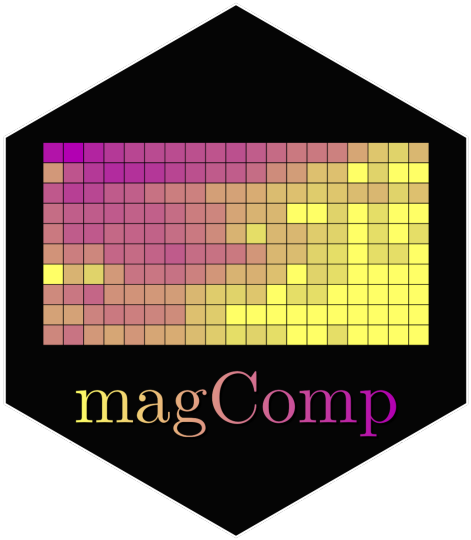


Methods and Implementation Details

Andrew O. Finley

Updated November 4, 2026

Contents

1 Motivation and overview **1**

2 Data structure and notation **1**

3 Fixed-area count model **2**

3.1 Two-category conditional composition . . . 3

3.2 Fixed-area support and nested plots . . . 4

4 Point-sampling basal area **4**

5 Model terms **5**

6 Factor identification **6**

7 Prior distributions **6**

7.1 Regression coefficients and ordered-category smoothing . . . 6

7.2 Factor loadings . . . 7

7.3 Size and latent process priors . . . 7

7.4 Spatial and space-time covariance families . . . 8

7.5 Covariance priors and settings . . . 8

8 Augmentation and sampler notation **9**

8.1 Magnitude augmentation . . . 9

8.2 Composition augmentation . . . 10

8.3 Indices, offsets, and spatial precision . . . 10

8.4 Normal draws using Cholesky solves . . . 11

9 Sampler algorithm **11**

9.1 Covariance-parameter Metropolis step . . . 14

10 Numerical implementation **15**

10.1 Spatial covariance calculations and cost . . . 15

10.2 Augmentation generators and Metropolis adaptation . . . 15

10.3 Data preparation, aggregation, and initialization . . . 16

10.4 Connection to the native implementation . . . 16

11 Prediction and aggregation **16**

11.1 Fitted values and prediction outputs . . . 17

11.2 Standalone components 19
11.3 Stand and forest aggregation 20
11.4 Cross-validation 21

12 Scope of the implementation **21**

13 Software workflow **22**
13.1 Mathematical notation and software names 22
13.2 Preparing settings and fitting models 23

14 Diagnostics **25**
14.1 Posterior predictive checks 25
14.2 Combined-coefficient diagnostics 25

References **26**

1 Motivation and overview

The `magComp` package fits Bayesian magnitude-composition models for multivariate quantities observed over sampling units, spatial locations, or prediction cells. The motivating application is forest inventory stand-table prediction, where total abundance or another scalar forest attribute must be allocated among diameter classes. The same model structure applies to other settings where a nonnegative total and a category composition are jointly modeled.

For sampling unit $i = 1, \ldots, n$ and category $j = 1, \ldots, J$, the mean response is decomposed as

$$\mu_{ij} = M_i p_{ij}, \qquad p_{ij} \geq 0, \qquad \sum_{j=1}^{J} p_{ij} = 1. \tag{1}$$

The scalar M_i is the magnitude component. The implemented forest inventory models use stem density for fixed-area plots and basal-area density for horizontal point sampling. The vector $\boldsymbol{p}_i = (p_{i1}, \ldots, p_{iJ})^\top$ is the composition component, which allocates the magnitude among DBH classes or other categories.

The decomposition in Equation 1 separates two inferential tasks. The magnitude model estimates the amount of material at a location or in a sampling unit. The composition model estimates how that material is divided among categories. This separation is useful when covariates explain total amount differently than they explain category allocation, and when prediction requires aggregation over many small spatial supports.

Users specify separate formulas for the magnitude and composition components. Each component can include fixed effects, unstructured random effects, spatial effects, and varying-coefficient terms appropriate to that component. Setting `composition = NULL` fits magnitude alone; setting `magnitude = NULL` fits composition alone. An omitted formula still defaults to an intercept, so disabling a component requires explicit `NULL`. At least one component must be fitted.

2 Data structure and notation

The primary data structure is a tree list or event list with a sampling-unit identifier. Let $\boldsymbol{s}_i$ denote the spatial location or centroid for sampling unit i, and let $g(i)$ denote an optional group, stand, or stratum label. Tree-level records supply category membership, such as species or DBH class, and the unit's sampling information and covariates. Aggregated integer counts can be supplied through `tree_count`. A magnitude-only analysis may omit categories and supply total counts per unit; when categories are present, their counts are summed. A composition-only analysis requires category counts but needs no area or BAF for fitting. An empty sampling unit must still have a record with `tree_count = 0` and valid category and unit metadata; a unit absent from the input has no likelihood contribution.

For fixed-area stem counts, y_{ij} is the integer number of trees observed in unit i and category j. The observed total is

$$N_i = \sum_{j=1}^{J} y_{ij}. \tag{2}$$

The model notation uses $\boldsymbol{z}_i$ for magnitude predictors and $\boldsymbol{x}_i$ for composition predictors. The two predictor sets may be identical, overlap, or be disjoint. This distinction keeps the scientific

interpretation explicit: a covariate can influence the total amount, the allocation among categories, or both.

Magnitude regression coefficients are $\boldsymbol{\beta}$; composition intercepts and slopes are α_{0j} and $\boldsymbol{\alpha}_j$. Magnitude iid and spatial effects are δ_g and $v(\boldsymbol{s})$. Composition iid factors $\boldsymbol{u}_h$ have loadings $\boldsymbol{\gamma}_j$, whereas spatial factors $\boldsymbol{w}(\boldsymbol{s})$ have loadings $\boldsymbol{\lambda}_j$. The grouping maps $g(i)$ and $h(i)$ may differ. Additional terms and allocation blocks receive corresponding term and block indices. Section 13.1 connects these symbols to the software names.

3 Fixed-area count model

For a fixed-area sampling unit with common tree-observation area a_i across all categories, the total observed count is modeled with a negative-binomial magnitude likelihood,

$$N_i \mid M_i, r_M \sim \mathrm{NB}(a_i M_i, r_M), \tag{3}$$

where M_i is stem density per unit area and $r_M > 0$ is the negative-binomial size parameter. The package uses the mean-size parameterization,

$$E(N_i \mid M_i, r_M) = a_i M_i, \qquad \mathrm{Var}(N_i \mid M_i, r_M) = a_i M_i + \frac{(a_i M_i)^2}{r_M}. \tag{4}$$

The Poisson model is the limiting case as $r_M \to \infty$. The package uses a finite size for fitting and prediction; its prior and fixed-size option are described in Section 7.3.

The log-density linear predictor is

$$\log M_i = \boldsymbol{z}_i^\top \boldsymbol{\beta} + \delta_{g(i)} + v(\boldsymbol{s}_i) + \sum_{q=1}^{p_M} z_{iq} v_q(\boldsymbol{s}_i). \tag{5}$$

The term $\boldsymbol{z}_i^\top \boldsymbol{\beta}$ contains fixed effects, including an intercept. The optional term $\delta_{g(i)}$ is an iid group-level random effect. The optional term $v(\boldsymbol{s}_i)$ is a spatial random intercept. The optional terms $v_q(\boldsymbol{s}_i)$ are spatially varying coefficients.

Conditional on N_i, the observed class-count vector follows a multinomial allocation model,

$$(y_{i1}, \ldots, y_{iJ}) \mid N_i, \boldsymbol{p}_i \sim \mathrm{Multinomial}(N_i, \boldsymbol{p}_i). \tag{6}$$

The multinomial is represented through sequential binomials. Define the number remaining before category j as

$$R_{ij} = N_i - \sum_{h<j} y_{ih}, \qquad j = 1, \ldots, J-1. \tag{7}$$

Then

$$y_{ij} \mid R_{ij}, \pi_{ij} \sim \mathrm{Binomial}(R_{ij}, \pi_{ij}), \qquad j = 1, \ldots, J-1, \tag{8}$$

and the final class receives the remaining count, $y_{iJ} = R_{i,J-1} - y_{i,J-1}$. The continuation-ratio probabilities π_{ij} are transformed to category probabilities by

$$p_{ij} = \pi_{ij} \prod_{h<j} (1 - \pi_{ih}), \qquad j = 1, \ldots, J-1, \tag{9}$$

with

$$p_{iJ} = \prod_{h<J} (1 - \pi_{ih}). \tag{10}$$

Equations 9 and 10 imply $\sum_{j=1}^{J} p_{ij} = 1$ by construction.

The composition linear predictor is

$$\psi_{ij} = g_C(\pi_{ij}) = \alpha_{0j} + \boldsymbol{x}_i^\top \boldsymbol{\alpha}_j + \boldsymbol{\gamma}_j^\top \boldsymbol{u}_{h(i)} + \sum_{q=1}^{p_C} x_{iq} \boldsymbol{\gamma}_{jq}^\top \boldsymbol{u}_{q,h(i)} + \boldsymbol{\lambda}_j^\top \boldsymbol{w}(\boldsymbol{s}_i) + \sum_{q=1}^{p_C} x_{iq} \boldsymbol{\lambda}_{jq}^\top \boldsymbol{w}_q(\boldsymbol{s}_i). \tag{11}$$

where g_C is either the logit link or the probit link selected by `control$composition_link`. The inverse link is therefore $g_C^{-1}(\psi) = \Phi(\psi)$ for the default probit model and $g_C^{-1}(\psi) = \{1 + \exp(-\psi)\}^{-1}$ for the optional logit model. The coefficients α_{0j} are category-specific intercepts. The vector $\boldsymbol{\alpha}_j$ contains category-specific fixed slopes. The optional term $\boldsymbol{\gamma}_j^\top \boldsymbol{u}_{h(i)}$ is an unstructured factor random effect. The first summation provides group-varying composition coefficients through unstructured factor scores. The optional term $\boldsymbol{\lambda}_j^\top \boldsymbol{w}(\boldsymbol{s}_i)$ is a spatial factor effect. The final summation provides spatially varying composition coefficients through factor processes.

3.1 Two-category conditional composition

The same magnitude-composition decomposition applies when each response cell is defined by two categorical variables. Let A have levels $a = 1, \ldots, J_A$, and let B have levels $b = 1, \ldots, J_B$. The cell mean is

$$\mu_{iab} = M_i p_{iab}, \qquad a = 1, \ldots, J_A, \qquad b = 1, \ldots, J_B. \tag{12}$$

This covers species-group-by-DBH stand tables, but the notation is generic: A and B can be any two categorical response dimensions.

