Article:

APPLICATION

Methods in Ecology and Evolution | BRITISH ECOLOGICAL SOCIETY

ipmr: Flexible implementation of Integral Projection Models in R

Sam C. Levin[1,2,3] 🆔 | Dylan Z. Childs[4] 🆔 | Aldo Compagnoni[1,2,3] 🆔 | Sanne Evers[1,2,5] 🆔 |
Tiffany M. Knight[1,2,5] 🆔 | Roberto Salguero-Gómez[3] 🆔

[1]Institute of Biology, Martin Luther University Halle-Wittenberg, Halle (Saale), Germany

[2]German Centre for Integrative Biodiversity Research (iDiv) Halle-Jena-Leipzig, Leipzig, Germany

[3]Department of Zoology, University of Oxford, Oxford, UK

[4]Department of Animal and Plant Sciences, University of Sheffield, Sheffield, UK

[5]Department of Community Ecology, Helmholtz Centre for Environmental Research-UFZ, Halle (Saale), Germany

Correspondence
Sam C. Levin
Email: levisc8@gmail.com

Abstract

1. Integral projection models (IPMs) are an important tool for studying the dynamics of populations structured by one or more continuous traits (e.g. size, height, body mass). Researchers use IPMs to investigate questions ranging from linking drivers to population dynamics, planning conservation and management strategies, and quantifying selective pressures in natural populations. The popularity of stage-structured population models has been supported by *R* scripts and packages (e.g. IPMpack, popbio, popdemo, lefko3) aimed at ecologists, which have introduced a broad repertoire of functionality and outputs. However, pressing ecological, evolutionary and conservation biology topics require developing more complex IPMs, and considerably more expertise to implement them. Here, we introduce ipmr, a flexible *R* package for building, analysing and interpreting IPMs.

2. The ipmr framework relies on the mathematical notation of the models to express them in code format. Additionally, this package decouples the model parameterization step from the model implementation step. The latter point substantially increases ipmr's flexibility to model complex life cycles and demographic processes.

3. ipmr can handle a wide variety of models, including those that incorporate density dependence, discretely and continuously varying stochastic environments, and multiple continuous and/or discrete traits. ipmr can accommodate models with individuals cross-classified by age and size. Furthermore, the package provides methods for demographic analyses (e.g. asymptotic and stochastic growth rates) and visualization (e.g. kernel plotting).

4. ipmr is a flexible *R* package for integral projection models. The package substantially reduces the amount of time required to implement general IPMs. We also provide extensive documentation with six vignettes and help files, accessible from an R session and online.

KEYWORDS

elasticity, integral projection model, life history, population dynamics, population growth rate, sensitivity, structured populations

---

1 | INTRODUCTION

Integral projection models (IPMs) are an important and widely used tool for ecologists studying structured population dynamics in discrete time. Since the paper introducing IPMs was published over two decades ago (Easterling et al., 2000), at least 255 peer-reviewed publications on at least 250 plant species and 60 animal species have used IPMs (ESM, Table S1; Figure S1). These models have addressed questions ranging from invasive species population dynamics (e.g. Crandall & Knight, 2017), effect of climate drivers on population persistence (e.g. Compagnoni et al., 2021), evolutionary stable strategies (e.g. Childs et al., 2004) and rare/endangered species conservation (e.g. Ferrer-Cervantes et al., 2012).

The IPM was introduced as alternative to matrix population models, which model populations structured by discrete traits (Caswell, 2001). Some of the advantages of using an IPM include (a) the ability to model populations structured by continuously distributed traits, (b) the ability to flexibly incorporate discrete and continuous traits in the same model (e.g. seeds in a seedbank and a height-structured plant population, Crandall & Knight, 2017, or number of females, males and age-1 recruits for fish species, Erickson et al., 2017), (c) efficient parameterization of demographic processes with familiar regression methods (Coulson, 2012), and (d) the numerical discretization of continuous kernels (see below) means that the tools available for matrix population models are usually also applicable for IPMs. Furthermore, researchers have developed methods to incorporate spatial dynamics (Jongejans et al., 2011), environmental stochasticity (Rees & Ellner, 2009) and density/frequency dependence into IPMs (Adler et al., 2010; Ellner et al., 2016). These developments were accompanied by the creation of software tools and guides to assist with IPM parameterization, implementation and analysis. These tools range from R scripts with detailed annotations (Coulson, 2012; Ellner et al., 2016; Merow et al., 2014) to R packages (Metcalf et al., 2013; Shefferson et al., 2020).

