ChatGPT

Robust Strategies for Geographic Holdout in Spatial Model Training

Introduction: The Challenge of Spatial Overfitting

Training and evaluating classification models on geospatial data require special care due to spatial autocorrelation. Nearby locations tend to have similar features and classes, violating the assumption of independent training and test samples. Traditional random hold-out splits or cross-validation can produce **misleadingly optimistic accuracy** because the test data are often spatially intermingled with training data [1] [2]. Studies have shown that randomly splitting spatial data can **strongly overestimate model performance** and fail to reflect how well the model would transfer to new regions [2]. In ecological modeling, for example, random splits inflated accuracy and did not predict independent "out-of-region" performance [2]. The core issue is that spatial dependence means a model can effectively "peek" at nearby test points via training points in their vicinity [3] [4]. This leads to overfitting and overly complex models that appear accurate but won't generalize well beyond the sampled area [5].

To obtain **defensible estimates of generalization**, experts recommend spatially structured holdouts that approximate truly independent data [6]. Ideally, one would test models on completely independent regions or times [7] [6]. Since such independent datasets are rarely available, we mimic them by holding out entire geographic regions or using spatial cross-validation. This report surveys the most statistically robust methods for defining geographic holdout areas in large, high-resolution domains (e.g. geological province mapping), addressing key design questions. It synthesizes peer-reviewed literature and methodological best practices to guide **reproducible, unbiased model evaluation** under worst-case spatial overfitting scenarios.

Contiguous vs. Discontiguous Holdout Regions

A fundamental decision is whether to make the held-out areas one or more **contiguous blocks** or to use **discontiguous (scattered) holdouts**. Contiguous spatial holdouts (e.g. leaving out one large region) force testing on points that are far from any training point, thus greatly reducing spatial dependence between train and test sets [8]. By withholding a continuous area, you ensure test samples are **spatially distant** from training samples, increasing independence and reducing optimistic bias [8]. For example, holding out an entire province or subregion means the model is truly extrapolating to that region with no nearby training data [9]. This approach is **statistically stringent** – it emulates the scenario of applying a model in a new geographic region, thus providing a robust check on overfitting.

Discontiguous holdouts involve selecting multiple smaller areas or points spread across the map for testing (often ensuring a minimum distance between train and test). One implementation is a **checkerboard or grid-based partition**, where the study area is divided into many blocks and assigned to folds in an alternating pattern (somewhat discontiguous) [10]. Another variant is clustering points into groups that are not single contiguous region but still separated by distance. The advantage of discontiguous holdouts is that each fold (or test set) can sample the full range of environments in the study area, improving the

representativeness of each fold. It produces a more stable estimate of overall accuracy by averaging over many small gaps rather than one big gap. However, if the holdout patches are too small or too close to training data, spatial leakage can reoccur at the edges of each patch. **Spatial block cross-validation** methods often strike a balance: they divide the area into many blocks, then either (a) **group contiguous blocks** to form one large test region per fold, or (b) **assign non-contiguous blocks to folds** in a way that increases separation [10].

Notably, **Roberts et al. (2017)** demonstrate that grouping blocks into one contiguous test region per fold yields folds that are very different from each other in environmental space (simulating an **extrapolation test**) [10]. In contrast, systematically assigning smaller dispersed blocks to each fold makes the folds more similar in environmental distribution (more like repeated **interpolation tests**) [11]. Contiguous holdouts maximize the dissimilarity between training and testing sets, which is useful to **detect overfitting** and assess worst-case transferability [12]. Discontiguous holdouts (e.g. a checkerboard pattern), while still enforcing some distance, tend to yield training–testing splits that cover similar environmental gradients, thus providing a less extreme but still independent evaluation [11].

Recommendation: For strict **statistical validity**, contiguous holdout regions are often preferred because they clearly break the spatial dependence between train and test [8]. A large contiguous test area is a defensible choice to demonstrate the model's ability to generalize to wholly unseen locations. However, a single contiguous holdout can introduce high variance in estimates (results may depend on which region was held out). Thus, for **reproducibility**, many studies employ *k*-fold spatial cross-validation: repeating the process with multiple different contiguous regions as test folds and averaging performance [11]. This yields a more stable assessment while retaining the independence of contiguous splits. In practice, one might partition the map into, say, 4–5 large contiguous regions and perform cross-validation such that each region gets to be the holdout once. This approach combines the rigor of contiguous holdouts with the robustness of multiple tests.

