https://doi.org/10.5194/egusphere-2023-1308
Preprint. Discussion started: 5 July 2023

kNNDM: k-fold Nearest Neighbour Distance Matching Cross-Validation for map accuracy estimation

Jan Linnenbrink[1,*], Carles Milà[2,3*], Marvin Ludwig[1], and Hanna Meyer[1]

[1]Institute of Landscape Ecology, University of Münster, Münster, Germany
[2]Barcelona Institute for Global Health (ISGlobal), Barcelona, Spain
[3]Universitat Pompeu Fabra (UPF), Barcelona, Spain
[*]These authors contributed equally to this work.

Correspondence: Jan Linnenbrink (jan.linnenbrink@uni-muenster.de)

Abstract. Random and spatial Cross-Validation (CV) methods are commonly used to evaluate machine learning-based spatial prediction models, and the obtained performance values are often interpreted as map accuracy estimates. However, the appropriateness of such approaches is currently the subject of controversy. For the common case where no probability sample for validation purposes is available, in Milà et al. (2022) we proposed the Nearest Neighbour Distance Matching (NNDM) Leave-One-Out (LOO) CV method. This method produces a distribution of geographical Nearest Neighbour Distances (NND) between test and train locations during CV that matches the distribution of NND between prediction and training locations. Hence, it creates predictive conditions during CV that are comparable to what is required when predicting a defined area. Although NNDM LOO CV produced largely reliable map accuracy estimates in our analysis, as a LOO-based method, it cannot be applied to large datasets found in many studies.

Here, we propose a novel k-fold CV strategy for map accuracy estimation inspired by the concepts of NNDM LOO CV: the k-fold NNDM (kNNDM) CV. The kNNDM algorithm tries to find a k-fold configuration such that the Empirical Cumulative Distribution Function (ECDF) of NND between test and train locations during CV is matched to the ECDF of NND between prediction and training locations.

We tested kNNDM CV in a simulation study with different sampling distributions and compared it to other CV methods including NNDM LOO CV. We found that kNNDM CV performed similarly to NNDM LOO CV and produced reasonably reliable map accuracy estimates across sampling patterns with strong reductions in computation time for large sample sizes. Furthermore, we found a positive linear association between the quality of the match of the two ECDFs in kNNDM and the reliability of the map accuracy estimates.

kNNDM provided the advantages of our original NNDM LOO CV strategy while bypassing its sample size limitations.

1 Introduction

Spatial predictive modeling using machine learning methods is commonly used in ecology and environmental sciences to predict variables sampled at a limited set of locations to new, unobserved locations (see e.g. van den Hoogen et al., 2019; Sabatini et al., 2022; Moreno-Martínez et al., 2018; Hengl et al., 2017, for global studies). A key step in the spatial prediction workflow is the map accuracy assessment, i.e. the process whereby the quality of a prediction of a spatially-indexed variable in a finite and defined geographical area (e.g. a set of pixels forming a continuous surface) is estimated (Stehman et al., 2021; Wadoux et al., 2021). Although map accuracy assessment should ideally be done via design-based inference based on probability sampling (Wadoux et al., 2021), this is frequently not possible due to limited access to certain areas or expensive sampling methods (Martin et al., 2012). Instead, Cross-Validation (CV) methods are commonly used to estimate the map accuracy. Previous studies, however, showed significant differences in map accuracy estimates depending on the type of CV being used, which initiated discussions on the appropriateness of these strategies (Wadoux et al., 2021; Meyer and Pebesma, 2022; Milà et al., 2022; Roberts et al., 2017; Ploton et al., 2020). Since CV is also typically used during model selection (i.e during hyperparameter tuning (Schratz et al., 2019) and feature selection (Meyer et al., 2019)), reliable estimates of map accuracy are crucial to develop suitable prediction models.

Traditional CV methods that ignore the spatial structure of the data such as Leave-One-Out (LOO) or random k-fold CV (Hastie et al., 2009) have been found to provide reliable estimates of map accuracy when samples are randomly distributed within the entire prediction area, but not when they are clustered (Milà et al., 2022; Wadoux et al., 2021) or only covering parts of the prediction area (Meyer and Pebesma, 2021). As an alternative, spatial CV methods such as block CV (Wenger and Olden, 2012; Valavi et al., 2019; Roberts et al., 2017) or buffered-LOO CV (Telford and Birks, 2009; Le Rest et al., 2014; Brenning, 2022) are often used. Spatial CV methods are designed for geographical model transferability assessment, i.e. to test the ability of the model to make predictions for new geographic entities far away from the sampling areas, by designing a CV where independence between train and test data is sought (Roberts et al., 2017). Such strategies, however, have been found to overestimate the map error when reference data are regularly or randomly distributed within the prediction area. In some cases, this has been even reported for clustered samples (Wadoux et al., 2021) when used for spatial interpolation (Milà et al., 2022). Recent proposals of methods for map accuracy estimation include sampling-intensity weighted CV, as well as model-based geostatistical approaches (de Bruin et al., 2022). However, the results of de Bruin et al. (2022) showed that these two methods are not universal solutions and, depending on the sampling design, showed considerable over- or underestimation of the true map accuracy.

In our past work, we argued that the design of a CV method to provide a reliable estimate for map accuracy should be prediction-oriented, i.e. predictive conditions created during CV should resemble the conditions found when the trained model is applied to the prediction area (Milà et al., 2022; Meyer and Pebesma, 2022; Ludwig et al., 2023). Following this idea, in Milà et al. (2022) we considered predictive conditions in terms of geographical distances, and proposed the Nearest Neighbour Distance Matching (NNDM) LOO CV method, a prediction-oriented CV method based on spatial point patterns concepts (Milà et al., 2022). Briefly, NNDM aims to match the Empirical Cumulative Distribution Function (ECDF) of Nearest Neighbour Distances (NND) between test and train locations in the CV process, to the ECDF of NND between prediction and training locations found during prediction.

In Milà et al. (2022) we showed that when simple LOO CV is used for randomly distributed reference data within the prediction area, the distribution of NND between test and training locations during CV is similar to the distribution of NND between prediction and training locations (see Meyer and Pebesma (2022) for similar results for random k-fold CV). In the

https://doi.org/10.5194/egusphere-2023-1308
Preprint. Discussion started: 5 July 2023

case of clustered sampling designs, NND during CV were generally much shorter than prediction NND which led to significant error underestimation when validated with random CV (see also Ludwig et al., 2023). For regular samples, NND during LOO CV were found to be slightly longer than during prediction leading to slight error overestimation (Milà et al., 2022). With the newly developed NNDM LOO CV, we could produce comparable NND distributions regardless of the sampling design and provide more reliable estimates of map accuracy.

Even though our proposed method showed promising results across different sampling designs, prediction areas, and landscape autocorrelation ranges (Milà et al., 2022), the fact that it is a LOO-based CV method hinders its broader application given its high computational cost in medium and large datasets. To overcome this limitation, our aim is to extend the idea of NNDM LOO CV to a k-fold NNDM CV, hereafter kNNDM, that can be applied to larger datasets commonly found in ecology and the environmental sciences. The paper is organised as follows: after presenting our algorithm in section 2, in section 3 we reproduce the simulation study included in Milà et al. (2022) to assess the performance of kNNDM and explore how it compares to the original NNDM LOO CV as well as to other CV methods. Finally, section 4 discusses the strengths and limitations of our method and suggests future lines of work.

