downscale: An R Package for Downscaling Species Occupancy from Coarse-Grain Data to Predict Occupancy at Fine-Grain Sizes

Charles J. Marsh
University of Leeds

Louise J. Barwell
Centre for Ecology
and Hydrology

Yoni Gavish
University of Leeds

William E. Kunin
University of Leeds

Abstract

The geographical area occupied by a species is a valuable measure for assessing its conservation status. Coarse-grained occupancy maps are available for many taxa, e.g., as atlases, but often at spatial resolutions too coarse for conservation use. However, mapping occupancy at fine spatial resolution across the entire extent of the species' distribution is often prohibitively expensive for the majority of species. Occupancy downscaling is a technique to estimate finer scale occupancy from coarse scale maps, by using the occupancy-area relationship (OAR) which reflects how the proportion of area occupied increases with spatial grain size. Models that describe the OAR are fitted to observed occupancies at the available coarse-grain sizes and then extrapolated to predict occupancy at the finer grain sizes required. The **downscale** package in the R programming environment provides users with easy-to-use functions for downscaling occupancy with ten published models. First, `upgrain` calculates occupancy for multiple grain sizes larger than the input data. Normal methods for aggregating raster data increase the extent of the focal area as grain size increases which is undesirable, so the function fixes the extent for all grain sizes, assigning unsampled cells as absences. Four suggested methods are provided to enable this and `upgrain.threshold` provides diagnostic plots that allow the user to explore the inherent trade-off between making assumptions about unsampled locations and discarding information from sampled locations. `downscale` fits nine possible models to the data generated from `upgrain`. `hui.downscale` fits the special case of the Hui model. `predict` and `plot` extrapolate the fitted models to predict and plot occupancy at finer grain sizes. Finally, `ensemble.downscale` simultaneously fits two or more of the downscaling models and calculates mean predicted occupancy across all selected models. Here we describe the package and apply the functions to atlas data of a hypothetical UK species.

Keywords: area of occupancy, atlas data, conservation, IUCN Red List, occupancy-area relationship, R.

1. Introduction to downscaling species occupancy

The geographic range size of a species is an important measure of its ecology and conservation status (Gaston 1994). For example, the area of occupancy (AOO) is used to measure of extinction risk in the IUCN red list (IUCN 2012). Although easier to estimate than true abundance, the difficulty in estimating AOO lies in its scale-sensitivity: species appear to occupy more area when surveyed at coarse spatial resolutions than they do when finer grids are employed (Hartley and Kunin 2003). Relatively coarse-grained atlases are available for many taxa, as they can be generated with reasonable precision by aggregating opportunistically collected or systematically sampled biodiversity records. For example 12 UK atlases have been published in the last ten years alone (Powney and Isaac 2015). However, conservation decisions require rather finer-resolution information. Thus, for example, IUCN guidelines recommend a grain size of 4 km^2, and no larger than 10 km^2 (IUCN Standards and Petitions Subcommittee 2017) for estimating AOO, and arguably finer still maps would be appropriate for sessile or sedentary species with small home ranges. To create such fine resolution maps would require extensive and meticulous surveying across the full range of each species, which is impractical or unfeasible for the majority of species and locations.

A possible solution is to employ the occupancy-area relationship (OAR, or scale-area relationship), that is the increase in the area occupied by a species as grain size increases (Kunin 1998), to translate occupancy information across scales (Figure 1). The shape of this curve is indicative of the spatial distribution of the species: a shallow slope indicates a species with an aggregated distribution whereas a steep slope indicates a species with a dispersed pattern. If
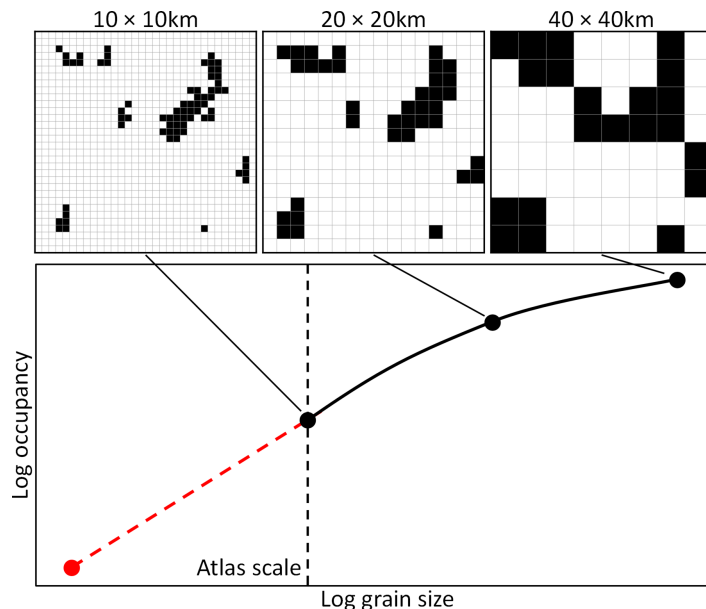

Figure 1: The process of downscaling species occupancy. A map of species occupancy (black cells) is taken at a large grain size such as atlas data (10km × 10km). The map is aggregated to larger grain sizes and a model fitted to describe the relationship between the increase in occupancy with the increase in grain size (black lines and points). The model is then extrapolated to predict occupancy at a finer grain size (red lines and points).

Model	Code	Equation	Source
Nachman	`"Nachman"`	$1 - e^{-cA^z}$	Nachman (1981)
Power law	`"PL"`	cA^z	Kunin (1998)
Logistic	`"Logis"`	$\dfrac{cA^z}{1 + cA^z}$	Hanski and Gyllenberg (1997)
Poisson	`"Poisson"`	$1 - e^{-\gamma A}$	Wright (1991)
Negative binomial	`"NB"`	$1 - (1 + \frac{\gamma A}{k})^{-k}$	He and Gaston (2000)
Generalized negative binomial	`"GNB"`	$1 - (1 + \frac{cA^z}{k})^{-k}$	He *et al.* (2002)
Improved negative binomial	`"INB"`	$1 - [c(\gamma A)^{b-1}]^{\frac{\gamma A}{1 - c(\gamma A)^{b-1}}}$	He and Gaston (2003)
Finite negative binomial	`"FNB"`	$1 - \frac{\Gamma(N + \frac{A_0 k}{A} - k)\Gamma(\frac{A_0 k}{A})}{\Gamma(N + \frac{A_0 k}{A})\Gamma(\frac{A_0 k}{A}) - k}$	Zillio and He (2010)
Thomas	`"Thomas"`	see text	Azaele *et al.* (2012)
Hui	`"Hui"`	see text	Hui *et al.* (2006, 2009); Hui (2009)

Table 1: Downscaling models implemented in the **downscale** package to predict logged occupancy. The fitted parameters b, c and z are constants, γ is mean density, k is an over-dispersion parameter and N the total number of individuals. A is the logged grain size and A_0 is the extent of the study area.

the relationship can be described for occupancies at readily-available coarse grain sizes where confidence is high and the proportion of false absences is low, then we can extrapolate the OAR to predict occupancy at the fine grain sizes necessary for conservation assessments, a process called occupancy *downscaling*.

Models for downscaling species occupancy

Many models have been proposed to model the OAR, and it appears that no one model consistently provides the best predictions (Azaele, Cornell, and Kunin 2012; Barwell, Azaele, Kunin, and Isaac 2014). In the package described here we implement ten published models (Nachman, power law, logistic, Poisson, negative binomial, generalized negative binomial, improved negative binomial, finite negative binomial, Thomas and Hui models). Details of the models can be found in Table 1, as well as in the original literature and the supplementary information of Barwell *et al.* (2014). The power law, Nachman and logistic models (Table 1) are three closely related models that seek to extrapolate the OAR slope at larger resolutions to predict occupancies at finer resolutions. Four of the models (negative binomial, improved negative binomial, finite negative binomial and generalized negative binomial models, Table 1) are based around the negative binomial distribution, and thus incorporate information on mean density and degree of aggregation at various resolutions. Interestingly, these four models

are also related to the Poisson model which assumes independence of individuals. Finally, the generalized negative binomial model can be reduced to the power law, logistic, Nachman, Poisson or negative binomial models through specific combinations of the parameters (He *et al.* 2002). None-the-less, despite differences in underlying theory, all these eight models work by first fitting a specific function to the OAR over several coarse scales and then extrapolating it down to finer resolutions.

