

Evers Sanne (Orcid ID: 0000-0002-8002-1658)
 Salguero-Gómez Rob (Orcid ID: 0000-0002-6085-4433)
 Levin Sam C. (Orcid ID: 0000-0002-3289-9925)

Methods in Ecology and Evolution

Handling editor: Professor Giovanni Strona

Rpadrino: an R package to access and use PADRINO, an open access database of Integral Projection Models

Sam C. Levin ^{*1,2,3}, Sanne Evers ^{1,2}, Tomos Potter ^{3,4}, Mayra Peña Guerrero ^{1,2}, Dylan Z. Childs ⁵, Aldo Compagnoni ^{1,2,3}, Tiffany M. Knight ^{1,2,6‡}, Roberto Salguero-Gómez ^{3‡}

¹Institute of Biology, Martin Luther University Halle-Wittenberg, Am Kirchtor 1, 06108 Halle (Saale), Germany

²German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig, Deutscher Platz 5e, 04103 Leipzig, Germany

³Department of Zoology, 11a Mansfield Rd, University of Oxford, Oxford, OX1 3SZ, UK

⁴Department of Biological Sciences, Florida State University, 319 Stadium Drive, Tallahassee, FL 32306-4340, USA

⁵Department of Animal and Plant Sciences, University of Sheffield, Sheffield, S10 2TN, UK

⁶Department of Community Ecology, Helmholtz Centre for Environmental Research-UFZ, Theodor-Lieser-Straße 4, 06120, Halle (Saale), Germany

‡ Joint senior authors

*Corresponding Author:

Sam C. Levin

Puschstraße 4, 04103 Leipzig, Germany

email: levisc8@gmail.com

Abstract

1. Discrete time structured population projection models are an important tool for studying population dynamics. Within this field, Integral Projection Models (IPMs) have become a popular method for studying populations structured by continuously

This article has been accepted for publication and undergone full peer review but has not been through the copyediting, typesetting, pagination and proofreading process which may lead to differences between this version and the [Version of Record](#). Please cite this article as doi: [10.1111/2041-210X.13910](https://doi.org/10.1111/2041-210X.13910)

distributed traits (*e.g.* height, weight). Databases of discrete time, discrete state structured population models, for example DATLife (life tables) and COMPADRE & COMADRE (matrix population models), have made quantitative syntheses straightforward to implement. These efforts allow researchers to address questions in both basic and applied ecology and evolutionary biology. Since their introduction in 2000, over 300 works containing IPMs have been published, offering opportunities for ecological synthesis too. We describe a novel framework to quickly reconstruct these models for subsequent analyses using *Rpadrino* R package, which serves as an interface to PADRINO, a new database of IPMs.

2. We introduce an R package, *Rpadrino*, which enables users to download, subset, reconstruct, and extend published IPMs. *Rpadrino* makes use of recently created software, *ipmr*, to provide an engine to reconstruct a wide array of IPMs from their symbolic representations and conduct subsequent analyses. *Rpadrino* and *ipmr* are extensively documented to help users learn their usage.
3. *Rpadrino* currently enables users to reconstruct 280 IPMs from 40 publications that describe the demography of 14 animal and 26 plant species. All of these IPMs are tested to ensure they reproduce published estimates. *Rpadrino* provides an interface to augment PADRINO with external data and modify parameter values, creating a platform to extend models beyond their original purpose while retaining full reproducibility.
4. PADRINO and *Rpadrino* provide a toolbox for asking new questions and conducting syntheses with peer-reviewed published IPMs. *Rpadrino* provides a user-friendly

interface so researchers do not need to worry about the database structure or syntax, and can focus on their research questions and analyses. Additionally, *Rpadrino* is thoroughly documented, and provides numerous examples of how to perform analyses which are not included in the package's functionality.

Keywords: database, demography, elasticity, life history, open access, population dynamics, sensitivity

Spanish abstract

1. Los modelos de proyección de población estructurados de tiempo discreto son una herramienta importante para estudiar la dinámicas poblacionales. Dentro de este campo, los Modelos de Proyección Integral (IPM, en inglés) se han convertido en un método popular para estudiar poblaciones estructuradas de acuerdo con rasgos continuos (por ejemplo, altura, peso). Existentes bases de datos demográficas de tiempo discreto, como por ejemplo DATLife (tablas de vida) y COMPADRE & COMADRE (modelos matriciales de población), han facilitado síntesis cuantitativas. Dichos esfuerzos han permitido a los investigadores abordar cuestiones tanto en ecología básica como aplicada y biología evolutiva. Desde su introducción en el año 2000, se han publicado más de 300 trabajos basados en IPMs, los cuales también ofrecen oportunidades para la síntesis ecológica. Aquí describimos un marco novedoso para reconstruir rápidamente estos modelos para posteriores análisis utilizando el paquete de R *Rpadrino*, el cual sirve como interfaz para *PADRINO*, una nueva base de datos de IPMs.

