



Reducing the computational cost of dynamic global vegetation models with locally weighted hierarchical clustering

Jamir Priesner^{1,2,a} , David Hötten^{1,3,b}, Christof Schötz^{1,2,c}, Britta Tietjen^{4,d}, Maximilian Gelbrecht^{1,2,e}, Maik Billing^{1,f}, Fabian Stenzel^{1,5,g}, and Kirsten Thonicke^{1,h}

¹ Potsdam Institute for Climate Impact Research (PIK), Member of the Leibniz Association, P. O. Box 60 12 03, 14412 Potsdam, Germany

² Technical University of Munich, Munich Climate Center and TUM School of Engineering and Design, Munich, Germany

³ Freie Universität Berlin, Department of Mathematics and Computer Science, Berlin, Germany

⁴ Freie Universität Berlin, Theoretical Ecology, Institute of Biology, Berlin, Germany

⁵ Stockholm Resilience Centre (SRC), Stockholm, Sweden

Received 30 April 2026 / Accepted 7 July 2026
© The Author(s) 2026

Abstract Dynamic Global Vegetation Models (DGVMs) with embedded tree demography and flexible plant traits simulate dynamics across tens of thousands of land grid cells, each containing tens of thousands of individual trees, making global simulations computationally expensive. Aggregating cells into climatically coherent clusters and running a single simulation per cluster reduce this cost by orders of magnitude, but cluster quality critically depends on an ecologically meaningful distance metric between cells. We present a framework in which spatially adaptive distance metrics are derived from Köppen–Geiger climate-zone labels via Large Margin Nearest Neighbour (LMNN) metric learning and used for hierarchical clustering. A multilayer perceptron predicts cell-specific positive semi-definite metric matrices, allowing feature weighting to adapt continuously across ecological regimes. We compare a learned metric that captures cross-feature interactions against a simpler variant without interaction terms and the uniform Euclidean metric. Experiments on 67,420 global land cells with 51 climate and soil features, testing cluster counts across five orders of magnitude, show that both learned metrics substantially improve agreement with Köppen–Geiger climate-zone boundaries compared to the Euclidean uniform metric. This improved ecological alignment translates into more accurate biomass reconstruction well beyond the training signal: the learned metrics outperform the Euclidean baseline across cluster counts spanning more than an order of magnitude above the 30-zone signal before all three metrics converge. For cluster counts ≥ 2132 , the Euclidean metric performs better, showing that at this scale, different features down-weighted by the metric learning—such as soil properties—become more important. The variant without interaction terms achieves biomass reconstruction accuracy equal to the interaction-based metric with directly interpretable feature weights. Evaluated with the individual-based DGVM LPJmL-FIT, more than 80% of cell–year pairs stay within the model-inherent stochastic noise floor already at an energy consumption of only 0.1% of the full-grid cost.

^a e-mail: jamir.priesner@pik-potsdam.de (corresponding author)

^b e-mail: davidho@pik-potsdam.de

^c e-mail: christof.schoetz@tum.de

^d e-mail: britta.tietjen@fu-berlin.de

^e e-mail: gelbrecht@pik-potsdam.de

^f e-mail: billing@pik-potsdam.de

^g e-mail: stenzel@pik-potsdam.de

^h e-mail: kirsten.thonicke@pik-potsdam.de

Abbreviations

Acronym	Definition
ARI	Adjusted Rand Index
CPU	Central processing unit
CV	Coefficient of variation
DGVM	Dynamic global vegetation model
ESM	Earth system model
GPU	Graphics processing unit
HPC	High performance computer
ISIMIP3a	Inter-sectoral impact model intercomparison project reanalysis phase 3
KG	Köppen–Geiger
LMNN	Large margin nearest neighbour
LW	Longwave (radiation)
MLP	Multilayer perceptron
MPI	Message passing interface
NMI	Normalised mutual information
PC	Principal component
PCA	Principal component analysis
PFT	Plant functional type
PSD	Positive semi-definite
ReLU	Rectified linear unit
SPEI	Standardised precipitation-evapotranspiration index
SW	Shortwave (radiation)
TPE	Tree-structured Parzen estimator
VegC	Vegetation carbon

Mathematical symbols

n	Number of land grid cells (= 67 420)
d	Number of input features (= 51)
K	Number of clusters
r	Maximum rank of the Low-rank metric/number of columns of $\mathbf{L}_i$ (= 8)
$\mathbf{X}$	Standardised feature matrix, $\mathbb{R}^{n \times d}$
$\mathbf{x}_i$	Feature vector of cell i , $\in \mathbb{R}^d$
$\mathbf{M}_i$	Cell-specific positive semi-definite metric matrix
$\mathbf{L}_i$	Low-rank factor matrix, $\in \mathbb{R}^{d \times r}$; $\mathbf{M}_i = \mathbf{L}_i \mathbf{L}_i^\top$
$\mathbf{w}_i$	Per-feature weight vector (Diagonal variant), $\in \mathbb{R}_{\geq 0}^d$
$\mathbf{I}_d$	$d \times d$ identity matrix
D_{maha}	Symmetrised local Mahalanobis distance (Low-rank)
$\mathbf{D}_{\text{maha}}$	Full $n \times n$ pairwise distance matrix, $[\mathbf{D}_{\text{maha}}]_{ij} = D_{\text{maha}}(\mathbf{x}_i, \mathbf{x}_j)$
D_{diag}	Symmetrised diagonal-weighted distance (diagonal)
α	Margin trade-off weight in the LMNN loss
Δ	Margin in the hinge loss term
Δ_{base}	Base margin before dimension scaling
d_0	Reference feature dimension for margin scaling (= 27)
λ_{reg}	Regularisation weight (Low-rank variant)
$\mathcal{T}_i$	Set of target (same-zone) neighbours of cell i
$\mathcal{I}_i$	Set of impostor (different-zone) cells for cell i
k_i	Number of target neighbours of cell i
$\lambda_k^{(i)}$	k th eigenvalue of $\mathbf{M}_i$
$\mathbf{v}_k^{(i)}$	k th eigenvector of $\mathbf{M}_i$
$p_j^{(i)}$	Weight profile: fraction of metric sensitivity on feature j at cell i
H_i	Metric isotropy (normalised Shannon entropy of the eigenvalue spectrum)
F	Cluster fraction, $F = K/n$
E	Energy saving factor, $E = n/K = 1/F$
δ	Accuracy threshold (95th percentile of inter-seed baseline error)

P_{acc}	Accuracy probability: fraction of cell×year pairs within δ
Q	Rao's quadratic entropy (functional diversity)
$\bar{S}$	Mean silhouette score (Appendix G only)
$\bar{s}$	Mean geographic spread, in degrees (Appendix I)
$\bar{v}$	Mean within-cluster feature variance
$d_i^{(m, K)}$	Local cluster diversity of method m at cluster count K in the 5×5 window around cell i
b_i^{KG}	KG-boundary proximity: local cluster diversity of the 30-class KG partition in the window around cell i
b_i^{BM}	Local biomass-gradient magnitude (window std of baseline biomass, normalised by its global std)
$\text{BAR}_{\text{KG}}(m, K)$	Boundary Allocation Ratio at KG boundaries: ratio of mean d_i in b^{KG} -top quartile to mean d_i in bottom quartile (Eq. 8)
$\text{BAR}_{\text{BM}}(m, K)$	Boundary Allocation Ratio at biomass gradients (analogous to BAR_{KG} using b^{BM} ; Appendix J)
ΔBAR	Misallocation gap, $\text{BAR}_{\text{KG}} - \text{BAR}_{\text{BM}}$
η	Learning rate (AdamW optimiser)
p	Dropout probability
h	Hidden-layer width factor
ϕ, ψ, θ	Trainable MLP parameters (Low-rank, Diagonal variants)
$\ \cdot\ _F$	Frobenius norm

1 Introduction

Dynamic Global Vegetation Models (DGVMs) were developed to study how vegetation dynamics affect biogeochemical and hydrological cycles, and to quantify the past, present, and future functioning of the terrestrial carbon sink [1–3]. A persistent challenge in DGVM development has been the demand for greater ecological realism, particularly with respect to biodiversity [4, 5]. First-generation DGVMs represent vegetation through a small set of plant functional types (PFTs), each characterised by a single average individual with fixed trait values, grouping functionally diverse species into a few biome-wide classes [1, 6]. This simplification fundamentally limits the ability of models to capture trait variation within biomes, community assembly, and the functional diversity that underpins ecosystem resilience and carbon cycling [1, 6–9].

Two broad approaches have emerged to overcome this limitation. The first is the introduction of explicit tree demography into DGVMs, where individual trees or cohorts are tracked through time, allowing forest structure and species composition to emerge from competitive dynamics and environmental filtering. This line of development produced models, such as LPJ-GUESS [10, 11], ED2 [12, 13], FATES [14], FORMIND [15], and TROLL [16], and was advocated for the next generation of ESM land components [17]. The second approach combines tree demography with the plant economics spectrum, allowing individual plants to express a continuous range of trait combinations rather than belonging to discrete PFT classes. Models following this approach—including LPJmL-FIT [4], aDGVM2 [6], and LPJ-GUESS-NTD [18]—can represent the within-biome trait diversity that underpins ecosystem functioning. Both approaches, and especially their combination, come with a substantial increase in computational cost: to capture stochastic gap dynamics and sufficient demographic and functional diversity, up to 2500 independent forest patches must be simulated per cell according to Thonicke et al. [5]. Applied to tens of thousands of grid cells worldwide for simulation periods spanning decades to centuries—or millennia when equilibrium runs are required [9, 19]. This makes global simulations with biodiversity-enhanced DGVMs computationally expensive [17, 20], entailing the need for HPC infrastructure to conduct them—which comes with substantial energy and financial costs [21–23]. These costs act as a barrier to participation, effectively limiting global vegetation modelling to researchers with access to adequate computing resources and funding [24]. Moreover, the energy demand also carries an environmental footprint that, depending on the energy source, may contribute to climate change [22, 23, 25]. These demands are further compounded by the trend towards higher spatial resolution in DGVMs [21, 26], and by the need to run large ensembles across multiple climate scenarios to adequately sample model uncertainty and internal climate variability [26, 27]—a requirement made particularly acute by vegetation's complex and nonlinear response to climate, whereby small shifts in temperature or precipitation can trigger disproportionate changes in ecosystem state [28, 29].

One way to mitigate this computational bottleneck is to exploit the spatial redundancy inherent in global simulations. At the standard $0.5^\circ \times 0.5^\circ$ grid commonly used in DGVMs, the 67,420 land cells include many groups with similar climate and soil conditions, yet each is simulated independently at full cost [2, 3, 30]. If cells with sufficiently similar environments were grouped into clusters and only representative cells simulated per cluster,

the computational cost would scale with the number of clusters K rather than the number of cells n , potentially reducing it by orders of magnitude while incurring only modest errors in the reconstructed output fields.

Several studies have demonstrated the value of partitioning grid cells into clusters to reduce redundant computation in spatially distributed models. Dietrich et al. [31] showed that clustering substantially reduces information loss in the global land-use model MAGPIE while preserving model accuracy, and Shivakumar et al. [32] applied a similar aggregation principle to water–energy–food nexus modelling. In hydrology, Ehret et al. [33] introduced adaptive clustering of model elements based on their dynamic similarity during runtime, achieving 75% reductions in computation time. Chaney et al. [34] demonstrated that hierarchical clustering of high-resolution environmental covariates into representative tiles can reproduce macroscale water, energy, and carbon cycle responses at a fraction of the computational cost of fully distributed land surface model runs. For vegetation modelling specifically, Renne et al. [35] developed a k -means-based site selection algorithm that assigns simulation outputs from a limited set of representative sites to environmentally analogous locations, demonstrated using an individual-based plant model across North American drylands.

Grouping grid cells into clusters requires a metric that quantifies the similarity between them. The approaches mentioned above have in common that they define similarity between grid cells using a globally uniform distance metric with equal or heuristically chosen feature weights (e.g. climate variables), treating the relative importance of each feature as spatially invariant. This assumption is ecologically implausible at the global scale: the features that best discriminate between ecologically distinct cells differ fundamentally across climate zones—for example, moisture limitations dominate in the subtropics, whereas temperature gradients are decisive at high latitudes [36–38]. A uniform metric may therefore group ecologically dissimilar cells into the same cluster or separate ecologically similar cells, systematically degrading the accuracy of reconstructed model outputs in regions where the most discriminating features are underweighted.

We address this limitation by introducing a locally adaptive, learned distance metric whose feature weights vary spatially according to regional ecological relevance. The approach requires two properties of the distance metric: (i) it should weight climate and soil features according to their local discriminative power, varying continuously across the globe, and (ii) it should be learned from generally available observation-based data without requiring prior DGVM simulations, so that the method is directly applicable to any model with spatially independent cells.

To this end, we extend Large Margin Nearest Neighbour (LMNN) metric learning [39]. While LMNN learns a single global Mahalanobis distance metric, we learn a locally adaptive metric by having a multilayer perceptron (MLP) predict a cell-specific metric matrix from the local climate and soil features. Because learning a full metric matrix for each cell would be computationally expensive and prone to overfitting, we adopt a *Low-rank* representation. Additionally, we evaluate a *Diagonal* variant that further reduces the number of learned parameters and improves interpretability of the learned feature weights.

For both methods, we use the Köppen–Geiger (KG) classification as the supervision signal: its globally consistent zone boundaries encode well-established relationships between climate and vegetation, making it a natural choice for guiding metric learning at the global scale [40–42]. Both learned metrics are benchmarked against a uniform *Euclidean* metric that treats all features as equally important.

A central empirical question addressed in this work is therefore at which cluster-count scales a metric trained on a fixed, relatively coarse supervision signal (here: 30 KG zones) yields better clusters than a uniform Euclidean metric.

We combine the learned distance metrics with hierarchical agglomerative clustering [43, 44], which produces a single dendrogram from which clusterings at any resolution can be extracted without retraining.

Using the individual-based DGVM LPJmL-FIT [4, 9] as a case study, we compare the three distance metrics across cluster counts spanning five orders of magnitude in computational cost reduction, evaluating both the structural quality of the clustering against KG climate zones and the accuracy of reconstructed biomass and functional diversity fields. We find that both learned metrics substantially outperform Euclidean clustering at low cluster numbers K , with the simpler Diagonal variant matching the Low-rank metric in reconstruction accuracy, while Euclidean clustering performs better at higher K .

2 Methods

The goal is to cluster the 67,420 land cells of the global 0.5° grid used in LPJmL-FIT for this study into $K \ll n$ groups of ecologically similar cells, so that simulating only one representative cell per cluster reproduces the full-grid DGVM output, including plant traits, with acceptable accuracy. The key methodological challenge is to define a distance metric between cells that reflects locally varying ecological relevance of the input features, rather than treating all features equally everywhere. The aim is to identify clusters that are as large as possible for computational efficiency while remaining ecologically homogeneous enough that a single representative cell can stand in for all cluster members with acceptable loss of accuracy. This requires (i) a source of ecological information

that encodes which features matter where, and (ii) a learning algorithm that translates this information into spatially adaptive feature weights.

For (i), we use the Köppen–Geiger climate classification (Sect. 2.2), whose zone boundaries encode well-established relationships between climate and vegetation distribution and thus provide a supervision signal that captures regionally varying ecological relevance. The framework is not tied to KG, however; any ecologically meaningful partition can serve as supervision (e.g., biome maps or ecoregion delineations). For (ii), we extend Large Margin Nearest Neighbour (LMNN) metric learning (Sect. 2.3), so that a multilayer perceptron (MLP) takes the climate and soil feature vector of each cell as input and predicts a cell-specific positive semi-definite (PSD) metric matrix $\mathbf{M}_i$ that concentrates distance sensitivity onto the features most discriminative for the local KG zone boundaries. The most general formulation would learn a full-rank $d \times d$ matrix per cell ($O(d^2)$ parameters), but this is computationally expensive and prone to overfitting. We therefore adopt a *Low-rank* variant of maximum rank $r \ll d$, in which $\mathbf{M}_i = \mathbf{L}_i \mathbf{L}_i^\top$ captures cross-feature interactions at a substantially reduced parameter count. As a further simplification, we evaluate a *Diagonal* variant in which $\mathbf{M}_i$ is a diagonal matrix whose entries weight each feature independently, without modelling cross-feature interactions. Both learned variants are compared against a uniform *Euclidean* baseline ($\mathbf{M}_i = \mathbf{I}_d$ for all cells). The resulting pairwise distance matrix is then used for hierarchical agglomerative clustering (Sect. 2.6), and the clustering quality is evaluated by reconstructing DGVM output fields from representative cells and comparing them against a full-grid baseline simulation (Sect. 2.7). The input feature matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ ($n = 67\,420$ valid land cells, $d = 51$ features) is prepared as described in Sect. 2.1; full details of feature computation, standardisation, and filtering are provided in the Supplementary Material (Sections S1–S3).

2.1 Data and features

As inputs we use global gridded climate data from the Inter-Sectoral Impact Model Intercomparison Project reanalysis phase 3 (ISIMIP3a) [45] at 0.5° longitude–latitude resolution—precipitation, near-surface air temperature, shortwave and longwave radiation, and relative humidity—together with soil depth [46] and a categorical soil-texture classification [47]. Only land cells with valid soil data were retained ($n = 67\,420$). Details of coordinate canonicalisation, regridding, and valid-cell masking are given in the Supplementary Material (Section S1).

From these inputs, 69 ecological features were initially derived per cell, including climatic summary statistics, monthly climatological profiles, Köppen–Geiger diagnostic scalars, drought indices (SPEI at three accumulation timescales), and soil indicators (full definitions in Supplementary Material Section S2). We retained this broader feature set, rather than only KG-diagnostic variables, because the downstream clustering is evaluated over a wide range of cluster counts and against DGVM outputs that are not limited to the 30 KG classes. This choice reflects a balance between the supervision signal and the intended use of the learned metric: training on 30 KG zones will primarily favour features that help separate those zones, while the additional hydroclimatic and soil features preserve information that may become relevant for ecologically meaningful distinctions within and across KG zones at higher K . Redundancy was reduced through correlation filtering (removing highly correlated pairs at $|\rho| > 0.95$; Supplementary Material Section S3.2) and block PCA compression of monthly temperature and precipitation profiles (retaining 99 % of within-block variance; Supplementary Material Section S3.3), yielding $d = 51$ features in ten groups (Table 1). All features were z-score standardised [48] (Supplementary Material Section S3.1).

Table 1 Summary of input feature groups ($d = 51$ after filtering). “Protected” denotes feature groups that were exempt from correlation filtering and PCA compression

Feature group	Features	n	Protected
Temperature summaries	Mean, seasonality	2	No
Temperature monthly PCs	PC ₁ , PC ₂ (99% var.)	2	Yes
Precipitation summaries	Mean, max, coeff. of variation, wet-day freq.	4	No
Precipitation monthly PCs	PC ₁ ...PC ₇ (99% var.)	7	Yes
KG-diagnostic scalars	Threshold-derived scalars	8	Yes
Radiation (SW & LW)	Mean, std, amplitude ($\times 2$)	6	No
Humidity	Mean, std, minimum	3	No
SPEI drought indices	Mean & std at $\tau=3, 6, 12$ months	6	No
Soil depth	Continuous	1	Yes
Soil texture	One-hot encoded (12 classes)	12	Yes
		51	

2.2 Köppen–Geiger supervision labels

Each grid cell was assigned a Köppen–Geiger climate zone from its monthly climatological temperature and precipitation series, following Peel et al. [42] with 30 subclasses (C/D threshold at 0°C): **A** tropical (temperature of the coldest month $T_{\text{cold}} \geq 18^\circ\text{C}$), **B** arid (annual precipitation $P_{\text{ann}} < 10 P_{\text{th}}$, where P_{th} is an aridity threshold depending on annual temperature and precipitation seasonality), **C** temperate ($0 \leq T_{\text{cold}} < 18^\circ\text{C}$), **D** continental ($T_{\text{cold}} < 0^\circ\text{C}$ and temperature of the warmest month $T_{\text{hot}} \geq 10^\circ\text{C}$), and **E** polar ($T_{\text{hot}} < 10^\circ\text{C}$); full criteria are given in Appendix B. These zone labels serve as the supervision signal for metric learning: the objective is not to reproduce the KG partition but to learn a distance metric that respects locally varying feature importance as encoded in the KG system. Once learned, the same metric supports clustering at any resolution—coarser or finer than KG.

2.3 Local Mahalanobis LMNN

2.3.1 Parameterisation

Low-rank variant. For each cell i with feature vector $\mathbf{x}_i \in \mathbb{R}^d$, an MLP $g_\phi : \mathbb{R}^d \rightarrow \mathbb{R}^{d \times r}$ predicts a Low-rank factor $\mathbf{L}_i \in \mathbb{R}^{d \times r}$, from which the local metric is

$$\mathbf{M}_i = \mathbf{L}_i \mathbf{L}_i^\top. \quad (1)$$

By construction $\mathbf{M}_i$ is PSD with rank $\leq r$. The tuned rank parameter $r = 8$ (for $d = 51$; Sect. 2.4) requires $d \cdot r = 408$ MLP outputs per cell. The MLP uses two hidden layers of equal width $\lfloor d \cdot h \rfloor = 178$ (with $h = 3.50$), ReLU activations, and dropout; architectural details are given in Appendix C.

Diagonal variant. The Diagonal variant replaces the full matrix with a per-feature weight vector. A separate MLP $h_\psi : \mathbb{R}^d \rightarrow \mathbb{R}^d$ predicts a positive weight vector $\mathbf{w}_i \in \mathbb{R}_{\geq 0}^d$ for each cell, yielding the diagonal metric $\mathbf{M}_i = \text{diag}(\mathbf{w}_i)$. Positivity is ensured by a Softplus output activation, and the weights are L_1 -normalised to sum to 1, making them directly interpretable as per-feature metric weight scores. The architecture uses two hidden layers of decreasing width ($\lfloor d \cdot h \rfloor = 108$ and $\lfloor d \cdot h/2 \rfloor = 54$, with $h = 2.12$) with ReLU activations and dropout. Because the output has d units rather than $d \cdot r$, the Diagonal MLP has substantially fewer parameters than the Low-rank variant.

2.3.2 Distance

For the Low-rank variant, the distance between cells i and j is defined using the average of their local metric matrices

$$D_{\text{maha}}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^\top \frac{\mathbf{M}_i + \mathbf{M}_j}{2} (\mathbf{x}_i - \mathbf{x}_j)}. \quad (2)$$

For the Diagonal variant, the corresponding distance uses the averaged weight vectors

$$D_{\text{diag}}(\mathbf{x}_i, \mathbf{x}_j) = \sqrt{\sum_{k=1}^d \frac{w_{ik} + w_{jk}}{2} (x_{ik} - x_{jk})^2}, \quad (3)$$

which is equivalent to Eq. (2) with $\mathbf{M}_i = \text{diag}(\mathbf{w}_i)$, but avoids the full matrix–vector product. Both formulations guarantee symmetry: $D(\mathbf{x}_i, \mathbf{x}_j) = D(\mathbf{x}_j, \mathbf{x}_i)$.

