

Introduction to distance sampling

Workshop, 21-23 August 2019

Centre for Research into Ecological and Environmental Modelling

Exercise 4. Variance estimation for systematic designs

In the lecture describing measures of precision, we explained that systematic survey designs usually have the best variance properties, but obtaining good estimates of the variance is a difficult problem for statisticians. In this exercise, we give an example of a situation where the systematic design gives a density estimate with much better precision than a random design. This means that the usual variance estimators used in the `ds` function, which are based on a random design, that are far too high. The true variance is low, but the estimated variance is high. We will see how to implement a post-stratification scheme that enables us to get a better estimate of the variance. We also look at another case to see that the unstratified variance estimates provided by `ds` are usually fine for a systematic design: things only go wrong when there are strong trends in animal density, especially when the strong trends are associated with changes in line length (e.g. the highest densities always occur on the shortest lines, or vice versa).

We begin with a population and survey shown below. The data used for this exercise were simulated on a computer: they are not real data. Note the characteristics for the data in Figure 1: extreme trends with very high density on short lines and very low density on long lines. Additionally, the systematic design has covered a fairly large proportion of the survey area (the covered region is shaded). These are danger signals that the usual `ds` variance estimators might not work well and a post-stratification scheme should be considered.

1 Objectives

The aims of this exercise are to illustrate:

1. Default variance estimation,
2. Variance estimation with bootstrapping,
3. Post-stratification to improve variance estimation,
4. When post-stratification is not needed (optional).

2 Getting started

Don't forget to load `Distance` if you haven't already done so.

```
# Load packages  
library(Distance)
```

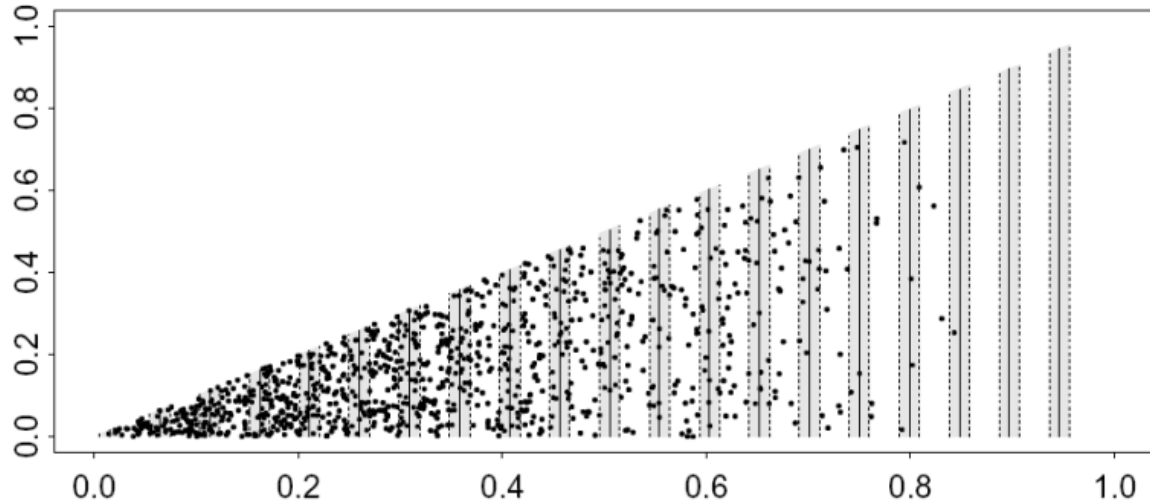


Figure 1: An example of survey data where there is a strong trend in density. The systematically placed search strips are shaded. Axis units are in kilometres.

3 Basic (default) variance estimation

In the code below, the necessary data file is imported and a simple model is fitted and a summary produced. Make a note of the CV of the density estimate - this is obtained using the default (analytical) estimator in the `ds` function and is based on the assumption that the lines were placed at random. This CV can then be compared with the CV estimates obtained from alternative methods.

```
# Import data
encounter.rate1 <- system.file("extdata", "IntroDS_4.1.csv", package = "dsdata")
sysvar1 <- read.csv(file=encounter.rate1, header=TRUE)
conversion.factor <- convert_units("metre", "kilometre", "square kilometre")
# Fit a simple model
sysvar1.hn <- ds(data=sysvar1, key="hn", adjustment=NULL,
                 convert.units=conversion.factor)
# Summary
sysvar1.hn$dht$individuals$D
sysvar1.hn$dht$individuals$N
```

The true density and abundance are known (because the data were simulated): the true abundance in the survey region was $N = 1000$ and $D = 2000$ animals per km^2 (i.e. 1000 animals in an area of size $A = 0.5\text{km}^2$). How do the point estimates compare with truth? What do you think about the precision of the estimates?

