

Analyzing camera trap data with BUGS

Important!

This practical uses the data on golden cats in Peninsular Malaysia analysed with PRESENCE in a previous practical. You should work through the PRESENCE analysis before tackling BUGS; you can access it at the [wcsmalaysia.org](http://www.wcsmalaysia.org) website:

http://www.wcsmalaysia.org/stats/Golden_cats_PRESENCE.htm

We will also assume that you are familiar with the concepts of Bayesian analysis, in particular the concept of probability as degree-of-belief. You may want to look at the [wcsmalaysia.org](http://www.wcsmalaysia.org) page on “Bayes in Brief”:

http://www.wcsmalaysia.org/stats/Bayes_in_Brief.htm

A. Formatting the data for OpenBUGS

Download and install OpenBUGS as described on the [wcsmalaysia.org](http://www.wcsmalaysia.org) web site (http://www.wcsmalaysia.org/stats/Software_summary.htm#BUGS).

Download the data files in Golden_cats_BUGS.zip from [wcsmalaysia.org](http://www.wcsmalaysia.org) and extract them to a suitable place on your hard drive.

Open the file Golden_cats_BUGS.xls in MS Excel or a similar spreadsheet program.

The first worksheet, “raw data”, is the same as that in Golden_cats_PRESENCE.xls. The second worksheet, “input for BUGS”, is also very similar to the input we used for presence. Each row corresponds to a camera trap site; we have excluded sites in plantations and those where the camera malfunctioned.

- The first column indicates whether the site is in logged forest (1) or unlogged, primary forest (0); the first 169 sites are in primary forest, the others in logged forest. This column will be imported into BUGS as a vector called “Logged”: the square brackets in the column heading indicate to BUGS that this is a vector.
- The remaining columns correspond to camera trapping ‘occasions’, each covering 5 days, with 1 if golden cats were photographed and 0 if they were not. One camera was out for 17 x 5 days, so there are 17 columns of capture data; for the other sites the columns are filled with NA where data are Not Available. These columns will be imported into BUGS as a matrix called “h” (for capture History): the comma and number in the square brackets in the heading tell BUGS which column of the matrix to use.
- The last row has “END” in capital letters in the first column.

We need to save these data in a tab-delimited text file.

Go to File > Save As... and change Save as type: at the foot of the dialog box to “Text (Tab delimited) (*.txt)”. Give the file a suitable name, such as Golden_cats_BUGS.txt and press Save. Press OK and Yes in the confirmation boxes which pop up. Then close the file *without* saving.

Now open the file Golden_cats_BUGS.txt (or whatever you called it) in a text editor such as Notepad. You’ll see that Excel has put quotes around the column headings which contain a comma (this is necessary for comma delimited files, but not for tab delimited files). Delete all the quotes: use Edit > Replace..., putting a quotation mark in the first box (Find what:) and leaving the second box (Replace with:) blank. Save and close the file.

B. A first analysis

We will begin with a model which uses the habitat covariable (Logged vs Primary forest) for both occupancy and detection probability. This is the same as the “psi(habitat), p(habitat)” model in PRESENCE.

Start BUGS and open the file “GCat model psi(habitat) p(habitat).odc”.

If you have used R (or S-plus), you will find that the BUGS code looks familiar. Don’t be fooled, though, there are some important differences. To begin with, the code specifies the connections between the variables, not the operations to carry out, so the order of the lines doesn’t matter in BUGS.

The model to be simulated is defined in the curly-brackets following ‘model’. The ‘for i in 1:162’ loop deals with each of the 162 sites in turn, telling BUGS that:

- The value for psi for each site is calculated from the Logged covariate and the coefficients a[1] and a[2], so that
 - if Logged[i] = 0 (ie. primary forest), $\text{logit}(\text{psi}[i]) = a[1]$
 - if Logged[i] = 1 (ie. logged forest), $\text{logit}(\text{psi}[i]) = a[1] + a[2]$

Note that we can use the logit() function on the left of the assignment arrow in BUGS.