On the other hand, if the goal is to estimate performance under more average conditions (interpolation within the overall domain), a discontiguous or systematic block approach can be used, but care must be taken to maintain sufficient distance between train and test blocks. The **key** is that any holdout scheme must avoid test points having nearby training points; whether contiguity is required depends on data and goals. When in doubt, err on the side of **larger, contiguous withheld areas** to ensure independence [8], and use multiple folds or repeats to mitigate variability.

Aligning Holdouts with the Principal Axis of Variation

Another consideration is whether the holdout regions should be oriented or **rotated** to align with the study area's principal geographic or environmental axis. In some datasets, there is a dominant spatial gradient (e.g. north–south climate gradient, or east–west geological trend). One might rotate or shape the holdout area so that its orientation matches this axis, under the intuition that this captures the major source of variation. For example, if geological provinces stretch northeast–southwest, a holdout block aligned along that same axis might encompass a full cross-section of one province rather than cutting across provinces arbitrarily.

Literature guidance on rotation is limited, and evidence suggests that the **exact shape or orientation of blocks is less critical than their size** and separation [13]. A recent simulation study (Stock et al. 2025) varied block shape (e.g. square vs. elongated vs. irregular) and found shape had only *minor effects* on error

estimates compared to block size [13]. The most important factor was ensuring blocks are large enough to capture spatial autocorrelation ranges; whether blocks were rotated or not did not significantly change the outcomes in their tests [13]. Thus, from a statistical standpoint, it is usually **not necessary to rotate holdout grids**, unless a rotated grid better fits natural subregions of the landscape.

That said, aligning holdout regions with meaningful geographic features or axes can improve interpretability and relevance. If the study area has an obvious directional anisotropy (e.g. environmental gradients run east-west), then defining holdout blocks aligned along that direction can ensure each block experiences a distinct portion of that gradient. Some workflows choose a block layout that follows the **principal axis of spatial structure** so that each fold corresponds to a slice of the landscape in a coherent way. This can be helpful if one wants to say, *"we withheld the northernmost 20% of the area"* or *"we withheld a western transect of the region"*, aligning with how the model might be applied in practice (such as predicting further northward or westward).

Recommendation: In general, **prioritize block size and separation over orientation**. Use block shapes that are convenient (squares or rectangles are common) and sufficiently large. Only consider rotation if the un-rotated grid would cut through known natural units or if the study's main gradient is diagonal such that an axis-aligned block would mix very different environments. If rotating the holdout regions can align them with known geographic or geological features (e.g. aligning with mountain range direction or stratigraphic zones), it may enhance the meaningfulness of the holdout, but it is not strictly required for statistical validity [13]. Any rotation should still preserve contiguity and adequate size of holdouts. In summary, **rotating holdouts is an optional refinement** – standard unrotated blocks are usually sufficient and easier to implement, as evidenced by minimal impact of block shape in studies [13].

Placement of Holdout Blocks: Center vs. Edge vs. Random Location

Should the holdout region be in the **center** of the study area, at an **edge**, or somewhere random? This choice can affect both realism and bias. There is no one-size-fits-all rule, but we can consider the implications:

- **Central Holdout:** If a block is withheld in the interior of the map (surrounded by training data on all sides), the model is tested on an "island" of unseen area in the middle of a well-sampled region. This creates many training/test boundaries around the holdout, which increases the chance of **edge effects** (since the test pixels border training pixels along all edges of the holdout). A central holdout simulates an unsampled "hole" in an otherwise sampled landscape. This might be appropriate if, for instance, a particular central zone had no data during training and one wants to predict into it. However, it is somewhat *artificial* if the training dataset actually had samples throughout the area – intentionally removing a center chunk may not correspond to a real use-case unless data were truly missing there.

- **Edge Holdout:** Withholding a block at the periphery (e.g. the entire east edge or a corner of the domain) means the model is tested on the extreme end of the spatial extent. This often represents a more **realistic extrapolation scenario**: for example, training on the interior of a region and then applying the model outward beyond the surveyed area. An edge holdout has fewer adjacent training areas (one side of the test region is just the boundary of the study area), which can reduce the length of train-test contact boundary. If environmental conditions vary across the region (say a gradient from west (dry) to east (wet)), holding out the extreme wet eastern edge forces the model to predict

in a climate not seen in training – a tough but informative test. Edge holdouts are common when researchers specifically want to assess **transferability to new regions** contiguous to the study area [9]. For example, Radosavljevic and Anderson (2014) evaluated species distribution models by training in one part of a species range and testing on another disjoint part [14]. Such approaches effectively use one region (often an edge or separate range) as the holdout to examine model generalization.