2.3.3 Loss function

The MLP parameters are optimised with a loss function derived from the standard LMNN objective [39]; full mathematical definitions are given in Appendix D. For each training cell i , its k_i nearest same-zone neighbours (in Euclidean feature space) form the *target set* $\mathcal{T}_i$, while the nearest neighbour cells belonging to a different KG zone form the *different-zone set* $\mathcal{I}_i$ (capped at 50 per cell).

The loss comprises three components weighted by a trade-off parameter α :

1. A **pull term** that shrinks the learned distance between a cell i and the corresponding $\mathcal{T}_i$ cells, encouraging climatically similar cells to be placed close together.

2. A **margin term** that penalises violations of a minimum gap Δ between the distance to a neighbour in $\mathcal{T}_i$ and the distance to a neighbour in $\mathcal{I}_i$, encouraging zone boundaries to remain well-separated in the learned metric space.
3. A **regularisation term** (Low-rank variant only) that anchors each metric matrix $\mathbf{M}_i$ to the identity, preventing metric collapse ($\mathbf{M}_i \rightarrow \mathbf{0}$) or explosion.

The Diagonal variant uses only the pull and margin terms; its L_1 -normalised output layer implicitly constrains weight magnitudes, playing a role analogous to the explicit regularisation of the Low-rank variant. The margin Δ is scaled with feature dimension, $\Delta = \Delta_{\text{base}} \sqrt{d/d_0}$, to maintain comparable violation magnitudes across feature sets ($d_0 = 27$ is the feature count of the initial tuning round).

2.4 Hyperparameter tuning

Hyperparameters for both variants were selected independently via Bayesian optimisation (Optuna [49]), using a three-phase search that tuned the loss weights (trade-off α , margin Δ , and regularisation strength), optimiser settings (learning rate, weight decay, batch size), and MLP architecture (width factor h and dropout) in sequence. Performance was evaluated by Adjusted Rand Index (ARI) at $K = 30$ clusters; training used early stopping on a held-out 20% split (Fig. 1). Search ranges and all tuned values are listed in Appendix E (Tables 3–5).

2.5 Interpretation of metric feature weights

Because the two variants differ fundamentally in their metric structure—the Low-rank metric produces a $d \times r$ factor matrix, while the Diagonal metric reduces this to d independent per-feature weights—the resulting feature importance scores must be interpreted differently for each. We begin with the Diagonal variant, where the interpretation is more straightforward.

Diagonal variant. The per-feature weights w_{ij} directly quantify how strongly each feature influences the learned distance metric—that is, how much the metric stretches along a given feature axis to separate cells into their Köppen–Geiger zones. Although these weights are related to the ecological relevance of each feature, the relationship is indirect: a high weight means that the feature is discriminative for distinguishing climate zones, not necessarily that it is the most important driver of local vegetation dynamics. Because the weights are L_1 -normalised, they satisfy $\sum_j w_{ij} = 1$ and are directly interpretable as the fraction of total metric sensitivity allocated to feature j at cell i . The *dominant metric feature* per cell is the feature j with the largest weight w_{ij} .

Low-rank variant. Because the Low-rank metric matrix $\mathbf{M}_i$ captures cross-feature interactions, its diagonal entries alone do not fully characterise which features drive the local distance. We therefore analyse the eigenstructure of each cell’s metric. Since each metric matrix $\mathbf{M}_i$ is symmetric and positive semi-definite, it admits an eigendecomposition

$$\mathbf{M}_i = \sum_{k=1}^r \lambda_k^{(i)} \mathbf{v}_k^{(i)} \mathbf{v}_k^{(i)\top}, \quad (4)$$

where r is the maximum rank (here $r=8$), $\lambda_1^{(i)} \geq \dots \geq \lambda_r^{(i)} \geq 0$ are the eigenvalues, and $\mathbf{v}_k^{(i)} \in \mathbb{R}^d$ are the corresponding orthonormal eigenvectors. Geometrically, $\mathbf{M}_i$ defines an ellipsoid in feature space whose principal axes are the eigenvectors and whose axis lengths are proportional to the eigenvalues; the top eigenvector $\mathbf{v}_1^{(i)}$ is the direction along which the metric stretches distances most strongly. We define the *weight profile* of cell i as the squared loadings of the top eigenvector

$$p_j^{(i)} = [v_{1,j}^{(i)}]^2, \quad j = 1, \dots, d, \quad (5)$$

which sums to one and quantifies how strongly each original feature contributes to the dominant metric direction at that cell. A uniform profile ($p_j^{(i)} \approx 1/d$ for all j) indicates that the dominant metric direction is distributed approximately evenly across features.

To identify groups of cells that share similar weight profiles, we cluster the cells based on their Low-rank weight profiles using K -means ($k=12$, selected by silhouette analysis; Appendix G); the resulting clusters are called *weight regimes*. Each regime centroid can be read as a weight-fraction bar chart revealing which features dominate in that region (Supplementary Material Section S6).

Metric isotropy. A learned metric can either concentrate almost all of its sensitivity on a single feature direction or spread it more evenly across many features, and this distinction is ecologically informative: a strongly concentrated (anisotropic) metric indicates that, locally, one or few features dominate the separation of climate

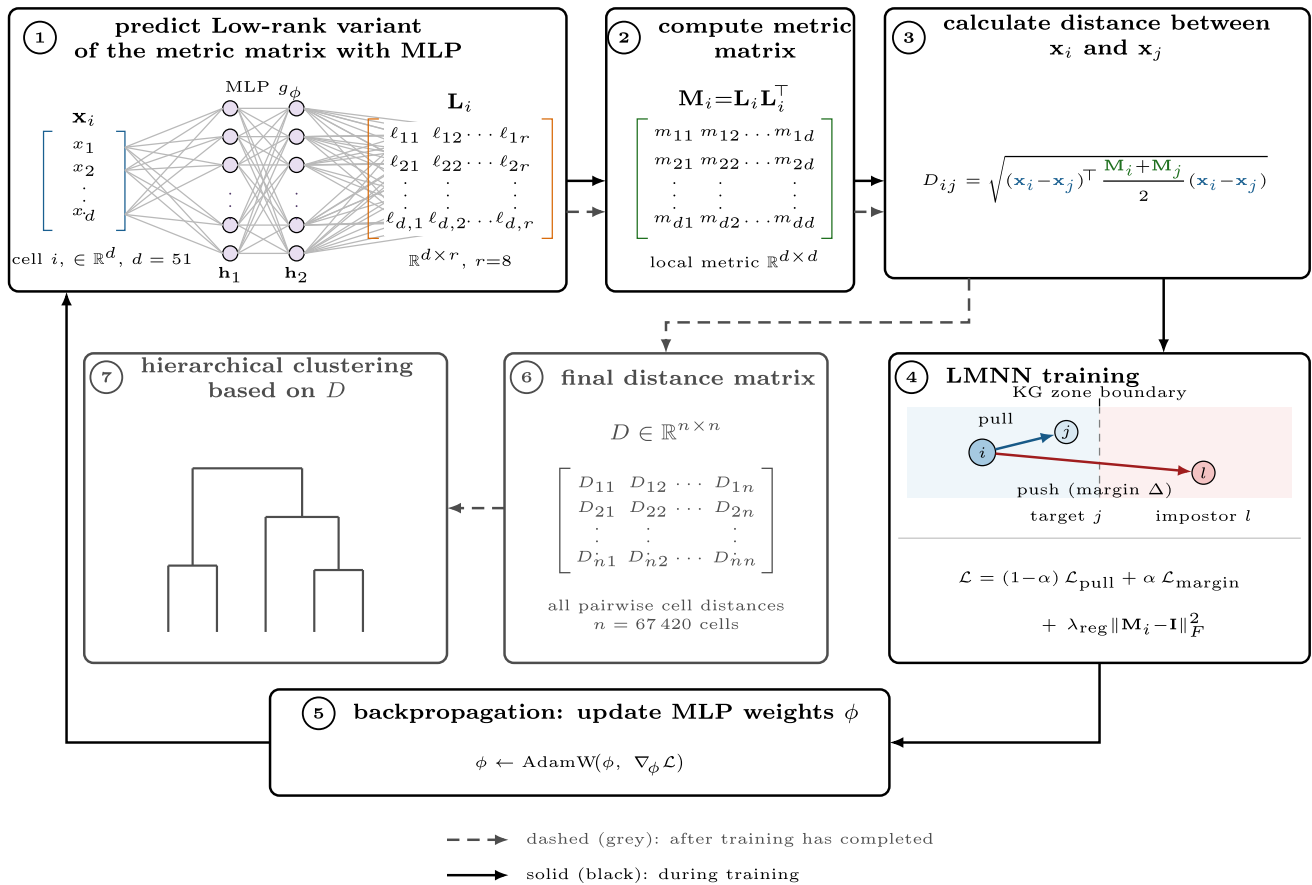


Fig. 1 Schematic illustration of the key steps in the metric learning and clustering process, shown for the Low-rank variant. **(1)** For each cell i , an MLP g_ϕ maps the feature vector $\mathbf{x}_i \in \mathbb{R}^d$ ($d = 51$) to a Low-rank factor matrix $\mathbf{L}_i \in \mathbb{R}^{d \times r}$ with rank $r = 8$. **(2)** The local metric matrix is formed as $\mathbf{M}_i = \mathbf{L}_i \mathbf{L}_i^\top$. **(3)** The distance between two cells is calculated (Eq. (2)). **(4)** Multi-Layer-Perceptron (MLP) parameters ϕ are trained with a large-Margin-Nearest-Neighbour (LMNN) objective that pulls each cell towards its target neighbours and pushes impostors beyond a margin Δ , with regularisation towards the identity metric. **(5)** Parameters are updated by backpropagation (AdamW); steps 1–5 are iterated until performance convergences. **(6)** After convergence, the trained metric is applied to all $n = 67420$ cells to assemble the full pairwise distance matrix $D \in \mathbb{R}^{n \times n}$, on which **(7)** hierarchical clustering is performed. Solid arrows denote operations carried out at the single-cell or single-pair level during training; dashed grey arrows denote operations carried out for all cells after training has completed. For the Diagonal variant, step **(2)** is not required, since the MLP directly outputs the weight vector $\mathbf{w}_i \in \mathbb{R}^d$ and the distance is calculated using Eq. (3)

zones, whereas a more isotropic metric indicates that many features contribute jointly. Quantifying this concentration, therefore, reveals *how* and *where* the metric specialises across the globe, and indicates whether the additional cross-feature capacity of the Low-rank variant is actually being exploited rather than collapsing onto a single direction. The concentration of each cell's metric onto a subset of features is quantified by the *metric isotropy*, defined as the normalised Shannon entropy of the eigenvalue spectrum

$$H_i = - \sum_{j=1}^d \frac{\lambda_j^{(i)}}{\sum_{l=1}^d \lambda_l^{(i)}} \log \frac{\lambda_j^{(i)}}{\sum_{l=1}^d \lambda_l^{(i)}}, \quad (6)$$

where $H_i / \ln(d)$ ranges from 0 (metric dominated by a single direction) to 1 (isotropic metric with all eigenvalues equal). For the Diagonal variant, the per-feature weights serve as the eigenvalues.

2.6 Hierarchical clustering

After metric learning, the full $n \times n$ pairwise distance matrix $\mathbf{D}_{\text{maha}}$ with entries $[\mathbf{D}_{\text{maha}}]_{ij} = D_{\text{maha}}(\mathbf{x}_i, \mathbf{x}_j)$ was computed, using a batched strategy that keeps peak memory independent of n (Supplementary Material Section S4).

Agglomerative (bottom-up) hierarchical clustering was chosen over partitional methods such as k -means for two reasons. First, it produces a full dendrogram from which a partition at any cluster count $K \in [2, n]$ can be extracted by a single cut, without recomputing the clustering for each K ; this is particularly valuable when the optimal compression level is not known *a priori* and must be evaluated across several orders of magnitude in K . Second, the learned local metrics produce a cell-specific $\mathbf{M}_i$ that varies across the globe, so the pairwise distance matrix is the natural input representation; hierarchical clustering operates directly on a precomputed distance matrix, whereas k -means requires a centroid update step that is not straightforward to define for locally varying Mahalanobis metrics. Amongst the standard linkage criteria, complete linkage [50] was applied

$$d_{\text{complete}}(C_1, C_2) = \max_{\mathbf{x} \in C_1, \mathbf{y} \in C_2} D_{\text{maha}}(\mathbf{x}, \mathbf{y}). \quad (7)$$

Complete linkage bounds the maximum within-cluster distance, ensuring that every cell in a cluster is climatically similar to the representative cell—a property well suited to the goal of faithful climate input substitution. Ward linkage was also tested with the same learned distance matrix but without dedicated hyperparameter tuning; complete linkage achieved a higher Adjusted Rand Index (ARI = 0.367 vs. 0.323 at $K=30$; Appendix F), and was therefore retained. In the following, the three distance metrics are referred to as **Euclidean**, **Diagonal**, and **Low-rank**; these labels match the figure legends throughout.

2.7 DGVM simulation and reconstruction-based evaluation

DGVMs simulate vegetation dynamics driven by local climate and soil conditions—computing processes such as photosynthesis, growth, competition, establishment, and mortality to yield ecosystem quantities, including biomass, carbon fluxes, and vegetation composition. In the class of models considered here, each grid cell is simulated independently with no lateral exchanges of carbon, water, or individuals between cells, so a cell's trajectory is fully determined by its own local environmental inputs; two cells with sufficiently similar inputs will therefore produce nearly identical output, which is the property that makes cluster-based cost reduction possible. We evaluate the clustering framework using LPJmL-FIT [4, 5, 9, 19], an individual-based DGVM that simulates trees with continuously varying functional traits. The model requires climate (temperature, precipitation, radiation, atmospheric CO_2 concentration) and soil texture as input and calculates essential ecosystem processes in daily time steps. Vegetation dynamics and vegetation biomass depend on photosynthesis, competition, establishment, and mortality of tree individuals. Detailed model descriptions can be found in earlier publications [4, 5, 9]. Because LPJmL-FIT assigns functional trait values stochastically during tree establishment, its simulation output contains inherent random variability. To quantify this stochastic noise floor and provide a reference against which clustering errors can be benchmarked, two cell-based *baseline* simulations were performed on the full global grid ($n = 67\,420$ cells), each with a different random seed. For both runs a 1000-year spin-up was performed, followed by a 2000–2019 transient simulation, each using 25 patch replicates per cell. All simulations were executed on the PIK HPC [51]; hardware and cost details are given in Appendix H. In practice, substantially more patch replicates are used to better sample stochastic trait variation—e.g., 1000–2500 in Thonicke et al. [5]—which would scale the per-run cost proportionally and further motivate the energy savings achievable through clustering. Seed 1 serves as the baseline reference; the difference between the two seeds quantifies the model-inherent stochastic noise floor against which clustering errors are benchmarked.

In operational use, the clustering pipeline would reduce computational cost by simulating only the representative cell of each cluster with LPJmL-FIT, and assigning the representative cell's output to every cluster member. Because each cluster's representative cell is a member of the original global grid, its simulation output is already available in any existing cell-based baseline run. For evaluation purposes, re-simulating representative cells is therefore unnecessary: one can *reconstruct* cluster-based output fields by assigning each cluster member the baseline output of its representative.

To evaluate the clustering, the dendrogram was cut at each of several cluster counts ($K \in \{7, 21, 67, 213, 674, 2132, 6742, 21320\}$) corresponding to fractions of the total number of grid cells $F = K/n$ (equivalently, energy-saving factor $E = 1/F = n/K$) that are approximately evenly spaced on a logarithmic scale from $F \approx 0.0001$ to $F \approx 0.3$. For each cluster, a single representative cell—the *medoid*—was selected as the cell with minimal sum of pairwise distances to all other cluster members within the precomputed distance matrix of the respective method. Each cluster member was then assigned the medoid's time-series from the baseline simulation, producing a reconstructed output field at the full-grid resolution.

2.8 Evaluation metrics

The clustering pipeline is evaluated using three complementary criteria: (i) *ecological-structure agreement* with the KG supervision signal (Hungarian matching at $K=30$); (ii) *partition geometry* in input and physical space (KG-boundary cluster allocation BAR_{KG} , and within-cluster feature variance $\bar{v}$; a complementary geographic-spread diagnostic $\bar{s}$ is reported in Appendix I); and (iii) *downstream simulation fidelity* (biomass reconstruction accuracy and functional-diversity preservation). Criteria (i) and (ii) diagnose how the metric shapes the partition, whereas criterion (iii) tests whether those partition properties translate into useful reductions of DGVM error.

Hungarian matching against Köppen–Geiger zones. To test whether metric learning recovered the Köppen–Geiger supervision signal, the partition at $K=30$ was compared against the KG reference, which comprises 30 subclasses (Sect. 2.2; Appendix B). Each cluster was mapped to a unique KG zone via the Hungarian algorithm [52], and the *match ratio* reports the fraction of correctly reassigned cells (Supplementary Material Section S5, “Hungarian Zone Matching”).

KG-boundary cluster allocation (BAR_{KG}). To probe where a method concentrates its clusters, we compute for every land cell i its *local cluster diversity* $d_i^{(m,K)}$, defined as the number of distinct cluster labels found in the 5×5 neighbourhood of i (the centre cell plus 2 cells in each direction, $\pm 1.0^\circ$ at the 0.5° grid) divided by the number of valid cells in that window. The same statistic evaluated on the 30-class KG partition yields a reference field b_i^{KG} that is large wherever several KG zones meet. The *Boundary Allocation Ratio at KG boundaries* is the ratio of mean cluster density in the top-25% vs. bottom-25% of cells by b^{KG} value, computed after correcting for cluster size

$$\text{BAR}_{\text{KG}}(m, K) = \frac{\overline{d^{(m,K)}}_{b^{\text{KG-top 25\%}}}}{\overline{d^{(m,K)}}_{b^{\text{KG-bottom 25\%}}}}. \quad (8)$$

A value of 1 corresponds to a spatially uniform cluster allocation; values above 1 indicate that cluster density is concentrated near KG-zone borders; values below 1 indicate cluster density is concentrated in zone interiors. This diagnostic probes how closely each method’s partition geometry is shaped by the KG training signal. A complementary biomass-gradient variant BAR_{BM} and the misallocation gap ΔBAR are deferred to Appendix J; formal definitions are given in Supplementary Material Section S5.

Within-cluster feature variance. The mean within-cluster feature variance measures how homogeneous the clusters are in the original feature space. For each cluster c , the variance is the mean per-feature variance $v_c = \frac{1}{d} \sum_{j=1}^d \text{Var}(x_{.j} \mid c)$ over the d standardised input features, and the global mean is $\bar{v} = \sum_c (n_c/n) v_c$. This metric is reported separately for three feature groups: all 51 features, KG features, and soil features (Supplementary Material Section S5).

Biomass reconstruction error. As a proxy for the accuracy of vegetation dynamics, the absolute per-cell, per-year deviation of reconstructed biomass (kg m^{-2}) from the baseline simulation was computed over the 2000–2019 transient period. An accuracy probability was defined as the fraction of cell $\times$ year pairs for which the absolute biomass error does not exceed a threshold δ

$$P_{\text{acc}}(K) = \frac{1}{nT} \sum_{i=1}^n \sum_{t=1}^T \mathbf{1}(|y_{it}^{\text{cluster}} - y_{it}^{\text{baseline}}| \leq \delta), \quad (9)$$

where δ is set to the 95th percentile of absolute errors in the inter-seed baseline comparison (seed 2 vs. seed 1 at full-grid resolution), so that by construction $P_{\text{acc}} = 0.95$ for the baseline.

Functional diversity (Rao’s Q). Because one of the central motivations for individual-based DGVMs is their ability to represent functional diversity and its role in ecosystem functioning [4, 53], we additionally evaluate Rao’s quadratic entropy [54, 55] Q , computed per grid cell from specific leaf area, wood density, maximum rooting depth, and tree height. Per-cell time-mean Q values from the reconstruction-based outputs were compared against the baseline full-grid simulation; the relative deviation ΔQ quantifies how well the clustering preserves structural and trait diversity (Supplementary Material Section S5, “Rao’s Quadratic Entropy”).

AI-assisted implementation. All code written to produce the results presented in this work—including data preparation, metric learning, hyperparameter tuning, clustering, and data analysis—was developed with the assistance of GitHub Copilot [56] within VS Code. This tool was used to generate, debug, and refine code scripts throughout the implementation. All generated code was reviewed, tested, and validated by the authors, who take full responsibility for its correctness and the results produced.

3 Results

The results are presented in four steps. First, we characterise whether the two LMNN variants recover spatially coherent, ecologically interpretable weight patterns without any explicit spatial information (“Learned metric structure”). Second, we evaluate how those weight patterns translate into cluster partitions at $K = 30$ through match ratios and boundary-allocation statistics against the Köppen–Geiger reference (“Cluster maps and climate zone agreement”). Third, the core performance question—how quickly biomass reconstruction error shrinks as more clusters are added, and at which cluster counts each metric excels—is addressed across five orders of magnitude in the energy-saving factor E (“Reconstruction accuracy across cluster counts”, Sect. 3.3). Finally, we examine where reconstruction errors arise spatially and whether the clustering preserves functional diversity beyond biomass (“Spatial error patterns and functional diversity”, Sect. 3.4).

3.1 Learned metric structure

A central question is whether the learned local metric weights form ecologically meaningful spatial patterns without being given any explicit spatial information. Figure 2 demonstrates that both the Diagonal and Low-rank variants learn spatially smooth, climatically interpretable metric structures across the globe.

Diagonal variant. The dominant-feature map (Fig. 2a) colours each cell by the feature carrying the largest weight w_{ij} . It reveals that temperature features dominate the learned metric across most of the globe. Warmest-month temperature (purple) dominates across the boreal zone and high latitudes, where the D/E threshold controls the forest–tundra boundary [42, 57]; the temperature annual cycle PC1 (red), which captures intra-annual temperature amplitude, dominates across a comparable area spanning mid-latitude and subarctic regions; coldest-month temperature (salmon) dominates across the tropics and subtropics—Amazon, Congo basin, Maritime Continent, and beyond—wherever the 18 °C isotherm provides the primary KG boundary [40–42]; and mean annual temperature (yellow) emerges across the hot arid belt, where the hot/cold B-subtype depends on annual mean thresholds [42]. Together, these four temperature features account for over 80% of all cells. Precipitation and humidity features (blues) become dominant only where moisture-related KG boundaries—the second letter (f, s, w, m) or the arid (B) type—provide the locally strongest contrast, notably in monsoon regions, semi-arid transition zones, and parts of southern Africa and East Asia [36, 42, 58].

