

APPLICATION

tidysdm: Leveraging the flexibility of *tidymodels* for species distribution modelling in R

Michela Leonardi¹  | Margherita Colucci^{1,2}  | Andrea Vittorio Pozzi¹  |
Eleanor M. L. Scerri^{2,3,4}  | Andrea Manica¹ 

¹Evolutionary Ecology Group, Department of Zoology, University of Cambridge, Cambridge, UK

²Human Palaeosystems Research Group, Max Planck Institute of Geoanthropology Jena, Jena, Germany

³Department of Classics and Archaeology, University of Malta, Msida, Malta

⁴Institute of Prehistoric Archaeology, University of Cologne, Cologne, Germany

Correspondence

Andrea Manica

Email: am315@cam.ac.uk

Present address

Michela Leonardi, Natural History Museum, London, UK

Funding information

Natural Environment Research Council, Grant/Award Number: NE/S007164/1; Leverhulme Trust, Grant/Award Number: RPG-2020-317; Lise Meitner Pan-African Evolution Research Group

Handling Editor: Eric Ward

Abstract

1. In species distribution modelling (SDM), it is common practice to explore multiple machine learning (ML) algorithms and combine their results into ensembles. In R, many implementations of different ML algorithms are available but, as they were mostly developed independently, they often use inconsistent syntax and data structures. For this reason, repeating an analysis with multiple algorithms and combining their results can be challenging.
2. Specialised SDM packages solve this problem by providing a simpler, unified interface by wrapping the original functions to tackle each specific requirement. However, creating and maintaining such interfaces is time-consuming, and with this approach, the user cannot easily integrate other methods that may become available.
3. Here, we present *tidysdm*, an R package that solves this problem by taking advantage of the *tidymodels* universe. *tidymodels* provide standardised grammar, data structures and modelling interfaces, and a well-documented infrastructure to integrate new algorithms and metrics. The wide adoption of *tidymodels* means that most ML algorithms and metrics are already integrated, and the user can add additional ones. Moreover, because of the broad adoption of *tidymodels*, new statistical approaches tend to be implemented quickly, making them easily integrated into existing pipelines and analyses.
4. *tidysdm* takes advantage of the *tidymodels* universe to provide a flexible and fully customisable pipeline to fit SDM. It includes SDM-specific algorithms and metrics, and methods to facilitate the use of spatial data within *tidymodels*.
5. Additionally, *tidysdm* is the first software that natively allows SDM to be performed using data from different periods, expanding the availability of SDM for scholars working in palaeontology, archaeology, palaeobiology, palaeoecology and other disciplines focussing on the past.

KEYWORDS

biogeography, paleoecology, R package, species distribution modelling, tidyverse

Michela Leonardi and Margherita Colucci contributed equally to this work.

This is an open access article under the terms of the [Creative Commons Attribution](https://creativecommons.org/licenses/by/4.0/) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2024 The Author(s). *Methods in Ecology and Evolution* published by John Wiley & Sons Ltd on behalf of British Ecological Society.

1 | INTRODUCTION

Species distribution modelling (SDM, also known as ecological niche modelling [ENM]; habitat suitability models [HSM]; and other acronyms, Guisan et al., 2017) is a class of biological methods that use the occurrences of a given species and the associated environmental variables to predict if (and how much) a geographic area is suitable for that species (Elith & Leathwick, 2009). SDMs have been widely used, for example, to predict distributions of invasive species in new ranges (Elith, 2017), assess habitat suitability under future climatic scenarios (Franklin, 2023), reconstruct past ranges based on archaeological/palaeontological remains (e.g. Leonardi et al., 2018) or present-day surveys (e.g. Miller et al., 2021), and detect niche changes through time (e.g. Leonardi et al., 2022).

In computational biology, SDM has been among the first fields to use machine learning (ML), and it is standard practice to explore multiple algorithms and combine their results into ensembles (Araújo & New, 2007). R Core Team (2024) is the most adopted framework to fit SDMs. Most ML algorithms are already available in at least one package, with common approaches (e.g. random forests) being implemented in several of them. However, as many packages have been developed independently, they often differ in their data structures and syntax, posing a challenge to the user. As a practical example, when two algorithms require differently formatted inputs and produce incompatible outputs, comparing their results and merging them into ensembles is difficult. Specialised SDM packages have been developed to solve this problem (Sillero et al., 2023); each offers a different, unified interface to multiple algorithms by providing wrapper functions around each of them. Thanks to this, the user does not have to worry about the specific requirements of each package behind every used ML algorithm(s).

Such specialised packages greatly simplify the user experience, but the user can only work with what they already include. For example, when a new algorithm becomes available, there is no easy way for a user to integrate it into their analysis. Similarly, it is generally not possible to add a custom metric when tuning models or use a new strategy for hyperparameter tuning (e.g. exploring Bayesian tuning or racing methods instead of a standard grid search).