The implemented two-category model uses a conditional factorization,

$$p_{iab} = q_{ia} r_{ib|a}, \tag{13}$$

where

$$q_{ia} = P(A_i = a), \qquad r_{ib|a} = P(B_i = b \mid A_i = a). \tag{14}$$

Users choose which variable is A by listing it first in `categories`; the second variable is modeled conditionally on the first. Either variable can be ordered or unordered. Reversing the category columns and the corresponding composition formulas fits the reverse conditioning, B then $A \mid B$. These are different model specifications and need not give equivalent fits under structured predictors or priors. For forest stand tables, a natural choice is a nominal leading category, such as species group, followed by an ordered child category, such as DBH class. For nominal leading categories, the continuation-ratio representation is conditional on the user-specified level order. This ordering does not affect the ability of a saturated model to represent a multinomial distribution, but fitted covariate effects and random-effect loadings can depend on the chosen order when structure or shrinkage is imposed. Sensitivity to nominal category ordering is therefore a useful diagnostic when the leading margin is unordered.

Define the probability vectors $\boldsymbol{q}_i = (q_{i1}, \ldots, q_{iJ_A})^\top$ and $\boldsymbol{r}_{i\cdot|a} = (r_{i1|a}, \ldots, r_{iJ_B|a})^\top$. Let $y_{ia+} = \sum_{b=1}^{J_B} y_{iab}$, $\boldsymbol{y}_{iA+} = (y_{i1+}, \ldots, y_{iJ_A+})^\top$, and $\boldsymbol{y}_{iaB} = (y_{ia1}, \ldots, y_{iaJ_B})^\top$. The likelihood is represented by two allocation blocks. The first allocates the total count among the categories of A,

$$\boldsymbol{y}_{iA+} \mid N_i, \boldsymbol{q}_i \sim \text{Multinomial}(N_i, \boldsymbol{q}_i), \tag{15}$$

and the second allocates each parent count among the categories of B,

$$\boldsymbol{y}_{iaB} \mid y_{ia+}, \boldsymbol{r}_{i\cdot|a} \sim \text{Multinomial}(y_{ia+}, \boldsymbol{r}_{i\cdot|a}), \qquad a = 1, \ldots, J_A. \tag{16}$$

If $y_{ia+} = 0$, the corresponding child allocation contributes no likelihood information for unit i and leading level a.

Each allocation block is fit using the sequential-binomial representation in Equations 7–10. Thus the leading allocation has $J_A - 1$ continuation-ratio linear predictors, and the child allocation has $J_A(J_B - 1)$ linear predictors when child-category parameters are allowed to vary by leading level. Fixed effects, smoothing, iid factor terms, GP factor terms, and NNGP factor terms are attached to allocation blocks.

The predictor vectors $\boldsymbol{x}_i^{(A)}$ and $\boldsymbol{x}_i^{(B|A)}$ can differ. Each parent has its own child coefficients and loading rows, while each child factor term shares its score vectors across parents. Thus species-specific size distributions can load differently on common group-level or spatial patterns.

Equations 3, 15, and 16 imply the parent-level and cell-level means

$$E(y_{ia+} \mid M_i, \boldsymbol{q}_i) = a_i M_i q_{ia} \tag{17}$$

and

$$E(y_{iab} \mid M_i, \boldsymbol{q}_i, \boldsymbol{r}_{i\cdot|a}) = a_i M_i q_{ia} r_{ib|a}. \tag{18}$$

The total count is modeled once, then allocated through the composition blocks.

3.2 Fixed-area support and nested plots

The count likelihood requires a common observation area a_i across categories within a sampling unit. It does not accept a unit-by-category matrix of areas. For nested regeneration, sapling, and overstory subplots, stem-count analyses must first define a common support. Tree expansion factors do not make counts from unequal category-specific areas compatible with this likelihood.

4 Point-sampling basal area

The point-sampling family models basal-area density using horizontal point-sampling tally counts.

For horizontal point sampling with basal area factor BAF_i, a tree with basal area BA_{it} has tree factor

$$\text{TF}_{it} = \frac{\text{BAF}_i}{\text{BA}_{it}}. \tag{19}$$

This expansion factor converts an observed in-tree to a per-unit-area tree count contribution. Other attributes expanded with these tree factors are outside the implemented count likelihood; see Section 12.

Basal area under horizontal point sampling is a special count-based case. The number of in-trees at point i, denoted m_i, satisfies

$$m_i \mid M_i^{BA}, r_M^{BA} \sim \text{NB}\left(\frac{M_i^{BA}}{\text{BAF}_i}, r_M^{BA}\right), \tag{20}$$

where M_i^{BA} is basal area per unit area. The magnitude component in Equation 5 can be used for $\log M_i^{BA}$, and the composition component allocates basal area among categories. The negative-binomial variance is $M_i^{BA}/\,\mathrm{BAF}_i + (M_i^{BA}/\,\mathrm{BAF}_i)^2/r_M^{BA}$, with the Poisson point-sampling tally model recovered as $r_M^{BA} \to \infty$.

This point-sampling model does not explicitly account for forest boundaries that truncate tree inclusion zones. Restricting tallies to forest trees near a boundary can reduce expected counts even when basal-area density is unchanged. The reduction depends on tree size and is not generally captured by a single area adjustment. Stand or forest inference from such points requires additional modeling of these boundary effects.

5 Model terms

Fixed effects use ordinary R formula syntax. Magnitude effects enter the log-density predictor; composition effects enter each allocation block's continuation-ratio predictors. The following terms specify random intercepts or factor contributions, with `s1` and `s2` denoting coordinate columns. Additional coordinate columns are allowed.

Component	Formula term	Contribution
Magnitude	`iid(group)`	Iid group effect
	`gp(s1, s2)`	Dense GP effect
	`nngp(s1, s2, m = 15)`	NNGP effect
Composition	`factor_iid(group, K = 2)`	K iid group factors
	`factor_gp(s1, s2, K = 2)`	K dense GP factors
	`factor_nngp(s1, s2, K = 2, m = 15)`	K NNGP factors

Prefixing any term with `x:` multiplies its contribution by covariate `x`, giving a group-varying or spatially varying coefficient. Include `x` separately in the fixed-effect formula when an average fixed slope is desired. For example, `x:iid(group)` gives an iid magnitude slope, and `x:factor_nngp(s1, s2, K = 2)` gives a spatial composition slope.

Multiple process terms may enter additively. For composition block C, let $\mathcal{T}_C^u$ and $\mathcal{T}_C^w$ denote its iid and spatial terms. Their combined contribution is

$$f_{ij}^{(C)} = \sum_{t \in \mathcal{T}_C^u} d_{it}^{(C)} \boldsymbol{\gamma}_{jt}^\top \boldsymbol{u}_{t,h_t(i)} + \sum_{t \in \mathcal{T}_C^w} d_{it}^{(C)} \boldsymbol{\lambda}_{jt}^\top \boldsymbol{w}_t(\boldsymbol{s}_i). \tag{21}$$

Here $d_{it}^{(C)} = 1$ for an intercept term and equals the covariate for a varying-coefficient term. Each term has its own rank, loading matrix, scores, anchors, and covariance parameters when applicable. Increasing rank permits richer residual dependence across categories. Section 6 defines the loading constraints. In conditional blocks, parent-specific loading rows act on shared scores as described in the two-category model.

6 Factor identification

For a K-factor spatial composition term, collect the loadings in the $L \times K$ matrix

$$\boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{\lambda}_1^\top \\ \vdots \\ \boldsymbol{\lambda}_L^\top \end{bmatrix}. \tag{22}$$

For a one-category allocation block, $L = J - 1$. For a conditional $B \mid A$ block with parent-specific child stick rows, $L = J_A(J_B - 1)$, with rows ordered by the parent-child stick labels described in Equation 16. Independent spatial factors have unit-scale parent GP correlation functions, with NNGP approximations where specified. Iid factors are independent standard normals; their loading matrix $\boldsymbol{\Gamma}$ uses the same constraints, with γ replacing λ below. The implementation selects K ordered anchor rows. If $a_1, \ldots, a_K$ are the anchor row indices, then

$$\lambda_{a_k,h} = 0 \quad (h > k), \qquad \lambda_{a_k,k} > 0. \tag{23}$$

After placing the anchor rows first in their specified order, their submatrix is lower triangular with positive diagonal. Non-anchor rows have unrestricted loadings. Anchor rows, their order, and factor rank are reported as part of the model specification. For an iid intercept factor term this gives within-group covariance $\boldsymbol{\Gamma}\boldsymbol{\Gamma}^\top$; for a dense-GP intercept factor term, the within-location covariance is $\boldsymbol{\Lambda}\boldsymbol{\Lambda}^\top$. The first anchor's factor contribution has variance equal to its squared diagonal loading. The NNGP version uses the corresponding unit-scale parent-GP approximation. The factor scale is fixed while the loadings determine the amplitude. The positive-diagonal constraints follow the residual-factor construction in Taylor-Rodriguez et al. (2019, Appendix B).

When anchors are omitted, the software ranks stick rows by the trial-weighted variance of empirical logits $\text{logit}\{(y + 0.5)/(R + 1)\}$, even for a probit fit, and fills any remaining slots in row order. Conditional blocks rank all parent-child stick rows.

7 Prior distributions

7.1 Regression coefficients and ordered-category smoothing

Fixed-effect coefficients in the magnitude component have normal priors,

$$\boldsymbol{\beta} \sim \text{N}(\mathbf{0}, \sigma_\beta^2 \boldsymbol{I}). \tag{24}$$

Here σ_β^2 is `magnitude$coefficient_variance` in the prior list. The interface does not accept arbitrary prior means or covariance matrices. Without smoothing, composition intercepts and slopes have independent normal priors,

$$\alpha_{0j} \sim \text{N}(0, \sigma_{\alpha,0}^2), \qquad \boldsymbol{\alpha}_j \sim \text{N}(\mathbf{0}, \sigma_\alpha^2 \boldsymbol{I}). \tag{25}$$

The software settings `intercept_variance` and `slope_variance` specify $\sigma_{\alpha,0}^2$ and σ_α^2, respectively. For an ordered block, let $\boldsymbol{\theta}_q = (\theta_{q1}, \ldots, \theta_{q,J-1})^\top$ collect its intercepts ($q = 0$, $\theta_{0j} = \alpha_{0j}$) or one predictor's slopes ($q > 0$, θ_{qj} is element q of $\boldsymbol{\alpha}_j$). For indices within the coefficient vector, the difference matrices act as

$$(\boldsymbol{D}_1\boldsymbol{\theta}_q)_j = \theta_{q,j+1} - \theta_{qj}, \qquad (\boldsymbol{D}_2\boldsymbol{\theta}_q)_j = \theta_{q,j+2} - 2\theta_{q,j+1} + \theta_{qj}.$$

RW1 penalizes adjacent differences; RW2 penalizes changes in adjacent slopes. The complete proper prior is

$$\boldsymbol{\theta}_q \sim \mathrm{N}(\mathbf{0}, \boldsymbol{P}_q^{-1}), \qquad \boldsymbol{P}_q = \boldsymbol{V}_q^{-1} + \tau_q^{-2} \boldsymbol{D}_{d_q}^\top \boldsymbol{D}_{d_q}, \tag{26}$$

where d_q is the selected difference order, $\boldsymbol{V}_0 = \sigma_{\alpha,0}^2 \boldsymbol{I}$, and $\boldsymbol{V}_q = \sigma_\alpha^2 \boldsymbol{I}$ for slopes. Omit the difference term when smoothing is disabled. The normal base precision makes the prior proper; the difference penalty alone would be singular. Smoothing variances are fixed hyperparameters: `intercept_smoothing_variance` specifies τ_0^2, and `slope_smoothing_variance` supplies a common value for the slope vectors within a block.

