

Scalable spatio-temporal community modelling via Krylov subspace methods

Ongoing technological advance has dramatically expanded the set of data collection methods available for community ecologists. Nowadays various recording devices, such as camera traps, audiorecorders and water/air probe samplers, being properly set-up, are capable to operate in effectively autonomous manner constantly. Combined with recent advances in automatic species identification methods from raw recordings, these machines can yield community data on an unprecedented scale in terms of size and collection standardisation. Moreover, all this data is typically automatically annotated with spatial and temporal information, enabling ecologists to assess observed spatio-temporal patterns and better infer community-shaping drivers.

However, such big community data also lead to substantial challenges in its analysis. Though joint species distribution models (JSDM) capable to deal with explicit spatio-temporal sampling designs have been introduced half a decade ago, their scalability is far behind of what is required for newly-produced datasets. In this work we specifically aim to alleviate such limitation and develop a method applicable to datasets on hundreds of species recorded for thousands of times in hundreds of locations.

We start by revising the common latent-factor-based JSDM formulation and re-introduce spatio-temporal Gaussian process (GP) priors for the latent factors. While such model set-up is fairly simple, its fitting has been the crucial bottleneck of modern methods. More specifically, the existing approach is doomed by involved matrix inversions that scale cubically with respect to the amount of observed spatio-temporal units. We overcome it with carefully designed computations that are inspired by recent successes in GP machine learning literature. Those are based on the following tricks: 1) we constrain to a family of separable GP covariance functions, enabling Matrix Normal distribution compact form, 2) we never explicitly invert large matrices, but use iterative Krylov subspace methods (requiring matrix-vector multiplications only), 3) we design a custom adaptive preconditioner for faster iterative methods' convergence, and 4) we accelerate the computationally intensive model fitting parts with precompiled GPU-deployable TensorFlow code.

We implement our development as an extension of the popular Hmsc package in R, aiming for increased reusability by other ecologists. We demonstrate its performance and efficiency using real data case study.