The Thomas model (Azaele *et al.* 2012), rather than having a single equation that describes the OAR, incorporates spatial point processes to allow for a more flexible approach to including species aggregations. The spatial point process implemented is the shot noise Cox process: $P(A) = 1 - exp\{-\rho \int [1 - exp(-\mu \int_A k(||\vec{c} - \vec{x}||)d\vec{x})]d\vec{c}\}$, where $k(||\vec{x}||)$ is an isotropic bivariate Gaussian distribution with variance σ^2. In order to simplify the model several key assumptions are made: μ is a constant; there is translational and rotational invariance; the geometry of the study region is smoothed so that a rectangular spatial window is used in the computation of the spatial point process; there is temporal stationarity; and the model uses a simple form for the pair correlation function.

The Hui model, as a spatially-explicit model, requires species occupancy at only one spatial grain. It uses conditional probabilities (joint-count statistics) using two estimated probabilities: the probability that a randomly chosen cell is occupied; and the probability that a cell adjacent to an occupied cell will also be occupied. As the occupancy of a coarse-grain cell is the combination of occupancies of multiple fine-grain cells (a percolation process), Bayes' theorem is then applied to predict the number of occupancies at any finer grain size. Thorough details of the model may be found in Hui *et al.* (2006), Hui *et al.* (2009), and Hui (2009).

2. The downscale package

Here we describe the package **downscale** for the statistical language R (R Core Team 2018) that makes available the ten downscaling models described above along with functions for preparing coarse-scale data, creating maps of occupancy at increasing grain sizes, plotting results, and an ensemble method for running multiple models and averaging their predictions. During modeling the free parameters for the models (minus the Hui model) are estimated by numerical minimization of the sum square of predicted occupancy in log space, $\frac{1}{N} \cdot \sum_{A=1}^{g} (\log p_{A,i}^{pred} - \log p_{A,i}^{obs})^2$, where $p_{A,i}^{pred}$ and $p_{A,i}^{obs}$ are the predicted and observed occupancy at grain A for species i and g is the number of observed grain sizes. Minimization is carried out using a modification of the Levenberg-Marquardt algorithm in the **minpack.lm** package (Elzhov, Mullen, Spiess, and Bolker 2016). For the Hui model, root solving is carried out by the `uniroot` function. Integration in the Thomas model is carried out using the **cubature** package (Johnson and Narasimhan 2017). In the finite negative binomial model, where the multiplication of multiple gamma functions may result in vectors larger than R storing capacity, we use multiple precision floating point numbers (function `mpfr` in package **Rmpfr**, Mächler (2018)). Spatial manipulation is carried out using the **raster** (Hijmans 2017) and **sp** (Pebesma and Bivand 2005; Bivand, Pebesma, and Gomez-Rubio 2013) packages. The package is distributed on CRAN at https://CRAN.R-project.org/package=downscale and can be installed through:

```
R> install.packages("downscale")
R> library("downscale")
```
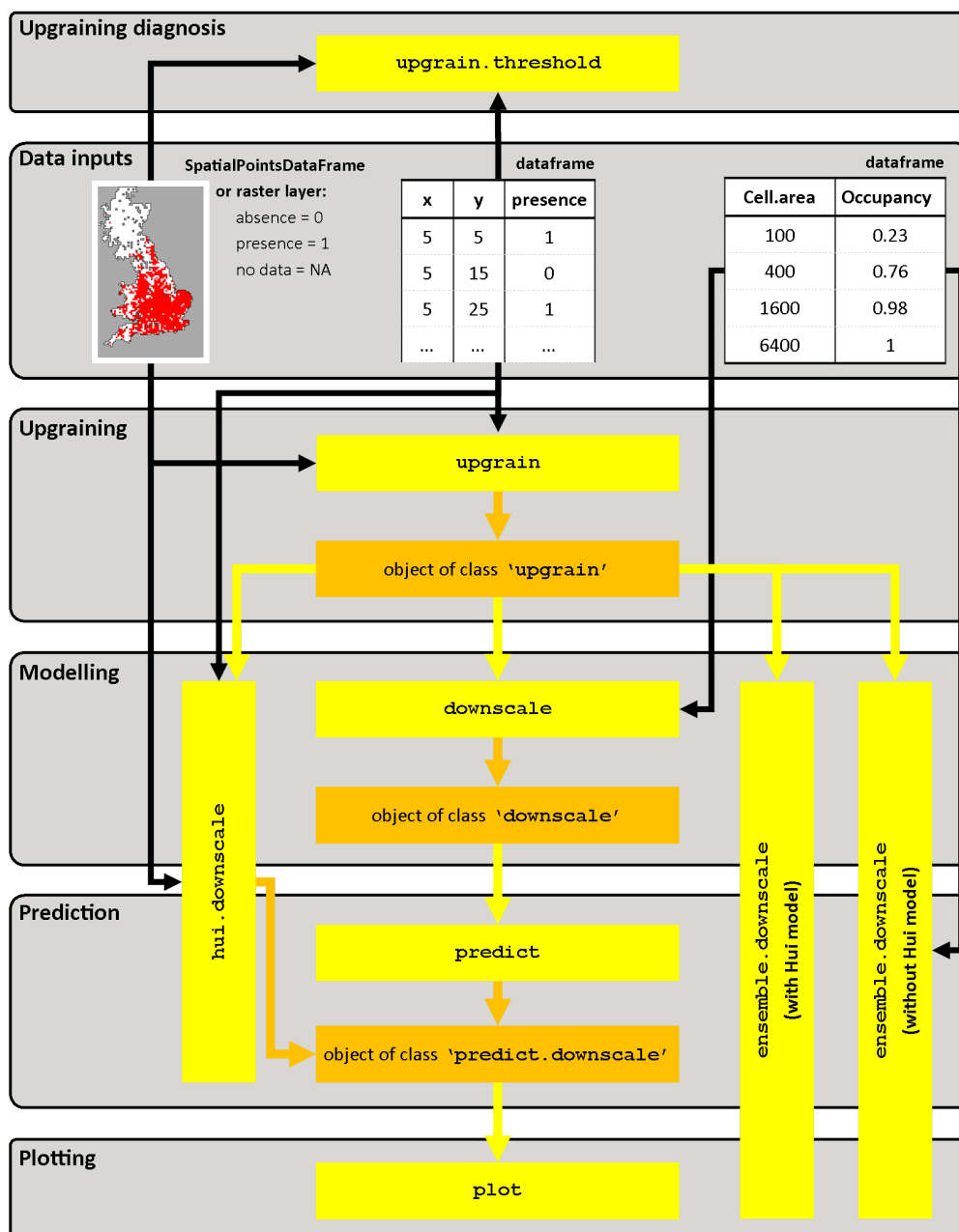
Figure 2: Structure of the **downscale** package showing all eight functions (yellow) and the three output object classes (orange).

