A Wildlife Simulation Package (*WiSP*)

Walter Zucchini[1], Martin Erdelmeier[1] and David Borchers[2]

[1] Institut für Statistik und Ökonometrie, Georg-August-Universität Göttingen, Platz der Göttinger Sieben 5, Göttingen, Germany

[2] Research Unit for Wildlife Population Assessment, Mathematical Institute, North Haugh, University of St Andrews, St Andrews, Fife, KY16 9SS, Scotland

Abstract. `WiSP` is an *R* library of functions designed as a teaching tool to illustrate methods used to estimate the abundance of closed wildlife populations. It enables users to generate animal populations having realistically complex spatial and individual characteristics, to generate survey designs for a variety of survey techniques, to survey the populations and to estimate the abundance. It can be used to assess properties of estimators when the model assumptions are violated.

1 Introduction

The problem of estimating the abundance of wildlife populations has generated an enormous literature describing a rich variety of methods (see e.g. Seber, 1982) necessitated by divergent physical, behavioural and demographic characteristics of animals (and plants) and considerations of the costs and practicability of surveying populations. Borchers, Buckland and Zucchini (2002) illustrate that, despite their superficial diversity, most of the standard methods are fundamentally similar and can be understood as variations on a single theme. The *Wildlife Simulation Package* (*WiSP*) was primarily designed to illustrate the ideas described in that book.

The main objective in developing *WiSP* was to create a convenient environment for students to experiment with different survey designs in an artificial but realistic setting, namely using generated populations with versatile spatial and animal-level characteristics. The advantage of course is that the true abundance is known and it is therefore possible to study how the properties of estimators are affected by the design parameters (survey effort, observer skill, etc.) and by specific violations of the model assumptions.

WiSP is a library of functions in *R* (Ihaka and Gentleman, 1996) that is available as free software under the terms of the Free Software Foundation's GNU General Public License (Hornik, 2002). Important advantages of *R* in our application are that it is easy for users of *WiSP* to modify the existing functions (e.g. change an estimator), to add new functions (e.g. a new survey design) and to utilize the full range of *R* tools for interactive data manipulation, calculation and graphical display. It has a standardized LaTeX-like documentation format for generating *help* documentation. A most convenient, if not critical, feature of *R,* from the point of

view of the system design for *WiSP,* is that it is an object-based envirnonment.

   This paper is organized as follows: Section 2 outlines the main concepts associated with the estimation of animal abundance, and the methods that are currently implemented in *WiSP*. Section 3 sketches the features of *WiSP* with special emphasis on the classes of objects that are defined in *WiSP*. Section 4 offers some concluding remarks.

2 Building Blocks

The model describing the state of a population, that is the spatial distribution and characteristics of the individual animals is called the ***state model***. The term *population* is more complex in the context of estimating animal abundance than it is in standard statistical investigations. In some cases information regarding specific characteristics of the individual animals must be collected during the survey. The individuals might congregate in groups of varying sizes; each animal has a gender, age, size, and so on. Such attributes, as well as behavioural factors, can affect the probability that a given animal will be detected or captured. The properties of abundance estimators depend on the state of the population; inappropriate assumptions about this state can result in misleading point and interval estimation. Thus for the purposes of comparing survey strategies or estimators in a simulated setting and (especially) for assessing the sensitivity of abundance estimators to violations in model assumptions, the software must be able to generate populations that are realistically complex in their spatial and animal-level characteristics.

   The ***observation model*** specifies our assumptions about the probabilistic process that determines which animals are detected or captured, given the search region, the positions and other characteristics of the animals as well as features of the survey (visibility conditions, survey effort, etc.). This model applies to the ***covered region*** (the sub-region that is searched), which is chosen using a set of rules, called the ***survey design***, and usually involves some randomisation. The surveyor has control of the survey design, e.g. the survey effort (number of traps, observers, how long they search, etc.), the shape of the survey units (rectangular, circular, strips, etc.) and the method to select them (simple random sampling, systematic sampling on a grid, etc.). *WiSP* currently implements the following methods:

   Plot sampling covers methods such as quadrat sampling and strip sampling, which, although different in their implementation, are effectively identical from a statistical point of view. They differ only in the shape of the survey units; it is assumed that animals in the covered region are detected with certainty.