Despite the array of resources available to researchers, implementing an IPM is still not a straightforward exercise. For example, an IPM that simulates a population for 100 time steps requires the user to either write or adapt from published guides multiple functions (e.g. to summarize demographic functions into the proper format), implement the numerical approximations of the model's integrals, ensure that individuals are not accidentally sent beyond the integration bounds ('unintentional eviction', sensu Williams et al., 2012) and track how the population state changes over the course of a simulation. Stochastic IPMs present further implementation challenges. In addition to the aforementioned elements, users must generate the sequence of environments that the population experiences. There are multiple ways of simulating environmental stochasticity, each with their own strengths and weaknesses (Metcalf et al., 2015).

ipmr manages these key details while providing the user flexibility in their models. ipmr uses the rlang package for metaprogramming (Henry & Wickham, 2020), which enables ipmr to provide a miniature domain-specific language for implementing

IPMs. ipmr aims to mimic the mathematical syntax that describes IPMs as closely as possible (Figure 1; Box 1; Tables 1 and 2). This R package can handle models with individuals classified by a mixture of any number of continuously and discretely distributed traits. Furthermore, ipmr introduces specific classes and methods to deal with both discretely and continuously varying stochastic environments, density-independent and -dependent models, as well as age-structured populations (Case Study 2). ipmr decouples the parameterization (i.e. regression model fitting) and implementation steps (i.e. converting the regression parameters into a full IPM), and does not attempt to help users with the parameterization task. This provides greater flexibility in modelling trait–demography relationships, and enables users to specify IPMs of any functional form that they desire.

2 | TERMINOLOGY AND IPM CONSTRUCTION

An IPM describes how the abundance and distribution of trait values (also called *state variables/states*, denoted z and z') for a population changes in discrete time. The distribution of trait values in a population at time t is given by the function $n(z, t)$. A simple IPM for the trait distribution z' at time $t + 1$ is then

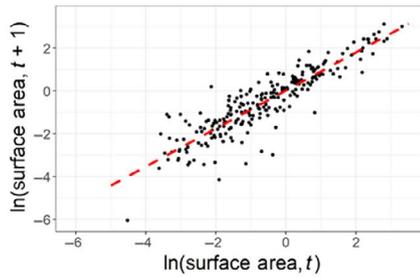$$n(z', t + 1) = \int_L^U K(z', z)\, n(z, t)\, dz. \qquad (1)$$

$K(z', z)$, known as the *projection kernel*, describes all possible transitions of existing individuals and recruitment of new individuals from t to $t + 1$, generating a new trait distribution $n(z', t + 1)$. L, U are the lower and upper bounds for values that the trait z can have, which defines the *domain* over which the integration is performed. The integral $\int_L^U n(z, t)\, dz$ gives the total population size at time t.

To make the model more biologically interpretable, the projection kernel $K(z', z)$ is usually split into *sub-kernels* (Equation 2). For example, a projection kernel to describe a life cycle where individuals can survive, transition to different state values, and reproduce via sexual and asexual pathways, can be split as follows.

$$K(z', z) = P(z', z) + F(z', z) + C(z', z), \qquad (2)$$

where $P(z', z)$ is a sub-kernel describing transitions due to survival and trait changes of existing individuals, $F(z', z)$ is a sub-kernel describing per-capita sexual contributions of existing individuals to recruitment and $C(z', z)$ is a sub-kernel describing per-capita asexual contributions of existing individuals to recruitment. The sub-kernels are typically comprised of functions derived from regression models that relate an individual's trait value z at time t to a new trait value z' at $t + 1$. For example, the P kernel for Soay sheep *Ovis aries* on St. Kilda (Equation 3) may contain two regression models: (a) a logistic regression of survival on log body mass (Equation 4) and (b)

Mathematical/graphical notation

ipmr/R representation

Fit vital rate models (1)

Other packages
(e.g. lme4, brms, mgcv, stats, nlme)

lm(size_2 ~ size_1)