Child coefficient vectors are smoothed separately within each parent, so their joint prior precision is block diagonal over parents. No difference penalty crosses a parent boundary. Smoothing acts on continuation-ratio coefficients; the nonlinear transforms in Equations 9 and 10 need not produce smooth category proportions.

7.2 Factor loadings

Estimated off-diagonal iid and spatial loading elements have priors $\gamma_{jk} \sim \mathrm{N}(0, c_\gamma)$ and $\lambda_{jk} \sim \mathrm{N}(0, c_\lambda)$, respectively. Each term's `loading_variance` specifies its corresponding c_γ or c_λ, with block defaults and per-term overrides. Each anchor-diagonal loading has the corresponding normal prior restricted to $(0, \infty)$. The setting is the normal variance before truncation, so a value c gives half-normal scale $\sqrt{c}$. The default is one.

7.3 Size and latent process priors

When the magnitude negative-binomial size is estimated, it has a Gamma prior,

$$r_M \sim \mathrm{Gamma}(a_r, b_r), \tag{27}$$

using the shape-rate convention. The automatic setting `priors$magnitude$size = "auto"` estimates size with $(a_r, b_r) = (1, 0.01)$ when the magnitude formula has no iid or spatial terms, and otherwise fixes size at `control$nb_size` (default 1000). This policy applies to fixed-area and point-sampling models. Composition factor terms do not restrict size estimation. Its shape and rate can be changed through `magCompPriors()`. An explicit `size = NULL` fixes size. Joint estimation of size and magnitude random-effect terms is not supported.

For magnitude iid term t, including an intercept or varying slope,

$$\delta_{gt} \mid \sigma_{\delta,t}^2 \sim \mathrm{N}(0, \sigma_{\delta,t}^2), \qquad \sigma_{\delta,t}^2 \sim \mathrm{IG}(a_\delta, b_\delta). \tag{28}$$

Each term has a separate variance, with default inverse-gamma shape and scale (a_δ, b_δ) supplied by `magnitude$iid_variance`, defaulting to $(2, 1)$. A named magnitude term can override these hyperparameters. For a spatial magnitude random effect,

$$\boldsymbol{v} \mid \sigma_v^2, \phi_v \sim \mathrm{N}\{\mathbf{0}, \sigma_v^2 \boldsymbol{R}(\phi_v)\}, \tag{29}$$

where $\boldsymbol{v} = (v(\boldsymbol{s}_1), \ldots, v(\boldsymbol{s}_n))^\top$ and $\boldsymbol{R}(\phi_v)$ is a spatial correlation matrix or its NNGP approximation. For each magnitude GP or NNGP term, including a varying slope, the variance prior is

$$\sigma_v^2 \sim \mathrm{IG}(a_\sigma, b_\sigma), \tag{30}$$

which gives a Gibbs update conditional on $\boldsymbol{v}$ and ϕ_v. Here $\mathrm{IG}(a, b)$ has density proportional to $x^{-a-1}\exp(-b/x)$, and its shape and scale are supplied by `magnitude$spatial_variance`, with named per-term overrides. For a composition spatial factor term, the sampler uses independent unit-variance parent GP priors for the factor columns,

$$\boldsymbol{w}_k \mid \phi_{w,k} \sim \mathrm{N}\{\boldsymbol{0}, \boldsymbol{R}(\phi_{w,k})\}, \qquad k = 1, \ldots, K. \tag{31}$$

Here $\boldsymbol{w}_k = (w_k(\boldsymbol{s}_1), \ldots, w_k(\boldsymbol{s}_n))^\top$ contains factor k over the observed locations. For `factor_nngp()`, this full spatial prior is replaced by the NNGP conditional approximation.

7.4 Spatial and space-time covariance families

The exponential correlation is the default. The same covariance registry supports dense GP and NNGP terms in magnitude, leading allocation, and conditional allocation models, including varying coefficients. Write $h = \|\boldsymbol{s} - \boldsymbol{s}'\|$, $u = |t - t'|$, and let d_s be the number of spatial coordinates. The available unit-variance correlations are

$$\rho_{\mathrm{exp}}(h; \phi) = \exp(-\phi h), \tag{32}$$

$$\rho_{\mathrm{Mat}}(h; \phi, \nu) = \frac{(\phi h)^\nu K_\nu(\phi h)}{2^{\nu-1}\Gamma(\nu)}, \tag{33}$$

$$\rho_{\mathrm{sep}}(h, u; \phi, \lambda) = \exp(-\phi h - \lambda u), \tag{34}$$

$$\rho_{\mathrm{mix}}(h, u) = \alpha \exp(-\phi_1 h - \lambda_1 u) + (1 - \alpha)\exp(-\phi_2 h - \lambda_2 u), \tag{35}$$

$$\rho_{\mathrm{G}}(h, u) = T^{-\delta - d_s/2}\exp\{-ch^{2\gamma}/T^{\beta\gamma}\}, \qquad T = 1 + au^{2\alpha}. \tag{36}$$

The Matérn correlation is one at zero distance and reduces to the exponential at $\nu = 1/2$. For the Gneiting (2002) family, $a, c > 0$, $0 < \alpha, \gamma \leq 1$, $0 \leq \beta \leq 1$, and $\delta \geq 0$. All decay parameters and Matérn smoothness are positive; the mixture weight lies in $[0, 1]$. Magnitude variance remains a separate parameter, while composition factor correlations have unit marginal variance.

Use `cov_model = "exp"`, `"matern"`, `"sep_exp"`, `"multi_res_sep_exp"`, or `"gneiting"` in a process term. For a space-time family the final coordinate is numeric time, for example `nngp(easting, northing, year, cov_model = "sep_exp")`. Actual time gaps enter the covariance. NNGP neighbor search uses $\{h^2 + (s_{\mathrm{search}}u)^2\}^{1/2}$, where the positive `st_scale` setting defaults to one. This setting affects the directed neighbor graph only; correlations always use the original spatial and temporal lags. The graph is fixed during sampling, and prediction inherits its search scale. Different sampling units must have distinct full coordinate tuples; space-time models therefore permit revisits at different dates, but a spatial-only term cannot represent those revisits as separate latent nodes.

7.5 Covariance priors and settings

Each magnitude process term and each composition factor has its own covariance parameter vector, denoted $\boldsymbol{\vartheta}$ below. Every estimated entry has an independent bounded uniform prior. Priors, starts, and initial proposal standard deviations share a term's named `covariance` list. For example, `phi = uniform(0.1, 5)` specifies prior bounds, `phi = 1` a starting value, and `phi = fixed(1)` a fixed value in the starting list. Factor-specific named overrides are supported. `magCompCovModels()` lists parameter names and domains; `magCompSettings()` supplies resolved templates.

For automatic spatial bounds let D be maximum spatial separation and $r_0 = \frac{1}{2}\,\mathrm{median}_i\, d_i^{\mathrm{NN}}$, using distinct spatial sites. For numerical scaling, replace r_0 by at least $\sqrt{\epsilon_{\mathrm{machine}}}D$. Defaults for `exp`,

`matern`, and `sep_exp` are

$$\phi \sim \text{Unif}\{3/(2D), 3/\min(r_0, D/2)\}. \tag{37}$$

The initial value is $6/D$, replaced by the support midpoint if needed. For the mixture, $\phi_1 \sim \text{Unif}(3/D, 30/D)$ and $\phi_2 \sim \text{Unif}\{30/D, 3/\min(r_0, 0.05D)\}$, so component 1 describes longer spatial dependence. The mixing weight has prior $\text{Unif}(0, 1)$. User-specified supports replace these defaults; overlapping component supports can permit label switching.

If T_0 is the time span and Δ_0 the smallest positive time gap, temporal exponential decays have prior $\text{Unif}\{3/(2T_0), 6/\Delta_0\}$. This calibrates the upper extent of temporal dependence to twice the observed time domain. Matérn smoothness defaults to $\nu \sim \text{Unif}(0.05, 2)$. Gneiting defaults fix $\alpha = 1, \gamma = 1/2, \delta = 0$, estimate $\beta \sim \text{Unif}(0, 1)$, and use the spatial ϕ support for c. Its default support for a is $[C/(2T_0)^2, C/(\Delta_0/2)^2]$, with $C = 0.05^{-2/d_s} - 1$, calibrated at those default shape values. A prior or ordinary numeric start can release a default-fixed shape parameter; δ then defaults to $\text{Unif}(0, 2)$. Other automatic starts use support midpoints. If a coordinate domain has no positive span, its corresponding support or fixed value must be supplied. These are uniform priors on covariance parameters, not on effective ranges.

8 Augmentation and sampler notation

The sampler retains magnitude random effects and composition factor scores as latent variables. Conditional on augmentation variables, their likelihood contributions are normal kernels. Section 8.4 defines the dense and sparse Cholesky draws used in the sampler.

8.1 Magnitude augmentation

Write e_i for the observation multiplier: $e_i = a_i$ for fixed-area plots and $e_i = 1/\text{BAF}_i$ for point sampling. In the sampler formulas, N_i denotes the observed total tally, M_i the corresponding magnitude, and r_M the size. For point sampling these stand for m_i, M_i^{BA}, and r_M^{BA}, respectively. The count mean is $e_i M_i$. Define

$$\eta_i^{(M)} = \log M_i, \qquad o_i^{(M)} = \log e_i - \log r_M, \qquad \psi_i^{(M)} = \eta_i^{(M)} + o_i^{(M)}. \tag{38}$$

For fixed r_M, the negative-binomial likelihood kernel is

$$\frac{\exp\{N_i \psi_i^{(M)}\}}{\{1 + \exp(\psi_i^{(M)})\}^{N_i + r_M}}, \qquad \kappa_i^{(M)} = \frac{N_i - r_M}{2}. \tag{39}$$

Pólya–Gamma augmentation (Polson et al., 2013) gives

$$\omega_i^{(M)} \mid \cdot \sim \text{PG}(N_i + r_M, \psi_i^{(M)}), \qquad \mathcal{K}_i^{(M)}(\psi_i^{(M)}) \propto \exp\left\{\kappa_i^{(M)} \psi_i^{(M)} - \frac{\omega_i^{(M)}}{2}(\psi_i^{(M)})^2\right\}. \tag{40}$$

Here $\mathcal{K}_i^{(M)}$ is the augmented joint-density kernel as a function of the predictor, holding the count and augmentation draw fixed.

