Neural dynamic *N*-mixture model: a deep learning framework to infer demographic rates from count data

François Leroy [1], Marta A. Jarzyna [1,2]

[1] Department of Evolution, Ecology and Organismal Biology, The Ohio State University, Columbus, Ohio, 43210, USA

[2] Translational Data Analytics Institute, The Ohio State University, Columbus, Ohio, 43210, USA

Correspondence: François Leroy
Email address: francois.libert.leroy@gmail.com

Abstract

Changes in population abundance arise from underlying demographic processes, namely survival and recruitment, and knowledge of these two vital rates is crucial for better understanding biodiversity changes. Although demographic data based on individual encounter histories are often sparse in space and time, the dynamic N-mixture model provides an alternative by inferring survival and recruitment from repeated count data while accounting for imperfect detection. Here, we build on recent advances in neural hierarchical modelling and develop a neural dynamic N-mixture model. Specifically, we implement and optimize the likelihood of this model, including the memory-heavy transition matrix, as a negative log-likelihood loss function within a neural network framework, enabling inference of demographic parameters from count data using gradient-based optimization. We also implement a forward–backward algorithm to estimate latent abundance, survivors, and recruits. Using simulated data, we benchmark our neural implementation against a Bayesian implementation and show improved recovery of parameters and latent states, while substantially reducing computation time. Using computer-vision simulations, we show that the framework can be extended to image-based and other high-dimensional covariates through convolutional neural networks or other network architectures. We further apply the model to a case-study using the Swiss Green Woodpecker monitoring data. Our implementation provides a fast, flexible, and extensible framework for fitting dynamic N-mixture models with neural networks, opening opportunities to combine hierarchical ecological inference with data sources such as remote sensing, acoustic data, and other forms of ecological monitoring at large spatiotemporal scales.

1. Introduction

Over the Anthropocene, human activities have profoundly impacted ecosystems worldwide (IPBES, 2019), and a closely monitored indicator of this impact is temporal change in local population abundances, ΔN (Callaghan et al., 2024). Net changes in abundance over time (Fig. 1) are governed by the underlying demographic processes of recruitment R (*i.e.* number of new individuals entering the population through birth, maturation, or immigration) and survival S, which is the counterpart of the number of lost individuals L (removed from the population by death or emigration). While abundance has frequently declined across many species and ecosystems (Leroy et al., 2026; Rosenberg et al., 2019, Living Planet Index, LPI; < www.livingplanetindex.org/>, but see Leung et al., 2020; Toszogyova et al., 2024), the underlying processes of survival and recruitment remain poorly understood despite their potential to provide critical insights into how populations respond to environmental pressures and, in turn, to inform targeted and efficient conservation strategies.

Traditionally, inference of survival and recruitment has relied on capture–mark–recapture models (Sandercock, 2006), which use individual encounter histories to estimate demographic parameters (Cormack, 1964; Jolly, 1965; Lebreton et al., 1992; Pradel, 1996; Royle & Dorazio, 2008; Seber, 1965; Telenský et al., 2024). While powerful, these approaches require repeated captures of marked individuals, making them costly and often limited in spatial and temporal extent. On the other hand, count data are widely available across large spatial and temporal scales (e.g. Christmas Bird Count, National Audubon Society, 2020; Breeding Bird Survey, Ziolkowski Jr. et al., 2022), but carry no direct information on demographic processes. The dynamic N-mixture model (Dail & Madsen, 2011) addresses this by linking temporal changes in latent abundance to underlying survival and recruitment processes, formulated as a Hidden Markov Model with latent population states and imperfect detection (Zucchini et al., 2017). This framework has since been applied beyond conservation ecology (Kidwai et al., 2019; Priol et al., 2014), including in epidemiology (B. J. Brintz et al., 2023; DiRenzo et al., 2019; Parker et al., 2021). Despite its flexibility and broad applicability, Bayesian implementations of this model can be computationally demanding and slow to fit (Kéry & Royle, 2020).

In parallel with these developments, deep learning has increasingly been adopted in ecology for modeling complex, nonlinear relationships and integrating high-dimensional and even multimodal data (Christin et al., 2019; Desjardins-Proulx et al., 2019; Pollock et al., 2025), making it particularly attractive for utilizing large-scale ecological datasets that combine biodiversity monitoring, remote sensing, and climate data. However, most deep learning applications in ecology typically do not distinguish between underlying ecological processes and imperfect observations, nor do they explicitly model latent population states, demographic dynamics, or detection processes (Christin et al., 2019; Pichler & Hartig, 2023). A notable exception is Joseph (2020), who brilliantly introduced a framework combining the inferential power of hierarchical models with the predictive power and flexibility of neural networks. The core idea is to embed the likelihood of a hierarchical model within a neural network training

objective by expressing it as a negative log-likelihood loss function. While Joseph (2020) provided implementation for several widely used hierarchical models, including the N-mixture model (Royle, 2004), the static (MacKenzie et al., 2002) and dynamic (MacKenzie et al., 2003) occupancy models, capture-recapture model (Langrock & King, 2013), and a movement model using computer vision (Joseph, 2020), the dynamic N-mixture model (Dail & Madsen, 2011) was not included in his initial implementation. Given the importance of this model for ecology and related fields, we see an opportunity to provide the scientific community with a ready-to-use implementation within a deep learning framework.

Here, we introduce the neural implementation of the dynamic N-mixture model. By embedding the dynamic N-mixture model likelihood into a neural network loss function, our approach enables the parametric estimation of abundance, survival, recruitment, and detection as flexible functions of covariates, while retaining the hierarchical structure of the original model. We further extend this framework by providing estimates of latent demographic quantities, including abundance (N), survival (S), and recruitment (R), using a forward–backward algorithm (Rabiner, 1989; Zucchini et al., 2017) adapted to neural network outputs. Because the likelihood is embedded within a neural network framework, the same implementation can also be coupled with architectures designed for high-dimensional inputs, such as convolutional neural networks, enabling demographic inference from image-based covariates.

The main methodological challenges of our implementation arise from the optimization of the loss function, particularly due to the high-dimensional transition matrices involved. We address this challenge by implementing a fully vectorized likelihood that exploits parallel computation on modern GPU, enabling efficient model fitting. Together, these developments provide a fast neural hierarchical implementation for estimating demographic processes from count data, which we validate using simulations, benchmark against a Bayesian implementation, and illustrate with both an empirical case study and image-based simulations.

