# model in BUGS
cat('
model{

# priors
alpha1 <- 1/(2*sigma*sigma)
sigma ~ dunif(0, 10)
psi ~ dunif(0, 1)

beta0 ~ dnorm(0, 0.01)T(-10,10)
beta1 ~ dnorm(0, 0.01)T(-10,10)

# likelihood
for(j in 1:ntraps){
  p0[j] <- exp(logit.p0[j])/(1 + exp(logit.p0[j]))
  logit.p0[j] <- beta0 + beta1*d2TT[j]
}
```

```
for(i in 1:M){
  z[i] ~ dbern(psi)
  s[i,1] ~ dunif(Xl, Xu)
  s[i,2] ~ dunif(Yl, Yu)

  for(j in 1:ntraps){
    y[i,j] ~ dbin(mu[i,j], K[j])
    mu[i,j] <- z[i]*p[i,j]
    mu2[i,j] <- mu[i,j]*K[j]
    p[i,j] <- p0[j]*exp(-alpha1*d[i,j]*d[i,j])
    d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

    # discrepancy for Bayesian p-value GOF
    ynew[i,j] ~ dbin(mu[i,j], K[j])
    err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
    errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
  }
  expected[i] <- sum(mu2[i,])
  nsum[i] <- sum(y[i,])
  nsumnew[i] <- sum(ynew[i,])

  err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
  err1new[i] <- pow(pow(nsumnew[i], 0.5) - pow(expected[i],0.5),
2)
}

for(j in 1:ntraps){
  traptotals[j] <- sum(y[1:M,j])
  traptotalsnew[j] <-sum(ynew[1:M,j])
  expectedtrap[j] <- sum(mu[,j])*K[j]
  err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
  err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
pow(expectedtrap[j], 0.5), 2)
}

X3obs<- sum(err3[])
X3new<- sum(err3new[])
X2obs<- sum(err1[])
X2new<- sum(err1new[])
X1obs <- sum(err[,])
X1new <- sum(errnew[,])

# derived values
N <- sum(z[1:M])
D <- N/(area*25) # convert dens/cell to dens/sq.km
}
', file = 'modelBinom.txt')

data <- list(y = y,
             traplocs = traplocs,
```

```
              d2TT = as.numeric(scale(d2TT)),
              K = K,
              M = M,
              ntraps = ntraps,
              Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
              area = area)

inits <- function() {list(sigma = runif(1, 0, 2),
                          psi = runif(1, 0, 1),
                          beta0 = rnorm(1, 0, 10),
                          beta1 = rnorm(1, 0, 10),
                          z = zst, s = sst)}

params <- c('psi', 'sigma', 'N', 'D', 'p0', 'alpha1', 'beta0',
'beta1', 'X1obs', 'X1new', 'X2obs', 'X2new', 'X3obs', 'X3new')



ni <- 200000
nb <- 100000
nt <- 100
nc <- 3

#*****************************************
# Do the MCMC stuff calling WinBUGS from R
#*****************************************

start <- as.POSIXlt(Sys.time())
print(start)
out <- jags(data = data, inits = inits, parameters.to.save =
params, model.file = "modelBinom.txt",
            n.chains = nc, n.thin = nt, n.iter = ni, n.burnin =
nb,parallel=T)
end <- as.POSIXlt(Sys.time())
print(end)
duration <- print(difftime(end,start,units='mins'))

print(out,3)
save(out, file='OutputSCR_dist.out')
```

APPENDIX 3. Scripts in R to conduct case study analysis with the basic spatial capture-recapture model with individual heterogeneity (SCRh).

```
# Install and set up required packages
ipak <- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
}

packages <-
c("lattice","coda","R2OpenBUGS","R2jags","rjags",'mcmcplots','run
jags','jagsUI','mcmc')
ipak(packages)

library(sp)
library(rgdal)
library(scrbook)

nc.ele <- read.table('ElephantWorking_160115.csv', header = T,
sep = ',')

## GIS analysis for APNR elephants in R

# Set CRS (DD, WGS84)
# Transform to UTM
coordinates(nc.ele) <- c('xcoord', 'ycoord')
crsll <- CRS('+proj=longlat + ellps=WGS84')
eledata <- SpatialPoints(nc.ele, proj4string = crsll)
crsutm <- CRS('+proj=utm +zone=36 +south +datum=WGS84 +units=m
+no_defs +ellps=WGS84 +towgs84=0,0,0')
eledata <- spTransform(eledata, crsutm)

# read APNR study area shapefile
apnr <- readOGR('.', 'APNR_total_2014_UTM')

# create grid over study area
clsz <- 5000
bb <- bbox(apnr)
bb
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
apnr.grid <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
apnr.sg <- SpatialGrid(apnr.grid, proj4string = crsutm)
ele.cell <- over(eledata, apnr.sg)
ele.spt <- unique(coordinates(apnr.sg)[ele.cell,])
ele.spt <- SpatialPoints(ele.spt, proj4string = crsutm)
ele.spx <- SpatialPixels(ele.spt, proj4string = crsutm)
ele.sg <- as(ele.spx, 'SpatialGrid')
```

```
# set up encounter data file
ele.edf <- matrix(NA, nrow = dim(nc.ele)[1], ncol = 4)
colnames(ele.edf) <- c('sessID', 'indID', 'occID', 'trapID')
ele.edf[,1] <- rep(1, dim(nc.ele)[1])
ele.edf[,2] <- nc.ele$indID
ele.edf[,3] <- nc.ele$occID

# renumber grid cell numbers to trap ID integers
ele.edf[,4] <- over(eledata, ele.spx)

## Assemble trap data file (TDF)
nocc <- max(nc.ele$occID)
ele.tdf <- matrix(NA, nrow = dim(coordinates(ele.spx))[1], ncol =
3 + nocc)
colnames(ele.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
ele.tdf[,1] <- 1:dim(coordinates(ele.spx))[1]
ele.tdf[,2:3] <- coordinates(ele.spx)
trap.dates <- table(ele.edf[,4], ele.edf[,3])
trap.dates[trap.dates>1] <- 1 # trap x occasion -- information on
when 'traps' were 'operational'
ele.tdf[,4:(nocc+3)] <- trap.dates
trapOp <- ele.tdf[,4:ncol(ele.tdf)]

# create encounter arrays
y3d <- SCR23darray(ele.edf, ele.tdf) # ind x trap x occ
y <- apply(y3d, c(1, 2), sum) # ind x trap -- summed across
occasion

# set up data for analysis
traplocs <- as.matrix(ele.tdf[,2:3])
coord.scale <- 5000
traplocs[,1] <- traplocs[,1] - min(traplocs[,1])
traplocs[,2] <- traplocs[,2] - min(traplocs[,2])
traplocs <- traplocs/coord.scale
ntraps <- nrow(traplocs)

buffer <- 3
Xl <- min(traplocs[,1]) - clsz/(2*coord.scale) - buffer
Xu <- max(traplocs[,1]) + clsz/(2*coord.scale) + buffer
Yl <- min(traplocs[,2]) - clsz/(2*coord.scale) - buffer
Yu <- max(traplocs[,2]) + clsz/(2*coord.scale) + buffer
area <- (Xu - Xl) * (Yu - Yl)