- **Randomly Located Block:** Choosing a holdout region at random (or using multiple random blocks in cross-validation) can avoid subjective bias in placement. If done repeatedly, random placement allows averaging over many possible holdout locations, which is good for **robustness**. On any single random split, however, the difficulty of the task could vary – a random block might by chance be an easy region (e.g. very similar to the training area around it) or a very challenging one (if it encompasses unique terrain). To ensure fairness, one could repeat with several random block placements and report the range of outcomes. In practice, many adopt a **spatial *k*-fold CV** where the folds are essentially randomly placed blocks (subject to contiguity or distance constraints) [11]. This yields an ensemble of holdout positions (some central, some edge, some in-between) and the average performance is taken. Stock et al. (2025) found that the *assignment* of blocks to folds (i.e. which region is test vs train) had only a minor effect on error estimates when averaged out [13]. This suggests no inherent bias for center vs edge if multiple folds are used; what matters more is using **sufficiently large and well-separated blocks** in whatever positions they are [15].

Recommendation: Align the holdout placement with the **study goals and data distribution**. If the likely application of the model is to predict beyond the current spatial extent, an **edge or external holdout** is logical and defensible. If instead the concern is unsampled pockets within the area, an **internal holdout** might be warranted. In many cases, especially for large domains, it's wise to do **several different holdout placements (e.g. via cross-validation)** to ensure the conclusions aren't an artifact of a particular region's idiosyncrasies. Leading studies often emphasize withholding *meaningful subregions*: for instance, Stock et al. (2025) concluded the best strategy was leaving out entire **sub-basins** of their study area – essentially using natural divisions as holdouts [16]. Following this principle, if your domain has known subdivisions (geological provinces, eco-regions, basins), consider using those as holdouts rather than an arbitrary square in the middle. This enhances interpretability and realism: the model is tested on a whole logical unit it wasn't trained on, which stakeholders can relate to. Ultimately, **any placement is acceptable if justified**, but you should document why that region was chosen and ideally demonstrate that results are consistent under different plausible holdout locations.

Holdout Size and Proportion: Is 25% the Magic Number?

Using 25% of data or area for holdout is a common heuristic (e.g. a 75/25 train-test split or 4-fold cross-validation). However, there is nothing sacrosanct about 25%. The optimal or appropriate **holdout proportion depends on the data size, autocorrelation range, and the desired confidence in error estimates**.

Conventional practice: In machine learning at large, reserving about 20–30% of data for testing is routine. This is often a compromise between having enough training data to build a good model and enough test data to reliably estimate performance. Many geospatial studies have followed suit with roughly one-quarter of locations or area as independent test [14]. For example, a study might train on 75% of the study area and evaluate on a contiguous 25% region withheld. If doing spatial *k*-fold CV, using 4 folds yields 25% per fold,

or 5 folds yields 20% per fold, both commonly seen. **Thus, 25% is more of a convention than a statistically "optimal" fraction.**

Insights from literature: The **critical factor is not the exact percentage, but the size of holdout relative to spatial autocorrelation scale**. Stock et al. (2025) systematically varied the number of folds (which inversely changes holdout size) and found that, as long as blocks were large enough, the number of folds had only minor impact on error estimates [13]. In their synthetic tests, whether they used 2 folds (50% holdout each time), 5 folds (~20% each), or 10 folds (10% each) did not drastically change the averaged accuracy estimate – provided each test block was sufficiently big to be independent [13]. The **block size** itself was far more influential: too small a block (even if it's 10% of area) might still include points very near training points, giving optimistic results. Conversely, very large blocks (e.g. one entire half of the map as test) ensure independence but can be overly pessimistic, as the training set then covers a much smaller environmental range. Indeed, Stock et al. noted that **very large blocks sometimes led to overestimation of error (overly low accuracy)** in their experiments [17] – essentially, the model was challenged with a very different test set than it would normally face, making the task artificially difficult.

So, while **25% is a reasonable starting point**, there is no universal optimal fraction. If you have a huge dataset, you might afford a larger holdout (e.g. 30–40%) to really stress-test the model. If data are limited, you might use repeated *folding* to effectively test on more data while keeping each fold smaller. Some robust approaches use **repeated spatial splitting** (Monte Carlo spatial cross-validation): e.g. randomly hold out ~10% of locations as many times, ensuring each holdout meets distance criteria, then aggregate results. This yields a distribution of performance estimates for different ~10% test sets. The effective test coverage can be large (many points seen in some fold), but each individual test is small and independent. The trade-off is between **variance and bias** of the estimate: larger holdouts (fewer folds) give a very stringent test each time (high variance across folds, but low bias towards optimism), whereas smaller holdouts (more folds) give more stable but slightly less independent tests.