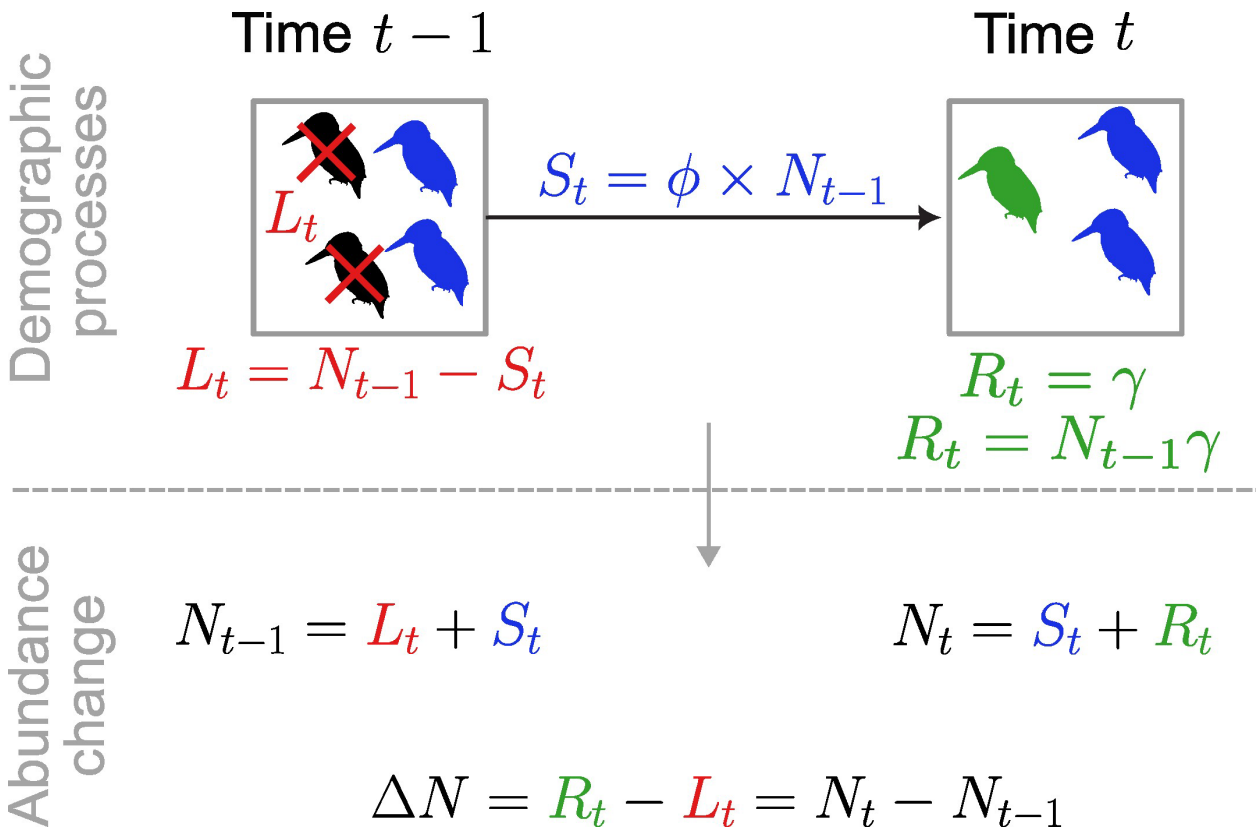

Fig. 1 | Change in abundance (ΔN) arises from three demographic components: the number of survivors (S_t, blue), losses (L_t, red), and recruits (R_t, green). Survival is governed by the survival probability ϕ, of which the loss is the counterpart, while recruitment follows a Poisson process with rate γ (constant recruitment) or $N_{t-1}\gamma$ (per-capita recruitment).

2. Material and Methods

2.1 The Dynamic *N*-mixture model

Inference of population abundance (N) from count data is commonly performed using hierarchical models that separate the underlying ecological state from the observation process (eq. 1), accounting for imperfect detection (Gelman & Hill, 2021; Kéry & Royle, 2015, 2020; Royle, 2004; Royle & Dorazio, 2008). The N-mixture model (Royle, 2004) combines repeated counts with a latent abundance process, typically modeled as a Poisson (or negative binomial) distribution. A key limitation of this model, however, is the assumption of population closure during repeated surveys, implying that abundance remains constant over the sampling period.

The dynamic N-mixture model (Dail & Madsen, 2011) relaxes this assumption by allowing abundance to vary over time through survival and recruitment processes. Specifically, population size is modeled as a function of the number of surviving individuals (eq.3) and newly recruited individuals (eq. 4) between time steps, providing a direct link between temporal changes in abundance and underlying demographic processes (eq. 5).

We consider $i = 1, \ldots, E$ sampling locations (e.g. sites), and $t = 1, \ldots, T$ primary sampling occasions (e.g. years). The observation model (eq. 1) is shared between the static (Royle, 2004) and dynamic (Dail & Madsen, 2011) formulations and assumes:

$$n_{it} \sim \text{Bin}(N_{it}, p), \qquad (1)$$

where N_{it} is the latent abundance and p the detection probability (assumed to be constant here for simplicity). The corresponding likelihood is:

$$L(\{N_{it}\}, p \mid \{n_{it}\}) = \prod_{i=1}^{E} \left\{ \prod_{t=1}^{T} \text{Bin}(n_{it}; N_{it}, p) \right\} \qquad (2)$$

The state process describes temporal dynamics of abundance. Survival is modeled as:

$$S_{it+1} \sim \text{Bin}(N_{it}, \phi), \quad (3)$$

where ϕ is the survival probability. Recruitment is modeled as either absolute or density-dependent:

$$R_{it+1} \sim \text{Poisson}(\gamma) \qquad \text{or} \qquad R_{it+1} \sim \text{Poisson}(\gamma \text{N}_{\text{it}}), \qquad (4)$$

where γ is a constant or per-capita recruitment rate, respectively, and total abundance N evolves as:

$$N_{it+1} = S_{it+1} + R_{it+1} \qquad (5)$$

Initial abundance is specified as:

$$N_{i1} \sim \text{Poisson}(\lambda) \qquad (6)$$

Following Dail & Madsen (2011), marginalizing over latent states yields the integrated likelihood:

$$\begin{aligned} L(p, \lambda, \gamma, \omega \mid \{n_{it}\}) \\ = \prod_{i=1}^{E} \left[\sum_{N_{i1}=n_{i1}}^{\infty} \cdots \sum_{N_{iT}=n_{iT}}^{\infty} \left\{ \left(\prod_{t=1}^{T} \text{Bin}(n_{it}; N_{it}, p) \right) \right. \right. \\ \left. \left. \times \frac{e^{-\lambda} \lambda^{N_{i1}}}{N_{i1}!} \cdot \prod_{t=2}^{T} P_{N_{it-1}, N_{it}} \right\} \right] \end{aligned} \qquad (7)$$

Importantly, the transition matrix between $N_{it-1} = j$ and $N_{it} = k$, noted $P_{N_{it-1}, N_{it}}$ in eq. 7, is given by:

$$P_{jk} = \sum_{c=0}^{\min(j,k)} \text{Bin}(c; j, \phi) \cdot \text{Pois}(k - c; \gamma) \quad (8)$$

2.2 Neural implementation of the dynamic *N*-mixture model

A simplified representation of the dynamic N-mixture model within a deep learning framework is shown in Fig. 2. The neural network maps input covariates to the four model parameters: λ (initial abundance), γ (recruitment), ϕ (survival probability), and p (detection probability). To enforce parameter constraints, we apply appropriate activation functions at the output layer, namely a logistic function for ϕ and p, and an exponential function for λ and γ (Fig. 2).

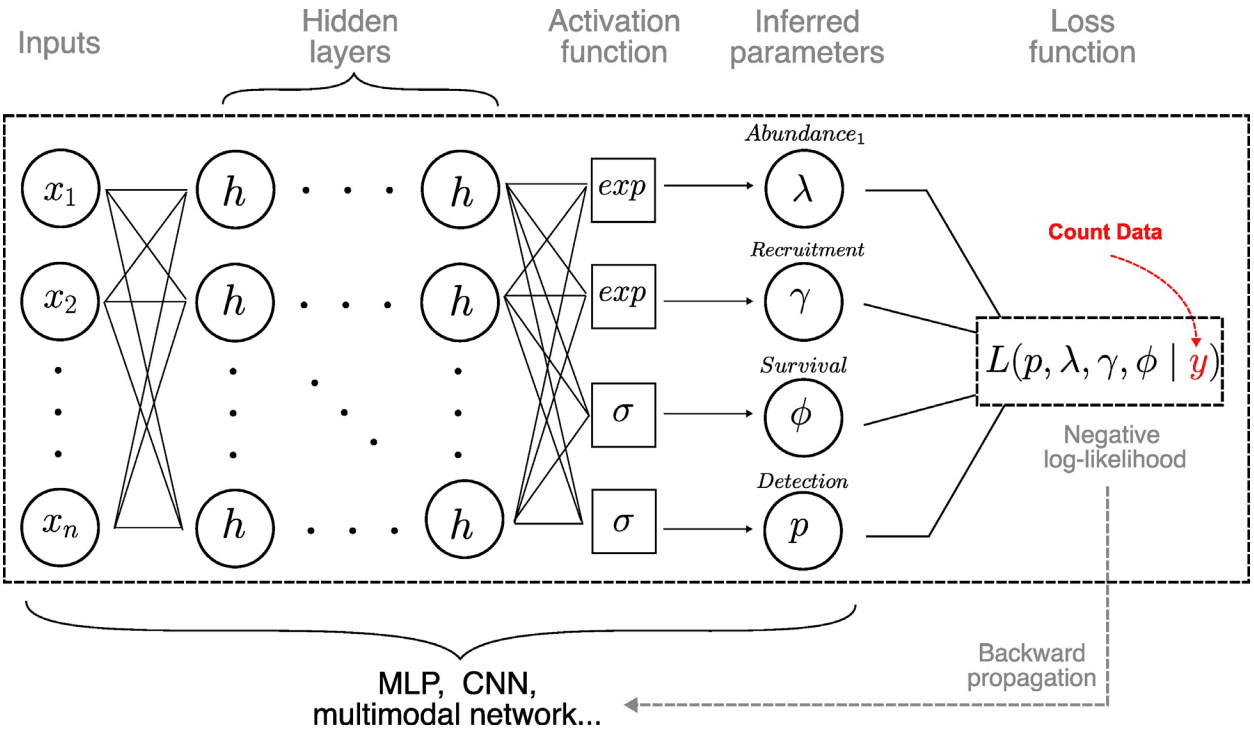

Fig. 2 | Simplified graph of the neural dynamic N-mixture model. Input covariates $(x_1, \ldots, x_n)$ are passed through a neural network to infer the model parameters λ (initial abundance), γ (recruitment), ϕ (survival), and p (detection probability). Here, h represents hidden units within the network. Outputs are transformed using appropriate link functions to enforce parameter constraints: an exponential function for λ and γ, and a logistic function for ϕ and p. These parameters are then used to compute the negative log-likelihood of the dynamic N-mixture model, obtained by marginalizing over latent abundance states, which serves as the loss function for training via backpropagation.