M <- 1000
nind <- dim(y3d)[1]
nz <- M - nind

sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
```

```
}
zst <- c(rep(1, nind), rep(0, M - nind))

MASK <- ele.tdf[,4:ncol(ele.tdf)]
K <- dim(y3d)[3]
newy <- array(0, dim = c(nind + nz, ntraps, K))
for (j in 1:nind) {
  newy[j, 1:ntraps, 1:K] <- y3d[j, 1:ntraps, 1:K]
}

y3d <- newy
K <- apply(MASK, 1, sum)
y <- apply(y3d, c(1, 2), sum)

################################

## Model using Binomial

# model in BUGS
cat('
    model{

    # priors
    alpha1 <- 1/(2*sigma*sigma)
    sigma ~ dunif(0, 10)
    psi ~ dunif(0, 1)

    alpha0 ~ dnorm(0, 0.01)T(-10,10)
    sigma.eps ~ dunif(0, 10)
    tau.eps <- 1/(sigma.eps*sigma.eps)

    # likelihood
    for(i in 1:M){

    logit(p0[i]) <- alpha0 + eps[i]
    eps[i] ~ dnorm(0, tau.eps)T(-10,10)

    z[i] ~ dbern(psi)
    s[i,1] ~ dunif(Xl, Xu)
    s[i,2] ~ dunif(Yl, Yu)

    for(j in 1:ntraps){
    y[i,j] ~ dbin(mu[i,j], K[j])

    mu[i,j] <- z[i]*p[i,j]
    mu2[i,j] <- mu[i,j]*K[j]

    p[i,j] <- p0[i]*exp(-alpha1*d[i,j]*d[i,j])
    d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

    # discrepancy for Bayesian p-value GOF
    ynew[i,j] ~ dbin(mu[i,j], K[j])
```

```
    err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
    errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
    }
    expected[i] <- sum(mu2[i,])
    nsum[i] <- sum(y[i,])
    nsumnew[i] <- sum(ynew[i,])

    err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
    err1new[i] <- pow(pow(nsumnew[i], 0.5) -
pow(expected[i],0.5), 2)
    }

    for(j in 1:ntraps){
    traptotals[j] <- sum(y[1:M,j])
    traptotalsnew[j] <-sum(ynew[1:M,j])
    expectedtrap[j] <- sum(mu[,j])*K[j]
    err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
    err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
pow(expectedtrap[j], 0.5), 2)
    }

    X3obs<- sum(err3[])
    X3new<- sum(err3new[])
    X2obs<- sum(err1[])
    X2new<- sum(err1new[])
    X1obs <- sum(err[,])
    X1new <- sum(errnew[,])

    # derived values
    N <- sum(z[1:M])
    D <- N/(area*25) # convert dens/cell to dens/sq.km
    }
    ', file = 'modelBinom.txt')


data <- list(y = y,
             traplocs = traplocs,
             K = K,
             M = M,
             ntraps = ntraps,
             Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
             area = area)



inits <- function() {list(sigma = runif(1, 0, 2),
                          psi = runif(1, 0, 1),
                          alpha0 = rnorm(1, 0, 2),
                          sigma.eps = runif(1,0,2),
                          z = zst, s = sst)}
```

```
params <- c('psi', 'sigma', 'N', 'D', 'alpha0', 'alpha1',
'sigma.eps', 'X1obs', 'X1new', 'X2obs', 'X2new', 'X3obs',
'X3new')



ni <- 200000
nb <- 100000
nt <- 100
nc <- 3

#*****************************************
# Do the MCMC stuff calling WinBUGS from R
#*****************************************

start <- as.POSIXlt(Sys.time())
print(start)


out <- jags(data = data, inits = inits, parameters.to.save =
params, model.file = "modelBinom.txt", n.chains = nc, n.thin =
nt, n.iter = ni, n.burnin = nb,parallel=T)


end <- as.POSIXlt(Sys.time())

print(end)

duration <- print(difftime(end,start,units='mins'))


print(out,3)

save(out,file='out.Rdata')
```

APPENDIX 4. Scripts in R to conduct case study analysis with spatial capture-recapture model with “distance to research base” covariate and individual heterogeneity (SCRdh).

```
# Install and set up required packages
ipak <- function(pkg){
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg))
    install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE)
}

packages <-
c("lattice","coda","R2OpenBUGS","R2jags","rjags",'mcmcplots','run
jags','jagsUI','mcmc')
ipak(packages)

library(sp)
library(rgdal)
library(scrbook)

nc.ele <- read.table('ElephantWorking_160115.csv', header = T,
sep = ',')

## GIS analysis for APNR elephants in R

# Set CRS (DD, WGS84)
# Transform to UTM
coordinates(nc.ele) <- c('xcoord', 'ycoord')
crsll <- CRS('+proj=longlat + ellps=WGS84')
eledata <- SpatialPoints(nc.ele, proj4string = crsll)
crsutm <- CRS('+proj=utm +zone=36 +south +datum=WGS84 +units=m
+no_defs +ellps=WGS84 +towgs84=0,0,0')
eledata <- spTransform(eledata, crsutm)

# read APNR study area shapefile
apnr <- readOGR('.', 'APNR_total_2014_UTM')

# create grid over study area
clsz <- 5000
bb <- bbox(apnr)
bb
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
apnr.grid <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
apnr.sg <- SpatialGrid(apnr.grid, proj4string = crsutm)
ele.cell <- over(eledata, apnr.sg)
ele.spt <- unique(coordinates(apnr.sg)[ele.cell,])
ele.spt <- SpatialPoints(ele.spt, proj4string = crsutm)
ele.spx <- SpatialPixels(ele.spt, proj4string = crsutm)
```

```
ele.sg <- as(ele.spx, 'SpatialGrid')

## covariate layers

# distance to human settelments
anth <- readOGR('.', 'Anthropogenic_UTM')
proj4string(anth) <- crsutm
d2anth.rast <- distanceFromPoints(ele.sg, anth)
d2anth <- as(d2anth.rast, 'SpatialPixelsDataFrame')
d2anth.ele <- over(eledata, d2anth)

# distance to research base
mich <- subset(anth, anth$Property=='Tanda Tula')
d2mich.rast <- distanceFromPoints(ele.spx, mich)
d2mich <- as(d2mich.rast, 'SpatialPixelsDataFrame')
d2mich.ele <- over(eledata, d2mich)

# set up encounter data file
ele.edf <- matrix(NA, nrow = dim(nc.ele)[1], ncol = 4)
colnames(ele.edf) <- c('sessID', 'indID', 'occID', 'trapID')
ele.edf[,1] <- rep(1, dim(nc.ele)[1])
ele.edf[,2] <- nc.ele$indID
ele.edf[,3] <- nc.ele$occID

# renumber grid cell numbers to trap ID integers
ele.edf[,4] <- over(eledata, ele.spx)

## Assemble trap data file (TDF)
nocc <- max(nc.ele$occID)
ele.tdf <- matrix(NA, nrow = dim(coordinates(ele.spx))[1], ncol =
3 + nocc)
colnames(ele.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
ele.tdf[,1] <- 1:dim(coordinates(ele.spx))[1]
ele.tdf[,2:3] <- coordinates(ele.spx)
trap.dates <- table(ele.edf[,4], ele.edf[,3])
trap.dates[trap.dates>1] <- 1 # trap x occasion -- information on
when 'traps' were 'operational'
ele.tdf[,4:(nocc+3)] <- trap.dates
trapOp <- ele.tdf[,4:ncol(ele.tdf)]

