Statistical modelling of species distributions using presence-only data

A semantic and graphical approach using the tree of life

Juan Manuel Escamilla Mólgora

Supervisors: Prof. Peter Atkinson,

Dr. Luigi Sedda,

Prof. Peter Diggle

Lancaster Environment Centre and Centre for Health Informatics,

Computing and Statistics

Lancaster University

This dissertation is submitted for the degree of

Doctor of Philosophy
December 2020

To Laura Reinelt, for her love, support and daily company...

To my parents, Teresa Mólgora Buchanan and Sergio Escamilla Suárez for giving me life and the motivation of pursuing a career in science...

To my aunt, María Mólgora for her teachings...

"Good acts nor fair institutions cannot exist if life is not reaffirmed first. The universal principle of ethics, politics, economics and all sciences is the acknowledgement of life."

...

The model of boundless progress adopted by the modern civilisation has assumed the existence of unlimited resources and energy that ultimately negates the material conditions for development and reproduction of life on Earth and, consequently, the human life.

...

A critical deconstruction of the principles and foundations of all human activities is needed, if humanity is willing to survive and live in dignity. (Dussel, 1998) [1] [2].

[1] In Spanish Dussel, E., 1998. Ética de la liberación en la edad de la globalización y de la exclusión, Segunda ed. Simancas Ediciones, S.A., Valladolid.

[2] English version: Dussel, E., 2013. Ethics of Liberation. Duke University Press. doi:10.2307/j.ctv1131d8k

DECLARATION

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgements. This dissertation contains fewer than 65,000 words including appendices, bibliography, footnotes, tables and equations and has fewer than 150 figures.

Juan Manuel Escamilla Mólgora

December 2020

ACKNOWLEDGEMENTS

I want to deeply thank my supervisors for their trust, help and mentorship. Personally, Peter Atkinson for his support, thoughtful ideas, trust and encouragement to pursue ambitious problems; Luigi Sedda for his infinite patience, his always-available counseling, both on technical and emotional matters, and his friendship; Peter Diggle for accepting me as a student and showing me the elegance, formality and precision of statistics. I also want to thank my partner Laura Reinelt for her patience, support and will to share her life with me. My friends in Lancaster, especially, Jaqueline Stenfert-Kroese, Camila Silva, Fernando Melo, Runmei Wang (LEC), Erick Chacón, Claudio Fonterre and Ollatunji Johnson (CHICAS) for their love, support, and ideas shared; My parents, my aunt and my brother for their love and support. My friends in Mexico, especially, Everardo Robredo, Juan Carlos González and Bruno Barrales for their help and discussions around ecology and species distributions. The free and open source community for sharing freely their work with the world. The people of Mexico for supporting with their taxes my living and tuition fees during the length of the scholarship.

ABSTRACT

Understanding the mechanisms that determine and differentiate the establishment of organisms in space is an old and fundamental question in ecology. The emergence of life's spatial patterns is guided by the confluence of three forces: the environmental filtering, which unbalances the probability of establishment for organisms given their evolutionary adaptations to local environmental conditions; the biological interactions, which restrict their establishment according to the presence (or absence) of other organisms; the diversification of organisms' strategies (traits) to migrate and adapt to changing environments. The main hypothesis in this research is that the accumulated knowledge of biodiversity occurrences, the species taxonomic classification and geospatial environmental data can be integrated into a unified modelling framework to characterise the joint effect of these three forces and, thus, contribute with more general, accurate and statistically sound species distributions models (SDM)s.

The first part of this thesis describes the design and implementation of a knowledge engine capable to synthesise and integrate environmental geospatial data, taxonomic relationships and species occurrences. It uses semantic queries to instantiate complex data structures, represented as networks of concepts (knowledge graphs). Local taxonomic trees, distributed over a hierarchical spatial system of regular lattices are used as knowledge graphs to perform data synthesis, geoprocessing, and transformations. The implementation uses efficient call-by-need evaluations that facilitates spatial and scale analysis on large datasets.

The second part of the thesis corresponds to the statistical specification and implementation of two modelling frameworks for species distribution models (one for single species and other for multiple species). These models are designed for presence-only observations; obtained from the knowledge engine. The common specification of these models are that presence-only observations are the joint effect of two latent processes: one, that defines the species presence (ecological suitability); and other, that defines the probability of being sampled (sampling effort). The single species framework uses an informative sample, chosen by the modeller, to account for the sampling effort. Three modelling strategies are proposed for accounting the joint effect of the ecological and sampling process (independent processes, a common spatial random effect and correlated processes). The tree models were compared to the maximum entropy model (MaxEnt), a popular algorithm used in SDMs. All models showed a better predictive performance than MaxEnt.