2. Presentamos el paquete de R Rpadrino, el cual permite descargar, crear subconjuntos, reconstruir y extender IPM publicados. Rpadrino hace uso del software creado recientemente, ipmr, para proporcionar el motor necesario para reconstruir una amplia gama de IPMs a partir de sus representaciones simbólicas y a partir de los cuales realizar análisis. Rpadrino e ipmr están ampliamente documentados para ayudar a sus usuarios a conocer su uso.

3. Actualmente, Rpadrino permite reconstruir 280 IPM de 40 publicaciones, las cuales describen la demografía de 14 especies de animales y 26 de plantas. Estos IPM han sido cuidadosamente investigados para garantizar que reproduzcan las estimaciones publicadas. Rpadrino proporciona una interfaz para aumentar PADRINO con datos externos y modificar los valores de los parámetros, creando una plataforma para ampliar los modelos más allá de su propósito original y manteniendo la total reproducibilidad.

4. PADRINO y Rpadrino conforman una caja de herramientas para examinar nuevas preguntas y realizar síntesis con IPMs publicados. Rpadrino proporciona una interfaz fácil de usar para que los investigadores no tengan que preocuparse por la estructura o la sintaxis de la base de datos y puedan concentrarse en sus preguntas y análisis. Además, Rpadrino está completamente documentado y proporciona numerosos ejemplos de cómo realizar análisis que no están incluidos en la funcionalidad del paquete.

Palabras clave: acceso abierto, base de datos, demografía, dinámica poblacional, elasticidad, historia de vida, sensibilidad

Introduction

Demography provides an excellent approach to examine the ecology (Crone et al. 2011), evolutionary biology (Metcalf & Pavard 2007), and conservation biology of any species (Doak & Morris 2001). Environmental conditions and biotic interactions influence vital rates (*e.g.* survival, development, reproduction) across the entire life cycle, which then govern its short-term and long-term performance (Caswell 2001). A variety of methods exist for combining vital rates into demographic models; discrete-time, structured population models are among the most popular (Crone et al. 2011, Caswell 2001). Indeed, there is a rich history of using such structured population models across a variety of sub-disciplines in ecology (*e.g.* Leslie 1945, Caswell 2001, Easterling et al. 2000, Adler et al. 2010, Ellner et al. 2016).

In ecology, matrix projection models (MPMs) are the most widely used structured population model. MPMs divide the population into discrete classes corresponding to some trait value (*e.g.* developmental state, age, or size), and then model the population using vital rates computed for each class. Researchers have also recognized that, for some species, vital rates are best predicted as a function of one or more continuous traits (*e.g.* size, height, mass), rather than as a function of discrete classes (Easterling et al. 2000). Integral projection models (IPMs), which are continuously structured population models, have become an increasingly important tool for ecologists interested in addressing broad biological questions through a demographic lens (Gonzalez et al. 2021). IPMs combine vital rate functions of continuous traits into projection kernels, which describe how the

abundance and distribution of trait values in a population change in discrete time (Easterling et al. 2000). IPMs have been used to investigate a variety of topics, such as invasive species spread (*e.g.* Jongejans et al. 2011, Erickson et al. 2017), evolutionary stable strategies (*e.g.* Childs et al. 2004), the effect of climate drivers on population persistence (Salguero-Gómez et al. 2012, Compagnoni et al. 2021a), and linking evolutionary feedbacks to population dynamics (Coulson et al. 2011).

In order to reconstruct and use an IPM, researchers need, at a minimum, the symbolic representation of the model and the associated parameter values. Existing demographic databases enter transition values directly, rather than a symbolic version of the model and the values associated with the symbols separately. For example, COMPADRE and COMADRE store transition matrices as numeric matrices (sub-matrices corresponding to survival and development (U), sexual reproduction (F), asexual reproduction (C), and their sum A), rather than symbolic matrices with parameter values separately. In general, this data format limits the variety of potential analyses, because individual matrix elements may be composed of multiple vital rates and this information is lost by storing only the resulting values (*i.e.* the elements of F may be comprised of both probability of reproducing and the per-capita number of propagules produced). To avoid this issue for IPMs, one needs to reconstruct the IPM using the functional form of the kernels and vital rates, as well as the associated parameter estimates. One can use tools that associate the symbols with their values to accomplish this task (*e.g.* metaprogramming and *rlang*, Henry & Wickham 2021). *ipmr* is an *R* package for users to interactively develop their own IPMs from symbolic model representations and parameter estimates, and perform downstream

analyses (Levin et al. 2021). *Rpadrino* extends this framework to include *reconstructing* previously published IPMs that are stored in the PADRINO database.