   Removal, catch-effort and change-in-ratio methods require at least two survey occasions but involve no assumptions regarding the state model. The animals captured on each occasion are counted and removed from the population. The *removal* method assumes that capture probabilities are equal for all animals, that they remain the same on every survey occasion and that the captures are independent. These assumptions are relaxed in *catch-effort* models in which the capture probability can depend the catch-effort on each occasion; the estimates are based on changes in the catch-per-unit-effort rather on changes in the catch. *Change-in-ratio* methods constitute an additional refinement; these make use of changes in the proportions of animals of various types (sex, age, type, etc.)

following removals of known numbers of animals of each type

   Mark-recapture methods are similar to the removal methods but the captured animals are not removed; they are marked and replaced in the population. The methods vary in complexity depending on the model assumptions. In the simplest case one assumes that on each occasion each animal is equally likely to be captured; other models allow for various kinds of heterogeneity, for example that the capture probabilities depend on certain (observable) features of the captured animals, on their capture history and so on.

   Distance sampling methods estimate the relationship between the detection probability and the distance of detected animals to the centreline of strip of given width (line transect sampling) or to the centre of a circle of given radius (point transect sampling). It is assumed that detection is certain if the distance is zero. (This assumption is unrealistic for some animals (e.g. whales) and can be relaxed by using *double platform methods* in which two observers independently record their detections and the distance of the animals from the centreline.) Only a single survey is required; randomisation is usually applied to select the strips (or circles) that make up the covered region. Here too the methods vary in complexity depending on the assumptions made and the type of information collected.

   It is clear that both the nature and credibility of the assumptions underlying the above methods differ substantially; consequently the properties of the estimators will vary enormously. Of course that does not imply that the models with the least restrictive assumptions will perform best because, among other things, their performance depends on the sample size, which, if small, reverses the benefits gained by relaxing suspect assumptions. That is why it is important to compare the properties of estimators when the assumptions are violated.

   Currently *WiSP* can compute maximum likelihood point estimates and at least one type of interval estimate (non-parametric/parametric bootstrap, asymptotic normal-based or profile likelihood confidence intervals) for all simple cases, and for some of the more complex versions of the above models.

3 Design and object-classes in *WiSP*

We now outline the main components of *WiSP,* its design, object-classes and a few functions, relying on examples to illustrate some features of the library. The *WiSP* library is designed to simulate the abundance estimation process in its entirety. The main steps are specifying the survey region and the properties of the state model, generating a population, specifying the survey design and the observation model, conducting the survey and, finally, computing point and interval abundance estimates. These steps, which are illustrated in Figure 1, determine the main classes of objects defined in *WiSP,* namely *region, population, design, sample, estimate* objects and a number of parameter objects that specify the properties the state model, the observation model, the survey design and the method of estimation.

   In the current implementation a region is a simply a rectangle with given height and width. The parameters of the state model are specified using a function *setpars.population* that creates an object in the class *pars.population*. Objects in that class contain details of the probability distribution of the group sizes, the distribution of exposure (catchability/detectibility), the number and distribution of