The multi-species modelling framework is a generalisation of the single species framework for developing a joint species distribution model for presence-only data. The specification is a multilevel hierarchical logistic model with a single spatial random effect, common to all species of interest. The sampling effort is modelled as a *complementary sample* obtained by complementary observations from the taxa of interest using a regional taxonomic tree. The model was tested against simulated data. All simulated parameters were covered by the credible intervals of the posterior sampling. A study case in Easter Mexico was presented as an application of the model. The results obtained in the case study were consistent with the macroecological theory. The model showed to be effective in removing bias and noise given by the sampling effort. This effect was particularly impressive in in urban areas, where the sampling intensity is greater. The research presented here provides an interdisciplinary approach for modelling joint species distributions aided by the automated selection of biological, spatial and environmental context.

TABLE OF CONTENTS

List of figures **xix**

List of tables **xxvii**

List of symbols **xxxiii**

List of acronyms **xxxv**

1 Introduction **1**

1.0.1 A constructivistic representation of knowledge 4

1.1 Aim of the thesis 6

1.2 Structure of the thesis 7

1.2.1 Part I: the knowledge system 8

1.2.2 Part II: The statistical framework 9

I The knowledge engine **13**

2 Biospytial: graph-based geospatial computing **15**

2.1 Introduction 17

2.2 An *Open Source* graph-based engine for geospatial analysis 22

2.2.1 System architecture 23

2.2.2 Knowledge representation 26

2.2.3 Integrating data with graph structures and object mappings 29
2.2.4 Graph Traversals 30
2.2.5 Geospatial management and processing 32
2.3 Using Biospytial to analyse the Tree of Life 34
2.3.1 Study Area 35
2.3.2 Data used 35
2.3.3 Traversals on the Knowledge Graph 36
2.4 Worked examples 37
2.4.1 Additional data used 39
2.4.2 Methodology 39
2.4.3 Results of the worked example 40
2.4.4 Discussion 41
2.5 Tutorial 43
2.5.1 Selecting the node *Jaguar* 43
2.5.2 Converting cells to local taxonomic trees 46
2.5.3 Exploratory analysis on a single Tree 47
2.5.4 Tree operations 49
2.5.5 Selecting nodes from the Red List 52
2.5.6 Trimming trees 55
2.5.7 Associated raster (environmental) information 58
2.5.8 Extracting raster objects from arbitrary polygons 60
2.5.9 Network visualisation and analysis 64
2.6 Conclusions 67
2.7 Availability of supporting source code and requirements 68
2.8 Availability of supporting data 69
2.9 Abbreviations 69

2.10 Funding 70
2.11 Authors' contributions 70
2.12 Competing interests 70
2.13 Acknowledgments 70
2.14 Adding data in Biospytial 72
2.14.1 Aims of this tutorial 72
2.14.2 Assumptions 72
2.14.3 Converting the data to a Django Model 73
2.14.4 Vector data 73
2.14.5 Export Shapefile into the Database (Geoprocessing Container) 74
2.14.6 Example 2: Adding vector data 76
2.14.7 Add raster data 77
2.15 Mathematical formalisms 79
2.15.1 Mathematical definitions 79
2.15.2 Biological definitions 81
2.16 Theoretical consequences 81
2.17 Formal data specification 83

II The statistical framework 87

3 A framework for modelling species distributions with presence-only data 89
3.1 Introduction 91
3.2 Materials and Methods 96
3.2.1 Model summary 96
3.2.2 A Choosing Principle for obtaining presences, relative absences and missing observations 99
3.3 Applications 102

3.3.1 Study region . . . 104
3.3.2 Occurrence data . . . 107
3.3.3 Treatments for missing data . . . 107
3.3.4 Explanatory variables . . . 108
3.3.5 Data preprocessing . . . 109
3.3.6 Inference and prediction . . . 109
3.3.7 Comparison between models . . . 110
3.3.8 Comparison against Maxent . . . 110
3.4 Results . . . 112
3.4.1 Presence of Pines . . . 112
3.4.2 Results for the Presence of Flycatchers (family *Tyrannidae*) . . . 114
3.5 Discussion . . . 121
3.5.1 The role of the choosing principle . . . 122
3.5.2 Advantages in using this framework . . . 124
3.5.3 Limitations . . . 125
3.6 Conclusions . . . 126
3.7 Data and source code availability . . . 126
3.8 Declarations . . . 127
3.8.1 Funding . . . 127
3.8.2 Authors' contributions . . . 127
3.8.3 Conflicts of interest . . . 127
3.9 Supplementary materials I: Framework specification . . . 128
3.9.1 Latent variables P_Y and P_X . . . 129
3.9.2 Random effects . . . 130
3.9.3 Three models for spatial autocorrelation . . . 131
3.10 Supplementary materials II . . . 135

3.10.1 Estimates for the predicted presence of Pines using botanical records as sample . . . 135
3.10.2 Maps of posterior variables for the presence of Pines . . . 138
3.11 Estimates for the predicted presence of tyranids using birds records as sample 144
3.11.1 Maps of posterior probabilities for Tyranids . . . 146