# create encounter arrays
y3d <- SCR23darray(ele.edf, ele.tdf) # ind x trap x occ
y <- apply(y3d, c(1, 2), sum) # ind x trap -- summed across
occasion

# set up data for analysis
traplocs <- as.matrix(ele.tdf[,2:3])
coord.scale <- 5000
traplocs[,1] <- traplocs[,1] - min(traplocs[,1])
traplocs[,2] <- traplocs[,2] - min(traplocs[,2])
traplocs <- traplocs/coord.scale
ntraps <- nrow(traplocs)
```

```
buffer <- 3
Xl <- min(traplocs[,1]) - clsz/(2*coord.scale) - buffer
Xu <- max(traplocs[,1]) + clsz/(2*coord.scale) + buffer
Yl <- min(traplocs[,2]) - clsz/(2*coord.scale) - buffer
Yu <- max(traplocs[,2]) + clsz/(2*coord.scale) + buffer
area <- (Xu - Xl) * (Yu - Yl)

M <- 1000
nind <- dim(y3d)[1]
nz <- M - nind

sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
}
zst <- c(rep(1, nind), rep(0, M - nind))

MASK <- ele.tdf[,4:ncol(ele.tdf)]
K <- dim(y3d)[3]
newy <- array(0, dim = c(nind + nz, ntraps, K))
for (j in 1:nind) {
  newy[j, 1:ntraps, 1:K] <- y3d[j, 1:ntraps, 1:K]
}

y3d <- newy
K <- apply(MASK, 1, sum)
y <- apply(y3d, c(1, 2), sum)

# distance to Tunda Tula covariate
d2TT.rast <- distanceFromPoints(ele.spt, mich)
d2TT <- extract(d2TT.rast, ele.spt)

#################################

## Model using Binomial

# model in BUGS
cat('
    model{

    # priors
    alpha1 <- 1/(2*sigma*sigma)
    sigma ~ dunif(0, 100)
    psi ~ dunif(0, 1)

    alpha0 ~ dnorm(0, 0.01)T(-5,5)
    sigma.eps ~ dunif(0, 100)
    tau.eps <- 1/(sigma.eps*sigma.eps)

    beta1 ~ dunif(-5, 5)
```

```
    # likelihood
    for(i in 1:M){

    for(j in 1:ntraps){
    logit(p0[i,j]) <- alpha0 + beta1*d2TT[j] + eps[i]
    }
    eps[i] ~ dnorm(0, tau.eps)T(-5,5)

    z[i] ~ dbern(psi)
    s[i,1] ~ dunif(Xl, Xu)
    s[i,2] ~ dunif(Yl, Yu)

    for(j in 1:ntraps){
    y[i,j] ~ dbin(mu[i,j], K[j])

    mu[i,j] <- z[i]*p[i,j]
    mu2[i,j] <- mu[i,j]*K[j]

    p[i,j] <- p0[i,j]*exp(-alpha1*d[i,j]*d[i,j])
    d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

    # discrepancy for Bayesian p-value GOF
    ynew[i,j] ~ dbin(mu[i,j], K[j])
    err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
    errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
    }
    expected[i] <- sum(mu2[i,])
    nsum[i] <- sum(y[i,])
    nsumnew[i] <- sum(ynew[i,])

    err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
    err1new[i] <- pow(pow(nsumnew[i], 0.5) -
pow(expected[i],0.5), 2)
    }

    for(j in 1:ntraps){
    traptotals[j] <- sum(y[1:M,j])
    traptotalsnew[j] <-sum(ynew[1:M,j])
    expectedtrap[j] <- sum(mu[,j])*K[j]
    err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
    err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
pow(expectedtrap[j], 0.5), 2)
    }

    X3obs<- sum(err3[])
    X3new<- sum(err3new[])
    X2obs<- sum(err1[])
    X2new<- sum(err1new[])
    X1obs <- sum(err[,])
    X1new <- sum(errnew[,])
```

```
    # derived values
    N <- sum(z[1:M])
    D <- N/(area*25) # convert dens/cell to dens/sq.km
    }
    ', file = 'modelBinom.txt')


data <- list(y = y,
             traplocs = traplocs,
             d2TT = as.numeric(scale(d2TT)),
             K = K,
             M = M,
             ntraps = ntraps,
             Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
             area = area)



inits <- function() {list(sigma = runif(1, 0, 100),
                          psi = runif(1, 0, 1),
                          alpha0 = rnorm(1, 0, 10),
                          sigma.eps = runif(1, 0, 10),
                          beta1 = runif(1, -3, 3),
                          z = zst, s = sst)}

params <- c('psi', 'sigma', 'N', 'D', 'alpha0', 'alpha1',
'beta1', 'sigma.eps', 'X1obs', 'X1new', 'X2obs', 'X2new',
'X3obs', 'X3new')



ni <- 200000
nb <- 100000
nt <- 100
nc <- 3

#*****************************************
# Do the MCMC stuff calling WinBUGS from R
#*****************************************

start <- as.POSIXlt(Sys.time())
print(start)


out <- jags(data = data, inits = inits, parameters.to.save =
params, model.file = "modelBinom.txt",
            n.chains = nc, n.thin = nt, n.iter = ni, n.burnin =
nb,parallel=T)


end <- as.POSIXlt(Sys.time())
```

```
print(end)

duration <- print(difftime(end,start,units='mins'))


print(out,3)

save(out,file='OutputSCR_het+dist.out')
```

APPENDIX 5. Scripts in R to conduct simulation analysis of the basic spatial capture-recapture model assuming no herd structure in data (SCR0).

```
## SIMULATE ELEPHANT SCR DATA
## Investigation of effects of aggregation on estimation

library(sp)
library(scrbook)
library(runjags)
library(coda)

# define state-space S, N, nocc
buffer <- 3
xlim <- c(0, 5)
ylim <- c(0, 6)
Xl <- xlim[1] - buffer
Xu <- xlim[2] + buffer
Yl <- ylim[1] - buffer
Yu <- ylim[2] + buffer
area <- (Xu - Xl) * (Yu - Yl)

N <- 450
nocc <- 10
nsim <- 50

# place for output
# fixed herds as random effects
sim.data <- matrix(NA, nrow = nsim, ncol = 12)
colnames(sim.data) <- c('simNo', 'Ntrue', 'Nest', 'NSD','Dtrue',
'Dest', 'DSD', 'T1', 'T2', 'T3', 'noHerds', 'avgHerdSz')

## Settings, model, parameters for JAGS - set only once
# BUGS settings
nc = 3
nAdaptSteps = 1000
nb = 1000
nu = 1000
nt = 30
ni = ceiling(nu*nt/nc)
ni

# model specification in BUGS
cat('
model{

# priors
alpha0 ~ dnorm(0, 0.1)
logit(p0) <- alpha0
sigma ~ dunif(0, 10)
alpha1 <- 1/(2*sigma*sigma)
psi ~ dunif(0, 1)
```