2 Algorithm description

The algorithmic objective of kNNDM is the same as in the original NNDM LOO approach: to match the ECDF of the NND between test and training locations during CV, to the ECDF of NND between prediction and training locations found during prediction. While in the original NNDM LOO approach we focused on matching NND ECDF up to the range of the residual spatial autocorrelation of the variable being modelled (Milà et al., 2022), here we match all the ECDF to avoid the complexity and possible pitfalls of semivariogram estimation (e.g. trend, anisotropy, sampling size and spacing, preferential sampling; see Webster and Oliver (2007) for an overview). We define distances as Euclidean for projected coordinates and spherical for geographical coordinates.

In kNNDM, the definitions of the different NND ECDF remain the same as in the original NNDM LOO algorithm, where j is the index for training points and i is the index for prediction points (see detailed definitions with equations in Milà et al. (2022)):

- $\hat{G}_j(r)$ is the NND ECDF between test and train locations during LOO CV and expresses the proportion of training points that have another training point at a distance equal or lower than r.
- $\hat{G}_{ij}(r)$ is the NND ECDF between prediction and training locations and expresses the proportion of prediction points that have a sampling point at a distance equal or lower than r.
- $\hat{G}^*_j(r,\mathbf{L})$ is the NND ECDF between test and train locations during a CV defined in $\mathbf{L}$ and expresses the proportion of test points that have a training point at a distance equal or lower than r during a given CV strategy. Note that $\mathbf{L} = \{\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_{n_j}\}$ is a list of sets $\mathbf{l}_j$ containing the indices of the samples to fit the model to when holding out observation j during CV. Note that since in kNNDM we leave out data points in folds rather than one by one, $\mathbf{l}_j$ will be the exactly the same for all samples belonging to the same fold.

Another important component of our approach is how to measure the quality of the match between $\hat{G}^*_j(r,\mathbf{L})$ and $\hat{G}_{ij}(r)$ given a fold configuration. We do that by using the Wasserstein statistic (Dowd, 2020; Vaserstein, 1969), which compares the distribution of two samples by calculating the integral of the absolute value differences between the two ECDFs. In our case, 95 we define W as the integral over the geographical distances r of the absolute value differences between $\hat{G}^*_j(r,\mathbf{L})$ and $\hat{G}_{ij}(r)$:

$$W = \int |\hat{G}^*_j(r,\mathbf{L}) - \hat{G}_{ij}(r)|\mathrm{d}r$$

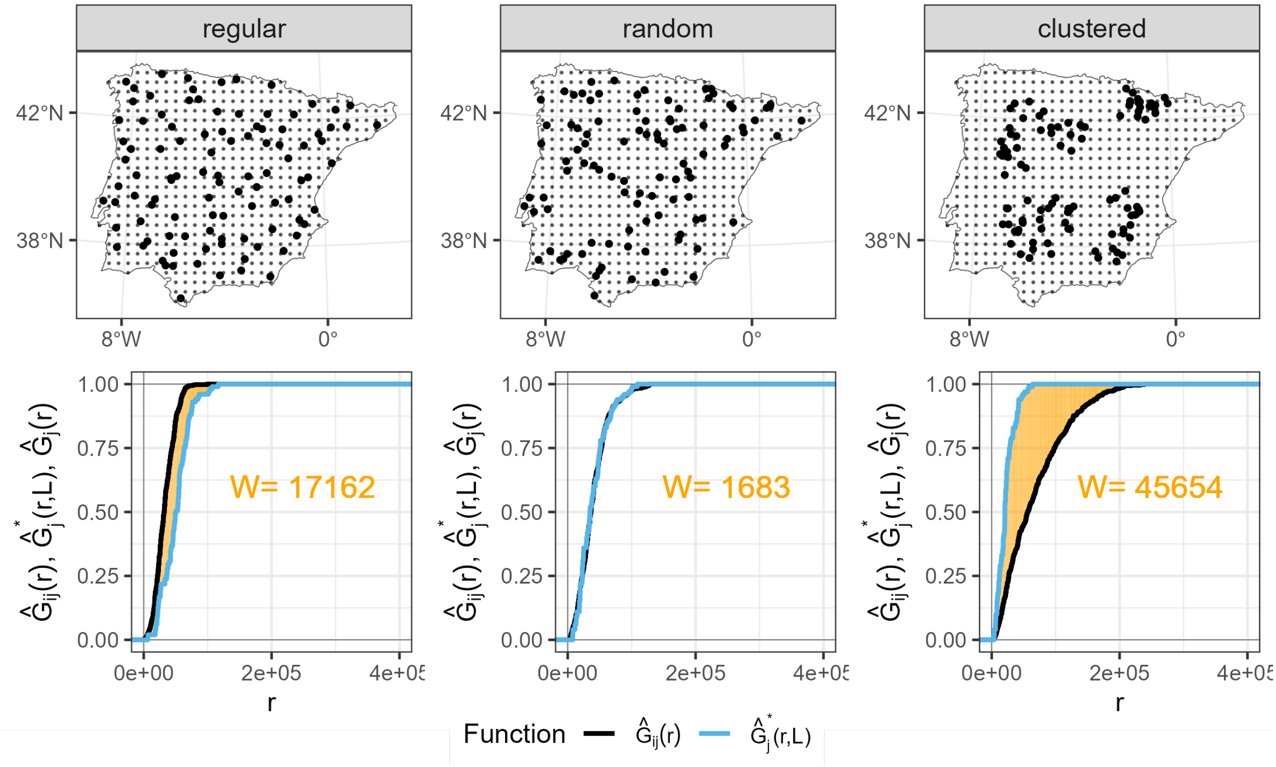

Figure 1. Top row: prediction points (regular grid) and simulated training points with different spatial distributions (bold). Bottom row: NND ECDF between training locations found during LOO CV ($\hat{G}_j(r)$, blue) and NND ECDF between prediction and training locations ($\hat{G}_{ij}(r)$, black) corresponding to each of the sampling distributions in the top row. The shaded yellow area corresponds to the W statistic, whose value is printed in the plots.

Small values of W indicate that the two ECDFs are similar, while W will be larger if they differ. To illustrate this point, we show the calculation of the W statistic between $\hat{G}_j(r)$ (ECDF of NND during LOO CV) and $\hat{G}_{ij}(r)$ (ECDF of NND during prediction) for different sampling patterns (Figure 1). As shown in Milà et al. (2022), when samples are randomly-distributed 100 within the prediction area, $\hat{G}_j(r)$ resembles $\hat{G}_{ij}(r)$ (since $\hat{G}_j(r) \approx \hat{F}_j(r)$ under complete spatial randomness, see Baddeley et al. (2015)) and therefore the value of W is small. However, in presence of clustered samples, LOO CV NND are shorter than prediction NND, and thus $\hat{G}_j(r) > \hat{G}_{ij}(r)$, resulting in a large W value. The opposite occurs when training samples follow a regular sampling pattern, also leading to a larger W statistic compared to random sampling.

The objective of the kNNDM algorithm (Figure 2) is to find a k-fold configuration such that W measuring the match between $\hat{G}^*_j(r,\mathbf{L})$ and $\hat{G}_{ij}(r)$ is minimised. The first step is to test whether the training points are clustered with respect to the prediction points; to do that we compute $\hat{G}_j(r)$ and $\hat{G}_{ij}(r)$, and then test the null hypothesis $G_j(r) \leq G_{ij}(r)$ vs. the alternative $G_j(r) > G_{ij}(r)$ with a one-sided Kolmogorov–Smirnov (KS) two-sample test (Conover, 1999). If the null hypothesis is not rejected, the algorithm returns a random k-fold CV since it is the appropriate option for random and regular samples (Meyer and Pebesma, 2022; Wadoux et al., 2021; de Bruin et al., 2022). If, however, the null hypothesis is rejected (p-value < 0.05), we proceed to cluster the training points based on their coordinates into a range of $q_i \in Q$ clusters, where Q is defined as an integer sequence of length 100 ranging between k and N (the total number of training points) equally-spaced in the logarithmic scale and back-transformed, to try configurations with a small number of clusters more intensively. Currently, hierarchical and k-means clustering methods are implemented.

