MetaWeave: a reproducible framework for the spatial inference of ecological networks

Gabriel Munoz-Acevedo

Concordia University, Biology Department, Montreal, QC, Canada.

gabrielmunoz1891@gmail.com

Abstract

A regional species-interaction network, or metaweb, describes interactions that potentially occur among species within a region. MetaWeave is an R package that combines regional metawebs, local species assemblages, and alternative interaction models to reconstruct local networks. Its main feature is to separate the effects of interaction assumptions from changes in community composition. We illustrate the package with three applications: a synthetic benchmark comparing interaction models against a known generating process; spatial downscaling of a Neotropical palm–mammal metaweb; and a reanalysis of six Ecuador plant–frugivore networks. In the Ecuador example, an externally fitted regional stochastic block model is applied to both observed and richness-matched alternative assemblages. Holding the observed assemblages fixed, the block model increased simulated nestedness relative to neutral link allocation at the two lowest sites, while the composition contrasts were less consistent across sites. These comparisons are conditional on species recorded in the interaction data and use a model fitted to the pooled networks. MetaWeave 1.0.0 is available on CRAN. By retaining local interaction matrices and allowing assemblages and interaction rules to vary independently, MetaWeave supports explicit comparisons of ecological network hypotheses across space.

Introduction

Understanding the organization of ecological communities requires more than mapping species richness and their geographic ranges, as these alone do not capture how species interact. Ecological networks (e.g. producer-consumer networks) capture this variation by describing how interactions among species vary across space and time (Strydom et al. 2021). Yet documenting ecological networks comprehensively remains difficult and costly at broad scales and high resolutions (Strydom et al. 2021; Banville et al. 2025). Regional metawebs offer a starting point to understand the geographic organization of ecological interactions (Saravia et al. 2022; Dansereau, Barros, and Poisot 2024). Regional metawebs typically aggregate local data on species interactions across a region into a single binary or probabilistic (meta) network (Strydom et al. 2021; Banville et al. 2025). Therefore, regional metawebs inform about the interaction potential among species in a regional species pool (Saravia et al. 2022; Marjakangas et al. 2022). When combined with spatially explicit species distribution maps, it becomes possible to infer where interactions may occur, including in communities that have not been directly sampled (Saravia et al. 2022; Dansereau, Barros, and Poisot 2024; Strydom et al. 2021).

Species co-occurrences provides opportunities for interaction, but whether species interact depends on factors such as trait compatibility, ecological roles, and environmental conditions. Consequently, communities with similar species richness may differ substantially in their interaction structure and responses to disturbance (Montoya, Pimm, and Solé 2006). Several modelling approaches describe how species distributions,

functional traits and environmental conditions can inform local network inference from a regional metaweb (Strydom et al. 2021; Saravia et al. 2022; Marjakangas et al. 2022). However, available implementations remain fragmented, and general software tools for the spatial analysis of regional metawebs are limited (Ohlmann, Garnier, and Vuillon 2023). This limits the ability to systematically translate regional interaction information into spatially explicit predictions of local network structure. A common framework would make these analyses easier to reproduce, extend and compare across systems.

MetaWeave is an R package that integrates local spatial network inference within a modular and reproducible framework. Building on capabilities available in existing open-source tools, MetaWeave combines spatially explicit species assemblages with regional metawebs to generate local interaction matrices, retaining species-pair probabilities for subsequent analysis. By separating species distribution data, interaction models, and network outputs into interchangeable components, MetaWeave enables users to explore spatial variation in potential ecological networks and compare interaction assumptions within a consistent analytical workflow.

Methods and features

From species distributions to local assemblages

The spatial workflow takes one raster layer per species, organized into trophic levels or ecological guilds, with all layers sharing the same projection, extent, resolution, and grid alignment. Species names and species pools should be harmonized across distribution

data and interaction models. MetaWeave uses the `terra` package to handle these raster data and determine which species co-occur in each geographic cell (Hijmans 2025). Continuous occurrence probabilities must first be converted to presence–absence maps. Geographic cells are kept only if they contain at least one species, or any user-defined number, from each trophic level or ecological guild. Each retained geographic cell represents a local assemblage composed of the species present in that cell. Assemblages can also be supplied directly as site-level species lists, as in the Ecuador case study. At this stage, species co-occurrence represents an opportunity for interaction (Strydom et al. 2021; Dansereau, Barros, and Poisot 2024); interactions are inferred in the following step.

A common interface for interaction inference

For each local assemblage, MetaWeave applies an inference model through a common interface:

$$\hat{G}_\ell = \mathcal{I}(C_\ell, M),$$

where $\hat{G}_\ell$ is the inferred network at locality ℓ, C_ℓ is its species assemblage, M is the supplied model and $\mathcal{I}$ is the inference procedure.

The inferred local network is represented as an interaction matrix, with entries corresponding to binary links, interaction probabilities, or weights depending on the interaction model. Local matrices are retained together with cell identifiers and coordinates, allowing network summaries to be mapped while preserving the underlying species interactions.

Alternative models of interaction formation

MetaWeave supports several ways to translate regional metawebs into local species assemblages with their respective predicted interaction networks. Available approaches to users mainly differ in how interaction probabilities are specified, while keeping stable the spatial definition of local communities (Strydom et al. 2021; Banville et al. 2025).

Species-pair probability matrices. MetaWeave can use a species-by-species matrix containing interaction probabilities between zero and one. For each local assemblage, the corresponding rows and columns are extracted to define the local interaction matrix. These probabilities may come from trait-matching models, expert inputs, or other external analyses (Banville et al. 2025). A binary regional metaweb can be used in the same way, with one indicating a recorded regional interaction and zero an unrecorded pair. MetaWeave uses these matrices as supplied and does not fit the underlying interaction model.

Stochastic block models. Species can also be assigned to ecological guilds (i.e. blocks), with interaction probabilities defined between blocks rather than individual species (Banville et al. 2025). Species belonging to the same block therefore share the same interaction probabilities with species in another block. For each local assemblage, MetaWeave expands these block-level probabilities into a species-level interaction matrix. Model fitting and block selection remain external to MetaWeave and can therefore be carried out using any suitable modelling framework (Terry 2024).

User-defined models. MetaWeave also allows users to provide their own interaction model. Predictions can depend on the species present in a cell as well as contextual information such as location, environmental conditions, or species traits (Strydom et al. 2021; Banville et al. 2025). The prediction function is supplied by the user, while MetaWeave handles its application across local assemblages and the resulting spatial outputs. This allows alternative ecological hypotheses to be evaluated within the same spatial workflow without requiring MetaWeave to fit the underlying model. While species-pair matrices provide fixed interaction values that are simply subset to each local assemblage, user-defined models calculate interaction values dynamically for each assemblage and can incorporate local environmental or trait information.

Figure 1 summarizes this separation of inputs. Local assemblages can change while the interaction model stays fixed, or alternative models can be applied to the same assemblages. The output in either case is a collection of local species-pair matrices that can be summarized or analysed with other network tools.

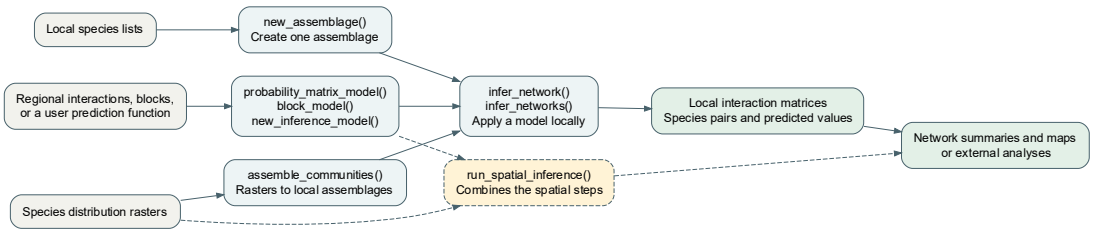


Figure 1: ***MetaWeave workflow.*** *`assemble_communities()` constructs assemblages from species distributions, while `new_assemblage()` accepts a local species list. Model constructors define alternative interaction rules, and `infer_network()` or `infer_networks()` applies a rule to local assemblages. `run_spatial_inference()` combines the spatial steps, producing local matrices and mapped summaries.*

Main functions

MetaWeave is organized around a small set of functions corresponding to the main steps of the workflow. Additional helper functions for preparing spatial inputs, constructing network objects, simulation, and metric calculation are described in the package repository and documentation.

- `assemble_communities(distributions, min_species = 1L)` builds local assemblages from a named list of aligned species-distribution rasters. `min_species` sets the minimum number of species required in each ecological group for a cell to be retained.

- `probability_matrix_model(probability_matrix, row_group, column_group, name = "species probability matrix")` defines fixed species-pair interaction probabilities, or binary regional links, between two ecological groups.

- `block_model(row_lookup, column_lookup, theta, row_group, column_group, species_col = "species", guild_col = "guild")` defines interaction probabilities among ecological blocks or guilds and expands them to the species represented in each local assemblage.

- `new_inference_model(predict, name = "custom", parameters = list(), subclass = NULL)` creates a user-defined inference model. The supplied `predict` function receives the model and a local assemblage and returns an ecological network.

- `infer_network(assemblage, model)` applies an interaction model to a single local assemblage.
- `infer_networks(assemblages, model, template = NULL)` applies the same model across a collection of local assemblages and retains their cell identifiers and coordinates.
- `reconstruct_networks(distributions, model, min_species = 1L)` combines assemblage construction and network inference across the distribution data.
- `run_spatial_inference(distributions, model, min_species = 1L, threshold = NULL)` runs the complete spatial workflow, including local assemblage construction, network inference, network summaries, and mapping of the resulting metrics.

Getting started with a minimal example

The following example uses made-up species, a two-by-two probability matrix, and a four-cell spatial grid. It needs only the released `metaweave` package and its `terra` dependency; no external data or model fitting are required. Rows of `p` are plants and columns are animals. Each entry is a supplied interaction probability conditional on both species being present, not an occurrence probability estimated by MetaWeave.

```
library(metaweave)

## Define probability matrix
p <- matrix(c(0.8, 0.2, 0.3, 0.7),
            nrow = 2, byrow = TRUE,
            dimnames = list(c("plant_a", "plant_b"),
```

```
                              c("animal_x", "animal_y")))

## Create model object
model <- probability_matrix_model(p, "plants", "animals")
## Create spatial grid
grid <- create_standard_grid(c(0, 2, 0, 2), resolution = 1)
## Map plant and animal distribution
plants <- terra::rast(list(grid, grid))
animals <- terra::rast(list(grid, grid))
names(plants) <- rownames(p)
names(animals) <- colnames(p)
terra::values(plants) <- cbind(c(1, 1, 0, 1), c(1, 0, 1, 1))
terra::values(animals) <- cbind(c(1, 1, 1, 0), c(0, 1, 1, 1))

## Run spatial inference workflow
result <- run_spatial_inference(
  list(plants = plants, animals = animals),
  model,
  min_species = 1
)

## Extract results
result$result$index[, c("cell_id", "expected_links")]
## Plot results
terra::plot(result$spatial$data[["expected_links"]],
            main = "Expected links per cell", axes = FALSE)
```

The index reports expected links of 1.1, 1.0, 1.0, and 0.9 for cells 1–4, respectively (Fig. 2). These are sums of the retained pair probabilities, not observed link counts. The local interaction matrices remain available in `result$result$networks`.