8.2 Composition augmentation

In each allocation block, use y_{ij}, R_{ij}, and ψ_{ij} for the binomial count, remaining count, and continuation-ratio predictor. Define weights ω_{ij} and canonical linear terms h_{ij} so that the augmented kernel is

$$\exp\left\{h_{ij}\psi_{ij} - \frac{\omega_{ij}}{2}\psi_{ij}^2\right\}. \tag{41}$$

For the optional logit link,

$$\omega_{ij} \mid \cdot \sim \mathrm{PG}(R_{ij}, \psi_{ij}), \qquad h_{ij} = \kappa_{ij}^{(C)} = y_{ij} - R_{ij}/2. \tag{42}$$

For the default probit link, introduce one latent normal for each of the R_{ij} observations available at the allocation step. Conditional on the count, draw y_{ij} independent $\mathrm{N}(\psi_{ij}, 1)$ variables restricted to $(0, \infty)$ and $R_{ij} - y_{ij}$ restricted to $(-\infty, 0]$. This is the grouped-count version of latent-normal probit augmentation (Albert and Chib, 1995; Rouder and Lu, 2005). With latent values ζ_{ijt}, retain their sum,

$$S_{ij} = \sum_{t=1}^{R_{ij}} \zeta_{ijt}, \qquad h_{ij} = S_{ij}, \qquad \omega_{ij} = R_{ij}. \tag{43}$$

The same normal coefficient, loading, and score updates then apply to both links. When $R_{ij} = 0$, set $h_{ij} = \omega_{ij} = 0$ and draw no augmentation variables.

8.3 Indices, offsets, and spatial precision

The following algorithm applies to every fitted allocation block. Within a block, j indexes a coefficient/loading row. For a leading block these are the $J_A - 1$ continuation-ratio rows; for a conditional block they are the $J_A(J_B - 1)$ parent–child rows. Let $\mathcal{I}_j$ contain the data rows governed by coefficient row j. A child data row represents a sampling unit and parent context (i, a); its group and spatial indices refer back to the original sampling unit. Thus shared factor scores are updated using contributions from all their associated parent contexts.

Write X_{iq} for the composition design entry, with $X_{i0} = 1$, and $\boldsymbol{\theta}_q$ for the coefficient vector over all L coefficient rows: $\theta_{0j} = \alpha_{0j}$, with $q > 0$ indexing slopes. The prior precision $\boldsymbol{P}_q$ is defined in Equation 26; for child blocks it contains one diagonal block per parent. Let d_{it} denote a process term's covariate multiplier: one for an intercept term and the specified covariate for a varying slope. Offsets below always use the latest values of every component except the one being updated.

For a spatial or space-time term with covariance parameters $\boldsymbol{\vartheta}$, let $\boldsymbol{H}_{\boldsymbol{\vartheta}}$ denote its unit-scale prior precision, and let $d_{\boldsymbol{\vartheta}}$ denote the log determinant of its unit-scale covariance:

$$\begin{aligned} &\boldsymbol{H}_{\boldsymbol{\vartheta}} = \boldsymbol{R}_{\boldsymbol{\vartheta}}^{-1}, && d_{\boldsymbol{\vartheta}} = \log|\boldsymbol{R}_{\boldsymbol{\vartheta}}| && \text{(dense GP)}, \\ &\boldsymbol{H}_{\boldsymbol{\vartheta}} = (\boldsymbol{I} - \boldsymbol{B}_{\boldsymbol{\vartheta}})^\top \boldsymbol{F}_{\boldsymbol{\vartheta}}^{-1}(\boldsymbol{I} - \boldsymbol{B}_{\boldsymbol{\vartheta}}), && d_{\boldsymbol{\vartheta}} = \sum_i \log F_i(\boldsymbol{\vartheta}) && \text{(NNGP)}. \end{aligned} \tag{44}$$

NNGP locations are ordered by the first coordinate, with subsequent coordinates breaking ties. Let $\mathcal{N}_i$ be the at most m nearest earlier locations under the search distance in Section 7.4. For the chosen unit-scale correlation, solve

$$\boldsymbol{R}_{\mathcal{N}_i,\mathcal{N}_i}\boldsymbol{b}_i(\boldsymbol{\vartheta}) = \boldsymbol{R}_{\mathcal{N}_i,i}, \qquad F_i(\boldsymbol{\vartheta}) = 1 - \boldsymbol{R}_{i,\mathcal{N}_i}\boldsymbol{b}_i(\boldsymbol{\vartheta}). \tag{45}$$

Row i of $\boldsymbol{B}_{\vartheta}$ contains $\boldsymbol{b}_i^\top$ in its neighbor columns and zeros elsewhere; $\boldsymbol{F}_{\vartheta} = \mathrm{diag}(F_i)$. The first location has no neighbors and $F_1 = 1$. In these equations vectors are in spatial order. This conditional construction follows Finley et al. (2019). For a spatial vector $\boldsymbol{v}$ with variance s^2, its log prior, up to a constant, is

$$G(\boldsymbol{v}; \vartheta, s^2) = -\tfrac{1}{2}\{n \log s^2 + d_{\vartheta} + s^{-2}\boldsymbol{v}^\top \boldsymbol{H}_{\vartheta}\boldsymbol{v}\}. \tag{46}$$

Use the vector's own number of locations for n. Composition spatial factors have $s^2 = 1$.

8.4 Normal draws using Cholesky solves

Each unrestricted normal update below specifies a precision $\boldsymbol{Q}$ and linear term $\boldsymbol{b}$, defining the full conditional

$$\boldsymbol{\zeta} \mid \cdot \sim \mathrm{N}(\boldsymbol{Q}^{-1}\boldsymbol{b}, \boldsymbol{Q}^{-1}). \tag{47}$$

For Gibbs updates, $\mid \cdot$ denotes conditioning on the data and latest values of the other parameters and latent variables. The draw operations below compute this distribution using solves, without forming its covariance.

For a dense precision, $\mathcal{G}_D(\boldsymbol{Q}, \boldsymbol{b})$ factors $\boldsymbol{Q} = \boldsymbol{L}\boldsymbol{L}^\top$ and computes

$$\begin{gathered} \boldsymbol{L}\boldsymbol{a} = \boldsymbol{b}, \qquad \boldsymbol{L}^\top \boldsymbol{m} = \boldsymbol{a}, \qquad \boldsymbol{L}^\top \boldsymbol{e} = \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{I}), \\ \boldsymbol{\zeta} = \boldsymbol{m} + \boldsymbol{e}. \end{gathered} \tag{48}$$

LAPACK performs the Cholesky factorization and mean solves; a BLAS triangular solve supplies the random component. This operation is used for regression coefficients, smoothed coefficient vectors, iid factor-score vectors, and dense GP latent vectors. The coefficient draws use dense factorization even when the smoothing prior precision is banded.

For an NNGP latent vector, the posterior precision has the sparse form

$$\boldsymbol{Q}_{\mathrm{post}} = s^{-2}\boldsymbol{H}_{\vartheta} + \mathrm{diag}(D_i), \tag{49}$$

where D_i collects the augmented-likelihood weights at location i. It is assembled directly from the neighbor conditionals and likelihood contributions. For a sparse precision $\boldsymbol{Q}$, $\mathcal{G}_S(\boldsymbol{Q}, \boldsymbol{b})$ uses CHOLMOD's fill-reducing permutation $\boldsymbol{\Pi}$ and factorization $\boldsymbol{\Pi}\boldsymbol{Q}\boldsymbol{\Pi}^\top = \boldsymbol{L}\boldsymbol{L}^\top$ to compute

$$\begin{gathered} \boldsymbol{L}\boldsymbol{a} = \boldsymbol{\Pi}\boldsymbol{b}, \qquad \boldsymbol{L}^\top \boldsymbol{m}_{\mathrm{perm}} = \boldsymbol{a}, \qquad \boldsymbol{L}^\top \boldsymbol{e} = \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{I}), \\ \boldsymbol{\zeta} = \boldsymbol{\Pi}^\top(\boldsymbol{m}_{\mathrm{perm}} + \boldsymbol{e}). \end{gathered} \tag{50}$$

CHOLMOD's sparse solves compute the mean and random component. Symbolic analysis is reused for matching sparsity patterns, while numerical factorization is repeated for each current precision. The CHOLMOD permutation is distinct from the spatial ordering used to define NNGP neighbors.

9 Sampler algorithm

Algorithm 1 gives one complete sweep in implementation order. It visits the components and terms included in the fitted model, using the priors in Section 7 and draw operations in Section 8.4.

Algorithm 1: Latent magnitude–composition posterior sampler

Setup. Construct counts, designs, priors, anchors, and starting values. Build spatial orderings and neighbor lists once, and initialize covariance quantities and sparse symbolic factorizations. For each iteration:

1. **Update negative-binomial size when estimated.** Given the current M_i, the marginal conditional density is

$$p(r_M \mid \cdot) \propto r_M^{a_r-1} e^{-b_r r_M} \prod_i \frac{\Gamma(N_i + r_M)}{\Gamma(r_M) N_i!} \left(\frac{r_M}{r_M + e_i M_i}\right)^{r_M} \left(\frac{e_i M_i}{r_M + e_i M_i}\right)^{N_i}. \tag{51}$$

   Propose

$$\log r_M^\star = \log r_M + s_r \epsilon, \qquad \epsilon \sim \mathrm{N}(0,1). \tag{52}$$

   Let $\mathcal{L}_M(r)$ be the log negative-binomial likelihood above. Accept with probability

$$\min\{1, \exp[H(r_M^\star) - H(r_M)]\},$$

   where

$$H(r) = \mathcal{L}_M(r) + (a_r - 1)\log r - b_r r + \log r. \tag{53}$$

   The final term accounts for proposing on the log scale. This step integrates out the previous magnitude augmentation variables; refresh them immediately in Step 2 before any other magnitude update.

2. **Draw magnitude augmentation variables and coefficients.** Using the updated size and current predictor, draw $\omega_i^{(M)}$ from Equation 40. Let $f_i^{(M)}$ be the sum of all magnitude random-effect contributions and let $\boldsymbol{Z}$ have rows $\boldsymbol{z}_i^\top$. Set $\boldsymbol{\Omega}_M = \mathrm{diag}(\omega_i^{(M)})$ and

$$\begin{aligned} \boldsymbol{Q}_\beta &= \sigma_\beta^{-2}\boldsymbol{I} + \boldsymbol{Z}^\top \boldsymbol{\Omega}_M \boldsymbol{Z}, \\ \boldsymbol{b}_\beta &= \boldsymbol{Z}^\top\{\boldsymbol{\kappa}_M - \boldsymbol{\Omega}_M(\boldsymbol{o}^{(M)} + \boldsymbol{f}^{(M)})\}, \\ \boldsymbol{\beta} &\leftarrow \mathcal{G}_D(\boldsymbol{Q}_\beta, \boldsymbol{b}_\beta). \end{aligned} \tag{54}$$

   Retain these augmentation weights throughout the magnitude-term updates.

3. **Update each magnitude process term in formula order.** For term t, define $o_{it}^{(-)}$ as the current magnitude predictor excluding that term, but including $\log e_i - \log r_M$.

   (a) For an iid term, update each group separately:

$$\begin{aligned} Q_{gt} &= \sigma_{\delta,t}^{-2} + \sum_{i:g_t(i)=g} \omega_i^{(M)} d_{it}^2, \\ b_{gt} &= \sum_{i:g_t(i)=g} d_{it}\{\kappa_i^{(M)} - \omega_i^{(M)} o_{it}^{(-)}\}, \\ \delta_{gt} \mid \cdot &\sim \mathrm{N}(b_{gt}/Q_{gt}, Q_{gt}^{-1}), \\ \sigma_{\delta,t}^2 \mid \cdot &\sim \mathrm{IG}\left(a_{\delta,t} + G_t/2,\ b_{\delta,t} + \tfrac{1}{2}\sum_g \delta_{gt}^2\right). \end{aligned} \tag{55}$$

   Here G_t is the number of groups. Sample the variance after all effects of that term have been updated.