Recommendation: Ensure the **holdout region is big enough** to capture spatial structures – typically at least on the order of the autocorrelation range (e.g. if data are correlated up to ~10 km, a holdout block might be tens of kilometers wide). In practice, 20–30% of the area in each fold often achieves this, but check spatial correlograms of your data to guide the choice [17]. Using **25% is conventional and often works well**, but feel free to adjust: e.g. use 33% (three-fold CV) for very large heterogeneous areas to push the model harder, or 10% (ten-fold CV) if data are precious and you want to maximize training size while still rotating through test areas. If using 25%, know that it's not "magic" – complement it with sensitivity analysis (does using 20% or 30% change conclusions?) to strengthen your justification. The **key justifications** for any choice should be (a) independence (blocks are big enough), and (b) sufficient test sample size (the holdout has enough data to reliably measure performance). Many studies stick to ~25% simply because it balances those factors reasonably in typical cases. But an **alternative justification** could be scenario-driven: for example, "We held out one quadrant of the map (~25% of area) because that simulates leaving out an entire province," which is defensible beyond just the percentage.

In summary, **there is no strict optimal proportion**; 25% is a useful rule-of-thumb, but your context (data size and spatial structure) should dictate the holdout size. Always explain why your chosen fraction or fold count provides a robust test (e.g. "blocks of ~50 km, which is beyond the 30 km spatial correlation range, resulting in ~20% of points per fold"). If unsure, err toward a slightly larger holdout to avoid any lingering spatial dependence, as a slightly pessimistic error estimate is more defensible than an optimistic one.

Mitigating Spatial Autocorrelation and Edge Effects

Defining holdout regions in space introduces some pitfalls of its own, namely **spatial autocorrelation leakage across boundaries** and **edge effects**. Here are strategies to handle them:

- **Spatial Buffers:** A widely recommended practice is to place a buffer zone around test points or regions, within which training points are excluded [18] [19]. In other words, even outside the formal holdout area, one does not train on points that are very close to the test area boundary. The buffer width is often chosen based on the autocorrelation range of the data (e.g. if data are significantly correlated up to 5 km, then exclude training data within 5 km of any test point). This ensures **no training sample lies immediately adjacent to a test sample** [19]. For example, **leave-one-out with buffering** (sometimes called **h-block cross-validation** in geostatistics) holds one observation as test and drops all training points within a radius *h* around it [18]. More generally, with block holdouts, one can **expand the holdout area by a buffer**: do not use training data in a rim surrounding the test block. This buffer approach directly tackles the edge effect: it prevents the model from getting "too close" to the test locations. As a result, test predictions are truly independent, at the cost of discarding some data near the boundaries. Modern implementations (e.g. the R `blockCV` package) support spatial buffering, generating folds such that *the testing set never directly abuts training samples* [19].

- **Accounting for Autocorrelation in Block Size:** As noted earlier, using blocks larger than the autocorrelation distance inherently reduces the leakage. A variogram or Moran's I correlogram of predictor/response variables can guide the **minimum block size** [20]. If points 10 km apart are essentially independent, then blocks of size ~20 km or more will largely ensure independence without needing huge buffers. In practice, one might try a few block sizes and inspect whether cross-validation results stabilize once blocks exceed a certain size (this implies that residual spatial autocorrelation is addressed). Stock et al. (2025) explicitly recommend using **correlograms of predictors to choose a good block size** [17].

- **Overlap and Edge Correction:** When a holdout is in the interior, its edges face training data on all sides. This can create a situation where points just outside the holdout are in training, providing the model with almost-test information. Buffering addresses this, but one can also mitigate by how predictions are made. For instance, if evaluating a raster prediction, you could mask out a thin edge strip of the predicted holdout region to avoid including locations too influenced by immediately neighboring training data. Another approach is to use **smoothly decaying weights** near boundaries in certain kriging or autoregressive models, but for standard ML models, a hard buffer is simpler.

- **Multiple Discontiguous Buffers (Distance-Based CV):** An extreme approach to avoid any spatial autocorrelation is distance-based CV where each test point has a buffer and training comprises all points beyond that distance. This is essentially the buffered leave-one-out mentioned above [18]. Pohjankukka et al. (2017) and others found that this method yields much more realistic error estimates for spatial data [18]. The downside is it **uses only distant points for training each test case**, potentially underutilizing data. It's most feasible when data are abundant or one is focusing on local prediction error rather than building one global model.