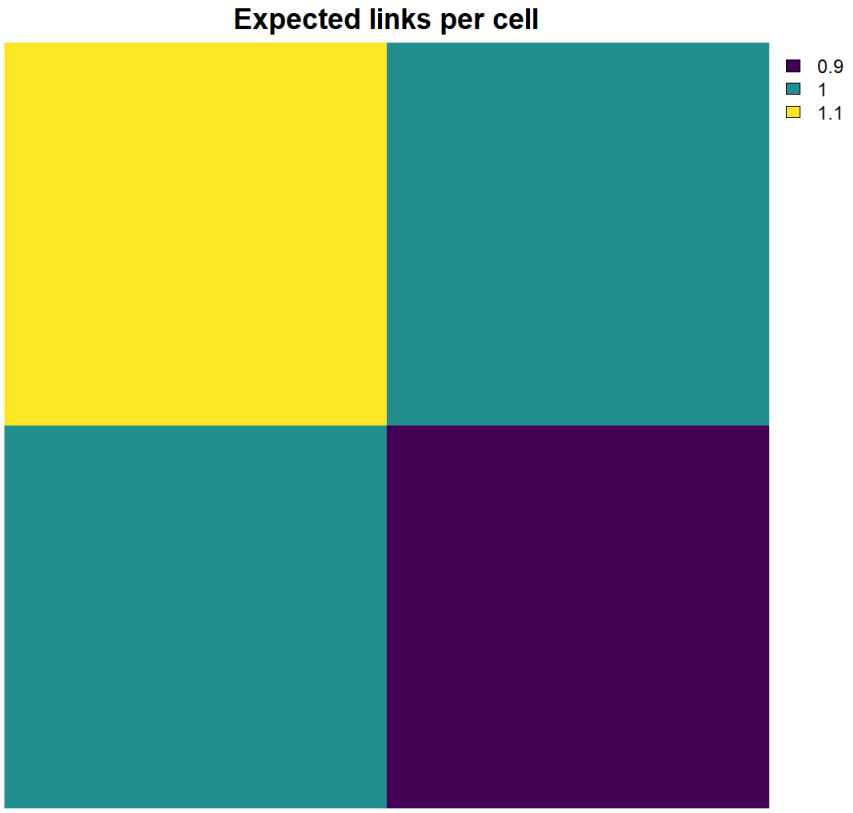


Figure 2: ***Expected interaction links across four invented cells.*** *Colours represent sums of the supplied species-pair interaction probabilities.*

Synthetic evaluation benchmark

We evaluated MetaWeave using a synthetic bipartite system with 24 producer species, 18 consumer species, and 225 spatial cells distributed along an environmental gradient. Species distributions generated spatial turnover among cells, whereas local interaction probabilities varied with block structure, species-pair heterogeneity, and environmental context. We compared three inference models: a block model, a regional species-pair

probability model, and a data-generating model supplied with the true data-generating process. All three models were applied to the same local assemblages.

The data-generating model recovered the interaction probabilities exactly, with zero probability RMSE (Fig. 3A) and expected connectance values that fell on the one-to-one line (Fig. 3B). This simply confirms that the custom-model and spatial workflow preserve the probabilities they are given. Among the two alternative models, the regional species-pair model was closer to the generating process than the block model. It had lower probability RMSE (0.112 vs. 0.147; Fig. 3A), lower Brier score and log loss (Fig. 3C), and slightly lower errors for expected links and connectance. Both models reproduced the broad pattern in expected connectance, but the species-pair model remained closer to the true values (Fig. 3B).

These results show that MetaWeave carries differences among interaction models through the same spatial workflow. In this example, the stochastic block model was not necessarily a poor model; it is simply less well matched to this particular simulated system, where interactions varied strongly among individual species pairs. This benchmark should therefore be taken as an overall workflow demonstration rather than a ranking of model classes. The simulation and scoring workflow is provided in Supplementary material 1 (`S1-synthetic-benchmark.qmd`).

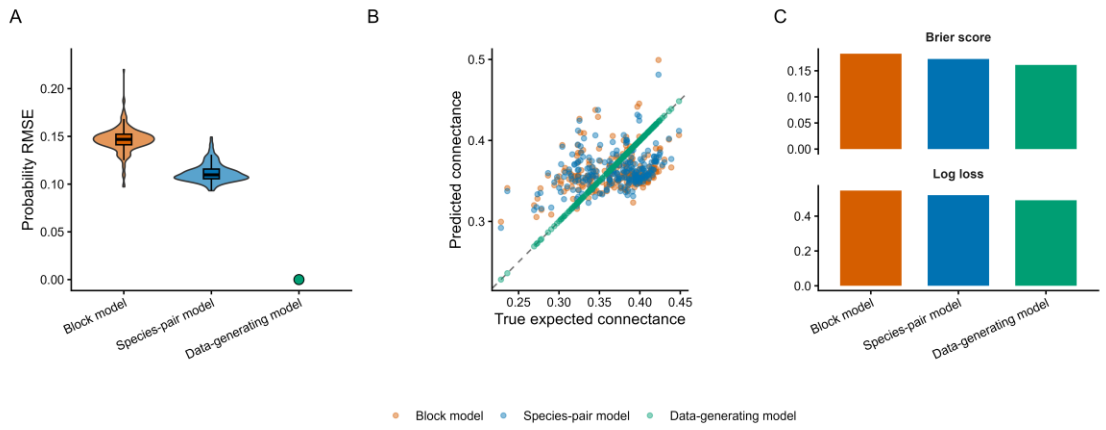


Figure 3: ***Synthetic evaluation of spatial network inference with MetaWeave.*** *(A) Site-level RMSE between predicted and generating interaction probabilities across 225 cells. (B) Predicted versus generating expected connectance; the dashed line denotes equality. (C) Brier score and logarithmic loss against sampled binary interactions. The data-generating model is the oracle and has zero probability and connectance error by construction. Its nonzero scores against binary interactions reflect stochastic realization. All models use identical assemblages.*

Case studies

Case study 1: Downscaling a palm–mammal frugivory neotropical seed dispersal metaweb

We used a Neotropical palm–mammal interaction dataset to translate regional interaction information into spatial predictions. Interaction records from Muñoz, Trøjelsgaard, and Kissling (2019) were combined with palm and mammal range maps, with species names standardized and duplicate records removed. The matched dataset included 51 palm species, 68 mammal species, and 234 recorded interactions. Local communities were defined on a 1° WGS84 grid, retaining cells with at least one species from each group. Palm

and mammal richness varied across the region (Fig. 4A,B). Supplementary material 2 details the regional inputs, MetaWeave reconstruction and map calculations.

We compared two ways of translating the same regional network to local communities: retaining recorded links when species co-occurred, or assigning probabilities with a stochastic block model fitted in `cassandRa` 0.2.0 (Terry 2024). Both approaches used the same local assemblages. Across 1,254 cells, the recorded-metaweb approach produced 21.17 possible links on average, compared with 12.11 expected links under the block model. Their broad spatial patterns were similar, but the block model generally predicted fewer interactions (Fig. 4C,D), with the largest differences concentrated along the Yungas and the Andean–Amazon transition zone (Fig. 4E).

The lower number of expected links under the block model reflects a different way of translating the same regional interaction information to local communities. The model can assign some probability to species pairs that were not recorded regionally while also reducing the expected contribution of recorded pairs from one to a value below one. As a result, the differences in Fig. 4E show where the two interaction assumptions lead to different local network predictions. This is the main value of the comparison: by keeping local species composition unchanged, MetaWeave makes it possible to identify where interaction models agree, where they differ, and where additional local sampling may be most informative. Without a separate validation step, these predictions describe potential local interaction structure rather than confirmed encounters or seed-dispersal effectiveness. The regional inputs, local reconstruction, and map workflow are provided in Supplementary material 2.

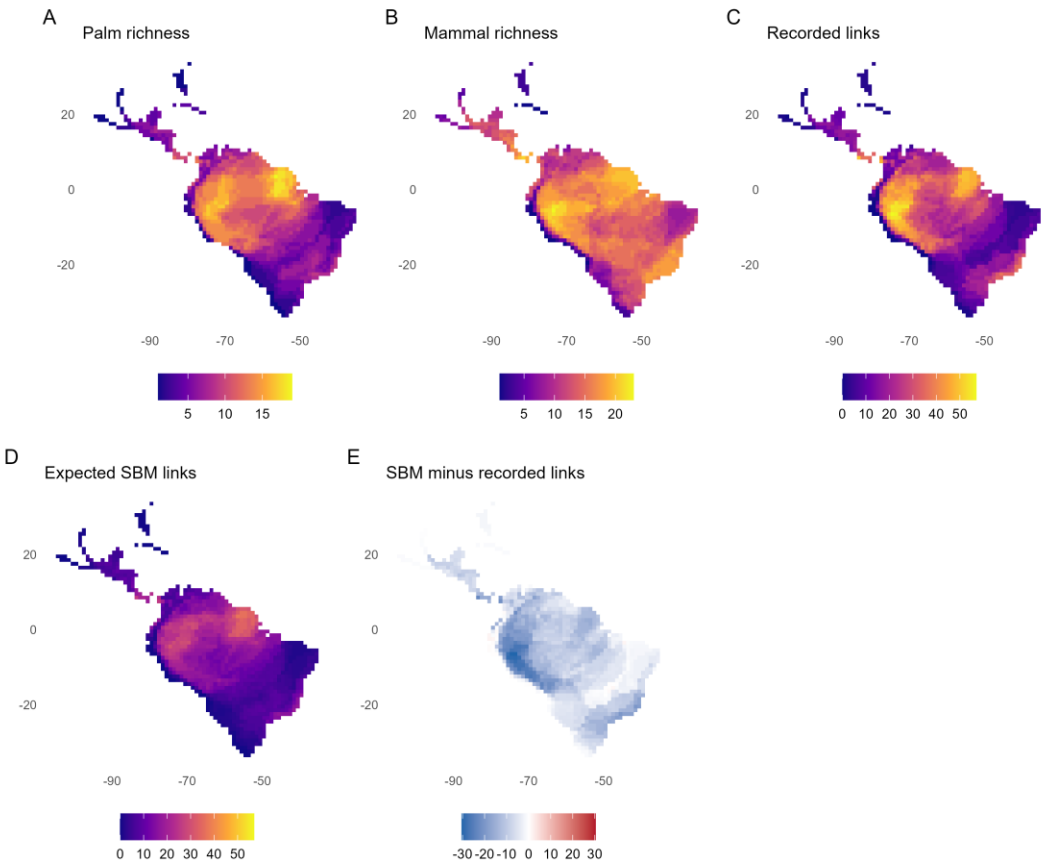


Figure 4: ***Palm–mammal network reconstructions from the same local assemblages. (A–B) Richness of retained palm and mammal species.*** *(C) Recorded regional links whose species co-occur locally. (D) Sum of block-model probabilities, expressed as expected links. (E) Expected block-model links minus recorded-baseline links. Panels C and D use the same colour limits; panel E uses a symmetric scale centred on zero. Positive and negative differences reflect redistribution of probability across recorded and unrecorded pairs. Blank cells fall outside the eligible assemblages. The maps show model-dependent interaction potential, not observations of realized interactions.*

Case study 2: Community and interaction assembly in Ecuador plant–frugivore networks

We used MetaWeave and the six plant– bird frugivore networks provided in (Marjakangas et al. 2022; Muñoz 2021) to infer the effects of community and interaction assembly process in network structure across an elevational gradient in the Andes of Ecuador. Their interaction records define 102 plant and 127 frugivore species and 536 distinct regional pairs. We pooled the recorded links across sites into one binary metaweb and fitted a bipartite stochastic block model to it with `cassandRa` (Terry 2024). The selected fit assigned plants and frugivores to ten blocks each. This model was fitted only at the regional level. At each site, MetaWeave applied the same block memberships and block-pair probabilities to the locally recorded species, yielding a probability-valued interaction matrix.