2 | TIDYMODELS

The challenges associated with the heterogeneous landscape of ML packages are not specific to SDMs but apply to any field using ML algorithms. Groups of packages, such as *tidymodels* (Kuhn & Wickham, 2024) and *mlr3* (Lang et al., 2019) in R, and *scikit-learn* (Pedregosa et al., 2011) in Python, have emerged as field-agnostic solutions that attempt to provide a standardised interface to multiple algorithms. *tidymodels*, in particular, is characterised by a highly modular approach.

How does it work? Instead of having a collection of custom wrappers, *tidymodels* provides, through several interdependent packages, functions that allow the user to define and register new elements

from each step of a ML pipeline (e.g. data pre-processing steps, algorithms, metrics). Once a new element has been registered, it is immediately compatible with all other functions of the *tidymodels* universe. For example, the package *parsnip* (Kuhn & Vaughan, 2024) includes functions, documentation and tutorials on defining algorithms, as well as pre-prepared definitions for a large number of commonly used ML algorithms. These can then be evaluated with any of the metrics found within the *yardstick* package (Kuhn et al., 2024). In turn, *yardstick* not only provides a large range of widely used metrics, but also allows to define new ones that can then be used with any algorithm set up with the *parsnip* model specification. Importantly, this modular infrastructure allows the 'hot-plugging' of additions without modifying existing packages. Users can simply implement new functions (e.g. to use a new algorithm or metric) and mix them with existing ones from the whole *tidymodels* universe. Furthermore, developers can simply provide packages with the implementation of a new algorithm or metric and have them immediately integrated into existing pipelines.

3 | A 'TIDYMODELS' APPROACH TO SDM

Here, we introduce *tidysdm*, a package that facilitates fitting correlative SDMs with *tidymodels*. In addition to the whole set of algorithms and metrics already available in *tidymodels*, *tidysdm* facilitates SDMs by providing:

- Additional methods to several *tidymodels* functions to handle spatial data;
- Implementations of SDM-relevant algorithms and metrics (e.g. MaxEnt, Phillips et al., 2004; Boyce Continuous Index, Hirzel et al., 2006) unavailable from existing *tidymodels* packages.

By building on the modular infrastructure of *tidymodels*, *tidysdm* does not need to create complete solutions from scratch, because it can take advantage of a large community of developers: objects created within *tidysdm* can be fed to functions from other packages. For example, sensitivity and specificity, two widely used metrics for SDM, are already provided by the *yardstick* package (respectively as *sens()* and *spec()*). There was no need to reimplement these functions in *tidysdm*, because any model based on the *tidymodels* standards can be directly fed to *yardstick::sens()* and *yardstick::spec()* to get an output that can be used in any *tidysdm* pipeline. This is a simple example but allows us to see how much the *tidymodels* infrastructure allows flexibility while at the same time simplifying the way pipelines can be built.

The wide adoption of *tidymodels* also provides a wide range of additional packages for various specialised applications. For example, it is important to be able to interpret ML results and compare the behaviour of different algorithms, but, in principle, this can be challenging. DALEX (Biecek, 2018) is one of the most widely adopted algorithm-agnostic packages for ML interpretability analysis. It can take objects fitted with *tidymodels*, thus providing cutting-edge tools

to interpret any algorithm that we might want to use in our SDM ensembles.

Finally, *tidysdm* contains functions to easily work with data from different periods (palaeontological, archaeological, archaeozoological, or, more generally, time-scattered data). This is a task that, with most SDM packages, is either impossible or requires extensive tweaking. Access and manipulation of (paleo)climate are facilitated by the integration with *pastclim* (Leonardi et al., 2023): from version 2.0, it includes present-day climate and future reconstructions, in addition to palaeoclimate. *pastclim* is not required, though: users can import environmental data through their preferred pipeline, as *tidysdm* handles rasters with the *terra* package (Hijmans, 2024).

4 | ANALYSIS WORKFLOW AND EXAMPLES

The examples presented in this manuscript are based on the version of the software available at the time of publication. The links in this text connect to a live version of the package website that may be modified in case of future releases. A copy of the current version of the package and all associated data and vignettes can be found in Leonardi et al. (2024).

4.1 | Applications with present-day data

We illustrate an example of a standard SDM pipeline modelling contemporary species in a dedicated vignette (https://evolecolgroup.github.io/tidysdm/articles/a0_tidysdm_overview.html). The document shows the full flexibility offered in analysing the Iberian lizard (*Lacerta schreiberi*) following (Rödder & Schulte, 2010). The website (<https://evolecolgroup.github.io/tidysdm>) also has a manual including all functions, and a vignette (https://evolecolgroup.github.io/tidysdm/articles/a2_tidymodels_additions.html) that illustrates a few additional features from *tidymodels* and companion packages that can enhance SDM pipelines. This final vignette only contains a few examples; we envisage many more applications of the many options available in *tidymodels* thanks to its widespread use in multiple fields.