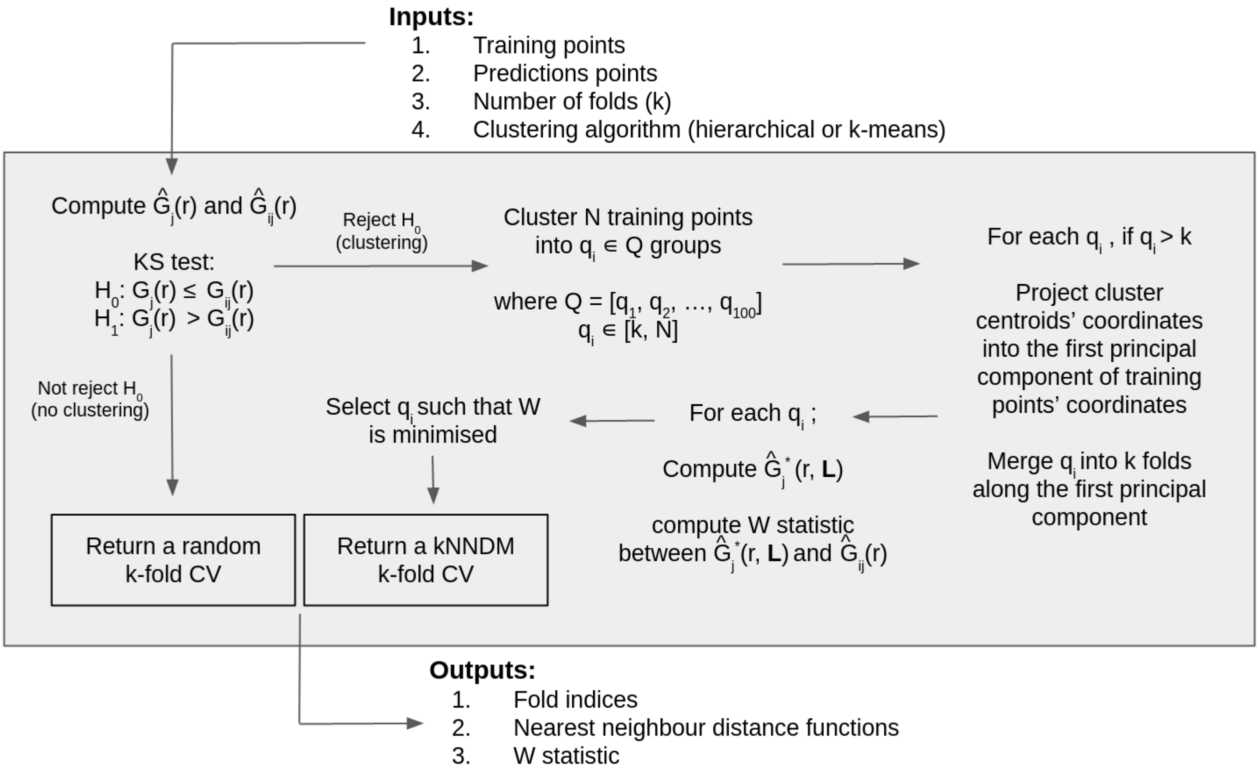
Inputs:
1. Training points
2. Predictions points
3. Number of folds (k)
4. Clustering algorithm (hierarchical or k-means)
Compute Ĝj(r) and Ĝij(r)
KS test:
H0: Gj(r) ≤ Gij(r)
H1: Gj(r) > Gij(r)
Reject H0 (clustering)
Cluster N training points into qi ∊ Q groups
where Q = [q1, q2, …, q100]
qi ∊ [k, N]
For each qi , if qi > k
Project cluster centroids' coordinates into the first principal component of training points' coordinates
Merge qi into k folds along the first principal component
Not reject H0 (no clustering)
Select qi such that W is minimised
For each qi ;
Compute Ĝj*(r, L)
compute W statistic between Ĝj*(r, L) and Ĝij(r)
Return a random k-fold CV
Return a kNNDM k-fold CV
Outputs:
1. Fold indices
2. Nearest neighbour distance functions
3. W statistic

Figure 2. Workflow of the kNNDM algorithm.

Next, for every configuration where $q > k$, we merge the resulting q clusters into the final k folds along the first principal component of the training points' coordinates to prevent contiguous clusters in space to be in the same fold. Briefly, we compute

https://doi.org/10.5194/egusphere-2023-1308
Preprint. Discussion started: 5 July 2023

the first principal component of the training points' coordinates to capture the direction with the most spatial variability, project the q cluster centroids into that first component and order them according to it, and finally merge q into k folds by giving different fold levels to contiguous clusters in that dimension. Large clusters with a proportion greater than $1/k$ of the training data are not merged to keep fold balance. Once this procedure is completed, we compute $\hat{G}_j^*(r,\mathbf{L})$ and W for each fold configuration candidate and select the one with the smallest W, i.e. the one that provides the best match between $\hat{G}_j^*(r,\mathbf{L})$ and $G_{ij}(r)$.

As an illustration of how kNNDM works in cases where samples are clustered within the prediction area, Figure 3 shows a 2-fold kNNDM CV configuration for different number of clusters q, their respective NND ECDF functions, and the W statistic between $\hat{G}_j^*(r,\mathbf{L})$ and $G_{ij}(r)$ assessing the match. A low number of clusters leads to a strong partition of the geographical space and long NND between test and train points during CV, which are actually longer than NND encountered when predicting from the reference data. As the number of clusters increases, $\hat{G}_j^*(r,\mathbf{L})$ gets closer to $\hat{G}_j(r)$, i.e. the NND ECDF encountered during LOO CV. In this example, the kNNDM algorithm would select the configuration with $q = 21$ since it minimises the W statistic, i.e. provides the best match.