Deterministic:

$$n(z', t+1) = \int_L^U K(z', z) n(z, t) dz$$

Parameter re-sampled stochastic:

$$n(z', t+1) = \int_L^U K(z', z, \theta) n(z, t) dz$$

Decide if IPM is (2)
simple/general,
density-(in)dependent,
deterministic/stochastic,
parameter/kernel stochastic

init_ipm("simple", "di", "det")

init_ipm("simple", "di", "stoch", "param")
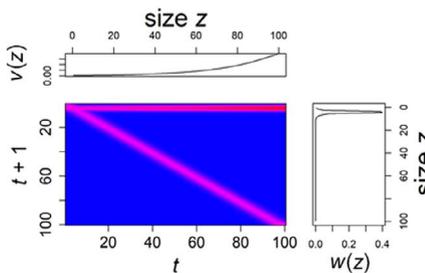
Sub-kernel formula:
$$P(z', z) = s(z) * G(z', z)$$
Vital rate expressions:
$$\text{Logit}(s)) = \beta_0 + \beta_1 * z$$
$$G(z', z) = f_G(z', \mu(z), \sigma)$$
$$\mu(z) = \beta_0 + \beta_1 * z$$

Symbolically define kernels (3)

define_kernel(
name = "P",
formula = s*G,
s = plogis(s_int + s_slope * z_1),
G = dnorm(z_2, mu_G, sigma_G),
mu_G = G_int + G_slope * z_1)

$$L = 1.2, U = 7.8, N_{\text{mesh}} = 100$$
$$\theta \sim \text{Norm}(0, \sigma_{\text{Temp}})$$

Numerically define (4)
kernels and initial conditions

define_ipml(),
define_domains(),
define_pop_state(),
define_env_state()

Generate model object (5)

make_ipm()

Population level inference (6)

lambda(), right_ev(), left_ev(),
mean_kernel(), plot(),
other packages (e.g. popbio, popdemo)

FIGURE 1 There are generally six steps in defining an IPM with ipmr. (1) Vital rate models are fit to demographic data collected from field sites. This step requires the use of other packages, as ipmr does not contain facilities for regression modelling. The figure on the left shows the fitted relationship between size at t and $t + 1$ for *Carpobrotus* spp. in Case Study 1. (2) The next step is deciding what type of IPM is needed. This is determined by both the research question and the data used to parameterize the regression models. This process is initiated with init_ipm(). In step (3), kernels are defined using ipmr's syntax to represent kernels and vital rate functions. (4) Having defined symbolic representations of the model, the numerical definition is given. Here, the integration rule, domain bounds and initial population conditions are defined. For some models, initial environmental conditions can also be defined. (5) make_ipm() numerically implements the proto_ipm object, (6) which can then be analysed further. The figure at the bottom left shows a $K(z', z)$ kernel created by make_ipm() and make_iter_kernel(). The line plots above and to the right display the left and right eigenvectors, extracted with left_ev() and right_ev(), respectively

a linear regression of log body mass at $t + 1$ on log body mass at t (Equations 5–6). In this example, f_G is a normal probability density function with μ_G given by the linear predictor of the mean, and with σ_G computed from the standard deviation of the residuals from the linear regression model.

$$P(z', z) = s(z) * G(z', z), \qquad (3)$$

$$\text{Logit}(s(z)) = \alpha_s + \beta_s * z, \qquad (4)$$

$$G(z', z) = f_G(z', \mu_G, \sigma_G), \qquad (5)$$

BOX 1 Code to implement a simple IPM from parameter estimates in ipmr. Because ipmr does not include functions to assist with regression modelling, this example skips the step of working with actual data and instead uses hypothetical parameter values. We see that given this set of conditions, if nothing were to change, the population would increase by ~2% each year. The case studies provide details on further use cases and analyses that are possible with ipmr.

```r
library(ipmr)

# This section produces the result of Step 1 in Figure 1.

data_list <- list(
    s_i = -0.65,         # Intercept of the survival model (Logistic regression)
    s_z = 0.75,          # Slope of the survival model
    G_i = 0.96,          # Intercept of the growth model (Gaussian regression)
    G_z = 0.66,          # Slope of the growth model
    sd_G = 0.67,         # Standard deviation of residuals of growth model
    mu_r = -0.08,        # Mean of the recruit size distribution
    sd_r = 0.76,         # Standard deviation of the recruit size distribution
    r_n_i = -1,          # Intercept of recruit production model (Poisson regression)
    r_n_z = 0.3          # Slope of recruit production model.
)

#  Step 2 in Figure 1. This is how ipmr initializes a model object.
#  All functions prefixed with define_* generate proto_ipm objects. These
#  are converted into IPMs using the make_ipm() function in step 5.

example_proto_ipm <- init_ipm(sim_gen   = "simple",
                              di_dd     = "di",
                              det_stoch = "det")

# Step 3 in Figure 1. Note the link between how the model was defined
# mathematically and how it is defined here.

example_proto_ipm <- define_kernel(
    example_proto_ipm,
    name          = "P",
    formula       = surv * Grow,
    surv          = plogis(s_i + s_z * z_1),
    Grow          = dnorm(z_2, mu_G, sd_G),
    mu_G          = G_i + G_z * z_1,
    data_list     = data_list,
    states        = list(c("z"))
)

example_proto_ipm <- define_kernel(
    example_proto_ipm,
    name          = "F",
    formula       = recr_number * recr_size,
    recr_number   = exp(r_n_i + r_n_z * z_1),
    recr_size     = dnorm(z_2, mu_r, sd_r),
    data_list     = data_list,
    states        = list(c("z"))
)
```