- The occupancy state for each site, the 0s and 1s in z, is modelled as (the ~ symbol means ‘modelled as’) a Bernoulli distribution with probability of occupancy = psi.
- The vector tmp is the same as z but using 1 and 2 instead of 0 and 1; we’ll use this to choose the value of the detection probability, p, in the last step.
- Detection probabilities, p, are in a matrix with 2 rows:
 - Row 1, p[1,], is the detection probability to use if the site is not occupied (if tmp = 1); it is filled with 0s.
 - Row 2, p[2,] is used if the site is occupied; it is calculated from the Logged covariate and the coefficients b[1] and b[2], again using the logit function.
- The modelled detection histories in h, a matrix with 162 rows and 17 columns, use a Bernoulli distribution and the appropriate row of p for each site.

The model definition also includes the prior distributions to use for the coefficients, a[1], a[2], b[1] and b[2]. The priors are all defined as very broad normal distributions with mean 0 and sd 10. BUGS uses an unusual convention: the second term in dnorm is the precision (tau), defined as 1/sd; so tau = 0.1 corresponds to sd = 10.

(Ignore the lines beginning with the hash, #, for the moment.)

Now we’ll get BUGS to run the model.

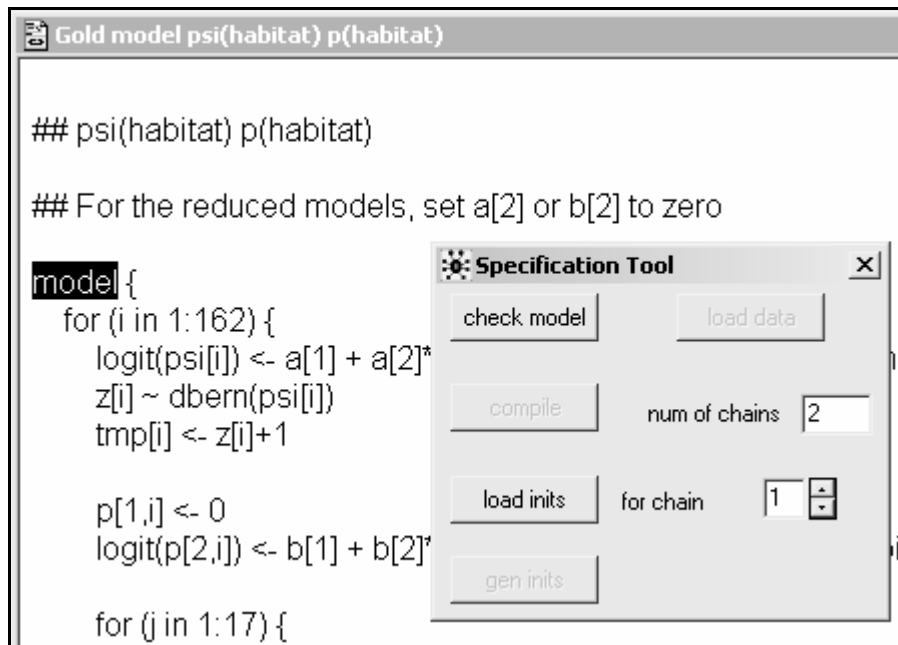
Click on Model > Specification... to open the Specification Tool.

In the window headed “GCat model psi(habitat) p(habitat)”, double click on the word ‘model’ before the curly bracket to highlight it (see screen-shot below), then press ‘check model’ in the Specification Tool. Look at the status bar at the bottom of the BUGS window and you should see “model is syntactically correct”.

Now go to File > Open... and open the file with the data, “Golden_cats_BUGS.txt” (or whatever you called it). Place the cursor anywhere in the window with the data and press ‘load data’. You should see ‘data loaded’ in the status bar.

In the Specification Tool, change the ‘num of chains’ to 2, then press ‘compile’; ‘model compiled’ should appear.

We’ll get BUGS to generate the initial values for the simulation: press ‘gen inits’. The status should now show ‘initial values generated, model initialized’.



Now you can close the Specification Tool: press Ctrl-F4 or click on the  in the top left corner.

Next we need to tell BUGS what we want to monitor. For the first few updates we'll only monitor a and b: if they have converged the other values will be fine:

Click on Inference > Samples... to open the Sample Monitor Tool.

In the 'node' box, type 'a', then press 'set'. Do the same for 'b'.