We compared three fixed-link-count ensembles at each site. **A** retained the observed plant and frugivore species and allocated links uniformly among eligible pairs. **B** retained those species and allocated links using the regional block-model probabilities as sampling weights. **C** sampled plants and frugivores with equal probability from their respective regional pools, preserving the site's richness in each group, and applied the same block model. Each realization contained the site's observed number of distinct links. The **B – A** contrast isolates the change in interaction rule at fixed composition; **B – C** changes composition under a fixed regional rule. We calculated binary NODF nestedness with `vegan` and bipartite modularity with `bipartite` for the observed networks and 49 realizations per scenario. The regional-pool randomization assumes equal selection

probability and does not represent habitat filtering or dispersal limitation. The scripts, fit and metric settings are documented in Supplementary material 3.

At the two 1,000-m sites, the regional block model produced more nested networks than uniform link allocation with the same local species and link count: mean B − A differences were 13.0 NODF units at Bombuscaro and 15.9 at Copalinga (Fig. 5A). The corresponding modularity differences were about −0.05 at both sites (Fig. 5B). At the 2,000-m sites, the nestedness contrast was smaller (4.2–4.6 units), and at the 3,000-m sites its simulation interval included zero. The composition contrasts B − C were less consistent: their 95% simulation intervals included zero for nestedness at all six sites, although the two 1,000-m sites showed lower modularity under their observed assemblages than under richness-matched random assemblages.

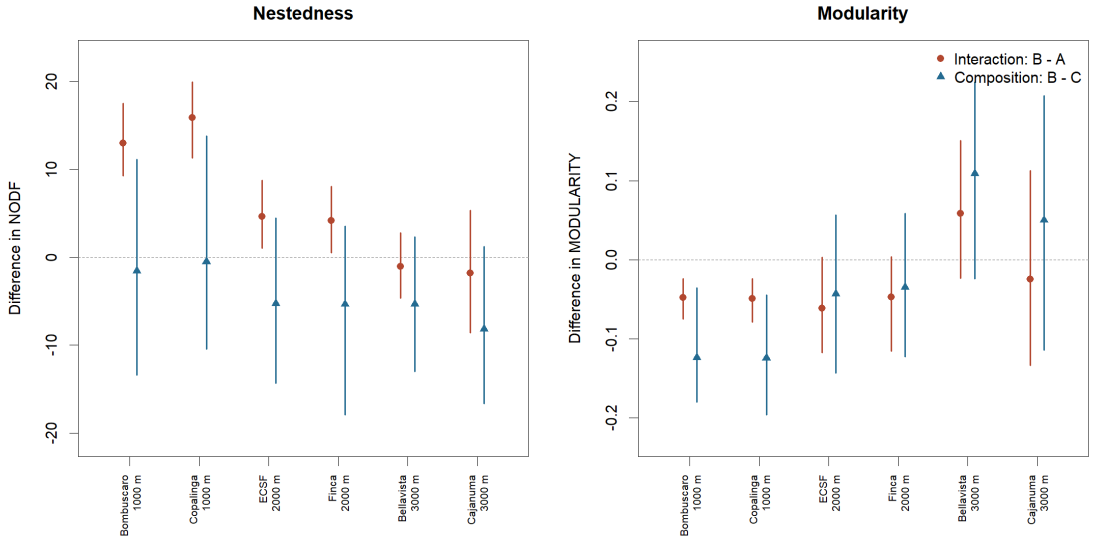


Figure 5: ***The assembly of networks between fleshy-fruited plants - bird frugivores across an elevational gradient in Ecuador.*** *(A) Difference in binary NODF nestedness and*

(B) difference in bipartite modularity across six sites ordered by elevation. Circles show the interaction-rule contrast B − A: the regional SBM-weighted ensemble minus the uniform-link ensemble for the same local species. Triangles show the composition contrast B − C: observed local species minus richness-matched random species under the same regional SBM. Points are means and vertical lines span the central 95% of 49 paired simulation differences. All ensembles preserve each site's observed plant richness, frugivore richness and number of distinct links. These are in-sample structural comparisons; the intervals describe simulation variation rather than uncertainty over Ecuador sites. These differences are in the metric's original units, not standardized scores.

Discussion

Relationship to existing software

MetaWeave overlaps with existing tools in its use of regional interaction information and local networks, while providing an explicit interface between **assemblages** and **interaction models**. The following comparison locates that contribution within the software used in this study. The distinctions concern documented workflows, not a claim that other packages cannot be extended by users.

Software	Documented role	Relationship to MetaWeave
`metanetwork`	Builds, analyses and visualizes regional trophic metawebs and local networks from a metaweb and optional community or trophic information (Ohlmann, Garnier, and Vuillon 2023).	Shares the regional-to-local network setting; MetaWeave exposes alternative inference models as interchangeable inputs to the same local assemblages.
`net.raster`	Combines species-distribution rasters and a bipartite interaction network to map network and species-level metrics	Shares raster-based spatial analysis; MetaWeave retains each local pair matrix and permits supplied pair probabilities, block

Software	Documented role	Relationship to MetaWeave
	(Oliveira et al. 2025).	probabilities or user-defined predictions.
`cassandRa`	Fits statistical models to bipartite interaction data, including stochastic block models and estimates of missing links (Terry 2024).	Supplies fitted block memberships and probabilities; MetaWeave applies those parameters to observed or alternative local assemblages without refitting.

Intended uses, limitations and future extensions

MetaWeave provides a reproducible framework for examining how assumptions about species interactions and changes in community composition influence predictions of ecological network structure (Strydom et al. 2021; Banville et al. 2025). The synthetic benchmark shows how alternative interaction models can be evaluated against a known generating process. The palm–mammal application compares two interaction assumptions on identical spatial assemblages. The Ecuador application shows the complementary use: a fitted regional interaction model can be held fixed while local species identities change. Its nestedness and modularity contrasts suggest that interaction assumptions can matter even when richness and link count are controlled, but the fitted block model did not reproduce all observed structure. Together, the applications illustrate the value of varying assemblages and models independently.

Metaweave allows the reconstruction of both binary (presence/absence) and probabilistic interaction networks. When inferring bipartite probability matrices, summing pairwise interaction probabilities gives the expected number of links without requiring interactions

to be independent. However, these marginal probabilities do not describe whether particular interactions tend to occur together, exclude one another, or otherwise depend on the structure of the realized network (Banville et al. 2025). Such dependencies require additional assumptions about network realization. As a result, nonlinear properties such as nestedness or robustness cannot generally be obtained by simply evaluating the probability matrix directly, because this is not equivalent to averaging those properties across possible binary-network realizations (Banville et al. 2025). Nonlinear network properties can instead be estimated from repeated binary realizations of each local probability matrix, with the realization model specifying any dependence among interactions (Banville, Gravel, and Poisot 2023; Banville et al. 2025).

MetaWeave outputs depend on the quality of the input data and the assumptions used (Strydom et al. 2021; Banville et al. 2025). Errors in species distributions, taxonomy, or interaction records can affect local predictions, and broad range overlap may overestimate where species actually meet. Block models may reflect uneven sampling because unrecorded interactions are treated as zeros (Banville et al. 2025). In the Ecuador comparison, species that were present but never recorded interacting are absent from the inferred assemblages, and equal-probability sampling from the regional pool omits environmental filtering and dispersal differences. The pooled block model also includes information from every focal site, so the case is a structural reanalysis rather than an independent prediction test. Users can include abundance, phenology, or environmental effects when these are part of the interaction model, but MetaWeave does not infer them automatically. Uncertainty can be explored by repeating reconstructions with alternative

assemblages or models. Network metrics such as connectance, nestedness, or modularity should therefore be treated as model-based predictions until validated with independent local observations (Dansereau, Barros, and Poisot 2024; Banville et al. 2025).

MetaWeave 1.0.0 also includes an experimental `maxent_model()` option for generating binary-network ensembles constrained by connectance, an exact link count, or a specified degree sequence (Banville, Gravel, and Poisot 2023). These ensembles can support null-model tests and sensitivity analyses of network properties such as nestedness or modularity; those properties are outputs to evaluate, not constraints imposed by this function. The link-count route samples uniformly under its exact count constraint. The degree-sequence route uses an approximate swap-chain sampler without a built-in convergence diagnostic, so its sampling behavior requires further evaluation. Future priorities include improved uncertainty propagation, more informative error diagnostics, and more memory-efficient processing for large spatial applications.

Software and data availability

MetaWeave 1.0.0 is available on CRAN under the MIT license and requires R ≥ 4.1.0. The package imports `terra` for raster operations and base R's `methods` and `stats` packages for object methods and statistical operations; its source code, documentation, and installation instructions are available from the MetaWeave repository. The synthetic benchmark and palm–mammal workflows are supplied in Supplementary materials 1 and 2. Ecuador interaction records and traits are archived by Muñoz (2021); Supplementary

material 3 contains the pooled-SBM fit, local reconstruction, and table and figure scripts used here.

REFERENCES

Banville, Francis, Dominique Gravel, and Timothée Poisot. 2023. “What Constrains Food Webs? A Maximum Entropy Framework for Predicting Their Structure with Minimal Biases.” *PLOS Computational Biology* 19 (9): e1011458. https://doi.org/10.1371/journal.pcbi.1011458.

Banville, Francis, Tanya Strydom, Penelope S. A. Blyth, Chris Brimacombe, Michael D. Catchen, Gabriel Dansereau, Gracielle Higino, et al. 2025. “Deciphering Probabilistic Species Interaction Networks.” *Ecology Letters* 28 (6): e70161. https://doi.org/10.1111/ele.70161.

Dansereau, Gabriel, Ceres Barros, and Timothée Poisot. 2024. “Spatially Explicit Predictions of Food Web Structure from Regional-Level Data.” *Philosophical Transactions of the Royal Society B: Biological Sciences* 379: 20230166. https://doi.org/10.1098/rstb.2023.0166.

Hijmans, Robert J. 2025. *Terra: Spatial Data Analysis*. https://doi.org/10.32614/CRAN.package.terra.

Marjakangas, Emma-Liina, Gabriel Muñoz, Shaun Turney, Jörg Albrecht, Eike L. Neuschulz, Matthias Schleuning, and Jean-Philippe Lessard. 2022. “Trait-Based Inference of Ecological Network Assembly: A Conceptual Framework and Methodological Toolbox.” *Ecological Monographs* 92 (2): e1502. https://doi.org/10.1002/ecm.1502.

Montoya, José M., Stuart L. Pimm, and Ricard V. Solé. 2006. “Ecological Networks and Their Fragility.” *Nature* 442: 259–64. https://doi.org/10.1038/nature04927.

Muñoz, Gabriel. 2021. “TraitBasedNetworkAssembly: Publication Release.” Zenodo. https://doi.org/10.5281/zenodo.5537283.

Muñoz, Gabriel, Kristian Trøjelsgaard, and W. Daniel Kissling. 2019. “A Synthesis of Animal-Mediated Seed Dispersal of Palms Reveals Distinct Biogeographical Differences in Species Interactions.” *Journal of Biogeography* 46: 466–84. https://doi.org/10.1111/jbi.13493.

Ohlmann, Marc, Jimmy Garnier, and Laurent Vuillon. 2023. “Metanetwork: A r Package Dedicated to Handling and Representing Trophic Metanetworks.” *Ecology and Evolution* 13 (8): e10229. https://doi.org/10.1002/ece3.10229.

Oliveira, C. V., G. Alves-Ferreira, F. M. M. Mota, D. C. Talora, and N. M. Heming. 2025. “Net.raster: Interaction Network Metrics for Raster Data.” *Ecological Informatics* 86: 102969. https://doi.org/10.1016/j.ecoinf.2024.102969.

Saravia, Leonardo A., Tomás I. Marina, Nadiah P. Kristensen, Marleen De Troch, and Fernando R. Momo. 2022. “Ecological Network Assembly: How the Regional Metaweb

Influences Local Food Webs." *Journal of Animal Ecology* 91 (3): 630–42. https://doi.org/10.1111/1365-2656.13652.