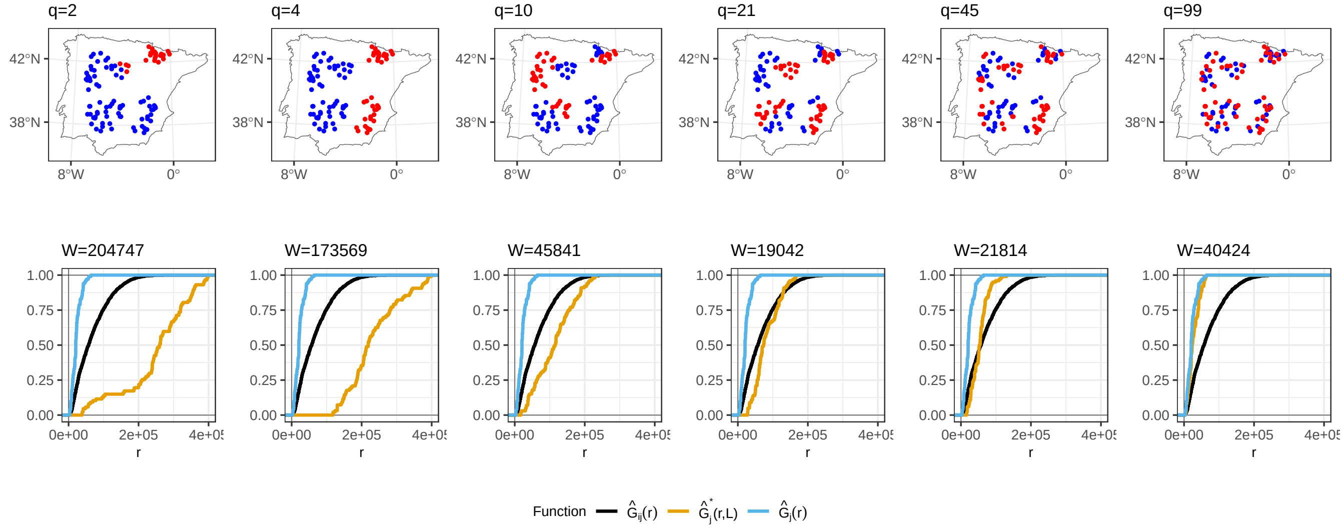

Figure 3. Top row: kNNDM with k=2 (red and blue points) for several number of clusters q. Prediction points consist of a regular grid spanning the whole polygon (not shown). Bottom row: NND ECDF between training locations during LOO CV ($\hat{G}_j(r)$, blue), between test and train locations during kNNDM CV ($\hat{G}_j^*(r,\mathbf{L})$, yellow), and between prediction and training locations ($\hat{G}_{ij}(r)$, black) corresponding to each CV configuration in the top row.

As practical considerations, in our implementation we have provided the possibility, as an alternative to the prediction points input, of supplying a polygon defining the prediction area, from which prediction points are sampled internally. Moreover, we have included a balancing parameter for the maximum single fold size allowed that discards non-compliant fold candidates. Regarding computational performance, our algorithm benefits from using projected coordinates since fast nearest neighbourhood searches in the Euclidean space can be done using the FNN package (Beygelzimer et al., 2022). Finally, when using kNNDM we recommend computing accuracy statistics such as coefficient of determination (R^2), the Root Mean Square Error (RMSE), or the Mean Absolute Error (MAE) in the stacked out-of-sample predictions rather than performing an average of the statistics computed in each of the folds, since the resulting folds can be unbalanced and $\hat{G}^*_j(r, \mathbf{L})$ is constructed using all CV folds simultaneously.

3 Simulation studies

3.1 Algorithm performance for map accuracy estimation

To investigate the performance of kNNDM CV and how it compares to the original NNDM LOO CV, we used the same simulation as in our previous work (see Milà et al. (2022) for a complete description). Briefly, we generated a virtual response surface, i.e. a spatially-indexed continuous variable between 0 and 1, using a selection of WorldClim bioclimatic variables for the Iberian peninsula (Figure 4A) and the virtualSpecies R package (Leroy et al., 2015). Next, we simulated training locations with five different distributions and a sample size of 100 (Figure 4B). We performed 100 iterations of the sampling simulation and, in each of them, we extracted the predictor (bioclimatic variables) and response data at the sampling points' locations and fitted Random Forest (RF) regression models, resulting in a total of 500 fitted models.

Each fitted RF model was used to predict the response in the whole prediction area, from which the true map accuracy was calculated (RMSE, MAE and R^2). Additionally, we used the following CV methods: 1) random 10-fold CV, 2) spatial 10-fold CV via k-means clustering (Brenning, 2012), 3) the original NNDM LOO CV and 4) 10-fold kNNDM CV. In contrast to the original simulation in Milà et al. (2022), here we matched all distances in the prediction area during NNDM rather than up to a certain range estimated from the data. In order to interpret results, we subtracted the true map accuracy metrics from each of the CV estimates to assess their performance (Figure 5).

kNNDM CV yielded reliable error estimates across all sampling patterns we considered, which were similar and in some cases even more accurate than those estimated via NNDM LOO CV (Figure 5). Variability of the differences was larger in kNNDM than in NNDM LOO CV for strongly clustered samples. Random 10-fold CV produced reliable estimates under random sampling patterns but failed for clustered data. The spatial 10-fold CV overestimated the mapping error except for the RMSE in the strong clustering scenario.

3.2 Relationship between the quality of the match and the map accuracy

In order to investigate the relationship between the quality of the match in kNNDM and the quality of the map accuracy metrics, we performed a second simulation using the same response and predictors and 100 iterations of our first simulation (section 3.1). However, in this second simulation, we 1) added two more extreme clustered sample configurations to extend the range of possible W values, 2) only used kNNDM CV, 3) did not check for clustering as a first step in kNNDM, i.e. we applied the clustering approach to all samples regardless of their distribution, and 4) we kept all candidate fold configurations $q_i \in Q$

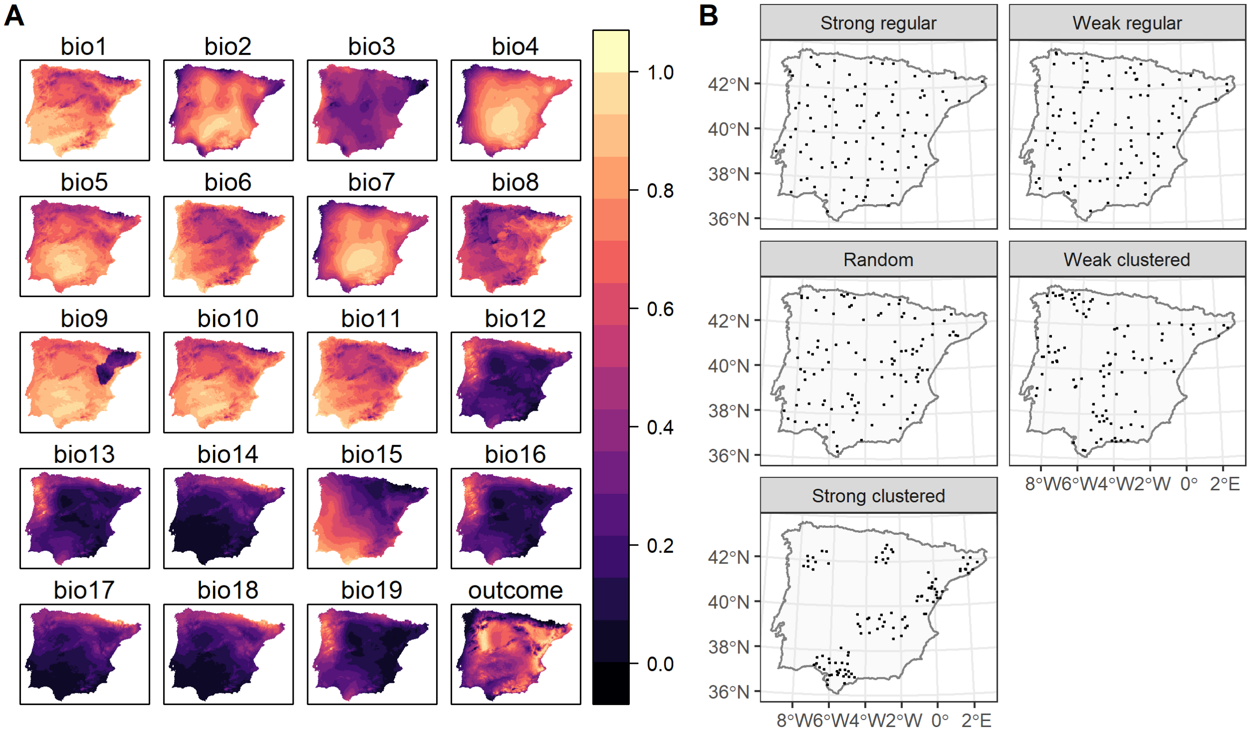

Figure 4. Simulation workflow: A) bioclimatic covariates and response (all linearly stretched to [0,1] for visualization purposes); and B) example of one iteration of the sample simulation. Figure reproduced as in Milà et al. (2022).

considered within kNNDM and their respective values of the W statistic, rather than just the one yielding the lowest W. We used all of these candidate CV splits to calculate CV map accuracy statistics, and computed the absolute value difference with respect to the true value of the map accuracy statistic. We then plotted these absolute value differences against the corresponding W statistic and fitted a linear regression to summarise the trend (Figure 6).