```
# likelihood
for(i in 1:M){

  z[i] ~ dbern(psi)
  s[i,1] ~ dunif(Xl, Xu)
  s[i,2] ~ dunif(Yl, Yu)

  for(j in 1:ntraps){
    y[i,j] ~ dbin(mu[i,j], K[j])
    mu[i,j] <- z[i]*p[i,j]
    mu2[i,j] <- mu[i,j]*K[j]
    p[i,j] <- p0*exp(-alpha1*d[i,j]*d[i,j])
    d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

    # discrepancy for Bayesian p-value GOF
    ynew[i,j] ~ dbin(mu[i,j],K[j])
    err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
    errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
  }
  expected[i] <- sum(mu2[i,])
  nsum[i] <- sum(y[i,])
  nsumnew[i] <- sum(ynew[i,])

  err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
  err1new[i] <- pow(pow(nsumnew[i], 0.5) - pow(expected[i],0.5),
2)
}

for(j in 1:ntraps){
  traptotals[j] <- sum(y[1:M,j])
  traptotalsnew[j] <-sum(ynew[1:M,j])
  expectedtrap[j] <- sum(mu[,j])*K[j]
  err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
  err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
pow(expectedtrap[j], 0.5), 2)
}

X3obs<- sum(err3[])
X3new<- sum(err3new[])
X2obs<- sum(err1[])
X2new<- sum(err1new[])
X1obs <- sum(err[,])
X1new <- sum(errnew[,])

# derived values
# no/dens herds
N <- sum(z[1:M])
D <- N/(area*25)
}
', file = 'modelNoHerd.txt')
```

```
params <- c('psi', 'sigma', 'N', 'D', 'alpha1',
            'X1obs', 'X1new', 'X2obs', 'X2new', 'X3obs', 'X3new')

## simulate and analyse multiple datasets

seed.no <- seq(2016, 2016+nsim-1)

begin <- as.POSIXlt(Sys.time())
for (b in 1:nsim){

set.seed(seed.no[b])

# population size and distribution of individuals across herds
herds1 <- rpois(N, 70)
herds2 <- table(herds1)
herds3 <-as.numeric(herds2)
nherds <- length(herds3)
herds <- sample(herds3, nherds)
#rmultinom(1, N, c(0.50, 0.25, 0.12, 0.07, 0.06))

# simulate activity centres
sx <- runif(nherds, Xl, Xu)
sy <- runif(nherds, Yl, Yu)
S <- cbind(sx, sy)

# Simulate encounter array
bb <- bbox(cbind(xlim, ylim))
clsz <- 1
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
encgrd.gt <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
encgrd.sg <- SpatialGrid(encgrd.gt)
encgrd.spt <- SpatialPoints(encgrd.sg)
traplocs <- coordinates(encgrd.spt)
ntraps <- nrow(traplocs)

## Assemble trap data file (TDF)
sim.tdf <- matrix(NA, nrow = ntraps, ncol = 3 + nocc)
colnames(sim.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
sim.tdf[,1] <- 1:ntraps
sim.tdf[,2:3] <- coordinates(encgrd.spt)
sim.tdf[,4:(nocc+3)] <- 1

# set mask
MASK <- sim.tdf[,4:ncol(sim.tdf)]
K <- apply(MASK, 1, sum)

# simulate prob(encounter) as fn of distance from act centre
p0 <- 0.08 # p0 = plogis(alpha0)
```

```
sigma <- 1.3
alpha1 <- 1/(2*sigma*sigma)
D <- e2dist(S, traplocs)
prob.enc <- p0*exp(-alpha1*D*D)

# simulate encounters with herds
Yherd <- array(NA, dim = c(nherds, ntraps, nocc))
for(i in 1:nherds){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      Yherd[i,j,k] <- rbinom(1, 1, prob.enc[i,j])
    }
  }
}

# simulate encounters with individuals given encounter with herd
herdID <- rep(1:nherds, herds)
Yind <- array(NA, dim = c(N, ntraps, nocc))
for(i in 1:N){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      if(Yherd[herdID[i],j,k] == 1) {Yind[i,j,k] <- rbinom(1, 1,
1/herds[herdID[i]])}
      else {Yind[i,j,k] <- 0}
    }
  }
}

# generate dataset of observed individuals
totalencs <- apply(Yind, 1, sum)
Yobs <- Yind[totalencs>0,,]
y <- apply(Yobs, c(1, 2), sum) # ind x trap -- summed across
occasion; required for sst

# data augmentation
M <- 1000
nind <- dim(Yobs)[1]
nz <- M - nind
y <- rbind(y, matrix(0, nrow = nz, ncol = ncol(y)))

# starting values for activity centres and DA
sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
}
zst <- c(rep(1, nind), rep(0, nz))

# specify data, initial values
data <- list(M = M,
             ntraps = ntraps,
             K = K,
             Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
```

```
              traplocs = traplocs,
              y = y,
              area = area)

inits <- function(){
  list(alpha0 = rnorm(1, 0, 0.1),
       sigma = runif(1, 0.4, 1),
       z = zst, s = sst)
}

cat('\n')
cat('###  SIMULATION ', b,' OF ', nsim,' ###\n')
cat('\n')

sim.out <- run.jags(method = 'parallel',
                    model = 'modelNoHerd.txt',
                    monitor = params,
                    data = data,
                    inits = inits,
                    n.chains = nc,
                    burnin = nb,
                    sample = ceiling(nu/nc),
                    adapt = nAdaptSteps,
                    thin = nt)
codaSamples <- as.mcmc.list(sim.out)

# extract sim no, N, D, 3 BPV tests, no herds, avg herd size
sim.data[b,'simNo'] <- b
sim.data[b,'Ntrue'] <- N
sim.data[b,'Nest'] <- sim.out$summary$statistics['N','Mean']
sim.data[b,'NSD'] <- sim.out$summary$statistics['N','SD']
sim.data[b,'Dtrue'] <- N/(area*25)
sim.data[b,'Dest'] <- sim.out$summary$statistics['D','Mean']
sim.data[b,'DSD'] <- sim.out$summary$statistics['D','SD']
bayespv.tmp <- rbind(sim.out$mcmc[[1]],
sim.out$mcmc[[2]],sim.out$mcmc[[3]])
sim.data[b,'T1'] <-
mean(bayespv.tmp[,'X1new']>bayespv.tmp[,'X1obs'])
sim.data[b,'T2'] <-
mean(bayespv.tmp[,'X2new']>bayespv.tmp[,'X2obs'])
sim.data[b,'T3'] <-
mean(bayespv.tmp[,'X3new']>bayespv.tmp[,'X3obs'])
sim.data[b,'noHerds'] <- nherds
sim.data[b,'avgHerdSz'] <- mean(herds)
save(sim.data, file = 'noHerd.out')
}
finish <- as.POSIXlt(Sys.time())
difftime(finish, begin, units = 'hours')

print(sim.out, digits = 4)
library(mcmcplots)
mcmcplot(codaSamples)
```

APPENDIX 5. Scripts in R to conduct simulation analysis of the basic spatial capture-recapture model using herd density and average herd size (SCRa).