These parameter estimates are used to compute a negative log-likelihood derived from the dynamic N-mixture model (eq. 7), which serves as the loss function optimized during training. In

contrast to Bayesian implementations that sample latent abundances N_{it}, our approach marginalizes the likelihood over all possible values of N_{it}. In practice, this is achieved by truncating the state space at a sufficiently large value $N_{max} \gg \max\{n_{i,t}\}$, replacing the infinite sums in eq. 7.

Direct evaluation of eq. 7 is complicated to compute in practice due to nested summations over latent states. To address this, we adopt the recursive formulation of the likelihood described in Dail & Madsen (2011, Appendix B). Specifically, we first define:

$$g_1(N_{it}) = \binom{N_{it}}{n_{it}} p^{n_{it}}(1-p)^{N_{it}-n_{it}} \qquad (9)$$

$$g_2(N_{i1}) = \frac{e^{-\lambda}\lambda^{N_{i1}}}{N_{i1}!} \qquad (10)$$

$$g_3(N_{it} \mid N_{it-1}) = P_{N_{it-1},N_{it}} \qquad (11)$$

With $P_{N_{it-1},N_{it}}$ defined in eq. 8.

Using this notation, the likelihood can be expressed recursively. Starting from the final time step T, we compute:

$$L(p_t, \lambda_i, \gamma, \omega \mid \{n_{it}\}) = \prod_{i=1}^{E}\left[\sum_{N_{i1}=n_{i1}}^{\infty} \cdots \sum_{N_{iT}=n_{iT}}^{\infty} \left\{\left(\prod_{t=1}^{T} g_1(N_{it})\right) \cdot g_2(N_{i1}) \cdot \prod_{t=2}^{T} g_3(N_{it} \mid N_{it-1})\right\}\right] \qquad (12)$$

Then, starting from the end of the time series, T, we first compute:

$$g^*(N_{iT-1}) = \sum_{N_{iT=0}}^{\infty} g_1(N_{iT}) g_3(N_{iT} \mid N_{iT-1}) \qquad (13)$$

And then we recursively compute:

$$g^*(N_{it-1}) = \sum_{N_{it=0}}^{\infty} g_1(N_{it}) g_3(N_{it} \mid N_{it-1}) g^*(N_{it}), \text{ for } 2 \le t \le T-1 \qquad (14)$$

This yields a simplified final likelihood:

$$L(p, \lambda, \gamma, \omega \mid \{n_{it}\}) = \prod_{i=1}^{E} \{ \sum_{N_{i1}=n_{i1}}^{\infty} g_1(N_{i1}) g_2(N_{i1}) g^*(N_{i1}) \} \qquad (15)$$

Despite this recursive formulation (eq. 15), the main computational bottleneck lies in the transition matrix P (eq. 8), which has dimension $N_{\max} \times N_{\max}$ and must be evaluated for each site and time step.

To address this, we implement a fully vectorized computation of the transition matrix (eq. 8) across sites, time steps, and latent states (https://anonymous.4open.science/r/neural_DNMM-E86B/). This allows all transition probabilities to be computed efficiently using tensor operations, avoiding explicit nested loops and significantly accelerating likelihood evaluation, albeit at the cost of increased memory usage.

Finally, missing observations were accommodated by masking the observation likelihood by setting them to zero, so they did not contribute to the negative log-likelihood loss.

2.3 Estimating N, S, and R

Building on the implementation of the negative log-likelihood (eq. 7) and transition matrix (eq. 8), we derive estimates of latent states, namely abundance (N), number of survivors (S), and number of recruits (R), by implementing the forward-backward algorithm (Rabiner, 1989; Zucchini et al., 2017). This algorithm is commonly used for Hidden Markov Models and, in our case, provides point estimates of latent states $P(N_{it} \mid y_{i,1:T})$.

The forward pass computes the distribution $P(N_{it} \mid y_{i,1:t})$, while the backward pass incorporates future observations to obtain smoothed posterior distribution $P(N_{it} \mid y_{i,1:T})$. We estimate abundance N_{it} as the posterior mean over all possible states.

Estimates of survival (S) and recruitment (R) are derived from the joint posterior distribution of N_t and N_{t+1} described above. For each possible transition, we derive the conditional distribution of S, and then derive R following eq. 5.

Although this procedure is computationally intensive due to the use of the transition matrix (eq. 8), we implement a batched version of the forward–backward algorithm over sites, enabling efficient computation even on memory-constrained GPUs.

2.4 Benchmarking the neural implementation against simulated data

2.4.1 Simulation with tabular covariates

To evaluate our deep learning implementation, we performed a simulation study. We simulated a single standardized covariate x, and used quadratic relationships between x and the four model parameters λ, ϕ, γ, and p. Using these parameter values, we then simulated latent abundance

(N), number of survivors (S), number of recruits (R), and observed repeated counts, allowing assessment of parameter recovery and latent-state estimation.

For the neural network implementation, we used a feed-forward architecture with a single hidden layer of 64 units and a sigmoid activation function. The same covariate x was used to infer all four parameters. The model was trained using our implementation of the negative log-likelihood (eq. 7). We then estimated values of N, S, and R using our version of the forward-backward algorithm. Because the data-generating process was known and followed the parameter assumptions of the dynamic N-mixture model, the objective of the simulation was to assess parameter recovery rather than predictive performance. Therefore, no validation set was used, and model performance was assessed by comparing inferred and simulated parameters and latent states. We used 300 epochs, with a learning rate of 0.01, a weight decay of 10^{-6}, and a batch size of 4.

For comparison with the conventional approach to fitting these models, we also fitted the same simulated data in a Bayesian framework using JAGS (Plummer, 2003). Quadratic relationships between x and each parameter were specified via linear predictors, with weakly informative Normal priors on regression coefficients. To reach full convergence (Kéry & Royle, 2020), we used 100,000 iterations across 3 chains, with an adaptation phase of 1,000 iterations, a burn-in of 75,000, and thinning of 100.

2.4.2 Simulation with image-based covariates

We also performed a computer-vision simulation. We generated simple 16×16 grayscale images representing a smooth visual gradient, ranging from bright values at the top and dark values at the bottom at one extreme, to the opposite pattern at the other extreme, with an intermediate bright central band (Fig. 4). The four model parameters λ, ϕ, γ, and p were then simulated to vary quadratically along this visual gradient. This allowed us to test whether a convolutional neural network could recover the underlying parameter relationships from image-based covariates. The CNN consisted of two convolutional layers with sigmoid activations and average pooling, followed by a fully connected layer and the same output transformations used above. The model was trained using only the simulated images as input, for 300 epochs, with a learning rate of 10^{-3}, weight decay of 10^{-6}, and a batch size of 32. Because computer vision pipelines are inherently tied to deep learning frameworks, no comparison to a Bayesian model implementation was made for this component.

Finally, we computed the Root Mean Squared Error (RMSE) between simulated and inferred values of N, S, and R.

The benchmarking analyses were conducted on the Ohio Supercomputer Center (https://www.osc.edu/). Neural network models were trained on an NVIDIA A100 (40GB) GPU using CUDA 12.8, PyTorch 2.8.0, and Python 3.9.21, while Bayesian analyses were run in R 4.4.0 with JAGS 4.3.2 on CPUs (~2.3 GHz).