4 Variance estimation with bootstrapping

Before starting the bootstrap, we create a function to harvest the abundance and density estimates from each bootstrap replicate.

```
# Create a function to obtain abundance (N) and density (D) summaries
DNhat_summarize_indiv <- function(ests, fit) {
```

```

return(data.frame(D=ests$individuals$D$Estimate,
                  N=ests$individuals$N$Estimate))
}

```

The following command performs the bootstrap.

```

# Bootstrap estimate of uncertainty
# Run the bootstrap (this can take a while if nboot is large!)
est.boot <- bootdht(model=sysvar1.hn, flatfile=sysvar1,
                   summary_fun=DNhat_summarize_indiv,
                   convert.units=conversion.factor, nboot=999)

```

The arguments for this command are:

- `model` - fitted detection function model object
- `flatfile` - data frame of the survey data
- `summary_fun` - function used to obtain the summary statistics from each bootstrap
- `convert.units` - conversion units for abundance estimation
- `nboot` - number of bootstrap samples to generate. Note, it can take a long time to produce a large number of bootstraps and so perhaps try a small number at first.

```

# See the results
summary(est.boot)

```

The summary includes:

- `Estimate` - the median value of the bootstrap estimates
- `se` is the standard deviation of the bootstrap estimates
- `lcl` and `ucl` are the limits for a 95% confidence interval.
- `cv` is the coefficient of variation ($CV = SE/Estimate$)

Are the bootstrapped confidence intervals for abundance and density similar to the analytical confidence intervals produced previously?

Recall that we have a particular situation in which we have systematically placed transects which are unequal in length. Furthermore, there exists an east-west gradient in animal density juxtaposed such that the shortest lines are those that pass through the portion of the study region with the highest density. In the next section, we examine a process by which we can use post-stratification to produce a better estimate of the variance in estimated abundance.

5 Post-stratification to improve variance estimation

The estimation of encounter rate variance in Exercise 4.1 used estimators that assumed the transect lines were randomly placed throughout the triangular region. In our case, the transects were not random, but systematic and, in some circumstances, taking this in account can substantially reduce the encounter rate variance. The data we are working with is an example of this, where there are very high densities on the very shortest lines. In samples of lines, collected using a completely random design, the sample, by chance, might not contain any very short lines, or it might contain several. The variance is therefore very high, because the density estimates will be greatly affected by how many lines fall into

the short-line / high-density region: we will get very low density estimates if there are no short lines, but very high density estimates if there are several short lines. By contrast, in a systematic sample, we cover the region methodically and we will always get nearly the same number of lines falling in the high density region. The systematic density variance is therefore much lower than the random placement density variance. Although there is no way of getting a variance estimate that is exactly unbiased for a systematic sample ¹, we can greatly improve on the random-based estimate by using a post-stratification scheme.

The post-stratification scheme works by grouping together pairs of adjacent lines from the systematic sample and each pair of adjacent lines is grouped into a stratum. The strata will improve variance estimation, because the systematic sample behaves more like a stratified sample than a random sample. This encounter rate estimator is called ‘O2’ (Fewster et al. 2009) and is implemented in the `dht2` function.

```
# Post-stratification - stratified variance estimation by grouping adjacent transec

# Ensure that Sample.Labels are numeric, this is required for O2 ordering
sysvar1$Sample.Label <- as.numeric(sysvar1$Sample.Label)

# Use the Fewster et al 2009, "O2" estimator
est.O2 <- dht2(sysvar1.hn, flatfile=sysvar1, strat_formula=~1,
               convert_units=conversion.factor, er_est="O2")
print(est.O2, report="density")
```

Note that this estimator assumes that the numbering of the transects (in this example `Sample.Label` takes values 1 to 20) has some geographical meaning (i.e. transect 1 is next to 2 and 2 is next to 3 etc.). If this is not the case, then the user can manually define some sensible grouping of transects and create a column called `grouping` in the data object.

6 Systematic designs where post-stratification is not needed (optional)

The simulated population shown in Figure 2 does not exhibit strong trends across the survey region, otherwise, the strip dimensions and systematic design are the same as for the previous example. These data are stored in `IntroDS_4.2.csv`.