The general flow of the **downscale** package is presented in Figure 2. The user may input four types of data: (a) A raster layer of presence-absence data (presence = 1; absence = 0; no data = NA). (b) An object of class `SpatialPointsDataFrame` with a data frame with column `presence` containing presence-absence data (presence = 1; absence = 0; no data = NA). (c) A data frame of sample (cell) coordinates and presence-absence data (presence = 1; absence = 0). Column names must be x, y, and `presence`. (d) A data frame of grain sizes (cell area) and occupancies in that order. If the user wishes to carry out downscaling with the Hui model

or for upgraining of atlas data (and exploration of upgraining thresholds) then the input data must be of type a, b or c. The package provides three sets of functions for each of the main stages of a downscaling analysis:

1. *Upgraining:* The process of calculating occupancy at multiple grain sizes coarser than the atlas data to generate enough data points to fit the downscaling functions (`upgrain` and `upgrain.threshold`).

2. *Modeling:* Fitting one or more downscaling functions to the coarse-grain data (`downscale` and `hui.downscale`).

3. *Prediction:* Extrapolation of the fitted downscaling functions to predict occupancy at grain sizes finer than the atlas data (`predict` and `plot`).

Note, that the `ensemble.downscale` function will run `downscale` and `predict` for a number of selected downscaling models and calculate the mean predicted occupancies across all models.

3. A worked example of the downscaling procedure

Here we work through the downscaling of a hypothetical UK species for which we have atlas data at a grain size of 10×10 km. We will upgrain in order to calculate occupancy at a further three grain sizes, and use these four data points to model the OAR. Finally we will extrapolate the fitted curve to estimate occupancy at a 1×1 km grain size.

3.1. Upgraining

In order to fit the downscaling models of all models other than the Hui model we require occupancy for at least three grain sizes. We must therefore aggregate our atlas data to coarser grain sizes, a process we refer to here as *upgraining* in order to distinguish it from upscaling, which may involve either coarsening grain size (as we do here) or increasing extent. However, if the boundaries of the atlas data are irregular, as we aggregate cells to increase grain size the extent also increases (Figure 3). As the downscaling models are modelling the change in proportion of occupancy (the total extent divided by the area of occupancy) this is undesirable. Instead we must ensure the extent is kept constant across scales by fixing the extent at all grain sizes to the extent of the largest grain size (Figure 4). For example, we could extend the atlas data by assigning unsampled cells that fall within the extent of the largest grain as absences. In this case the non-surveyed areas are largely sea and so are probably indeed absences, but in land-locked regions these areas could be suitable habitat. Instead we may choose to only keep those cells at the largest grain size that fall completely within the surveyed atlas data (Figure 5). Therefore no assumptions are made that unsampled cells are absences, but it also results in a large proportion of the original atlas data, even known presences, being excluded. This may be particularly pronounced for species that occupy the edges of the extent, such as coastal species, as very few of these edge cells will be retained using such a procedure. There is therefore a clear trade-off between assigning large areas of unsampled areas as absences, and discarding sampled areas and known presences. Instead it may be sensible to apply some threshold where only those cells at the largest grain size whose proportion of sampled area within them is larger than the threshold are kept while those with a proportion of sampled area smaller than the threshold are discarded.
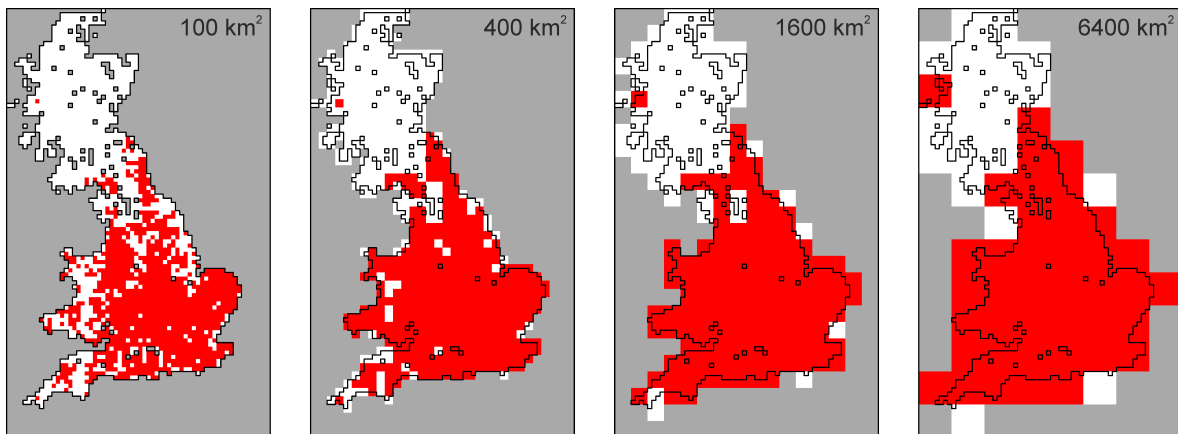
Figure 3: Upgrained presence (red cells) and absence (white cells) maps for a UK species without standardizing extent to the largest grain size. Unsampled cells are dark gray. As we upgrain the atlas data to larger grain sizes the total extent also increases.
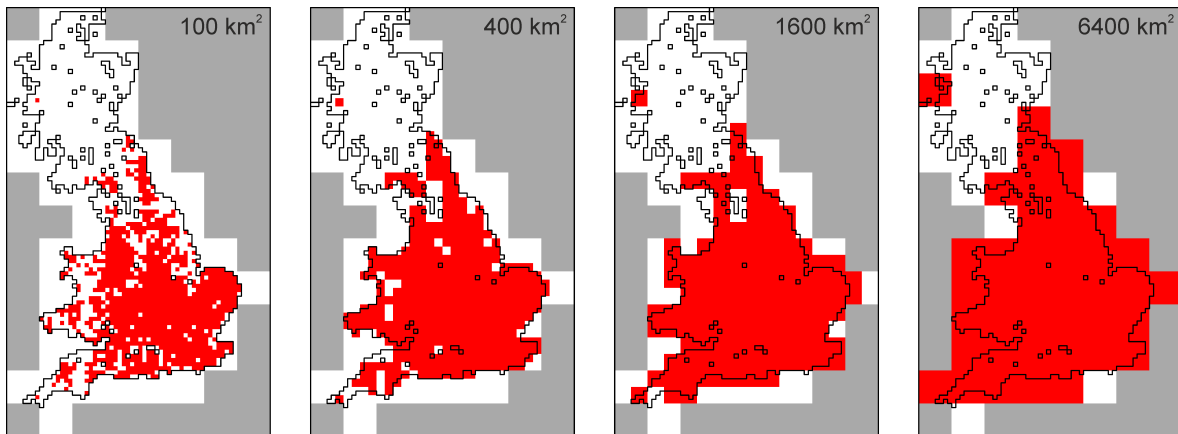

Figure 4: Upgrained presence (red cells) and absence (white cells) maps for a UK species after standardizing extent to the largest grain size. Unsampled cells are dark gray. The extent of the atlas data is extended to that of the largest grain size by assigning absences to unsampled cells.
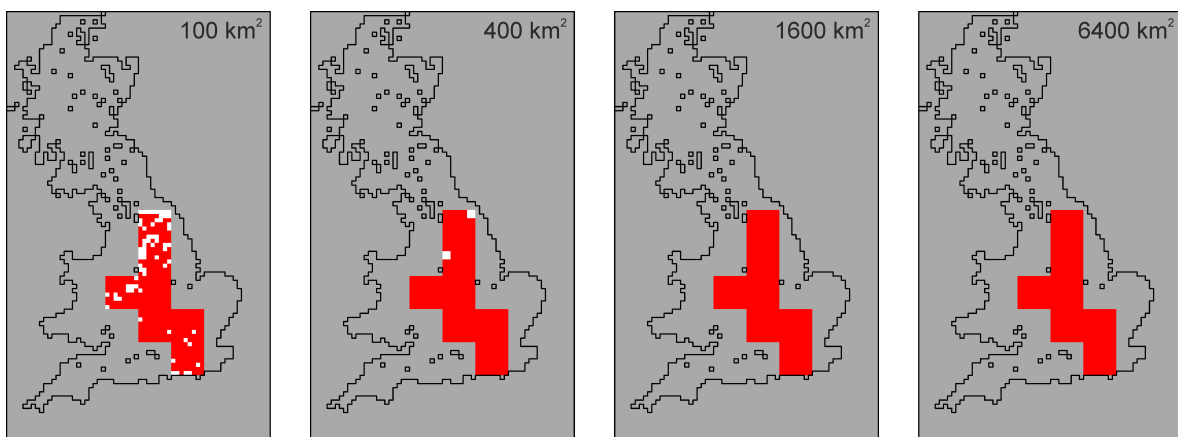

Figure 5: Upgrained presence (red cells) and absence (white cells) maps for a UK species after standardizing extent to those cells at the largest grain size that solely contain sampled atlas data. Sampled cells outside the selected cells are assigned as no data (dark gray).