The first step in any SDM workflow is data collection. Presence data can come from many sources, and there are packages dedicated to using some of these repositories (e.g. *rgbif* for downloading data from GBIF; Chamberlain et al., 2024). Once presences have been obtained and cleaned (e.g. with *CoordinateCleaner*, Zizka et al., 2019), *tidysdm* provides several functions to facilitate their preparation for the analysis. These include the thinning of presences (Boria et al., 2014), via the functions *thin_by_cell()* and *thin_by_dist()*, and the sampling of either the background via *sample_background()* or pseudo-absences with *sample_pseudoabs()* (Ferrier et al., 2002; Phillips et al., 2009; Renner et al., 2015). Each of these functions offers multiple strategies, for example sampling the background according to a bias surface or pseudo-absences with a minimal

distance from presences. They all work on *sf* objects to represent spatial points and *terra* *SpatRasters* to represent the rasters of pre-dictor variables.

Several algorithms struggle with correlated predictors, and the function *filter_collinear()* provides several approaches to select uncorrelated variables, including a stepwise approach to minimise pairwise correlation, and an option to minimise variance inflation (more information on these options is available on the manual page). We also provide a function *dist_pres_vs_bg()* to visualise and quantify the differences in distribution between the presences and the background/pseudo-absences.

Once presences and background/pseudo-absences have been obtained (Figures 1a and 2b), we take advantage of the standardised *tidymodels* workflow. *tidymodels* use *recipes* to wrap the data and pre-process them before analysis. A recipe can be thought of as a set of steps designed to clean and prepare a dataset, with a clear definition of the response variable (i.e. occurrences for an SDM) and the predictors (e.g. climatic variables). An advantage of this approach is that additional data (e.g. from multiple species) can be fed through the same recipe, an easy way to standardise data pipelines.

Once the data have been wrapped into a *recipe*, they can be split into training and testing folds. Using the *tidyverse* package *spatial-sample* (Mahoney et al., 2023), it is possible to use several sampling strategies (e.g. spatial block design) to split the data (Figures 1b and 2c). We can use a formal approach of creating a testing and training set, and then creating folds within the training set for cross-validation when tuning hyperparameters (the more conventional approach used in ML), or just work with cross-validation folds without a separated testing set (an approach often adopted in SDM analyses with limited amounts of data). *tidysdm* also allows importing of objects from BlockCV (Valavi et al., 2019), a powerful package that provides additional sampling strategies unavailable within *spatial-sample*.

The next step is to define algorithms and metrics used to run the models. Algorithms are specified via the package *parsnip*, which implements many of them (both ML, e.g. random forests and boosted trees, and more conventional statistical models, for example, Generalised Linear Models and Generalised Additive Model). But it is relatively straightforward to add additional algorithms; in *tidysdm*, we implement MaxEnt based on the package *maxnet* (Phillips et al., 2024). *parsnip* provides functions to define the model inputs, that is how predictors and the response are formatted. It also allows us to define hyperparameters, that is the parameters that control the learning process, and thus determine the values of the parameters fitted by a ML algorithm. Once we have used *parsnip* to set up the model specifications, the model can be fed with the data wrapped in *recipes* and made to interact with metrics and hyperparameter selection functions (see below). The model specification for MaxEnt can be used as a blueprint to add any other model, and there are detailed tutorials on how to define custom model specifications within *parsnip*. Importantly, such specifications do not have to be added as a package, they can be simply defined as functions that the user can then adopt within their *tidymodels* workflow. *tidysdm* also provides several model specifications for widely used algorithms (such as Random Forest or boosted