The relationship between the absolute value differences between CV and true map accuracy statistics (Figure 6) and W showed that, for all three statistics considered, a poor match between $\hat{G}_j^*(r,\mathbf{L})$ and $\hat{G}_{ij}(r)$ indicated by a greater W statistic led to a poor estimation of the true map accuracy, while the true map accuracy could be better estimated for well-matching distances. This positive association was linear for all three statistics with at least 60% explained variance.

3.3 Run-time analysis

Since our goal was to propose a computationally feasible alternative to NNDM for large datasets, we performed a run-time analysis to quantify the speed gains of kNNDM CV compared to NNDM LOO CV. We separately quantified the time spent on 1) finding the optimal CV split (i.e., running the kNNDM algorithm), 2) repetitively fitting the model according to the CV configuration, and 3) the total run-time, i.e. the sum of 1 and 2. We did that using the same simulation framework as in section

https://doi.org/10.5194/egusphere-2023-1308
Preprint. Discussion started: 5 July 2023

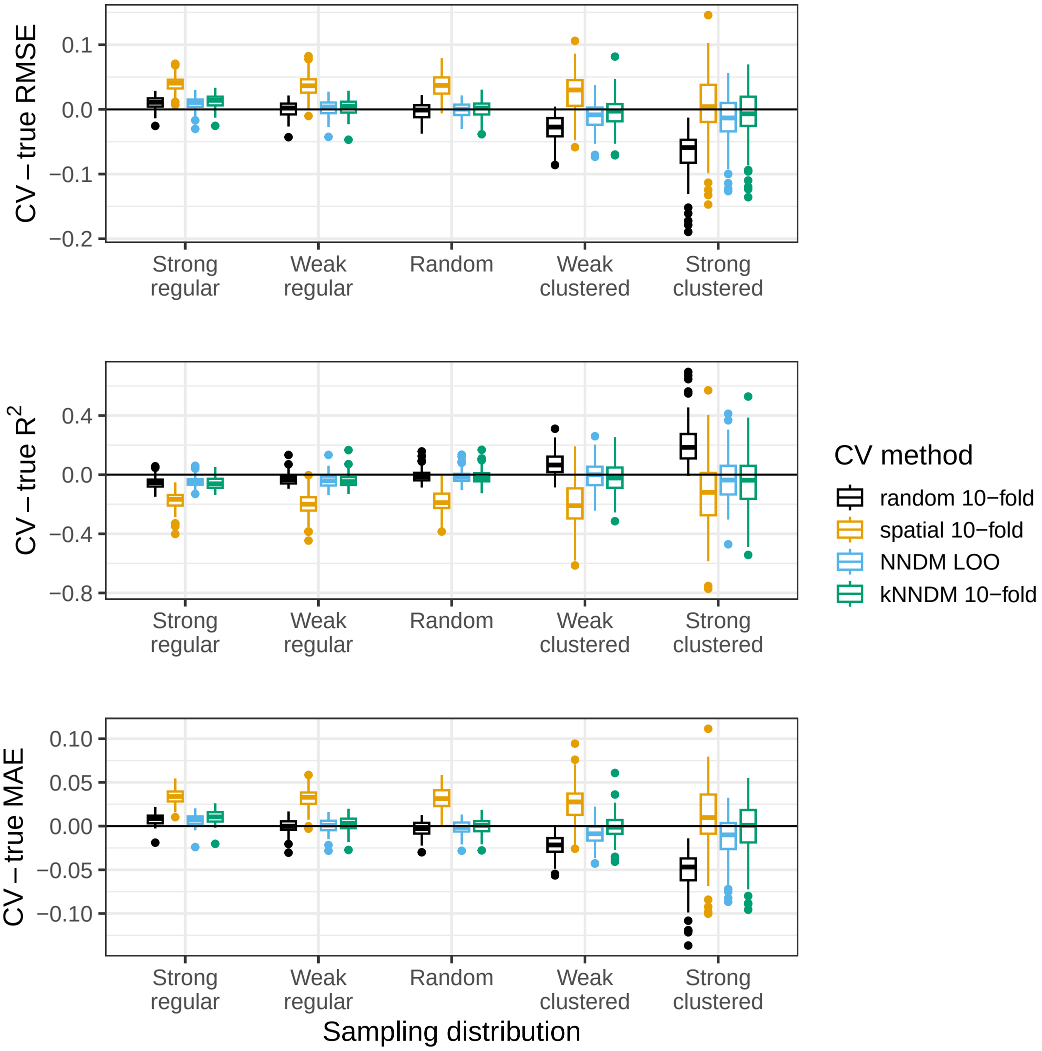

Figure 5. Differences between cross-validated and true RMSE, R^2, and MAE by sampling distribution and CV method for the simulated virtual response variable.

3.1 but with 50 different sample sizes, ranging from 100 to 4000 training points. We only used the strongly clustered and the random sampling designs for computational reasons. We used a maximum of 4000 training points, since the computational time exceeded one week per run for NNDM LOO CV. The analysis was carried out using a high-performance computation cluster utilizing up to 1.5 GB of RAM for each run using the Intel® Xeon® Gold 6140 CPU. 180

The run-time analysis showed large speed gains of kNNDM CV compared to NNDM LOO CV under all tested sample designs (Figure 7). For the random sampling design, the kNNDM algorithm was especially fast due to the prior test for clustering in the training data. This test returns a simple random k-fold CV when no clustering is detected, which is much

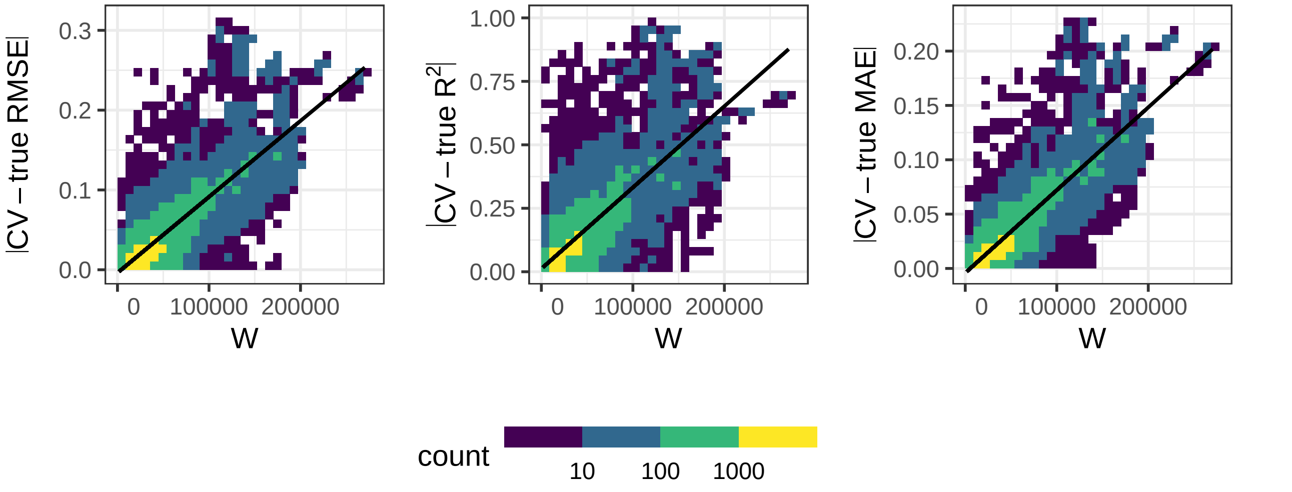

Figure 6. Relationship between the absolute value difference between the CV and true map accuracy statistics (y-axis) and W statistic (x-axis) from the second simulation study for the RMSE (left), R^2 (centre), and MAE (right) statistics. Colour represents the data point density. The black line shows the linear regression fit. The R^2 values of the regression models are 0.66, 0.6 and 0.73 for the RMSE, R^2, and MAE, respectively.