```
## SIMULATE ELEPHANT SCR DATA
## Investigation of effects of aggregation on estimation

library(sp)
library(scrbook)
library(runjags)
library(coda)

# define state-space S, N, nocc
buffer <- 3
xlim <- c(0, 5)
ylim <- c(0, 6)
Xl <- xlim[1] - buffer
Xu <- xlim[2] + buffer
Yl <- ylim[1] - buffer
Yu <- ylim[2] + buffer
area <- (Xu - Xl) * (Yu - Yl)

N <- 450
nocc <- 10
nsim <- 50

# place for output
# fixed herds as random effects
sim.data <- matrix(NA, nrow = nsim, ncol = 12)
colnames(sim.data) <- c('simNo', 'Ntrue', 'Nest', 'NSD','Dtrue',
'Dest', 'DSD', 'T1', 'T2', 'T3', 'noHerds', 'avgHerdSz')

## Settings, model, parameters for JAGS - set only once
# BUGS settings
nc = 3
nAdaptSteps = 1000
nb = 1000
nu = 1000
nt = 30
ni = ceiling(nu*nt/nc)
ni

# model specification in BUGS
cat('
model{

# priors
alpha0 ~ dnorm(0, 0.1)
logit(p0) <- alpha0
sigma ~ dunif(0, 10)
alpha1 <- 1/(2*sigma*sigma)
psi ~ dunif(0, 1)
```

```
# likelihood
for(i in 1:M){

  z[i] ~ dbern(psi)
  s[i,1] ~ dunif(Xl, Xu)
  s[i,2] ~ dunif(Yl, Yu)

  for(j in 1:ntraps){
    y[i,j] ~ dbin(mu[i,j], K[j])
    mu[i,j] <- z[i]*p[i,j]
    mu2[i,j] <- mu[i,j]*K[j]
    p[i,j] <- p0*exp(-alpha1*d[i,j]*d[i,j])
    d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

    # discrepancy for Bayesian p-value GOF
    ynew[i,j] ~ dbin(mu[i,j],K[j])
    err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
    errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
  }
  expected[i] <- sum(mu2[i,])
  nsum[i] <- sum(y[i,])
  nsumnew[i] <- sum(ynew[i,])

  err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
  err1new[i] <- pow(pow(nsumnew[i], 0.5) - pow(expected[i],0.5),
2)
}

for(j in 1:ntraps){
  traptotals[j] <- sum(y[1:M,j])
  traptotalsnew[j] <-sum(ynew[1:M,j])
  expectedtrap[j] <- sum(mu[,j])*K[j]
  err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
  err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
pow(expectedtrap[j], 0.5), 2)
}

X3obs<- sum(err3[])
X3new<- sum(err3new[])
X2obs<- sum(err1[])
X2new<- sum(err1new[])
X1obs <- sum(err[,])
X1new <- sum(errnew[,])

# derived values
# no/dens herds
Nh <- sum(z[1:M])
Dh <- Nh/(area*25)

# no/dens individuals
```

```
Ni <- Nh*avgHerdSz
Di <- Ni/(area*25)
}
', file = 'modelHerdAvg.txt')

params <- c('psi', 'sigma', 'Nh', 'Dh', 'Ni', 'Di', 'alpha1',
            'X1obs', 'X1new', 'X2obs', 'X2new', 'X3obs', 'X3new')

## simulate and analyse multiple datasets
seed.no <- seq(2016, 2016+nsim-1)

begin <- as.POSIXlt(Sys.time())
for (b in 1:nsim){

set.seed(seed.no[b])

# population size and distribution of individuals across herds
herds1 <- rpois(N, 70)
herds2 <- table(herds1)
herds3 <-as.numeric(herds2)
nherds <- length(herds3)
herds <- sample(herds3, nherds)
#rmultinom(1, N, c(0.50, 0.25, 0.12, 0.07, 0.06))

# simulate activity centres
sx <- runif(nherds, Xl, Xu)
sy <- runif(nherds, Yl, Yu)
S <- cbind(sx, sy)

# Simulate encounter array
bb <- bbox(cbind(xlim, ylim))
clsz <- 1
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
encgrd.gt <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
encgrd.sg <- SpatialGrid(encgrd.gt)
encgrd.spt <- SpatialPoints(encgrd.sg)
traplocs <- coordinates(encgrd.spt)
ntraps <- nrow(traplocs)

## Assemble trap data file (TDF)
sim.tdf <- matrix(NA, nrow = ntraps, ncol = 3 + nocc)
colnames(sim.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
sim.tdf[,1] <- 1:ntraps
sim.tdf[,2:3] <- coordinates(encgrd.spt)
sim.tdf[,4:(nocc+3)] <- 1

# set mask
MASK <- sim.tdf[,4:ncol(sim.tdf)]
K <- apply(MASK, 1, sum)
```

```
# simulate prob(encounter) as fn of distance from act centre
p0 <- 0.08 # p0 = plogis(alpha0)
sigma <- 1.3
alpha1 <- 1/(2*sigma*sigma)
D <- e2dist(S, traplocs)
prob.enc <- p0*exp(-alpha1*D*D)

# simulate encounters with herds
Yherd <- array(NA, dim = c(nherds, ntraps, nocc))
for(i in 1:nherds){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      Yherd[i,j,k] <- rbinom(1, 1, prob.enc[i,j])
    }
  }
}

# simulate encounters with individuals given encounter with herd
herdID <- rep(1:nherds, herds)
Yind <- array(NA, dim = c(N, ntraps, nocc))
for(i in 1:N){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      if(Yherd[herdID[i],j,k] == 1) {Yind[i,j,k] <- rbinom(1, 1,
1/herds[herdID[i]])}
      else {Yind[i,j,k] <- 0}
    }
  }
}

# generate dataset of observed individuals
totalencs <- apply(Yherd, 1, sum)
Yobs <- Yherd[totalencs>0,,]
obsHerdSize <- herds[totalencs>0]
y <- apply(Yobs, c(1, 2), sum) # ind x trap -- summed across
occasion; required for sst

# data augmentation
M <- 1000
nind <- dim(Yobs)[1]
nz <- M - nind
y <- rbind(y, matrix(0, nrow = nz, ncol = ncol(y)))

# starting values for activity centres and DA
sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
}
zst <- c(rep(1, nind), rep(0, nz))

# specify data, initial values
```

```
data <- list(M = M,
             ntraps = ntraps,
             K = K,
             Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
             traplocs = traplocs,
             y = y,
             area = area,
             avgHerdSz = mean(obsHerdSize))

inits <- function(){
  list(alpha0 = rnorm(1, 0, 0.1),
       sigma = runif(1, 0.4, 1),
       z = zst, s = sst)
}

cat('\n')
cat('###  SIMULATION ', b,' OF ', nsim,' ###\n')
cat('\n')

sim.out <- run.jags(method = 'parallel',
                    model = 'modelHerdAvg.txt',
                    monitor = params,
                    data = data,
                    inits = inits,
                    n.chains = nc,
                    burnin = nb,
                    sample = ceiling(nu/nc),
                    adapt = nAdaptSteps,
                    thin = nt)
codaSamples <- as.mcmc.list(sim.out)