2.5 Case study: the Swiss Green Woodpecker data

In addition to the simulation study, we tested our Neural dynamic N-mixture model on the Swiss breeding bird survey (Monitoring Häufige Brutvögel, Strebel et al., 2025) for the Eurasian Green Woodpecker (*Picus viridis*). The dataset comprises 267 sites, monitored from 2004 to 2017, with 2 or 3 secondary visits per site and year and has been previously used in Kéry & Royle (2020), providing a useful benchmark for evaluating model behavior.

We used forest cover and elevation as predictor for λ, Julian date and survey intensity as predictors for the detection probability p, and no covariates for recruitment γ and survival probability ϕ.

The neural network was implemented using separate parameter-specific heads: one for initial abundance λ, one shared head for survival ϕ and recruitment γ, and one for detection probability p. The λ and p heads each consisted of a single hidden fully connected layer with 64 units and sigmoid activation functions. In contrast, ϕ and γ were modeled as a linear layer to mimic the Bayesian implementation, yielding parameters shared across sites and time.

Unlike the simulation study, model tuning was required to avoid overfitting and ensure stable inference. We used weight decay (5×10^{-5}) and early stopping (Krogh & Hertz, 1991; Prechelt, 2012) based on a validation set (20% of the data), with a minimum improvement threshold of 10^{-4} and a patience of 15 epochs (Fig. S2). The model was trained using a learning rate of 5×10^{-4}.

For comparison, we fitted the same data in a Bayesian framework. Following Kéry & Royle (2020), quadratic relationships were specified between elevation and λ, and between $p$and both Julian date and survey intensity. To ensure convergence, we based the initial values and MCMC settings on Kéry & Royle (2020).

To compare both implementations, we generated conditional effect plots, varying each covariate across its observed range while holding others at their mean values.

The case study analyses were run locally. The neural network was trained on an NVIDIA RTX 3000 Ada (8GB) GPU using CUDA 12.6, PyTorch 2.8.0, and Python 3.12.11, while the Bayesian model was fitted in R 4.4.2 with JAGS 4.3.1 on CPU (~2.3 GHz). All neural network models were trained using the Adam optimizer (Kingma & Ba, 2014).

2.6 Code availability

All code and analyses required to reproduce the results are available at: https://anonymous.4open.science/r/neural_DNMM-E86B/.

3. Results

3.1 Method validation and benchmark against Bayesian framework

Both Bayesian and neural network approaches recovered the relationships between model parameters and the covariate x (Fig. 3a–d). Visual comparison suggests that the neural network generally provided a closer match to the simulated relationships for λ (abundance), γ (recruitment), and ϕ (survival probability), while both approaches performed similarly for detection probability p.

These patterns were consistent with the quantitative evaluation of latent-state estimation (Fig. 3e–j). Across the 11 simulated datasets, the neural network yielded lower mean RMSE for abundance N, number of survivors S, and number of recruits R compared to the Bayesian framework. We attribute these differences to an overestimation of R and S by the Bayesian approach, which propagates into a corresponding overestimation of N (Fig. S1).

The convolutional neural network also recovered the parameter relationships when using simulated images as input covariates (Fig. 4a–d). This computer-vision simulation produced low RMSE values for N, S, and R (Fig. 4e), suggesting that the framework can be extended to image-based covariates when the image features contain information about the underlying ecological parameters.

Importantly, our neural network implementation required substantially less computation time compared to the Bayesian implementation. Across the 11 simulated datasets, average runtime was approximately 15 minutes per model using a GPU (300 epochs), compared to approximately 2 hours for each Bayesian model (100,000 MCMC iterations) on the Ohio Supercomputer Center. This translates to a roughly eight-fold reduction in computation time.

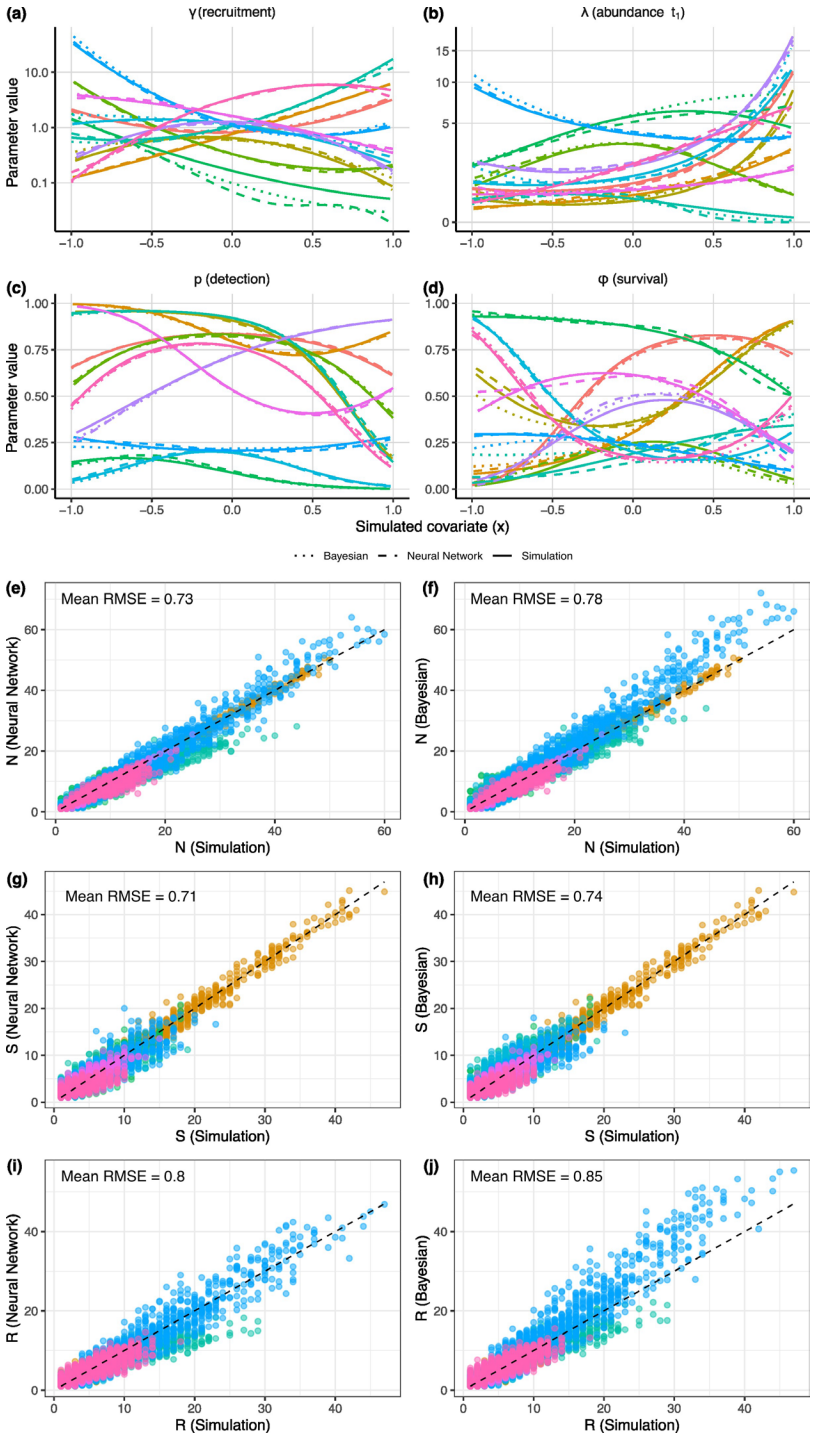

Fig. 3 | (a–d) Simulated (solid lines) and inferred relationships from the Bayesian (dotted lines) and neural network (dashed lines) frameworks between a standardized covariate x and the four parameters of interest: (a) γ (recruitment), (b) λ (abundance at time t_1), (c) p (detection probability), and (d) ϕ (survival probability). For visual separation, y-axis in (a) is log transformed and in (b) is square root transformed. (e–j). Estimated versus simulated values of abundance N (e–f), number of survivors S (g–h), and number of recruits R (i–j), for the neural network (left column: e, g, i) and Bayesian (right column: f, h, j) frameworks. Colors indicate different simulation scenarios ($n = 11$). Mean RMSE across simulations is reported in each panel.