user-specified types (e.g. female/male, size class, etc.) and options regarding the number of groups to be generated. Two of the required attributes of *population* objects are themselves objects, namely the *region* and the *density.population;* the latter controls the spatial distribution of the groups. Special functions are available to create and to modify *density.population* objects: *generate.density* creates a linear density surface over the region, *add.hotspot* increases/decreases the density around a given point; *set.stripe* changes the density in a rectangular sub-region. These functions enable the user generate populations having realistically complex spatial distributions (Figure 2).
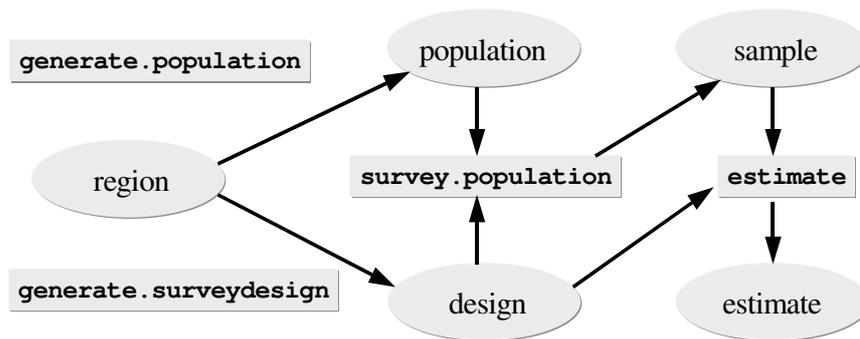

*Figure 1: Generic functions (rectangles) and objects (ovals) in **WiSP**.*

The observation model and the survey design are related because, *inter alia*, the latter determines where we look and how hard we look, but there is also an important distinction between them. The former determines the detection/capture probabilities, which can depend on properties of the population that are not taken into account by the survey design. It is precisely this distinction that enables us to assess the performance of the estimators when the model assumptions are violated. Thus two different classes of objects are needed: *design* objects (whose properties are defined by *design.parameter* objects created by appropriate functions) and *sample* objects created as a result of surveying the population using the given design. However, some properties of the observation process are controlled by parameter objects in a class *pars.survey*. Figure 3 illustrates this for a line transect example.

In practice the information available after completion of the survey comprises the survey design and the sample information because the latter depends on the survey method used. In plot sampling one merely has the counts in each plot; line transect samples must specify the distances of each detected group to the centreline and, possibly, other information about animal-level variables (e.g. group size) and/or survey-related variables (e.g. visibility conditions). Thus *sample* objects contain both design and sample information; they are passed to the appropriate estimation functions to create *estimate* objects that contain the relevant results, namely point

and interval estimates, graphical displays, diagnostics, etc. Figure 4 illustrates this for the example in Figure 3.
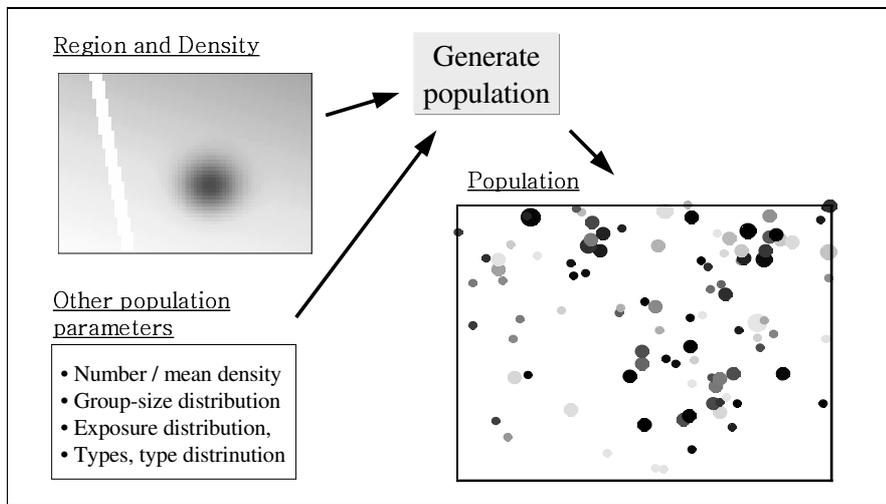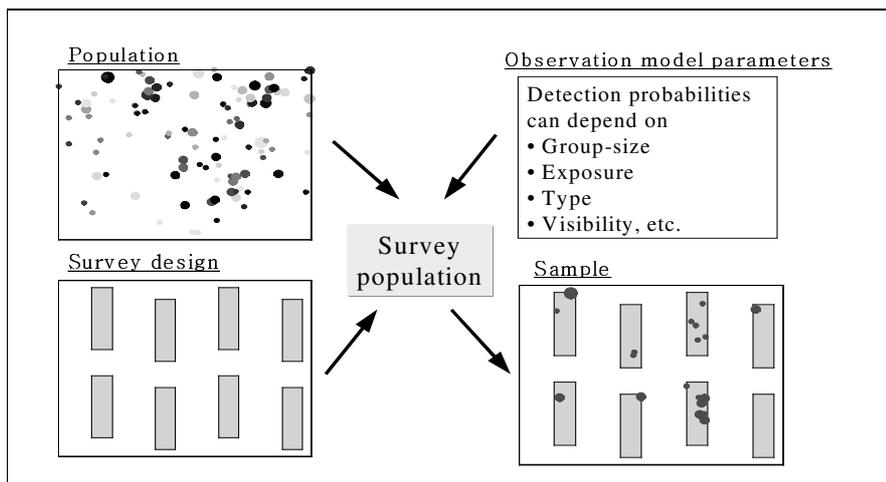