Strydom, Tanya, Michael D. Catchen, Francis Banville, Dominique Caron, Gabriel Dansereau, Philippe Desjardins-Proulx, Natalia R. Forero-Muñoz, et al. 2021. "A Roadmap Towards Predicting Species Interaction Networks (Across Space and Time)." *Philosophical Transactions of the Royal Society B: Biological Sciences* 376: 20210063. https://doi.org/10.1098/rstb.2021.0063.

Terry, Christopher. 2024. *cassandRa: Finds Missing Links and Metric Confidence Intervals in Ecological Bipartite Networks*. https://doi.org/10.32614/CRAN.package.cassandRa.

Supplementary material 1

A benchmark for local network inference with known-processes

Table of contents

Reconstruct local networks with MetaWeave 1

Generate regional and local interaction probabilities 3

Apply the known process as a custom MetaWeave model 3

Compare predictions with the known process 5

Connectance and binary interactions 7

If interaction probabilities change with the environment, how well can regional models recover them at individual sites? We explore this question with a simulated landscape of 225 cells, 24 producer species and 18 consumer species. For each possible pair, the simulation defines a known probability of interaction in each cell. We can compare predictions with those known values because we created the data ourselves.

The file `data/benchmark/benchmark_objects.rds` contains the regional model parameters and the known local probabilities. The code below recreates the species range maps using the original random seed. We compare a **block model**, in which species in the same pair of groups share an interaction probability, with a **species-pair model**, in which every pair has its own regional probability. Both models use the same species present in each cell. We then compare their predictions with the known probabilities from the simulation.

Reconstruct local networks with MetaWeave

The complete benchmark is supplied as an R script. Run the following command from the supplementary-material directory to regenerate the simulation, model results and scores. It is displayed without execution here; the document loads the saved results for a fast render.

```
source("scripts/synthetic_benchmark.R")
```

This chunk rebuilds the species ranges and passes them to `new_distribution_collection()` to identify the producer and consumer groups. `block_model()` creates a regional rule from block memberships and block-pair probabilities; `probability_matrix_model()` creates another rule from species-pair probabilities. We give each rule the same ranges in `run_spatial_inference()`, so their local predictions differ only in the interaction assumption.

```
# Build species ranges, then apply two regional interaction rules.
library(metaweave)
```

```r
library(terra)
library(tidyverse)

benchmark <- readRDS("data/benchmark/benchmark_objects.rds")

# Create an environmental gradient and reproducible species ranges.
set.seed(20260809)
grid <- rast(nrows = 15, ncols = 15, xmin = 0, xmax = 1,
             ymin = 0, ymax = 1, crs = "EPSG:4326")
xy <- xyFromCell(grid, seq_len(ncell(grid)))
environment <- as.numeric(scale(xy[, 1]))

make_traits <- function(species, centre) {
  data.frame(species = species,
             optimum = runif(length(species), -1.5, 1.5),
             breadth = runif(length(species), 0.65, 1.1),
             intercept = rnorm(length(species), centre, 0.25))
}

producer_traits <- make_traits(sprintf("producer_%02d", 1:24), 1.1)
consumer_traits <- make_traits(sprintf("consumer_%02d", 1:18), 0.9)

make_ranges <- function(traits) {
  layers <- lapply(seq_len(nrow(traits)), function(i) {
    chance <- plogis(traits$intercept[i] -
      (environment - traits$optimum[i])^2 / (2 * traits$breadth[i]^2))
    presence <- rbinom(length(chance), 1L, chance)
    layer <- rast(grid)
    values(layer) <- ifelse(presence == 1L, 1, NA_real_)
    names(layer) <- traits$species[i]
    layer
  })
  rast(layers)
}


species_ranges <- list(
  producers = new_distribution_collection(make_ranges(producer_traits),
                                          guild = "producers",
                                          metadata = producer_traits),
  consumers = new_distribution_collection(make_ranges(consumer_traits),
                                          guild = "consumers",
                                          metadata = consumer_traits)
)
```

The next chunk defines the two regional models and applies them to the same species ranges. `run_spatial_inference()` identifies the species in each cell and returns a local probability matrix and summaries. The code is shown for readers; the document uses the saved results to render quickly.

```r
# Compare block-level and species-pair interaction rules.
block_rule <- block_model(
  benchmark$producer_lookup, benchmark$consumer_lookup,
  benchmark$theta, "producers", "consumers"
)
pair_rule <- probability_matrix_model(
  benchmark$regional_probability, "producers", "consumers"
)

block_result <- run_spatial_inference(species_ranges, block_rule)
pair_result <- run_spatial_inference(species_ranges, pair_rule)
```

Generate regional and local interaction probabilities

We generate interaction probabilities in three steps. First, we assign each producer and consumer to one of three groups and set an interaction probability for each combination of groups. Second, we add random variation among species pairs within each combination. This gives each pair its own regional probability. Third, we give each pair an environmental optimum: its local probability increases where the environment is closer to that optimum.

The code below shows these calculations. It uses the group assignments and block probabilities stored in the supplied benchmark file. The resulting local probabilities are known because we define the process that generates them; they provide the reference for evaluating the alternative models. The code is displayed without running during rendering.

```r
# Continue the original random sequence after generating the ranges.
producer_names <- producer_traits$species
consumer_names <- consumer_traits$species
producer_blocks <- benchmark$producer_lookup$guild
consumer_blocks <- benchmark$consumer_lookup$guild
block_probability <- benchmark$theta[producer_blocks, consumer_blocks]
pair_effect <- matrix(rnorm(24 * 18, 0, 0.55), 24, 18)
pair_probability <- plogis(qlogis(block_probability) + pair_effect)
pair_optimum <- matrix(runif(24 * 18, -1.5, 1.5), 24, 18)
dimnames(pair_probability) <- dimnames(pair_optimum) <-
  list(producer_names, consumer_names)

# For a cell, retain its species and adjust probabilities by environment.
known_probabilities <- function(plants, consumers, cell_environment) {
  base <- pair_probability[plants, consumers, drop = FALSE]
  optimum <- pair_optimum[plants, consumers, drop = FALSE]
  context_match <- exp(-0.5 * ((cell_environment - optimum) / 0.8)^2)
  plogis(qlogis(base) - 0.9 + 1.8 * context_match)
}
```

Apply the known process as a custom MetaWeave model

The third model is an **oracle**: it is given the process that generated the data, including the environment at each cell. It is a check that MetaWeave carries a supplied prediction

through the workflow; its perfect recovery is expected and does not measure performance on independent data. The code below shows how a custom prediction function returns a local network and how `new_inference_model()` makes it available to MetaWeave.

```
# Turn the known process into a local prediction rule.
context_predict <- function(model, assemblage) {
  p <- model$parameters
  plants <- assemblage$species[["producers"]]
  consumers <- assemblage$species[["consumers"]]
  local_environment <- (assemblage$coordinates[[1]] - p$x_center) / p$x_scale
  base <- p$regional_probability[plants, consumers, drop = FALSE]
  optimum <- p$pair_optimum[plants, consumers, drop = FALSE]
  match <- exp(-0.5 * ((local_environment - optimum) / 0.8)^2)
  probability <- plogis(qlogis(pmin(pmax(base, 1e-8), 1 - 1e-8)) -
                          0.9 + 1.8 * match)
  new_ecological_network(probability, type = "probability",
                         metadata = list(cell_id = assemblage$cell_id,
                                         model = model$name,
                                         environment = local_environment),
                         row_group = "producers", column_group = "consumers")
}
oracle_rule <- new_inference_model(
  predict = context_predict, name = "known-process model",
  parameters = list(regional_probability = pair_probability,
                    pair_optimum = pair_optimum,
                    x_center = mean(xy[, 1]), x_scale = sd(xy[, 1]))
)
oracle_result <- run_spatial_inference(species_ranges, oracle_rule)
```

The table below shows `theta`, the three-by-three matrix of probabilities between producer and consumer groups. The species-pair matrix has 24 rows and 18 columns. These two matrices define the regional rules passed to MetaWeave; the known local probabilities provide the reference for the comparison.

```
# Display the regional probabilities supplied to MetaWeave.
as_tibble(benchmark$theta, rownames = "producer_block") |>
  knitr::kable(digits = 2, caption = "Interaction probabilities between
producer and consumer groups.")
```

Interaction probabilities between producer and consumer groups.

producer_block	C1	C2	C3
P1	0.70	0.22	0.08
P2	0.15	0.68	0.25
P3	0.10	0.28	0.72

```
tibble(producer_species = nrow(benchmark$regional_probability),
       consumer_species = ncol(benchmark$regional_probability),
       mean_pair_probability = mean(benchmark$regional_probability)) |>
  knitr::kable(digits = 3, caption = "Regional species-pair model.")
```

Regional species-pair model.

producer_species	consumer_species	mean_pair_probability
24	18	0.356

Compare predictions with the known process

The archived `context_model` result contains the exact probability matrix used to generate interactions in each cell. This chunk takes the local matrices in each `run_spatial_inference()` result and compares them with those known matrices. A small root-mean-square error (RMSE) means closer recovery of the known probabilities. To assess predictions against one possible binary realization, we draw links from the known probabilities with the same recorded seed for each cell and model.

```
# Score each local prediction against the same known process.
truth <- benchmark$results$context_model$result
models <- list(block = block_result$result, species_pair =
pair_result$result,
               known_process = truth)

score_one <- function(predicted, known, cell_id) {
  p <- predicted$matrix
  q <- known$matrix
  set.seed(100000L + cell_id)
  observed <- matrix(rbinom(length(q), 1L, as.vector(q)), nrow(q), ncol(q))
  safe <- pmin(pmax(p, 1e-8), 1 - 1e-8)
  tibble(
    probability_rmse = sqrt(mean((p - q)^2)),
    brier = mean((p - observed)^2),
    log_loss = -mean(observed * log(safe) + (1 - observed) * log(1 - safe)),
    true_connectance = mean(q), predicted_connectance = mean(p)
  )
}

cell_scores <- map_dfr(names(models), function(label) {
  output <- models[[label]]
  map_dfr(seq_along(output$networks), function(i) {
    score_one(output$networks[[i]], truth$networks[[i]],
              output$index$cell_id[i]) |>
      mutate(model = label, cell_id = output$index$cell_id[i], .before = 1)
  })
})

cell_scores |>
  as_tibble() |>
  group_by(model) |>
  summarise(across(c(probability_rmse, brier, log_loss), mean),
            .groups = "drop") |>
```

```
  knitr::kable(digits = 3, caption = "Mean prediction scores across cells;
lower values indicate better recovery.")
```

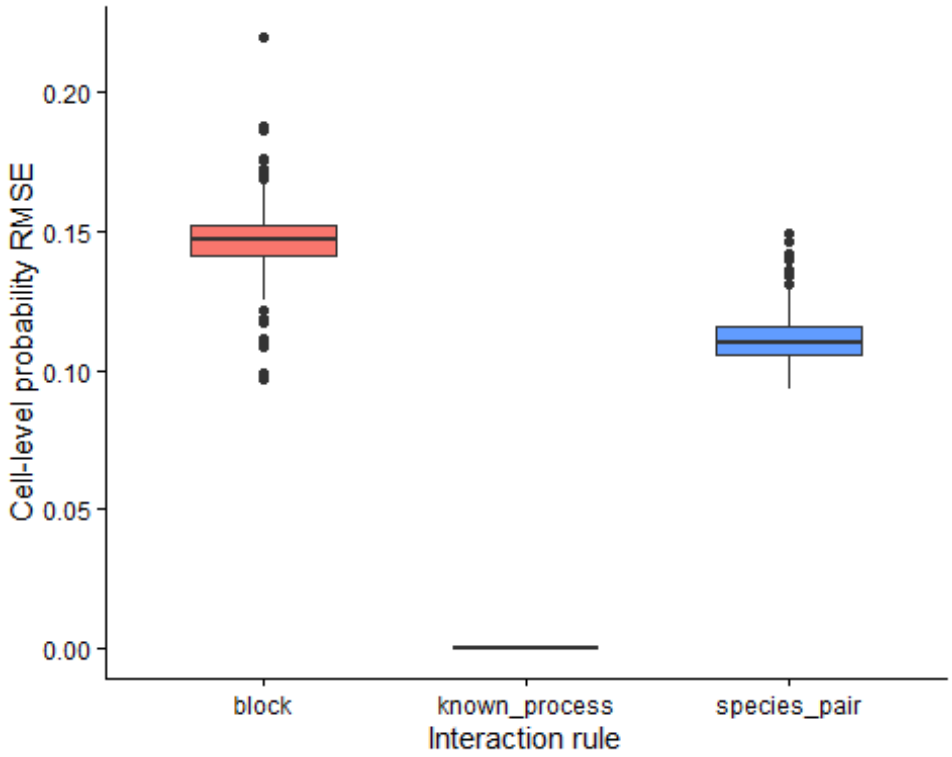
Mean prediction scores across cells; lower values indicate better recovery.

model	probability_rmse	brier	log_loss
block	0.147	0.183	0.547
known_process	0.000	0.161	0.491
species_pair	0.112	0.172	0.521

The known-process model has probability RMSE zero by construction. Its nonzero score against sampled binary interactions reflects random realization, not failure to recover its own probabilities. In the original benchmark, the species-pair model had lower mean probability RMSE than the block model (about 0.112 versus 0.147); the calculation above regenerates that comparison from MetaWeave's local matrices.