The `upgrain.threshold` function allows visualizations of this trade-off at the atlas scale (the smallest grain-size when fitting the models, here the 10×10 km cells) through four plots against threshold (Figure 6): (a) The total standardized extent. (b) The number of unsampled cells added and assigned as absences, and the number of sampled cells excluded and assigned as "no data". (c) The proportion of the original atlas data retained. (d) The proportion of known presences excluded.

First we must load our hypothetical species that is included with the package. As the data is a data frame it must have the column names `x`, `y` and `presence`, but the data may also be a `raster` layer or `SpatialPointsDataFrame` object.

```
R> data.file <- system.file("extdata", "atlas_data.txt",
+   package = "downscale")
R> atlas.data <- read.table(data.file, header = TRUE)
R> head(atlas.data)
```

```
      x   y  presence
1 8170  10         0
2 8130  20         0
3 8140  20         0
4 8160  20         0
5 8170  20         0
6 8140  30         0
```

Now we can explore our upgraining thresholds through the function `upgrain.threshold`. As the input data is a data frame and not a raster file, we must specify a value for the cell widths.

```
R> thresh <- upgrain.threshold(atlas.data = atlas.data, cell.width = 10,
+   scales = 3, thresholds = seq(0, 1, 0.01))
```

The output contains a data frame containing the data required for the first set of plots (Figure 6), and the thresholds identified after applying four potential threshold criteria that produce the maps in the second set of plots (Figure 7).

```
R> thresh$Thresholds
```

```
$Thresholds
  All_Sampled  All_Occurrences  Gain_Equals_Loss  Sampled_Only
1           0             0.04              0.51             1
```

The second set of plots (hit `return` or click on the plot window; Figure 7) are the standardized atlas data generated after applying four different threshold criteria built in to the package (Table 2): (a) All sampled cells are kept ("All Sampled", the default option). (b) All known occurrences are retained ("All Occurrences"). (c) Where the extent is the same as the original atlas data ("Gain Equals Loss"). (d) If we only keep fully sampled cells ("Sampled Only"). Once a threshold has been chosen, the `upgrain` function will calculate occupancy for each grain size. The user may use one of the optional threshold criteria (e.g., `method = "All_Sampled"`) or input a specified threshold (e.g., `threshold = 0.15`). By default the
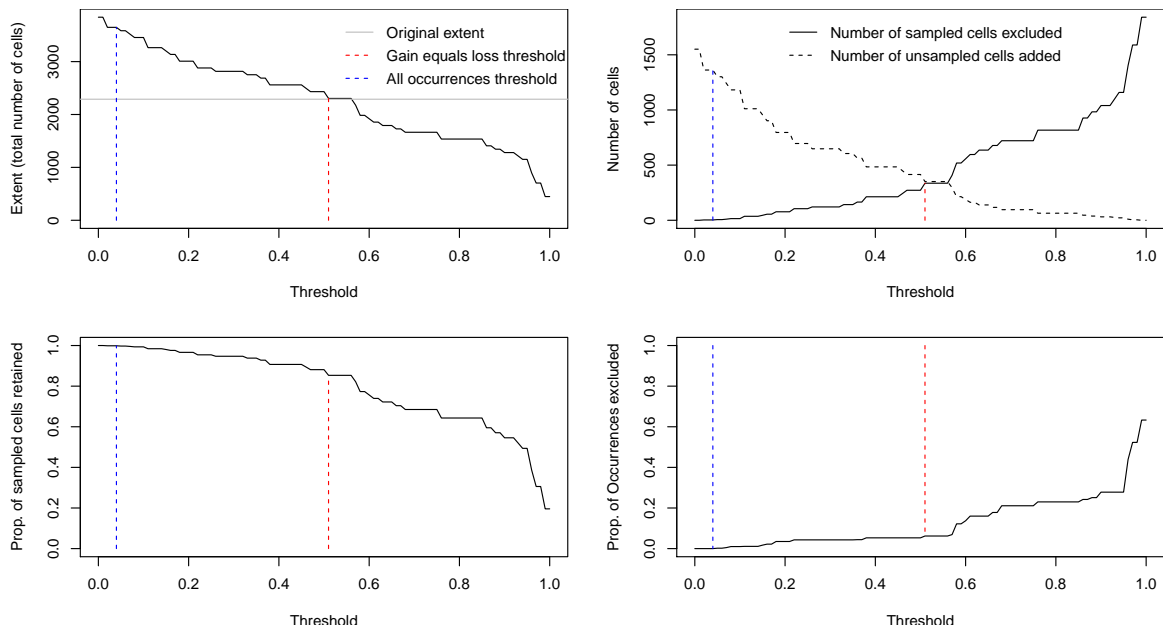
Figure 6: Diagnostic plots produced by `upgrain.threshold` used to explore the trade-off between assigning large areas of unsampled areas as absence, and discarding sampled areas and known presences. Two possible thresholds in the quantity of unsampled area allowed within cells at the largest grain size are identified: the "All Occurrences" threshold (blue line) and the "Gain Equals Loss" threshold (red line).
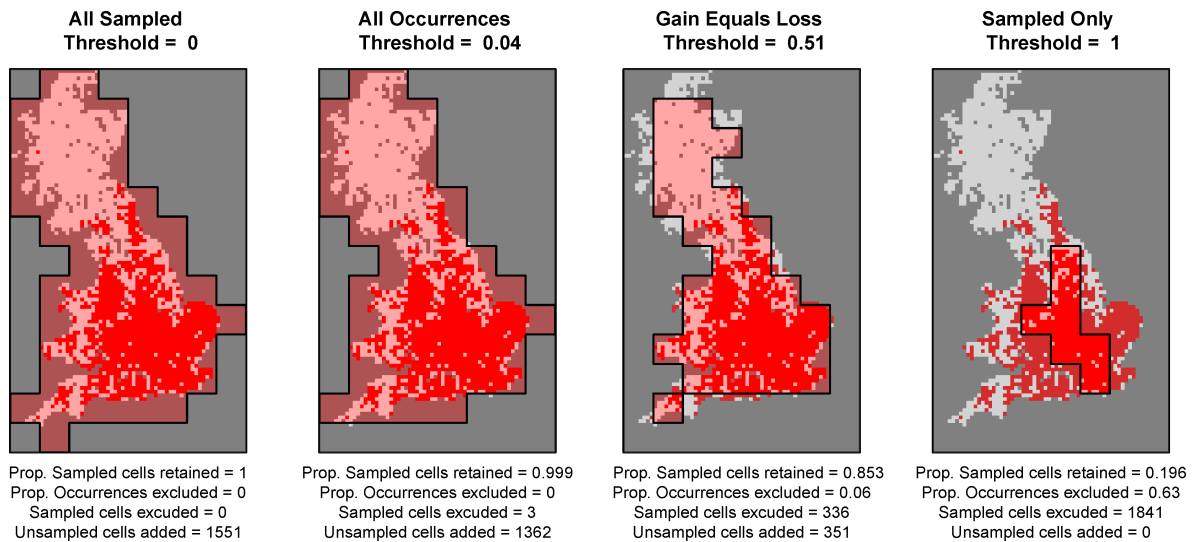

Figure 7: Maps of the atlas data (red = presence; light gray = absence; unsampled = dark gray) generated from `upgrain.threshold` overlain with polygons showing the standardized extent after applying each of four possible thresholds.