(b) For a GP or NNGP term, jointly update its location vector:

$$\begin{aligned} \boldsymbol{Q}_t &= \sigma_{v,t}^{-2}\boldsymbol{H}_{\vartheta_{v,t}} + \operatorname{diag}(\omega_i^{(M)} d_{it}^2), \\ b_{ti} &= d_{it}\{\kappa_i^{(M)} - \omega_i^{(M)} o_{it}^{(-)}\}, \\ \boldsymbol{v}_t &\leftarrow \begin{cases} \mathcal{G}_D(\boldsymbol{Q}_t, \boldsymbol{b}_t), & \text{GP}, \\ \mathcal{G}_S(\boldsymbol{Q}_t, \boldsymbol{b}_t), & \text{NNGP}, \end{cases} \\ \sigma_{v,t}^2 \mid \cdot &\sim \text{IG}\left(a_{\sigma,t} + n/2,\ b_{\sigma,t} + \tfrac{1}{2}\boldsymbol{v}_t^\top \boldsymbol{H}_{\vartheta_{v,t}} \boldsymbol{v}_t\right). \end{aligned} \tag{56}$$

After drawing the vector and variance, update the estimated covariance parameters jointly using the bounded-logit Metropolis step in Equation 63.

4. Update each composition block, leading block before child block.

(a) Refresh (h_{ij}, ω_{ij}) using the selected link as in Section 8.

(b) Update the intercept vector first, then each slope vector in design-column order. With $o_{ij}^{(-q)} = \psi_{ij} - X_{iq}\theta_{qj}$, form

$$\begin{aligned} D_{qj} &= \sum_{i \in \mathcal{I}_j} \omega_{ij} X_{iq}^2, \qquad b_{qj} = \sum_{i \in \mathcal{I}_j} X_{iq}\{h_{ij} - \omega_{ij} o_{ij}^{(-q)}\}, \\ \boldsymbol{Q}_q &= \boldsymbol{P}_q + \operatorname{diag}(D_{qj}), \qquad \boldsymbol{\theta}_q \leftarrow \mathcal{G}_D(\boldsymbol{Q}_q, \boldsymbol{b}_q). \end{aligned} \tag{57}$$

The draw includes all parent-specific coefficient vectors in a child block.

(c) For each factor term in that block, perform Step 5 before proceeding to the next allocation block. Keep the same augmentation variables through its coefficient and factor updates.

5. Update a composition factor term. Suppress its term index below; d_i is its multiplier. For a spatial term use loadings λ_{jk} and scores w_{gk}; for an iid term replace these with γ_{jk} and u_{gk}. Here $g(i)$ maps each allocation data row to its location or group.

(a) Use a working parameterization in which anchor diagonals, like other estimated loadings, have unrestricted zero-mean normal priors. Keep the upper-anchor zeros. Update estimated loadings one at a time, traversing loading rows and then factor columns. Let $x_i = d_i w_{g(i),k}$ and $o_{ij}^{(-jk)} = \psi_{ij} - x_i\lambda_{jk}$. With prior variance c_λ, calculate

$$V_{jk} = \left(c_\lambda^{-1} + \sum_{i \in \mathcal{I}_j} \omega_{ij} x_i^2\right)^{-1}, \qquad m_{jk} = V_{jk} \sum_{i \in \mathcal{I}_j} x_i\{h_{ij} - \omega_{ij} o_{ij}^{(-jk)}\}. \tag{58}$$

Draw every estimated working loading from $\text{N}(m_{jk}, V_{jk})$. After completing the loading sweep, let $s_k = \operatorname{sign}(\lambda_{a_k,k})$, where a_k is the kth anchor row, and jointly transform

$$\lambda_{jk} \leftarrow s_k\lambda_{jk} \quad \text{for every } j, \qquad w_{gk} \leftarrow s_k w_{gk} \quad \text{for every } g. \tag{59}$$

All anchor diagonals are then positive. For iid terms, use c_γ, γ_{jk}, and u_{gk}.

The joint sign transformation preserves every loading–score product, the likelihood, and the centered Gaussian priors. Folding the unrestricted posterior onto positive anchors gives exactly the specified half-normal diagonal priors, while preserving the other normal priors

and fixed zeros. The Gaussian loading sweep is equivariant under these sign transformations, so following it by sign alignment preserves the identified posterior. This update allows coordinated changes in loadings and scores when an anchor loading is weakly determined. It does not change the factor model or its covariance parameters.

(b) For an iid term, update its K-vector of scores jointly within each group. Write $\boldsymbol{x}_{ij} = d_i \boldsymbol{\gamma}_j$ and let $o_{ij}^{(-t)}$ exclude the entire current factor term. Sum over valid allocation observations associated with group g:

$$\begin{aligned} \boldsymbol{Q}_g &= \boldsymbol{I}_K + \sum_{(i,j):g(i)=g} \omega_{ij} \boldsymbol{x}_{ij} \boldsymbol{x}_{ij}^\top, \\ \boldsymbol{b}_g &= \sum_{(i,j):g(i)=g} \boldsymbol{x}_{ij} \{h_{ij} - \omega_{ij} o_{ij}^{(-t)}\}, \\ \boldsymbol{u}_g &\leftarrow \mathcal{G}_D(\boldsymbol{Q}_g, \boldsymbol{b}_g). \end{aligned} \tag{60}$$

(c) For a spatial term, update one complete score column at a time. Let $x_{ij} = d_i \lambda_{jk}$ and $o_{ij}^{(-k)} = \psi_{ij} - x_{ij} w_{g(i),k}$. Accumulate by location:

$$\begin{aligned} D_{gk} &= \sum_{(i,j):g(i)=g} \omega_{ij} x_{ij}^2, \qquad b_{gk} = \sum_{(i,j):g(i)=g} x_{ij} \{h_{ij} - \omega_{ij} o_{ij}^{(-k)}\}, \\ \boldsymbol{Q}_k &= \boldsymbol{H}_{\boldsymbol{\zeta}_{w,k}} + \operatorname{diag}(D_{gk}), \\ \boldsymbol{w}_k &\leftarrow \begin{cases} \mathcal{G}_D(\boldsymbol{Q}_k, \boldsymbol{b}_k), & \text{GP}, \\ \mathcal{G}_S(\boldsymbol{Q}_k, \boldsymbol{b}_k), & \text{NNGP}. \end{cases} \end{aligned} \tag{61}$$

After updating all score columns, update the covariance parameters of each factor jointly by Equation 63, with $\boldsymbol{v} = \boldsymbol{w}_k$ and $s^2 = 1$.

6. **Adapt proposals and retain post-burn-in draws.** Covariance blocks use an initial scale search during burn-in followed by diminishing adaptation; the separate size update adapts only during burn-in (Section 10). After the full sweep, save the parameter state when $k > n_{\text{burn}}$ and $(k - n_{\text{burn}})$ is divisible by n_{thin}, where k is the iteration number. Repeat for n_{samples} total iterations. Fitted probabilities and expected densities are reconstructed from retained parameter states. Use ℓ to index these retained posterior draws.

9.1 Covariance-parameter Metropolis step

The Gaussian process log density $G(\boldsymbol{v}; \boldsymbol{\vartheta}, s^2)$ uses the correlation family in Section 7.4. Form one block of estimated covariance parameters per magnitude term and per composition factor. Let $\mathcal{I}$ index its estimated entries, with uniform supports (L_p, U_p). Transform them to an unconstrained vector and propose

$$\eta_p = \log \frac{\vartheta_p - L_p}{U_p - \vartheta_p}, \qquad \boldsymbol{\eta}^\star = \boldsymbol{\eta} + s_b \boldsymbol{L}_b \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathrm{N}(\boldsymbol{0}, \boldsymbol{I}), \qquad \vartheta_p^\star = L_p + (U_p - L_p) \operatorname{logit}^{-1}(\eta_p^\star), \tag{62}$$

where $\boldsymbol{L}_b \boldsymbol{L}_b^\top = \boldsymbol{\Sigma}_b$ is the proposal shape matrix. Rebuild covariance quantities at $\boldsymbol{\vartheta}^\star$ and accept with probability

$$\min \left[1, \exp \left\{ G(\boldsymbol{v}; \boldsymbol{\vartheta}^\star, s^2) - G(\boldsymbol{v}; \boldsymbol{\vartheta}, s^2) + \sum_{p \in \mathcal{I}} \log \frac{(\vartheta_p^\star - L_p)(U_p - \vartheta_p^\star)}{(\vartheta_p - L_p)(U_p - \vartheta_p)} \right\} \right]. \tag{63}$$

The sum is the inverse-transformation Jacobian ratio. The uniform prior constants cancel. Magnitude uses its current variance $s^2 = \sigma_v^2$; each composition factor uses $s^2 = 1$. This is conditional on the current latent process vector. A one-parameter block is the scalar special case.

10 Numerical implementation

10.1 Spatial covariance calculations and cost

The dense GP backend constructs $\boldsymbol{R}_{\boldsymbol{\vartheta}}$, uses Cholesky to compute its log determinant, and explicitly obtains the *prior* precision. This costs $O(n^3)$ time and $O(n^2)$ storage. The prior precision is cached until an accepted covariance proposal changes it. Dense posterior draws also require an $n \times n$ factorization.

For NNGP, Equation 45 uses Cholesky factorization and triangular solves on matrices of size at most $m \times m$. These conditional quantities cost $O(nm^3)$ to rebuild for new covariance parameters. Without forming $\boldsymbol{H}_{\boldsymbol{\vartheta}}$, compute

$$\boldsymbol{v}^\top \boldsymbol{H}_{\boldsymbol{\vartheta}} \boldsymbol{v} = \sum_i \frac{\{v_i - \boldsymbol{b}_i^\top \boldsymbol{v}_{\mathcal{N}_i}\}^2}{F_i}, \qquad d_{\boldsymbol{\vartheta}} = \sum_i \log F_i. \tag{64}$$

The quadratic form costs $O(nm)$; sparse precision assembly costs $O(nm^2)$. Sparse factorization cost depends on graph fill-in. The R front end constructs a dense distance matrix for spatial defaults and directly scans earlier locations for neighbors. Independent chains can run in separate R processes; within each chain the updates and neighbor solves run serially.

GP and NNGP calculations use the specified correlation matrices without diagonal additions or conditional-variance floors.

10.2 Augmentation generators and Metropolis adaptation

Probit augmentation uses truncated-normal rejection sampling, with an exponential envelope in the tail.

Magnitude and logit-composition augmentation call `BayesLogit`'s hybrid Pólya–Gamma generator. It uses numerical approximations, including saddlepoint and large-shape normal approximations and a truncated gamma series, to sample from the distributions defined by the augmentation.