# extract sim no, N, D, 3 BPV tests, no herds, avg herd size
sim.data[b,'simNo'] <- b
sim.data[b,'Ntrue'] <- N
sim.data[b,'Nest'] <- sim.out$summary$statistics['Ni','Mean']
sim.data[b,'NSD'] <- sim.out$summary$statistics['Ni','SD']
sim.data[b,'Dtrue'] <- N/(area*25)
sim.data[b,'Dest'] <- sim.out$summary$statistics['Di','Mean']
sim.data[b,'DSD'] <- sim.out$summary$statistics['Di','SD']
bayespv.tmp <- rbind(sim.out$mcmc[[1]],
sim.out$mcmc[[2]],sim.out$mcmc[[3]])
sim.data[b,'T1'] <-
mean(bayespv.tmp[,'X1new']>bayespv.tmp[,'X1obs'])
sim.data[b,'T2'] <-
mean(bayespv.tmp[,'X2new']>bayespv.tmp[,'X2obs'])
sim.data[b,'T3'] <-
mean(bayespv.tmp[,'X3new']>bayespv.tmp[,'X3obs'])
sim.data[b,'noHerds'] <- nherds
sim.data[b,'avgHerdSz'] <- mean(herds)
save(sim.data, file = 'herdAvg.out')
}
finish <- as.POSIXlt(Sys.time())
difftime(finish, begin, units = 'hours')
```

```
print(sim.out, digits = 4)
library(mcmcplots)
mcmcplot(codaSamples)
```

APPENDIX 6. Scripts in R to conduct simulation analysis of the basic spatial capture-recapture model using herd identity as random effect (SCRh).

```
## SIMULATE ELEPHANT SCR DATA
## Investigation of effects of aggregation on estimation

library(sp)
library(scrbook)
library(runjags)
library(coda)

# define state-space S, N, nocc
buffer <- 3
xlim <- c(0, 5)
ylim <- c(0, 6)
Xl <- xlim[1] - buffer
Xu <- xlim[2] + buffer
Yl <- ylim[1] - buffer
Yu <- ylim[2] + buffer
area <- (Xu - Xl) * (Yu - Yl)

N <- 450
nocc <- 10
nsim <- 50

# place for output
# fixed herds as random effects
sim.data <- matrix(NA, nrow = nsim, ncol = 12)
colnames(sim.data) <- c('simNo', 'Ntrue', 'Nest', 'NSD','Dtrue',
'Dest', 'DSD', 'T1', 'T2', 'T3', 'noHerds', 'avgHerdSz')

## Settings, model, parameters for JAGS - set only once
# BUGS settings
nc = 3
nAdaptSteps = 1000
nb = 1000
nu = 3000
nt = 10
ni = ceiling(nu*nt/nc)
ni

# model specification in BUGS -- HERDS AS RANDOM EFFECTS
cat('
model{

# priors
gamma0 ~ dnorm(0, 0.01)
for (i in 1:nherds){
  gamma1[i] ~ dnorm(0, tau.g1)
}
tau.g1 <- 1/(sigma.g1*sigma.g1)
sigma.g1 ~ dunif(0, 10)
```

```
for (i in 1:nherds){
  qc[i] <- q0[i]/sum(q0[])
  q0[i] ~ dunif(0, 1)
}

for(i in 1:M){
  herdAug[i] ~ dcat(qc[])
}

sigma ~ dunif(0, 10)
alpha1 <- 1/(2*sigma*sigma)
psi ~ dunif(0, 1)

# likelihood
for(i in 1:M){
  z[i] ~ dbern(psi)
  s[i,1] ~ dunif(Xl, Xu)
  s[i,2] ~ dunif(Yl, Yu)

# random intercept
p0[i] <- exp(logit.p0[i])/(1 + exp(logit.p0[i]))
logit.p0[i] <- gamma0 + gamma1[herdAug[i]]

for(j in 1:ntraps){
  y[i,j] ~ dbin(mu[i,j], K[j])
  mu[i,j] <- z[i]*p[i,j]
  mu2[i,j] <- mu[i,j]*K[j]
  p[i,j] <- p0[i]*exp(-alpha1*d[i,j]*d[i,j])
  d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

  # discrepancy for Bayesian p-value GOF
  ynew[i,j] ~ dbin(mu[i,j],K[j])
  err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
  errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
}
expected[i] <- sum(mu2[i,])
nsum[i] <- sum(y[i,])
nsumnew[i] <- sum(ynew[i,])

err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
err1new[i] <- pow(pow(nsumnew[i], 0.5) - pow(expected[i],0.5), 2)
}

for(j in 1:ntraps){
  traptotals[j] <- sum(y[1:M,j])
  traptotalsnew[j] <-sum(ynew[1:M,j])
  expectedtrap[j] <- sum(mu[,j])*K[j]
  err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
  err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
```

```
pow(expectedtrap[j], 0.5), 2)
}

X3obs<- sum(err3[])
X3new<- sum(err3new[])
X2obs<- sum(err1[])
X2new<- sum(err1new[])
X1obs <- sum(err[,])
X1new <- sum(errnew[,])

# derived values
N <- sum(z[1:M])
D <- N/(area*25)
}
', file = 'modelRanEf.txt')

params <- c('psi', 'sigma', 'N', 'D', 'alpha1', 'gamma0',
'gamma1', 'X1obs', 'X1new', 'X2obs', 'X2new', 'X3obs', 'X3new')

## simulate and analyse multiple datasets

seed.no <- seq(2016, 2016+nsim-1)

begin <- as.POSIXlt(Sys.time())
for (b in 1:nsim){

set.seed(seed.no[b])

# population size and distribution of individuals across herds
herds1 <- rpois(N, 70)
herds2 <- table(herds1)
herds3 <-as.numeric(herds2)
nherds <- length(herds3)
herds <- sample(herds3, nherds)

# simulate activity centres
sx <- runif(nherds, Xl, Xu)
sy <- runif(nherds, Yl, Yu)
S <- cbind(sx, sy)

# Simulate encounter array
bb <- bbox(cbind(xlim, ylim))
clsz <- 1
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
encgrd.gt <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
encgrd.sg <- SpatialGrid(encgrd.gt)
encgrd.spt <- SpatialPoints(encgrd.sg)
traplocs <- coordinates(encgrd.spt)
ntraps <- nrow(traplocs)
```

```
## Assemble trap data file (TDF)
sim.tdf <- matrix(NA, nrow = ntraps, ncol = 3 + nocc)
colnames(sim.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
sim.tdf[,1] <- 1:ntraps
sim.tdf[,2:3] <- coordinates(encgrd.spt)
sim.tdf[,4:(nocc+3)] <- 1

# set mask
MASK <- sim.tdf[,4:ncol(sim.tdf)]
K <- apply(MASK, 1, sum)

# simulate prob(encounter) as fn of distance from act centre
p0 <- 0.08 # p0 = plogis(alpha0)
sigma <- 1.3
alpha1 <- 1/(2*sigma*sigma)
D <- e2dist(S, traplocs)
prob.enc <- p0*exp(-alpha1*D*D)

# simulate encounters with herds
Yherd <- array(NA, dim = c(nherds, ntraps, nocc))
for(i in 1:nherds){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      Yherd[i,j,k] <- rbinom(1, 1, prob.enc[i,j])
    }
  }
}

# simulate encounters with individuals given encounter with herd
herdID <- rep(1:nherds, herds)
Yind <- array(NA, dim = c(N, ntraps, nocc))
for(i in 1:N){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      if(Yherd[herdID[i],j,k] == 1) {Yind[i,j,k] <- rbinom(1, 1,
1/herds[herdID[i]])}
      else {Yind[i,j,k] <- 0}
    }
  }
}