Low-rank variant. K -means clustering of cells by their weight profiles (Sect. 2.5; $k=12$, selected by silhouette analysis, Appendix G) partitions the global land surface into 12 spatially coherent *weight regimes* (defined in Sect. 2.5) (Fig. 2b), whose centroid compositions are shown in the stacked-bar chart. The temperature dominance observed in the Diagonal variant also appears in the Low-rank variant: most regimes are temperature-weighted—R1, R2, R3, R4, R6, R8, R10, and R12 show clear temperature-feature dominance in their centroid profiles (Fig. 2b). Three regimes (R5, R7, R9) are precipitation- and moisture-dominated, and R11 shows mixed contributions. The geographic distribution of these regimes closely mirrors the KG zone map (Fig. 3a): temperature-dominated regimes align with the A/C/D/E first-letter zones, moisture-dominated regimes with arid and semi-arid areas (alignment with B first-letter zones), and the mixed regime (R11) with tropical savanna and outer-tropical transition zones where both temperature and precipitation gradients determine KG zone boundaries.

Metric isotropy. The normalised eigenvalue entropy $H_i/\ln(d)$ (Eq. 6; Fig. 2c, d) quantifies how concentrated each cell’s metric is onto a subset of features. Where isotropy is low, the dominant feature or eigenvector accounts for a large share of total metric sensitivity; where it is high, several features contribute comparably. The Diagonal metric (Fig. 2c) exhibits generally lower isotropy than the Low-rank metric (Fig. 2d), indicating that the independent per-feature weighting tends to concentrate sensitivity onto fewer features, whereas the full-matrix parameterisation distributes sensitivity more broadly through cross-feature interactions. Both variants show clear spatial structure in isotropy, with particularly low values in polar regions and high-elevation areas, confirming that the degree of metric concentration is itself an ecologically interpretable quantity that varies with climate context.

3.2 Cluster maps and climate-zone agreement

At $K = 30$ (the number of KG zones), the three distance metrics produce qualitatively different cluster partitions (Fig. 3c, f, i). Euclidean clustering struggles to differentiate the continental climate zones of the high northern latitudes: it assigns the vast majority of boreal and subarctic cells to a single large Dfc cluster, leaving almost no cells for the remaining snow-climate subtypes (Dfa, Dfb, Dfd) and virtually none for the monsoon-influenced Dw group. It also substantially overestimates and misplaces the hot semi-arid zone (BSH), extending it into regions where the KG classification assigns temperate or tropical types. These errors are reflected in a low overall match ratio of 28.6% after Hungarian label assignment (Fig. 3d), with mismatches distributed broadly across biomes rather than confined to gradual transition zones.

The Diagonal and Low-rank metrics recover substantially more of the KG partition, achieving match ratios of 64.6% and 65.2%, respectively (Fig. 3g, j). Both learned metrics produce visually similar cluster maps that

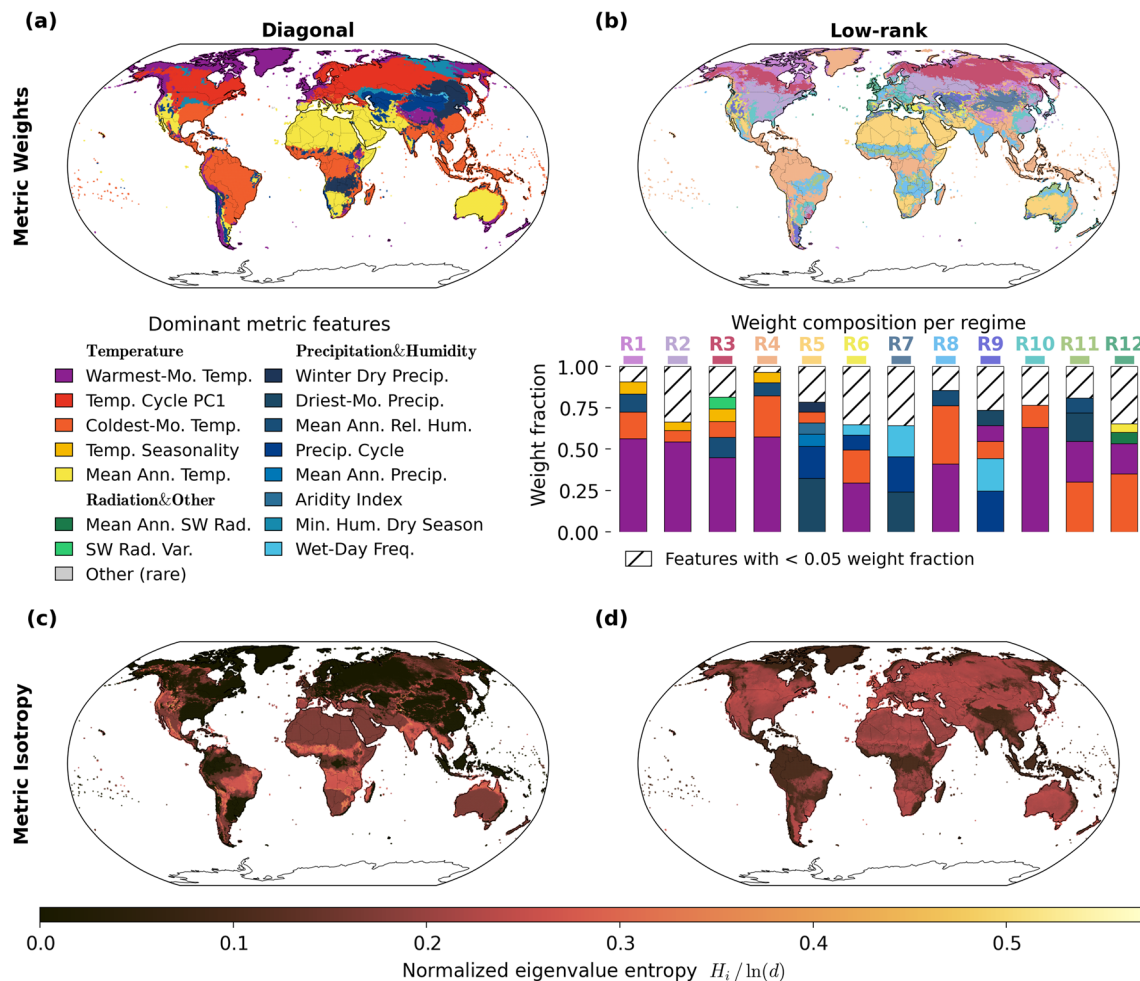


Fig. 2 Spatial patterns of feature dominance in the learned Diagonal and Low-rank metric weights. **(a)** Dominant metric feature per cell under the Diagonal variant, defined as the feature j with the largest weight w_{ij} ($\sum_k w_{ik} = 1$); features that are either dominant in at least 1% of all cells or contribute $\geq 5\%$ in any regime centroid receive a distinct colour; rarer dominant features are shown in grey. **(b)** Spatial distribution of the 12 Low-rank weight regimes obtained by K -means clustering of cells by their top eigenvector loading profiles (Sect. 2.5). The stacked-bar chart shows the feature composition of each regime centroid, with bar colours identifying features as in the legend of panel (a); only features contributing $\geq 5\%$ of the centroid weight are shown individually, with the hatched remainder representing the aggregate of all features below that threshold. Both panels show which input features drive the learned distance metric locally: panel (a) identifies the single most influential feature per cell under the Diagonal variant, while panel (b) captures the full dominant-direction composition under the Low-rank variant—grouping cells by the feature mix of their top eigenvector and summarising each group’s composition in the stacked-bar chart. **(c)** Diagonal and **(d)** Low-rank metric isotropy, quantified by the normalised eigenvalue entropy $H_i / \ln(d)$ (Eq. 6), where $d = 51$; both panels share a common colour scale. Values near 0 indicate a metric dominated by a single direction; values near 1 indicate an isotropic metric with all eigenvalues equal

closely resemble the KG reference (Fig. 3a), and their remaining mismatches concentrate along transition zones between major climate regions—boundaries where the KG classification itself represents a sharp discretisation of continuous climate gradients. Although the two learned metrics agree broadly, they exhibit numerous localised differences from each other and from the KG reference:

Most notably, neither of the metrics captures the tropical monsoon zone Am well; the Low-rank variant assigns extensive areas of Cfa (humid subtropical) where the KG classification maps Aw (tropical savanna); the Diagonal variant tends to expand Af (tropical rainforest) at the expense of Am and Aw; and in southern Africa, the Diagonal metric incorrectly assigns ET (tundra) to cells that should be BSh (hot semi-arid).

Two quantitative contrasts between the Euclidean and the learned metrics stand out in Fig. 3: the allocation of clusters to KG-zone boundaries (panel b) and the decomposition of within-cluster feature variance across feature groups (panels e, h, k).

The KG-boundary cluster allocation BAR_{KG} (Fig. 3b) exposes how directly each method's partition geometry is shaped by the KG training signal. The Diagonal and Low-rank metrics concentrate 2–3 times as much cluster density near KG-zone boundaries as in zone interiors, peaking at $\text{BAR}_{\text{KG}} \approx 3.0$ for Diagonal at $K = 67$ and ≈ 2.8 for Low-rank at $K = 30$ – 67 , before contracting towards unity as the cluster budget saturates the map at $K \gtrsim 6,720$. Euclidean clustering has no such signal and stays at $\text{BAR}_{\text{KG}} \approx 1.4$ – 1.6 throughout, driven only by the spatial autocorrelation of the raw features. This is a direct fingerprint of the LMNN objective: the learned metrics shrink within-zone distances and stretch across-zone distances, so complete-linkage places the earliest clusters astride the KG borders. Applying the same quartile contrast used for KG-boundary proximity to a biomass-gradient reference field yields a second allocation ratio BAR_{BM} (Appendix J, Fig. 8b): Euclidean reaches $\text{BAR}_{\text{BM}} \approx 2.0$ at $K \approx 213$ – 674 and stays above both learned metrics for all $K \geq 67$, whereas Diagonal and Low-rank plateau near 1.4–1.6.

The within-cluster feature variance curves (Fig. 3e, h, k) reveal qualitatively different behaviour across metrics. For Euclidean (e), the three feature subsets start close together near 0.8–0.9, remain relatively close across most of the K range with some separation at intermediate K , and all converge near zero at large K . For the learned metrics (h, k), the curves diverge substantially: KG-diagnostic variance starts around 0.5 and decays towards zero

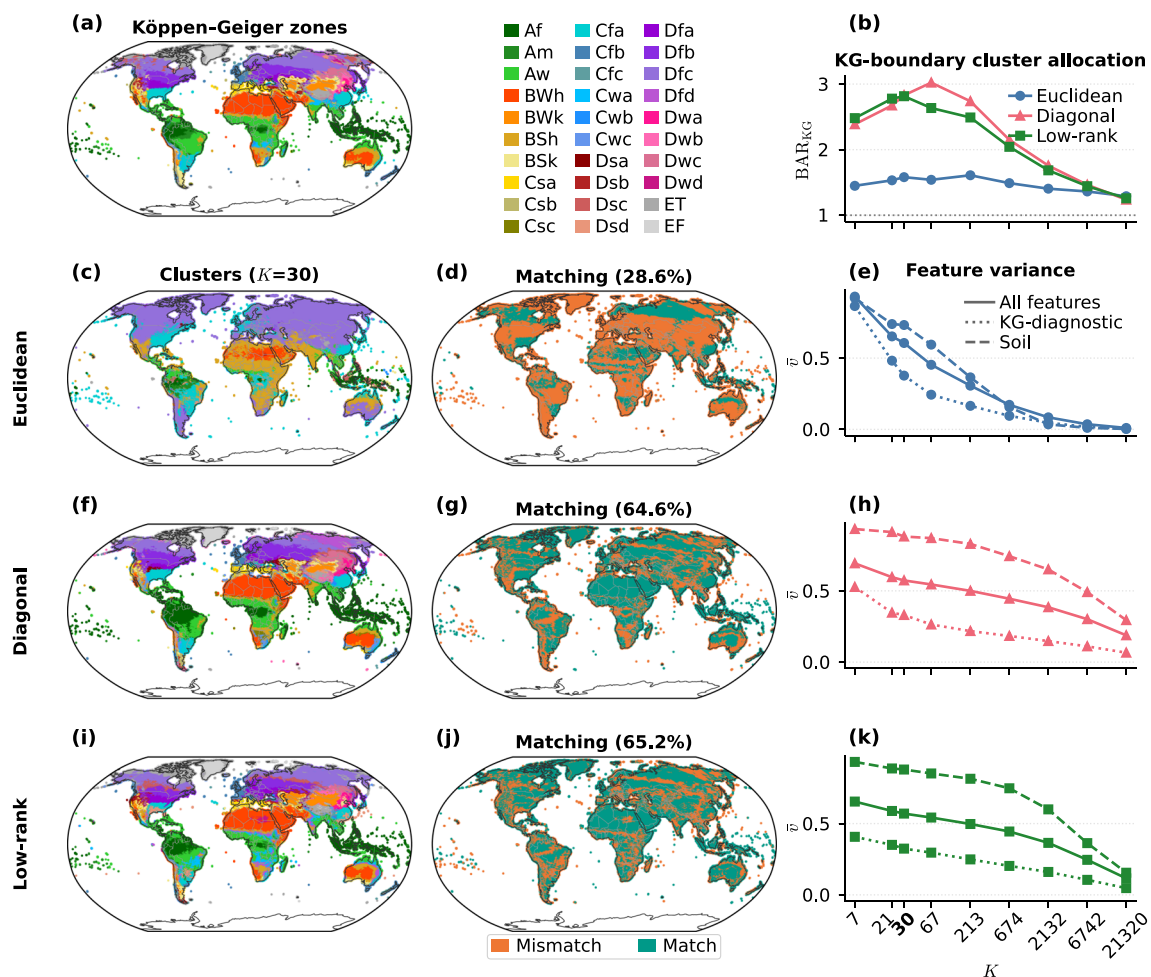


Fig. 3 Clustering results, KG-boundary cluster allocation, and within-cluster feature variance for Euclidean, Diagonal, and Low-rank distance metrics. (a) Reference map of the 30 Köppen-Geiger (KG) climate zones. (b) KG-boundary cluster allocation BAR_{KG} (Eq. 8): ratio of mean local cluster diversity in the top 25 % of cells by KG-boundary proximity to the mean in the bottom 25 %, as a function of the number of clusters K ; the dotted line at 1 marks a spatially uniform allocation. (c, f, i) Cluster maps at $K=30$ after Hungarian matching to KG zones for Euclidean, Diagonal, and Low-rank, respectively. (d, g, j) Spatial agreement with the KG reference (teal: match; orange: mismatch), with the match ratio shown in the panel title. (e, h, k) Mean within-cluster feature variance ($\bar{v}$, cluster-size-weighted) as a function of K , shown separately for all 51 features (solid), KG-diagnostic features (dotted), and soil features (dashed). Row labels on the left distinguish the three distance metrics. Panel (b) quantifies how strongly each method's partition geometry is anchored to the KG training signal, while panels (e, h, k) show how quickly each method achieves within-cluster homogeneity

with uniform decay rate (Low-rank) or steeper decay at low than at high K (Diagonal), while soil variance starts near 0.8 and decreases slowly at first before falling more steeply at high K , remaining above 0.2 even at large K ; the all-features curve lies between the two throughout. At $K = 2132$, for example, Euclidean yields a total within-cluster variance of ~ 0.10 (soil ~ 0.05), whereas the learned metrics reach 0.30–0.40 total (0.55–0.65 soil), indicating that the learned metrics concentrate feature weights heavily in the KG-diagnostic features while leaving soil features comparatively down-weighted.

3.3 Reconstruction accuracy

Each full baseline simulation (Sect. 2.7) requires a 1000-year spin-up ($\approx 13\,300$ CPU-hours; $\approx \text{€ } 290$; ≈ 237 kWh) and a 20-year transient run ($\approx 1\,150$ CPU-hours; $\approx \text{€ } 25$; ≈ 20 kWh) per seed (Appendix H); the inter-seed biomass differences establish the model-inherent noise floor against which all clustering errors are benchmarked below.

To determine the accuracy in reproducing global vegetation biomass distribution, we compare the deviation of each reconstructed cluster-based output from the baseline, and assess how quickly this deviation shrinks as more clusters are added (Fig. 4). The error distribution (Fig. 4a) and the accuracy probability $P(|\text{err}| \leq \delta)$ (Fig. 4b; Eq. 9) provide two complementary views: the boxplots characterise the full spread and central tendency of per-cell, per-year biomass deviations, while P quantifies the fraction of cell $\times$ year pairs whose absolute error stays within $\delta = 6.82 \text{ kg m}^{-2}$, the 95th percentile of the inter-seed error distribution.

At low cluster fractions ($F \leq 0.003$, corresponding to energy-saving factors $E \geq 333$), both learned metrics clearly outperform Euclidean clustering in both views: median biomass deviations are closer to zero, interquartile ranges are substantially narrower, and accuracy probabilities are markedly higher (e.g. $P \approx 0.77$ for Diagonal and $P \approx 0.71$ for Low-rank versus $P \approx 0.39$ for Euclidean at $F = 0.0001$). As F increases, all three methods converge; at high F ($0.01 \leq F \leq 0.3$), Euclidean achieves equal or higher P than the learned metrics, although the latter retain narrower interquartile ranges, indicating fewer extreme outliers. At $F = 0.3$, all three methods have an accuracy probability above 0.9 and Euclidean even exceeds the nominal baseline value of $P = 0.95$. This is possible, because the reconstruction uses the same random seed as the baseline it is compared against, eliminating stochastic mismatch; the inter-seed comparison underlying the baseline, by contrast, includes an irreducible noise component from independent random initialisation. For the same reason, all three methods at $F = 0.3$ produce error distributions in panel (a) that are narrower than the baseline noise floor.

Low-rank and Diagonal clustering perform similarly across most of the F range, with Diagonal achieving slightly higher P at the lowest cluster counts, confirming that learned per-feature weighting captures the beneficial structure; the additional cross-feature interactions in the Low-rank metric do not provide further improvement for biomass reconstruction in this experiment.

3.4 Spatial error patterns and functional diversity

Beyond aggregate error statistics, it is important to understand *where* reconstruction errors arise spatially. Figure 5 shows the per-cell relative biomass deviation for three representative clustering configurations chosen to illustrate all three distance metrics across a wide range of F (cf. Fig. 4a, b): the Diagonal metric at $F = 0.001$ ($K = 67$, $E = 1000$), where it performs best; the Low-rank metric at $F = 0.01$ ($K = 674$, $E = 100$), where Diagonal and Low-rank perform equally; and the Euclidean metric at $F = 0.1$ ($K = 6742$, $E = 10$), where it performs best. Relative deviations are largest where baseline biomass is small: semi-arid and polar interiors and transition zones between high-biomass vegetation (tropical or boreal forest) and adjacent low-biomass vegetation (savanna, steppe, tundra). Relative deviation maps for all cluster counts and all methods are provided in Appendix K; the corresponding absolute deviation maps are in Appendix L; and functional-diversity maps are in Appendix M.

At $E = 1000$ (Fig. 5a, b) the Diagonal metric reproduces the broad global biomass pattern, but percentage deviations exceeding $\pm 50\%$ appear across large parts of the tropics, semi-arid regions, and high-latitude forests, concentrated at transitions between high-biomass and low-biomass vegetation where steep baseline gradients amplify any feature mismatch between a cell and its representative. At $E = 100$ (Fig. 5c, d), deviations are substantially reduced, with strong residual signals confined to localised transition zones between major climate regions. At $E = 10$ (Fig. 5e, f), the deviation map is largely within $\pm 10\%$, visually approaching the cell-based baseline (Fig. 5g), though residual errors persist along KG zone boundaries. Across all three cluster counts, relative deviations from the baseline are highest in zones of steep biomass gradients, e.g., the transition between tropical rainforest and adjacent savanna, or the transition between boreal forest and adjacent tundra, where any mismatch between a cell and its representative incurs a large relative error. In the case of absolute deviations, shown in Appendix L, larger deviations also occur inside the high-biomass regions, reflecting the large biomass stocks in these regions.