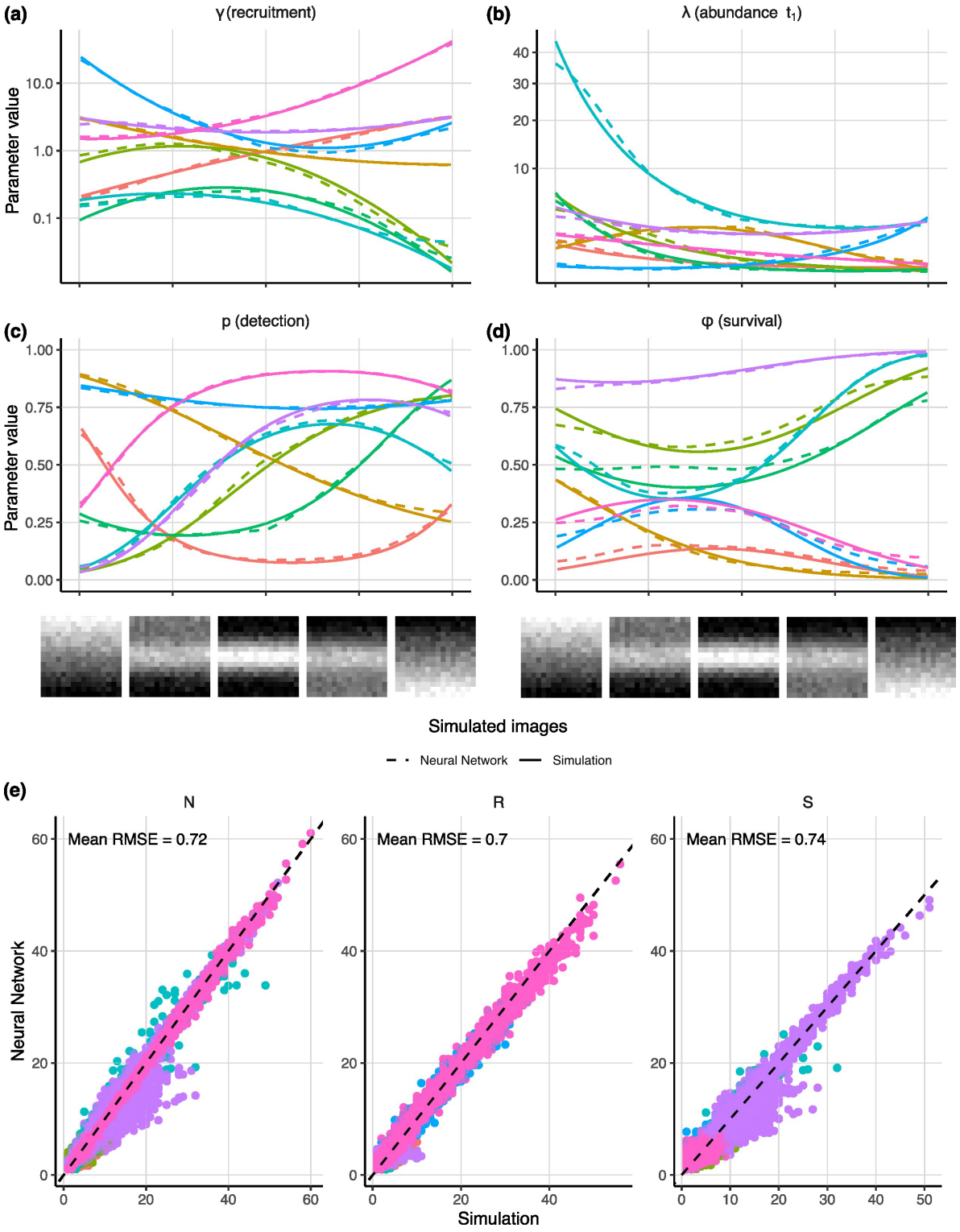

Fig. 4 | (a–d) Simulated (solid lines) and inferred relationships from the convolutional neural network (dashed lines) between the simulated image gradient (x-axis) and the four parameters of interest: (a) γ (recruitment), (b) λ (abundance at time t_1), (c) p (detection probability), and (d) ϕ (survival probability). For visual separation, y-axis in (a) is log transformed and in (b) is square root transformed. (e) Estimated versus simulated values of abundance N, number of survivors S, and number of recruits R.

3.2 Case study: inference of survival and recruitment for the Swiss Green Woodpecker

The inferred relationships between covariates and model parameters were broadly consistent across frameworks for abundance and elevation (decreasing effect), abundance and forest cover (increasing effect), and detection probability and Julian date (decreasing effect; Fig. 5a). However, the relationship between detection probability and survey intensity differed substantially. While the Bayesian model inferred a hump-shaped relationship with detection probability approaching zero at high survey intensity, the neural network inferred an increase followed by a plateau.

Estimates of survival probability ϕ were lower for the neural network (0.74) than for the Bayesian model (0.82, 95% credible interval $[0.80; 0.85]$), whereas recruitment γ was higher in the neural network (0.45) compared to the Bayesian model (0.38, 95% credible interval $[0.34; 0.42]$). These differences are reflected in the latent-state estimates (Fig. 5b), with the Bayesian framework producing higher numbers of survivors S and lower numbers of recruits R, resulting in an overall higher estimate of abundance N.

Using early stopping, training of the neural network required approximately 15 minutes over 206 epochs (Fig. S2). In contrast, fitting the Bayesian model required slightly more than 1 hour for 70,000 MCMC iterations. This corresponds to a roughly four-fold reduction in computation time.