faster than running the entire kNNDM algorithm (see section 2). Only in four cases the test did not return a random CV, and in those five cases the computational times were longer (red outliers in Figure 7). The NNDM LOO algorithm was slower compared to the kNNDM algorithm when the training data were randomly distributed, and much slower when they were clustered (Figure 7). Furthermore, the computational time of NNDM LOO CV increased exponentially with increasing sample size. This pattern arises from both the architecture of the NNDM LOO CV algorithm (Figure 7 left column), and from the difference between of LOO CV and k-fold CV in terms of model fitting, since NNDM LOO CV requires training $k = N$ models while the number of models trained during k-fold CV is usually much smaller, in this case $k = 10$ (Figure 7 middle column).

4 Discussion

In this work, we propose a new prediction-oriented CV strategy for map accuracy estimation named kNNDM that takes into account the geographical prediction space of the model. kNNDM extends the ideas of NNDM LOO CV (Milà et al., 2022) to a k-fold CV strategy that can be used for medium and even large reference datasets to estimate map accuracy. In the simulation study, kNNDM performed similarly to NNDM LOO CV and produced reasonably reliable map accuracy estimates for all sampling patterns. Thus, kNNDM provided the advantages of our original NNDM LOO strategy while bypassing its sample size limitations.

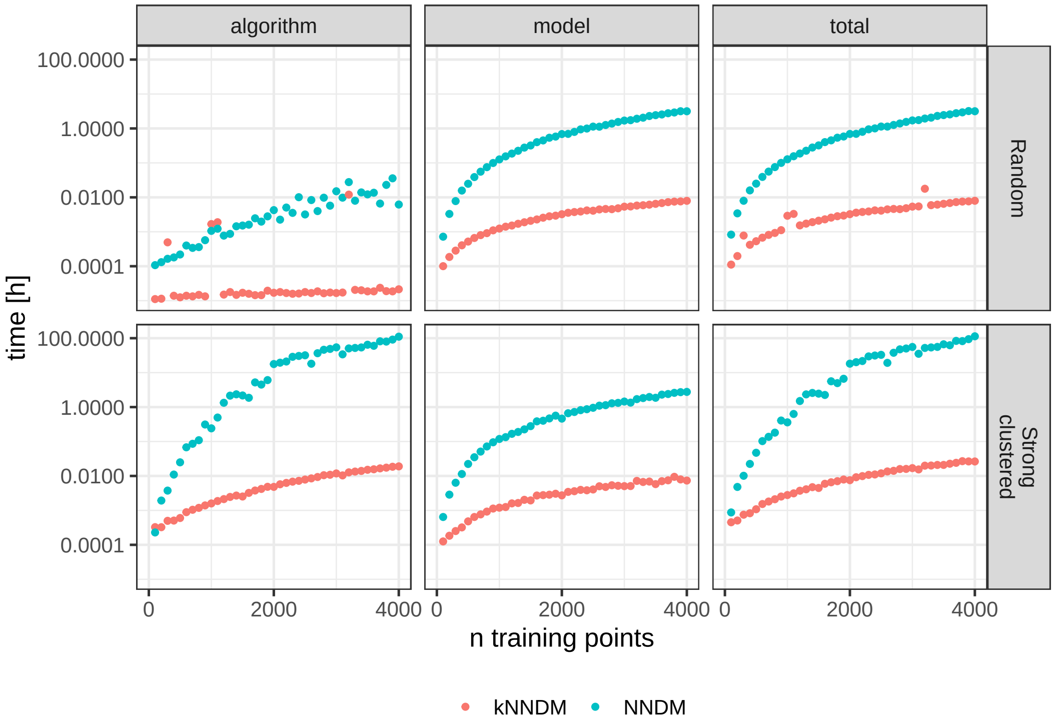

Figure 7. Differences in computational time between kNNDM CV (red) and NNDM LOO CV (blue) for two different sample designs (rows). Please note that the y-axis is log-scaled.

Similar to other studies (e.g. Wadoux et al., 2021; de Bruin et al., 2022), we observed that random k-fold CV returned reliable estimates of map accuracy with randomly-distributed samples within the prediction area, while being overly optimistic when samples were clustered. Also in agreement with other studies, we found that spatial CV methods that do not take into account the geographical prediction space tended to be overly pessimistic even with clustered samples within the prediction area (Wadoux et al., 2021; de Bruin et al., 2022; Milà et al., 2022), for example as a result of block sizes that do not match the prediction task. A unique finding of our study that deserves special attention is the linear and positive association we found between the W statistic measuring the quality of the match of the NND ECDFs during CV and prediction, and the quality of the estimation of the map accuracy statistics. In our simulations, the W statistic explained 60% or more of the absolute value differences between the CV and true map accuracy statistics, and thus further supports our suggestion to design CV strategies that try to match the predictive conditions of the models in terms of geographical NND.

Even though kNNDM overcomes the sample size limitations of NNDM LOO CV, there are still limitations of the approach. First, the flexibility of the matching in kNNDM is lower than in NNDM LOO CV, since every observation must be assigned to a fold. Moreover, it is also possible that the range of NND observed during prediction is different than the range of NND between training points, which might make the match impossible for some distances. This is especially the case when the prediction area is different from the training area (i.e. complete model transfer). Because of these reasons, the match between CV and prediction NND ECDF in kNNDM may not always be possible and an inspection of the NND ECDF like in Figure 1 should always be conducted. Similarly, if training data are very clustered within the prediction area, kNNDM may still fail to offer a CV configuration that matches the predictive conditions. In that case, we recommend users to allow for a greater maximum fold size, or ask for a lower fold number k, to account for potentially larger clusters. Secondly, both kNNDM and NNDM LOO CV algorithms are solely based on the geographical space; therefore, if the feature distribution between the training and prediction locations is very different, a feature-based CV strategy might be more appropriate (Roberts et al., 2017) or predictions should be limited to the area of applicability of the model (Meyer and Pebesma, 2021). Thirdly, NNDM-based CV methods do not address the small error overestimation for regular samples we found in our simulations here or in Milà et al. (2022), so map accuracy estimates will tend to be slightly conservative in such cases. Fourthly, NNDM-based methods are purely based on geographical distances and ignore the location of the training points or the direction of the distances, which can be problematic if non-stationarity or anisotropy of the errors are present (Brenning, 2022).