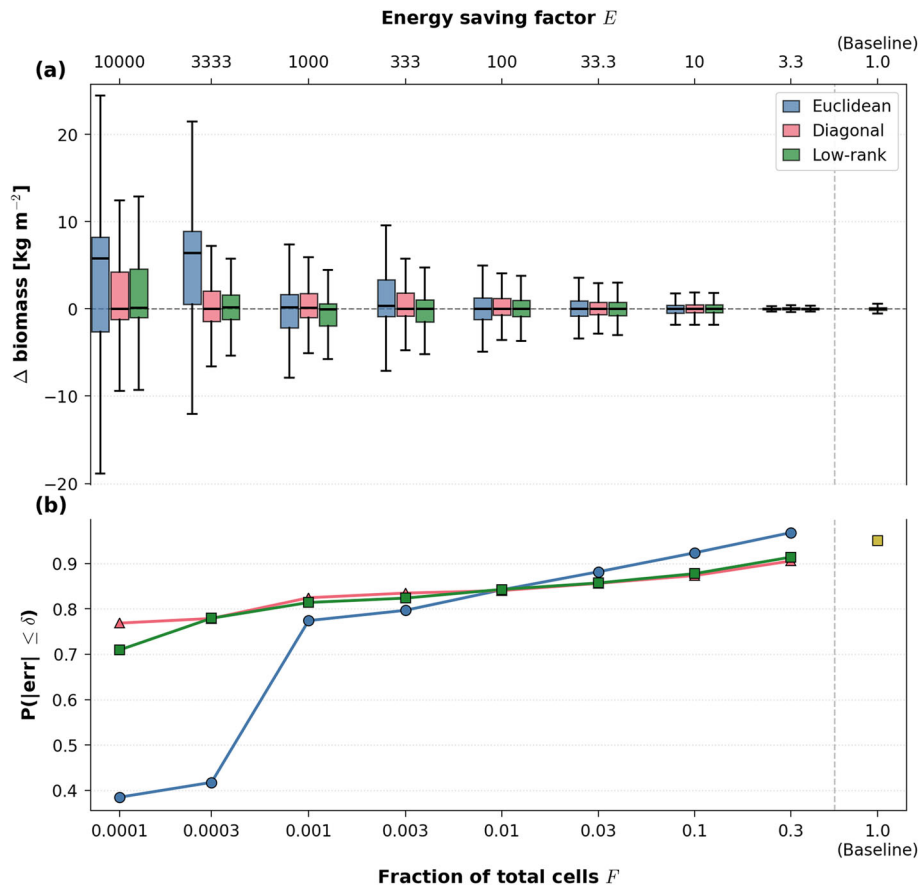


Fig. 4 Biomass reconstruction error across cluster counts for Euclidean, Diagonal, and Low-rank clustering. The horizontal axis shows the cluster fraction $F = K/n$ and equivalent energy-saving factor $E = 1/F$. **a** Grouped boxplots of per-cell, per-year absolute biomass deviations (reconstructed minus baseline, kg m⁻²) for Euclidean, Diagonal, and Low-rank clustering across all tested values of K , showing that median errors decrease and spread narrows as more clusters are used, with all three methods approaching the baseline noise floor at large F . The rightmost box (Baseline) shows the inter-seed baseline deviation (seed 2 vs. seed 1 at full resolution), representing the model-inherent noise floor against which clustering errors are benchmarked. **b** Accuracy probability $P(|\text{err}| \leq \delta)$: the fraction of cell×year pairs whose absolute biomass error does not exceed the threshold δ , defined as the 95th percentile of absolute baseline errors, so that $P = 0.95$ for the baseline by construction. The baseline value is shown on the right

While biomass is a key state variable that integrates vegetation growth and mortality, individual-based DGVMs such as LPJmL-FIT can provide more information on vegetation dynamics, such as vegetation diversity, composition, and structure—and any operational clustering should preserve these higher-order properties. We therefore additionally evaluate Rao's quadratic entropy Q , a measure of functional and structural diversity of the simulated tree communities, computed per grid cell from the key functional traits specific leaf area, wood density, and maximum rooting depth, together with tree height as a structural-diversity variable (Fig. 5i). The ΔQ deviations follow a convergence pattern similar to the biomass errors: at low cluster counts ($F \leq 0.001$), median deviations and interquartile ranges are substantial, with spreads of ± 25 –50 % for the same reason that drives large relative biomass errors—when a representative cell from one vegetation type is assigned to a cluster member in a structurally different biome, both biomass and Q shift considerably. As F increases, ΔQ narrows steadily, and at $F = 0.3$ the cluster-induced deviations fall below the inter-seed baseline noise floor. This sub-baseline behaviour at high F is expected, because reconstruction assigns each cluster member the output of its representative from baseline 1 and compares against baseline 1 (identical seed), whereas the noise floor is defined by comparing baseline 1 against baseline 2 (independent seeds), so the cluster-based evaluation is not subject to the additional stochastic mismatch between seeds. Maps of Q and ΔQ are provided in Appendix M.

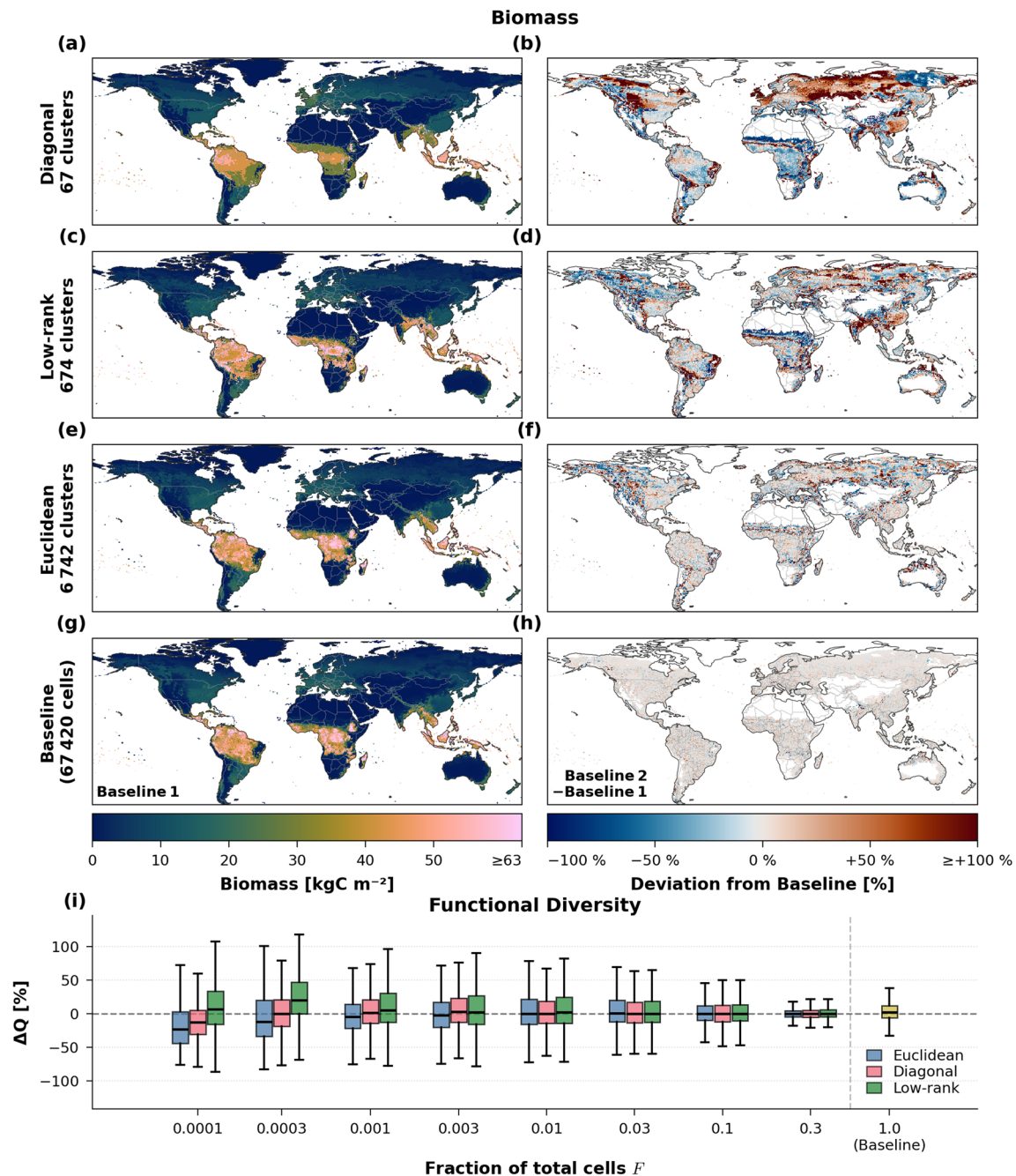


Fig. 5 Maps of reconstructed biomass, spatial distribution of relative error, and functional diversity at selected cluster counts. **a, b** Diagonal metric at $K = 67$ ($E = 1000$): time-mean biomass [kgC m⁻²] and percentage deviation from baseline. **c, d** Low-rank metric at $K = 674$ ($E = 100$). **e, f** Euclidean baseline at $K = 6742$ ($E = 10$). Deviation maps (right column) use a diverging scale capped at $\pm 100\%$. Panels **g** and **h** show the cell-based baseline itself (Seed 1; 67 420 cells) and the percentage deviation of an independent cell-based simulation with a different random seed (Seed 2 minus Seed 1), respectively; this seed-to-seed difference quantifies the model-inherent stochastic noise floor for the map deviations. The biomass colour scale is capped at the 99th percentile (≥ 63 kgC m⁻²); values above this threshold are shown in the top colour. **i** Boxplots of the relative deviation ΔQ [%] of Rao's quadratic entropy (Supplementary S4) between reconstruction-based and cell-based outputs, across all cell $\times$ year pairs, as a function of the fraction of total grid cells $F = K/67\,420$. Rao's Q is computed per grid cell from specific leaf area, wood density, and maximal rooting depth as plant functional traits, and tree height as a simulated structural-diversity variable. The rightmost box ($F = 1.0$, Baseline) shows ΔQ between the two independent cell-based seeds (Seed 2 minus Seed 1), providing the stochastic noise floor against which cluster-induced deviations should be evaluated

4 Discussion

We set out to test whether spatially adaptive, KG-supervised distance metrics produce ecologically meaningful clusters of DGVM input features at multiple resolutions and whether these clusters accurately reconstruct the vegetation state of the full-grid simulation. Three metrics were compared at cluster counts spanning five orders of magnitude: a uniform **Euclidean** metric (the standard choice in prior clustering-based cost-reduction work), and two locally adaptive LMNN variants—**Diagonal** (per-feature weights only) and **Low-rank** (with cross-feature interactions).

This section is organised around three questions raised by the results: (i) how distance-metric choice shapes clustering quality and where its advantage is bounded by the scale of the supervision signal (Sect. 4.1); (ii) whether the additional model complexity of the Low-rank variant is warranted given the KG training signal (Sect. 4.2); (iii) the energy-saving potential versus the one-off cost of metric learning (Sect. 4.3); and finally the scope, limitations, and implications for future work (Sect. 4.4).

4.1 Metric choice, error sources, and the scale of the supervision signal

Prior clustering-based cost-reduction approaches in land surface and vegetation modelling [31, 33–35] rely on globally uniform distance metrics with equal or heuristically chosen feature weights. Our results show that locally adapted feature weights drastically improve global clustering results for cluster numbers < 674 compared to the baseline with equal weights for all features everywhere. At $K = 30$, the Diagonal and Low-rank variants recover 64.6 % and 65.2 % of the KG reference partition, against 28.6 % for Euclidean (Fig. 3d, g, j), and the same advantage persists in reconstruction accuracy across two orders of magnitude ($K \lesssim 674$, $E \gtrsim 100$; Fig. 4). The mechanism behind this advantage is made explicit by the KG-boundary cluster allocation BAR_{KG} (Fig. 3b): the learned metrics concentrate 2–3 times as much cluster density per unit area on KG-zone boundaries as on zone interiors, peaking at $\text{BAR}_{\text{KG}} \approx 3.0$ at $K = 30$ –67, whereas Euclidean remains at $\text{BAR}_{\text{KG}} \approx 1.4$ –1.6 throughout. The learned metrics therefore encode the KG training signal directly into the partition geometry, at exactly the spatial grain where that signal is informative.

This same mechanism, however, is also the main source of errors in the reconstruction of simulated biomass. KG borders do not necessarily align with the biomass gradients simulated by DGVMs (Figs. 3a, 5g); consequently, concentrating clusters near KG boundaries places the available resolution where it is less needed — at zone interiors—rather than at the steep biomass gradients where it would reduce reconstruction error most. The ratio of cluster diversity close to biomass gradients to cluster diversity away from biomass gradients BAR_{BM} confirms that this misallocation is much stronger for the learned metrics compared to the Euclidean one (Appendix J, Fig. 8b). Despite this misallocation, the learned variants achieve high accuracies already at low K , clearly outperforming the Euclidean variant for $K \lesssim 674$. By concentrating all metric sensitivity onto the features that define KG boundaries—warmest-month and coldest-month temperature, and related annual cycle statistics (Fig. 2a)—LMNN keeps cells of the same broad biome type together even when they are geographically far apart, producing internally homogeneous large clusters that Euclidean, with equal feature weights, cannot replicate. At higher cluster counts, however, this concentration of metric sensitivity onto broad climate features—and the consequent downweighting of local variables such as soil texture (Fig. 3h, k)—becomes a disadvantage as it is no longer appropriate once within-biome heterogeneity drives the remaining error. The mean geographic spread $\bar{s}$ (Appendix I, Fig. 7) quantifies this: the learned metrics retain 20–40° geographic spread even at $K = 21\,320$, whereas Euclidean converges to compact, near-contiguous clusters ($\bar{s} \approx 3^\circ$ at the same K), because it does not suppress local feature dimensions. Near $K \approx 674$, the geographic spread $\bar{s}$ of Euclidean clusters falls below that of both learned metrics—precisely where the Euclidean accuracy probability rises above that of both learned metrics (Fig. 4b)—indicating that this difference in geographic spread is a key factor in the relative performance of the metrics. The operational recommendation is therefore simple: use KG-supervised metrics in the high-saving regime ($E \geq 100$), where the training signal is aligned with the partition grain; at lower savings, the uniform metric is preferable. Extending the advantage to higher K would require a finer supervision signal, e.g., ecoregion classifications with 800 or more classes [59, 60]. Additional analysis of mechanisms leading to reconstruction errors is provided in Appendix J.

4.2 Diagonal versus Low-rank: interpretability and overfitting

The Diagonal and Low-rank variants achieve nearly identical match ratios with Köppen–Geiger zones at $K = 30$ (64.6 % vs. 65.2 %) and similar biomass reconstruction accuracy across the full range of K (Fig. 4). Under the KG supervision signal, per-feature weighting already captures the temperature and moisture axes that define the KG first-letter boundaries (Fig. 2), so the additional cross-feature interactions of the Low-rank parameterisation add little discriminative power for the error-dominating population identified in Sect. 4.1. The Diagonal variant is

therefore recommended as the default for operational use under KG-based supervision, with two practical advantages: its learned weights are directly interpretable as per-feature weight maps (Fig. 2a), and it has a smaller parameter count and lower overfitting risk than the Low-rank variant due to its smaller number of trained parameters. The Low-rank formulation remains valuable with finer supervision signals where discriminating between ecologically similar zones may require cross-feature interactions that the Diagonal parameterisation cannot represent, and where the 12 spatially coherent weight regimes recovered by eigendecomposition of the Low-rank matrices (Fig. 2b) suggest the capacity to carry such sub-biome structure.

4.3 Energy consumption reduction potential and metric learning cost

The recommended Diagonal metric learning pipeline requires ~ 10 kWh for full tuning and training, or ~ 2 GPU-hours for production training alone when reusing previously tuned hyperparameters (full breakdown in Appendix H). This is less than 5 % of the ≈ 237 kWh consumed by a single 25-patch LPJmL-FIT spin-up (Sect. 3) and negligible relative to the simulation energy it enables to save. Crucially, tuning depends on the feature set and model architecture, not on the climate data, so it needs to be performed only once and transfers directly to new climate scenarios, future projections, or higher-resolution grids using the same features. Only the ~ 2 GPU-hour production training needs to be repeated per dataset. The learned distance matrix itself supports hierarchical clustering at any K without retraining, so the cost of switching cluster counts is effectively zero.

Translated into simulation savings: research-grade LPJmL-FIT configurations use up to 2500 patch replicates [5], $100\times$ the 25-patch baseline used here. At $E = 1000$ ($K = 67$), where more than 80 % of cell-year pairs already stay within the inter-seed noise floor (Fig. 4b), the combined spin-up and 20-year transient simulation cost falls from approximately € 31 500 (25 700 kWh) to € 31.5 (25.7 kWh) for a 2 500-patch configuration—a reduction of 99.9 % in both cost and energy. These savings compound across every additional ensemble member, climate scenario, or parameter sweep, and are therefore particularly useful in model development, where full accuracy is often less critical than in production runs for published research. Calibration runs, benchmarking experiments, and regression testing all involve many repeated global-scale simulations, for which energy use can be reduced by three orders of magnitude with only modest loss of accuracy.

4.4 Scope, limitations, and future directions

This paper introduces the methodological framework and provides a proof-of-concept demonstration using LPJmL-FIT as an example that can be applied to other models with independent cells as well. Nevertheless, several avenues for further development remain. The current feature set of 51 variables spans the major climate, radiation, and soil dimensions relevant to global vegetation, but the optimal number and selection of features for the metric learning objective has not been systematically explored. Different feature sets, dimensionality reduction strategies, and feature engineering choices may yield improved clustering quality, particularly when combined with alternative supervision signals. The Köppen–Geiger classification with 30 zones was chosen, because it provides a well-established, physically interpretable partition at a coarse scale where reduction in computational cost is particularly high, yet the framework is equally compatible with finer or ecologically richer schemes. Ecoregion classifications with more than 800 regions [59, 60] would provide substantially more detailed supervision, potentially improving cluster quality at high resolutions and making the Low-rank metric’s capacity for cross-feature interactions more valuable. Other options include learning from existing biomass or vegetation maps, extending the training to higher cluster numbers, or combining multiple classification schemes; that the metric trained on only 30 classes generalises effectively across orders of magnitude in K suggests that finer supervision could yield further improvements. The method assumes that grid cells are simulated independently without lateral fluxes of carbon, water, or individuals between cells—an assumption that holds for LPJmL-FIT and many other DGVMs simulating unmanaged vegetation [2, 3, 10, 11] but would need modification for models with explicit spatial interaction, such as landscape-scale models with seed dispersal, wildfire spreading or hydrological routing. Adapting the approach to another model is largely a matter of input and output handling: the existing gridded input data are clustered, only the representative cell of each cluster is simulated, and cluster members are assigned their representative’s output to reconstruct a full grid. The main practical challenge is the model’s grid format. Models accepting an arbitrary list of cells are the most straightforward, while models that require a regular-grid format need either masking non-representative cells or placing the representatives on a new regular grid.

The learned metric can be reused for any vegetation model satisfying the same condition of independent grid cells and for any desired number of clusters. For climate projections, the Köppen–Geiger classification can be recomputed from the projected climate data and the metric retrained. Because the KG boundaries and the clustering are derived from the *same* climate data, any variability in the underlying climate, e.g., due to climate change, propagates consistently to the supervision signal for metric learning, the clustering, and the model output. However, the more climate changes in the simulated time-period, the less accurate are the summarising features used for KG classification and clustering regarding the representation of the climate and the climate-zone boundaries during

that period, increasing the error of the metric learning and clustering. To minimise the influence of inter-period climate variability, the KG classification, the clustering, and the model output can be computed over shorter, matching time windows.

For ensemble runs, a small set of representative ensemble members could be trained once and the corresponding metrics reused across ensembles. Whether retraining is necessary at all depends on how much the Köppen–Geiger boundaries shift under the projected climate. If the shift in zone assignments is small relative to the method’s inherent uncertainty—as discussed in Sect. 4.1—retraining would not yield a meaningful improvement and the existing distance matrix can be reused directly.

The choice of cluster count and distance metric should be guided by the research question. Studies that require only aggregated outputs—such as global or biome-level totals of biomass, carbon fluxes, or productivity—could operate at low K with learned metrics, where the reconstruction captures broad spatial patterns and the energy savings are largest. When spatially resolved accuracy matters—for example in regional impact assessments or biodiversity analyses that depend on local vegetation structure—higher K with the Euclidean metric offers the best per-cell fidelity (Sect. 4.1).

Future work will explore finer supervision signals, optimal feature selection, and interpolation methods for transition zones, and will deliver a ready-to-use software package automating the full pipeline and providing guidelines for choosing the cluster count for specific applications.

5 Conclusion

Global simulations with biodiversity-enhanced DGVMs are computationally expensive, but many of the tens of thousands of land grid cells share similar environmental conditions.

Exploiting this redundancy by simulating only a representative cell per cluster and assigning its output to all cluster members can reduce cost by orders of magnitude. To find suitable clusters for this purpose, we developed a two-stage pipeline: a local LMNN framework learns cell-specific distance metrics supervised by Köppen–Geiger climate-zone boundaries, followed by hierarchical clustering of all land cells using the learned distance metrics. This produces a single dendrogram from which partitions at any cluster count K can be extracted without retraining. Because the framework requires only *spatially independent* grid cells—that is, cells whose DGVM state evolves without lateral exchanges of carbon, water, or individuals—it is directly applicable to any DGVM of this class.

The learned metrics substantially outperform the uniform Euclidean baseline in the high-saving regime ($E \geq 100$, $K \leq 674$): at $E = 1000$ ($K = 67$), more than 80 % of cell–year pairs already stay within the model-inherent stochastic noise floor, and functional diversity follows the same convergence pattern. At lower saving factors, i.e., at higher spatial resolution, where locally important features like soil texture—which are down-weighted in metric learning on the coarse KG zones—drive remaining errors, the Euclidean metric performs equally well or better.

Of the two learned variants, the Diagonal metric—which learns per-feature weights without cross-feature interactions—matches the Low-rank variant in reconstruction accuracy while providing more easily interpretable feature weights and being computationally lighter; it is the recommended default for KG-based supervision. The Low-rank formulation retains value as a more expressive alternative that may prove advantageous with finer supervision signals such as ecoregion classifications.

The practical benefits of our method are substantial. Evaluated with the DGVM LPJmL-FIT, more than 80% of cell–year pairs already stay within the 95th-percentile threshold of the model-inherent stochastic noise floor at $E = 1000$ ($K = 67$, consuming only 0.1 % of full-grid energy). Assuming global model runs with LPJmL-FIT were performed with the same number of patches used for regional studies, clustering at $E = 1000$ could save up to 25 000 kWh and € 31 000 for model spin-up alone.

This reduction in energy and cost not only reduces the carbon footprint of global-scale DGVM research, but also helps remove a major infrastructure barrier for research groups worldwide and democratises access to global-scale DGVM research.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1140/epjs/s11734-026-02526-1>.

Acknowledgements The authors gratefully acknowledge the Ministry of Research, Science and Culture (MWFK) of Land Brandenburg for supporting this project by providing resources on the high-performance computer system at the Potsdam Institute for Climate Impact Research and thank Boris Sakschewski for the acquisition of the funding by the Fachagentur Nachwachsende Rohstoffe (FNR) (Grant No. 2219WK39A4).

Author contribution statement

Conceptualization: JP; **Methodology:** JP, CS, DH, and MG; **Software:** JP; **Validation:** JP; **Formal analysis:** JP and DH; **Investigation:** JP; **Data curation:** JP; **Visualisation:** JP, BT, MB, DH, KT, and FS; **Writing – original draft:** JP; **Writing – review & editing:** JP, FS, CS, KT, BT, MB, MG, and DH; **Supervision:** KT, CS, MG, and BT; **Funding acquisition:** FS.

Funding Open Access funding enabled and organized by Projekt DEAL. This research has been supported by the FORMAS project ReForMit, the Fachagentur Nachwachsende Rohstoffe (FNR) project Waldspektrum (Grant No. 2219WK39A4), the European Union’s Horizon Europe research and innovation programme ClimTip (Grant Agreement No. 101137601), and the Deutsche Bundesstiftung Umwelt (DBU) scholarship programme. Maik Billing was funded by the Planetary Boundaries Science Lab’s research effort at the Potsdam Institute of Climate Impact Research (PIK).

Data Availability The atmospheric climate input data used in this study are openly available through the Inter-Sectoral Impact Model Intercomparison Project (ISIMIP) repository under a CC0 1.0 Public Domain Dedication: Lange, S., Mengel, M., Treu, S., & Büchner, M. (2023). ISIMIP3a atmospheric climate input data (v1.2). ISIMIP Repository. <https://doi.org/10.48364/ISIMIP.982724.2> Soil depth data were obtained from the global gridded dataset of Pelletier, J.D., Broxton, P.D., Hazenberg, P., et al. (2016). A gridded global data set of soil, intact regolith, and sedimentary deposit thicknesses for regional and global land surface modelling. *Journal of Advances in Modelling Earth Systems*, 8(1), 41–65. <https://doi.org/10.1002/2015MS000526> Soil texture data were obtained from the Harmonized World Soil Database (HWSD) version 1.2, produced by FAO, IIASA, ISRIC, ISSCAS, and JRC (2012), available at <https://pure.iiasa.ac.at/id/eprint/17595/>

Declarations

Conflict of interest The authors declare no conflict of interest.