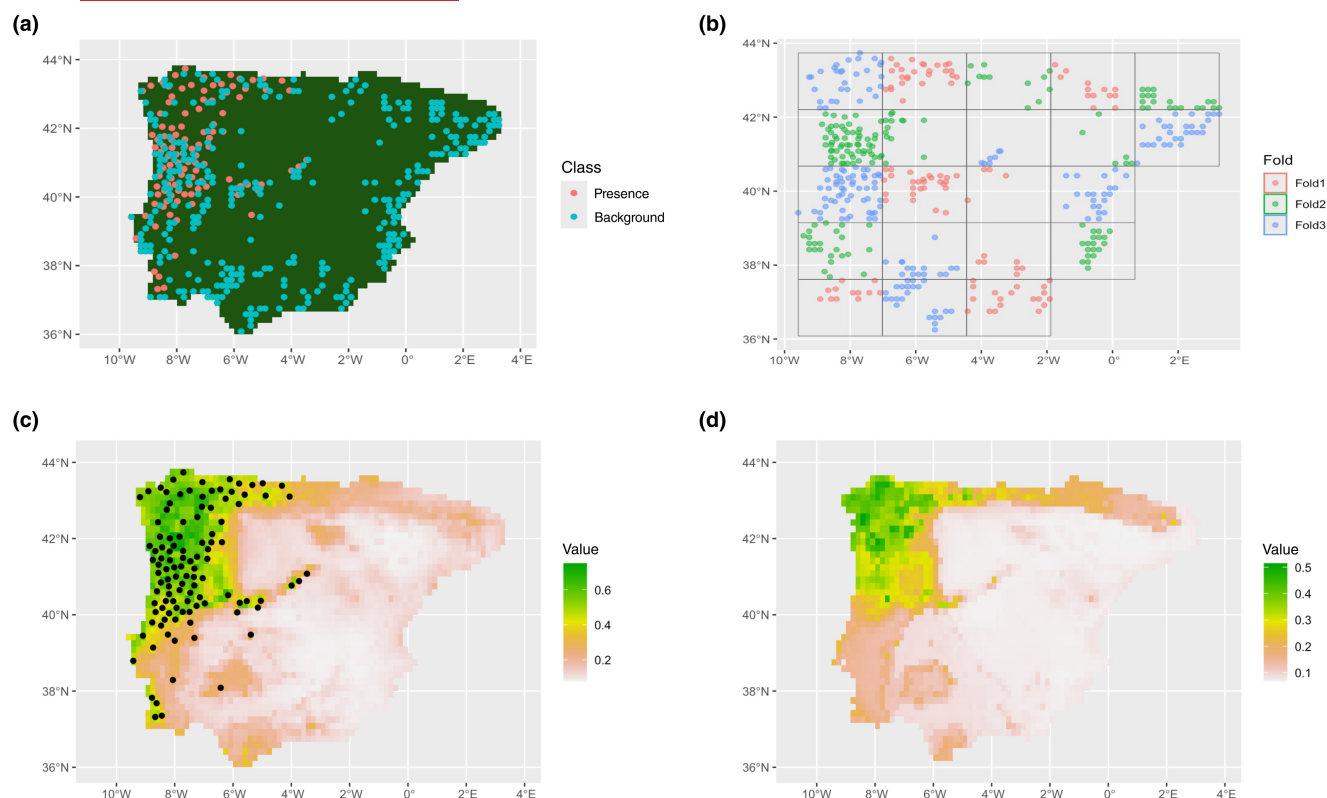


FIGURE 1 Outputs of the species distribution modelling (SDM) pipeline using present-day occurrences (Example 1). (a) thinned presences and sampled background, biased using the records of eight genera of Lacertidae from the Iberian peninsula; (b) spatial blocks for cross-validation; (c) projection of the ensemble over the present; (d) projection of the ensemble over the future.

trees), which define default hyperparameter values and which of these should be tuned based on common SDM practices.

Before evaluating the models, we need to define which metrics to use to assess the model's goodness-of-fit. This is done with the *yardstick* package, containing metrics that can be used, for example, to tune hyperparameters with cross-validation, or to assess against a test dataset not used for training. Similarly to *parsnip*, *yardstick* both implements widely used metrics (e.g. specificity and sensitivity) and provides an infrastructure to add custom ones. In *tidysdm*, we add commonly adopted metrics in SDM, such as true skill statistic (TSSmax) and the Boyce Continuous Index. They provide a blueprint for adding custom ones, and there is extensive documentation in the *yardstick* package on how to do so. Again, the advantage of *tidymodels* is that we can add custom metrics as new user-defined functions, and directly integrated into a pipeline without the need to modify any package.

Having chosen the models and metrics of interest, it is time to 'tune' the algorithms; for each algorithm, this process consists of exploring different values for the hyperparameters and choose, based on the previously defined cross-validation folds, which combination of values gives the best predictions. The package *tune* (Kuhn, 2024) provides the standard grid approach where multiple combinations of hyperparameter values are explored, but it is also possible to use alternative approaches such as Bayesian or racing algorithms, which adaptively find the best hyperparameter combinations. Thanks to the *workflows* package (Vaughan & Couch, 2024), it is easy to standardise

how each algorithm is treated, providing an easy way to parallelise computation (both on a single machine or splitting it on a cluster, if models are large and/or memory and computationally intensive).