# generate dataset of observed individuals
totalencs <- apply(Yind, 1, sum)
Yobs <- Yind[totalencs>0,,]
herdObs <- herdID[totalencs>0]
apply(Yobs, c(1, 3), sum) # ind x occ -- summed across traps, to
check if it looks okay
y <- apply(Yobs, c(1, 2), sum) # ind x trap -- summed across
occasion; required for sst

# data augmentation
M <- 1000
```

```
nind <- dim(Yobs)[1]
nz <- M - nind
y3d <- array(0, dim = c(M, ntraps, nocc))
for (j in 1:nind) {
  y3d[j, 1:ntraps, 1:nocc] <- Yobs[j, 1:ntraps, 1:nocc]
}
yAug <- apply(y3d, c(1, 2), sum)  # ind x trap -- summed across
occasion
herdNA <- rep(NA, nz) # for all-zero CHs
herdAug <- c(herdObs, herdNA)

# starting values for activity centres and DA
sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
}
zst <- c(rep(1, nind), rep(0, nz))

# specify data, initial values, parameters to monitor
data <- list(nherds = nherds,
             M = M,
             herdAug = herdAug,
             ntraps = ntraps,
             nocc = nocc,
             K = K,
             Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
             traplocs = traplocs,
             y = yAug,
             area = area)

inits <- function(){
  list(gamma0 = rnorm(1, 0, 0.01),
       gamma1 = rnorm(nherds, 0, 0.01),
       sigma.g1 = runif(1, 0, 10),
       q0 = runif(nherds, 0, 1),
       sigma = runif(1, 0.4, 1),
       z = zst, s = sst)
}

cat('\n')
cat('###  SIMULATION ', b,' OF ', nsim,' ###\n')
cat('\n')

sim.out <- run.jags(method = 'parallel',
                    model = 'modelRanEf.txt',
                    monitor = params,
                    data = data,
                    inits = inits,
                    n.chains = nc,
                    burnin = nb,
                    sample = ceiling(nu/nc),
                    adapt = nAdaptSteps,
```

```
                    thin = nt)
codaSamples <- as.mcmc.list(sim.out)

# extract sim no, N, D, 3 BPV tests, no herds, avg herd size
sim.data[b,'simNo'] <- b
sim.data[b,'Ntrue'] <- N
sim.data[b,'Nest'] <- sim.out$summary$statistics['N','Mean']
sim.data[b,'NSD'] <- sim.out$summary$statistics['N','SD']
sim.data[b,'Dtrue'] <- N/(area*25)
sim.data[b,'Dest'] <- sim.out$summary$statistics['D','Mean']
sim.data[b,'DSD'] <- sim.out$summary$statistics['D','SD']
bayespv.tmp <- rbind(sim.out$mcmc[[1]],
sim.out$mcmc[[2]],sim.out$mcmc[[3]])
sim.data[b,'T1'] <-
mean(bayespv.tmp[,'X1new']>bayespv.tmp[,'X1obs'])
sim.data[b,'T2'] <-
mean(bayespv.tmp[,'X2new']>bayespv.tmp[,'X2obs'])
sim.data[b,'T3'] <-
mean(bayespv.tmp[,'X3new']>bayespv.tmp[,'X3obs'])
sim.data[b,'noHerds'] <- nherds
sim.data[b,'avgHerdSz'] <- mean(herds)
save(sim.data, file = 'RanEff.out')
}
finish <- as.POSIXlt(Sys.time())
difftime(finish, begin, units = 'hours')

print(sim.out, digits = 4)
library(mcmcplots)
mcmcplot(codaSamples)
```

APPENDIX 7. Scripts in R to conduct simulation analysis of the basic spatial capture-recapture model using individual identity as random effect (SCRi).

```
## SIMULATE ELEPHANT SCR DATA
## Investigation of effects of aggregation on estimation

library(sp)
library(scrbook)
library(runjags)
library(coda)

# define state-space S, N, nocc
buffer <- 3
xlim <- c(0, 5)
ylim <- c(0, 6)
Xl <- xlim[1] - buffer
Xu <- xlim[2] + buffer
Yl <- ylim[1] - buffer
Yu <- ylim[2] + buffer
area <- (Xu - Xl) * (Yu - Yl)

N <- 450
nocc <- 10
seed.start <- 2016
nsim <- 25

# place for output
# fixed herds as random effects
sim.data <- matrix(NA, nrow = nsim, ncol = 12)
colnames(sim.data) <- c('simNo', 'Ntrue', 'Nest', 'NSD','Dtrue',
'Dest', 'DSD', 'T1', 'T2', 'T3', 'noHerds', 'avgHerdSz')

## Settings, model, parameters for JAGS - set only once
# BUGS settings
ni <- 150000
nb <- 50000
nt <- 100
nc <- 3

# model specification in BUGS -- Individual heterogeneity
cat('
model{

# priors

alpha1 <- 1/(2*sigma*sigma)
sigma ~ dunif(0.01, 10)
psi ~ dunif(0, 1)

alpha0 ~ dunif(-15, 5)
sigma.eps ~ dunif(0.1, 5)
tau.eps <- 1/(sigma.eps*sigma.eps)
```

```
# likelihood
for(i in 1:M){
  z[i] ~ dbern(psi)
  s[i,1] ~ dunif(Xl, Xu)
  s[i,2] ~ dunif(Yl, Yu)

  # random intercept
  p0[i] <- exp(logit.p0[i])/(1 + exp(logit.p0[i]))
  logit.p0[i] <- alpha0 + eps[i]
  eps[i] ~ dnorm(0, tau.eps)

  for(j in 1:ntraps){
    y[i,j] ~ dbin(mu[i,j], K[j])

    mu[i,j] <- z[i]*p[i,j]
    mu2[i,j] <- mu[i,j]*K[j]

    p[i,j] <- p0[i]*exp(-alpha1*d[i,j]*d[i,j])
    d[i,j] <- pow(pow(s[i,1] - traplocs[j,1], 2) + pow(s[i,2] -
traplocs[j,2], 2), 0.5)

    # discrepancy for Bayesian p-value GOF
    ynew[i,j] ~ dbin(mu[i,j],K[j])
    err[i,j] <- pow(pow(y[i,j], 0.5) - pow(K[j]*mu[i,j], 0.5), 2)
    errnew[i,j] <- pow(pow(ynew[i,j], 0.5) - pow(K[j]*mu[i,j],
0.5), 2)
  }

expected[i] <- sum(mu2[i,])
nsum[i] <- sum(y[i,])
nsumnew[i] <- sum(ynew[i,])

err1[i] <- pow(pow(nsum[i], 0.5)  - pow(expected[i], 0.5), 2)
err1new[i] <- pow(pow(nsumnew[i], 0.5) - pow(expected[i],0.5), 2)
}

for(j in 1:ntraps){
  traptotals[j] <- sum(y[1:M,j])
  traptotalsnew[j] <-sum(ynew[1:M,j])
  expectedtrap[j] <- sum(mu[,j])*K[j]
  err3[j] <- pow(pow(traptotals[j], 0.5) - pow(expectedtrap[j],
0.5), 2)
  err3new[j] <- pow(pow(traptotalsnew[j], 0.5) -
pow(expectedtrap[j], 0.5), 2)
}