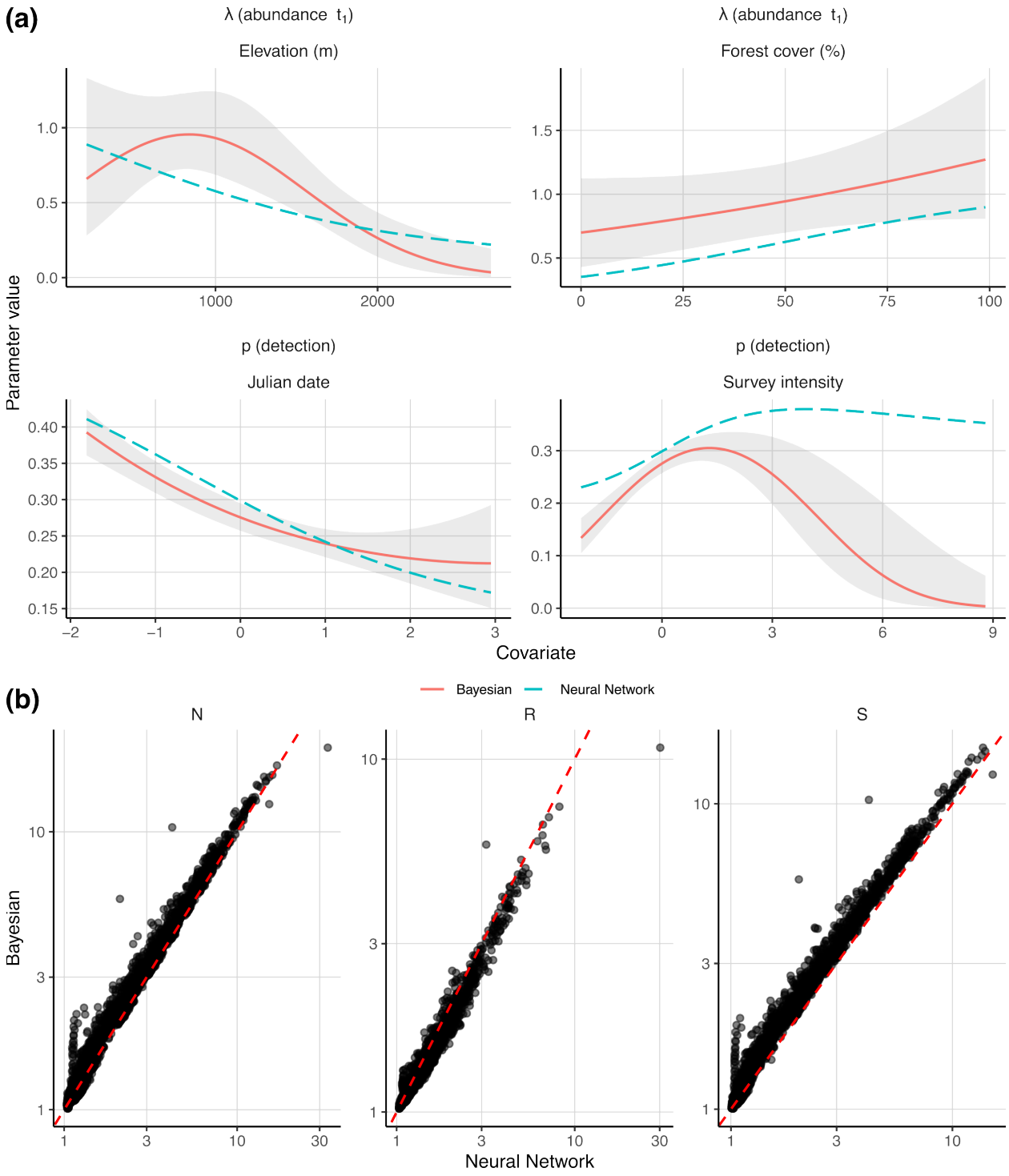

Fig. 5 | (a) Conditional effect plots showing the inferred relationships between λ (abundance at time 1; top row) and detection probability p (bottom row) and their respective covariates, with all other covariates held at their mean values. Solid lines represent the Bayesian estimates, dashed lines represent the neural network estimates, and shaded areas indicate the 95% credible intervals from the Bayesian framework. (b) Estimated values of abundance N, number of recruits R, and number of survivors S from the neural network versus the Bayesian framework. The dashed line indicates the 1:1 relationship. Axes are log-transformed.

4 Discussion

Here, we developed a neural network implementation of the dynamic N-mixture model, providing a computationally fast and flexible alternative to traditional Bayesian approaches, with improved accuracy in parameter recovery and latent-state inference. In addition, our computer-vision simulation illustrates how the same framework can be extended to high-dimensional inputs, such as images, while retaining the hierarchical structure of the original model.

4.1 Inferential performance

Compared to the known simulated truth, the neural network implementation performed better in both parameter recovery and latent-state inference (Fig. 3). Admittedly, this result was somewhat unexpected, as the quadratic relationships used to generate the data were explicitly specified in the Bayesian model, potentially giving it an advantage over the neural network, which had to learn these relationships from the data during training. Furthermore, the neural network architecture used in both the simulation study and case study was intentionally simple, consisting of a single hidden layer with 64 units, suggesting that the results reported here likely represent a conservative lower bound on performance, as the architecture and hyperparameters could be further optimized.

The superior performance of the neural network can also be seen in the simulation analysis and the estimation of survival S, and recruitment R (and thus abundance N). On average, R was clearly overestimated in the Bayesian setting compared to the neural network estimates, while S was also overestimated, though to a lesser extent (Fig. 3e-j, Fig. S1). While the overestimation of R is surprising, the overestimation of S is a known problem of the dynamic N-mixture model (Bellier et al., 2016, 2018). A similar pattern is visible in the case study, where the estimates of ϕ and S are higher for the Bayesian than for neural network framework (Fig. 5b). Although both frameworks use the same likelihood, it is plausible that the added flexibility of the neural network allows it to mitigate the overestimation of S.

The computer-vision simulation further illustrates the flexibility of the framework. When using simulated images as covariates, the convolutional neural network recovered the underlying parameter relationships and produced low RMSE values for N, S, and R (Fig. 4). This suggests that the neural dynamic N-mixture framework is not restricted to tabular covariates, but can also learn from structured, high-dimensional inputs when these contain information about the underlying ecological parameters.

The Swiss Green Woodpecker case study further highlights differences between the two implementations in the inferred covariate relationships. While the inferred relationships for elevation, forest cover, and Julian date are similar in direction across both frameworks, differences in shape are apparent (Fig. 5). In particular, the quadratic relationship for elevation specified in the Bayesian model is not recovered by the neural network, which instead infers a near-linear declining response. This may indicate that the implemented quadratic form is not

fully supported by the data, and that it represents a restrictive parametric assumption (Kéry & Royle, 2020).

The most important difference between the two implementations is observed for survey intensity (Fig. 5a). The Bayesian model infers a hump-shaped relationship, with detection probability decreasing at high values of survey intensity, whereas the neural network infers an increasing relationship that reaches a plateau. Given that detection probability is generally expected to increase with survey effort (Guillera-Arroita et al., 2010), the neural network result likely represents a more plausible relationship.

One possible explanation for this discrepancy lies in the use of regularization during neural network training. The use of weight decay, validation-based early stopping, and data splitting favors smoother and more stable relationships, limiting the risk of fitting noise-induced curvature. In contrast, the Bayesian model relies on a fixed quadratic specification without explicitly penalizing unnecessary curvature. In settings where the relationship between a parameter and a covariate is weakly identifiable, this constraint may lead to the inference of spurious curvature, resulting in an analogous effect to overfitting that we mitigate in our neural network training.

4.2 Computational performance

For both the simulation study and the case study, the computing time of the neural model was substantially faster than the Bayesian implementation. For the Swiss Green Woodpecker dataset, training of the neural network required approximately 15 minutes using 206 training epochs on a local 8 GB GPU, while we report a fitting time of more than an hour, and Kéry & Royle (2020) report runtimes of approximately 83 minutes for a comparable Bayesian model. Similarly, for the simulated datasets, the neural network required ca. 15 minutes for 300 training epochs, compared to an average of approximately 2 hours for the Bayesian implementation.

This difference arises from the computational characteristics of the two approaches. Neural networks rely on gradient-based optimization, which can be efficiently parallelized on modern hardware (LeCun et al., 2015), while MCMC-based Bayesian inference requires iterative sampling. Although the runtime of the Bayesian implementation depends on the choice of MCMC hyperparameters, the settings used here were sufficient to ensure adequate convergence.

Notably, the computational cost of the neural network could be further reduced. In our implementation, relatively small batch sizes (4 for simulations and 16 for the case study) and a high number of training epochs (300 for simulations) were used, both of which increase runtime. Alternative choices of hyperparameters could further improve computational efficiency.

Our results confirm the potential of neural network implementations to substantially reduce computation time in ecological models (Abdelwahed et al., 2026; LeCun et al., 2015; Pichler & Hartig, 2021; Wilkinson et al., 2019), which may facilitate the analysis of larger datasets or more complex model structures that are often computationally prohibitive in Bayesian frameworks.

4.3 Next steps, limits and considerations

Several extensions could broaden the applicability of our neural dynamic N-mixture implementation. A logical next step would be to implement the existing advances of the original dynamic N-mixture model (Bellier et al., 2016; B. Brintz et al., 2018; DiRenzo et al., 2019; Hostetler & Chandler, 2015; Rossman et al., 2016; Zhao et al., 2022; Zhao & Royle, 2019; Zipkin et al., 2014, 2017). Beyond single-species applications, our implementation could also provide a basis for multi-species extensions. Inspired by neural hierarchical models and joint species distribution models, shared hidden layers could act as latent factors representing common environmental or spatial gradients, while species-specific embeddings could allow demographic responses to vary among species (Joseph, 2020). Such a framework could jointly estimate species-specific survival, recruitment, abundance, and detection parameters from repeated count data, while borrowing information across species and accounting for imperfect detection.

Also, an important avenue for future work lies in leveraging the ability of neural networks to integrate complex, multimodal, and high-dimensional data sources (Pollock et al., 2025). Our computer-vision simulation provides a proof of concept for this direction, showing that image-based covariates can be incorporated directly into the neural dynamic N-mixture framework. In ecological applications, this could allow predictors derived from satellite imagery, acoustic data, camera traps, or other remote-sensing products to inform survival and recruitment while retaining an explicit observation model for count data. For example, spatially explicit image-derived habitat features could be combined with large-scale monitoring programs such as the Breeding Bird Survey (Ziolkowski Jr. et al., 2022) or Christmas Bird Count (National Audubon Society, 2020) to improve inference on demographic processes across broad spatial scales.

The main but known limitation (see B. Brintz et al., 2018) of our neural implementation comes from the marginalization of the likelihood. Depending on N_{max}, the construction of transition matrices across sites and time steps can result in substantial memory requirements, particularly when using large batch sizes. In our implementation, this issue can be mitigated by reducing the batch size, lowering N_{max}, or using hardware with larger GPU memory. While this was not problematic for the ecological datasets considered here, which involved relatively low counts, it may become limiting for datasets with larger counts (e.g. in epidemiological datasets, as in B. Brintz et al., 2018). One solution could be to implement banded or sparse approximations of the transition matrix, reducing memory usage while retaining computational efficiency, or to use the asymptotic approximation proposed by B. Brintz et al. (2018). Thus, although the proposed neural network implementation benefits from parallelization, its scalability remains constrained by memory requirements, and further methodological developments are needed to extend its applicability to large-count systems.