In the code below, these data are imported into R and a simple detection function model is fitted. The default estimate of variance is then compared to that obtained using the ‘O2’ estimator (Fewster et al. 2009).

```
# When post-stratification isn't needed

# Import the data
encounter.rate2 <- system.file("extdata", "IntroDS_4.2.csv", package = "dsdata")
sysvar2 <- read.csv(file=encounter.rate2, header=TRUE)
# Ensure that Sample.Labels are numeric, for O2 ordering
sysvar2$Sample.Label <- as.numeric(sysvar2$Sample.Label)
```

¹because it is effectively a sample of size 1- only the first line position was randomly chosen and the rest followed on deterministically from there.

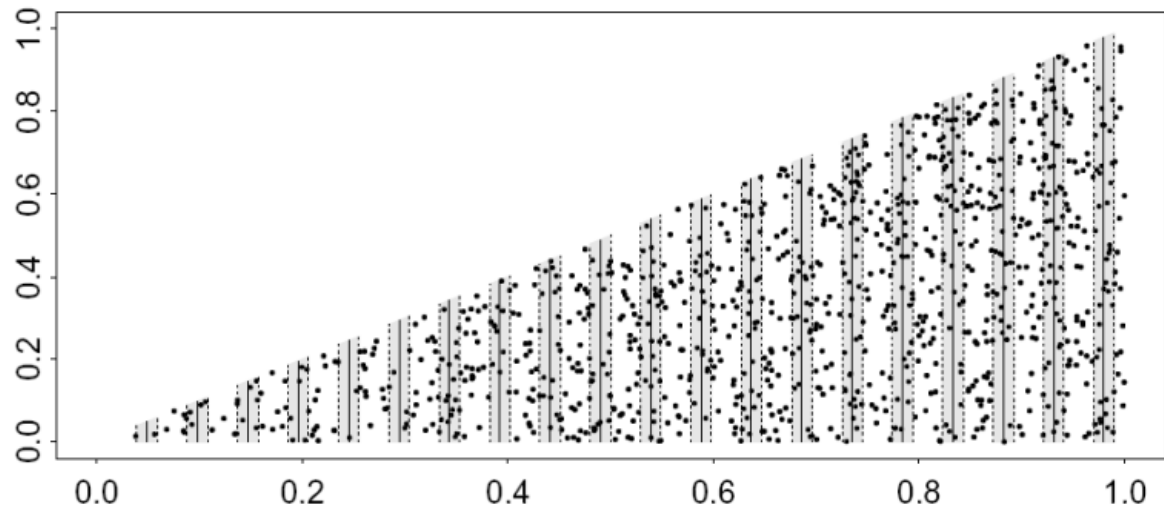


Figure 2: An example of survey data that does not exhibit a trend in density. The systematically placed search strips are shaded. Axis units are in kilometres.

```
# First fit a simple model
sysvar2.hn <- ds(sysvar2, key="hn", adjustment=NULL,
                 convert_units=conversion.factor)
# Obtain default estimates for comparison
sysvar2.hn$dht$individuals$D
sysvar2.hn$dht$individuals$N
# Now use Fewster et al 2009, "02" estimator
est2.02 <- dht2(sysvar2.hn, flatfile=sysvar2, strat_formula=~1,
                 convert_units=conversion.factor, er_est="02")
print(est2.02, report="both")
```

Did you see a difference in the CV and 95% confidence interval between the two estimators?

7 References

Fewster RM, Buckland ST, Burnham KP, Borchers DL, Jupp PE, Laake JL and Thomas L (2009) Estimating the encounter rate in distance sampling. *Biometrics* 65:225-236

Solution 4. Variance estimation for systematic designs

8 Basic (default) variance estimation

Recall the data for this example, in which we have a strong gradient in animal density across our study region and at the same time we have a difference in the lengths of the transects, such that short transects are in regions of high animal density and long transects are in regions of low animal density.

```
library(Distance)
# Import data
encounter.rate1 <- system.file("extdata", "IntroDS_4.1.csv", package = "dsdata")
sysvar1 <- read.csv(file=encounter.rate1, header=TRUE)
conversion.factor <- convert_units("metre", "kilometre", "square kilometre")
# Fit a simple model
sysvar1.hn <- ds(data=sysvar1, key="hn", adjustment="cos",
                 convert.units=conversion.factor)
# Summary
sysvar1.hn$dht$individuals$D

##   Label Estimate      se      cv      lcl      ucl      df
## 1 Total 2044.592 566.3958 0.2770214 1161.012 3600.614 20.74468

sysvar1.hn$dht$individuals$N

##   Label Estimate      se      cv      lcl      ucl      df
## 1 Total 1022.296 283.1979 0.2770214 580.506 1800.307 20.74468
```

The point estimates are good ($\hat{D} = 2,044$ animals per unit area and $\hat{N} = 1,022$ - note the size of the area) but the precision obtained with the default estimator is poor: estimated abundance ranges from about 580 to 1,800 - a three-fold difference over which we are uncertain. Given that our survey covered 40% of the triangular region and had a good sample size (254 animals on 20 transects), this would be a disappointing result in practice.