BOX 1 (Continued)

```
# Step 4 in Figure 1. These next 3 functions define:
# 1. The numerical integration rules and how to iterate the
#   model (define_impl).
# 2. The range of values the the trait "z" can take on, and the number of
#   meshpoints to use when dividing the interval (define_domains).
# 3. The initial population state (define_pop_state).

example_proto_ipm <- define_impl(
    example_proto_ipm,
    list(
        P = list(int_rule = "midpoint", state_start = "z", state_end = "z"),
        F = list(int_rule = "midpoint", state_start = "z", state_end = "z")
    )
)

example_proto_ipm <- define_domains(
    example_proto_ipm,
    z = c(-2.65, 4.5, 250)  # format: c(L, U, m), m is number of meshpoints
)

example_proto_ipm <- define_pop_state(
    example_proto_ipm,
    n_z = rep(1/250, 250)
)

# Step 5 in Figure 1.

example_ipm <- make_ipm(example_proto_ipm)

# Step 6 in Figure 1.

lambda(example_ipm)
```

TABLE 1 Translations between mathematical notation, R's formula notation and ipmr's notation for the simplified version of Bogdan et al.'s *Carpobrotus* IPM. The ipmr column contains the expressions used in each kernel's definition. R expressions are not provided for sub-kernels and model iteration procedures because they typically require defining functions separately, and there are many ways to do this step (examples are in the R code for each case study in the appendix). The plogis() function computes the inverse logit transformation of an expression. s corresponds to survival, G corresponds to change in size conditional on survival, r_p is the probability of reproducing, r_n is the number of propagules produced by reproductive individuals and p_r is the probability that a propagule becomes a new recruit at $t + 1$

Math formula	R formula	ipmr
$\mu_G = \alpha_G + \beta_G * z$	size_2 ~ size_1, family =gaussian()	mu_G = G_int + G_slope * z
$G(z', z) = f_G(z', \mu_G, \sigma_G)$	G = dnorm(z_2, mu_G, sd_G)	G = dnorm(z_2, mu_G, sd_G)
$\text{logit}(s(z)) = \alpha_s + \beta_s * z$	surv ~size_1, family = binomial()	s = plogis(s_int + s_slope * z)
$\log(r_n(z)) = \alpha_{r_n} + \beta_{r_n} * z$	fec ~size_1, family = poisson()	r_n = exp(r_n_int + r_n_slope * z)
$\text{logit}(r_p(z)) = \alpha_{r_p} + \beta_{r_p} * z$	repr ~ size_1, family = binomial()	r_p = plogis(r_p_int + r_p_slope * z)
$r_d(z') = f_{r_d}(z', \mu_{r_d}, \sigma_{r_d})$	dnorm(z_2, mu_f_d, sigma_f_d)	r_d = dnorm(z_2, f_d_mu, f_d_sigma)
$p_r = \frac{\text{\# Recruits}(t+1)}{\text{\# flowers}(t)}$	p_r = n_new_recruits / n_flowers	p_r = n_new / n_flowers
$P = s(z) * G(z', z)$		P = s * G
$F(z', z) = r_p(z) * r_n(z) * r_d(z') * p_r$		F = r_p * r_n * r_d * p_r
$n(z', t+1) = \int_L^U [P(z', z) + F(z', z)]n(z, t)dz$		