The boxplot shows how accuracy varies among cells in the two MetaWeave results. It displays the scores calculated above; it does not make another network prediction.

```
# Compare the distribution of prediction errors across cells.
ggplot(as_tibble(cell_scores),
       aes(x = model, y = probability_rmse, fill = model)) +
  geom_boxplot(width = 0.55, show.legend = FALSE) +
  labs(x = "Interaction rule", y = "Cell-level probability RMSE") +
  theme_classic()
```

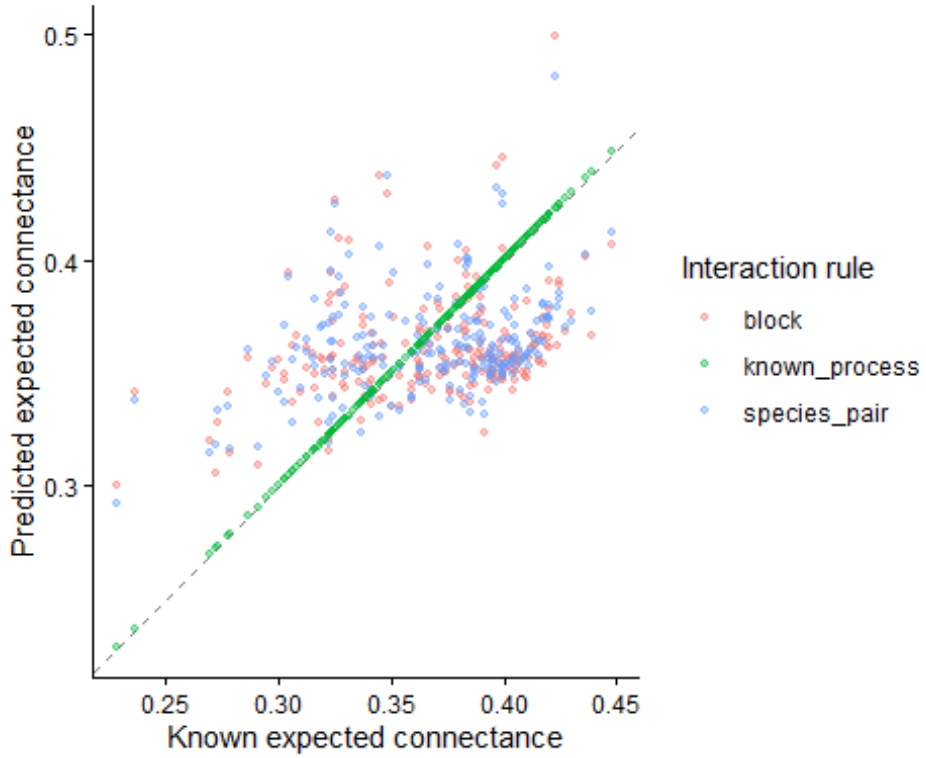


This benchmark demonstrates how a common MetaWeave workflow carries different interaction assumptions through the same landscape. The ranking depends on this particular generating process, which contains substantial species-pair variation; it is not a general ranking of model classes.

Connectance and binary interactions

The manuscript also compares expected connectance: the average interaction probability across possible pairs in a cell. Points close to the diagonal show that the predicted and known values agree. This summary can look good even when some individual pair probabilities differ, which is why it complements the RMSE above.

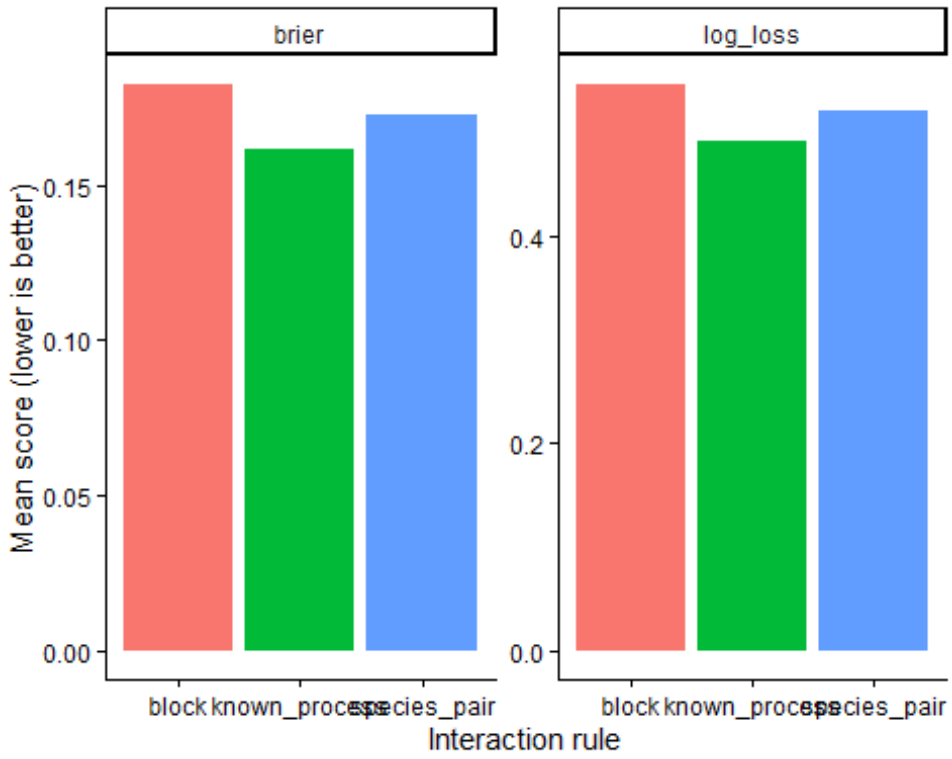
```
# Compare predicted and generating connectance for every cell.
ggplot(cell_scores, aes(true_connectance, predicted_connectance, colour =
model)) +
  geom_abline(slope = 1, intercept = 0, linetype = 2, colour = "grey50") +
  geom_point(alpha = 0.4, size = 1) +
  coord_equal() +
  labs(x = "Known expected connectance", y = "Predicted expected
connectance",
       colour = "Interaction rule") +
  theme_classic()
```



Brier score and log loss compare predicted probabilities with one sampled binary network per cell. Lower values indicate better predictions for that realization. The same sampled links are used for all three models, so the comparison is not affected by giving each model

a different set of observed links. Even the oracle has nonzero scores because an interaction sampled from a probability is random.

```
# Summarize the two scores for the sampled binary links.
cell_scores |>
  group_by(model) |>
  summarise(across(c(brier, log_loss), mean), .groups = "drop") |>
  pivot_longer(c(brier, log_loss), names_to = "score", values_to = "value")
|>
  ggplot(aes(model, value, fill = model)) +
  geom_col(show.legend = FALSE) +
  facet_wrap(~ score, scales = "free_y") +
  labs(x = "Interaction rule", y = "Mean score (lower is better)") +
  theme_classic()
```



Supplementary material 2

From a palm–mammal metaweb to local interaction maps

Table of contents

Question and study system 1

Inputs and regional interaction rules 1

Reconstruct the same local assemblages 4

One cell in detail 5

Patterns across the landscape 6

Interpretation and limits 8

Question and study system

Where do two interpretations of regional palm–mammal frugivory records predict different local networks? The regional data document which pairs have interacted somewhere in the Neotropics. The matched range maps cover 51 palms and 68 mammals on a 1° grid. A cell is eligible when it contains at least one species from each group. These ranges indicate potential co-occurrence, not verified occupancy or interaction at every cell.

We compare two interaction rules using the same local species lists. The **recorded metaweb** is a table of regional interactions: a pair receives 1 if it was documented and 0 otherwise. A zero means no record in this dataset; it does not prove the species cannot interact. The **stochastic block model** (SBM) groups species by their regional interaction patterns. It assigns a probability to each combination of palm and mammal groups. We provide the fitted model and show how MetaWeave uses it to predict local networks.

Inputs and regional interaction rules

The original interaction records are from Muñoz, Trøjelsgaard and Kissling (2019). We include a copy of the records, two sets of range maps, and a file containing the regional interaction matrix and fitted SBM parameters. The species names in these inputs have already been matched. These are the inputs used for the manuscript maps. The supplement starts with the prepared maps; it does not rebuild them from the original range sources.

The first chunk reads those inputs before constructing a MetaWeave model. It reports the size of the matched regional network and fitted SBM, so the species pool used by the later `probability_matrix_model()` and `block_model()` calls is clear.

```
# Read regional evidence and the matched spatial inputs.
library(metaweave)
library(terra)
library(tidyverse)

records <- read.csv("data/palm-mammal/PalmFrugDatasetOCT2018.csv")
regional_records <- records |>
  filter(frugClass == "MAMMAL", biogeographicRegion == "Neotropics")
palms <- rast("data/palm-mammal/palms.tif")
mammals <- rast("data/palm-mammal/mammals.tif")
regional <- readRDS("data/palm-mammal/fitted-models.rds")

tibble(
  source_records = nrow(regional_records),
  palms = nrow(regional$regional_web),
  mammals = ncol(regional$regional_web),
  documented_pairs = sum(regional$regional_web),
  palm_blocks = nrow(regional$theta_sbm),
  mammal_blocks = ncol(regional$theta_sbm)
) |>
  pivot_longer(everything(), names_to = "Measure", values_to = "Value") |>
  knitr::kable(caption = "Regional interaction data and fitted model.")
```

Regional interaction data and fitted model.

Measure	Value
source_records	581
palms	51
mammals	68
documented_pairs	234
palm_blocks	7
mammal_blocks	7

The regional matrix contains 234 documented pairs among the species with matched range maps. Some original records involve species outside this matched set and are therefore excluded. We fitted an SBM to the matrix using `cassandRa`. The selected model placed palms in seven groups and mammals in seven groups according to their interaction patterns. Each entry in `theta_sbm` gives the estimated interaction probability between one palm group and one mammal group.