Consent for publication Not applicable.

Ethical approval Not applicable.

Materials availability Not applicable.

Code availability All code used to produce the results and figures are available via Zenodo at <https://doi.org/10.5281/zenodo.21621236>. The implementation relies on NumPy [61], SciPy [50], PyTorch [62], xarray [63], pandas [64], Matplotlib [65], and Cartopy [66].

Use of AI in manuscript preparation AI language model tools—GitHub Copilot [56] in VS Code, and Claude [67] via the Consensus literature-search connector [68]—were used during the preparation of this manuscript to assist with text formulation, literature search, and improvement of phrasing and clarity. All AI-assisted text was critically reviewed, edited, and approved by the authors. The authors take full responsibility for the accuracy, originality, and integrity of the final manuscript.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix B: Köppen–Geiger climate-zone definitions

Table 2 lists all 30 Köppen–Geiger subclasses used as supervision labels, following the criteria of Peel et al. [42] with the C/D boundary at 0°C.

Table 2 Köppen–Geiger climate-zone definitions. T_{hot} : temperature of the warmest month; T_{cold} : temperature of the coldest month; P_{ann} : annual precipitation; P_{dry} : precipitation of the driest month; P_{th} : aridity threshold (depends on annual temperature and precipitation seasonality)

Zone	Name	Key criteria
<i>A—Tropical</i> ($T_{\text{cold}} \geq 18^\circ\text{C}$)		
Af	Tropical rainforest	$P_{\text{dry}} \geq 60\text{ mm}$
Am	Tropical monsoon	$P_{\text{dry}} < 60\text{ mm}, \geq 100 - P_{\text{ann}}/25$
Aw	Tropical savanna	$P_{\text{dry}} < 100 - P_{\text{ann}}/25$
<i>B—Arid</i> ($P_{\text{ann}} < 10 P_{\text{th}}$)		
BWh	Hot desert	$P_{\text{ann}} < 5 P_{\text{th}}, T_{\text{ann}} \geq 18^\circ\text{C}$
BWk	Cold desert	$P_{\text{ann}} < 5 P_{\text{th}}, T_{\text{ann}} < 18^\circ\text{C}$
BSh	Hot semi-arid (steppe)	$P_{\text{ann}} \geq 5 P_{\text{th}}, T_{\text{ann}} \geq 18^\circ\text{C}$
BSk	Cold semi-arid (steppe)	$P_{\text{ann}} \geq 5 P_{\text{th}}, T_{\text{ann}} < 18^\circ\text{C}$
<i>C—Temperate</i> ($0 \leq T_{\text{cold}} < 18^\circ\text{C}, T_{\text{hot}} > 10^\circ\text{C}$)		
Csa	Hot-summer Mediterranean	Dry summer, $T_{\text{hot}} \geq 22^\circ\text{C}$
Csb	Warm-summer Mediterranean	Dry summer, $T_{\text{hot}} < 22^\circ\text{C}$
Csc	Cold-summer Mediterranean	Dry summer, $< 4\text{ months} > 10^\circ\text{C}$
Cwa	Humid subtropical (dry winter)	Dry winter, $T_{\text{hot}} \geq 22^\circ\text{C}$
Cwb	Subtropical highland (dry winter)	Dry winter, $T_{\text{hot}} < 22^\circ\text{C}$
Cwc	Cold subtropical (dry winter)	Dry winter, $< 4\text{ months} > 10^\circ\text{C}$
Cfa	Humid subtropical	No dry season, $T_{\text{hot}} \geq 22^\circ\text{C}$
Cfb	Oceanic	No dry season, $T_{\text{hot}} < 22^\circ\text{C}$
Cfc	Subpolar oceanic	No dry season, $< 4\text{ months} > 10^\circ\text{C}$
<i>D—Continental</i> ($T_{\text{cold}} < 0^\circ\text{C}, T_{\text{hot}} \geq 10^\circ\text{C}$)		
Dsa	Hot-summer cont. Mediterranean	Dry summer, $T_{\text{hot}} \geq 22^\circ\text{C}$
Dsb	Warm-summer cont. Mediterranean	Dry summer, $T_{\text{hot}} < 22^\circ\text{C}$
Dsc	Subarctic (dry summer)	Dry summer, $< 4\text{ months} > 10^\circ\text{C}$
Dsd	Extremely cold (dry summer)	Dry summer, $T_{\text{cold}} < -38^\circ\text{C}$
Dwa	Hot-summer humid cont. (dry winter)	Dry winter, $T_{\text{hot}} \geq 22^\circ\text{C}$
Dwb	Warm-summer humid cont. (dry winter)	Dry winter, $T_{\text{hot}} < 22^\circ\text{C}$
Dwc	Subarctic (dry winter)	Dry winter, $< 4\text{ months} > 10^\circ\text{C}$
Dwd	Extremely cold (dry winter)	Dry winter, $T_{\text{cold}} < -38^\circ\text{C}$
Dfa	Hot-summer humid continental	No dry season, $T_{\text{hot}} \geq 22^\circ\text{C}$
Dfb	Warm-summer humid continental	No dry season, $T_{\text{hot}} < 22^\circ\text{C}$
Dfc	Subarctic (boreal)	No dry season, $< 4\text{ months} > 10^\circ\text{C}$
Dfd	Extremely cold	No dry season, $T_{\text{cold}} < -38^\circ\text{C}$
<i>E—Polar</i> ($T_{\text{hot}} < 10^\circ\text{C}$)		
ET	Tundra	$0 \leq T_{\text{hot}} < 10^\circ\text{C}$
EF	Ice cap	$T_{\text{hot}} < 0^\circ\text{C}$

Appendix C: MLP architecture details

Appendix C.1: Low-rank variant

The MLP $g_\phi: \mathbb{R}^d \rightarrow \mathbb{R}^{d \times r}$ that predicts the Low-rank factor $\mathbf{L}_i$ has the following architecture:

$$\mathbf{h}_1 = \text{ReLU}(\mathbf{W}_1 \mathbf{x}_i + \mathbf{b}_1), \quad (\text{C1})$$

$$\mathbf{h}_2 = \text{Dropout}(\mathbf{h}_1, p), \quad (\text{C2})$$

$$\mathbf{h}_3 = \text{ReLU}(\mathbf{W}_2 \mathbf{h}_2 + \mathbf{b}_2), \quad (\text{C3})$$

$$\mathbf{h}_4 = \text{Dropout}(\mathbf{h}_3, p), \quad (\text{C4})$$

$$\text{vec}(\mathbf{L}_i) = \mathbf{W}_3 \mathbf{h}_4 + \mathbf{b}_3, \quad (\text{C5})$$

where $\mathbf{W}_1 \in \mathbb{R}^{178 \times 51}$, $\mathbf{W}_2 \in \mathbb{R}^{178 \times 178}$, and $\mathbf{W}_3 \in \mathbb{R}^{408 \times 178}$ are learnable weight matrices (with hidden dimension $\lfloor d \cdot h \rfloor = \lfloor 51 \cdot 3.50 \rfloor = 178$), $\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3$ are learnable bias vectors, and $p = 0.285$ is the dropout probability applied independently at each Dropout layer during training. The output vector $\text{vec}(\mathbf{L}_i) \in \mathbb{R}^{d \cdot r} = \mathbb{R}^{408}$ is reshaped to $\mathbf{L}_i \in \mathbb{R}^{51 \times 8}$. No Softplus or normalisation is applied; PSD-ness is guaranteed by the Gram construction $\mathbf{M}_i = \mathbf{L}_i \mathbf{L}_i^\top$. Output-layer weights are initialised from $\mathcal{N}(0, 0.01^2)$ with zero bias, so that $\mathbf{M}_i = \mathbf{L}_i \mathbf{L}_i^\top \approx \mathbf{0}$ at initialisation. The regularisation term (D15) pulls the metric towards the identity $\mathbf{I}_d$ during training.

PSD guarantee. For any matrix $\mathbf{L}_i \in \mathbb{R}^{d \times r}$ and any vector $\mathbf{v} \in \mathbb{R}^d$, $\mathbf{v}^\top \mathbf{M}_i \mathbf{v} = \mathbf{v}^\top \mathbf{L}_i \mathbf{L}_i^\top \mathbf{v} = \|\mathbf{L}_i^\top \mathbf{v}\|^2 \geq 0$, so $\mathbf{M}_i$ is PSD by construction regardless of the MLP output.

Appendix C.2: Diagonal variant

The Diagonal MLP $f_\theta : \mathbb{R}^d \rightarrow \mathbb{R}_{\geq 0}^d$ predicts a non-negative weight vector $\mathbf{w}_i$ using a decreasing-width architecture:

$$\mathbf{h}_1 = \text{ReLU}(\mathbf{W}_1 \mathbf{x}_i + \mathbf{b}_1), \quad (\text{C6})$$

$$\mathbf{h}_2 = \text{Dropout}(\mathbf{h}_1, p), \quad (\text{C7})$$

$$\mathbf{h}_3 = \text{ReLU}(\mathbf{W}_2 \mathbf{h}_2 + \mathbf{b}_2), \quad (\text{C8})$$

$$\mathbf{h}_4 = \text{Dropout}(\mathbf{h}_3, p), \quad (\text{C9})$$

$$\tilde{\mathbf{w}}_i = \text{Softplus}(\mathbf{W}_3 \mathbf{h}_4 + \mathbf{b}_3), \quad (\text{C10})$$

$$\mathbf{w}_i = \frac{\tilde{\mathbf{w}}_i}{\|\tilde{\mathbf{w}}_i\|_1}, \quad (\text{C11})$$

where $\mathbf{W}_1 \in \mathbb{R}^{108 \times 51}$, $\mathbf{W}_2 \in \mathbb{R}^{54 \times 108}$, and $\mathbf{W}_3 \in \mathbb{R}^{51 \times 54}$ are learnable weight matrices. The hidden dimensions follow a halving scheme: $h_1 = \max(16, \lfloor d \cdot h \rfloor) = 108$ and $h_2 = \max(8, \lfloor h_1/2 \rfloor) = 54$, with $h = 2.12$. Dropout probability is $p = 0.285$. The element-wise Softplus ensures all weights are strictly positive, and the ℓ_1 -normalisation makes $\mathbf{w}_i$ a proper distribution over features. No near-zero initialisation is required: the Softplus activation and subsequent normalisation keep outputs bounded regardless of initial parameter values.

Appendix D: Loss function details

For each training cell i , let $\mathcal{T}_i$ denote its k_i nearest same-zone neighbours (in Euclidean feature space) and $\mathcal{I}_i$ the set of different-zone cells in the training batch (capped at 50). Following the large-margin nearest-neighbour formulation [39], the Low-rank variant combines a pull term, a squared-hinge margin term, and a regularisation term

$$\mathcal{L}_{\text{Low-rank}} = (1 - \alpha) \mathcal{L}_{\text{pull}} + \alpha \mathcal{L}_{\text{margin}} + \lambda_{\text{reg}} \mathcal{L}_{\text{reg}}, \quad (\text{D12})$$

where $\alpha \in [0, 1]$ is the trade-off weight that balances the pull and margin terms, $\lambda_{\text{reg}} \geq 0$ is the regularisation weight, and the component losses are

$$\mathcal{L}_{\text{pull}} = \frac{1}{k_i} \sum_{j \in \mathcal{T}_i} D_{\text{maha}}^2(\mathbf{x}_i, \mathbf{x}_j), \quad (\text{D13})$$

$$\mathcal{L}_{\text{margin}} = \frac{1}{k_i |\mathcal{I}_i|} \sum_{j \in \mathcal{I}_i} \sum_{l \in \mathcal{I}_i} \max(0, D_{\text{maha}}^2(\mathbf{x}_i, \mathbf{x}_j) - D_{\text{maha}}^2(\mathbf{x}_i, \mathbf{x}_l) + \Delta)^2, \quad (\text{D14})$$

$$\mathcal{L}_{\text{reg}} = \frac{1}{n} \sum_{i=1}^n \|\mathbf{M}_i - \mathbf{I}_d\|_F^2. \quad (\text{D15})$$

The pull term (D13) draws same-zone cells together [39]. The margin term (D14) enforces a minimum gap Δ between the distance to a same-zone neighbour and the distance to a different-zone cell; the squared hinge provides continuous gradients and penalises large violations more strongly than small ones [39]. The effective margin Δ is derived from a tuned base margin Δ_{base} by the dimension scaling $\Delta = \Delta_{\text{base}} \sqrt{d/d_0}$, where d is the feature dimension and d_0 is a reference dimension (the feature count of the initial tuning round); this keeps the margin comparable across feature sets of differing dimensionality. The regularisation term (D15), averaged over all n training cells, anchors each local metric matrix $\mathbf{M}_i$ to the identity matrix $\mathbf{I}_d \in \mathbb{R}^{d \times d}$, where $\|\cdot\|_F$ denotes the Frobenius norm (the square root of the sum of squared matrix entries). The identity is the natural reference point: D_{maha} with $\mathbf{M}_i = \mathbf{I}_d$ reduces to the plain squared Euclidean distance, so the penalty encourages each cell's metric to deviate from uniform feature weighting only where the training signal provides evidence for it. Regularising a learned Mahalanobis metric towards a fixed prior such as the identity is a standard means of avoiding degenerate solutions [69], in which the metric is anchored to a baseline matrix and only allowed to depart from it where warranted by the data. Here, this is particularly important, because each $\mathbf{M}_i$ is produced by the MLP rather than constrained directly: without this term, gradient descent can drive $\mathbf{M}_i$ towards zero (making all cells appear equidistant) or to unbounded magnitudes [69]. The weight λ_{reg} controls how strongly the metric is pulled back towards the Euclidean baseline relative to the pull and margin objectives.

The Diagonal variant uses a two-term loss without explicit regularisation

$$\mathcal{L}_{\text{Diag}} = (1 - \alpha) \mathcal{L}_{\text{pull}} + \alpha \mathcal{L}_{\text{margin}}, \quad (\text{D16})$$

where the pull and margin terms are defined identically to Eqs. (D13)–(D14) but with D_{diag} (Eq. 3) in place of D_{maha} . No explicit regularisation is needed here, because D_{diag} is parameterised by a non-negative weight vector $\mathbf{w}_i$ that is L_1 -normalised after each gradient step, bounding each entry to $[0, 1]$ and preventing collapse or explosion by construction.

Appendix E: Hyperparameter settings

Tables 3 and 4 list the tuned hyperparameter values used in the Low-rank and Diagonal production runs, respectively. All values were selected by the two-round Optuna search described in Sect. 2.4. Both variants are trained with the AdamW optimiser [70], an adaptive gradient method that maintains per-parameter learning rates based on running estimates of the first and second gradient moments. Unlike standard Adam, AdamW applies weight decay directly to the parameters rather than folding it into the gradient update, which prevents the adaptive learning rates from distorting the regularisation effect.

Table 5 summarises the search ranges used in the initial and refined Optuna tuning rounds ($d = 51$). Parameters marked with ^L apply only to the Low-rank variant; all other parameters are shared.

Appendix F: Linkage method comparison

To verify the choice of complete linkage, Ward linkage was tested on the same learned distance matrix (Low-rank metric, $d=51$ features) without dedicated hyperparameter tuning. Table 6 reports the Adjusted Rand Index (ARI) and Normalised Mutual Information (NMI) at $K=30$ clusters for both linkage methods. Complete linkage achieves a higher ARI, indicating better agreement with the Köppen–Geiger partition used as the evaluation target, while the two methods produce similar NMI values. Because the hyperparameter search (Sect. 2.4) was conducted with complete linkage, comparing against a non-tuned Ward run establishes a lower bound on the performance gap; dedicated tuning for Ward might narrow the difference, but the result supports the choice of complete linkage as the default.

Table 3 Tuned hyperparameter values for the **Low-rank** (local Mahalanobis) LMNN production run ($d = 51$, $n = 67\,420$ cells)

Hyperparameter	Symbol	Value
Feature dimension	d	51
Low-rank factor rank	r	8
Margin trade-off weight	α	0.452
Base margin	Δ_{base}	0.842
Effective margin	$\Delta = \Delta_{\text{base}} \sqrt{d/d_0}$	1.157
Margin reference dimension	d_0	27
Regularisation weight	λ_{reg}	1.24×10^{-4}
Learning rate (AdamW)	η	1.96×10^{-3}
Weight decay (AdamW)	–	1.07×10^{-3}
Batch size	–	512
Hidden width factor	h	3.50
Hidden dimensions	(h_1, h_2)	(178, 178)
Dropout probability	p	0.285
Early-stopping patience	–	15 epochs
Validation split	–	20%
Max impostors per anchor	–	50

Table 4 Tuned hyperparameter values for the **Diagonal** (local weighted) LMNN production run ($d = 51$, $n = 67\,420$ cells)

Hyperparameter	Symbol	Value
Feature dimension	d	51
Margin trade-off weight	α	0.786
Base margin	Δ_{base}	0.585
Effective margin	$\Delta = \Delta_{\text{base}} \sqrt{d/d_0}$	0.804
Margin reference dimension	d_0	27
Learning rate (AdamW)	η	1.46×10^{-3}
Weight decay (AdamW)	–	3.12×10^{-4}
Batch size	–	512
Hidden width factor	h	2.12
Hidden dimensions	(h_1, h_2)	(108, 54)
Dropout probability	p	0.285
Early-stopping patience	–	15 epochs
Validation split	–	20%
Max impostors per anchor	–	50

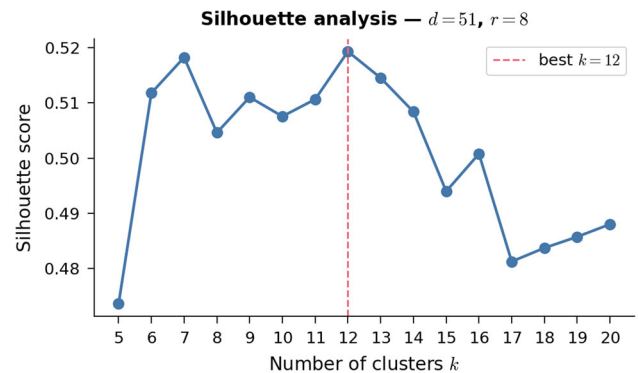
Table 5 Optuna search ranges for the initial and refined tuning rounds. Phase 2 parameters were fixed in the refined round. ^L Low-rank variant only

Parameter	Phase	Initial range	Refined range
α	1	[0.3, 0.8]	[0.40, 0.65]
Δ_{base}	1	[0.5, 2.0]	[0.70, 1.10]
r^{L}	1	{2, 4, 8, 16}	{2, 4, 8}
Batch size	2	{128, 256, 512, 1024}	(fixed at 512)
η	2	$[10^{-4}, 10^{-2}]$	(fixed)
Weight decay	2	$[10^{-5}, 10^{-2}]$	(fixed)
$\lambda_{\text{reg}}^{\text{L}}$	2	$[10^{-6}, 10^{-2}]$	$[5 \times 10^{-5}, 5 \times 10^{-4}]$
h	3	[1.0, 6.0]	[2.5, 4.5]
p (dropout)	3	[0.0, 0.5]	[0.15, 0.40]

Table 6 Linkage comparison at $K=30$ using the Low-rank distance matrix ($d=51$, $n=67\,420$). Ward linkage was run without dedicated hyperparameter tuning

Linkage method	ARI	NMI
Complete	0.367	0.621
Ward	0.323	0.629

Fig. 6 Mean silhouette score as a function of the number of K -means clusters k applied to the Low-rank weight-profile matrix ($d=51$ features, $n = 67\,420$ cells). The silhouette score peaks at $k=12$, which is adopted for the regime analysis in the main text



Appendix G: Weight-regime cluster selection

The number of weight regimes k used in the eigenvector-profile analysis (Sect. 2.5) was selected by evaluating K -means clustering of the $n \times d$ weight-profile matrix for $k \in \{5, \dots, 20\}$ and choosing the value that maximises the mean silhouette score (Fig. 6). The silhouette score peaks at $k=12$ (silhouette = 0.519), followed by $k=7$ (silhouette = 0.518) and $k=13$ (silhouette = 0.515). Values of $k < 5$ were excluded, because very coarse partitions merge ecologically distinct regimes (e.g., all temperature-related features collapse into a single cluster). The selected $k=12$ resolves the full spectrum of climate–vegetation contrasts visible in the dominant-feature map (Fig. 2b).

Appendix H: Simulation infrastructure and metric learning cost

Both metric learning pipelines were executed on NVIDIA H100 80 GB HBM3 GPUs via the PIK HPC cluster (partition `gpu`, QoS `gpupreempt`), each using a single GPU and eight CPU cores. Infrastructure was identical for both variants.

The hyperparameter tuning pipeline consists of four sequential phases: Phase 1 tunes loss-landscape parameters and Phase 2 tunes optimiser dynamics, both on a 15 002-cell subsample. Phase 3 tunes architecture parameters on the full 67 420-cell grid, and Phase 4 performs the final production training and distance matrix computation on the full grid. Table 7 summarises the per-phase breakdown.

Pruning and convergence behaviour. On the subsample phases (1–2), both variants completed all requested trials. On the full-data Phase 3, 3 of 5 Diagonal trials and 1 of 3 Low-rank trials were pruned by the Optuna TPE sampler. Pruned trials consume only a fraction of the compute of a completed trial.

Energy consumption estimate. The NVIDIA H100 SXM GPU used for all experiments has a maximum thermal design power (TDP) of 700 W [71]. At full TDP, the 14.5 GPU-hours required for the complete Diagonal tuning and training pipeline (Table 7) correspond to an upper bound of $14.5 \text{ h} \times 0.70 \text{ kW} = 10.15 \text{ kWh} \approx 10 \text{ kWh}$. In practice, GPU utilisation during training and hyperparameter tuning rarely reaches sustained full TDP, so actual energy consumption is likely somewhat below this ceiling. The Low-rank pipeline (9.4 GPU-hours total; Table 7) correspondingly consumes at most $9.4 \times 0.70 = 6.6 \text{ kWh}$. Both figures are upper bounds based on peak TDP and should be interpreted accordingly.