Here, we introduce *Rpadrino*. *Rpadrino* provides access to PADRINO, an open access database of IPMs. Specifically, PADRINO houses symbolic representations of IPMs, their parameter values, and associated metadata to aid users in selecting appropriate models. *Rpadrino* is an R package that enable users to download PADRINO, manage the dataset locally, modify, reconstruct, and analyze IPMs from PADRINO. In the following, we describe how to interact with PADRINO using *Rpadrino* and discuss future directions for *Rpadrino* and PADRINO. We also provide two case studies that demonstrate (1) how to use PADRINO and *Rpadrino* to reconstruct published IPMs, conduct perturbation analyses, compute some life cycle events, and troubleshoot problems, and (2) how to use *Rpadrino* and *ipmr* to combine PADRINO IPMs with user-specified IPMs, and then how to use PADRINO data with other databases, using BIEN (Maitner et al. 2017) and COMPADRE (Salguero-Gómez et al. 2014) as examples. The latter is intended to demonstrate the potential for *Rpadrino* in broad, interoperable, macro-ecological applications. Finally, our supplementary materials also contain a detailed overview of the PADRINO database, along with the associated assumptions and challenges.

An introduction to IPMs and PADRINO

First, we provide a brief review of how IPMs are structured. The simplest form of the IPM can be expressed as

$$n(z', t + 1) = \int_L^U [P(z', z) + F(z', z) + C(z', z)] n(z, t) dz, (1)$$

where $n(z', t + 1)$ and $n(z, t)$ are the distributions of trait values (z) of individuals in the population at time $t + 1$ and t , $P(z', z)$ is a kernel describing survival and development of existing individuals, $F(z', z)$ is a kernel describing per-capita sexual reproduction, and $C(z', z)$ is a kernel describing per-capita asexual reproduction (*i.e.* clonal reproduction). Each kernel may be comprised of any number of vital rate functions (Ellner, Childs, & Rees 2016). Analytical solutions to the integrals in Eq 1 are not available (Ellner & Rees 2006). Therefore, the integrals are numerically approximated, resulting in a large iteration matrix (typically ranging from 45×45 to 1000×1000 in dimension, based on data from PADRINO), and then some quantities of interest are computed (Ellner, Childs & Rees 2016).

Before introducing *Rpadrino*, we provide a brief overview of PADRINO. PADRINO is an open-access database of integral projection models. PADRINO defines a syntax to symbolically represent IPMs as text strings, and stores the values of those symbols in separate tables. The syntax used is very similar to the mathematical notation of IPMs, and is largely “language-agnostic” (*i.e.* aims to avoid idiosyncrasies of specific programming languages). For example, a survival/growth kernel with the form $P(z', z) = s(z) * G(z', z)$ would be `P = s * G` in PADRINO’s syntax. $G(z', z) = f_G(z' | \mu_g(z), \sigma_g)$ (where f_G denotes a normal probability density function) becomes `G = Norm(mu_g, sd_g)`. This notation should be translatable to many computing languages beyond just *R* (*e.g.* Python or Julia). Additionally, PADRINO stores extensive metadata to help researchers find IPMs that work for their questions. A more complete description of the database, how IPMs are digitized, and the associated challenges is available in the ESM and the project webpage (<https://padrinodb.github.io/Padrino/>, Appendix, Tables S1 and S2).

Rpadrino and ipmr

Rpadrino is an R package that contains functions for downloading the PADRINO database, data querying and management, modifying existing functional forms and parameter values, and reconstructing models. Model reconstruction is powered by the *ipmr* R package (Levin et al. 2021). While users do not need to know how to use *ipmr* to use *Rpadrino*, the two packages are designed to work with and enhance each other. This means that users can combine IPMs reconstructed with *Rpadrino* with IPMs of their own constructed with *ipmr* in a single, coherent analysis (case study 2). Furthermore, users can go from downloading the database to reconstructing IPM objects in as little as 3 function calls. A more in depth workflow is provided below.

The flexibility of IPMs and their broad application across ecology, evolution, and conservation biology mean that there is no fixed set of steps in a workflow using *Rpadrino*. However, there are generally four steps that a researcher must take when using *Rpadrino*. The first step is to identify studies of interest (Figure 1, Step 1a), and, optionally, augment PADRINO's metadata with additional information from other sources (*e.g.* environmental data, GBIF, Figure 1, Step 1b). *Rpadrino* represents PADRINO objects as a list of `data.frames` (referred to as tables in subsequent text). *Rpadrino* uses the shared `ipm_id` column across all tables to track information related to each IPM. Therefore, subsetting relies on identifying the correct `ipm_ids`, and then using those to select the IPMs of interest (Box 1, case study 1 and 2). `data.frames` should be familiar to most R users, and the ability to modify them should readily accommodate the range of further analyses that researchers may be interested in. Users may augment any table with additional information

corresponding to, for example, spatial or temporal covariates from other open access databases. Furthermore, *Rpadrino* provides numerous access functions for metadata that streamline subsetting (Box 1).