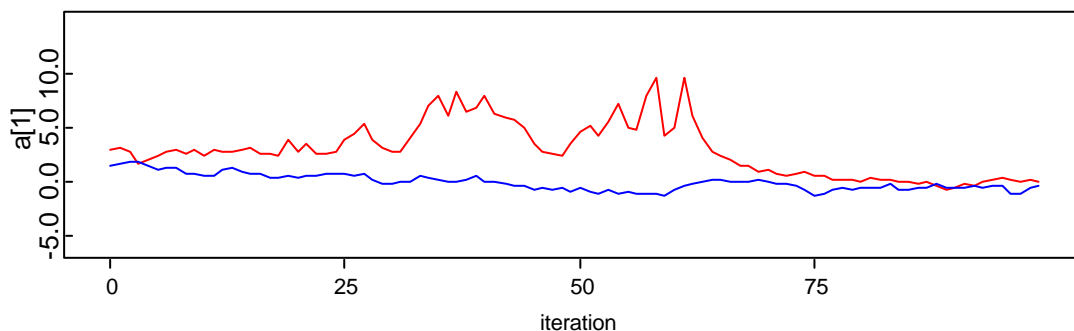
Now we're all set to run some simulations! We'll just do 100 to begin with:

Click on Model > Update... to open the Update Tool.

Change the value in the 'updates' box to 100 and press 'update'. The status bar should show 'model is updating' for a while, then change to '100 updates took 3 s' (or maybe longer if you have a slow machine).

In the Sample Monitor Tool, change the 'node' to a and press 'history'. Then do the same with b.

You will see histories for 100 iterations for a[1], a[2], b[1] and b[2], looking rather like the graph below. These are based on random numbers, so will be different each time the model is run.



BUGS is running 2 separate Markov chains with different starting values, each shown as line with a different colour in the graphs. The results we get should in the long term be the same for both chains. The graph above indicates that 100 iterations is not 'long term' and we need to do many more.

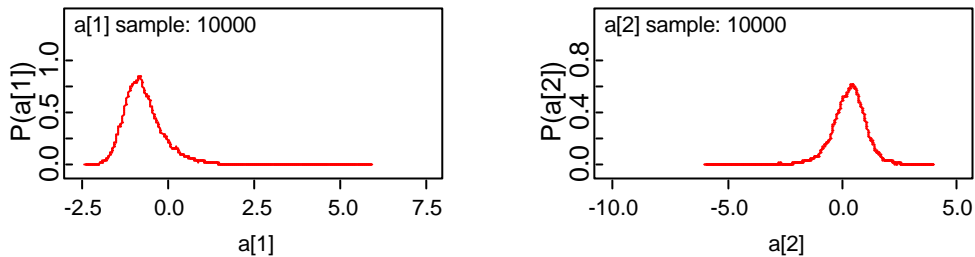
Change the value in the 'updates' box to 900 and press 'update' again. This will take longer so be patient! When it's done, check the histories for a and b again.

1000 iterations should be enough to get convergence with this model. We'll treat these first 1000 iterations as burn-in and do more to estimate the actual values. For these we want estimates for psi, p and z as well as a and b.

In the 'node' box of the Sample Monitor Tool, type 'psi', then press 'set'. Do the same for 'p' and 'z'.

Change the value in the 'updates' box to 5000 and press 'update' again. You can watch the progress in the 'iteration' box in the Update Tool. This might be a good time to take a coffee break!

When it's done, change the 'beg' (= beginning) value in the Sample Monitor Tool to 1001. You can check the histories if you wish, but no problems should crop up at this stage. Look at the densities and the stats for a and b.



	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
a[1]	-0.6264	0.7473	0.04285	-1.592	-0.7598	1.2	1001	10000
a[2]	0.2619	0.8728	0.04582	-1.71	0.3366	1.778	1001	10000

The density curves are the posterior probability density functions. The horizontal axis gives a range of values for a[1] or a[2], and the vertical axis indicates the probability of the corresponding value. (Remember that Bayesians use 'probability' to mean 'how certain we are'.) So the most probable value of a[1] is about -0.6, and any value below -1.5 or above 1.2 is pretty improbable. The interpretation of the a[2] density plot is similar.