9 Variance estimation with bootstrapping

Before starting the bootstrap, we create a function to harvest the abundance and density estimates from each bootstrap sample.

```
# Create a function to obtain abundance (N) and density (D) summaries
DNhat_summarize_indiv <- function(ests, fit) {
  return(data.frame(D=ests$individuals$D$Estimate,
                    N=ests$individuals$N$Estimate))
}
```

The following command performs the bootstrap.

```
# Bootstrap estimate of uncertainty
# Run the bootstrap (this can take a while!)
est.boot <- bootdht(model=sysvar1.hn, flatfile=sysvar1,
                    summary_fun=DNhat_summarize_indiv,
                    convert.units=conversion.factor, nboot=99)
```

```
# See results
summary(est.boot)
```

```
## Bootstrap results
##
## Boostrops          : 99
## Successes          : 99
## Failures           : 0
##
## Estimate      se      ucl      lcl      cv
## D  2075.84 688.55 3636.01 1008.62 0.33
## N  1037.92 344.27 1818.01  504.31 0.33
```

The bootstrap results are very similar to the analytical results, as we would expect, because again this process assumed the transects were placed at random.

10 Post-stratification to improve variance estimation

```
## Post-stratification by O2 estimator

# ensure that Sample.Labels are numeric, for O2 ordering
sysvar1$Sample.Label <- as.numeric(sysvar1$Sample.Label)

# Using the Fewster et al 2009, "O2" estimator
est.O2 <- dht2(sysvar1.hn, flatfile=sysvar1,
               strat_formula=~1, convert_units=conversion.factor, er_est="O2")
print(est.O2, report="density")
```

```
## Summary statistics:
## .Label Area CoveredArea Effort   n   k      ER se.ER cv.ER
## Total  0.5      0.1922   9.61 254 20 26.431 1.459 0.055
##
## Density estimates:
## .Label Estimate      se   cv      LCI      UCI      df
## Total 2044.592 162.914 0.08 1744.988 2395.636 75.871
##
## Component percentages of variance:
## .Label Detection      ER
## Total      52.03 47.97
```

The precision of the estimated abundance has greatly improved in the post-stratified analysis.

It must be remembered that we have not made any change to our data by the post-stratification; we are using getting a better estimate of the variance. In this case, the increase in precision could make a fundamental difference to the utility of the survey: it might make the difference between being able to make a management decision or not. Usually, trends will not be as extreme as they are in this example and post-stratification will not make a great difference. Such an example is illustrated in the next problem.

11 Systematic designs where post-stratification is not needed (optional)

These data did not exhibit strong trends across the survey region and, hence, there are no great differences between the CVs and 95% confidence intervals using the two methods.

```
# Import the data
encounter.rate2 <- system.file("extdata", "IntroDS_4.2.csv", package = "dsdata")
sysvar2 <- read.csv(file=encounter.rate2, header=TRUE)
# Ensure that Sample.Labels are numeric, for 02 ordering
sysvar2$Sample.Label <- as.numeric(sysvar2$Sample.Label)
# First fit a simple model
sysvar2.hn <- ds(sysvar2, key="hn", adjustment=NULL,
                 convert.units=conversion.factor)
# Obtain default estimates for comparison
sysvar2.hn$dht$individuals$D

##   Label Estimate      se      cv      lcl      ucl      df
## 1 Total 1954.016 160.5554 0.08216691 1657.275 2303.888 50.59549

sysvar2.hn$dht$individuals$N

##   Label Estimate      se      cv      lcl      ucl      df
## 1 Total 977.0078 80.27771 0.08216691 828.6377 1151.944 50.59549

# Now use Fewster et al 2009, "02" estimator
est2.02 <- dht2(sysvar2.hn, flatfile=sysvar2, strat_formula=~1,
                convert_units=conversion.factor, er_est="02")
print(est2.02, report="density")

## Summary statistics:
##   .Label Area CoveredArea Effort   n   k   ER se.ER cv.ER
##   Total 0.5      0.2058 10.29 252 20 24.49 1.594 0.065
##
## Density estimates:
##   .Label Estimate      se      cv      LCI      UCI      df
##   Total 1954.015 162.491 0.083 1653.804 2308.723 49.172
##
## Component percentages of variance:
##   .Label Detection      ER
##   Total      38.76 61.24
```