The code below shows how the regional matrix was passed to `cassandRa` and how the species groups and their interaction probabilities were obtained. MetaWeave uses these quantities to assign probabilities to local species pairs. The fit is supplied with the supplement, so this code is displayed without refitting the model when the document renders. A new fit can differ because the fitting procedure is stochastic.

```
# Convert the matched regional network to cassandRa's input form.
cass_network <- cassandRa::CreateListObject(regional$regional_web)
fits <- cassandRa::FitSBM(list = cass_network, n_SBM = 10, G = NULL)
best <- fits$SBM1

# These are the quantities that MetaWeave's block_model() needs.
block_probabilities <- best$Omega_rs
palm_blocks <- best$SB_H
mammal_blocks <- best$SB_W
```

The following table shows the estimated interaction probability for each combination of palm and mammal groups:

```
# Display the fitted probabilities used to reconstruct local networks.
theta_table <- regional$theta_sbm
rownames(theta_table) <- paste0("P", seq_len(nrow(theta_table)))
colnames(theta_table) <- paste0("M", seq_len(ncol(theta_table)))
as_tibble(theta_table, rownames = "Palm group") |>
  knitr::kable(digits = 2, caption = "Interaction probabilities for palm (P)
and mammal (M) groups.")
```

Interaction probabilities for palm (P) and mammal (M) groups.

Palm group	M1	M2	M3	M4	M5	M6	M7
P1	0.00	0.15	0.00	0.02	0.25	0.04	0.04
P2	0.00	0.03	0.01	0.27	0.07	0.00	0.00
P3	0.01	0.39	0.00	0.00	0.02	0.15	0.00
P4	0.00	0.02	0.17	0.00	0.00	0.10	0.02
P5	0.28	0.10	0.02	0.05	0.00	0.42	0.00
P6	0.00	0.27	0.00	0.12	0.03	0.40	0.36
P7	0.00	0.50	0.03	0.21	0.00	0.00	0.06

MetaWeave accepts either regional representation as an interaction rule. `probability_matrix_model()` takes the recorded species-pair matrix, including its zeroes and ones. `block_model()` takes a block assignment for every retained species and the matrix of block-pair probabilities. Both objects tell MetaWeave how to fill a local interaction matrix.

```
# Express the two regional assumptions as interchangeable MetaWeave rules.
recorded_rule <- probability_matrix_model(
  regional$regional_web, "producers", "consumers"
)
sbm_rule <- block_model(
  regional$producer_lookup, regional$consumer_lookup,
  regional$theta_sbm, "producers", "consumers"
)
```

Reconstruct the same local assemblages

Each raster layer shows the range of one species. `run_spatial_inference()` finds the palms and mammals whose ranges overlap each eligible cell, forms their possible pairs, and applies the chosen regional rule. We apply both rules to the same ranges. The resulting tables contain coordinates, species richness, expected links and mean interaction probability for each cell. The code below joins these summaries by cell identifier. The local predictions are included with the supplement and loaded for rendering.

```
# Reconstruct the same assemblages and join their predictions by grid cell.
species_ranges <- list(producers = palms, consumers = mammals)
recorded <- run_spatial_inference(species_ranges, recorded_rule)
sbm <- run_spatial_inference(species_ranges, sbm_rule)

# Compare the saved cell summaries from the two model runs.
recorded_cells <- as_tibble(recorded$result$index) |>
  select(cell_id, x, y, row_richness, column_richness,
         recorded_links = expected_links,
         recorded_connectance = mean_probability)
sbm_cells <- as_tibble(sbm$result$index) |>
  select(cell_id, sbm_links = expected_links,
         sbm_connectance = mean_probability)
cells <- recorded_cells |>
  left_join(sbm_cells, by = "cell_id") |>
  mutate(link_difference = sbm_links - recorded_links,
         connectance_difference = sbm_connectance - recorded_connectance)

cells |>
  summarise(eligible_cells = n(),
            mean_palms = mean(row_richness),
            mean_mammals = mean(column_richness),
            mean_recorded_links = mean(recorded_links),
            mean_sbm_links = mean(sbm_links)) |>
  pivot_longer(everything(), names_to = "Measure", values_to = "Value") |>
  knitr::kable(digits = 2, caption = "Mean local network summaries across
grid cells.")
```

Mean local network summaries across grid cells.

Measure	Value
eligible_cells	1254.00
mean_palms	8.41
mean_mammals	13.60
mean_recorded_links	21.17
mean_sbm_links	12.11

The recorded rule produces a binary local matrix: a regional pair is retained wherever both species' ranges cover the cell. The SBM produces a probability matrix. Its **expected links** are the sum of those probabilities, not a sampled number of observed interactions. The mean probability across possible pairs is expected connectance. It helps distinguish interaction density from the effect of having more species and more possible pairs.

One cell in detail

The cell with the most recorded links offers a concrete comparison. Each `run_spatial_inference()` result also retains its local matrices in `$result$networks`. This chunk retrieves the same cell from both results and separates the SBM's expected-link total into contributions from recorded and unrecorded regional pairs.

```
# Inspect how recorded and unrecorded pairs contribute in one cell.
example <- cells |> slice_max(recorded_links, n = 1, with_ties = FALSE)
recorded_position <- match(example$cell_id, recorded$result$index$cell_id)
sbm_position <- match(example$cell_id, sbm$result$index$cell_id)
recorded_matrix <- recorded$result$networks[[recorded_position]]$matrix
sbm_matrix <- sbm$result$networks[[sbm_position]]$matrix

tibble(
  cell_id = example$cell_id,
  palms = nrow(recorded_matrix),
  mammals = ncol(recorded_matrix),
  possible_pairs = length(recorded_matrix),
  recorded_links = sum(recorded_matrix),
  sbm_expected_on_recorded_pairs = sum(sbm_matrix[recorded_matrix == 1]),
  sbm_expected_on_unrecorded_pairs = sum(sbm_matrix[recorded_matrix == 0]),
  sbm_expected_links = sum(sbm_matrix)
) |>
  pivot_longer(everything(), names_to = "Measure", values_to = "Value") |>
  knitr::kable(digits = 2, caption = "Recorded and expected links in the
example cell.")
```

Recorded and expected links in the example cell.

Measure	Value
cell_id	34666.00
palms	17.00
mammals	21.00
possible_pairs	357.00
recorded_links	57.00
sbm_expected_on_recorded_pairs	12.97
sbm_expected_on_unrecorded_pairs	13.71
sbm_expected_links	26.67

The SBM can add expected links among unrecorded pairs while reducing the contribution of recorded pairs from exactly one to a probability below one. Its total can therefore be lower even when it gives unrecorded pairs positive probability. `SBM minus recorded` is a **signed difference in expected totals**, not a count of newly proposed links.

Patterns across the landscape

The five maps below correspond to the manuscript figure. Richness maps show the species pool in each cell. The middle two maps show local links under each regional rule on the same colour scale. The final map shows where the rules disagree. Blank areas contain no eligible assemblage in these inputs. The next chunk defines a plotting helper for the cell summaries returned by `run_spatial_inference()`; it performs no further network inference.

```
# Define consistent colour scales for the spatial comparison.
library(ggplot2)
library(patchwork)

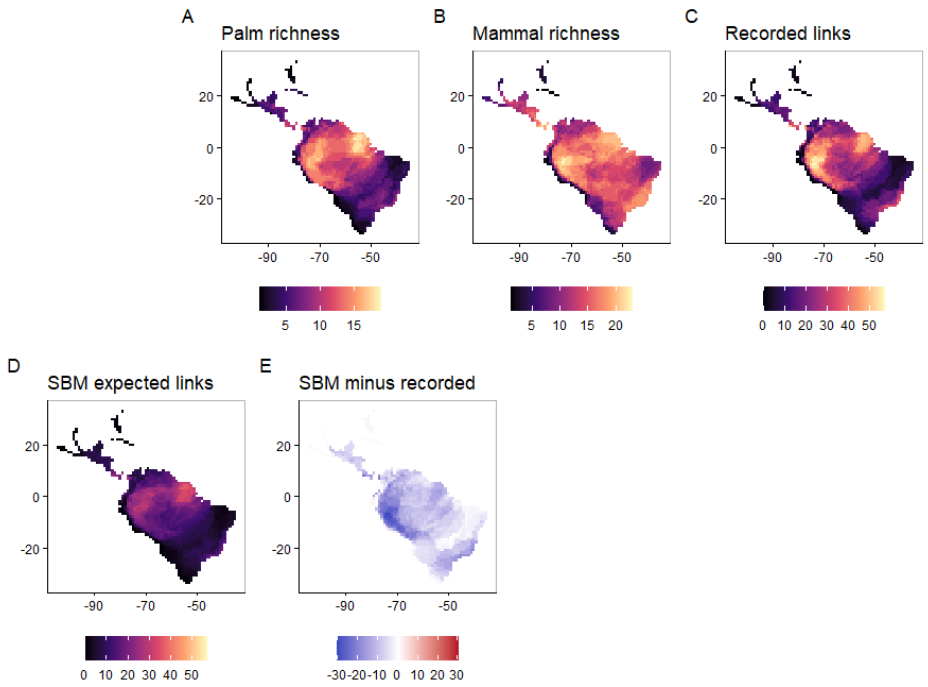
map_panel <- function(variable, title, limits = NULL, difference = FALSE) {
  p <- ggplot(cells, aes(x = x, y = y, fill = .data[[variable]])) +
    geom_raster() + coord_equal() +
    labs(title = title, x = NULL, y = NULL, fill = NULL) +
    theme_classic() +
    theme(panel.border = element_rect(fill = NA, colour = "grey70"),
          legend.position = "bottom")
  if (difference) {
    p + scale_fill_gradient2(low = "#3B4CC0", mid = "white",
                             high = "#B40426", midpoint = 0,
                             limits = limits)
  } else {
    p + scale_fill_viridis_c(option = "magma", limits = limits)
  }
}

link_max <- max(c(cells$recorded_links, cells$sbm_links))
difference_max <- max(abs(cells$link_difference))
```

The plotting chunk maps the two MetaWeave outputs side by side and uses their joined cell summary to calculate the signed difference. Panels C and D share a colour range, making their predicted link totals directly comparable.

```
# Map richness and local predictions on the shared grid.
panel_a <- map_panel("row_richness", "Palm richness")
panel_b <- map_panel("column_richness", "Mammal richness")
panel_c <- map_panel("recorded_links", "Recorded links", c(0, link_max))
panel_d <- map_panel("sbm_links", "SBM expected links", c(0, link_max))
panel_e <- map_panel("link_difference", "SBM minus recorded",
                     c(-difference_max, difference_max), TRUE)
```

```
(panel_a | panel_b | panel_c) /
  (panel_d | panel_e | plot_spacer()) +
  plot_annotation(tag_levels = "A")
```



Local palm–mammal network reconstructions from the same species ranges. (A–B) Palm and mammal richness. (C) Recorded regional pairs whose species co-occur. (D) Expected links from the regional SBM. (E) SBM expected links minus recorded links, with zero at the colour midpoint. Maps represent potential interactions, not observed local encounters.

Across the analyzed cells, the recorded baseline has about 21.17 potential links per cell and the SBM about 12.11 expected links. The broad patterns are similar, while the largest signed differences in the manuscript map occur around the Yungas and Andean–Amazon transition. These are **unweighted cell averages**; geographic cells on a longitude–latitude grid do not have equal physical area. A cell's link total also depends partly on its species richness, so connectance is a useful companion measure.

The final table uses the same MetaWeave cell summaries to compare expected connectance, which accounts for the number of possible pairs, and the typical signed link difference. It summarizes predictions rather than adding observations or refitting the regional model.

```
# Summarize interaction density alongside the signed link difference.
cells |>
  summarise(mean_recorded_connectance = mean(recorded_connectance),
            mean_sbm_connectance = mean(sbm_connectance),
            median_link_difference = median(link_difference)) |>
  pivot_longer(everything(), names_to = "Measure", values_to = "Value") |>
```

```
  knitr::kable(digits = 3, caption = "Connectance and difference in expected
links.")
```

Connectance and difference in expected links.