In summary, **spatial buffering is a crucial technique** to handle edge effects. Figure 1 illustrates this concept schematically, showing that a buffer around test points means no training points fall within that buffer distance:

Spatial buffering concept: A buffer zone (gray band) around each test point or region is applied. Training samples within the buffer (red X's) are excluded, leaving only distant training data (green points) to fit the model. This prevents any training sample from being directly adjacent to a test sample [19].

By using buffers and sufficiently large holdouts, one can largely eliminate residual spatial autocorrelation between training and test sets. It's important to report the **buffer distance or how autocorrelation was accounted for**, as this improves the credibility of the evaluation. For example, *"We excluded training samples within 5 km of the holdout region boundary, corresponding to the range at which Moran's I fell to near zero"*. This communicates that you actively guarded against subtle spatial leakage. Leading workflows in ecology and remote sensing emphasize spatially independent evaluation either by blocking or buffering; failing to address it can lead to inflated performance metrics [4].

Practices in Leading Geospatial Fields

What do **gold-standard studies** in relevant domains do? Across geomorphometry, landscape ecology, and geospatial AI, there is a clear trend toward **spatially aware validation**:

- **Landscape Ecology / Species Distribution Modeling:** These communities were among the first to recognize the issue. Araújo and colleagues (2005) advocated testing models on independent regions (e.g. geographically distinct populations) to assess transferability [7]. More recently, Fourcade et al. (2018) demonstrated that machine-learning models that seemed superior under random CV lost their edge under spatially blocked evaluation – highlighting that many published models were overfit to spatial sampling artifacts [9]. The recommendation from such studies is to **divide the study area into a few spatially separated blocks** to truly test model transferability [2]. Likewise, Radosavljevic & Anderson (2014) introduced the idea of **"masked geographically structured"** evaluation for MaxEnt models, implemented in the ENMeval tool, including techniques like checkerboard partitioning and latitudinal band partitioning. Today, **spatial block cross-validation is increasingly standard** in species distribution modeling and ecology [4] [18]. Tools like `blockCV` in R (Valavi et al. 2019) and `CAST` (Meyer et al. 2018) were created to facilitate this. In fact, **Roberts et al. (2017)** provided an influential review in *Ecography* on structured CV for spatial (and temporal) data, which has been widely cited in ecology [4]. The bottom line: leading ecology papers now typically avoid naive random splits; reviewers often expect to see some spatially explicit validation or at least a justification for not doing so.

- **Geomorphometry / Environmental Mapping:** In geomorphology and Earth observation (e.g. soil mapping, landform classification, climate downscaling), researchers have also adopted spatial holdouts. For example, Hengl et al. (2018) in a global soil mapping study used spatial cross-validation to evaluate their models, finding much lower accuracy than random CV – a dose of reality that made their performance claims more credible. A study on global potential natural vegetation mapping reported an accuracy of ~68% with conventional validation but only ~33% with spatially explicit validation [21], reinforcing how challenging realistic prediction is. These fields often deal with strong spatial patterns, so **block or distance-based CV** is crucial. Geomorphometric classification (e.g. identifying landform units from DEMs) can suffer from overfitting if the training and test windows

are adjacent. Thus, state-of-the-art workflows use holdout blocks separated by distance or different regions (e.g. training on certain mapped quadrangles and testing on others). We see parallels with ecology: for instance, the concept of an **h-block CV** (withholding a region around each validation point) comes from geostatistics and is applied in environmental modeling to handle spatial autocorrelation [18].

- **Geospatial AI / Remote Sensing:** In the era of deep learning on satellite imagery, there has been a learning curve. Early studies sometimes reported very high accuracies using randomly sampled training/test pixels from the same image – inadvertently benefitting from spatial autocorrelation and even identical neighborhoods. However, awareness is rising. Recent work in GeoAI emphasizes testing on **holdout scenes or geographic folds** [4] [22]. For example, if training a CNN on aerial images for land cover, one might train on images from Region A and test on Region B. The *IEEE GeoAI Handbook* (2023) explicitly discusses spatial CV methods for exactly this reason [23] [24]. Researchers like Beery et al. (2018) showed that models can latch onto location-specific cues that don't generalize [25], so evaluating on a truly separate location is essential to reveal such issues. In high-resolution mapping of geological features, one best practice is to leave out whole map sheets or survey areas as test. The community is moving toward **benchmark datasets split by geography** (e.g. the DeepGlobe land cover challenge had held-out regions). In summary, the cutting edge of geospatial AI adopts the same principles: **spatially segregated evaluation** to avoid inflated performance [4]. Tools and libraries are being updated to support this (e.g. `spatialsample` in R, and scikit-learn extensions for spatial splitting).