After having tuned the workflow, *tidysdm* allows the creation of ensembles. *simple_ensemble()* does so by keeping the best version of each algorithm (i.e. the one with the highest value of the chosen metric). Users can also filter algorithms based on their goodness-of-fit metric, thus using in the ensemble only the algorithms that exceed a given threshold. The function *repeat_ensemble()* assembles ensembles when the process of filtering data and sampling background/pseudo-absences is repeated multiple times to explore the effect of stochasticity. Thanks to the modularity of *tidymodels*, any combination of data-wrangling functions, algorithms and metrics can be combined within one of these repeat ensembles. It is also possible to use more complex ensemble approaches, and the package *stacks* (Couch & Kuhn, 2024) provides an implementation of stacking ensembles, an approach commonly used in ML outside SDMs.

The final step of an SDM workflow is generally to project the models onto a map, both for the present and the future or the past. Besides providing standard *predict()* functions for ensembles that take tibbles or data frames of predictors, *tidysdm* also has a *predict_raster()* function that can take *terra* *SpatRaster* objects as inputs and directly return rasters (Figures 1c and 2d). To assess the validity of predictions, there is an implementation of Multivariate environmental similarity surfaces (MESS, Elith et al., 2010) with *extrapol_mess()*, and the

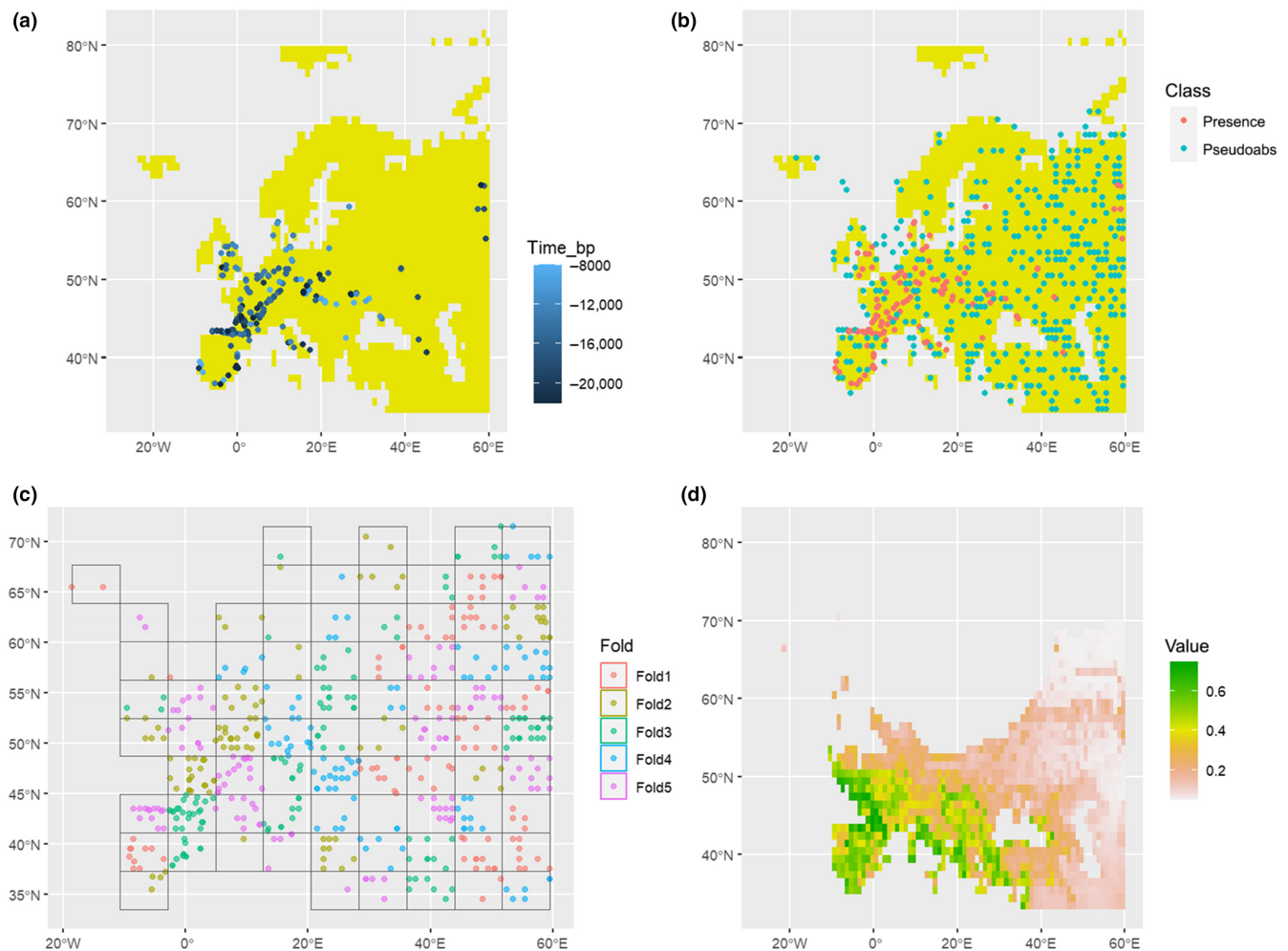


FIGURE 2 Outputs of the species distribution modelling (SDM) pipeline using time-scattered occurrences (Example 2). (a) Presences included in the analyses; (b) thinned presences and sampled pseudo-absences; (c) Spatial blocks for cross-validation; (d) Projection of the ensemble over the Last Glacial Maximum (21,000 years BP).

package *waywiser* (Mahoney, 2023) provides functions for estimating the area of applicability (*sensu* Meyer & Pebesma, 2021).