The second step in the *Rpadrino* workflow is to construct a list of `proto_ipm` objects using `pdb_make_proto_ipm()` (Figure 1, Box 1). This function translates PADRINO's syntax into *ipmr* code, and then builds a `proto_ipm` object for each unique `ipm_id`. For some models, users may choose to create deterministic or stochastic IPMs at this step. *Rpadrino*'s default behavior is to generate deterministic models whenever possible. This behavior encompasses instances where authors generated models with no time or space varying parameters, and where authors included discretely varying environments. The latter can be implemented as deterministic models because all parameter values are known before the IPM is built. IPMs with continuous environmental variation require sampling the environment at each model iteration, usually by sampling from distributions randomly. These are always considered stochastic models. This is also the step where, if needed, users should combine their own `proto_ipm`'s produced by *ipmr* with the `proto_ipm`'s produced by *Rpadrino*.

The third step in the *Rpadrino* workflow is creating IPM objects with `pdb_make_ipm()` (Figure 1, Box 1). `pdb_make_ipm()` uses *ipmr*'s `make_ipm()` function to build IPM objects. Users may specify additional options to pass to `make_ipm()` (e.g. normalize the population size to always equal 1, return the vital rate function values as well as the sub-kernels and population state). The various arguments users can modify are described in the *ipmr* documentation for `make_ipm()`.

The fourth and final step in an *Rpadrino* workflow is to conduct the analyses of interest (Figure 1, Box 1). *Rpadrino* provides functions to extract per-capita growth rates, eigenvectors (Caswell 2001, Ellner, Childs & Rees 2016 Ch. 2, demonstrated in Box 1), assess convergence to asymptotic dynamics (Caswell 2001), compute mean kernels for stochastic IPMs (Ellner, Childs & Rees Ch 7), and modify existing IPMs with new parameter values and functional forms. Additionally, the documentation on the *Rpadrino* website (<https://padrinODB.github.io/Rpadrino/index.html>) and the Supplementary Materials for this paper contain details on how to conduct more complicated analyses with IPM objects (e.g. perturbation analyses (Ellner, Childs & Rees 2016 Ch 4), size at death calculations (Metcalf et al. 2009)). The package documentation and the recent publication describing *ipmr* also contain code demonstrating analyses on single IPM objects (Levin et al. 2021). These can be extended via the `apply` family of functions.

```
# Install and Load the CRAN version:
# install.packages("Rpadrino")

library(Rpadrino)

# Step 1 from main text -----
# pdb_download() downloads a copy PADRINO. We can specify a path to save the
# downloaded database using `save = TRUE` and
# `destination = 'path/to/file/'`. We'll call the object we create 'pdb',
# which is short for Padrino DataBase.
```

Accepted Article

```
pdb <- pdb_download(save = FALSE)
```

```
# We can use Rpadrino's metadata accessors to get a selection of ipm_ids  
# that we want to use. For this example, we'll select models for Carpobrotus  
# species and Geum radiatum. First, extract the 'species_accepted' column.  
# The output of this will be named, and the names are the ipm_id associated  
# with each piece of metadata. Thus, we can subset the names of the 'spps'  
# object to get the ipm_ids we need.
```

```
spps <- pdb_species_accepted(pdb)
```

```
ids <- names(spps)[spps %in% c("Carpobrotus_spp", "Geum_radiatum")]
```

```
# Step 2 from main text -----
```

```
# Next, we create a list of proto_ipm's using pdb_make_proto_ipm().
```

```
my_proto_ipms <- pdb_make_proto_ipm(pdb, ipm_id = ids)
```

```
# Step 3 from main text -----
```

```
# After creating the proto_ipm list, we can call pdb_make_ipm() to construct  
# actual IPM objects.
```

```
my_ipms <- pdb_make_ipm(my_proto_ipms)
```

```
# Step 4 from main text -----
```

```
# After re-building our published IPMs, the next step is to analyze them.
```

*# In this case, we'll just extract the asymptotic population growth rates,
stable size distribution, and the reproductive values. Note that for the
Geum IPMs, there are multiple year-specific values that are returned.
ALL values related to population-level traits are computed via iteration,
as this approach handles more complicated IPM systems more efficiently
than eigenvector/eigenvalue based approaches for larger IPMs, and
introduces little to no additional computation time for simpler and/or
smaller IPMs.*

```
lambdas <- lambda(my_ipms)
ssds     <- right_ev(my_ipms, iterations = 150, tolerance = 1e-7)
repro_vs <- left_ev(my_ipms , iterations = 150, tolerance = 1e-7)
```