Possible future points for investigation regarding kNNDM include a simulation study comparing newly proposed CV-based map accuracy estimation methods (de Bruin et al., 2022) as well as feature-based CV methods (Roberts et al., 2017) in a larger variety of scenarios, also including classification problems, 2) exploring the extension of kNNDM to feature space, and 3) investigating how kNNDM CV can affect feature selection (Meyer et al., 2019), hyperparameter tuning (Schratz et al., 2019), and model applicability (Meyer and Pebesma, 2021). A further possible extension to the kNNDM algorithm is the exclusion of training points during CV, which might help to achieve a better match in strongly clustered designs without the need to increase fold sizes. Furthermore, it might be beneficial to develop and integrate a one-sided Wasserstein test instead of using the Kolmogorov-Smirnov two-sample test to test whether the training points are clustered as the first step of the algorithm, since the Wasserstein test has a greater power than Kolmogorov-Smirnov's (Dowd, 2020) and would be also be more consistent with the rest of the algorithm, which also uses the Wasserstein statistic.

Finally, we would like to emphasize again that both NNDM and kNNDM CV do not replace established strategies to estimate map accuracy via design based inference (as outlined in Wadoux et al., 2021), which should always be preferred. Here, we present a methodology to estimate map accuracy via CV. For the common case that no probability sample is available, we recommend CV via NNDM LOO or kNNDM to estimate map accuracy.

Code availability. All simulations were carried out in R v 4.2.1 (R Core Team, 2022). The most important packages used include `twosamples` (Dowd, 2022) for efficient calculation of the W statistic, `doParallel` (Corporation and Weston, 2022) for parallelization, `tidyverse` (Wickham et al., 2019) for data manipulation and `ggplot2` (Wickham, 2016) for data visualization. We used `sf` (Pebesma, 2018) for vector data operations, `terra` (Hijmans, 2023) for raster data operations and `caret` (Kuhn, 2022) for model fitting. NNDM LOO as well as the newly suggested kNNDM algorithms are implemented in the `CAST` package v 0.7.2 (Meyer et al., 2023). The code to perform the analysis and generate the figures included in the article is available at https://doi.org/10.6084/m9.figshare.23514135.v1, where the packages and the versions used for the simulations are listed.

Author contributions. All authors conceived the ideas and designed the study. JL and CM developed the algorithm, carried out the analysis, and drafted the manuscript. ML and HM contributed to discussions and drafts. All authors gave final approval for publication.

Competing interests. The authors declare no conflict of interest.

Acknowledgements. The work was supported by the Federal Ministry of Economic Affairs and Climate action of Germany (project number 50EE2009). CM was supported by a PhD fellowship of the *Severo Ochoa Centre of Excellence* programme awarded to ISGlobal.

References

Baddeley, A., Rubak, E., and Turner, R.: Spatial point patterns: methodology and applications with R, CRC press, 2015.

Beygelzimer, A., Kakadet, S., Langford, J., Arya, S., Mount, D., and Li, S.: FNN: Fast Nearest Neighbor Search Algorithms and Applications, https://CRAN.R-project.org/package=FNN, r package version 1.1.3.1, 2022.

Brenning, A.: Spatial cross-validation and bootstrap for the assessment of prediction rules in remote sensing: The R package sperrorest, in: 2012 IEEE International Geoscience and Remote Sensing Symposium, pp. 5372–5375, https://doi.org/10.1109/IGARSS.2012.6352393, 2012.

Brenning, A.: Spatial machine-learning model diagnostics: a model-agnostic distance-based approach., International Journal of Geographical Information Science, https://doi.org/10.1080/13658816.2022.2131789, 2022.

Conover, W. J.: Practical nonparametric statistics, vol. 350, john wiley & sons, 1999.

Corporation, M. and Weston, S.: doParallel: Foreach Parallel Adaptor for the 'parallel' Package, https://CRAN.R-project.org/package=doParallel, r package version 1.0.17, 2022.

de Bruin, S., Brus, D. J., Heuvelink, G. B., van Ebbenhorst Tengbergen, T., and Wadoux, A. M.-C.: Dealing with clustered samples for assessing map accuracy by cross-validation, Ecological Informatics, 69, 101 665, https://doi.org/10.1016/j.ecoinf.2022.101665, 2022.

Dowd, C.: A New ECDF Two-Sample Test Statistic, https://doi.org/10.48550/ARXIV.2007.01360, 2020.

Dowd, C.: twosamples: Fast Permutation Based Two Sample Tests, https://CRAN.R-project.org/package=twosamples, r package version 2.0.0, 2022.

Hastie, T., Tibshirani, S., and Friedman, H.: The Elements of Statistical Learning. Data Mining, Inference, and Prediction., Springer Science Business Media, 2009.

Hengl, T., Mendes de Jesus, J., Heuvelink, G. B. M., Ruiperez Gonzalez, M., Kilibarda, M., Blagotic, A., Shangguan, W., Wright, M. N., Geng, X., Bauer-Marschallinger, B., Guevara, M. A., Vargas, R., MacMillan, R. A., Batjes, N. H., Leenaars, J. G. B., Ribeiro, E., Wheeler, I., Mantel, S., and Kempen, B.: SoilGrids250m: Global gridded soil information based on machine learning., PloS one, 12, e0169 748, place: United States, 2017.

Hijmans, R. J.: terra: Spatial Data Analysis, https://CRAN.R-project.org/package=terra, r package version 1.7-3, 2023.

Kuhn, M.: caret: Classification and Regression Training, https://CRAN.R-project.org/package=caret, r package version 6.0-93, 2022.

Le Rest, K., Pinaud, D., Monestiez, P., Chadoeuf, J., and Bretagnolle, V.: Spatial leave-one-out cross-validation for variable selection in the presence of spatial autocorrelation, Global Ecology and Biogeography, 23, 811–820, https://doi.org/https://doi.org/10.1111/geb.12161, 2014.

Leroy, B., Meynard, C. N., Bellard, C., and Courchamp, F.: virtualspecies, an R package to generate virtual species distributions, Ecography, https://doi.org/10.1111/ecog.01388, 2015.

Ludwig, M., Moreno-Martinez, A., Hölzel, N., Pebesma, E., and Meyer, H.: Assessing and improving the transferability of current global spatial prediction models, Global Ecology and Biogeography, n/a, https://doi.org/https://doi.org/10.1111/geb.13635, 2023.

Martin, L. J., Blossey, B., and Ellis, E.: Mapping where ecologists work: biases in the global distribution of terrestrial ecological observations, Frontiers in Ecology and the Environment, 10, 195–201, https://doi.org/https://doi.org/10.1890/110154, 2012.

Meyer, H. and Pebesma, E.: Predicting into unknown space? Estimating the area of applicability of spatial prediction models, Methods in Ecology and Evolution, https://doi.org/https://doi.org/10.1111/2041-210X.13650, 2021.

Meyer, H. and Pebesma, E.: Machine learning-based global maps of ecological variables and the challenge of assessing them, Nature Communications, 13, 2208, 2022.

Meyer, H., Reudenbach, C., Wöllauer, S., and Nauss, T.: Importance of spatial predictor variable selection in machine learning applications – Moving from data reproduction to spatial prediction, Ecological Modelling, 411, 108 815, https://doi.org/https://doi.org/10.1016/j.ecolmodel.2019.108815, 2019.

Meyer, H., Milà, C., Ludwig, M., and Linnenbrink, J.: CAST: 'caret' Applications for Spatial-Temporal Models, https://github.com/HannaMeyer/CAST, https://hannameyer.github.io/CAST/, 2023.