Covariance blocks initially use diagonal proposal shape matrices whose standard deviations are the supplied `tuning` values (default 0.1), with multiplier $s = 1$. An initial search uses batches of complete sampler sweeps, ending when that block's acceptance is between 0.15 and 0.45 (0.30 and 0.60 for a scalar), or after 20 batches or the burn-in budget. Acceptance below 0.02, 0.10, or the target interval multiplies s by 0.1, 0.25, or 0.5, respectively; acceptance above 0.65 or the target interval multiplies it by 3 or 1.5. This search is skipped when there is no burn-in.

Subsequent batches use diminishing adaptation of both shape and scale, including during retained sampling. For adaptive batch b, let $\widehat{\boldsymbol{\Sigma}}_b$ be the empirical covariance of transformed chain states (including rejections), and $\hat{a}_b$ its acceptance fraction. Then

$$\log s \leftarrow \log s + g_b(\hat{a}_b - a_*), \qquad \boldsymbol{\Sigma} \leftarrow (1 - g_b)\boldsymbol{\Sigma} + g_b \widehat{\boldsymbol{\Sigma}}_b, \quad g_b = (b+1)^{-0.8}. \tag{65}$$

Defaults use batches of 25, with $a_* = 0.234$ for multivariate blocks and 0.44 for scalars. A batch of size one updates scale only. The scale multiplier is bounded to $[10^{-6}, 10^{6}]$. The proposal Cholesky factor is recomputed after shape adaptation; this does not alter the model covariance. Diminishing adaptation follows the adaptive-MCMC approach discussed by Roberts and Rosenthal (2009); convergence still requires assessment for the fitted model. `adapt = FALSE` disables both search and adaptation.

The separate negative-binomial size update retains scalar adaptation during burn-in only: at batch b, add or subtract $\min(0.01, b^{-1/2})$ from its log proposal scale according to whether acceptance exceeds its target (default 0.44). All adaptation histories are specific to their chain and covariance block or size parameter.

10.3 Data preparation, aggregation, and initialization

Different sampling units in a spatial term must have distinct coordinate vectors. Tree records and child parent-context rows within the same unit share its effects. Prediction at repeated coordinates is described in Section 11.

An allocation block with only an intercept and no factor terms aggregates counts and remaining counts by coefficient row before augmentation. This is valid because the combined observations have the same predictor. Without smoothing, the coefficients have independent scalar normal updates; with smoothing, they use the joint update in Step 4.

Default magnitude starts regress $\log\{(N_i + 0.5)/e_i\}$ on the fixed effects and project residuals onto the group or spatial terms. Composition starts transform empirical probabilities $(y_{ij} + 0.5)/(R_{ij} + 1)$ with the selected link, regress them on fixed effects, and initialize residual factors by singular value decomposition. Child-block regressions are separate within each parent, while factor scores remain shared across parents.

Iid factor terms are initialized before spatial terms, subtracting each term's contribution before initializing the next. Rotation, sign, and scale adjustments enforce the anchor constraints and normalize starting scores, transferring amplitude to the loadings. These adjustments apply only to initialization. Users can inspect and replace starts through the settings interface in Section 13.2.

10.4 Connection to the native implementation

Algorithm step	Main C++ routine(s) in `magcomp_sampler.cpp`
1	`updateMagnitudeNbSize`
2	`updateMagnitudeCoefficients`
3	`updateMagnitudeIidEffects`, `updateMagnitudeSpatialEffect`
4	`updateAllocationBlock`
5	`updateAllocationFactor`
6	`runSampler`, `CovarianceMetropolis::update`

11 Prediction and aggregation

Let ℓ index a retained posterior draw. Prediction at a new location or grid cell $\boldsymbol{s}_0$ proceeds draw by draw from the posterior, including all terms in the fitted magnitude and composition models. Observed group effects are reused, and one new Gaussian effect per unseen group, term, and

posterior draw is shared by every prediction row in that group. Magnitude iid draws use the corresponding term's posterior variance draw, and composition iid factors use identity covariance. Spatial coordinates matching fitted locations reuse their stored effects; repeated new coordinates share one spatial-effect draw per term and posterior draw. Coordinate matching is exact. Dense-GP predictions draw jointly from the conditional normal distribution. NNGP predictions order new coordinates by the same coordinate rule and choose neighbors from all observed locations and earlier prediction locations in the same call. For each NNGP term's prediction call, the ordering, neighbor identities, and required distances are computed once and reused across chains, draws, and factors. Conditional covariance calculations still use each draw's parameters, and new-location effects are sampled sequentially within each draw. Consequently the NNGP approximation depends on the requested prediction set; separate prediction batches omit cross-batch conditional links and do not reproduce a single joint call.

At each retained draw, evaluate the magnitude and composition predictors at the requested units. Exponentiating the magnitude predictor gives M_0; Equations 9 and 10, with the fitted inverse link, give p_{0j}. For two-category responses, reconstruct $p_{0ab} = q_{0a} r_{0b|a}$ using Equation 13. Expected category density and count are $M_0 p_{0j}$ and $a_0 M_0 p_{0j}$. At observed units the same calculation uses stored latent draws and original areas. Form these quantities within each draw before summarizing.

11.1 Fitted values and prediction outputs

The location argument and the target define four operations for the full magnitude–composition model. Each uses the full fitted model:

1. **Observed units, expected quantities.** `fitted(fit, target = "expected")` uses the stored draws $M_i^{(\ell)}$, $p_{ij}^{(\ell)}$, and original exposure a_i. It returns draw-level conditional means and proportions as defined below.

2. **Observed units, replicate observations.** `fitted(fit, target = "observation")` uses those same stored latent draws and original exposures, then generates an NB total and multinomial category vector. These replicates support model-fit assessment.

3. **New units, expected quantities.** `predict(fit, newdata, target = "expected")` evaluates the full model at the supplied rows. It reuses observed group and coincident spatial effects and draws new group/spatial effects as described above, then constructs $M_0^{(\ell)}$ and $p_{0j}^{(\ell)}$.

4. **New units, simulated observations.** `predict(fit, newdata, target = "observation")` first performs the same latent prediction as case 3, then draws an NB total and multinomial category vector at each new unit.

For cases 2 and 4, conditional on posterior draw ℓ, the fixed-area observation simulation at either an observed or new unit u is

$$N_u^{(\ell)} \sim \mathrm{NB}\left(a_u M_u^{(\ell)}, r_M^{(\ell)}\right), \qquad \mathbf{Y}_u^{(\ell)} \mid N_u^{(\ell)} \sim \mathrm{Multinomial}\left(N_u^{(\ell)}, \boldsymbol{p}_u^{(\ell)}\right). \tag{66}$$

Here the NB arguments denote mean and size. Size is the matching retained posterior sample when estimated, or the fitted fixed `control$nb_size` value. One category-count vector per unit and retained draw determines all four observation outputs. For the full model, the defaults return all four quantities:

Quantity	`target = "expected"`	`target = "observation"`
`magnitude`	$M_u^{(\ell)}$	$N_u^{(\ell)}/a_u$
`proportion`	$p_{uj}^{(\ell)}$	$Y_{uj}^{(\ell)}/N_u^{(\ell)}$
`density`	$M_u^{(\ell)} p_{uj}^{(\ell)}$	$Y_{uj}^{(\ell)}/a_u$
`count`	$a_u M_u^{(\ell)} p_{uj}^{(\ell)}$	$Y_{uj}^{(\ell)}$

For conditional two-category responses, j denotes a final response cell and $p_{uj}^{(\ell)}$ is the parent–child product in Equation 13. Both observation cases use this full cell vector for their multinomial draw.

When $N_u^{(\ell)} = 0$, magnitude, density, and count are zero, while all proportions for that unit and draw are `NA`. Sample output retains these entries to preserve alignment. Observation-proportion summaries condition on positive totals and are undefined if no positive draws remain. Other quantity summaries include zero-total draws. Expected proportions $p_{uj}^{(\ell)}$ need no such conditioning.

Both functions default to `target = "expected"` and `summary = "mean"`. The `type` argument selects the quantities to return; its default selects all four for the full model. Standalone defaults are given in Section 11.2. Results are `magCompPrediction` objects whose metadata identifies the target, units, draws, and any conditioning on positive totals.

For `summary = "samples"`, category outputs have dimensions unit by category by retained sample, with a final chain dimension when there are multiple chains. Magnitude omits the category dimension. Draw and chain positions align across every output in the same result. Means and quantiles pool retained draws across chains. Algebraic relationships among draw-level quantities need not hold after separately summarizing each quantity.

For results retained with `summary = "samples"`, `summary(result)` computes means, standard deviations, and quantiles from the stored draws. The help topic `magCompPrediction` describes output dimensions and metadata in detail.

Full-model and magnitude-only prediction require exposure whenever counts are requested or observations are simulated. Supply it through `area` or the fitted design's area column in `newdata`. Expected magnitude, proportion, and density alone need no exposure. `fitted()` uses the original exposures.

Use external `set.seed()` for reproducible serial workflows; for parallel fitting, supply `chain_seeds` in `magCompControl()`.

Both targets support `fixedAreaStems()` and `pointBasalArea()`. For point sampling, Equation 20 gives the observation draws

$$\begin{aligned} m_u^{(\ell)} &\sim \mathrm{NB}\left(\frac{M_u^{BA,(\ell)}}{\mathrm{BAF}_u}, r_M^{BA,(\ell)}\right), \\ \mathbf{Y}_u^{BA,(\ell)} \mid m_u^{(\ell)} &\sim \mathrm{Multinomial}\left(m_u^{(\ell)}, \boldsymbol{p}_u^{BA,(\ell)}\right), \end{aligned} \tag{67}$$

where $\boldsymbol{p}_u^{BA,(\ell)}$ allocates basal area among categories. The multinomial step is omitted for magnitude-only models; composition-only models instead condition on an observed or supplied tally. Expected magnitude and category density are $M_u^{BA,(\ell)}$ and $M_u^{BA,(\ell)} p_{uj}^{BA,(\ell)}$. Observation outputs are the sampled basal-area estimates $\mathrm{BAF}_u\, m_u^{(\ell)}$ and $\mathrm{BAF}_u\, Y_{uj}^{BA,(\ell)}$, respectively; counts are in-tree tallies and proportions are their fractions. Zero-tally proportions follow the same undefined-value rules as fixed-area observations.

The point-sampling exposure is $1/\,\mathrm{BAF}_u$, not a physical plot area. Include the fitted design's BAF column in `newdata`, or supply the reciprocal BAF through `area`. BAF is constant within each point but can differ between points. Basal-area outputs inherit the BAF units, for example square feet per acre. Individual-tree diameters are not required for a total basal-area fit.

For a fixed-area model, with `grid$plot_area` specifying sampled plot areas:

```
set.seed(42)
fitted_expected <- fitted(fit, summary = "samples")
fitted_observation <- fitted(fit, target = "observation",
                             summary = "samples")
new_expected <- predict(fit, grid, area = "plot_area",
                        summary = "samples")
new_observation <- predict(fit, grid, area = "plot_area",
                           target = "observation",
                           summary = "samples")
new_observation$count
```

By default, `predict()` uses every retained posterior draw. The optional `n_draws` argument selects that many approximately evenly spaced draws per chain before constructing new effects and observations, preserving their alignment and joint spatial dependence. Reducing this count saves time and memory at the cost of greater Monte Carlo variability, especially for interval endpoints.