Box 1: *An example of a simple analysis workflow using Rpadrino . The first step in using Rpadrino is to install and load the package. After that, we can use Rpadrino to download PADRINO and, optionally, save it locally on our computer. Once the data are downloaded, we can make use of Rpadrino's metadata accessor functions to quickly select models that meet our criteria (step 1). The concept of the ipm_id is explained in greater detail in the Appendix of this manuscript. The next step is to use these ipm_ids to create a list of proto_ipm's using pdb_make_proto_ipm() (step 2). After this step, we can create actual IPM objects using pdb_make_ipm() (step 3). Once IPM objects are created, the following steps are according to the demands of the research question. In this case, asymptotic population growth rates, stable size distributions, and reproductive values are extracted (step 4). Note that since the Geum radiatum model includes a number of year-specific estimates, multiple values are generated for each quantity we want to extract. The concise representation and reconstruction of*

models such as this is powered by *ipmr*'s parameter set index notation, which is described in greater detail on the package website (<https://levisc8.github.io/ipmr/articles/index-notation.html>). However, users do not need to be familiar with this notation unless they wish to modify the IPM in question (see case study 1 for an example of modifying PADRINO IPMs with *Rpadrino*).

Challenges

There are numerous challenges associated with reproducing published IPMs. Challenges related to digitizing and storing IPMs are discussed in the ESM. Important challenges remain in the reconstruction of IPMs. Semi- or non-parametric models may be used to generate IPMs whose functional form is not known *a priori*. We have not yet developed a general syntax for representing these models in PADRINO, though work is ongoing. Additionally, *ipmr* is not yet able to handle two-sex models (*e.g.* Stubberud et al. 2019), time-lagged models (*e.g.* Kuss et al. 2008), or periodic models (*e.g.* Letcher et al. 2014). These types of IPMs do not yet represent a substantial portion of the literature. Nonetheless, it is our intention to continue developing functionality to accommodate them in future releases of *Rpadrino*, *ipmr*, and PADRINO.

Opportunities and Future Directions

Rpadrino presents unique opportunities for synthesis in both theoretical and applied contexts. The expanded range of phylogenetic and geographical coverage can be used in conjunction with other demographic databases (*e.g.* COM(P)ADRE (Salguero-Gómez et

Accepted Article

al. 2014, Salguero-Gómez et al. 2016), popler (Compagnoni et al. 2019), DatLife (DatLife 2021)) to power larger scale syntheses than were possible before (e.g. Compagnoni et al. 2021b). For example, one could use IPMs from PADRINO and matrix population models from COMPADRE and COMADRE to create life tables (Jones et al. 2021), which could then be combined with life tables from DATLife for further analysis (e.g. Jones et al. 2014). The intermediate life table conversion steps may not be necessary, as many of the same life history traits and population level parameters may be calculated from all of these models (Caswell 2001, Ellner, Childs & Rees 2016). Furthermore, recent publications combine biotic and abiotic interactions into demographic models providing a robust theoretical toolbox for exploring species responses to environmental drivers such as climate change (e.g. Simmonds et al. 2020, Abrego et al. 2021). *Rpadrino* also provides functionality to modify parameter values and functional forms of the IPMs it stores, giving theoreticians a wide array of realistic life histories to experiment with. These features will enable researchers to carry out more detailed and comprehensive analyses at various spatial, temporal, and phylogenetic scales. The examples given here are far from an exhaustive list, but hopefully demonstrates the potential for this new tool in demography, ecology, and evolutionary biology.

Table 1: Taxonomic representation of IPMs accessible via *Rpadrino*. These numbers represent the number of models that are error checked and accurately reproduce the published IPM (see 'Data Validation' in the Appendix for more details). Models that are partially entered or still

contain errors are not considered here. We are in the process of correcting them and/or retrieving additional information from the authors. See Appendix for details.

Kingdom	# of Unique ipm_ids	# of Unique Species	# of Publications
Totals	280	56	40
Animalia	22	16	14
Plantae	258	40	26

Acknowledgments

We want to thank L Sfedu for help designing the figures, and the associate editor and two anonymous reviewers for comments that greatly improved this manuscript.

Funding: R.S-G. was supported by a NERC Independent Research Fellowship (NE/M018458/1). SCL, AC, SE, and TMK were funded by the Alexander von Humboldt Foundation in the framework of the Alexander von Humboldt Professorship of TM Knight.

Conflict of Interest statement: The authors declare no conflicts of interest.