Milà, C., Mateu, J., Pebesma, E., and Meyer, H.: Nearest neighbour distance matching Leave-One-Out Cross-Validation for map validation, Methods in Ecology and Evolution, 13, 1304–1316, https://doi.org/https://doi.org/10.1111/2041-210X.13851, 2022.

Moreno-Martínez, , Camps-Valls, G., Kattge, J., Robinson, N., Reichstein, M., Bodegom, P. v., Kramer, K., Cornelissen, J. H. C., Reich, P., Bahn, M., Niinemets, , Peñuelas, J., Craine, J. M., Cerabolini, B. E. L., Minden, V., Laughlin, D. C., Sack, L., Allred, B., Baraloto, C., Byun, C., Soudzilovskaia, N. A., and Running, S. W.: A methodology to derive global maps of leaf traits using remote sensing and climate data, Remote Sensing of Environment, 218, 69–88, https://doi.org/https://doi.org/10.1016/j.rse.2018.09.006, 2018.

Pebesma, E.: Simple Features for R: Standardized Support for Spatial Vector Data, The R Journal, 10, 439–446, https://doi.org/10.32614/RJ-2018-009, 2018.

Ploton, P., Mortier, F., Réjou-Méchain, M., Barbier, N., Picard, N., Rossi, V., Dormann, C., Cornu, G., Viennois, G., Bayol, N., Lyapustin, A., Gourlet-Fleury, S., and Pélissier, R.: Spatial validation reveals poor predictive performance of large-scale ecological mapping models, 11, 4540, https://doi.org/10.1038/s41467-020-18321-y, 2020.

R Core Team: R: A Language and Environment for Statistical Computing, R Foundation for Statistical Computing, Vienna, Austria, https://www.R-project.org/, 2022.

Roberts, D. R., Bahn, V., Ciuti, S., Boyce, M. S., Elith, J., Guillera-Arroita, G., Hauenstein, S., Lahoz-Monfort, J. J., Schröder, B., Thuiller, W., Warton, D. I., Wintle, B. A., Hartig, F., and Dormann, C. F.: Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure, 40, 913–929, https://doi.org/10.1111/ecog.02881, 2017.

Sabatini, F. M., Jiménez-Alfaro, B., Jandt, U., Chytrý, M., Field, R., Kessler, M., Lenoir, J., Schrodt, F., Wiser, S. K., Arfin Khan, M. A. S., Attorre, F., Cayuela, L., De Sanctis, M., Dengler, J., Haider, S., Hatim, M. Z., Indreica, A., Jansen, F., Pauchard, A., Peet, R. K., Petřík, P., Pillar, V. D., Sandel, B., Schmidt, M., Tang, Z., van Bodegom, P., Vassilev, K., Violle, C., Alvarez-Davila, E., Davidar, P., Dolezal, J., Hérault, B., Galán-de Mera, A., Jiménez, J., Kambach, S., Kepfer-Rojas, S., Kreft, H., Lezama, F., Linares-Palomino, R., Monteagudo Mendoza, A., N'Dja, J. K., Phillips, O. L., Rivas-Torres, G., Sklenář, P., Speziale, K., Strohbach, B. J., Vásquez Martínez, R., Wang, H.-F., Wesche, K., and Bruelheide, H.: Global patterns of vascular plant alpha diversity, Nature Communications, 13, 4683, https://doi.org/10.1038/s41467-022-32063-z, 2022.

Schratz, P., Muenchow, J., Iturritxa, E., Richter, J., and Brenning, A.: Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data, Ecological Modelling, 406, 109–120, https://doi.org/https://doi.org/10.1016/j.ecolmodel.2019.06.002, 2019.

Stehman, S. V., Pengra, B. W., Horton, J. A., and Wellington, D. F.: Validation of the U.S. Geological Survey's Land Change Monitoring, Assessment and Projection (LCMAP) Collection 1.0 annual land cover products 1985–2017, Remote Sensing of Environment, 265, 112 646, https://doi.org/https://doi.org/10.1016/j.rse.2021.112646, 2021.

Telford, R. and Birks, H.: Evaluation of transfer functions in spatially structured environments, Quaternary Science Reviews, 28, 1309 – 1316, https://doi.org/https://doi.org/10.1016/j.quascirev.2008.12.020, 2009.

Valavi, R., Elith, J., Lahoz-Monfort, J. J., and Guillera-Arroita, G.: blockCV: An r package for generating spatially or environmentally separated folds for k-fold cross-validation of species distribution models, Methods in Ecology and Evolution, 10, 225–232, https://doi.org/10.1111/2041-210X.13107, 2019.

van den Hoogen, J., Geisen, S., Routh, D., Ferris, H., Traunspurger, W., Wardle, D. A., de Goede, R. G. M., Adams, B. J., Ahmad, W., Andriuzzi, W. S., Bardgett, R. D., Bonkowski, M., Campos-Herrera, R., Cares, J. E., Caruso, T., de Brito Caixeta, L., Chen, X., Costa, S. R., Creamer, R., Mauro da Cunha Castro, J., Dam, M., Djigal, D., Escuer, M., Griffiths, B. S., Gutiérrez, C., Hohberg, K., Kalinkina, D., Kardol, P., Kergunteuil, A., Korthals, G., Krashevska, V., Kudrin, A. A., Li, Q., Liang, W., Magilton, M., Marais, M., Martín, J. A. R., Matveeva, E., Mayad, E. H., Mulder, C., Mullin, P., Neilson, R., Nguyen, T. A. D., Nielsen, U. N., Okada, H., Rius, J. E. P., Pan, K., Peneva, V., Pellissier, L., Carlos Pereira da Silva, J., Pitteloud, C., Powers, T. O., Powers, K., Quist, C. W., Rasmann, S., Moreno, S. S., Scheu, S., Setälä, H., Sushchuk, A., Tiunov, A. V., Trap, J., van der Putten, W., Vestergård, M., Villenave, C., Waeyenberge, L., Wall, D. H., Wilschut, R., Wright, D. G., Yang, J.-i., and Crowther, T. W.: Soil nematode abundance and functional group composition at a global scale, 572, 194–198, https://doi.org/10.1038/s41586-019-1418-6, number: 7768 Publisher: Nature Publishing Group, 2019.

Vaserstein, L. N.: Markov processes over denumerable products of spaces, describing large systems of automata, Problemy Peredachi Informatsii, 5, 64–72, 1969.

Wadoux, A. M.-C., Heuvelink, G. B., de Bruin, S., and Brus, D. J.: Spatial cross-validation is not the right way to evaluate map accuracy, Ecological Modelling, 457, 109 692, https://doi.org/https://doi.org/10.1016/j.ecolmodel.2021.109692, 2021.

Webster, R. and Oliver, M. A.: Geostatistics for environmental scientists, John Wiley & Sons, 2007.

Wenger, S. J. and Olden, J. D.: Assessing transferability of ecological models: an underappreciated aspect of statistical validation, Methods in Ecology and Evolution, 3, 260–267, https://doi.org/10.1111/j.2041-210X.2011.00170.x, 2012.

Wickham, H.: ggplot2: Elegant Graphics for Data Analysis, Springer-Verlag New York, https://ggplot2.tidyverse.org, 2016.

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T. L., Miller, E., Bache, S. M., Müller, K., Ooms, J., Robinson, D., Seidel, D. P., Spinu, V., Takahashi, K., Vaughan, D., Wilke, C., Woo, K., and Yutani, H.: Welcome to the tidyverse, Journal of Open Source Software, 4, 1686, https://doi.org/10.21105/joss.01686, 2019.