To facilitate these best practices, several **open-source packages** and workflows have emerged. We've mentioned R's `blockCV` [26], which allows users to easily create spatial folds (blocked or buffered). Python's scikit-learn can be paired with custom cross-validator classes to achieve similar splits, and new libraries specific to spatial ML are under development. The existence of these tools underscores that **spatial holdout is considered a gold-standard approach** in modern geospatial modeling [26]. In many published studies, authors will explicitly state they performed spatial cross-validation or independent spatial holdouts, often citing the above literature to justify it. This lends credibility to their claims because it addresses the overfitting concern head-on.

In contrast, papers that use only random splits on spatial data increasingly face skepticism: reviewers may ask for additional spatial testing or downweight the reported accuracy. Thus, following these leading practices is important not just statistically but also for the **defensibility** of one's results in the eyes of the scientific community.

Comparison of Validation Methods and Their Use-Cases

To summarize the discussed methods, the table below compares key approaches to partitioning spatial data for model validation, along with their assumptions, strengths, and typical use cases:

Validation Method	Partitioning Strategy & Assumptions	Pros and Cons	Example Use Cases
Random Hold-out (Non-Spatial)	Randomly splits points or samples irrespective of location. Assumes data are i.i.d. (no spatial structure).	**Pros:** Maximizes training data overlap with feature space of test; easy to implement; low variance in performance estimate (every fold mixes data). **Cons:** Severely **biased if spatial autocorrelation** exists – overly optimistic accuracy [4]; not representative of new-location performance [2].	*Baseline or naive approach* when spatial independence truly holds (rare). Sometimes used for comparison purposes in literature (to show the contrast with spatial CV). Not recommended for final assessment in spatial studies.
Contiguous Block CV (Spatial Blocks)	Divides area into large contiguous blocks (often by grid or natural units). Each fold holds out one block (all points within). Assumes that blocking captures spatial dependence (points in other blocks are far enough to be independent).	**Pros:** Yields **spatially independent** test sets [8]; simulates extrapolation to an unseen region; helps detect overfitting clearly. **Cons:** Higher variance (performance can differ by which block is held out); if one block has unique conditions, results can fluctuate. Requires enough data to set aside entire regions.	*Geographic generalization tests* – e.g. train on one region, test on a disjoint region. **Ecology:** evaluating species models by withholding one region of occurrence [9]. **Remote sensing:** train on one satellite tile, test on another. Used when we want to rigorously test transferability or when deploying model to map new areas.

Validation Method	Partitioning Strategy & Assumptions	Pros and Cons	Example Use Cases
Discontiguous Blocks / Systematic Folds	Uses many smaller spatial blocks (e.g. grid cells) and assigns them to folds in a checkerboard or optimized pattern. Ensures a minimum separation between any train-test pairs but each fold contains multiple patches.	**Pros:** More **balanced representation** – each fold sees a spread of the area; performance is averaged over several dispersed gaps, reducing sensitivity to any one region. Often yields intermediate realism (reduces autocorrelation without extreme extrapolation). **Cons:** Somewhat less stringent than one big block – folds might still share some broad-scale environment overlap. Implementation can be complex (need to ensure distance criteria).	*Robust interpolation assessment* – e.g. **species distribution modeling** where checkerboard CV is used to account for spatial sorting bias while retaining more data [9]. **Land-cover mapping:** ensuring training and test pixels are from different parts of an image. Good for getting an overall accuracy that is less optimistic than random CV but still using all data in cross-validation.
Spatial Buffering (Distance-Based)	Not a partition per se, but a rule layered onto CV: impose a buffer (e.g. X km) around each test sample or region; exclude any training sample within that buffer [19]. Assumes a known distance beyond which data are approx independent. Often combined with LOOCV or block CV.	**Pros: Maximally breaks local autocorrelation** – no train sample is near a test sample [19]. Can use all data as test eventually (LOOCV style), yielding thorough use of dataset. **Cons:** Can be **data-hungry** – many samples get excluded (the buffer zones) reducing effective training data for each fold. Computationally intensive if leave-one-out. Possibly overly conservative if buffer distance is large.	*Fine-scale ecological modeling:* e.g. **habitat modeling** with presence/absence points – use buffered LOOCV to avoid any pseudo-replication [18]. *Geostatistics:* **h-block validation** for variogram modeling (leave out a region of radius *h* around each point). When data are dense and one wants to be absolutely sure of independence.