4.2 | Applications with time-scattered data

The use of time-scattered presences (e.g. radiocarbon dates) can be challenging with conventional SDM packages: Most packages assume that all predictors (e.g. climate) come from a single raster (i.e. a single point in time) where bands represent different variables. *tidysdm* provides alternative versions for all its data-handling functions that can use *SpatRasterDatasets* objects from the *terra* package, where each variable has its own dataset with multiple time points. Thus, it is straightforward to thin presences (Figure 2a), sample background/pseudo-absences (Figure 2b) and wrap the resulting data into a recipe that can then be used for an SDM workflow. Functions that allow for a time dimension have the suffix *_time()*, such as *thin_by_cell_time()* or *sampling_background_time()*. We provide a worked example of how

these functions can be used in a vignette reconstructing the niche of horses from radiocarbon dates (https://evolecolgroup.github.io/tidysdm/articles/a1_palaeodata_application.html).

5 | CONCLUSIONS

tidysdm provides a modular framework that leverages the power of the *tidymodels* universe, with applications for both modern SDM (e.g. in ecology, conservation, biogeography and zoology) and paleo-communities (palaeoecology, archaeology, archaeozoology, palaeontology, evolutionary ecology, macroevolution and population genetics/genomics). A major advantage is that users can integrate any custom algorithm or metric directly into their workflow with minimal effort, without having to modify any package. Furthermore, any addition to the *tidymodels* universe (e.g. stacked ensembles) is immediately usable for SDMs, allowing developments from the broader community of ML practitioners to be quickly adopted.

AUTHOR CONTRIBUTIONS

Andrea Manica wrote the R package and the function documentation and built the website with inputs from Margherita Colucci, Andrea Vittorio Pozzi and Michela Leonardi. Michela Leonardi defined the SDM pipelines with inputs from Andrea Manica and Margherita Colucci. Andrea Manica and Margherita Colucci wrote the vignettes with inputs from Andrea Vittorio Pozzi and Michela Leonardi. Michela Leonardi and Margherita Colucci designed the figures. All authors tested the package and wrote the manuscript. Andrea Manica, Michela Leonardi and Eleanor M. L. Scerri provided the funding.

ACKNOWLEDGEMENTS

We thank Johanna Pajmans and Sophie Dennis for testing the package during its development.

FUNDING INFORMATION

M.L. and A.M. were funded by the Leverhulme Research grant RPG-2020-317. M.C. and E.M.L.S. were funded by the Lise Meitner Pan-African Evolution Research Group. A.V.P. is supported by the Natural Environment Research Council grant number: NE/S007164/1.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflict of interest.

PEER REVIEW

The peer review history for this article is available at <https://www.webofscience.com/api/gateway/wos/peer-review/10.1111/2041-210X.14406>.

DATA AVAILABILITY STATEMENT

The data, code and vignettes presented in this manuscript are available via <https://doi.org/10.5281/zenodo.13154767> (Leonardi et al., 2024).

PACKAGE INSTALLATION AND AVAILABILITY

The package is open-source and available from CRAN at <https://cran.r-project.org/web/packages/tidysdm/index.html> and github <https://github.com/EvolEcolGroup/tidysdm>. It has a dedicated website at <https://evolecolgroup.github.io/tidysdm/index.html>, which contains a manual, vignettes and all the details about the licence and the dependencies. The examples, vignettes and links included in this manuscript are based on the version of the software available at the time of publication. A copy of it, including all associated data and vignettes, can be found at <https://doi.org/10.5281/zenodo.13154767> (Leonardi et al., 2024). When using *tidysdm*, please cite the present work.