Author Contributions: Sam C. Levin designed *ipmr* and *Rpadrino* with contributions from all authors, and Sam C. Levin implemented the packages. Sanne Evers, Tomos Potter, Mayra Pena-Guerrero, and Sam C. Levin entered the data into PADRINO. Sam C. Levin wrote the first draft of the manuscript and all authors provided comments.

Data availability: PADRINO (Levin et al. 2022) is available via the *Rpadrino* R package, as well as on Github (<https://github.com/padrinoDB/Padrino>) and Zenodo (<https://zenodo.org/badge/latestdoi/109448718>). *Rpadrino* (Levin et al. 2022) is

available on CRAN (<https://cran.r-project.org/package=Rpadrino>), Github (<https://github.com/padrinoDB/Rpadrino>), and Zenodo (<https://zenodo.org/badge/latestdoi/124245125>). *pdbDigitUtils* (Levin SC 2022) is available on Github (<https://github.com/padrinoDB/pdbDigitUtils>) and Zenodo (<https://zenodo.org/badge/latestdoi/348737812>). There is no other data associated with this paper.

Citations

1. Easterling, M.R., Ellner, S.P., & Dixon, P.M. (2000). Size specific sensitivity: applying a new structured population model. *Ecology* 81(3): 694-708.
2. Ellner, S.P. & Rees, M. (2006). Integral Projection Models for species with complex demography. *The American Naturalist* 167(3): 410-428.
3. DATLife – The Demography Across the Tree of Life – database. Max-Planck Institute for Demographic Research (Germany). Available at www.datlife.org. (2021)
4. Salguero-Gómez, R., Jones, O.R., Archer, C.R., Bein, C., de Buhr, H., Farack, C., Gottschalk, F., Hartmann, A., Henning, A., Hoppe, G., Roemer, G., Ruoff, T., Sommer, V., Wille, J. Voigt, J., Zeh, S., Viereg, D., Buckley, Y.M., Che-Castaldo, J., Hodgson, D., et al. (2016) COMADRE: a global database of animal demography. *Journal of Animal Ecology* 85: 371-384. <https://doi.org/10.1111/1365-2656.12482>
5. Salguero-Gómez, R., Jones, O.R., Archer, C.A., Buckley, Y.M., Che-Castaldo, J., Caswell, C., Hodgson, D., Scheuerlein, A., Conde, D.A., Brinks, E., de Buhr, H., Farack, C., Gottschalk, F., Hartmann, A., Henning, A., Hoppe, G., Roemer, G., Runge, J., Ruoff, T., et

al. (2014) The COMPADRE Plant Matrix Database: an online repository for plant population dynamics. *Journal of Ecology* 103: 202-218.

<https://doi.org/10.1111/1365-2745.12334>

6. Compagnoni, A., Pardini, E., & Knight, T.M. (2021a). Increasing temperature threatens an already endangered coastal plant species. *Ecosphere* 12(3): e03454. <https://doi.org/10.1002/ecs2.3454>
7. Compagnoni, A., Levin, S.C., Childs, D.Z., Harpole, S., Paniw, M., Roemer, G., Burns, J.H., Che-Castaldo, J., Rueger, N., Kunstler, G., Bennett, J.M., Archer, C.R., Jones, O.R., Salguero-Gómez, R., & Knight, T.M. (2021b). Herbaceous perennial plants with short generation time have stronger responses to climate anomalies than those with longer generation time. *Nature Communications* 12: 1824. <https://doi.org/10.1038/s41467-021-21977-9>
8. Levin, S.C., Childs, D.Z., Compagnoni, A., Evers, S., Knight, T.M., & Salguero-Gómez, R. (2021) ipmr: Flexible implementation of Integral Projections Models in R. *Methods in Ecology and Evolution*. <https://doi.org/10.1111/2041-210X.13683>
9. Jongejans, E., Shea, K., Skarpaas, O., Kelly, D., & Ellner, S.P. (2011). Importance of individual and environmental variation for invasive species spread: a spatial integral projection model. *Ecology* 92(1): 86-97. <https://doi.org/10.1890/09-2226.1>
10. Erickson, R.A., Eager, E.A., Brey, M.B., Hansen, M.J., & Kocovsky, P.M. (2017). An integral projection model with YY-males and application to evaluating grass carp

control. *Ecological Modelling* 361: 14-25.