Training implementation and comparability. The Diagonal and Low-rank variants were trained with different implementations of the LMNN loss, so their per-phase wall times (Table 7) are *not* directly comparable as a measure of inherent computational complexity. The Diagonal training loop materialises a full $B \times B$ pairwise distance matrix per batch ($B = 512$, yielding 262 144 distance evaluations) and mines triplets with explicit per-anchor iteration, resulting in ~ 77 s per epoch on the full grid. The Low-rank implementation pre-selects triplets by index and computes distances only for the resulting anchor–target and anchor–impostor pairs, achieving ~ 31 s per epoch despite having $8\times$ more MLP parameters (57 834 vs. 7 091). Both variants completed the full 100-epoch

Table 7 Computational cost breakdown for Diagonal and Low-rank LMNN hyperparameter tuning and production training on a single NVIDIA H100 80 GB HBM3 GPU. “Trials” is the number requested by the Optuna optimiser; “completed” is the number that ran to convergence (the remainder were pruned via early stopping). Wall times are from the actual SLURM job logs

Phase	Diagonal			Low-rank		
	Trials (completed)	Data	Wall time	Trials (completed)	Data	Wall time
1	20 (20)	15 k sub	4.9 h	25 (25)	15 k sub	4.3 h
2	15 (15)	15 k sub	4.9 h	15 (15)	15 k sub	2.4 h
3	5 (2)	full	2.6 h	3 (2)	full	1.4 h
4	1 (1)	full	2.2 h	1 (1)	full	1.2 h
Tuning (1–3)			12.4 h			8.1 h
Production (4)			2.2 h			1.2 h
Total			14.5 h			9.4 h

budget in Phase 4. In principle, the Diagonal variant should be cheaper to train and tune than the Low-rank variant: its MLP is smaller, its per-cell distance computation scales as $O(d)$ rather than $O(d^2)$, and its hyperparameter search space contains fewer dimensions (no rank parameter, no regularisation weight). The higher observed wall times for the Diagonal pipeline are therefore an artefact of the less efficient training-loop implementation rather than a fundamental property of the metric variant. A unified, optimised training loop would likely reverse this ordering.

Distance matrix computation. The final production distance matrix ($67\,420 \times 67\,420$) required 100 s for the Diagonal metric and 335 s for the Low-rank metric ($\sim 3.4\times$), confirming the expected quadratic scaling in feature dimension d for the Low-rank variant. On the 15,002-cell subsample used during tuning, per-evaluation distance matrix cost was approximately 4 s (Diagonal) versus 19 s (Low-rank).

Diagonal pipeline preemptions. The Diagonal pipeline was run with the `gpupreempt` QoS and required multiple SLURM job submissions before all four phases completed. The Optuna study is backed by SQLite, so interrupted jobs could be resumed without loss of completed trials. Job 1 completed Phases 1 and 2 but was preempted at the wall-time limit before Phase 3 began. Job 2 loaded the cached Phase 1–2 results and completed the remaining Phase 3 trials and Phase 4. The wall times in Table 7 reflect the actual execution times of each phase.

Comparison with DGVM simulation cost. In practice, only one metric variant is needed; the recommended Diagonal pipeline requires ~ 14.5 GPU-hours for full tuning and training, or ~ 2 GPU-hours for production training alone when reusing previously tuned hyperparameters. A single LPJmL-FIT baseline simulation with 25 patch replicates and 1 000-year spin-up requires $\sim 13\,300$ CPU-hours (≈ 237 kWh, $\approx \text{€ } 290$) and the 20-year transient run $\sim 1\,150$ CPU-hours (≈ 20 kWh, $\approx \text{€ } 25$) on the PIK HPC cluster, using 2048 MPI tasks distributed across 16 Lenovo SD665 V3 compute nodes, each equipped with two AMD EPYC 9554 Genoa processors (64 cores per socket, 128 cores per node, 3.1 GHz base clock) and 768 GiB of dual-ranked DDR5 memory (4800 MHz; ≈ 6 GiB per core). Research configurations routinely employ 1000–10,000 patch replicates [5, 8], scaling these costs by a factor of $40\times$ – $400\times$. The metric learning cost is thus negligible relative to even a single DGVM run and is amortised across all subsequent cluster-based simulations at any K .

Appendix I: Geographic spread of the partition

For each cluster c with member latitudes $\{\text{lat}_i\}_{i \in c}$ and longitudes $\{\text{lon}_i\}_{i \in c}$, the per-cluster geographic spread is $s_c = \sqrt{\text{Var}(\text{lat}_c) + \text{Var}(\text{lon}_c)}$ (degrees), and the global mean spread is the cluster-size-weighted average $\bar{s} = \sum_c (n_c/n) s_c$; formal details are given in Supplementary Material Section S5, “Geographic Spread”. Low $\bar{s}$ indicates geographically compact clusters; high $\bar{s}$ indicates geographically dispersed clusters.

The learned metrics retain substantial geographic spread at every K , because the LMNN objective loads sensitivity onto the small number of features that discriminate KG zones; climatically similar but geographically distant cells (e.g., temperate grasslands in Patagonia and central Asia) therefore continue to merge as K grows, whereas a uniform Euclidean metric treats all 51 features equally and converges to geographically compact clusters once the cluster budget is dense enough to resolve continental-scale spatial autocorrelation. This geographic-spread gap is the direct cause of the soil- and micro-climate-variance gap reported in Fig. 3h, k, and of the Euclidean reconstruction advantage above the crossover ($K \gtrsim 674$, $E \lesssim 100$; Sect. 4.1).

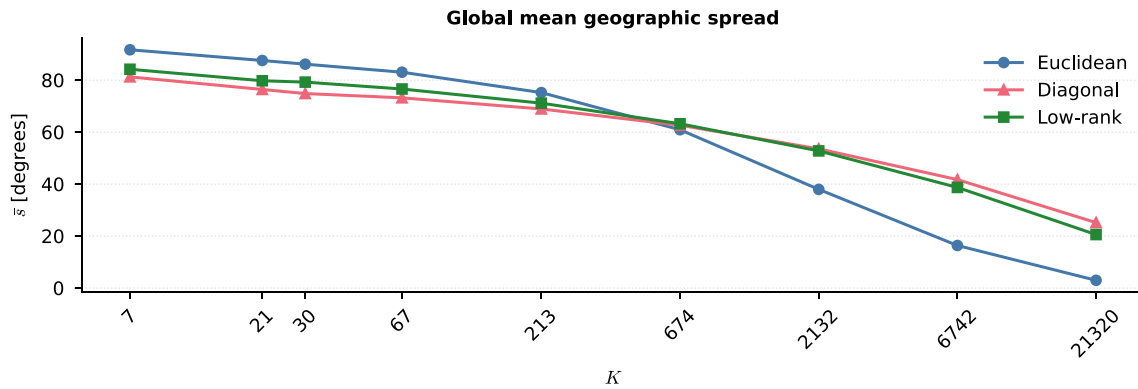


Fig. 7 Global mean geographic spread $\bar{s}$ as a function of cluster count K for Euclidean, Diagonal, and Low-rank clustering. All three methods decrease monotonically with K , but at very different rates: the Euclidean spread shrinks from $\sim 87^\circ$ at $K = 7$ to $\sim 3^\circ$ at $K = 21,320$, whereas the learned spreads decrease only from $\sim 80^\circ$ to $\sim 20\text{--}40^\circ$ over the same range. The Euclidean curve crosses below both learned curves near $K \approx 674$ —the same cluster count at which the reconstruction-accuracy ranking reverses (Fig. 4b)

Appendix J: Cluster allocation: KG boundaries versus biomass gradients

This appendix quantifies the extent to which reconstruction error is controlled by how the cluster budget is distributed over the map—relative to the KG training signal on the one hand and to simulated biomass gradients on the other. The conclusion is that the learned metrics concentrate cluster density at KG-zone boundaries whether or not those coincide with biomass gradients, controlling the error at low-to-moderate K ; Euclidean clustering allocates cluster density proportionally to biomass gradients and overtakes the learned metrics at high K exactly where that property matters.

Appendix J.1: Allocation of clusters to KG boundaries and biomass gradients

The local cluster diversity $d_i^{(m,K)}$ and the KG-boundary proximity b_i^{KG} are defined in Sect. 2.8; formal equations are in Supplementary Material Section S5. The complementary biomass-gradient reference field is:

- **Biomass-gradient magnitude b_i^{BM} :** the standard deviation of time-mean baseline biomass over the 5×5 window, normalised by the global biomass standard deviation. High values identify cells in steep simulated biomass transitions (tropical-forest edges, boreal-tundra ecotone, biome margins in general).

Applying the same quartile contrast as Eq. 8 to b^{BM} yields $\text{BAR}_{\text{BM}}(m, K)$, which probes alignment with what actually drives biomass errors rather than with the training signal. The *misallocation gap* $\Delta\text{BAR}(m, K) = \text{BAR}_{\text{KG}}(m, K) - \text{BAR}_{\text{BM}}(m, K)$ is large and positive for a method that concentrates clusters at climatic borders that are *not* biomass transitions.

Result: a sharp methodological dissociation. Figure 8 shows the three diagnostics across cluster counts spanning four orders of magnitude ($K \in \{7, 21, 30, 67, 213, 674, 2132, 6742, 21320\}$).

Three observations stand out. First, the learned metrics concentrate 2–3 times as much cluster density at KG-zone boundaries as in zone interiors, peaking at $\text{BAR}_{\text{KG}} \approx 3.0$ for Diagonal at $K=67$ and ≈ 2.8 for Low-rank at $K=30\text{--}67$. This is a direct signature of the training signal: the LMNN objective shrinks within-zone distances and stretches across-zone distances, so complete-linkage places the earliest clusters astride the KG borders. Euclidean clustering has no such signal and shows a modest $\text{BAR}_{\text{KG}} \approx 1.4\text{--}1.6$ throughout.

Second, the biomass-gradient allocation reverses the ranking. BAR_{BM} for Euclidean grows to ≈ 2.0 at $K = 213\text{--}674$ and stays above both learned metrics for all $K \geq 67$. This too is predicted by the mechanism: because KG borders only partially coincide with biomass gradients, a cluster budget aimed at KG borders *undershoots* the biomass transitions and over-refines biomass-homogeneous borders.

Third, the misallocation gap ΔBAR is large and positive for the learned metrics at low-to-moderate K ($+1.5$ for Diagonal at $K = 67$, $+1.1$ for Low-rank at $K = 30$), weakly negative for Euclidean at moderate K (-0.3 to -0.5 at $K = 213\text{--}674$), and contracts towards zero for all three methods at very high K once the cluster budget saturates the map.

Two error regimes. At low-to-moderate K ($K \lesssim 674$), misallocation dominates: the large positive gap ($\Delta\text{BAR} \geq +1.1$ for Diagonal and Low-rank) means that a substantial fraction of the cluster budget is spent

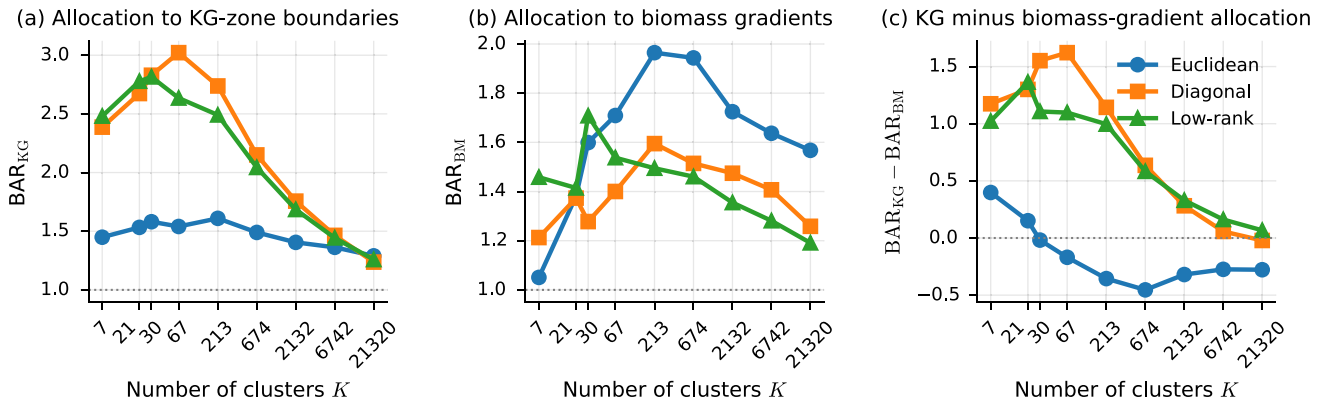


Fig. 8 Where do clusters concentrate? Single-number-per- K diagnostics of cluster allocation, computed on a 5×5 window ($\pm 1.0^\circ$ half-width) around each land cell. **a** BAR_{KG} : ratio of mean local cluster diversity in the top 25 % of cells by KG-boundary proximity b^{KG} to the mean in the bottom 25 %. A value of 2–3 indicates that the method uses roughly 2–3 times as many clusters per unit area near KG zone boundaries as in zone interiors. **b** BAR_{BM} : the same statistic using biomass-gradient magnitude b^{BM} as the reference field instead of KG boundaries. **c** Misallocation gap $\Delta \text{BAR} = \text{BAR}_{\text{KG}} - \text{BAR}_{\text{BM}}$. Positive values mean that the method spends more of its cluster budget on KG boundaries than on biomass transitions; negative values mean the opposite. The dotted line at 1 in (a, b) and at 0 in (c) marks a spatially uniform allocation

on KG borders that are not biomass transitions. The learned metrics nonetheless outperform Euclidean, because their much higher BAR_{KG} (2.4–3.0 vs. 1.4–1.6) provides the KG-aligned skeleton that the Euclidean partition is still missing at this scale; biome-level internal homogeneity gained from strong KG-boundary alignment outweighs the misallocation penalty throughout this regime. At higher K ($K \gtrsim 674$), this advantage disappears and geographic compactness becomes the limiting factor. The performance reversal visible in Fig. 4b coincides with the point where Euclidean’s mean geographic spread $\bar{s}$ falls below that of both learned metrics (Fig. 7): as both BAR_{KG} and BAR_{BM} approach unity, Euclidean’s compact clusters ($\bar{s} \approx 3^\circ$ at $K = 21,320$) guarantee that soil, radiation, and local microclimate are nearly uniform within each cluster, so residual biomass heterogeneity is attacked directly. The learned metrics, by contrast, retain clusters of $20\text{--}40^\circ$ geographic spread, because their narrow feature sensitivity (Fig. 3h, k) continues to merge climatically similar but geographically distant cells. Consequently, learned-metric within-cluster soil variance stays $45\times\text{--}90\times$ larger than Euclidean’s at $K = 6,742$ and $K = 21,320$, and per-cell errors remain higher correspondingly.

Appendix K: Biomass reconstruction maps

See Figs. 9, 10, 11.

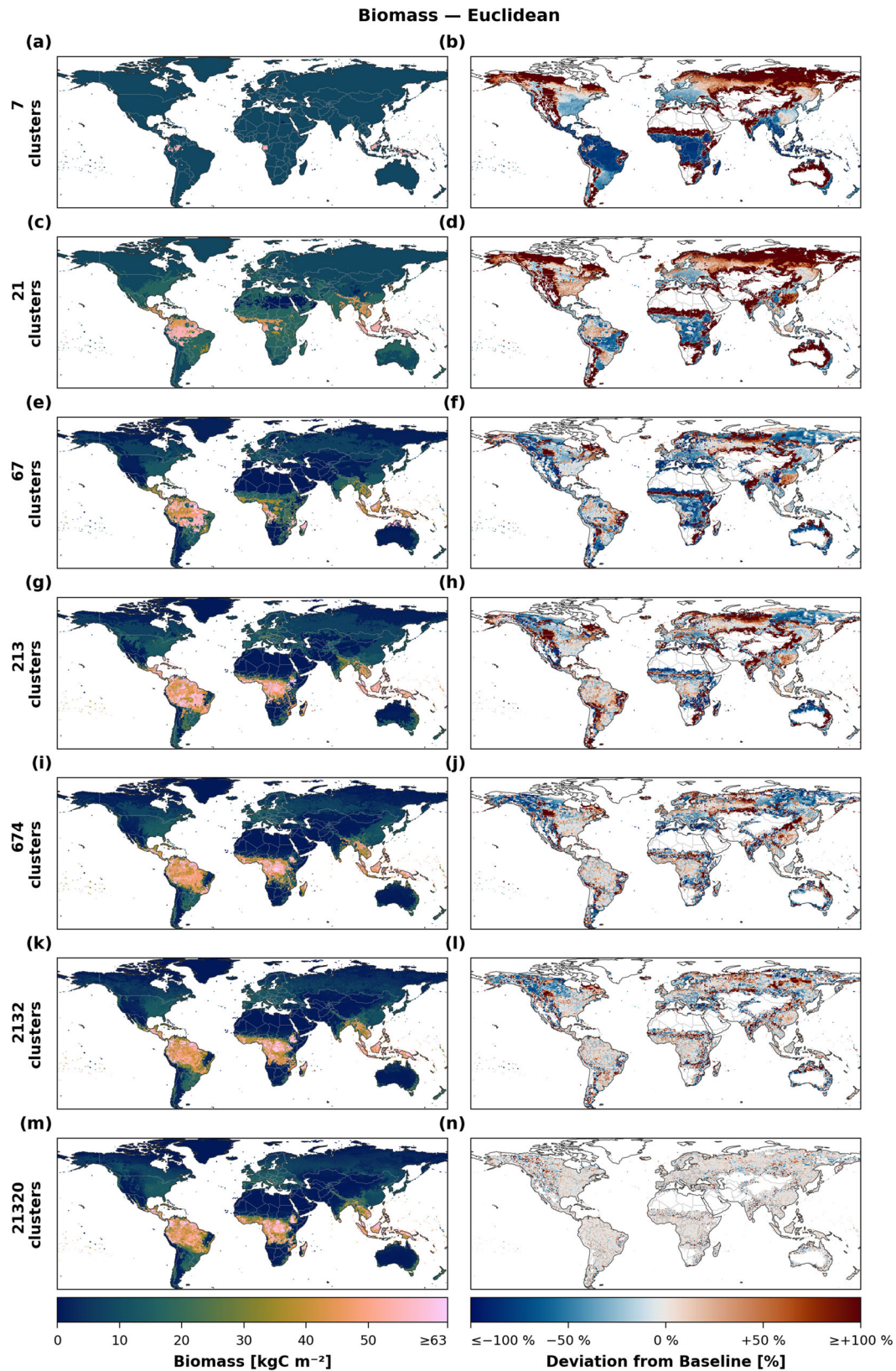


Fig. 9 Spatial biomass distribution and percentage deviation for the Euclidean metric across all cluster counts except $K = 6742$ (shown in Fig. 5). Left column: time-mean above- and belowground biomass [kgC m^{-2}]. Right column: percentage deviation from the cell-based baseline, capped at $\pm 100\%$

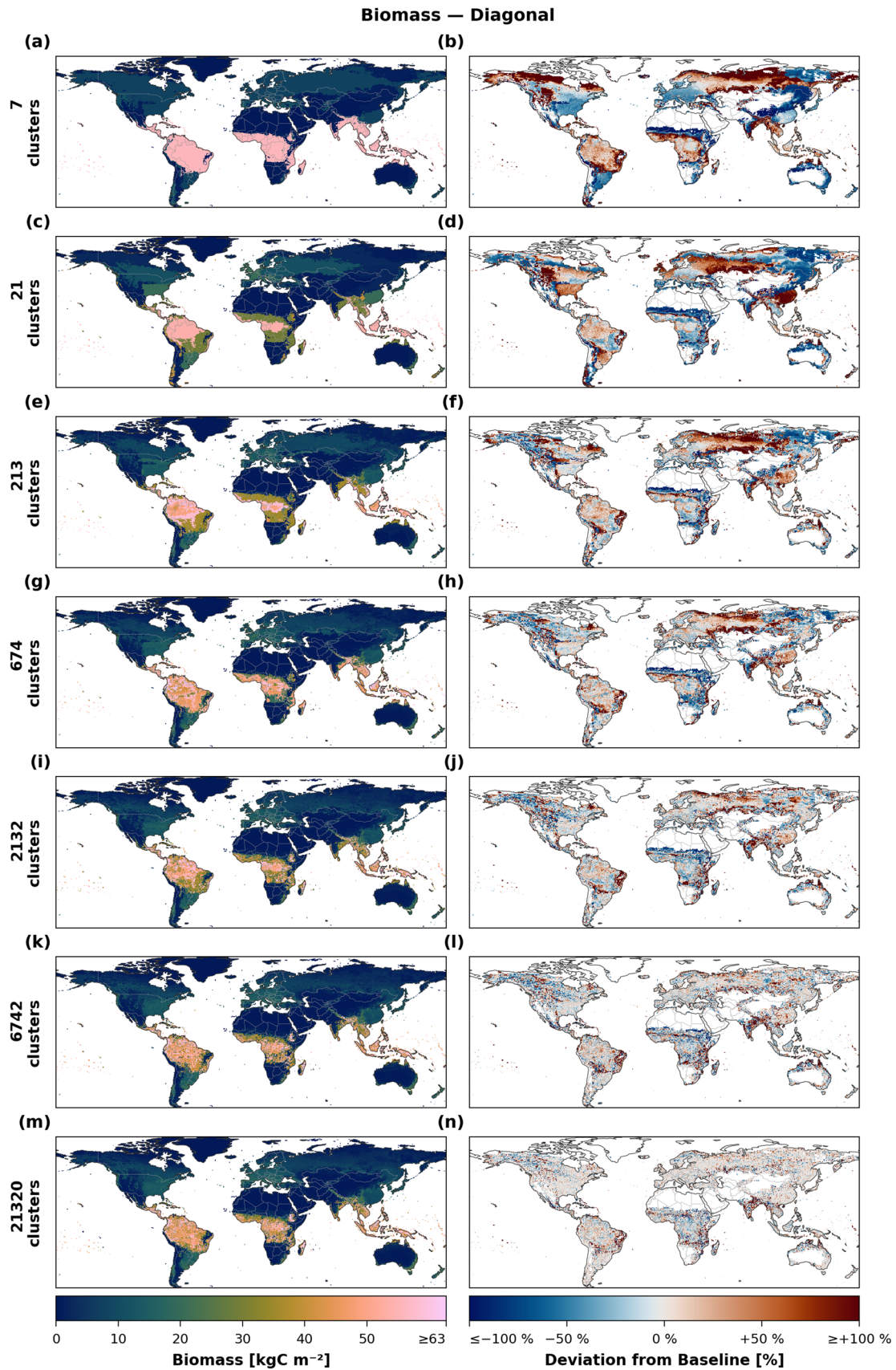


Fig. 10 Spatial biomass distribution and percentage deviation for the Diagonal metric across all cluster counts except $K = 67$ (shown in Fig. 5). Layout as in Fig. 9