TABLE 2 Translations between mathematical notation, R's formula notation and ipmr's notation for Ellner et al. (2016) *Ovis aries* IPM. The ipmr column contains the expressions used in each kernel's definition. R expressions are not provided for sub-kernels and model iteration procedures because they typically require defining functions separately, and there are many ways to do this step (examples are in the R code for each case study in the appendix). ipmr supports a suffix based syntax to avoid repetitively typing out the levels of discrete grouping variables. These are represented as 'a' in the Math column, 'age' in the R formula column, and are highlighted in bold in the ipmr column. *s* corresponds to survival, *G* corresponds to change in size conditional on survival, m_p is the probability of mating, r_p is the probability that a mating produces a new recruit at $t + 1$ and *B* is the size distribution of new recruits at $t + 1$ whose mean depends on parent size at time *t*. F_a is divided by 2 because this IPM only tracks females

Math formula	R formula	ipmr
$\text{Logit}(s(z, a)) = \alpha_s + \beta_{s,z} * z + \beta_{s,a} * a$	surv ~ size_1 + age, family = binomial()	s_age = plogis(s_int + s_z * z_1 + s_a * **age**)
$G(z', z, a) = f_G(z', \mu_G(z, a), \sigma_G)$	G = dnorm(size_2, mu_G_age, sigma_G)	G_age = dnorm(z_2, mu_G_age, sigma_G)
$\mu_G(z, a) = \alpha_G + \beta_{G,z} * z + \beta_{G,a} * a$	size_2 ~ size_1 + age, family = gaussian()	mu_G_age = G_int + G_z * z + G_a * **age**
$\text{Logit}(m_p(z, a)) = \alpha_{m_p} + \beta_{m_p,z} * z + \beta_{m_p,a} * a$	repr ~size_1 + age, family = binomial()	m_p_age = plogis(m_p_int + m_p_z * z + m_p_a * **age**)
$\text{Logit}(r_p(a)) = \alpha_{r_p} + \beta_{r_p,a} * a$	recr ~age, family = binomial()	r_p_age = plogis(r_p_int + r_p_a * **age**)
$B(z', z) = f_B(z', \mu_B(z), \sigma_B)$	b = dnorm(size_2, mu_rc_size, sigma_rc_size)	rc_size = dnorm(z_2, mu_rc_size, sigma_rc_size)
$\mu_B(z) = \alpha_B + \beta_{B,z} * z$	rc_size_2 ~ size_1, family = gaussian()	mu_rc_size = rc_size_int + rc_size_z * z
$P_a(z', z) = s(z, a) * G(z', z, a)$		P_age = s_age * g_age * d_z
$F_a(z', z) = s(z, a) * m_p(z, a) * r_p(a) * B(z', z) / 2$		F_age = s_age * f_p_age * r_p_age * rc_size / 2
$n_0(z', t+1) = \sum_{a=0}^{M+1} \int_L^U F_a(z', z) n_a(z, t) dz$		
$n_a(z', t+1) = \int_L^U P_{a-1}(z', z) n_{a-1}(z, t) dz$		
$n_{M+1}(z', t+1) = \int_L^U [P_{M+1}(z', z) n_{M+1}(z, t) + P_M(z', z) n_M(z, t)] dz$		

$$\mu_G = \alpha_G + \beta_G * z. \qquad (6)$$

Analytical solutions to the integral in Equation 1 are usually not possible (Ellner & Rees, 2006). However, numerical approximations of these integrals can be constructed using a numerical integration rule. A commonly used rule is the midpoint rule (more complicated and precise methods are possible and will be implemented, though are not yet, see Ellner et al., 2016, Chapter 6). The midpoint rule divides the domain $[L, U]$ into m artifical size bins centered at z_i with width $h = (U - L)/m$. The midpoints $z_i = L + (i - 0.5) * h$ for $i = 1, 2, \ldots, m$. The midpoint rule approximation for Equation 1 then becomes:

$$n(z_j, t+1) = h \sum_{i=1}^{m} K(z_j, z_i) n(z_i, t). \qquad (7)$$

In practice, the numerical approximation of the integral converts the continuous projection kernel into a (large) discretized matrix. A matrix multiplication of the discretized projection kernel and the discretized trait distribution then generates a new trait distribution, a process referred to as *model iteration* (sensu Easterling et al., 2000).