<https://doi.org/10.1016/j.ecolmodel.2017.07.030>

11. Childs, D.Z., Rees, M., Rose, K.E., Grubb, P.J., & Ellner, S.P. (2004). Evolution of size-dependent flowering in a variable environment: construction and analysis of a stochastic integral projection model. *Proceedings of the Royal Society B* 271(1547): 425-434. <https://doi.org/10.1098/rpsb.2003.2597>
12. Ellner, S.P., Childs, D.Z., Rees, M. (2016) Data-driven modelling of structured populations: a practical guide to the integral projection model. Basel, Switzerland: Springer International Publishing AG
13. Leslie, P.H. (1945). The use of matrices in certain population mathematics. *Biometrika* 33(3): 183-212.
14. Caswell, H. (2001) Matrix population models: construction, analysis, and interpretation, 2nd edn. Sunderland, MA: Sinauer Associates Inc
15. Adler, P.B., Ellner, S.P. & Levine, J.M. (2010). Coexistence of perennial plants: an embarrassment of niches. *Ecology Letters* 13: 1019-1029. <https://doi.org/10.1111/j.1461-0248.2010.01496.x>
16. Abrego, N., Roslin, T., Huotari, T., Ji, Y., Schmidt, N.M., Wang, J., Yu, D.W., & Ovaskainen, O. (2021) Accounting for species interactions is necessary for predicting how arctic arthropod communities respond to climate change. *Ecography*. <https://doi.org/10.1111/ecog.05547>

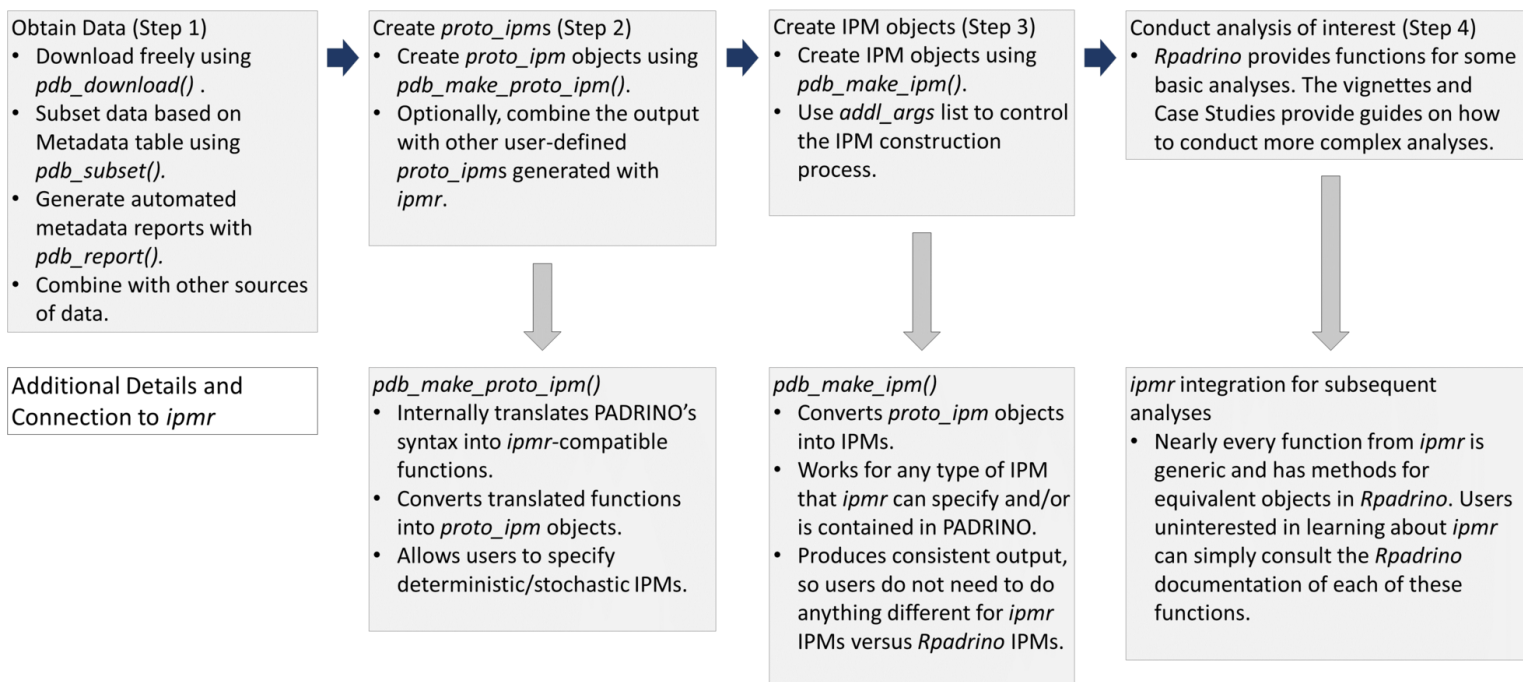
- Accepted Article
17. Henry, L., & Wickham, H. (2021). rlang: Functions for Base Types and Core R and 'Tidyverse' Features. R package version 0.4.11. <https://CRAN.R-project.org/package=rlang>
 18. Crone, E.E., Menges, E.S., Ellis, M.M., Bell, T., Bierzychudek, P., Ehrlen, J. et al. (2011) How do ecologists use matrix population models? *Ecology Letters* 14(1): 1-8. DOI: <https://doi.org/10.1111/j.1461-0248.2010.01540.x>
 19. Doak, D. & Morris W.F. (2002). *Quantitative Conservation Biology: Theory and Practice of Population Viability Analysis*. Oxford, UK: Oxford University Press
 20. Salguero-Gómez, R. & Gamelon M. (2021). *Demographic Methods Across the Tree of Life*. Oxford, UK: Oxford University Press
 21. Coulson, T., MacNulty, D.R., Stahler, D.R., von Holdt, B., Wayne, R.K., & Smith, D.W. (2011). Modeling effects of environmental change on wolf population dynamics, trait evolution, and life history. *Science* 334(6060): 1275-1278. <https://doi.org/10.1126/science.1209441>
 22. Stott, I., et al. (2011). A framework for studying transient dynamics of population projection matrix models. *Ecology Letters* 14: 959-970. DOI: <https://doi.org/10.1111/j.1461-0248.2011.01659.x>
 23. Clark, J.S. (2003) Uncertainty and variability in demography and population growth: A hierarchical approach. *Ecology* 84(6): 1370-1381.
 24. Kuss, P., Rees, M., Aegisdottir, H.H., Ellner, S.P., & Stoecklin, J. (2008). Evolutionary demography of long-lived monocarpic perennials: A time-lagged integral projection