Figure 2: Generating a population.

Figure 3: The input objects needed to survey a population.

A very convenient feature of R is the facility to define generic functions for standard commands such as *print*, *plot*, *and summary*. One defines which properties of objects in a given class are to be printed, plotted or summarized after which the command *plot(object)*, for example, will generate the plot that is appropriate for the object. Similarly, the command as.population(<appropriate information>) will to coerce information into an object of the class *population* and *is.population* (*object*) checks if a given object is indeed in the class *population*.
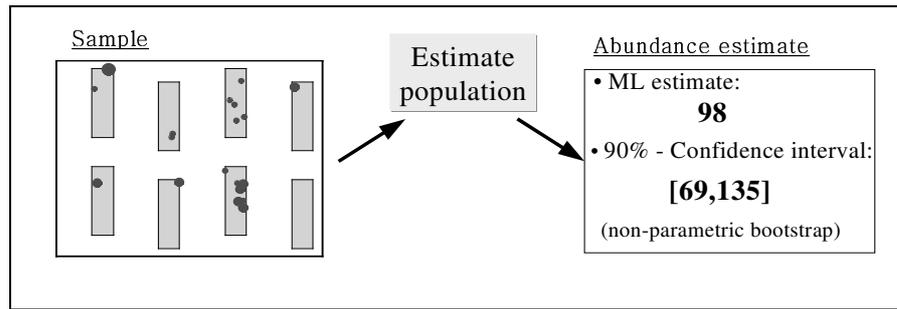
Figure 4: Estimates are computed using sample objects which must contain information about the survey design.

4 Concluding remarks

WiSP is primarily intended as a teaching tool to illustrate the basic principles underlying the most popular (closed population) abundance estimation techniques and to study their strengths and weaknesses. In its current form it does not have the functionality of the more comprehensive software packages available for the analysis of field data. (Useful lists of links can be found at www.phidot.org/software and www.mbr.nbs.gov/software.html.)

The free availability of **R** was an important consideration in selecting it as the language for **WiSP** but the other advantages that **R** offers were even more important in terms of future development of the **WiSP** library. The code can be made available in text form which makes it conveniently accessible to users who wish to modify it define new functions to cover a greater range of survey designs, estimators, or types of complexities in the state process. The object-orientated design of **WiSP** that is facilitated by **R** renders the task of making such modifications and enhancements incomparably less forbidding than would be the case if a language without this feature had been used.

References

Borchers, D.L. Buckland, S.T. & Zucchini W. (2002). *Estimating Animal Abundance: closed populations*, Springer-Verlag, London.

Hornik (2002), "The R FAQ", http://www.gnu.org/copyleft/gpl.html

Ihaka, Ross and Gentleman, Robert (1996), R: A Language for Data Analysis and Graphics, *Journal of Computational and Graphical Statistics*, 5(3), 299-314.

Seber, G.A.F. (1982), *The estimation of animal abundance and related* information *parameters: Second Edition*. Griffin, London.