Measure	Value
mean_recorded_connectance	0.189
mean_sbm_connectance	0.094
median_link_difference	-8.265

Interpretation and limits

MetaWeave makes the interaction assumption explicit while holding local composition fixed. The difference map identifies cells where sampling local networks would be especially useful for distinguishing the two regional interpretations. Neither prediction has been validated against independent local interaction observations here. Range overlap does not establish local occupancy, and a possible fruit interaction does not by itself establish effective seed dispersal.

The included range maps reproduce the spatial comparison for the matched species. Their original sources, versions, rasterization steps, taxonomic decisions and redistribution terms still need to be documented before the data package is publicly distributed.

Supplementary material 3

Testing interaction assembly in Ecuadorian plant–frugivore networks

Table of contents

Ecological question 1

Load the regional model and reconstruct a site 2

How the regional fit was obtained 3

Generate the comparison 5

Sites along the gradient 6

Read the contrasts 7

Ecological question

Can regional interaction structure help explain the nestedness and modularity of six observed local plant–frugivore networks in Ecuador? The records come from the case study presented in Marjakangas et al., 2022. The included data contain observed interactions at six sites across an elevational gradient in the Andes of Ecuador. We combine all recorded species pairs into one regional network, also called a metaweb, and fit one stochastic block model (SBM) to it. The SBM groups species by their interaction patterns and estimates an interaction probability for each pair of groups.

This supplement includes the records, the fitted regional model, and results from 49 simulated networks for each scenario at each site. The document loads the fitted model and simulation results when it renders; the fitting and simulation code is shown for readers but is not run. Only one small local network is reconstructed to illustrate MetaWeave. The sites span 1,000, 2,000 and 3,000 m elevation, with two networks at each level. Elevation orders the comparison; it is not an input to the regional SBM. The question is whether its interaction probabilities and the species present locally lead to different structural predictions along this gradient.

We compare three ensembles at each site. Every simulated network has the observed number of distinct links.

Ensemble	Species	Link allocation
A: neutral	Observed local species	Equal chance among possible pairs
B: regional SBM	Observed local species	Weighted by regional SBM probabilities
C: changed composition	Random species from the	Weighted by the same

Ensemble	Species	Link allocation
	regional pools, matched to local plant and frugivore richness	regional SBM

B – A changes the interaction rule while holding composition fixed. **B – C** changes composition while holding the regional interaction rule fixed. These are differences in network metrics, not z scores. Random regional species are sampled with equal probability, an explicit simplifying assumption about arrival and establishment.

Load the regional model and reconstruct a site

We fit one SBM to the interactions pooled across all six sites. The model assigns each plant and frugivore to a group and estimates a probability of interaction for each combination of plant and frugivore groups. These assignments and probabilities are included with the supplement and form the regional input to MetaWeave. We use this same model at every site.

In the first chunk, `block_model()` turns those saved assignments and probabilities into one regional interaction rule. `new_assemblage()` lists the plants and frugivores recorded at a site, and `infer_network()` applies the rule to that assemblage. The returned matrix contains one interaction probability for each locally possible species pair.

```
# Apply the pooled regional SBM to a locally recorded assemblage.
library(metaweave)
library(tidyverse)

source_data <- new.env()
load("data/ecuador/dataSet1.RData", envir = source_data)
records <- source_data$dataSet1$N
records <- records |>
  filter(frequency > 0) |>
  mutate(across(c(site, plantCode, animalCode), as.character))

regional <- readRDS("data/ecuador/pooled_sbm_fit.rds")
regional_rule <- block_model(
  regional$row_lookup, regional$column_lookup, regional$theta,
  "plants", "animals"
)

sites <- sort(unique(records$site))
first_site <- records |> filter(site == sites[1])
local_plants <- sort(unique(first_site$plantCode))
local_animals <- sort(unique(first_site$animalCode))
community <- new_assemblage(
  list(plants = local_plants, animals = local_animals), cell_id = 1L
)
local_probabilities <- infer_network(community, regional_rule)$matrix
```

```
tibble(site = sites[1], plants = nrow(local_probabilities),
       frugivores = ncol(local_probabilities),
       expected_links = round(sum(local_probabilities), 1)) |>
  knitr::kable(digits = 1, caption = "An example local network inferred from
the regional SBM.")
```

An example local network inferred from the regional SBM.

site	plants	frugivores	expected_links
Bellavista	21	15	19.4

Each entry of `local_probabilities` is the regional model's probability for a pair whose species occur at this site. MetaWeave supplies this local matrix. The next step uses its values as weights in a fixed-link-count simulation; they are not independent Bernoulli draws.

How the regional fit was obtained

To fit the regional model, we first combine the records from all six sites into a single binary network. Its entries represent observed species pairs: a pair is marked present if recorded at least once. The following displayed code shows the essential fitting and translation steps. It is not rerun in this document; `cassandRa` is required only when fitting a new regional model.

```
# Make one binary metaweb from all sites, ignoring repeat frequency.
plant_pool <- sort(unique(records$plantCode))
animal_pool <- sort(unique(records$animalCode))
metaweb <- matrix(0L, length(plant_pool), length(animal_pool),
                  dimnames = list(plant_pool, animal_pool))
metaweb[cbind(match(records$plantCode, plant_pool),
              match(records$animalCode, animal_pool))] <- 1L

# Fit once regionally; do not refit the model at individual sites.
set.seed(5537283)
candidate_fits <- cassandRa::FitSBM(
  list = cassandRa::CreateListObject(metaweb), n_SBM = 10, G = NULL
)
best <- candidate_fits$SBM1
theta <- best$Omega_rs
rownames(theta) <- paste0("plant_block_", seq_len(nrow(theta)))
colnames(theta) <- paste0("animal_block_", seq_len(ncol(theta)))
row_lookup <- tibble(species = plant_pool,
                     guild = rownames(theta)[as.integer(best$SB_H)])
column_lookup <- tibble(species = animal_pool,
                        guild = colnames(theta)[as.integer(best$SB_W)])
```

The regional network contains 102 plants, 127 frugivores and 536 recorded species pairs. The selected SBM has ten plant groups and ten frugivore groups. The following table shows

the estimated probabilities between those groups, which `block_model()` uses to build local predictions.

```
# Display the regional network size and fitted interaction probabilities.
tibble(plants = nrow(regional$metaweb),
       frugivores = ncol(regional$metaweb),
       regional_links = sum(regional$metaweb),
       plant_blocks = nrow(regional$theta),
       frugivore_blocks = ncol(regional$theta)) |>
  pivot_longer(everything(), names_to = "Measure", values_to = "Value") |>
  knitr::kable(caption = "Regional network and fitted block model.")
```

Regional network and fitted block model.

Measure	Value
plants	102
frugivores	127
regional_links	536
plant_blocks	10
frugivore_blocks	10

```
theta_table <- regional$theta
colnames(theta_table) <- paste0("F", seq_len(ncol(theta_table)))
rownames(theta_table) <- paste0("P", seq_len(nrow(theta_table)))
as_tibble(theta_table, rownames = "Plant group") |>
  knitr::kable(digits = 2, caption = "Interaction probabilities for plant (P)
and frugivore (F) groups.")
```

Interaction probabilities for plant (P) and frugivore (F) groups.

Plant group	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
P1	0.03	0.07	0.00	0.00	0.00	0.22	0.17	0.08	0.01	0.08
P2	0.00	0.01	0.26	0.19	0.00	0.02	0.00	0.08	0.16	0.00
P3	0.00	0.00	0.62	0.38	0.39	0.16	0.04	0.44	0.48	0.04
P4	0.00	0.00	0.36	0.00	0.00	0.00	0.00	0.00	0.19	0.00
P5	0.00	0.01	0.00	0.00	0.00	0.00	0.02	0.00	0.00	0.16
P6	0.00	0.00	0.14	0.00	0.00	0.00	0.00	0.10	0.00	0.00
P7	0.02	0.18	0.00	0.00	0.00	0.00	0.05	0.01	0.00	0.01
P8	0.00	0.00	0.00	0.00	0.00	0.01	0.21	0.00	0.00	0.00
P9	0.12	0.00	0.62	0.00	0.00	0.00	0.00	0.23	0.41	0.00
P10	0.27	0.00	0.00	0.00	0.00	0.00	0.04	0.00	0.00	0.00

Generate the comparison

The following code constructs the three sets of simulated networks. For each site, it starts with the recorded species and links, then draws networks under each of the three rules described above. The regional model is fitted once to the pooled network; it is not refitted for each draw.

```
# Generate the three ensembles using the pooled regional model.
  library(vegan)
  library(bipartite)

  local_matrix <- function(x) {
    plants <- sort(unique(x$plantCode))
    animals <- sort(unique(x$animalCode))
    m <- matrix(0L, length(plants), length(animals),
                dimnames = list(plants, animals))
    m[cbind(match(x$plantCode, plants),
            match(x$animalCode, animals))] <- 1L
    m
  }

  probabilities <- function(plants, animals, id) {
    assemblage <- new_assemblage(
      list(plants = plants, animals = animals), cell_id = id
    )
    infer_network(assemblage, regional_rule)$matrix
  }


  fixed_links <- function(p, n, weighted) {
    positions <- sample.int(length(p), n, replace = FALSE,
                            prob = if (weighted) as.vector(p) else NULL)
    m <- matrix(0L, nrow(p), ncol(p))
    m[positions] <- 1L
    m
  }


  network_metrics <- function(m) {
    c(nodf = as.numeric(vegan::nestednodf(m)$statistic[["NODF"]]),
      modularity = as.numeric(bipartite::computeModules(m)@likelihood))
  }
```

Within each draw, `new_assemblage()` supplies either the observed species or a richness-matched selection from the regional pool to `infer_network()`. MetaWeave reconstructs the local probability matrix under the *same* regional `block_model()` each time. The subsequent link sampling and NODF and modularity calculations use base R, vegan and

`bipartite`; they follow MetaWeave's prediction. The document shows the code and loads its saved output for the tables and figure.

```
repetitions <- 49L
set.seed(5537283)
plant_pool <- sort(unique(records$plantCode))
animal_pool <- sort(unique(records$animalCode))
scenarios <- c("A_observed_neutral", "B_observed_sbm", "C_random_sbm")
draws <- array(NA_real_, c(length(sites), repetitions, 3L, 2L),
               dimnames = list(sites, NULL, scenarios,
                               c("nodf", "modularity")))
observed <- matrix(NA_real_, length(sites), 2L,
                   dimnames = list(sites, c("nodf", "modularity")))
site_info <- data.frame(site = sites, elevation = NA_real_,
                        plants = NA_integer_, animals = NA_integer_,
                        links = NA_integer_)

for (i in seq_along(sites)) {
  x <- records[records$site == sites[i], ]
  m <- local_matrix(x)
  n_links <- sum(m)
  site_info$elevation[i] <- unique(x$elevation)
  site_info$plants[i] <- nrow(m)
  site_info$animals[i] <- ncol(m)
  site_info$links[i] <- n_links
  observed[i, ] <- network_metrics(m)
  p_local <- probabilities(rownames(m), colnames(m), i)

  for (r in seq_len(repetitions)) {
    random_plants <- sort(sample(plant_pool, nrow(m)))
    random_animals <- sort(sample(animal_pool, ncol(m)))
    p_random <- probabilities(random_plants, random_animals, i)
    draws[i, r, "A_observed_neutral", ] <-
      network_metrics(fixed_links(p_local, n_links, FALSE))
    draws[i, r, "B_observed_sbm", ] <-
      network_metrics(fixed_links(p_local, n_links, TRUE))
    draws[i, r, "C_random_sbm", ] <-
      network_metrics(fixed_links(p_random, n_links, TRUE))
  }
}
analysis <- list(site_info = site_info, observed = observed,
                 draws = draws, reps = repetitions, seed = 5537283L)
```

Sites along the gradient

The number of plants, frugivores and distinct links varies among sites. Every ensemble retains its site's plant richness, frugivore richness and link count, so the contrasts below do not arise simply from drawing networks of different sizes.