ORCID

Michela Leonardi  <https://orcid.org/0000-0001-8933-9374>

Margherita Colucci  <https://orcid.org/0000-0002-0934-194X>

Andrea Vittorio Pozzi  <https://orcid.org/0000-0003-3208-503X>

Eleanor M. L. Scerri  <https://orcid.org/0000-0002-7468-9977>

Andrea Manica  <https://orcid.org/0000-0003-1895-450X>

REFERENCES

- Araújo, M. B., & New, M. (2007). Ensemble forecasting of species distributions. *Trends in Ecology & Evolution*, 22, 42–47. <https://doi.org/10.1016/j.tree.2006.09.010>
- Biecek, P. (2018). DALEX: Explainers for complex predictive models in R. *Journal of Machine Learning Research*, 19(84), 1–5. <https://doi.org/10.48550/arXiv.1806.08915>
- Boria, R. A., Olson, L. E., Goodman, S. M., & Anderson, R. P. (2014). Spatial filtering to reduce sampling bias can improve the performance of ecological niche models. *Ecological Modelling*, 275, 73–77. <https://doi.org/10.1016/j.ecolmodel.2013.12.012>
- Chamberlain, S., Barve, V., Mcglinn, D., Oldoni, D., Desmet, P., Geffert, L., & Ram, K. (2024). *rgbif*: Interface to the global biodiversity information facility API. R package version 3.7.9.3. <https://CRAN.R-project.org/package=rgbif>
- Couch, S., & Kuhn, M. (2024). *Stacks*: Tidy model stacking. <https://stacks.tidymodels.org/>, <https://github.com/tidymodels/stacks>
- Elith, J. (2017). Predicting distributions of invasive species. In A. Robinson, T. Walshe, M. Burgman, & M. Nunn (Eds.), *Invasive species: Risk assessment and management* (pp. 93–129). Cambridge University Press. <https://doi.org/10.1017/9781139019606.006>
- Elith, J., Kearney, M., & Phillips, S. (2010). The art of modelling range-shifting species. *Methods in Ecology and Evolution*, 1, 330–342. <https://doi.org/10.1111/j.2041-210X.2010.00036.x>
- Elith, J., & Leathwick, J. R. (2009). Species distribution models: Ecological explanation and prediction across space and time. *Annual Review of Ecology, Evolution, and Systematics*, 40, 677–697. <https://doi.org/10.1146/annurev.ecolsys.110308.120159>
- Ferrier, S., Watson, G., Pearce, J., & Drielsma, M. (2002). Extended statistical approaches to modelling spatial pattern in biodiversity in northeast New South Wales. I. species-level modelling. *Biodiversity and Conservation*, 11, 2275–2307. <https://doi.org/10.1023/A:1021374009951>
- Franklin, J. (2023). Species distribution modelling supports the study of past, present and future biogeographies. *Journal of Biogeography*, 50, 1–13. <https://doi.org/10.1111/jbi.14617>
- Guisan, A., Thuiller, W., & Zimmermann, N. E. (2017). *Habitat suitability and distribution models: With applications in R*. Cambridge University Press. <https://doi.org/10.1017/9781139028271>
- Hijmans, R. (2024). *terra*: Spatial data analysis. R package version 1.7-41 <https://github.com/rspatial/terra>
- Hirzel, A. H., Le Lay, G., Helfer, V., Randin, C., & Guisan, A. (2006). Evaluating the ability of habitat suitability models to predict species presences. *Ecological Modelling*, 199, 142–152. <https://doi.org/10.1016/j.ecolmodel.2006.05.017>
- Kuhn, M. (2024). *Tune*: Tidy tuning tools. R package version 1.2.1. <https://github.com/tidymodels/tune>, <https://tune.tidymodels.org/>
- Kuhn, M., & Vaughan, D. (2024). *parsnip*: A common API to modeling and analysis functions. <https://github.com/tidymodels/parsnip>, <https://parsnip.tidymodels.org/>
- Kuhn, M., Vaughan, D., & Hvitfeldt, E. (2024). *yardstick*: Tidy characterizations of model performance. <https://github.com/tidymodels/yardstick>
- Kuhn, M., & Wickham, H. (2024). *tidymodels*: A collection of packages for modeling and machine learning using tidyverse principles. <https://www.tidymodels.org/>
- Lang, M., Binder, M., Richter, J., Schratz, P., Pfisterer, F., Coors, S., Au, Q., Casalicchio, G., Kotthoff, L., & Bischl, B. (2019). *mlr3*: A modern object-oriented machine learning framework in R. *Journal of Open Source Software*, 4(44), 1903 <https://joss.theoj.org/papers/10.21105/joss.01903>
- Leonardi, M., Boschin, F., Boscato, P., & Manica, A. (2022). Following the niche: The differential impact of the last glacial maximum on four European ungulates. *Communications Biology*, 5(1), 1–10. <https://doi.org/10.1038/s42003-022-03993-7>