For observed-count checks, `ppcSummary()` uses `fitted(target = "observation")`; see Section 14.

11.2 Standalone components

For magnitude-only models, the default outputs are `magnitude` and `count`. Expected values are $M_u^{(\ell)}$ and $e_u M_u^{(\ell)}$. For fixed-area observations, draw $N_u^{(\ell)} \sim \mathrm{NB}(a_u M_u^{(\ell)}, r_M^{(\ell)})$ and return $N_u^{(\ell)}/a_u$ and $N_u^{(\ell)}$. There is no category allocation.

Composition-only models default to `proportion`. Expected probabilities $p_{uj}^{(\ell)}$ need neither exposure nor a supplied total. To obtain conditional counts, let N_u be a known nonnegative integer total. `fitted()` uses the observed unit totals, while `predict()` requires `total_count`, supplied as a column name or one value per prediction unit. Then

$$E(Y_{uj} \mid N_u, \boldsymbol{p}_u^{(\ell)}) = N_u p_{uj}^{(\ell)}, \qquad \mathbf{Y}_u^{(\ell)} \mid N_u, \boldsymbol{p}_u^{(\ell)} \sim \mathrm{Multinomial}(N_u, \boldsymbol{p}_u^{(\ell)}).$$

The expected target returns $N_u p_{uj}^{(\ell)}$; the observation target returns the multinomial draw. Observation proportions are $Y_{uj}^{(\ell)}/N_u$. With $N_u = 0$, counts are zero and observation proportions are undefined, while expected probabilities remain defined. Two-category models still use the joint probabilities $p_{uab}^{(\ell)} = q_{ua}^{(\ell)} r_{ub|a}^{(\ell)}$.

Conditional `density` outputs divide expected or simulated counts by exposure e_u, requiring plot area for fixed-area stems or $e_u = 1/\,\mathrm{BAF}_u$ for point-sampling tallies. Count and proportion outputs need no exposure. The supplied totals are fixed across posterior draws, so these summaries condition on abundance rather than estimating its uncertainty.

11.3 Stand and forest aggregation

`standSummary()` takes a result from `fitted()` or `predict()` with `summary = "samples"`. It aggregates the supplied density draws without generating new predictions or changing their expected or observation target. Category `density` is used when available; otherwise `magnitude` supplies overall density as a single category named `total`.

Let S be a stand, stratum, or forest target area and $\mathcal{S}_0$ the set of prediction locations. The area $A_S(\boldsymbol{s}_0)$ is the land within S represented by location $\boldsymbol{s}_0$, and is zero where that location does not contribute to S. For a cell grid, use each cell's intersection area with S; the cell center need not lie inside S. These areas sum to the represented area of S, giving a finite-grid approximation to its summary. The required `represented_area` weights are distinct from plot areas $a(\boldsymbol{s}_0)$, or reciprocal BAF, used to construct predictions. The function does not infer weights from sampling exposures, grid spacing, or polygons. A cell crossing several stands contributes the same density draw to each, with its corresponding intersection area. This treats density as constant within the cell and preserves the dependence shared by those stands.

Write $d_j^{(\ell)}(\boldsymbol{s}_0)$ for the supplied category density draw.

For a full fixed-area model, the expected target supplies $M^{(\ell)}(\boldsymbol{s}_0)p_j^{(\ell)}(\boldsymbol{s}_0)$ and the observation target supplies $Y_j^{(\ell)}(\boldsymbol{s}_0)/a(\boldsymbol{s}_0)$. For point sampling, these densities are $M^{BA,(\ell)}(\boldsymbol{s}_0)p_j^{BA,(\ell)}(\boldsymbol{s}_0)$ or $\mathrm{BAF}(\boldsymbol{s}_0)Y_j^{BA,(\ell)}(\boldsymbol{s}_0)$, respectively. Thus BAF is already incorporated in point-sampling density; it is not applied again during aggregation. For each draw, the category total is

$$T_{Sj}^{(\ell)} = \sum_{\boldsymbol{s}_0 \in \mathcal{S}_0} A_S(\boldsymbol{s}_0) d_j^{(\ell)}(\boldsymbol{s}_0), \tag{68}$$

and the corresponding density is

$$\bar{d}_{Sj}^{(\ell)} = \frac{T_{Sj}^{(\ell)}}{\sum_{\boldsymbol{s}_0 \in \mathcal{S}_0} A_S(\boldsymbol{s}_0)}. \tag{69}$$

For conditional two-category responses, j indexes the joint parent–child cell; expected probabilities are $p_{ab} = q_a r_{b|a}$. Stand-level proportions are $T_{Sj}^{(\ell)} / \sum_h T_{Sh}^{(\ell)}$. They account for both represented area and abundance, rather than averaging local proportions by area alone. Point-sampling totals are basal area, densities are basal area per unit land area, and proportions describe basal-area composition. Fixed-area totals and densities refer to stems. Represented areas must use the land-area unit of the supplied density.

Magnitude-only inputs support overall density or total. Composition-only inputs support category summaries when conditional density draws are supplied, using observed totals for `fitted()` or known `total_count` for `predict()`. Those totals remain fixed across draws; the stand results retain that conditioning. Proportions alone do not provide the abundance information needed for pooled stand composition and are rejected.

Supply a data frame or tibble through `data`, with columns for prediction IDs, group IDs, and represented areas. A prediction ID can appear in several rows when its cell contributes to more than one group. Omitting `group` pools all supplied areas. The result is a `magCompPrediction` object containing the selected `density`, `total`, or `proportion` summaries.

Internally, a sparse group-by-location matrix of represented areas multiplies the density draws, preserving category, draw, and chain alignment. Stratum and spatial effects already enter through

the predictions, so aggregation uses the same calculation for these models. Prediction units must represent the appropriate model stratum when relationships differ across stratum boundaries.

For example, continuing the fixed-area prediction above, let `cell_stand_areas` contain prediction IDs, stand IDs, and cell–stand intersection areas in hectares:

```
stands <- standSummary(new_observation, data = cell_stand_areas,
                       unit = prediction_id, group = stand_id,
                       represented_area = area_ha, summary = "samples")
summary(stands)
stands$density
```

Means and intervals are computed from the aggregated draws in Equations 68 and 69, pooling chains. Expected inputs give intervals for expected quantities; observation inputs also retain count variation. All density and total draws are retained, including zeros. Proportions are undefined at zero totals, so their summaries condition on a positive total. Jointly predict the desired spatial support before aggregation: `standSummary()` preserves the supplied dependence but cannot reconstruct links omitted by separate prediction calls.

11.4 Cross-validation

`magCompCV()` rebuilds training data with complete sampling units held out and retains the original category levels. It accepts the same `target`, `type`, `summary`, and `probs` options as `predict()`, including defaults determined by the active model components. Each fold predicts its held-out units jointly using their original sampling exposures and all fitted terms. Automatic starts use training data only. Category sums are formed within each draw before summaries. Fold identities remain separate because draws from different fold fits are not joint predictions across all units. Magnitude-only CV compares total counts or magnitude. Composition-only CV assesses allocation conditional on the held-out observed totals: probabilities are predicted from the training fit, and any count or density outputs use those known totals.

For observed category counts Y_{ij} and exposure a_i, the comparison values are Y_{ij}, Y_{ij}/a_i, N_i/a_i, and Y_{ij}/N_i for count, density, magnitude, and proportion, respectively. Observed proportions are undefined for $N_i = 0$. For point sampling, replace a_i by exposure $1/\,\mathrm{BAF}_i$ in the comparison values above: counts are in-tree tallies and magnitude and density are sampled basal-area estimates. The function returns predictions and observations without calculating CRPS or coverage statistics.

12 Scope of the implementation

The package supports magnitude-only counts, composition-only allocation, or both components. Composition may have one categorical response or a conditional two-category response. Each magnitude and allocation formula requires an intercept, and only exponential spatial correlation is implemented. Each spatial term spans all sampling units; separate spatial processes by stratum are not available. Magnitude and composition latent processes have independent priors.

Weighted tree attributes, such as expanded volume, biomass, or carbon, are generally noninteger quantities and cannot be substituted into the count likelihood. The point-sampling basal-area family uses integer tally counts, with the boundary limitation described in Section 4.

13 Software workflow

Function arguments, return values, and runnable examples are documented in R's help pages. Start with `help("magComp-package")` for an overview; `help("magCompNotation")` connects parameter names to the notation below, and `help("magCompPrediction")` describes fitted and prediction outputs. The compiled function reference is also available as `dev/manual/magComp-manual.pdf` in the source repository.

13.1 Mathematical notation and software names

Public settings use descriptive names within magnitude, composition-block, and named-term branches. In the following table, "start" means an entry in `starting`, and "prior" means an entry in `priors`. Term names distinguish multiple effects of the same kind.

Quantity	Symbol	Software setting
Magnitude coefficients	$\boldsymbol{\beta}$	Magnitude start `coefficients`; prior `coefficient_variance` is σ^2_β.
Composition coefficients	$\alpha_{0j}, \boldsymbol{\alpha}_j$	Block start `coefficients`; priors `intercept_variance` and `slope_variance` are $\sigma^2_{\alpha,0}$ and σ^2_α.
Magnitude iid effects	δ_{gt}	Magnitude term start `effects`; `variance` is $\sigma^2_{\delta,t}$.
Magnitude spatial effects	$v_t(\boldsymbol{s}_i)$	Magnitude term start `effects`; `variance` is $\sigma^2_{v,t}$, and `covariance$phi` is $\phi_{v,t}$.
Composition iid factors	$\boldsymbol{\gamma}_{jt}, \boldsymbol{u}_{th}$	Factor-term starts `loadings` and `scores`; prior `loading_variance` is $c_{\gamma,t}$.
Composition spatial factors	$\boldsymbol{\lambda}_{jt}, \boldsymbol{w}_t(\boldsymbol{s}_i)$	Factor-term starts `loadings` and `scores`; prior `loading_variance` is $c_{\lambda,t}$, and start `covariance$phi` contains $\phi_{w,t,k}$.
Negative-binomial size	r_M	Magnitude `size` prior and start when estimated; `control$nb_size` when fixed.

A term's `variance` prior entry contains inverse-gamma shape and scale; its starting entry contains a numerical variance. Likewise, a `covariance$phi` prior contains uniform bounds, while its starting entry contains a decay value. Coefficient and loading prior variances are fixed hyperparameters; they are distinct from the estimated magnitude process variances.

The public `samples(fit)` extractor returns all retained parameter draws in the same component, block, and term branches as starting values. For example,

```
draws <- samples(fit)
draws$magnitude$coefficients
draws$magnitude$terms$stand$variance
draws$composition$blocks$dbh$terms$shape$loadings
```

The first array contains $\boldsymbol{\beta}$ draws, the second contains $\sigma^2_{\delta,\text{stand}}$ draws for an iid `stand` term, and the third contains the loadings of factor term `shape` in block `dbh`.

Sample and chain positions align across all arrays. The extractor returns an object with `summary()` and `plot()` methods; `start`, `thin`, and `chains` select retained draws consistently. The help page for `samples()` describes the array dimensions and chain-pooling options.