Threshold	Name	Description
0	All Sampled	All of the original atlas data is included.
Blue line (species-specific)	All Occurrences	The threshold where no occurrences in the atlas data are excluded.
Red line (atlas-specific)	Gain Equals Loss	The threshold where the number of sampled atlas cells reclassified as unsampled equals the number of unsampled cells reclassified as absences. The new standardized extent matches as close as possible the extent of the original atlas.
1	Sampled Only	Only cells that contain 100% sampled atlas data are included.

Table 2: The four suggested methods for selecting a threshold when using the `upgrain` function. Maps created after applying each threshold are presented in Figure 7.

presence-absence map for each grain size is plotted although plotting can be suppressed (`plot = FALSE`) The output is an object of class `upgrain` that contains the threshold used, the extent after standardization, a data frame of occupancy for each grain size after standardization, a data frame of occupancy for each grain size without standardization and a raster layer of the atlas data after standardization. The rasters for all other grain sizes can also be outputted (`return.rasters = TRUE`). Note, that the `upgrain` function can only return square cells and that the upgrained scales are created by doubling of cell widths (i.e., by aggregating four neighboring cells arranged in a 2×2 matrix). Here we will stick with the "All Sampled" threshold.

```
R> occupancy <- upgrain(atlas.data = atlas.data, cell.width = 10,
+     scales = 3, method = "All_Sampled", plot = TRUE,
+     return.rasters = FALSE)
R> occupancy$occupancy.orig

$occupancy.orig
  Cell.area Extent Occupancy
1       100 228900 0.4552206
2       400 266400 0.5645646
3      1600 312000 0.6102564
4      6400 384000 0.6666667


R> occupancy$occupancy.stand

$occupancy.stand
  Cell.area Extent Occupancy
1       100 384000 0.2713542
2       400 384000 0.3916667
3      1600 384000 0.4958333
4      6400 384000 0.6666667
```
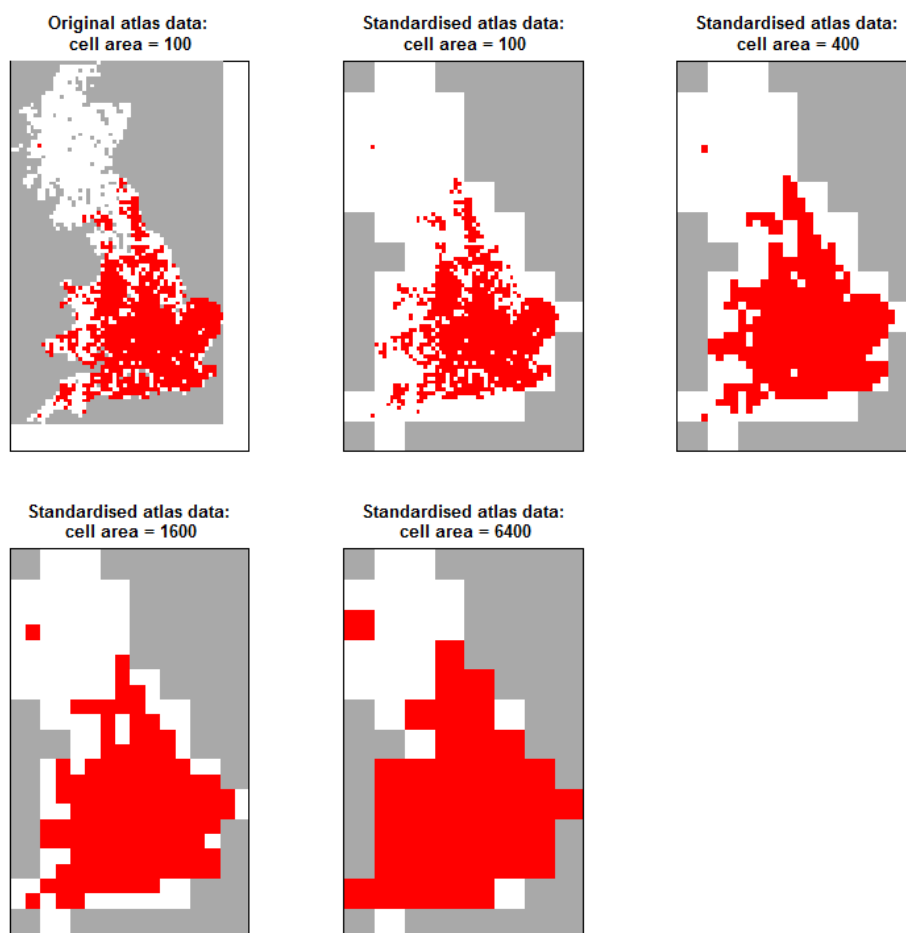
Figure 8: Maps of the atlas data (red = presence; light gray = absence; unsampled = dark gray) generated from `upgrain` using the "All Sampled" threshold.

3.2. Modeling and prediction

Fitting of the downscaling models can be carried out on the output of `upgrain` or a data frame of grain sizes and occupancies. For all models except the Hui model fitting of the models is carried out with the `downscale` function. The output object is of class `downscale` and contains the optimized parameter values, the input occupancy values and the extent (required for converting proportion occupancy to area of occupancy).

```
R> logis <- downscale(occupancies = occupancy, model = "Logis")
```

The `downscale` object is then used as input in order to predict occupancy at finer grain sizes using the `predict` function. Plotting of log occupancy against log grain size can either be called directly from `predict` (`plot = TRUE`) or from the dedicated function `plot` that also allows for all aspects of the plot to be altered. The default plot settings are the observed occupancies in black and the predicted occupancies in red. By predicting occupancy for the observed grain sizes as well as the fine grain sizes a visualization of the fit of the models can be made.
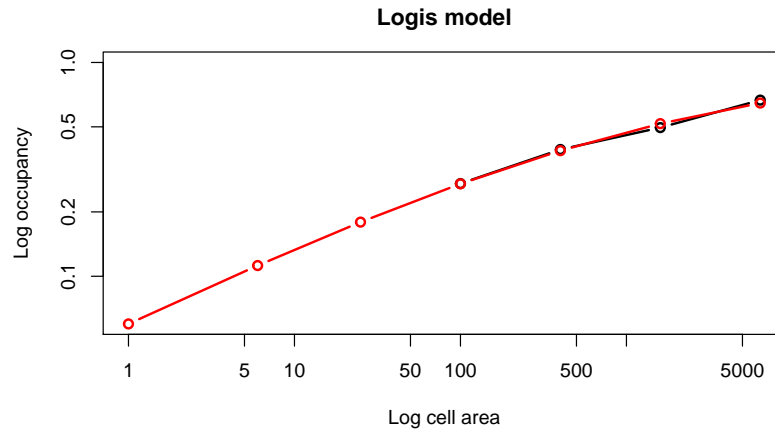
Figure 9: Observed (black) and predicted occupancies (red) against grain size for the logistic model.

```
R> pred <- predict(logis, new.areas = c(1, 6, 25, 100, 400, 1600, 6400))


R> pred$predicted

  Cell.area  Occupancy        AOO
1          1 0.05985936   22985.99
2          6 0.11223014   43096.37
3         25 0.17918593   68807.40
4        100 0.27067303  103938.44
5        400 0.38685570  148552.59
6       1600 0.51752119  198728.14
7       6400 0.64583371  248000.14
```