model. Journal of Ecology 96: 821-832. DOI: <https://doi.org/10.1111/j.1365-2745.2008.01374.x>

25. Letcher, B.H., Schueller, P., Bassar, R.D., Nislow, K.H., Coombs, J.A., Sakrejda, K. *et al.* (2014). Robust estimates of environmental effects on population vital rates: an integrated capture-recapture model of seasonal brook trout growth, survival and movement in a stream network. Journal of Animal Ecology 84(2): 337-352. <https://doi.org/10.1111/1365-2656.12308>
26. Stubbred, M.W., Vindenes, Y., Vollestad, L.A, Winfield, I.J., Stenseth, N.C., & Langangen, O. (2019). Effects of size- and sex-selective harvesting: an integral projection model approach. Ecology and Evolution 9: 12556-12570. <https://doi.org/10.1002/ece3.5719>
27. Maitner, B., Boyle B., Casler N., Condit R., Donoghue J., Duran S.M., *et al.* (2017) The bien r package: A tool to access the Botanical Information and Ecology Network (BIEN) database. Methods in Ecology and Evolution 9(2): 373-379. <https://doi.org/10.1111/2041-210X.12861>
28. Simmonds E.G., Cole, E.F., Sheldon, B.C., & Coulson, T. (2020). Phenological asynchrony: a ticking time-bomb for seemingly stable populations? Ecology Letters 23(12): 1766-1775. <https://doi.org/10.1111/ele.13603>
29. Kousta, S., Pastrana, E., & Swaminathan, S. (2019). Three approaches to support reproducible research. Science Editor 42(3): 77-82.

30. Jones, O.R., Scheuerlein, A., Salguero-Gómez, R., Camarda, C.G., Schaible, R., Casper, B.B., *et al.* (2014). The diversity of ageing across the tree of life. *Nature* 505: 169-173. <https://doi.org/10.1038/nature12789>
31. Jones, O.R., Barks, P., Stott, I., James, T.D., Levin, S.C., Petry, W.K. *et al.* (2021). Rcompadre and Rage - two R packages to facilitate the use of the COMPADRE and COMADRE databases and calculation of life history traits from matrix population models. *Methods in Ecology and Evolution* 13(4): 770-781. <https://doi.org/10.1111/2041-210X.13792>
32. Levin, S.C., Evers, S., Potter, T., Guerrero, M.P., Childs, D.Z., Compagnoni, A., Knight, T.M., & Salguero-Gómez, R. (2022). PADRINO v0.0.1. <https://doi.org/10.5281/zenodo.6573870>
33. Levin, S.C., Evers, S., Potter, T., Guerrero, M.P., Childs, D.Z., Compagnoni, A., Knight, T.M., & Salguero-Gómez, R. (2022). Rpadrino v0.0.4.1: Methods in Ecology and Evolution Release. <https://doi.org/10.5281/zenodo.6573880>
34. Levin, S., C. (2022). pdbDigitUtils: Methods in Ecology and Evolution Release. <https://doi.org/10.5281/zenodo.6573882>

Rpadrino Workflow



MEE3_13910_figure_1.png