Equations 1 and 2 are an example of a *simple IPM*. A critical aspect of ipmr's functionality is the distinction between *simple IPMs* and *general IPMs*. A simple IPM incorporates a single continuous state variable. Equations 1 and 2 represent a simple IPM because there is only one continuous state, *z*, and no additional discrete

states. A general IPM models one or more continuous state variables, and/or discrete states. General IPMs are useful for modelling species with more complex life cycles. Many species' life cycles contain multiple life stages that are not readily described by a single state variable. Similarly, individuals with similar trait values may behave differently depending on environmental context. For example, Bruno et al. (2011) modelled aspergillosis impacts on sea fan coral *Gorgonia ventalina* population dynamics by creating a model where colonies were cross classified by tissue area (continuously distributed) and infection status (a discrete state with two levels—infected and uninfected). Coulson et al. (2010) constructed a model for Soay sheep where the population was structured by body weight (continuously distributed) and age (discrete state). Mixtures of multiple continuous and discrete states are also possible. Indeed, the vital rates of many species with complex life cycles are often best described with multivariate state distributions (Caswell & Salguero-Gómez, 2013). A complete definition of the simple/general distinction is given in Ellner et al. (2016, Chapter 6).

2.1 | A brief worked example of a simple IPM

Box 1 shows a brief example of how ipmr converts parameter estimates into an IPM. Perhaps the most frequently used metric derived from IPMs is the asymptotic per-capita population growth rate (λ, Caswell, 2001). When $\lambda > 1$, the population is growing,

while $\lambda < 1$ indicates population decline. ipmr makes deriving estimates of λ straightforward. Box 1 demonstrates how to parameterize a simple, deterministic IPM and estimate λ. The example uses a hypothetical species that can survive and grow, and reproduce sexually (but not asexually, so $C(z', z) = 0$ in Equation 2). The population is structured by size, denoted z and z', and there is no seedbank.

The $P(z', z)$ kernel is given by Equation 3, and the vital rates therein by Equations 4–6. The $F(z', z)$ kernel is given Equation 8:

$$F(z', z) = r_d(z') * r_n(z), \tag{8}$$

$$r_d(z') = f_{r_d}(z', \mu_{r_d}, \sigma_{r_d}), \tag{9}$$

$$\text{Log}(r_n(z)) = \alpha_{r_n} + \beta_{r_n} * z. \tag{10}$$

Equation 9 is a recruit size distribution (where f_{r_d} denotes a normal probability density function), and Equation 10 describes the number of new recruits produced by plants as a function of size z.

The code in Box 1 substitutes the actual probability density function (dnorm()) for f_G and f_{r_d}, and uses inverse link functions instead of link functions. Otherwise, the math and the code should look quite similar.

2.2 | Case study 1: A simple IPM

One use for IPMs is to evaluate potential performance and management of invasive species in their non-native range (e.g. Erickson et al., 2017). Calculating sensitivities and elasticities of λ to kernel perturbations can help identify conservation management strategies (Baxter et al., 2006; Caswell, 2001; de Kroon et al., 1986; Ellner et al., 2016). Bogdan et al. (2021) constructed a simple IPM for a *Carpobrotus* species growing north of Tel Aviv, Israel. The model includes four regressions, and an estimated recruit size distribution. Table 1 provides the mathematical formulae, the corresponding R model formulae and the ipmr notation for each one. The case study materials also offer an alternative implementation that uses the generic predict() function to generate the same output. The final part of the case study provides examples of functions that compute kernel sensitivity and elasticity, the per-generation growth rate, and generation time for the model, as well as how to visualize these results.

2.3 | Case study 2: A general age × size IPM

We use an age- and size-structured IPM from Ellner et al. (2016) to illustrate how to create general IPMs with ipmr. This case study demonstrates the suffix syntax for vital rate and kernel expressions, which is a key feature of ipmr (highlighted in bold in the 'ipmr' column in Table 2). The suffixes appended to each variable name in the ipmr formulation correspond to the subscript- and/or

superscript used in the mathematical formulation. ipmr internally expands the model expressions and substitutes the range of ages and/or grouping variables in for the suffixes. This allows users to specify their model in a way that closely mirrors its mathematical notation, and saves users from the potentially error-prone process of re-typing model definitions many times or using for loops over the range of discrete states. The case study then demonstrates how to compute age-specific survival and fertility from the model outputs.

3 | DISCUSSION OF ADDITIONAL APPLICATIONS

We have shown above how ipmr handles a variety of model implementations that go beyond the capabilities of existing scripts and packages. The underlying implementation based on metaprogramming should be able to readily incorporate future developments in parameterization methods. Regression modelling is a field that is constantly introducing new methods. As long as these new methods have functional forms for their expected value (or a function to compute them, such as predict()), ipmr should be able to implement IPMs using them.