The 'stats' give the numbers corresponding to the posterior probabilities – these are based on random numbers, so change slightly each time the model is run. BUGS reports the mean (and standard deviation) and the median of the posterior distribution, and the 95% credible interval ('val2.5pc' is the lower limit and 'val97.5pc' is the upper limit). The median is easily interpreted: the probability of higher values = the probability of lower values. 'Monte Carlo error' is the error introduced by the random sampling procedure; that should be much smaller than the sd, at most 5%; this is a bit big and we should do more iterations to reduce this. We'll look at the other results before deciding to invest time in that.

It looks as if the effect of the Logged variable is not very big. Remember that a[2] represents the difference in occupancy between the Logged and Primary habitats on the logit scale. We can convert that to an odds ratio:

$$OR = e^{0.26} = 1.3$$

A word about odds and odds ratios, in case you aren't familiar with them: the odds of occupancy is:

$$\begin{aligned} & \text{probability occupied} / \text{probability not occupied} \\ & = \text{psi} / (1 - \text{psi}) \end{aligned}$$

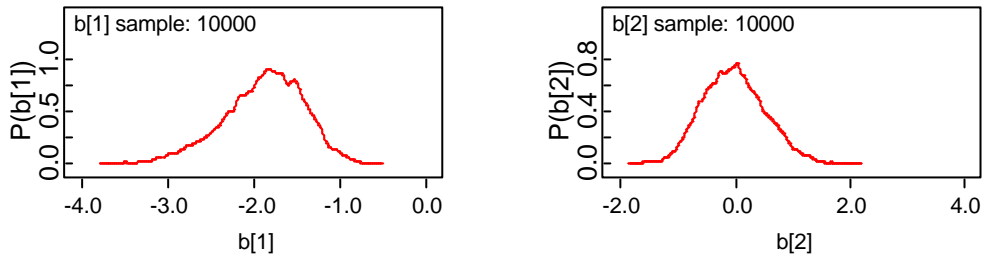
So if $\text{psi}_{\text{Primary}} = 0.35$, odds of occupancy = $0.35 / (1 - 0.35) = 0.35 / 0.65 = 0.54$ for Primary habitat.

The odds ratio is odds of occupancy in Logged habitat / odds of occupancy in Primary habitat. If $OR = 1.3$, the odds of occupancy in Logged habitat is $0.54 \times 1.3 = 0.70$. We can convert back to probability with:

$$\begin{aligned} \text{psi} & = \text{odds of occupancy} / (1 + \text{odds of occupancy}) \\ & = 0.70 / (1 + 0.70) = 0.70 / 1.70 = 0.41 \end{aligned}$$

So $\text{psi}_{\text{Logged}} = 0.41$. The difference between 0.35 and 0.41 is small when compared with the 95% CrI for psi.

An odds ratio of 1 means no difference, so this is not very big. Moreover, the credible interval for $a[2]$ straddles 0: we would not be surprised if the true value were zero.



	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
b[1]	-1.897	0.4682	0.02365	-2.935	-1.853	-1.083	1001	10000
b[2]	-0.01863	0.5428	0.0261	-1.002	-0.03999	1.098	1001	10000

Here the MC error is just on 5%. However the effect of the Logging variable on detection is clearly negligible, even if not actually zero. So we should look at a model without this effect.

If you look at the density for ψ , BUGS will produce 172 little graphs. But the first 69 of these are all identical (these are Primary habitat sites) and the last 101 are also identical (Logged sites). We can simplify things by putting $\psi[69:70]$ in the node box in the Sample Monitor Tool.

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
$\psi[69]$	0.3529	0.1467	0.008162	0.1692	0.3187	0.7685	1001	10000
$\psi[70]$	0.4123	0.1095	0.004397	0.2449	0.3966	0.6845	1001	10000

The difference between these is small and the credible intervals overlap – in fact the CrI for Logged sites is inside that for Primary sites.

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
$p[2,69]$	0.1392	0.05213	0.002376	0.05046	0.1355	0.253	1001	10000
$p[2,70]$	0.1318	0.03204	0.001006	0.07239	0.1307	0.1984	1001	10000

Here the difference is tiny, as we saw when looking at $b[2]$.