The computational advantage of the neural implementation is most evident when using GPU accelerations. When run on CPUs, the gain relative to Bayesian implementations is considerably reduced, which may limit accessibility in some settings. However, GPU computing is becoming

increasingly available through institutional clusters, cloud-based platforms, and consumer-level hardware, suggesting that this limitation may become less restrictive over time.

Finally, from an inferential perspective, both frameworks present somewhat different challenges. While Bayesian implementations require careful tuning of initial values and MCMC settings to ensure convergence, neural network approaches instead require attention to hyperparameters, like epochs, batch size, and regularization strategies, including the choice of learning rate, weight decay, early stopping, and validation data splitting. These choices can influence both model stability and inference and thus require careful tuning.

5. Conclusion

We developed a neural implementation of the dynamic N-mixture model, providing a fast and flexible alternative to traditional Bayesian inference for estimating survival, recruitment, abundance, and detection from repeated count data. By reformulating the original likelihood as a negative log-likelihood loss function, our framework enables gradient-based optimization while retaining the hierarchical structure of the original model. We further implemented a forward–backward algorithm to recover latent abundance, survivors, and recruits from the fitted neural model. In benchmarking simulations, our implementation improved parameter and latent-state recovery while substantially reducing computation time compared to a Bayesian implementation. The Swiss Green Woodpecker case study and the computer-vision simulation further illustrate the flexibility of the framework, including its potential to incorporate high-dimensional, multimodal covariates such as remote sensing imagery, acoustic data, or other ecological monitoring products. By providing an open implementation, our work offers a scalable foundation for extending dynamic N-mixture models to larger datasets, more flexible covariate structures, and broad ecological data sources.

Acknowledgements

We thank Maxwell Joseph for helpful discussions, resource sharing, and feedback on our implementation. We also thank Tanya Berger-Wolf, Laura Pollock, Justin Kitzes, David Rolnick, Sara Beery, Kaitlyn Gaynor, and Graham Taylor and members of the National Science Foundation AI and Biodiversity Change (ABC) Global Center for valuable discussions. The authors thank the Ohio Supercomputer Center for their computational resources. **Funding:** F.L. and M.A.J. were funded by the NSF OISE-2330423.

References

Abdelwahed, H. R., Teng, M., Zbinden, R., Pollock, L., Larochelle, H., Tuia, D., & Rolnick, D. (2026). CISO: Species distribution modelling Conditioned on Incomplete Species Observations. *Methods in Ecology and Evolution*, *17*(3), 947–962. https://doi.org/10.1111/2041-210x.70238

Bellier, E., Kéry, M., & Schaub, M. (2016). Simulation-based assessment of dynamic *N* -mixture models in the presence of density dependence and environmental stochasticity. *Methods in Ecology and Evolution*, *7*(9), 1029–1040. https://doi.org/10.1111/2041-210X.12572

Bellier, E., Kéry, M., & Schaub, M. (2018). Relationships between vital rates and ecological traits in an avian community. *Journal of Animal Ecology*, *87*(4), 1172–1181. https://doi.org/10.1111/1365-2656.12826

Brintz, B., Fuentes, C., & Madsen, L. (2018). An Asymptotic Approximation to the N-Mixture Model for the Estimation of Disease Prevalence. *Biometrics*, *74*(4), 1512–1518. https://doi.org/10.1111/biom.12913

Brintz, B. J., Madsen, L., & Fuentes, C. (2023). A spatially explicit N-mixture model for the estimation of disease prevalence. *Statistical Modelling*, *23*(1), 31–52. https://doi.org/10.1177/1471082X211020872

Callaghan, C. T., Santini, L., Spake, R., & Bowler, D. E. (2024). Population abundance estimates in conservation and biodiversity research. *Trends in Ecology & Evolution*, *39*(6), 515–523. https://doi.org/10.1016/j.tree.2024.01.012

Christin, S., Hervet, É., & Lecomte, N. (2019). Applications for deep learning in ecology. *Methods in Ecology and Evolution*, *10*(10), 1632–1644. https://doi.org/10.1111/2041-210X.13256

Cormack, R. M. (1964). Estimates of Survival from the Sighting of Marked Animals. *Biometrika*, *51*(3/4), 429. https://doi.org/10.2307/2334149

Dail, D., & Madsen, L. (2011). Models for Estimating Abundance from Repeated Counts of an Open Metapopulation. *Biometrics*, *67*(2), 577–587. https://doi.org/10.1111/j.1541-0420.2010.01465.x

Desjardins-Proulx, P., Poisot, T., & Gravel, D. (2019). Artificial Intelligence for Ecological and Evolutionary Synthesis. *Frontiers in Ecology and Evolution*, *7*, 402. https://doi.org/10.3389/fevo.2019.00402

DiRenzo, G. V., Che-Castaldo, C., Saunders, S. P., Campbell Grant, E. H., & Zipkin, E. F. (2019). Disease-structured *N* -mixture models: A practical guide to model disease dynamics using count data. *Ecology and Evolution*, *9*(2), 899–909. https://doi.org/10.1002/ece3.4849

Gelman, A., & Hill, J. (2021). *Data analysis using regression and multilevel/hierarchical models* (23rd printing). Cambridge Univ. Press.

Guillera-Arroita, G., Ridout, M. S., & Morgan, B. J. T. (2010). Design of occupancy studies with imperfect detection. *Methods in Ecology and Evolution*, *1*(2), 131–139. https://doi.org/10.1111/j.2041-210X.2010.00017.x

Hostetler, J. A., & Chandler, R. B. (2015). Improved state-space models for inference about spatial and temporal variation in abundance from count data. *Ecology*, *96*(6), 1713–1723. https://doi.org/10.1890/14-1487.1

IPBES. (2019). *Global assessment report on biodiversity and ecosystem services of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services* (Version 1). Zenodo. https://doi.org/10.5281/ZENODO.3831673

Jolly, G. M. (1965). Explicit Estimates from Capture-Recapture Data with Both Death and Immigration-Stochastic Model. *Biometrika*, *52*(1/2), 225. https://doi.org/10.2307/2333826

Joseph, M. B. (2020). Neural hierarchical models of ecological populations. *Ecology Letters*, *23*(4), 734–747. https://doi.org/10.1111/ele.13462

Kéry, M., & Royle, J. A. (2015). *Applied Hierarchical Modeling in Ecology: Analysis of distribution, abundance and species richness in R and BUGS: Volume 1:Prelude and Static Models* (1st edition). Academic Press.

Kéry, M., & Royle, J. A. (2020). *Applied Hierarchical Modeling in Ecology: Analysis of Distribution, Abundance and Species Richness in R and BUGS: Volume 2: Dynamic and Advanced Models* (1st edition). Academic Press.

Kidwai, Z., Jimenez, J., Louw, C. J., Nel, H. P., & Marshal, J. P. (2019). Using N-mixture models to estimate abundance and temporal trends of black rhinoceros (Diceros bicornis L.) populations from aerial counts. *Global Ecology and Conservation*, *19*, e00687. https://doi.org/10.1016/j.gecco.2019.e00687

Kingma, D. P., & Ba, J. (2014). *Adam: A Method for Stochastic Optimization* (Version 9). arXiv. https://doi.org/10.48550/ARXIV.1412.6980

Krogh, A., & Hertz, J. (1991). A Simple Weight Decay Can Improve Generalization. In J. Moody, S. Hanson, & R. P. Lippmann (Eds.), *Advances in Neural Information Processing Systems* (Vol. 4). Morgan-Kaufmann. https://proceedings.neurips.cc/paper_files/paper/1991/file/8eefcfdf5990e441f0fb6f3fad709e21-Paper.pdf

Langrock, R., & King, R. (2013). Maximum likelihood estimation of mark–recapture–recovery models in the presence of continuous covariates. *The Annals of Applied Statistics*, *7*(3). https://doi.org/10.1214/13-AOAS644