Finally, one particularly useful aspect of the package is the proto_ipm data structure. The proto_ipm is the common data structure used to represent every model class in ipmr and provides a concise, standardized format for representing IPMs. Furthermore, the proto_ipm object is created without any raw data, only functional forms and parameters. We are in the process of creating the PADRINO IPM database using ipmr and proto_ipms as an 'engine' to re-build published IPMs using only functional forms and parameter estimates. This database could act as an IPM equivalent of the popular COMPADRE and COMADRE matrix population model databases (Salguero-Gómez et al., 2016; Salguero-Gómez et al., 2014). Recent work has highlighted the power of syntheses that harness many structured population models (Adler et al., 2014; Compagnoni et al., 2021; Salguero-Gómez et al., 2016). Despite the wide variety of models that are currently published in the IPM literature, ipmr's functional approach is able to reproduce nearly all of them without requiring any raw data at all.

CONFLICTS OF INTEREST
The authors declare no conflicts of interest.

AUTHORS' CONTRIBUTIONS
All authors contributed to package design. S.C.L. implemented the package. All authors wrote the first draft of the manuscript and contributed to revisions.

ORCID

Sam C. Levin https://orcid.org/0000-0002-3289-9925
Dylan Z. Childs https://orcid.org/0000-0002-0675-4933
Aldo Compagnoni https://orcid.org/0000-0001-8302-7492
Sanne Evers https://orcid.org/0000-0002-8002-1658
Tiffany M. Knight https://orcid.org/0000-0003-0318-1567
Roberto Salguero-Gómez https://orcid.org/0000-0002-6085-4433

REFERENCES

Adler, P. B., Ellner, S. P., & Levine, J. M. (2010). Coexistance of perennial plants: An embarassment of niches. *Ecology Letters*, *13*, 1019–1029. https://doi.org/10.1111/j.1461-0248.2010.01496.x

Adler, P. B., Salguero-Gómez, R., Compagnoni, A., Hsu, J. S., Ray-Mukherjee, J., Mbeau-Ache, C., & Franco, M. (2014). Functional traits explain variation in plant life history strategies. *Proceedings of the National Academy of Sciences of the United States of America*, *111*(2), 740–745. https://doi.org/10.1073/pnas.1315179111

Baxter, P. W. J., McCarthy, M. A., Possingham, H. P., Menkhorst, P. W., & McLean, N. (2006). Accounting for management costs in sensitivity analyses of matrix population models. *Conservation Biology*, *20*(3), 893–905. https://doi.org/10.1111/j.1523-1739.2006.00378.x

Bogdan, A., Levin, S. C., Salguero-Gómez, R., & Knight, T. M. (2021). Demographic analysis of Israeli *Carpobrotus* populations: Management strategies and future directions. *PLoS ONE*, *16*(4), e0250879. https://doi.org/10.1101/2020.12.08.415174

Bruno, J. F., Ellner, S. P., Vu, I., Kim, K., & Harvell, C. D. (2011). Impacts of aspergillosis on sea fan coral demography: Modeling a moving target. *Ecological Monographs*, *81*(1), 123–139. https://doi.org/10.1890/09-1178.1

Caswell, H. (2001). *Matrix population models: Construction, analysis, and interpretation* (2nd ed.). Sinauer Associates Inc.

Caswell, H., & Salguero-Gómez, R. (2013). Age, stage and senescence in plants. *Journal of Ecology*, *101*(3), 585–595. https://doi.org/10.1111/1365-2745.12088

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: Biological Sciences*, *271*(1547), 425–434. https://doi.org/10.1098/rpsb.2003.2597

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-Gomez, R., & Knight, T. M. (2021). 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

Coulson, T. N. (2012). Integral projection models, their construction and use in posing hypotheses in ecology. *Oikos*, *121*, 1337–1350. https://doi.org/10.1111/j.1600-0706.2012.00035.x

Coulson, T., Tuljapurkar, S., & Childs, D. Z. (2010). Using evolutionary demography to link life history theory, quantitative genetics and population ecology. *Journal of Animal Ecology*, *79*, 1226–1240. https://doi.org/10.1111/j.1365-2656.2010.01734.x

Crandall, R. M., & Knight, T. M. (2017). Role of multiple invasion mechanisms and their interaction in regulating the population dynamics of an exotic tree. *Journal of Applied Ecology*, *55*(2), 885–894. https://doi.org/10.1111/1365-2664.13020

de Kroon, H., Plaisier, A., van Goenendael, J., & Caswell, H. (1986). Elasticity: The relative contribution of demographic parameters to population growth rate. *Ecology*, *67*(5), 1427–1431.