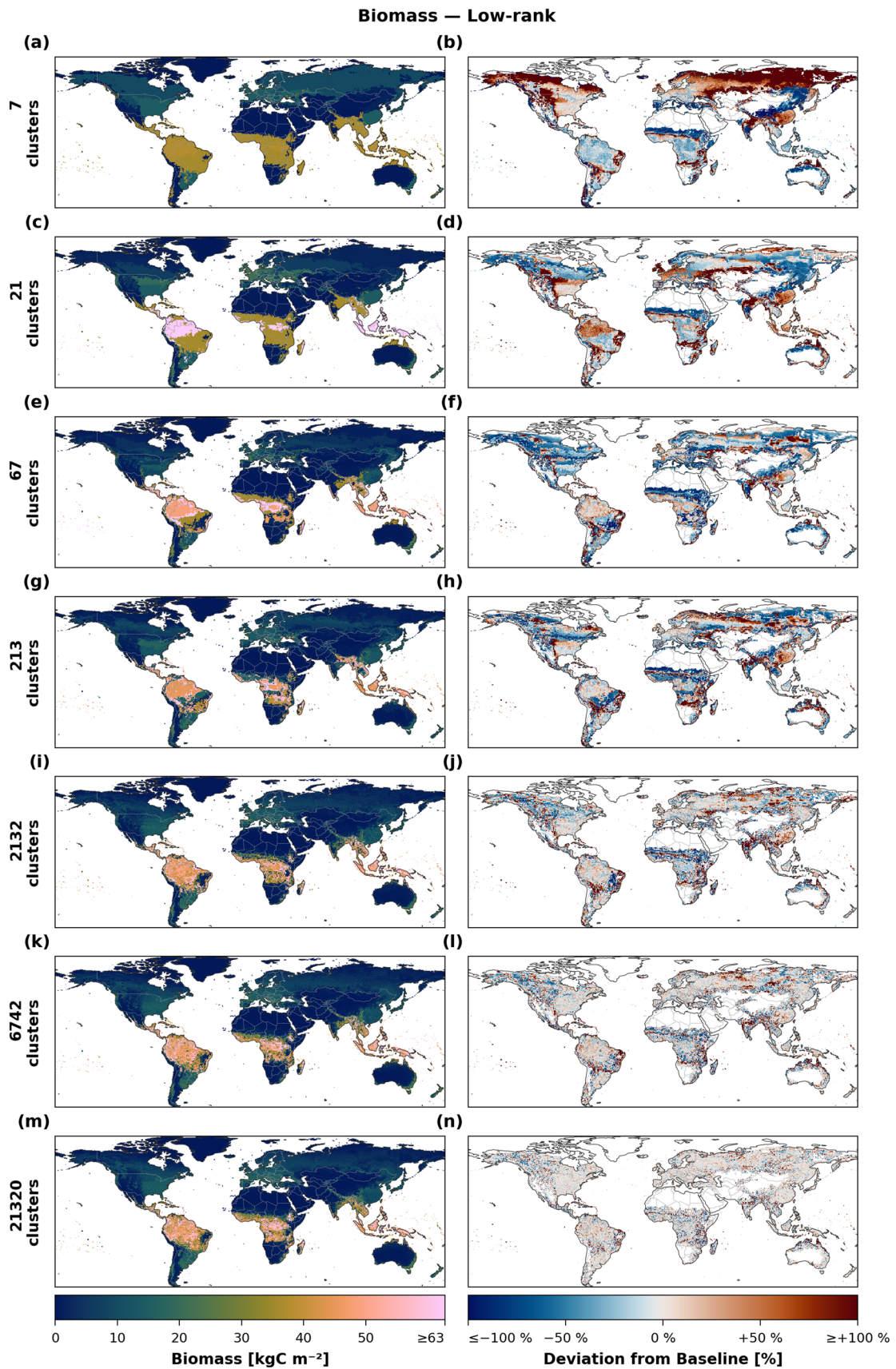
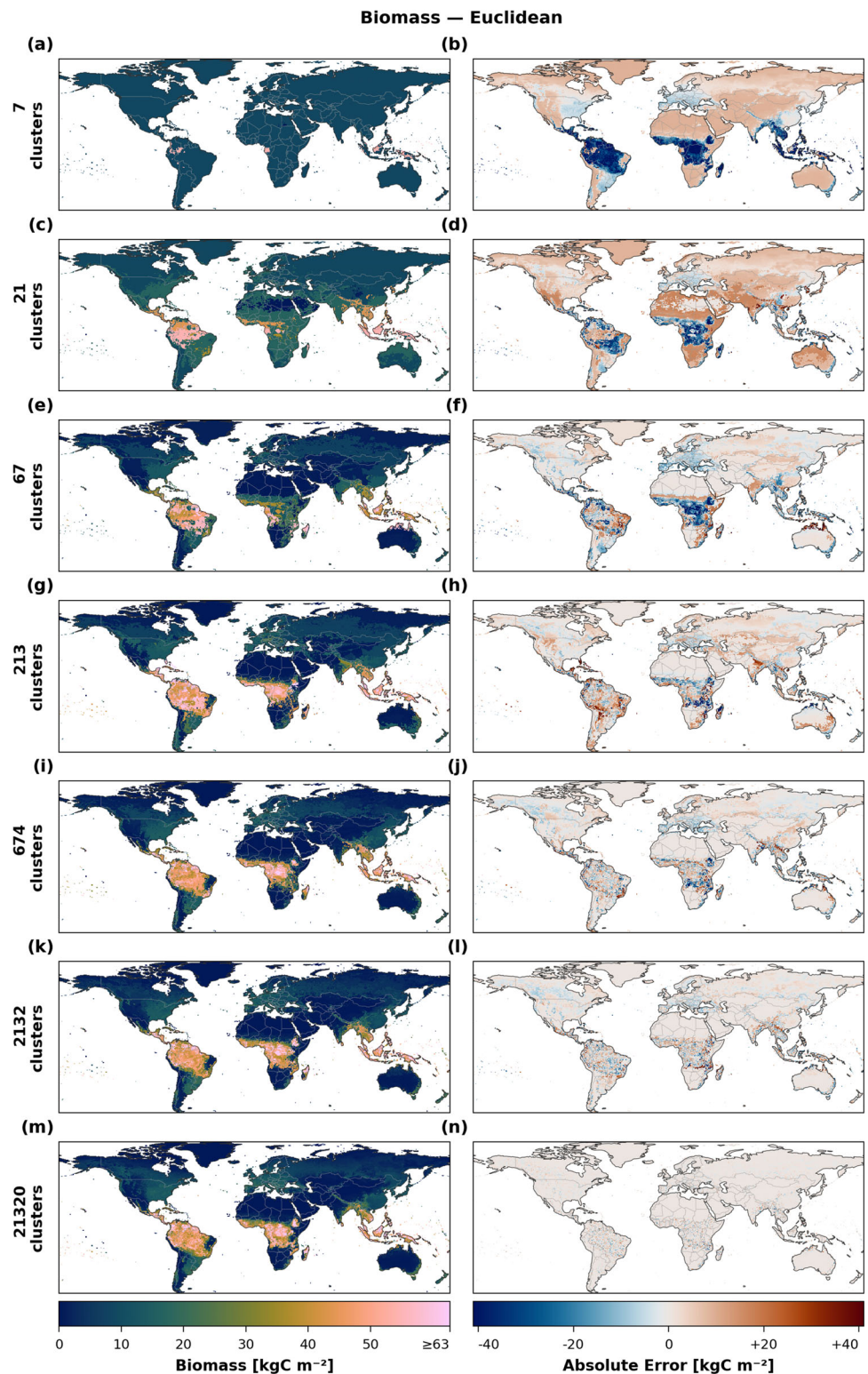


Fig. 11 Spatial biomass distribution and percentage deviation for the Low-rank metric across all cluster counts except $K = 674$ (shown in Fig. 5). Layout as in Fig. 9

Appendix L: Absolute biomass deviation maps

See Figs. 12, 13, 14.

Fig. 12 Absolute biomass deviation for the Euclidean metric across all cluster counts except $K = 6742$ (shown in Fig. 5). Left column: time-mean above- and belowground biomass [kgC m^{-2}] from the cell-based baseline. Right column: absolute deviation of reconstructed from baseline biomass [kgC m^{-2}], capped at the 99th percentile. High-biomass regions (tropics, boreal forests) dominate the absolute error signal, because the same fractional mismatch translates into a larger absolute penalty there



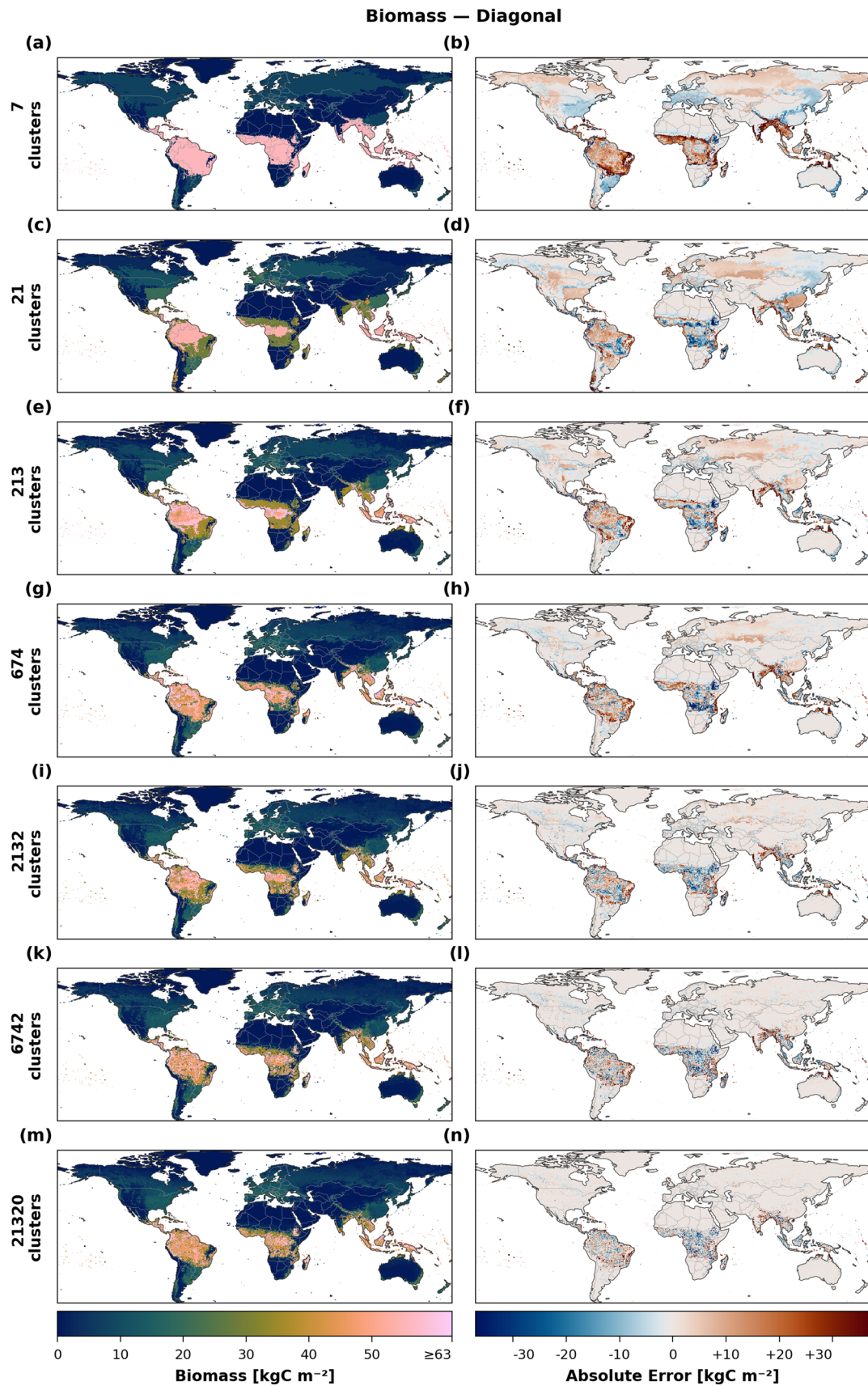


Fig. 13 Absolute biomass deviation for the Diagonal metric across all cluster counts except $K = 67$ (shown in Fig. 5). Layout as in Fig. 12

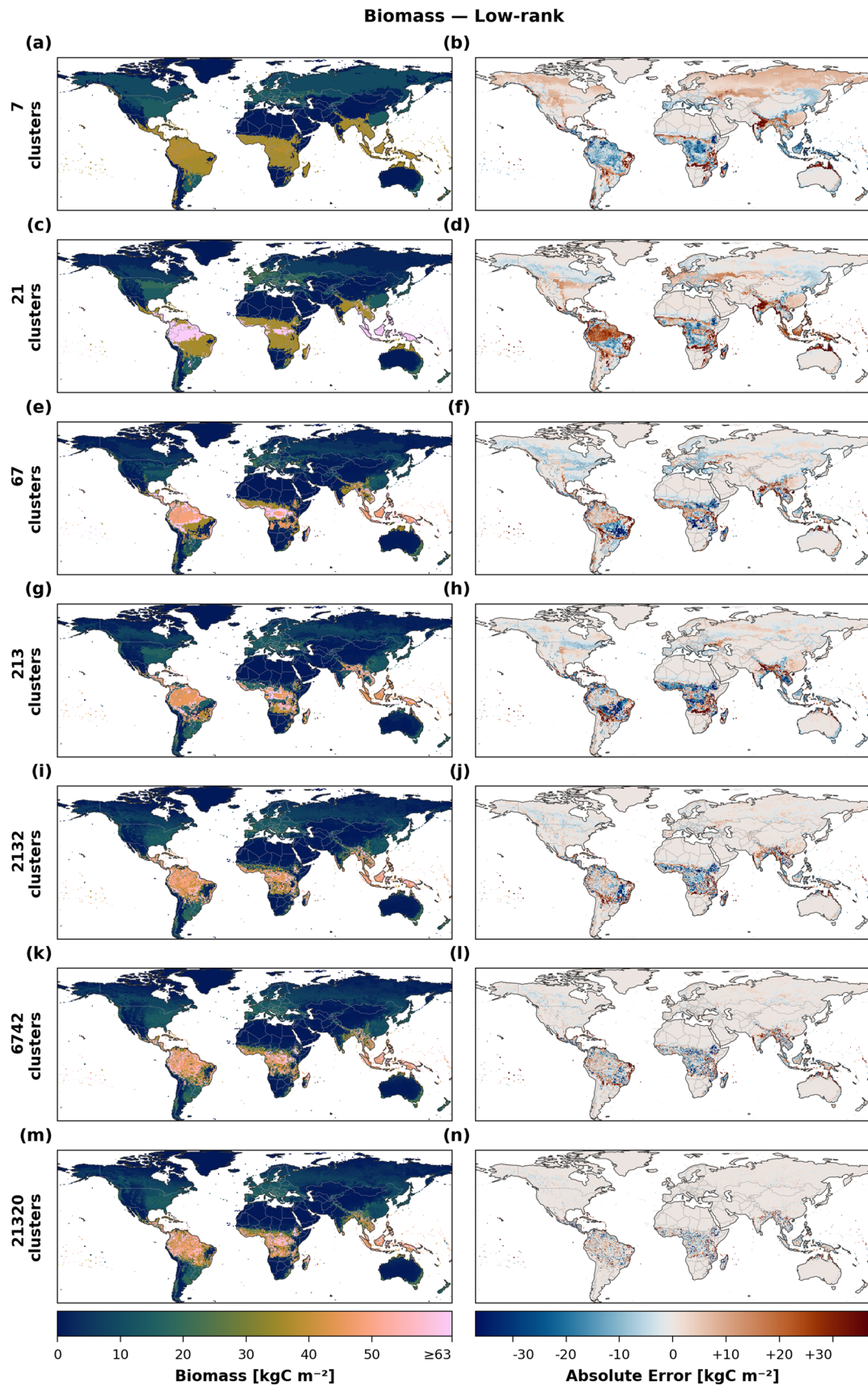


Fig. 14 Absolute biomass deviation for the Low-rank metric across all cluster counts except $K = 674$ (shown in Fig. 5). Layout as in Fig. 12

Appendix M: Functional diversity maps

See Figs. 15, 16, 17, 18.

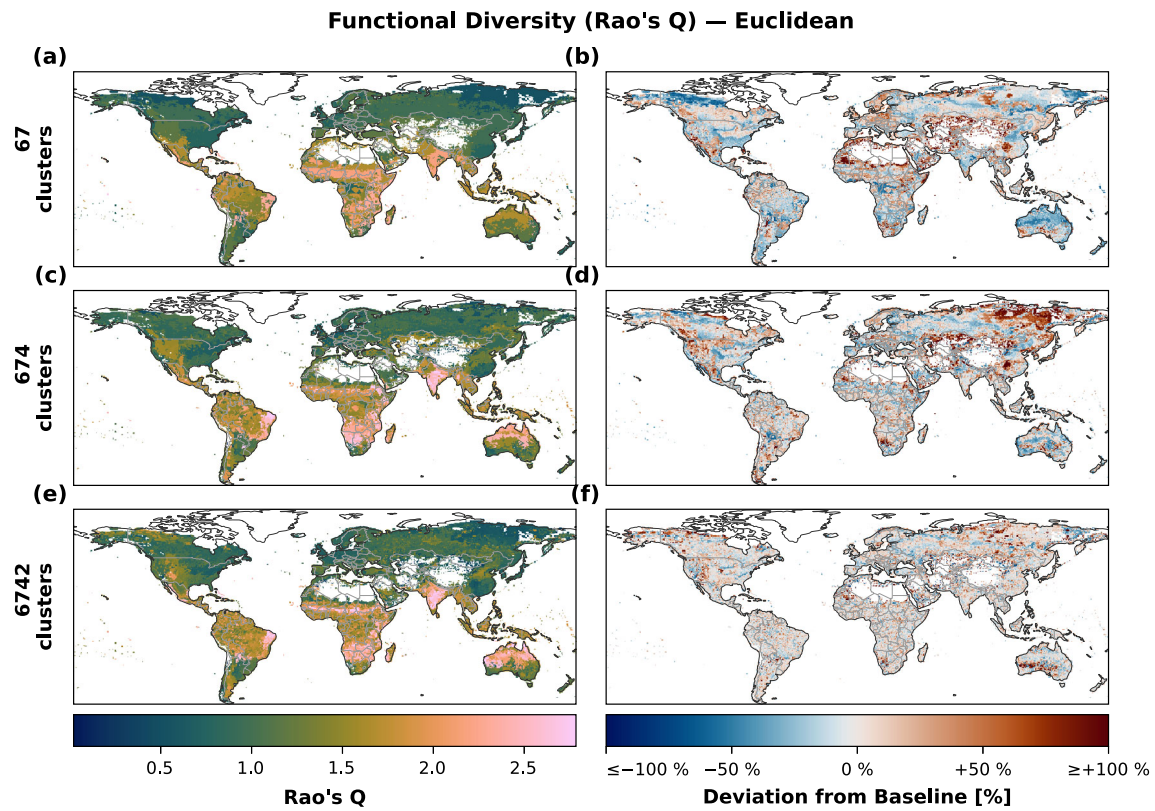


Fig. 15 Spatial distribution of reconstructed functional diversity (Rao's Q) and percentage deviation for the Euclidean metric at cluster counts $K \in \{67, 674, 6742\}$. Left column: time-mean Rao's quadratic entropy Q . Right column: percentage deviation from the cell-based baseline (Seed 1), capped at $\pm 100\%$

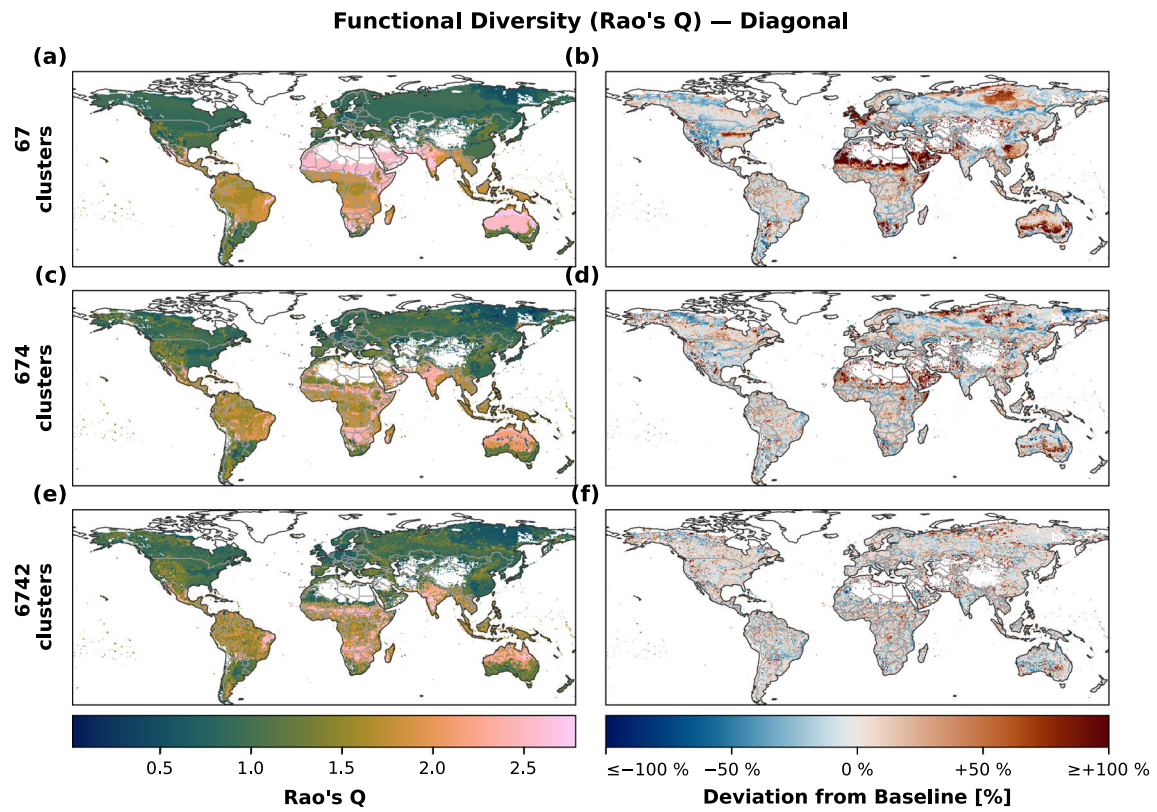


Fig. 16 Spatial distribution of reconstructed functional diversity (Rao's Q) and percentage deviation for the Diagonal metric at cluster counts $K \in \{67, 674, 6742\}$. Layout as in Fig. 15