Lebreton, J.-D., Burnham, K. P., Clobert, J., & Anderson, D. R. (1992). Modeling Survival and Testing Biological Hypotheses Using Marked Animals: A Unified Approach with Case Studies. *Ecological Monographs*, *62*(1), 67–118. https://doi.org/10.2307/2937171

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. https://doi.org/10.1038/nature14539

Leroy, F., Jarzyna, M. A., & Keil, P. (2026). Acceleration hotspots of North American birds' decline are associated with agriculture. *Science*, *391*(6788), 917–921. https://doi.org/10.1126/science.ads0871

Leung, B., Hargreaves, A. L., Greenberg, D. A., McGill, B., Dornelas, M., & Freeman, R. (2020). Clustered versus catastrophic global vertebrate declines. *Nature*, *588*(7837), 267–271. https://doi.org/10.1038/s41586-020-2920-6

MacKenzie, D. I., Nichols, J. D., Hines, J. E., Knutson, M. G., & Franklin, A. B. (2003). Estimating site occupancy, colonization, and local extinction when a species is detected imperfectly. *Ecology*, *84*(8), 2200–2207. https://doi.org/10.1890/02-3090

MacKenzie, D. I., Nichols, J. D., Lachman, G. B., Droege, S., Andrew Royle, J., & Langtimm, C. A. (2002). Estimating site occupancy rates when detection probabilities are less than one. *Ecology*, *83*(8), 2248–2255. https://doi.org/10.1890/0012-9658(2002)083%5B2248:ESORWD%5D2.0.CO;2

National Audubon Society. (2020). *The Christmas Bird Count Historical Results*. http://www.christmasbirdcount.org

Parker, M. R. P., Li, Y., Elliott, L. T., Ma, J., & Cowen, L. L. E. (2021). Under-reporting of COVID-19 in the Northern Health Authority region of British Columbia. *Canadian Journal of Statistics*, *49*(4), 1018–1038. https://doi.org/10.1002/cjs.11664

Pichler, M., & Hartig, F. (2021). A new joint species distribution model for faster and more accurate inference of species associations from big community data. *Methods in Ecology and Evolution*, *12*(11), 2159–2173. https://doi.org/10.1111/2041-210X.13687

Pichler, M., & Hartig, F. (2023). Machine learning and deep learning—A review for ecologists. *Methods in Ecology and Evolution*, *14*(4), 994–1016. https://doi.org/10.1111/2041-210X.14061

Plummer, M. (2003). *JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling. In K. Hornik, F. Leisch, & A. Zeileis (Eds.), Proceedings of the 3rd International Workshop on Distributed Statistical Computing, 124(125.10), 1–10.*

Pollock, L. J., Kitzes, J., Beery, S., Gaynor, K. M., Jarzyna, M. A., Mac Aodha, O., Meyer, B., Rolnick, D., Taylor, G. W., Tuia, D., & Berger-Wolf, T. (2025). Harnessing artificial intelligence to fill global shortfalls in biodiversity knowledge. *Nature Reviews Biodiversity*, *1*(3), 166–182. https://doi.org/10.1038/s44358-025-00022-3

Pradel, R. (1996). Utilization of Capture-Mark-Recapture for the Study of Recruitment and Population Growth Rate. *Biometrics*, *52*(2), 703. https://doi.org/10.2307/2532908

Prechelt, L. (2012). Early Stopping—But When? In G. Montavon, G. B. Orr, & K.-R. Müller (Eds.), *Neural Networks: Tricks of the Trade* (Vol. 7700, pp. 53–67). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-35289-8_5

Priol, P., Mazerolle, M. J., Imbeau, L., Drapeau, P., Trudeau, C., & Ramière, J. (2014). Using dynamic *N* -mixture models to test cavity limitation on northern flying squirrel demographic parameters using experimental nest box supplementation. *Ecology and Evolution*, *4*(11), 2165–2177. https://doi.org/10.1002/ece3.1086

Rabiner, L. R. (1989). A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, *77*(2), 257–286. https://doi.org/10.1109/5.18626

Rosenberg, K. V., Dokter, A. M., Blancher, P. J., Sauer, J. R., Smith, A. C., Smith, P. A., Stanton, J. C., Panjabi, A., Helft, L., Parr, M., & Marra, P. P. (2019). Decline of the North American avifauna. *Science*, *366*(6461), 120–124. https://doi.org/10.1126/science.aaw1313

Rossman, S., Yackulic, C. B., Saunders, S. P., Reid, J., Davis, R., & Zipkin, E. F. (2016). Dynamic *N* -occupancy models: Estimating demographic rates and local abundance from detection-nondetection data. *Ecology*, *97*(12), 3300–3307. https://doi.org/10.1002/ecy.1598

Royle, J. A. (2004). N-Mixture Models for Estimating Population Size from Spatially Replicated Counts. *Biometrics*, *60*(1), 108–115. https://doi.org/10.1111/j.0006-341X.2004.00142.x

Royle, J. A., & Dorazio, R. M. (2008). *Hierarchical modeling and inference in ecology: The analysis of data from populations, metapopulations and communities* (Transferred to digital print). Academic Press.

Sandercock, B. K. (2006). Estimation of Demographic Parameters from Live-Encounter Data: A Summary Review: <b/>. *Journal of Wildlife Management*, *70*(6), 1504–1520. https://doi.org/10.2193/0022-541X(2006)70%5B1504:EODPFL%5D2.0.CO;2

Seber, G. A. F. (1965). A Note on the Multiple-Recapture Census. *Biometrika*, *52*(1/2), 249. https://doi.org/10.2307/2333827

Strebel, N., Wechsler, S., Bühler, R., Häfliger, G., Keller, V., Kéry, M., Rogenmoser, C., Spiess, M., Varga, K., Volet, B., Zbinden, N., & Schmid, H. (2025). Data of the Swiss common breeding bird monitoring program. *Ecology*, *106*(12), e70268. https://doi.org/10.1002/ecy.70268

Telenský, T., Storch, D., Klvaňa, P., & Reif, J. (2024). Extension of Pradel capture–recapture survival-recruitment model accounting for transients. *Methods in Ecology and Evolution*, *15*(2), 388–400. https://doi.org/10.1111/2041-210X.14262

Toszogyova, A., Smyčka, J., & Storch, D. (2024). Mathematical biases in the calculation of the Living Planet Index lead to overestimation of vertebrate population decline. *Nature Communications*, *15*(1), 5295. https://doi.org/10.1038/s41467-024-49070-x

Wilkinson, D. P., Golding, N., Guillera-Arroita, G., Tingley, R., & McCarthy, M. A. (2019). A comparison of joint species distribution models for presence–absence data. *Methods in Ecology and Evolution*, *10*(2), 198–211. https://doi.org/10.1111/2041-210X.13106

Zhao, Q., Fuller, A. K., & Royle, J. A. (2022). Spatial dynamic N-mixture models with interspecific interactions. *Methods in Ecology and Evolution*, *13*(10), 2209–2221. https://doi.org/10.1111/2041-210X.13936

Zhao, Q., & Royle, J. A. (2019). Dynamic N-mixture models with temporal variability in detection probability. *Ecological Modelling*, *393*, 20–24. https://doi.org/10.1016/j.ecolmodel.2018.12.007

Ziolkowski Jr., D., Lutmerding, M., Aponte, V., & Hudson, M.-A. (2022). *2022 Release—North American Breeding Bird Survey Dataset (1966-2021)* [Csv]. U.S. Geological Survey. https://doi.org/10.5066/P97WAZE5

Zipkin, E. F., Rossman, S., Yackulic, C. B., Wiens, J. D., Thorson, J. T., Davis, R. J., & Grant, E. H. C. (2017). Integrating count and detection–nondetection data to model population dynamics. *Ecology*, *98*(6), 1640–1650. https://doi.org/10.1002/ecy.1831

Zipkin, E. F., Thorson, J. T., See, K., Lynch, H. J., Grant, E. H. C., Kanno, Y., Chandler, R. B., Letcher, B. H., & Royle, J. A. (2014). Modeling structured population dynamics using data from unmarked individuals. *Ecology*, *95*(1), 22–29. https://doi.org/10.1890/13-1131.1

Zucchini, W., MacDonald, I. L., & Langrock, R. (2017). *Hidden Markov Models for Time Series: An Introduction Using R* (2nd ed.). Chapman and Hall/CRC. https://doi.org/10.1201/b20790