Validation Method	Partitioning Strategy & Assumptions	Pros and Cons	Example Use Cases
Environment/ Stratified CV	Splits data by an environmental or categorical factor rather than purely by space. For example, hold out all data in one environmental cluster or one geological class. Assumes that major differences are along that factor.	**Pros:** Tests model's ability to extrapolate to a new environmental domain (even if geographically interspersed). Useful if certain environments are under-sampled – can ensure they appear wholly in test folds. **Cons:** If environment clusters are spatially intermixed, this might not break spatial autocorrelation completely. Also, model might still leverage spatial trends within each cluster.	*Geomorphology:* e.g. classify landforms and hold out entire geomorphic provinces or climate zones to test broader generalization. *Climate modeling:* train on one climate regime, test on another. Typically supplementary to spatial CV, used to specifically probe extrapolation across environmental gradients.

Notes: All the spatial methods above aim to **simulate independent validation** as if one had a separate hold-out dataset from a new region [6]. The choice among them depends on data volume, the spatial structure of the problem, and the question of interest (interpolation accuracy vs. extrapolation robustness). Often, researchers will try multiple methods (e.g. both buffer LOOCV and block CV) to see how sensitive results are – if a model performs well across all, confidence in its generalizability is high. In any case, **random hold-out should be viewed as a worst-case (most optimistic) scenario** and not solely relied upon for claims about real-world performance [9] [4].

Recommendations for Robust and Reproducible Spatial Validation

Drawing on the above findings, here are **practical recommendations** for defining holdout areas in geospatial model training, aimed at maximizing robustness and defensibility:

- **Use Spatially Independent Splits by Default:** Whenever spatial autocorrelation is present (virtually always in high-res geodata), avoid pure random splits. Implement spatial holdouts – either contiguous block hold-outs or an appropriate spatial CV – to ensure train and test are well-separated [4]. This provides a more honest assessment of model generalization, as emphasized in numerous studies [2] [4].

- **Choose Holdout Strategy Based on Goal:** If your primary concern is **worst-case generalization** (model transfer to a novel area), favor **contiguous block holdouts** that mimic that scenario [10]. If you want a robust estimate of overall performance within the domain (accounting for spatial structure), consider **k-fold spatial CV with multiple blocks** or a **buffered CV** approach to average out variability.

- **Determine Block Size via Spatial Range:** Use exploratory spatial analysis (Moran's I, variograms) on your data to estimate how far autocorrelation extends. Define holdout blocks (or buffer distances) at least as large as this range [17]. For example, if elevation values or class labels are correlated over ~5 km, make sure test blocks are larger than 5 km across (and/or use a 5 km buffer) so that training points outside cannot "inform" test points [17]. This will justify that your splits achieve near-independence.

- **25% Test is a Guideline, Not a Hard Rule:** Using ~20–25% of data for testing is common and usually reasonable, but **justify this choice**. Explain in terms of block size (e.g. "this corresponds to holding out a 50×50 km area, which covers ~25% of our region"). If data are highly abundant, you might set aside an even larger area for testing (for a tougher challenge), whereas if data are scarce, use repeated CV to make the most of it. The key is demonstrating that your results are **not sensitive to a specific split**. If possible, report additional experiments (e.g. another region as holdout, or different fold counts) to show consistency. This enhances reproducibility and defensibility.

- **Apply Buffers to Avoid Edge Leakage:** Incorporate a **buffer zone around test areas** [19]. Many tools automate this; if not, you can manually remove training samples within a certain distance from test points. This greatly reduces the chance that the model's predictions in the holdout were indirectly "seen" during training. It tackles the subtle overfitting at boundaries and is considered best practice in spatial validation [18]. Document the buffer distance and rationale (e.g. "no training point within 10 km of a test point"). This detail signals rigor to reviewers and readers.

- **Use Multiple Folds or Repeated Holds for Reproducibility:** A single holdout region could, by luck of the draw, be especially easy or hard for the model. To make results robust, use either *k*-fold spatial cross-validation (with k ~3–10, as data allows) [27] or multiple random block holdouts. This yields a distribution of performance estimates. If your model consistently performs well across all spatial folds, you can confidently claim it generalizes, which is very defensible. If performance varies, report that variability – it reflects uncertainty in spatial generalization. Reproducible science favors **averaging over multiple runs** rather than basing conclusions on one geographic split.