The following table lists the elevation, recorded species richness and number of distinct links at each site. These describe the observed networks used to set the size of each simulation.

```
# Describe the observed assemblages in elevation order.
site_table <- analysis$site_info |>
  as_tibble() |>
  arrange(elevation, site) |>
  select(site, elevation, plants, frugivores = animals, links)
site_table |>
  knitr::kable(col.names = c("Site", "Elevation (m)", "Plants", "Frugivores",
"Links"),
               caption = "Observed networks along the elevation gradient.")
```

Observed networks along the elevation gradient.

Site	Elevation (m)	Plants	Frugivores	Links
Bombuscaro	1000	24	50	182
Copalinga	1000	21	74	210
ECSF	2000	17	34	66
Finca	2000	20	38	82
Bellavista	3000	21	15	36
Cajanuma	3000	16	13	32

Read the contrasts

Nestedness (binary NODF) is higher when specialists' links tend to be subsets of generalists' links. Modularity is higher when interactions are more concentrated within groups. For each site we compare B with A and C draw by draw, then show the mean difference and its central 95% simulation interval. A circle in the figure represents **B – A**, the effect of using regional interaction weights for the observed species; a triangle represents **B – C**, the effect of keeping the observed species instead of a richness-matched regional draw. Neither symbol is an observed network metric.

The contrast chunk subtracts metrics calculated after sampling links from MetaWeave's local probability matrices. It pairs draws within a site and reports the mean and central interval of each metric difference. The subtraction is in NODF or modularity units, not a standardized score.

```
# Calculate interaction-rule and composition contrasts within each site.
contrast <- function(metric, reference) {
  b <- analysis$draws[, , "B_observed_sbm", metric]
  other <- analysis$draws[, , reference, metric]
  difference <- b - other
  tibble(
    site = dimnames(analysis$draws)[[1]],
    metric = metric,
```

```
    contrast = paste("B -", substr(reference, 1, 1)),
    mean = rowMeans(difference),
    lower = apply(difference, 1, quantile, 0.025),
    upper = apply(difference, 1, quantile, 0.975)
  )
}

comparison <- bind_rows(
  contrast("nodf", "A_observed_neutral"),
  contrast("nodf", "C_random_sbm"),
  contrast("modularity", "A_observed_neutral"),
  contrast("modularity", "C_random_sbm")
  ) |>
  left_join(as_tibble(analysis$site_info) |>
              select(site, elevation), by = "site")
comparison |>
  arrange(elevation, site, metric, contrast) |>
  knitr::kable(digits = 2, caption = "Mean contrasts and central 95%
simulation intervals.")
```

Mean contrasts and central 95% simulation intervals.

site	metric	contrast	mean	lower	upper	elevation
Bombuscaro	modularity	B - A	-0.05	-0.07	-0.02	1000
Bombuscaro	modularity	B - C	-0.12	-0.18	-0.04	1000
Bombuscaro	nodf	B - A	13.01	9.27	17.50	1000
Bombuscaro	nodf	B - C	-1.57	-13.41	11.16	1000
Copalinga	modularity	B - A	-0.05	-0.08	-0.02	1000
Copalinga	modularity	B - C	-0.12	-0.20	-0.04	1000
Copalinga	nodf	B - A	15.89	11.29	19.96	1000
Copalinga	nodf	B - C	-0.51	-10.48	13.78	1000
ECSF	modularity	B - A	-0.06	-0.12	0.00	2000
ECSF	modularity	B - C	-0.04	-0.14	0.06	2000
ECSF	nodf	B - A	4.63	1.04	8.73	2000
ECSF	nodf	B - C	-5.28	-14.33	4.44	2000
Finca	modularity	B - A	-0.05	-0.12	0.00	2000
Finca	modularity	B - C	-0.03	-0.12	0.06	2000
Finca	nodf	B - A	4.19	0.53	8.09	2000
Finca	nodf	B - C	-5.36	-17.94	3.52	2000
Bellavista	modularity	B - A	0.06	-0.02	0.15	3000
Bellavista	modularity	B - C	0.11	-0.02	0.22	3000
Bellavista	nodf	B - A	-1.03	-4.68	2.77	3000

site	metric	contrast	mean	lower	upper	elevation
Bellavista	nodf	B - C	-5.32	-12.98	2.30	3000
Cajanuma	modularity	B - A	-0.02	-0.13	0.11	3000
Cajanuma	modularity	B - C	0.05	-0.11	0.21	3000
Cajanuma	nodf	B - A	-1.79	-8.59	5.33	3000
Cajanuma	nodf	B - C	-8.15	-16.67	1.22	3000

The contrasts describe differences among simulated ensembles. The observed networks provide another reference: for example, a site may be more nested than all three ensemble means even when B − A is positive. The following table keeps observed structure and the three simulated means side by side. These means summarize 49 realizations, not 49 field networks.

Here A, B and C summarize networks sampled after MetaWeave's `infer_network()` step. The observed column comes from field records; MetaWeave does not estimate that column.

```
# Place observed metrics beside each ensemble's mean prediction.
ensemble_table <- map_dfr(c("nodf", "modularity"), function(metric) {
  tibble(
    site = analysis$site_info$site,
    elevation_m = analysis$site_info$elevation,
    metric = metric,
    observed = analysis$observed[, metric],
    A_neutral = rowMeans(analysis$draws[, , "A_observed_neutral", metric]),
    B_regional_SBM = rowMeans(analysis$draws[, , "B_observed_sbm", metric]),
    C_changed_species = rowMeans(analysis$draws[, , "C_random_sbm", metric])
  )
})
ensemble_table |>
  mutate(across(c(observed, A_neutral, B_regional_SBM,
                  C_changed_species), ~ round(.x, 2))) |>
  arrange(elevation_m, site, metric) |>
  knitr::kable(digits = 2,
               col.names = c("Site", "Elevation (m)", "Metric", "Observed",
"A", "B", "C"),
               caption = "Observed network metrics and ensemble means.")
```

Observed network metrics and ensemble means.

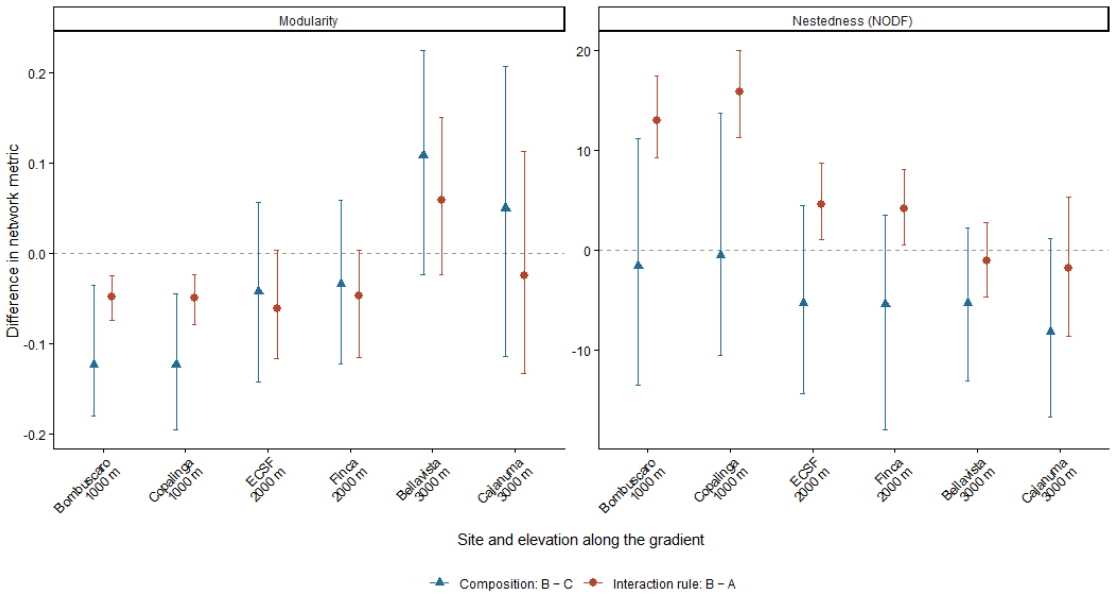
Site	Elevation (m)	Metric	Observed	A	B	C
Bombuscaro	1000	modularity	0.31	0.31	0.31	0.31
Bombuscaro	1000	nodf	41.11	41.11	41.11	41.11
Copalinga	1000	modularity	0.33	0.33	0.33	0.33
Copalinga	1000	nodf	58.37	58.37	58.37	58.37

Site	Elevation (m)	Metric	Observed	A	B	C
ECSF	2000	modularity	0.50	0.50	0.50	0.50
ECSF	2000	nodf	32.16	32.16	32.16	32.16
Finca	2000	modularity	0.49	0.49	0.49	0.49
Finca	2000	nodf	34.78	34.78	34.78	34.78
Bellavista	3000	modularity	0.63	0.63	0.63	0.63
Bellavista	3000	nodf	13.12	13.12	13.12	13.12
Cajanuma	3000	modularity	0.57	0.57	0.57	0.57
Cajanuma	3000	nodf	23.11	23.11	23.11	23.11

The figure plots the contrasts derived above, ordering the sites by the elevation supplied with their observed records. It visualizes the consequences of applying MetaWeave's regional rule to different local assemblages and link-allocation scenarios; elevation itself is not a predictor in `block_model()` or `infer_network()`.

```
# Keep site labels and both contrasts aligned along the elevation gradient.
site_labels <- site_table |>
  mutate(label = paste0(site, "\n", elevation, " m"))
plot_data <- comparison |>
  left_join(site_labels |> select(site, label), by = "site") |>
  mutate(label = factor(label, levels = site_labels$label),
         metric = recode(metric, nodf = "Nestedness (NODF)",
                         modularity = "Modularity"),
         contrast = recode(contrast, `B - A` = "Interaction rule: B − A",
                           `B - C` = "Composition: B − C"))

ggplot(plot_data, aes(x = label, y = mean, colour = contrast,
                      shape = contrast, group = contrast)) +
  geom_hline(yintercept = 0, linetype = 2, colour = "grey60") +
  geom_errorbar(aes(ymin = lower, ymax = upper), width = 0.12,
                position = position_dodge(width = 0.45)) +
  geom_point(size = 2.4, position = position_dodge(width = 0.45)) +
  facet_wrap(~ metric, scales = "free_y", nrow = 1) +
  scale_colour_manual(values = c("Interaction rule: B − A" = "#B2472F",
                                 "Composition: B − C" = "#246B91")) +
  scale_shape_manual(values = c("Interaction rule: B − A" = 16,
                                "Composition: B − C" = 17)) +
  labs(x = "Site and elevation along the gradient",
       y = "Difference in network metric", colour = NULL, shape = NULL) +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1),
        legend.position = "bottom")
```



Network assembly contrasts along the Ecuador elevation gradient. Sites are ordered from 1,000 to 3,000 m, with two sites per elevation. Circles show B − A (regional interaction rule); triangles show B − C (species composition). Points are means and vertical bars are central 95% intervals across 49 paired draws. The dashed line is zero difference.

At the two 1,000-m sites, regional interaction weighting increased nestedness by about 13 and 16 NODF units relative to uniform allocation, and reduced modularity by about 0.05. The nestedness difference was smaller at 2,000 m and its interval crossed zero at both 3,000-m sites. The B − C nestedness intervals crossed zero at all six sites. This is a pattern across these six networks, not a fitted elevation effect or a test of a general altitudinal trend.

An interval crossing zero means these 49 simulations do not clearly separate the two scenarios for that metric and site. The intervals describe variation among simulated networks, not uncertainty across Ecuadorian sites. The observed network is a separate reference; the contrasts isolate modeling choices rather than proving that any process caused its structure. The regional-pool randomization does not model unequal dispersal, habitat filtering, or species' abundance.