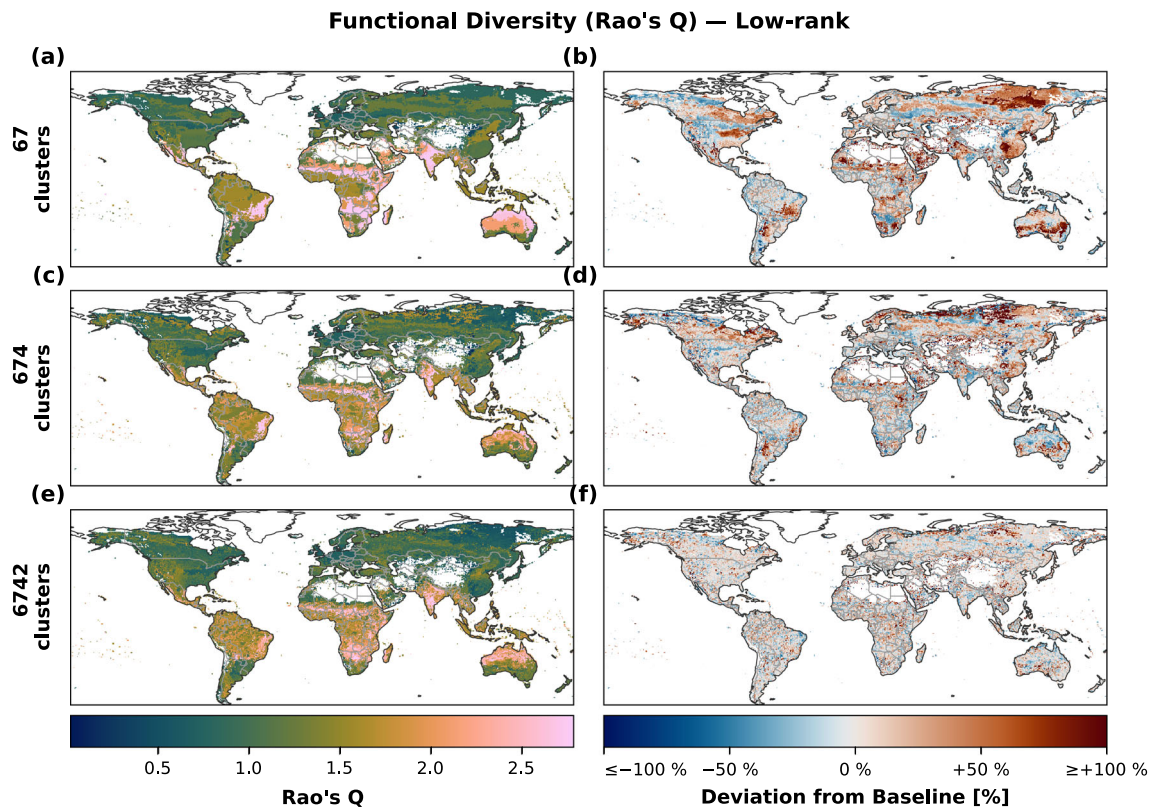


Fig. 17 Spatial distribution of reconstructed functional diversity (Rao's Q) and percentage deviation for the Low-rank metric at cluster counts $K \in \{67, 674, 6742\}$. Layout as in Fig. 15

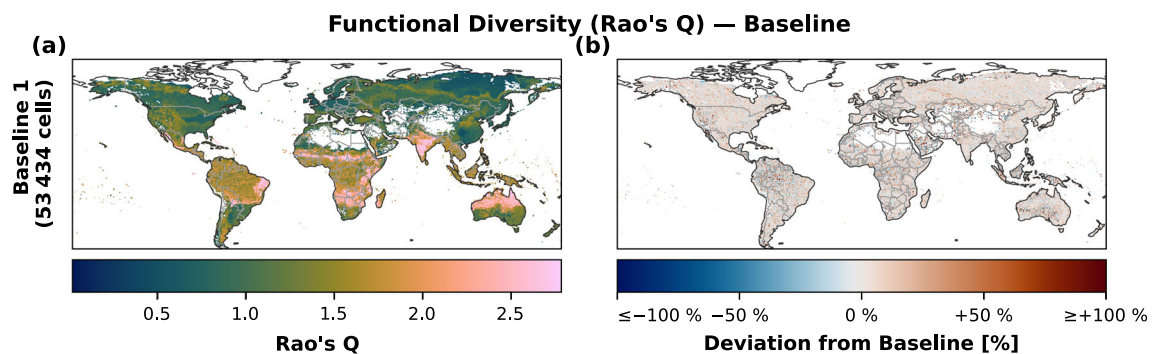


Fig. 18 Baseline functional diversity (Rao's Q) and inter-seed deviation. Left: time-mean Rao's Q from the cell-based baseline simulation (Seed 1; 53 434 valid cells). Right: percentage deviation of an independent cell-based simulation with a different random seed (Seed 2 minus Seed 1), quantifying the model-inherent stochastic noise floor

References

1. I.C. Prentice, A. Bondeau, W. Cramer, *et al.* Dynamic Global Vegetation modeling: Quantifying terrestrial ecosystem responses to large-scale environmental change. In: Canadell, J.G., Pataki, D.E., Pitelka, L.F. (eds.) *Terrestrial Ecosystems in a Changing World*, pp. 175–192. Springer, Berlin, Heidelberg (2007). https://doi.org/10.1007/978-3-540-32730-1_15
2. S. Sitch, C. Huntingford, N. Gedney *et al.*, Evaluation of the terrestrial carbon cycle, future plant geography and climate-carbon cycle feedbacks using five Dynamic Global Vegetation Models (DGVMs). *Glob. Change Biol.* **14**(9), 2015–2039 (2008). <https://doi.org/10.1111/j.1365-2486.2008.01626.x>
3. S. Schaphoff, W. Bloh, A. Rammig *et al.*, LPJmL4—a dynamic global vegetation model with managed land: Part I—model description. *Geoscientific Model Development* **11**(4), 1343–1375 (2018). <https://doi.org/10.5194/gmd-11-1343-2018>

4. B. Sakschewski, W. Bloh, A. Boit et al., Leaf and stem economics spectra drive diversity of functional plant traits in a dynamic global vegetation model. *Glob. Change Biol.* **21**(7), 2711–2725 (2015). <https://doi.org/10.1111/gcb.12870>
5. K. Thonicke, M. Billing, W. Bloh et al., Simulating functional diversity of European natural forests along climatic gradients. *J. Biogeogr.* **47**(5), 1069–1085 (2020). <https://doi.org/10.1111/jbi.13809>
6. S. Scheiter, L. Langan, S.I. Higgins, Next-generation dynamic global vegetation models: learning from community ecology. *New Phytol.* **198**(3), 957–969 (2013). <https://doi.org/10.1111/nph.12210>
7. B. Sakschewski, W. Bloh, A. Boit, Resilience of Amazon forests emerges from plant trait diversity. *Nat. Clim. Chang.* **6**, 1032–1036 (2016). <https://doi.org/10.1038/nclimate3109>
8. M. Billing, K. Thonicke, B. Sakschewski, W. Bloh, A. Walz, Future tree survival in European forests depends on understorey tree diversity. *Sci. Rep.* **12**, 20750 (2022). <https://doi.org/10.1038/s41598-022-25319-7>
9. M. Billing, B. Sakschewski, W. Bloh, J. Vogel, K. Thonicke, ‘How to adapt forests?’—Exploring the role of leaf trait diversity for long-term forest biomass under new climate normals. *Glob. Change Biol.* **30**(4), 17258 (2024). <https://doi.org/10.1111/gcb.17258>
10. B. Smith, I.C. Prentice, M.T. Sykes, Representation of vegetation dynamics in the modelling of terrestrial ecosystems: comparing two contrasting approaches within European climate space. *Glob. Ecol. Biogeogr.* **10**(6), 621–637 (2001). <https://doi.org/10.1046/j.1466-822X.2001.t01-1-00256.x>
11. B. Smith, D. Wårlind, A. Arneth et al., Implications of incorporating N cycling and N limitations on primary production in an individual-based dynamic vegetation model. *Biogeosciences* **11**(7), 2027–2054 (2014). <https://doi.org/10.5194/bg-11-2027-2014>
12. P.R. Moorcroft, G.C. Hurtt, S.W. Pacala, A method for scaling vegetation dynamics: the Ecosystem Demography model (ED). *Ecol. Monogr.* **71**(4), 557–586 (2001). <https://doi.org/10.2307/3100036>
13. description and evaluation of an individual-based forest simulator, M. Longo, R.G. Knox, D.M. Medvigy, Ecosystem demography model (ED v3.0). *Geoscientific Model Development* **12**, 4309–4346 (2019). <https://doi.org/10.5194/gmd-12-4309-2019>
14. C.D. Koven, R.G. Knox, R.A. Fisher et al., Benchmarking and parameter sensitivity of physiological and vegetation dynamics using the Functionally Assembled Terrestrial Ecosystem Simulator (FATES) at Barro Colorado Island, Panama. *Biogeosciences* **17**(11), 3017–3044 (2020). <https://doi.org/10.5194/bg-17-3017-2020>
15. R. Fischer, F. Bohn, M.D. Paula, Lessons learned from applying a forest gap model to understand ecosystem and carbon dynamics of complex tropical forests. *Ecol. Model.* **326**, 124–133 (2016). <https://doi.org/10.1016/j.ecolmodel.2015.11.018>
16. I. Maréchaux, J. Chave, An individual-based forest model to jointly simulate carbon and tree diversity in Amazonia: description and applications. *Ecol. Monogr.* **87**(4), 632–664 (2017). <https://doi.org/10.1002/ecm.1271>
17. R.A. Fisher, C.D. Koven, W.R.L. Anderegg et al., Vegetation demographics in Earth System Models: A review of progress and priorities. *Glob. Change Biol.* **24**(1), 35–54 (2018). <https://doi.org/10.1111/gcb.13910>
18. M. Paula, M. Forrest, L. Langan, Nutrient cycling drives plant community trait assembly and ecosystem functioning in a tropical mountain biodiversity hotspot. *New Phytol.* **232**, 551–566 (2021). <https://doi.org/10.1111/nph.17600>
19. J. Priesner, B. Sakschewski, M. Billing et al., What if extreme droughts occur more frequently? Mechanisms and limits of forest adaptation in pine monocultures and mixed forests in Berlin-Brandenburg, Germany. *Nat. Hazard.* **25**(9), 3309–3331 (2025). <https://doi.org/10.5194/nhess-25-3309-2025>
20. A.P.K. Argles, J.R. Moore, P.M. Cox, Dynamic Global Vegetation Models: Searching for the balance between demographic process representation and computational tractability. *PLOS Climate* **1**(9), 0000068 (2022). <https://doi.org/10.1371/journal.pclm.0000068>
21. M.C. Acosta, S. Palomas, S.V. Paronuzzi Ticco et al., The computational and energy cost of simulation and storage for climate science: lessons from CMIP6. *Geoscientific Model Development* **17**(8), 3081–3098 (2024). <https://doi.org/10.5194/gmd-17-3081-2024>
22. C.A. Silva, R. Vilaça, A. Pereira, R.J. Bessa, A review on the decarbonization of high-performance computing centers. *Renew. Sustain. Energy Rev.* **189**, 114019 (2024). <https://doi.org/10.1016/j.rser.2023.114019>
23. K. Ma, Y. Zhou, A comprehensive quantitative lifecycle cost and environmental impact analysis model for computing infrastructure. *MethodsX* **13**, 103009 (2024). <https://doi.org/10.1016/j.mex.2024.103009>
24. C.B. Staff, Researchers: The barriers to climate science in the global south (2021). <https://www.carbonbrief.org/researchers-the-barriers-to-climate-science-in-the-global-south/> Accessed 2026-04-29
25. L. Lannelongue, J. Grealey, M. Inouye, Green Algorithms: Quantifying the Carbon Footprint of Computation. *Advanced Science* **8**(12), 2100707 (2021). <https://doi.org/10.1002/adv.202100707>. _eprint: <https://advanced.onlinelibrary.wiley.com/doi/pdf/10.1002/adv.202100707> Accessed 2026-04-29
26. T. Schneider, L.R. Leung, R.C.J. Wills, Opinion: Optimizing climate models with process knowledge, resolution, and artificial intelligence. *Atmos. Chem. Phys.* **24**(12), 7041–7062 (2024). <https://doi.org/10.5194/acp-24-7041-2024>
27. C. Deser, F. Lehner, K.B. Rodgers et al., Insights from Earth system model initial-condition large ensembles and future prospects. *Nat. Clim. Chang.* **10**(4), 277–286 (2020). <https://doi.org/10.1038/s41558-020-0731-2>
28. D.I. Armstrong McKay, A. Staal, J.F. Abrams, et al. Exceeding 1.5°C global warming could trigger multiple climate tipping points. *Science* **377**(6611), 7950 (2022) <https://doi.org/10.1126/science.abn7950>
29. P. Das, M.D. Behera, P.S. Roy, S.K. Barik, Predicting tipping points of vegetation resilience as a response to precipitation: Implications for understanding impacts of climate change in India. *Biodivers. Conserv.* **33**(12), 3441–3458 (2024). <https://doi.org/10.1007/s10531-024-02804-1>

30. D. Otryakhin, D.M. Belda, A. Arneth, Comparison of simulations from a state-of-the-art dynamic global vegetation model (LPJ-GUESS) driven by low- and high-resolution climate data. *Geoscientific Model Development* **18**(22), 9101–9118 (2025). <https://doi.org/10.5194/gmd-18-9101-2025>
31. J.P. Dietrich, A. Popp, H. Lotze-Campen, Reducing the loss of information and gaining accuracy with clustering methods in a global land-use model. *Ecol. Model.* **263**, 233–243 (2013). <https://doi.org/10.1016/j.ecolmodel.2013.05.009>
32. A. Shivakumar, T. Alfstad, T. Niet, A clustering approach to improve spatial representation in water-energy-food models. *Environ. Res. Lett.* **16**(11), 114027 (2021). <https://doi.org/10.1088/1748-9326/ac2ce9>
33. U. Ehret, R. Pruijssen, M. Bortoli, R. Loritz, E. Azmi, E. Zehe, Adaptive clustering: reducing the computational costs of distributed (hydrological) modelling by exploiting time-variable similarity among model elements. *Hydrol. Earth Syst. Sci.* **24**, 4389–4411 (2020). <https://doi.org/10.5194/hess-24-4389-2020>
34. N.W. Chaney, M.H.J. Huijgevoort, E. Shevliakova, Harnessing big data to rethink land heterogeneity in Earth and environmental system models. *Hydrol. Earth Syst. Sci.* **22**, 3311–3330 (2018). <https://doi.org/10.5194/hess-22-3311-2018>
35. R.R. Renne, D.R. Schlaepfer, K.A. Palmquist, W.K. Lauenroth, J.B. Bradford, Estimating multivariate ecological variables at high spatial resolution using a cost-effective matching algorithm. *Ecosphere* **15**(3), 4811 (2024). <https://doi.org/10.1002/ecs2.4811>
36. F.I. Woodward, B.G. Williams, Climate and plant distribution at global and local scales. *Vegetatio* **69**, 189–197 (1987). <https://doi.org/10.1007/BF00038700>
37. A.W.R. Seddon, M. Macias-Fauria, P.R. Long, D. Benz, K.J. Willis, Sensitivity of global terrestrial ecosystems to climate variability. *Nature* **531**(7593), 229–232 (2016). <https://doi.org/10.1038/nature16986>
38. F. Jonard, S. De Canniere, A.F. Feldman, Observed water and light limitation across global ecosystems. *Biogeosciences* **19**, 5575–5590 (2022). <https://doi.org/10.5194/bg-19-5575-2022>
39. K.Q. Weinberger, L.K. Saul, Distance metric learning for large margin nearest neighbor classification. *J. Mach. Learn. Res.* **10**, 207–244 (2009)
40. M. Kottek, J. Grieser, C. Beck, B. Rudolf, F. Rubel, World map of the Köppen-Geiger climate classification updated. *Meteorol. Z.* **15**(3), 259–263 (2006). <https://doi.org/10.1127/0941-2948/2006/0130>
41. H.E. Beck, N.E. Zimmermann, T.R. McVicar, N. Vergopolan, A. Berg, E.F. Wood, Present and future Köppen-Geiger climate classification maps at 1-km resolution. *Scientific Data* **5**, 180214 (2018). <https://doi.org/10.1038/sdata.2018.214>
42. M.C. Peel, B.L. Finlayson, T.A. McMahon, Updated world map of the Köppen-Geiger climate classification. *Hydrol. Earth Syst. Sci.* **11**(5), 1633–1644 (2007). <https://doi.org/10.5194/hess-11-1633-2007>
43. F. Murtagh, P. Legendre, Ward's hierarchical agglomerative clustering method: which algorithms implement Ward's criterion? *J. Classif.* **31**(3), 274–295 (2014). <https://doi.org/10.1007/s00357-014-9161-z>
44. D. Müllner, Modern hierarchical, agglomerative clustering algorithms. *arXiv preprint arXiv:1109.2378* (2011) <https://doi.org/10.48550/arXiv.1109.2378>
45. S. Lange, M. Mengel, S. Treu, M. Büchner, ISIMIP3a atmospheric climate input data (v1.2). *ISIMIP Repository* (2023), <https://doi.org/10.48364/ISIMIP.982724.2>
46. J.D. Pelletier, P.D. Broxton, P. Hazenberg et al., A gridded global data set of soil, intact regolith, and sedimentary deposit thicknesses for regional and global land surface modeling. *Journal of Advances in Modeling Earth Systems* **8**(1), 41–65 (2016). <https://doi.org/10.1002/2015MS000526>
47. FAO, IIASA, ISRIC, ISSCAS, JRC: Harmonized World Soil Database (HWSD), version 1.2. Technical report, FAO, Rome, Italy and IIASA, Laxenburg, Austria (2012). Accessed: 2026-04-14. <https://pure.iiasa.ac.at/id/eprint/17595/>
48. F. Pedregosa, G. Varoquaux, A. Gramfort et al., Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.* **12**, 2825–2830 (2011)
49. T. Akiba, S. Sano, T. Yanase, T. Ohta, M. Koyama, Optuna: A next-generation hyperparameter optimization framework. In: *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2623–2631 (2019). <https://doi.org/10.1145/3292500.3330701>
50. P. Virtanen, R. Gommers, T.E. Oliphant, SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nat. Methods* **17**(3), 261–272 (2020). <https://doi.org/10.1038/s41592-019-0686-2>
51. Potsdam Institute for Climate Impact Research: HPC – High Performance Computing. <https://www.pik-potsdam.de/en/institute/about/it-services/hpc>. Accessed: 2026-04-01 (2025)
52. H.W. Kuhn, The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly* **2**(1–2), 83–97 (1955). Reprinted in *Naval Research Logistics*, **52**(1), 7–21 (2005); <https://doi.org/10.1002/nav.20053>
53. S.R. Weiskopf, F. Isbell, M.I. Arce-Plata, M. Di Marco, M. Harfoot, J. Johnson, S.B. Lerman, B.W. Miller, T.L. Morelli, A.S. Mori, E. Weng, S. Ferrier, Biodiversity loss reduces global terrestrial carbon storage. *Nat. Commun.* **15**, 4354 (2024). <https://doi.org/10.1038/s41467-024-47872-7>
54. C.R. Rao, Diversity and dissimilarity coefficients: A unified approach. *Theor. Popul. Biol.* **21**(1), 24–43 (1982)
55. Z. Botta-Dukát, Rao's quadratic entropy as a measure of functional diversity based on multiple traits. *J. Veg. Sci.* **16**(5), 533–540 (2005). <https://doi.org/10.1111/j.1654-1103.2005.tb02393.x>
56. GitHub, Inc. GitHub Copilot. <https://github.com/features/copilot>. Accessed: 2026-04-18 (2025)
57. F.I. Woodward, *Climate and Plant Distribution* (Cambridge University Press, Cambridge, 1987)
58. R.H. Whittaker, *Communities and Ecosystems*, 2nd edn. (Macmillan, New York, 1975)
59. D.M. Olson, E. Dinerstein, E.D. Wikramanayake et al., Terrestrial ecoregions of the world: a new map of life on Earth. *Bioscience* **51**(11), 933–938 (2001). [https://doi.org/10.1641/0006-3568\(2001\)051\[0933:TEOTWA\]2.0.CO;2](https://doi.org/10.1641/0006-3568(2001)051[0933:TEOTWA]2.0.CO;2)

60. E. Dinerstein, D. Olson, A. Joshi et al., An ecoregion-based approach to protecting half the terrestrial realm. *Bioscience* **67**(6), 534–545 (2017). <https://doi.org/10.1093/biosci/bix014>
61. C.R. Harris, K.J. Millman, S.J. Walt et al., Array programming with NumPy. *Nature* **585**(7825), 357–362 (2020). <https://doi.org/10.1038/s41586-020-2649-2>
62. A. Paszke, S. Gross, F. Massa, *et al.* PyTorch: An imperative style, high-performance deep learning library. In: *Advances in Neural Information Processing Systems*, pp. 8024–8035 (2019). <https://doi.org/10.48550/arXiv.1912.01703>
63. S. Hoyer, J. Hamman, xarray: N-D labeled arrays and datasets in Python. *Journal of Open Research Software* **5**(1), 10 (2017). <https://doi.org/10.5334/jors.148>
64. W. McKinney, Data structures for statistical computing in Python. In: *Proceedings of the 9th Python in Science Conference*, pp. 56–61 (2010). <https://doi.org/10.25080/majora-92bf1922-00a>
65. J.D. Hunter, Matplotlib: A 2D graphics environment. *Computing in Science & Engineering* **9**(3), 90–95 (2007). <https://doi.org/10.1109/mcse.2007.55>
66. Met Office: Cartopy: A Cartographic Python Library. (2015). Accessed: 2026-04-14. <https://scitools.org.uk/cartopy>
67. Anthropic: Claude – AI Assistant by Anthropic. <https://www.anthropic.com/claude>. Accessed: 2026-04-18 (2025)
68. Consensus NLP, Inc.: Consensus – AI-Powered Academic Search. <https://consensus.app>. Accessed: 2026-04-18 (2025)
69. J.V. Davis, B. Kulis, P. Jain, S. Sra, I.S. Dhillon, Information-theoretic metric learning. In: *Proceedings of the 24th International Conference on Machine Learning (ICML)*, pp. 209–216 (2007). <https://doi.org/10.1145/1273496.1273523>
70. I. Loshchilov, F. Hutter, Decoupled weight decay regularization. In: *International Conference on Learning Representations* (2019). Accessed: 2026-04-14. <https://openreview.net/forum?id=Bkg6RiCqY7>
71. NVIDIA Corporation: NVIDIA H100 Tensor Core GPU. <https://www.nvidia.com/en-us/data-center/h100/>. Accessed: 2026-04-20 (2024)