- Leonardi, M., Boschini, F., Giampoudakis, K., Beyer, R. M., Krapp, M., Bendrey, R., Sommer, R., Boscato, P., Manica, A., Nogues-Bravo, D., & Orlando, L. (2018). Late quaternary horses in Eurasia in the face of climate and vegetation change. *Science Advances*, 4, eaar5589. <https://doi.org/10.1126/sciadv.aar5589>
- Leonardi, M., Colucci, M., Pozzi, A. V., & Manica, A. (2024). *EvolEcolGroup/tidysdm*. v0.9.5 (v0.9.5). *Zenodo* <https://doi.org/10.5281/zenodo.13154767>
- Leonardi, M., Hallett, E. Y., Beyer, R., Krapp, M., & Manica, A. (2023). *pastclim* 1.2: An R package to easily access and use paleoclimatic reconstructions. *Ecography*, 2023, e06481. <https://doi.org/10.1111/ecog.06481>
- Mahoney, M. J. (2023). *waywiser*: Ergonomic methods for assessing spatial models. *arXiv* 2303.11312. <https://doi.org/10.48550/arXiv.2303.11312>
- Mahoney, M. J., Johnson, L. K., Silge, J., Frick, H., Kuhn, M., & Beier, C. M. (2023). Assessing the performance of spatial cross-validation approaches for models of spatially structured data. *arXiv* 2303.07334 <https://doi.org/10.48550/arXiv.2303.07334>
- Meyer, H., & Pebesma, E. (2021). Predicting into unknown space? Estimating the area of applicability of spatial prediction models. *Methods in Ecology and Evolution*, 12(9), 1620–1633. <https://doi.org/10.1111/2041-210X.13650>
- Miller, E. F., Leonardi, M., Beyer, R., Krapp, M., Somveille, M., Somma, G. L., Maisano-Delser, P., & Manica, A. (2021). Post-glacial expansion dynamics, not refugial isolation, shaped the genetic structure of a migratory bird, the Yellow Warbler (*Setophaga petechia*). *bioRxiv*: 2021.05.10.443405 <https://doi.org/10.1101/2021.05.10.443405>
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in python. *Journal of Machine Learning Research*, 12, 2825–2830.
- Phillips, S. J., Dudík, M., Elith, J., Graham, C. H., Lehmann, A., Leathwick, J., & Ferrier, S. (2009). Sample selection bias and presence-only distribution models: Implications for background and pseudo-absence data. *Ecological Applications*, 19, 181–197. <https://doi.org/10.1890/07-2153.1>
- Phillips, S. J., Dudík, M., & Schapire, R. E. (2004). A maximum entropy approach to species distribution modeling. In *Proceedings of the twenty-first international conference on machine learning* (Vol. 83). Association for Computing Machinery. <https://doi.org/10.1145/1015330.1015412>
- Phillips, S. J., Dudík, M., & Schapire, R. E. (2024). Maxent software for modeling species niches and distributions (Version 3.4.1). http://biodiversityinformatics.amnh.org/open_source/maxent/
- R Core Team. (2024). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. <https://www.R-project.org/>
- Renner, I. W., Elith, J., Baddeley, A., Fithian, W., Hastie, T., Phillips, S. J., Popovic, G., & Warton, D. I. (2015). Point process models for presence-only analysis. *Methods in Ecology and Evolution*, 6, 366–379. <https://doi.org/10.1111/2041-210X.12352>
- Rödger, D., & Schulte, U. (2010). Potential loss of genetic variability despite well established network of reserves: The case of the Iberian endemic lizard *Lacerta schreiberi*. *Biodiversity and Conservation*, 19, 2651–2666. <https://doi.org/10.1007/s10531-010-9865-2>
- Sillero, N., Campos, J. C., Arenas-Castro, S., & Barbosa, A. M. (2023). A curated list of R packages for ecological niche modelling. *Ecological Modelling*, 476, 110242. <https://doi.org/10.1016/j.ecolmodel.2022.110242>
- Valavi, R., Elith, J., Lahoz-Monfort, J. J., & Guillera-Aroita, G. (2019). *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>
- Vaughan, D., & Couch, S. (2024). *workflows*: Modeling workflows. R package version 1.1.4. <https://workflows.tidymodels.org>, <https://github.com/tidymodels/workflows>
- Zizka, A., Silvestro, D., Andermann, T., Azevedo, J., Duarte Ritter, C., Edler, D., Farooq, H., Herdean, A., Ariza, M., Scharn, R., Svanteson, S., Wengstrom, N., Zizka, V., & Antonelli, A. (2019). *CoordinateCleaner*: Standardized cleaning of occurrence records from biological collection databases. *Methods in Ecology and Evolution*, 744–751, 744–751. <https://doi.org/10.1111/2041-210X.13152>

How to cite this article: Leonardi, M., Colucci, M., Pozzi, A. V., Scerri, E. M. L., & Manica, A. (2024). *tidysdm*: Leveraging the flexibility of *tidymodels* for species distribution modelling in R. *Methods in Ecology and Evolution*, 15, 1789–1795. <https://doi.org/10.1111/2041-210X.14406>