Easterling, M. R., Ellner, S. P., & Dixon, P. M. (2000). Size specific sensitivity: Applying a new structured population model. *Ecology*, *81*(3), 694–708.

Ellner, S. P., Childs, D. Z., & Rees, M. (2016). *Data-driven modelling of structured populations: A practical guide to the integral projection model*. Springer International Publishing AG.

Ellner, S. P., & Rees, M. (2006). Integral projection models for species with complex demography. *The American Naturalist*, *167*(3), 410–428.

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

Ferrer-Cervantes, M. E., Mendez-Gonzalez, M. E., Quintana-Ascencio, P.-F., Dorantes, A., Dzib, G., & Duran, R. (2012). Population dynamics of the cactus *Mammillaria gaumeri*: An integral projection model approach. *Population Ecology*, *54*, 321–334. https://doi.org/10.1007/s10144-012-0308-7

Henry, L., & Wickham, H. (2020). *rlang: Functions for base types and core R and 'Tidyverse' features*. R package version 0.4.7. https://CRAN.R-project.org/package=rlang

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

Levin, S. C. (2021). Data from: Levisc8/ipmr: (Version v0.0.3). *Zenodo*, https://doi.org/10.5281/zenodo.5095062

Levin, S. C., Compagnoni, A. C., Childs, D. Z., Evers, S., Salguero-Gomez, R., & Knight, T. M. (2021). *ipmr: Fits Integral projection models using an expression based framework*. R package version 0.0.2. https://CRAN.R-project.org/package=ipmr

Merow, C., Dahlgren, J. P., Metcalf, C. J. E., Childs, D. Z., Evans, M. E. K., Jongejans, E., Record, S., Rees, M., Salguero-Gomez, R., & McMahon, S. M. (2014). Advancing population ecology with integral projection models: A practical guide. *Methods in Ecology and Evolution*, *5*, 99–110. https://doi.org/10.1111/2041-210X.12146S

Metcalf, C. J. E., Ellner, S. P., Childs, D. Z., Salguero-Gómez, R., Merow, C., McMahon, S. M., Jongejans, E., & Rees, M. (2015). Statistical modelling of annual variation for inference on stochastic population dynamics using Integral Projection Models. *Methods in Ecology and Evolution*, *6*(9), 1007–1017. https://doi.org/10.1111/2041-210X.12405

Metcalf, C. J. E., McMahon, S. M., Salguero-Gómez, R., & Jongejans, E. (2013). IPMpack: An R package for integral projection models. *Methods in Ecology and Evolution.*, *4*(2), 195–200. https://doi.org/10.1111/2041-210x.12001

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., Vieregg, D., Buckley,

Y. M., Che-Castaldo, J., ... Vaupel, J. W. (2016). COMADRE: A global database of animal demography. *Journal of Animal Ecology*, *85*, 371–384. https://doi.org/10.1111/1365-2656.12482

Rees, M., & Ellner, S. P. (2009). Integral projection models for populations in temporally varying environments. *Ecological Monographs*, *79*(4), 575–594. https://doi.org/10.1890/08-1474.1.

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., ... Vaupel, J. W. (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

Shefferson, R. P., Kurokawa, S., & Ehrlen, J. (2020). LEFKO3: Analysing individual history through size-classified matrix population models. *Methods in Ecology and Evolution*. https://doi.org/10.1111/2041-210X.13526

Williams, J. L., Miller, T. E. X., & Ellner, S. P. (2012). Avoiding unintentional eviction from integral projection models. *Ecology*, *93*(9), 2008–2014. https://doi.org/10.1890/11-2147.1

SUPPORTING INFORMATION

Additional supporting information may be found online in the Supporting Information section.

How to cite this article: Levin, S. C., Childs, D. Z., Compagnoni, A., Evers, S., Knight, T. M., & Salguero-Gómez, R. (2021). ipmr: Flexible implementation of Integral Projection Models in R. *Methods in Ecology and Evolution*, 12, 1826–1834. https://doi.org/10.1111/2041-210X.13683