X3obs<- sum(err3[])
X3new<- sum(err3new[])
X2obs<- sum(err1[])
X2new<- sum(err1new[])
X1obs <- sum(err[,])
X1new <- sum(errnew[,])
```

```
# derived values
N <- sum(z[1:M])
D <- N/(area*25)
}
', file = 'modelIndHet.txt')

params <- c('psi', 'sigma', 'N', 'D', 'alpha0', 'alpha1',
'sigma.eps', 'X1obs', 'X1new', 'X2obs', 'X2new', 'X3obs',
'X3new')

## simulate and analyse multiple datasets

seed.no <- seq(seed.start, seed.start+nsim-1)

begin <- as.POSIXlt(Sys.time())
set.seed(2017)

# population size and distribution of individuals across herds
herds1 <- rpois(N, 70)
herds2 <- table(herds1)
herds3 <-as.numeric(herds2)
nherds <- length(herds3)
herds <- sample(herds3, nherds)

# simulate activity centres
sx <- runif(nherds, Xl, Xu)
sy <- runif(nherds, Yl, Yu)
S <- cbind(sx, sy)

# Simulate encounter array
bb <- bbox(cbind(xlim, ylim))
clsz <- 1
cs <- c(clsz, clsz)
cc <- bb[,1] + (cs/2)
cd <- ceiling(diff(t(bb))/cs)
encgrd.gt <- GridTopology(cellcentre.offset = cc, cellsize = cs,
cells.dim = cd)
encgrd.sg <- SpatialGrid(encgrd.gt)
encgrd.spt <- SpatialPoints(encgrd.sg)
traplocs <- coordinates(encgrd.spt)
ntraps <- nrow(traplocs)

## Assemble trap data file (TDF)
sim.tdf <- matrix(NA, nrow = ntraps, ncol = 3 + nocc)
colnames(sim.tdf) <- c('trapID', 'xcoord', 'ycoord',
paste(1:nocc))
sim.tdf[,1] <- 1:ntraps
sim.tdf[,2:3] <- coordinates(encgrd.spt)
sim.tdf[,4:(nocc+3)] <- 1

# set mask
MASK <- sim.tdf[,4:ncol(sim.tdf)]
```

```
K <- apply(MASK, 1, sum)

# simulate prob(encounter) as fn of distance from act centre
p0 <- 0.08 # p0 = plogis(alpha0)
sigma <- 1.3
alpha1 <- 1/(2*sigma*sigma)
Dist <- e2dist(S, traplocs)
prob.enc <- p0*exp(-alpha1*Dist*Dist)

# simulate encounters with herds
Yherd <- array(NA, dim = c(nherds, ntraps, nocc))
for(i in 1:nherds){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      Yherd[i,j,k] <- rbinom(1, 1, prob.enc[i,j])
    }
  }
}

# simulate encounters with individuals given encounter with herd
herdID <- rep(1:nherds, herds)
Yind <- array(NA, dim = c(N, ntraps, nocc))
for(i in 1:N){
  for(j in 1:ntraps){
    for(k in 1:nocc){
      if(Yherd[herdID[i],j,k] == 1) {Yind[i,j,k] <- rbinom(1, 1,
1/herds[herdID[i]])}
      else {Yind[i,j,k] <- 0}
    }
  }
}

# generate dataset of observed individuals
totalencs <- apply(Yind, 1, sum)
Yobs <- Yind[totalencs>0,,]
herdObs <- herdID[totalencs>0]
y <- apply(Yobs, c(1, 2), sum) # ind x trap -- summed across
occasion; required for sst

# data augmentation
M <- 2000
nind <- dim(Yobs)[1]
nz <- M - nind
y3d <- array(0, dim = c(M, ntraps, nocc))
for (j in 1:nind) {
  y3d[j, 1:ntraps, 1:nocc] <- Yobs[j, 1:ntraps, 1:nocc]
}
yAug <- apply(y3d, c(1, 2), sum)  # ind x trap -- summed across
occasion
herdNA <- rep(NA, nz) # for all-zero CHs
herdAug <- c(herdObs, herdNA)

# starting values for activity centres and DA
```

```
sst <- cbind(runif(M, Xl, Xu), runif(M, Yl, Yu))
for(i in 1:nind){
  sst[i,1] <- mean(traplocs[y[i,]>0,1])
  sst[i,2] <- mean(traplocs[y[i,]>0,2])
}
zst <- c(rep(1, nind), rep(0, nz))

# specify data, initial values, parameters to monitor
data <- list(y = yAug,
             traplocs = traplocs,
             K = K,
             M = M,
             ntraps = ntraps,
             Xl = Xl, Xu = Xu, Yl = Yl, Yu = Yu,
             area = area)

inits <- function(){
  list(sigma = runif(1, 1, 2),
       psi = runif(1, 0, 1),
       alpha0 = runif(1, -2, 2),
       sigma.eps = runif(1, 1, 2),
       z = zst, s = sst)
}

#cat('\n')
#cat('###  SIMULATION ', b,' OF ', nsim,' ###\n')
#cat('\n')

start <- as.POSIXlt(Sys.time())
print(start)
out <- jags(data = data, inits = inits, parameters.to.save =
params, model.file = "modelIndHet.txt",
            n.chains = nc, n.thin = nt, n.iter = ni, n.burnin =
nb, parallel = T, verbose=T)
end <- as.POSIXlt(Sys.time())
print(end)
duration <- print(difftime(end,start,units='hours'))
save(out, file = 'IndHet.RData')

print(out, digits = 4)
mcmcplot(out$sims.list$psi)
mcmcplot(out$sims.list$sigma)
mcmcplot(out$sims.list$N)
mcmcplot(out$sims.list$D)
mcmcplot(out$sims.list$alpha0)
mcmcplot(out$sims.list$alpha1)
mcmcplot(out$sims.list$sigma.eps)

ntest <- 1000000
test.out <- matrix(NA, nrow = ntest, ncol = 8)
colnames(test.out) <- c('sigma', 'alpha1', 'sigma.eps',
'tau.eps', 'alpha0', 'eps', 'logit.p0', 'p0')
```

```
for(i in 1:ntest){

test.out[i, 'sigma'] <- runif(1, 0.05, 5) ##
test.out[i, 'alpha1'] <- 1/(2*test.out[i, 'sigma']*test.out[i,
'sigma'])

test.out[i, 'sigma.eps'] <- runif(1, 0.1, 5)
test.out[i, 'tau.eps'] <- 1/(test.out[i, 'sigma.eps']*test.out[i,
'sigma.eps'])

test.out[i, 'alpha0'] <- runif(1, -10, 5)

test.out[i, 'eps'] <- rnorm(1, 0, test.out[i, 'tau.eps'])

test.out[i, 'logit.p0'] <- test.out[i, 'alpha0'] + test.out[i,
'eps']
test.out[i, 'p0'] <- exp(test.out[i, 'logit.p0'])/(1 +
exp(test.out[i, 'logit.p0']))

}

summary(test.out[,'sigma'])
summary(test.out[,'alpha1'])
summary(test.out[,'sigma.eps'])
summary(test.out[,'tau.eps'])
summary(test.out[,'alpha0'])
summary(test.out[,'eps'])
summary(test.out[,'logit.p0'])
summary(test.out[,'p0'])
```