Finally, we'll look at z , the variable which indicates whether each site is occupied or not. The density "curves" are not very interesting, as the only possible values are 0 and 1. The stats are more useful, though here too the median and CrI are meaningless:

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
$z[1]$	0.174	0.3791	0.006797	0.0	0.0	1.0	1001	10000
$z[2]$	0.175	0.38	0.006731	0.0	0.0	1.0	1001	10000
$z[3]$	0.1893	0.3917	0.007025	0.0	0.0	1.0	1001	10000
$z[4]$	0.1921	0.394	0.006825	0.0	0.0	1.0	1001	10000
$z[5]$	0.225	0.4176	0.007364	0.0	0.0	1.0	1001	10000
$z[6]$	0.2024	0.4018	0.006025	0.0	0.0	1.0	1001	10000
$z[7]$	0.2063	0.4046	0.006535	0.0	0.0	1.0	1001	10000
$z[8]$	1.0	0.0	7.071E-13	1.0	1.0	1.0	1001	10000
$z[9]$	0.1723	0.3776	0.005944	0.0	0.0	1.0	1001	10000
$z[10]$	0.3162	0.465	0.006543	0.0	0.0	1.0	1001	10000
$z[11]$	1.0	0.0	7.071E-13	1.0	1.0	1.0	1001	10000

and so on...

For some sites the mean is 1, with $sd = 0$. If you cross-check with the raw data, you'll see that golden cats were photographed at these sites, so indeed we are sure that they are occupied. For the other sites, our opinion of whether they are occupied or not depends on the number of trapping occasions: sites 1 and 2 had cameras operating for 8 occasions ($= 8 \times 5 = 40$ nights) while site 10 had cameras out for 1 occasion only, so we doubt if sites 1 or 2 are occupied, but we would not be surprised if site 10 was occupied; since we have little data for site 10, the value here is just a little less than the overall value for ψ for Primary habitat. These results correspond to the conditional probability of occupancy in PRESENCE, the probability of occupancy given no detections.

C. Saving the results

Warning! BUGS does not save your results when it closes! If you have changed the model file or the data, it will ask if you want to save it, but the results will disappear forever the moment you close BUGS. That can be galling if you were running simulations which took 20 minutes to do 1000 updates.

To save the results, you need to copy/paste into a new document which you can then save with its own file name. You can use a Word document for this (as I did to produce this guide) or you can open a new file in BUGS and copy graphs and stats to that, adding whatever explanations and comments you want.

It's a good idea to save everything you want and close BUGS before running a different model, otherwise you can get confused about which results relate to which model.

D. Simplifying the model

The obvious first step in simplifying the model is to take out the habitat effect from detection probability and to try the model: $\text{psi}(\text{habitat}), p(\cdot)$.

To do that we need to change the model. We could delete the “+ b[2]*Logged[i]” at the end of the definition of $\text{logit}(p[2,i])$. That would be fine for this simple model, but for a complicated model with several covariates and many coefficients it is easy to make a mistake. Moreover, if you removed a variable completely from the model, you would have to take it out of the data file too, or BUGS would generate an error. A neater solution is to force b[2] to be 0.

Open the file “GCat model psi(habitat) p(habitat).odc” again in BUGS. Near the bottom of the file move the # symbol in the last two lines, so that they look like this:

```
#      b[2] ~ dnorm(0.0, 0.1)
      b[2] <- 0
```

Now BUGS will ignore the dnorm line and use the <- 0 line. Run the new model as before:

With the Model Specification Tool: Check the model
Load the data
Change the number of chains to 2, compile
Generate initial values.

With the Sample Monitor Tool: Set 'a' and 'b' in the node box

With the Update Tool: Run 1000 updates (burn in)

Sample Monitor Tool again: Check the histories for 'a' and 'b' (two for a, but now only one for b).
Set psi, p and z in the node box

Update Tool again: Run another 5000 updates

Sample Monitor Tool again: Change 'beg' to 1001
Look at the results

Here I'll just pick out the main results:

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
a[1]	-0.7347	0.4331	0.01284	-1.52	-0.7538	0.1693	1001	10000
a[2]	0.3234	0.5026	0.01291	-0.6273	0.3124	1.33	1001	10000

The MC_errors are now okay. The a[2] coefficient is still small and the 95% CrI still includes 0. Let's look at the 50% CrI as well:

In the Sample Monitor Tool, hold down the Ctrl key and click on “25” and “75” in the percentile box to highlight them. Then press stats.