The Hui model does not fit a model to observed occupancies at several grain sizes, but rather calculates the probability that a cell is occupied if adjacent cells are presences or absences at a single grain size. The input may be an atlas raster layer, or if a data frame of coordinates is used then the cell width and extent must also be inputted. We can therefore use the original non-standardized atlas data but if direct comparisons between different predictions from the Hui model and other models are to be made then it is important to use the standardized atlas data for the Hui model or else only make comparisons of the area of occupancy and not proportion occupancy. The easiest method is to use the `upgrain` object as the input for `hui.downscale` (the standardized atlas data is stored in the `upgrain` object as `$atlas.raster.stand`). As the Hui model carries out modeling and prediction in a single step a vector of grain sizes for which to predict occupancy are also required, although they must be smaller than the grain size of the atlas data. There is also a parameter for the tolerance value during root solving to estimate probability of absence at the fine scale.

```
R> hui.orig <- hui.downscale(atlas.data = atlas.data, cell.width = 10,
+    new.areas = c(1, 6, 25), extent= 228900, plot = FALSE)
```

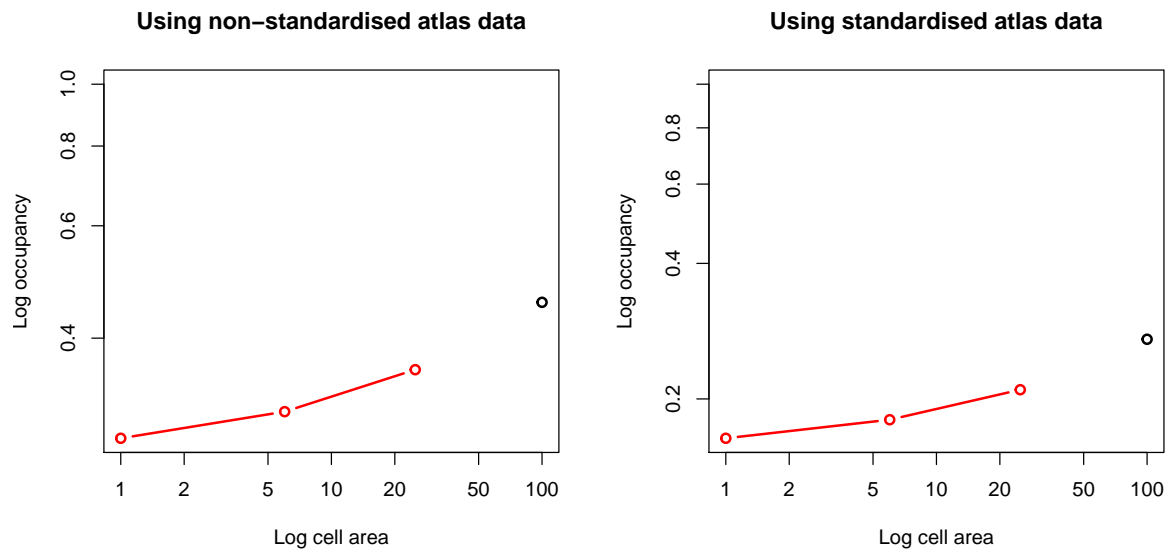**Using non–standardised atlas data**   **Using standardised atlas data**

Figure 10: Observed (black) and predicted occupancies (red) against grain size for the Hui model when using the non-standardized atlas data and when using the extent-standardized atlas data generated after applying the "All Sampled" threshold.

Alternatively we may use the extent-standardized atlas data stored within an object of class `upgrain`:

```
R> hui.stand <- hui.downscale(atlas.data = occupancy,
+    new.areas = c(1, 6, 25), plot = FALSE)
```

As we have added or removed presences and absences when standardizing the atlas data the predicted AOO values will differ when using the two atlas types. Also, as the extent of the atlas is likely to be altered during standardization (unless using the "Gain Equals Loss" threshold) it is important to only compare AOO values and not proportion occupancy.

```
R> hui.orig$predicted
```

```
  Cell.area Occupancy      AOO
1         1 0.2787162 63798.13
2         6 0.3068051 70227.68
3        25 0.3569965 81716.50
```

```
R> hui.stand$predicted
```

```
  Cell.area Occupancy      AOO
1         1 0.1635464 62801.81
2         6 0.1798366 69057.24
3        25 0.2096501 80505.63
```

```
R> par(mfrow = c(1, 2))
R> plot(hui.orig, main = "Using non-standardised atlas data")
R> plot(hui.stand, main = "Using standardised atlas data")
```
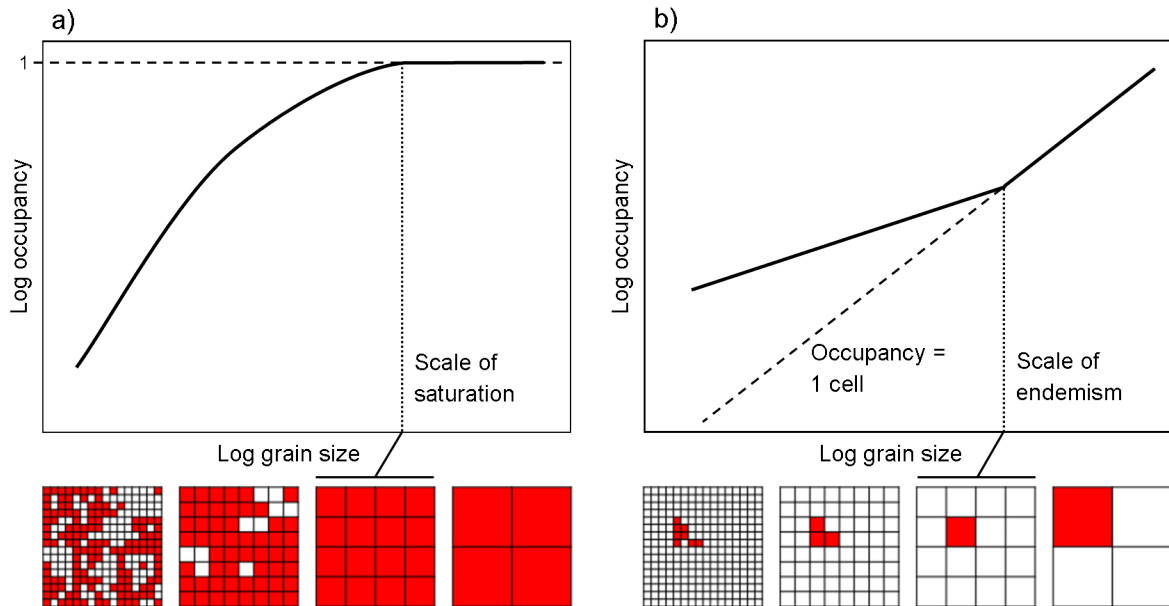
Figure 11: Occupancy-area relationships for two species showing (a) the scale of saturation (the grain size at which all cells are occupied, i.e., occupancy = 1) and (b) the scale of endemism (the scale at which only one cell is occupied). Occupancies of all grain sizes above these points should not be used when downscaling and will be set to NA by the relevant functions.

3.3. Further considerations during downscaling

For downscale modeling we must also check the data for scales of saturation and endemism. The scale of saturation is the finest grain size at which all cells become occupied (Figure 11a). The scale of endemism is the finest grain size where the entire distribution occurs in a single cell (Figure 11b). As occupancies at all grain sizes larger than these scales provide no additional information for modeling the occupancy-area curve (see details in Azaele *et al.* (2012)) they are removed prior to modeling, a procedure automatically carried out when using the `downscale`, `hui.downscale` and `ensemble.downscale` functions. This can lead, however, to insufficient scales remaining for modeling and an error message. Sometimes this error can be solved by lowering the threshold in the upgraining step. For the nine models that involve parameter optimization (i.e., excluding the Hui model) default starting values for the parameters are provided (Table 3) but users may specify their own parameters for all models if the default values are not optimizing suitably. In all cases, visual inspection of prediction plots are the most reliable methods of determining model fit. For the Thomas model the tolerance of the integration process may also be defined. The smaller the number the finer the increments of values during integration and therefore the greater the accuracy in the fitted parameters. However there is a trade-off in processing time, and very fine values may result in an hour or more for optimization. One solution is to fit the model using a large tolerance value and then re-run the modeling step with a finer tolerance value using the parameter values estimated from the previous step as the starting parameters. In the following example this approach reduces the processing time by around 90%.