- **Mimic Real Deployment Scenarios:** Design holdouts that resemble how the model will be applied. For instance, if you intend to use a geologic classifier in a neighboring region with similar orientation, hold out the fringe of your study area in that direction. If the model might face entirely new province types, hold out a whole province. This scenario-driven approach was highlighted by Stock et al. (2025) — the best block strategy "reflected the data and application" by leaving out whole meaningful subregions [16]. By aligning validation with use-case, you bolster the argument that your evaluation measures **realistic generalization performance**.

- **Report and Address Limitations Openly:** Even with these robust methods, acknowledge remaining limitations. For example, if parts of the study area had no data at all (unsampled conditions), note that performance in those truly novel conditions is still unknown [28]. Recognize that spatial CV "reduces but does not eliminate" all biases (models can still overfit in other ways) [29]. By discussing this, you demonstrate a balanced, defensible stance. Claims about model accuracy should be tempered with the spatial context: e.g. *"Our model achieved 85% overall accuracy when tested on provinces not used in training, suggesting strong generalizability within the study region [9]. However, unrepresented environmental extremes could pose further challenges."* This level of nuance is expected in top-tier studies.

In conclusion, **defensible geospatial modeling requires careful validation design**. Holdout areas should be defined with spatial independence in mind – typically as contiguous blocks or well-separated sets of points – and sized according to the data's autocorrelation structure. Whether contiguous or discontiguous, centered or edged, the holdouts must genuinely test the model's ability to predict beyond its training vicinity. By following the practices of leading fields (ecology, remote sensing, geomorphometry) – who increasingly use spatial blocking, buffering, and multiple-fold evaluations – one can avoid the pitfalls of spatial overfitting. The result is an evaluation that inspires confidence: if a model performs well under these stringent conditions, one can justifiably claim it will perform in real-world scenarios. Adopting these robust methods will simulate realistic generalization performance while maintaining scientific defensibility, aligning your work with the cutting edge of geospatial AI validation [4] [9].

Sources:

- Roberts et al. (2017). *Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure.* Ecography, 40(8), 913–929 [10] [13].
- Stock et al. (2025). *Choosing blocks for spatial cross-validation: lessons from a marine remote sensing case study.* Frontiers in Remote Sensing, 2:101097 [16] [30].
- Fourcade et al. (2018). *Paintings predict the distribution of species…* Ecography, 41(6), 803–813 (on overestimation with random CV) [9].
- Radosavljevic & Anderson (2014). *Making better Maxent models…* Ecography, 37(10), 951–963 (introduced masked geographically structured CV) [14].
- Valavi et al. (2019). *blockCV: an R package for spatially separated cross-validation of species distribution models.* Methods Ecol Evol, 10(2), 225–232 [26].
- Pohjankukka et al. (2017). *Spatial random forest and sperrorest:…* IEEE Transactions on Geoscience and Remote Sensing, 55(9), 5387–5399 [18].
- Araújo et al. (2005). *Validation of species–climate impact models under climate change.* Global Change Biol, 11(9), 1504–1513 [7].
- Beery et al. (2018). *Recognition in terra incognita.* ECCV – illustrating location-specific model failures [25].

---

[1] [4] [5] [6] [13] [15] [16] [17] [18] [20] [22] [25] [26] [27] [28] [29] [30] Frontiers | Choosing blocks for spatial cross-validation: lessons from a marine remote sensing case study
https://www.frontiersin.org/journals/remote-sensing/articles/10.3389/frsen.2025.1531097/full

[2] [9] [14] Paintings predict the distribution of species, or the challenge of selecting environmental predictors and evaluation statistics | Request PDF
https://www.researchgate.net/publication/320283321_Paintings_predict_the_distribution_of_species_or_the_challenge_of_selecting_environmental_predictors_and_evaluation_statistics

[3] [23] [24] Spatial cross validation_V1
https://www.acsu.buffalo.edu/~yhu42/papers/2023_GeoAIHandbook_SpatialCV.pdf

[7] [8] [10] [11] [12] Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure
https://www.wsl.ch/lud/biodiversity_events/papers/Roberts_et_al-2017-Ecography.pdf

[19] (blockCV) Spatial buffering resampling — mlr_resamplings_spcv_buffer • mlr3spatiotempcv
https://mlr3spatiotempcv.mlr-org.com/reference/mlr_resamplings_spcv_buffer.html

21 Global mapping of potential natural vegetation - PubMed Central
https://pmc.ncbi.nlm.nih.gov/articles/PMC6109375/