	mean	sd	MC_error	val2.5pc	val25.0pc	median	val75.0pc	val97.5pc	start	sample
a[1]	-0.7347	0.4331	0.01284	-1.52	-1.03	-0.7538	-0.4603	0.1693	1001	10000
a[2]	0.3234	0.5026	0.01291	-0.6273	-0.005566	0.3124	0.6446	1.33	1001	10000

The 25% percentile for a[2] is practically zero, so there's a 25% probability that a[2] is negative and the odds ratio < 1 (ie. occupancy is actually lower in Logged habitat than in Primary habitat). There is also a 25% probability that the value of a[2] is higher than 0.6446 (the 75% percentile), which corresponds to an odds ratio of:

$$OR = e^{0.6446} = 1.9$$

In short, we're still sceptical about the effect of Logging on occupancy.

	mean	sd	MC_error	val2.5pc	val25.0pc	median	val75.0pc	val97.5pc	start	sample
psi[69]	0.3305	0.09374	0.002815	0.1794	0.2632	0.32	0.3869	0.5422	1001	10000
psi[70]	0.4014	0.09831	0.003506	0.2434	0.3342	0.39	0.454	0.629	1001	10000

The difference is small and the 50% CrIs overlap. So the next step should be to run the psi(.), p(.) model.

Check that there is just one value for b, b[1], and b[2] is not reported; also that the value for p[2,] is the same for all sites.

Set up and run the psi(.), p(.) model.

Change the model by setting a[2] <- 0 as well as b[2] <- 0 and run it as before.

The results all seem to be in order, so we can report our conclusions about psi and p:

	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
psi[69]	0.3639	0.0736	0.002214	0.2432	0.3561	0.5317	1001	10000
psi[70]	0.3639	0.0736	0.002214	0.2432	0.3561	0.5317	1001	10000
	mean	sd	MC_error	val2.5pc	median	val97.5pc	start	sample
p[2,69]	0.1368	0.02725	7.225E-4	0.08649	0.1361	0.1935	1001	10000
p[2,70]	0.1368	0.02725	7.225E-4	0.08649	0.1361	0.1935	1001	10000

Occupancy (psi) is estimated as 0.36 (mean and median) with a 95% Credible Interval of [0.24, 0.53] and probability of detection is estimated as 0.14 with 95% CrI [0.09, 0.19].

The results will vary slightly each time you run the model in BUGS; differences will be smaller if you run more updates, so if you have time try running 20,000 or 100,000 instead of 5,000 before finalising your report.

E. Comparison of results from BUGS and PRESENCE

The analysis in PRESENCE also indicated that the best model was psi(.), p(.), with no effect of habitat on either detection or occupancy. This was based on AIC, an information-theoretic measure derived from likelihoods. The Bayesian equivalent is the Deviance Information Criterion (DIC), which BUGS can produce for suitable models; unfortunately DIC can't be computed for models which contain discrete nodes (the problem is z, which can only take the values 0 and 1).

For comparison, the psi(.), p(.) model in PRESENCE gave psi = 0.35 with 95% CI [0.23, 0.48] and p = 0.14 with 95% CI [0.09, 0.20]. We would be alarmed if the results were not closely similar, but we would not expect them to be identical, because we have used a different approach to the analysis.

The main difference between Bayesian and non-Bayesian methods is that Bayesian results (ie. posterior distributions) depend on prior information as well as the data. In this case, however, we used very 'flat' priors, so the posterior distributions are virtually the same as the likelihood curves.

In this situation, the main sources of difference are:

- PRESENCE uses maximum likelihood estimation, ie. it looks at the peak (mode) of the curve, not the mean or the median, which are reported in BUGS. These are only the same if the curve is symmetrical.
- PRESENCE estimates the variance (and s.d.) of the curve from the shape of the peak. Inferences based on the shape of the tails – including confidence intervals and p-values – involve assuming a specific shape for the curve, usually a normal or t-distribution. BUGS, on the other hand, explicitly explores the whole posterior distribution.

There are also of course conceptual differences. For example, we are 95% sure that the right answer lies within the Bayesian 95% credible interval; the frequentist confidence interval, however, involves imagining a huge number of camera trap surveys of golden cats at randomly selected sites in Malaysia, all carried out and analysed in the same way, whereupon we can say that 95% of the confidence intervals thus calculated will contain the true value.