Model	Parameter 1	Parameter 2	Parameter 3
Nachman	C = 0.01	z = 0.01	
PL	C = 0.01	z = 0.01	
Logis	C = 0.01	z = 0.01	
Poisson	gamma = 1e-8		
NB	gamma = 0.01	k = 0.01	
GNB	C = 0.00001	z = 1	k = 0.01
INB	C = 1	gamma = 0.01	b = 0.1
FNB	N = 10	k = 10	
Thomas	rho = 1e-8	mu = 10	sigma = 1

Table 3: Default starting parameters used in the optimization procedure for nine of the implemented downscaling models.

```
R> thomas <- downscale(occupancies = occupancy, model = "Thomas",
+    tolerance = 1e-03)
R> new.params <- list("rho" = thomas$pars["rho"], "mu" = thomas$pars["mu"],
+    "sigma" = thomas$pars["sigma"])
R> thomas <- downscale(occupancies = occupancy, model = "Thomas",
+    starting_params = new.params, tolerance = 1e-06)
```

3.4. Ensemble modeling

It remains unclear as to which model is most appropriate in which scenarios (Azaele *et al.* 2012; Barwell *et al.* 2014) and so it is probable that the user won't know which model will provide the most accurate predictions. We have therefore provided an ensemble function (`ensemble.downscale`) that will model and predict occupancy for multiple models simultaneously, greatly increasing the ease of running multiple models. It also applies a simple model averaged prediction, calculated as the means of the log occupancies.

Using `ensemble.downscale` some or all of the models can be selected. The input data is the same as for `downscale` and `hui.downscale` including tolerance values for modeling (`tolerance_pred`) and prediction in the Thomas model (`tolerance_pred`) and tolerance in the Hui model (`tolerance_hui`). Starting parameters for each model can be specified where the parameters for each model is an object in a list, for example the starting parameters for the Thomas model we previously estimated. The running status of the analysis is printed in the console but can be suppressed (`verbose = FALSE`). The predicted occupancies and area of occupancies for each model are stored in two data frames. If desired (`plot = TRUE`) the occupancies of each selected model is plotted individually (red) along with the observed data (black) and the mean ensemble prediction (gray, Figure 12).

```
R> ens <- ensemble.downscale(occupancies = occupancy,
+    new.areas = c(1, 6, 25, 100, 400, 1600, 6400), models = "all",
+    starting_params = list("Thomas" = new.params), verbose = FALSE,
+    plot = TRUE)


R> ens
```
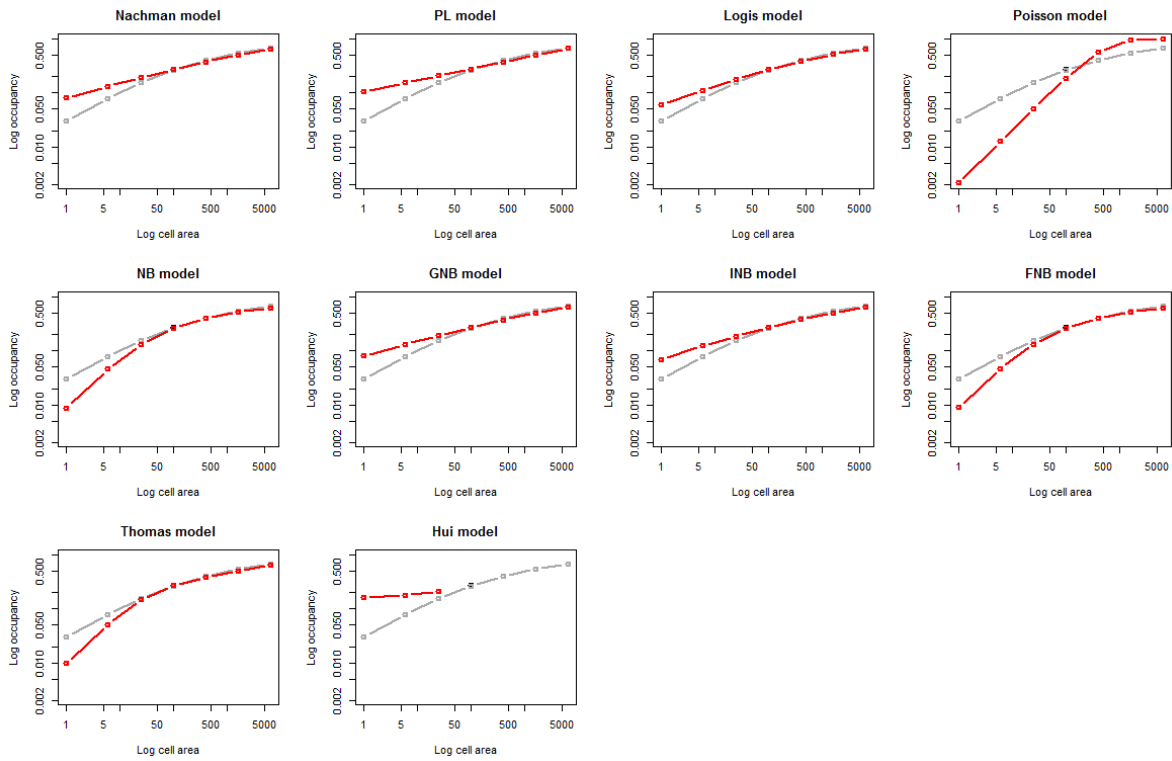
Figure 12: Plotting output when using `ensemble.downscale`. Predicted occupancies of each model is plotted individually (red) along with the observed data (black) and the mean ensemble prediction (gray).

```
$Occupancy
  Cell.area    Nachman          PL       Logis       Poisson          NB
1         1 0.08030822 0.1052773 0.05985936 0.002108798 0.008779066
2         6 0.13154860 0.1537930 0.11223014 0.012586272 0.045898017
3        25 0.19240413 0.2079890 0.17918593 0.051407171 0.132912956
4       100 0.27380698 0.2788650 0.27067303 0.190308932 0.265712040
5       400 0.38060345 0.3738932 0.38685570 0.570189130 0.404075936
6      1600 0.51186710 0.5013040 0.51752119 0.965872099 0.523487019
7      6400 0.65827020 0.6721323 0.64583371 0.999998643 0.620527568


         GNB        INB         FNB      Thomas        Hui       Means
1 0.08027777 0.07014747 0.008908909 0.009774488 0.1635464 0.03073178
2 0.13152076 0.12414319 0.046437791 0.051808327 0.1798366 0.07882954
3 0.19238489 0.18825999 0.133737854 0.148538890 0.2096501 0.15376831
4 0.27380164 0.27303401 0.265954200 0.272283180         NA 0.26119170
5 0.38061236 0.38214627 0.403406870 0.386136746         NA 0.40410502
6 0.51187754 0.51314908 0.522672643 0.504135325         NA 0.55055390
7 0.65825188 0.65599954 0.621741960 0.662667210         NA 0.68110759


$AOO
```

```
  Cell.area    Nachman         PL      Logis     Poisson          NB
1          1   30838.36   40426.47   22985.99    809.7786    3371.161
2          6   50514.66   59056.53   43096.37   4833.1284   17624.839
3         25   73883.19   79867.79   68807.40  19740.3537   51038.575
4        100  105141.88  107084.16  103938.44  73078.6298  102033.423
5        400  146151.73  143575.00  148552.59 218952.6260  155165.159
6       1600  196556.97  192500.74  198728.14 370894.8860  201019.015
7       6400  252775.76  258098.80  248000.14 383999.4791  238282.586
         GNB        INB        FNB      Thomas         Hui       Means
1   30826.66   26936.63   3421.021    3753.403    62801.81    11801.00
2   50503.97   47670.98  17832.112   19894.397    69057.24    30270.54
3   73875.80   72291.84  51355.336   57038.934    80505.63    59047.03
4  105139.83  104845.06 102126.413  104556.741          NA   100297.61
5  146155.15  146744.17 154908.238  148276.510          NA   155176.33
6  196560.98  197049.25 200706.295  193587.965          NA   211412.70
7  252768.72  251903.82 238748.913  254464.209          NA   261545.31
```