Use `type = "effects"` to extract individual effects and scores. The `type = "combined"` option returns fixed coefficients plus their matching random coefficients. Section 14.2 explains these combinations and their diagnostics.

For new units, request `keep_effects = TRUE` with prediction samples. The same extractor then returns the parameter and latent-effect draws used in those predictions, including combined coefficients at the new locations:

```
predictions <- predict(fit, newdata, target = "observation",
                       summary = "samples", n_draws = 100,
                       keep_effects = TRUE)
combined <- samples(predictions, type = "combined")
summary(combined)
```

These combinations remain on the linear-predictor scale and use the effects already drawn for prediction. Grouping refers to columns in `newdata`. Use `standSummary()` for area-weighted response summaries.

13.2 Preparing settings and fitting models

Use explicit `NULL` to omit a component:

```
magnitude_fit <- magComp(dat, magnitude = ~ p95, composition = NULL)
composition_fit <- magComp(dat, magnitude = NULL, composition = ~ p95)
```

The second call can also use `magCompComposition()` for a conditional two-category model. Section 11.2 describes the outputs.

The public prior and starting-value interfaces use matching magnitude, composition-block, and named-term branches. The prior constructor accepts magnitude and composition defaults plus block/term overrides. A prepared model can be inspected without sampling:

```
model <- magComp(dat, magnitude = ~ iid(stand, name = "stand"),
                 control = magCompControl(run_sampler = FALSE))
settings <- magCompSettings(model)
settings$priors$magnitude$terms$stand$variance <- c(shape = 3, scale = 1)
settings$starting$magnitude$terms$stand$variance <- 0.5
```

The settings table reports estimated, fixed, and constrained parameters, resolved priors, initial values, and formula term names. Its editable lists can be supplied to a new fit through `priors` and `starting`. `magCompStarting(model)` returns the original initial values; `use = "last"` or `use = "mean"` extracts posterior values for initializing another run, without restoring its RNG or adaptation state.

The main fitting function is `magComp()`, which runs the sampler by default. In `magCompControl()`, `n_samples` is the total number of iterations per chain, including `n_burn`. Thinning applies after burn-in, leaving $\lfloor(\texttt{n_samples} - \texttt{n_burn})/\texttt{n_thin}\rfloor$ draws per chain. Defaults are 2,000 total iterations, 1,000 burn-in iterations, and thinning by one. Chain lengths should be chosen using the diagnostics in Section 14.

A one-category fixed-area stand-table analysis has the form

```
d <- magCompData(trees,
                 unit = plot_id,
                 category = dbh_class,
                 coords = c(x, y),
                 design = fixedArea(area))

fit <- magComp(d,
               family = fixedAreaStems(),
               magnitude = ~ p95 + gap + gp(x, y),
               composition = ~ p25 + p50 + p95 +
                 factor_gp(x, y, K = 3) +
                 p95:factor_gp(x, y, K = 3),
               smooth = magCompSmooth(intercept = "rw2",
                                      slope = "rw2"))
```

A two-category table uses `categories` and an explicit `magCompComposition()` specification:

```
d <- magCompData(trees,
                 unit = plot_id,
                 categories = c(species_group, dbh_class),
                 ordered = c(species_group = FALSE,
                             dbh_class = TRUE),
                 coords = c(x, y),
                 design = fixedArea(area))

fit <- magComp(d,
               family = fixedAreaStems(),
               magnitude = ~ p95 + gap + nngp(x, y, m = 15),
               composition = magCompComposition(
                 species_group ~ p95 + gap + factor_iid(stratum, K = 1),
                 dbh_class | species_group ~ p25 + p50 + p95 +
                   factor_nngp(x, y, K = 3, m = 15) +
                   p95:factor_nngp(x, y, K = 3, m = 15)
               ),
               smooth = magCompSmooth(dbh_class =
                 list(intercept = "rw2", slope = "rw2")))
```

The left-hand side of each composition formula names the response category for that allocation block. A conditioning expression such as `dbh_class | species_group` defines the parent context.

Ordered category columns must be factors with levels in the intended order, or numeric values, which are sorted numerically. Character class labels require an explicit factor-level order. Coordinates, model predictors, and grouping columns must be constant within each sampling unit; tree-level responses and attributes may vary.

14 Diagnostics

`summary(fit)` reports posterior means, standard deviations, quantiles, $\widehat{R}$, and effective sample sizes for the scalar parameters in each active component. `plot(fit)` provides traceplots. These methods also support individual effects and combined coefficients through the same `type` argument as `samples()`.

The reported $\widehat{R}$ is the classical unsplit variance-ratio statistic, bounded below by one; it is not rank normalized. ESS truncates the within-chain autocorrelation sum at the first nonpositive adjacent lag-pair sum and adds the result across chains, without a between-chain correction. A high ESS can therefore coexist with chains at different locations. One-chain $\widehat{R}$ and diagnostics for constant traces are undefined. `spatialDiagnostics()` summarizes covariance parameters, spatial variance, loading-column norms, Metropolis acceptance, and correlations between covariance parameters and variance or loading norms.

14.1 Posterior predictive checks

`ppcSummary()` compares observed counts with replicate counts at the unit, category, or group level, as supported by the fitted components. Composition-only checks condition on the observed unit totals. Point-sampling checks compare in-tree tallies. Each call generates fresh replicates through `fitted(target = "observation", type = "count", summary = "samples")` and compares them with the observations. For the observed count or count sum T selected by the check, it reports the replicate mean and quantiles, empirical posterior predictive tail fractions $p_{\text{lower}} = P(T^{\text{rep}} \leq T \mid \text{data})$ and $p_{\text{upper}} = P(T^{\text{rep}} \geq T \mid \text{data})$, and

$$p_{\text{two}} = \min\{1, 2\min(p_{\text{lower}}, p_{\text{upper}})\}.$$

This last quantity is returned as `p_value`, a two-sided posterior predictive p-value. Small values flag an observed count in a tail of the replicate distribution; values near one indicate a central count. Ties count in both tails. The value is not a probability that the model is correct, and it does not have the usual hypothesis-test calibration because the same data inform the fit and the check. Replicate quantiles describe observation variation as well as parameter uncertainty, rather than uncertainty in an expected count alone. These checks assess the fitting data, not cross-validation predictions.

Plots of observed and fitted category profiles complement these checks. For ordered categories, inspect the resulting proportions as well as the coefficients, since smoothing acts on continuation-ratio predictors.

14.2 Combined-coefficient diagnostics

The `type = "combined"` option adds fixed coefficients and their matching random coefficients within each posterior draw. With iid and spatial intercept terms, the magnitude intercept at unit i is $\beta_0 + \delta_{g(i)} + v(\boldsymbol{s}_i)$, and the composition intercept is $\alpha_{0j} + \boldsymbol{\gamma}_j^\top \boldsymbol{u}_{h(i)} + \boldsymbol{\lambda}_j^\top \boldsymbol{w}(\boldsymbol{s}_i)$. Additional matching terms contribute additively. Conditional blocks retain their parent-specific coefficients and loadings.

Latent random intercepts can trade off with fixed intercepts even when fitted linear predictors are well identified. For a magnitude random-intercept term with effect $v_i^{(\ell)}$ at unit i and posterior draw

ℓ, define

$$\bar{v}^{(\ell)} = \frac{1}{n}\sum_{i=1}^{n} v_i^{(\ell)}, \tag{70}$$

where $v_i^{(\ell)}$ is the group effect $\delta_{g(i)}^{(\ell)}$ for an iid term or the spatial effect $v^{(\ell)}(\boldsymbol{s}_i)$. The combined magnitude intercept is

$$\tilde{\beta}_0^{(\ell)} = \beta_0^{(\ell)} + \bar{v}^{(\ell)}. \tag{71}$$

For allocation block b and stick row j, let $f_{ij,0}^{(C,b,\ell)}$ denote the sum of its intercept-factor contributions. Their average is

$$\bar{f}_{bj}^{(\ell)} = \frac{1}{|\mathcal{I}_{bj}|}\sum_{i\in\mathcal{I}_{bj}} f_{ij,0}^{(C,b,\ell)}, \tag{72}$$

using only the intercept terms in Equation 21. Here $\mathcal{I}_{bj}$ is the set of allocation rows governed by stick row j. For a leading block this set contains all units. For a conditional block it contains only rows in the corresponding parent context. The combined composition intercept is

$$\tilde{\alpha}_{0j}^{(b,\ell)} = \alpha_{0j}^{(b,\ell)} + \bar{f}_{bj}^{(\ell)}. \tag{73}$$

Traceplots, effective sample sizes, and $\widehat{R}$ for $\tilde{\beta}_0^{(\ell)}$ and $\tilde{\alpha}_{0j}^{(b,\ell)}$ help distinguish poor raw-intercept mixing from harmless intercept–latent-effect location confounding. For multiple random-intercept terms, sum their average contributions. Random slopes are combined separately with their corresponding fixed slopes; their effects are not multiplied by covariate values or added to the intercept. The `group = ".all"` option averages over fitted units as above; omitting `group` returns a combined coefficient for each unit, and naming a grouping column averages within its groups. These are equal-unit averages of coefficients. For example,

```
combined <- samples(fit, type = "combined", group = "stand")
summary(combined)
```

The equivalent call `summary(fit, type = "combined", group = "stand")` uses the same derived draws. These diagnostics summarize the existing draws without recentering the sampler or changing its prior.

References

Albert, J. H., and Chib, S. (1995). Bayesian residual analysis for binary response regression models. *Biometrika*, 82, 747–759.

Finley, A. O., Datta, A., Cook, B. D., Morton, D. C., Andersen, H. E., and Banerjee, S. (2019). Efficient algorithms for Bayesian nearest neighbor Gaussian processes. *Journal of Computational and Graphical Statistics*, 28, 401–414. doi:10.1080/10618600.2018.1537924.

Gneiting, T. (2002). Nonseparable, stationary covariance functions for space–time data. *Journal of the American Statistical Association*, 97, 590–600. doi:10.1198/016214502760047113.

Roberts, G. O., and Rosenthal, J. S. (2009). Examples of adaptive MCMC. *Journal of Computational and Graphical Statistics*, 18, 349–367. doi:10.1198/jcgs.2009.06134.

Polson, N. G., Scott, J. G., and Windle, J. (2013). Bayesian inference for logistic models using Polya-Gamma latent variables. *Journal of the American Statistical Association*, 108, 1339–1349.

Rouder, J. N., and Lu, J. (2005). An introduction to Bayesian hierarchical models with an application in the theory of signal detection. *Psychonomic Bulletin & Review*, 12, 573–604.

Taylor-Rodriguez, D., Finley, A. O., Datta, A., Babcock, C., Andersen, H.-E., Cook, B. D., Morton, D. C., and Banerjee, S. (2019). Spatial factor models for high-dimensional and large spatial data: An application in forest variable mapping. *Statistica Sinica*, 29, 1155–1180. doi:10.5705/ss.202018.0005. Positive-diagonal sampling for the paper's Γ matrix is detailed in Appendix B of arXiv:1801.02078v2.