4. Future directions

The development version of the package is held at the open source website r-forge (`https://r-forge.r-project.org/projects/downscale/`) and the package will continue to be updated, refined and new functionality added. New downscaling models will be added as and when they are published. Using ongoing research by the authors evaluating the performance of downscaling models across thousands of species as well as simulated species, we also hope to implement a multi-model weighted averaging measure, where model predictions are weighted by the likelihood of their accuracy based upon the species' characteristics such as prevalence, spatial autocorrelation and taxon. Furthermore we hope to add further functionality for additional applications, such as a framework for the evaluation of Red List status from species point records.

Finally, occupancy downscaling may inform how many occupancies there are at a certain grain size, but not where those occupancies may be. By contrast, species distribution models (SDMs, Elith and Leathwick (2009)) correlate known species occurrences with environmental variables in order to predict where species occupancies may occur, however, as they produce probability of occurrence maps they can not predict how many of those occupancies there should be. Therefore a further application for occupancy-downscaling is as a method of thresholding the outputs of SDMs. We also hope to develop hybrid SDMs that incorporate the information provided by the OARs obtained from downscaling coarse grain size atlas data in to the SDM process. Such functionality will be added to the package as they are developed.

Acknowledgments

References

Azaele S, Cornell SJ, Kunin WE (2012). "Downscaling Species Occupancy From Coarse Spatial Scales." *Ecological Applications*, **22**(3), 1004–14. `doi:10.1890/11-0536.1`.

Barwell LJ, Azaele S, Kunin WE, Isaac NJB (2014). "Can Coarse-Grain Patterns in Insect Atlas Data Predict Local Occupancy?" *Diversity and Distributions*, **20**(8), 895–907. `doi:10.1111/ddi.12203`.

Bivand RS, Pebesma E, Gomez-Rubio V (2013). *Applied spatial data analysis with* R. 2nd edition. Springer. URL `http://www.asdar-book.org/`.

Elith J, Leathwick JR (2009). "Species Distribution Models: Ecological Explanation and Prediction across Space and Time." *Annual Review of Ecology, Evolution, and Systematics*, **40**, 677–697. `doi:10.1146/annurev.ecolsys.110308.120159`.

Elzhov TV, Mullen KM, Spiess AN, Bolker B (2016). **minpack.lm**: R *Interface to the Levenberg-Marquardt Nonlinear Least-Squares Algorithm Found in* **MINPACK**, *Plus Support for Bounds*. R package version 1.2-1, URL `https://CRAN.R-project.org/package=minpack.lm`.

Gaston KJ (1994). "Measuring Geographic Range Sizes." *Ecography*, **17**(2), 198–205. `doi:10.1111/j.1600-0587.1994.tb00094.x`.

Hanski I, Gyllenberg M (1997). "Uniting Two General Patterns in the Distribution of Species." *Science*, **275**(5298), 397–400. `doi:10.1126/science.275.5298.397`.

Hartley S, Kunin WE (2003). "Scale Dependence of Rarity, Extincition Risk, and Conservation Priority." *Conservation Biology*, **17**(6), 1–12. `doi:10.1111/j.1523-1739.2003.00015.x`.

He F, Gaston KJ (2000). "Estimating Species Abundance from Occurrence." *The American Naturalist*, **156**(5), 553–559. `doi:10.1086/303403`.

He F, Gaston KJ (2003). "Occupancy, Spatial Variance, and the Abundance of Species." *The American naturalist*, **162**(3), 366–75. `doi:10.1086/377190`.

He F, Gaston KJ, Wu J (2002). "On Species Occupancy-Abundance Models." *Ecoscience*, **9**(1), 119–126. `doi:10.1080/11956860.2002.11682698`.

Hijmans RJ (2017). **raster**: *Geographic Data Analysis and Modeling*. R package version 2.6-7, URL `https://CRAN.R-project.org/package=raster`.

Hui C (2009). "On the Scaling Patterns of Species Spatial Distribution and Association." *Journal of Theoretical Biology*, **261**(3), 481–487. `doi:10.1016/j.jtbi.2009.08.015`.

Hui C, McGeoch MA, Reyers B, le Roux PC, Greve M, Chown SL (2009). "Extrapolating Population Size from the Occupancy-Abundance Relationship and the Scaling Pattern of Occupancy." *Ecological Applications*, **19**(8), 2038–2048. `doi:10.1890/08-2236.1`.

Hui C, McGeoch MA, Warren M (2006). "A Spatially Explicit Approach to Estimating Species Occupancy and Spatial Correlation." *Journal of Animal Ecology*, **75**(1), 140–147. `doi:10.1111/j.1365-2656.2005.01029.x`.

IUCN (2012). *IUCN Red List Categories and Criteria: Version 3.1.* 2nd edition. IUCN, Gland. URL `https://www.iucn.org/content/iucn-red-list-categories-and-criteria-version-31-second-edition`.

IUCN Standards and Petitions Subcommittee (2017). "Guidelines for Using the IUCN Red List Categories and Criteria, Version 13." URL `http://www.iucnredlist.org/documents/RedListGuidelines.pdf`.

Johnson SG, Narasimhan B (2017). **cubature**: *Adaptive Multivariate Integration over Hypercubes*. R package version 1.3-11, URL `https://CRAN.R-project.org/package=cubature`.

Kunin WE (1998). "Extrapolating Species Abundance across Spatial Scales." *Science*, **281**(5382), 1513–1515. `doi:10.1126/science.281.5382.1513`.

Mächler M (2018). **Rmpfr**: *R MPFR – Multiple Precision Floating-Point Reliable*. R package version 0.7-0, URL `https://CRAN.R-project.org/package=Rmpfr`.

Nachman G (1981). "A Mathematical Model of the Functional Relationship between Density and Spatial Distribution of a Population." *Journal of Animal Ecology*, **50**(2), 453–460. `doi:10.2307/4066`.

Pebesma EJ, Bivand RS (2005). "Classes and Methods for Spatial Data in R." *R News*, **5**(2). `doi:https://CRAN.R-project.org/doc/Rnews/`.

Powney GD, Isaac NJB (2015). "Beyond Maps: A Review of the Applications of Biological Records." *Biological Journal of the Linnean Society*, **115**, 532–542. `doi:10.1111/bij.12517`.

R Core Team (2018). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL `https://www.R-project.org/`.

Wright DH (1991). "Correlations between Incidence and Abundance Are Expected by Chance." *Journal of Biogeography*, **18**(4), 463–466. `doi:10.2307/2845487`.

Zillio T, He F (2010). "Modeling Spatial Aggregation of Finite Populations." *Ecology*, **91**(12), 3698–3706. `doi:10.1890/09-2233.1`.

Affiliation:

Charles J. Marsh
School of Biology
University of Leeds
Leeds SL2 9JT, United Kingdom
E-mail: charliem2003@gmail.com