4 A taxonomic-based joint species distribution model for presence-only data 153
4.1 Introduction . . . 156
4.2 Methods . . . 160
4.2.1 Support for missing data . . . 164
4.2.2 Obtaining the sampling effort with complementary taxa . . . 165
4.2.3 Model Implementation . . . 166
4.3 Validation with simulated data . . . 168
4.4 Application to biodiversity occurrences in eastern Mexico . . . 169
4.4.1 Study region . . . 169
4.4.2 Explanatory variables . . . 169
4.4.3 Occurrence and taxonomic data . . . 170
4.4.4 Selection of taxa . . . 171
4.4.5 Data preprocessing . . . 172
4.4.6 Model fitting . . . 172
4.4.7 Cross validation with occurrence observations . . . 173
4.4.8 Results . . . 173
4.4.9 Cross-validation . . . 177
4.5 Discussion . . . 177
4.6 Conclusion . . . 181
4.7 Acknowledgments . . . 181
4.8 Data and source code availability . . . 182

4.9 Data and source code availability . . . 182
4.10 Authors' contributions . . . 182
4.11 Appendix: Model description . . . 183
4.11.1 Support for missing data . . . 187
4.12 Appendix: Simulation study . . . 188
4.12.1 Model fitting . . . 189
4.12.2 Results . . . 190
4.13 Appendix: Implementation of the model in STAN . . . 194
4.14 Appendix: Visualisations of simulated data and associated spatial structures 200

5 General discussions and recommendations for the future 203
5.1 Spatial point processes as an alternative . . . 207
5.2 Limits and recommendations for the future . . . 210
5.2.1 Absence of rare species . . . 210
5.2.2 Limitations on the single species framework . . . 211
5.2.3 Limitations of the multiple species model . . . 211
5.2.4 Computational limitations . . . 212
5.2.5 Random effects as graph traversals . . . 213
5.2.6 The temporal dimension . . . 214

6 Conclusions 215

References 217

Appendix A Extra mathematical definitions 237
A.1 Additional mathematical definitions . . . 237
A.1.1 Network . . . 237
A.1.2 Algebraic operations . . . 238
A.2 Deprecated software and the future of the engine . . . 239

III Published research article 241

LIST OF FIGURES

2.1 The Biospytial System with the three interconnected modules. a) The GSPU, where semantic queries and graph traversals take place. b) The BCE, where object mappings, web services and the modelling framework takes place. It includes several libraries for performing exploratory analysis as well as Bayesian statistical inference and prediction using the probabilistic programming language: PYMC3; c) All the components can be allocated in the cloud and are connected using virtual and physical networks. d) The RGU, where the geoprocessing and spatial indexing occurs, storing efficiently any raster and vector data sources. e) Interactive access is possible in two ways: using an online web notebook (Jupyter) or an interactive console (iPython). 28

2.2 The graph showing the connection between a *Species* node and two *Cell* nodes. Here: the species is *Pharomachrus mocinno* (Quetzal) and the number shown in each Cell node is its respective ID number. This is an actual visualisation taken from data stored in our Knowledge Graph. 29

2.3 A subgraph from the Knowledge Engine that shows the second order degree of neighbouring cells where at least one occurrence of any type of mosquito (family *Culicidae*) was registered. This query exemplifies the use of recursive lookups. In this case the relationship "IS_NEIGHBOUR_OF" is traversed twice. 32

2.4 Raster manipulation in the knowledge engine. a) a multipolygon selection corresponding to Mexico, an instance from the class `Country` that maps into the *WorldBorders* dataset. b) An `Elevation` object (class `RasterData`) instantiated with a customized polygon, in this case a subregion of the object Mexico. c), d) and e) are `RasterData` objects derived from the Elevation object. The data and visualisations were produced using the engine's raster API. The code for generating these figures are in supplementary materials. . 34

2.5 A visualisation of a Local Taxonomic Tree built with the relationship: `IS_PARENT_OF`. The rectangles show zoomed areas in different sections of the tree (upper region for Birds (Order Aves), lower for plants (Order Magnoliopsida)). Colored nodes indicate distinct taxonomic levels (red : species, yellow: genera, grey: families, green orders, purple: classes). 38

2.6 Comparison of mean annual environmental ranges between treatments: All Mexico, threatened taxa and cells with occurrences of jaguars. See next section for more details. 41

2.7 The output of the method: `display_field()`, an easy way to visualise RasterData objects. 62

2.8 A composite figure showing two states of the interactive visualisation. Orange dots represent occurrences of threated species associated with the presence of jaguars (*P. Onca*). The inland red square shows the zoomed-in area depicted in the left side of the figure. The colored squares in the zoomed area shows the mean temperature associated with threatened vertebrates (phylum Chordata). The base map shows the elevation for all the country. See section 2.3.2 for information regarding the data used. 64

2.9 A tree visualization for the merged tree corresponding to threatened taxa, showing up to *Order* level. The size of the nodes is proportional to the taxonomic level (the largest is the root of the tree, the smallest are orders). The node colouring indicates the frequency of occurrence with respect to all the neighbouring cells (neighbours of jaguars) being the brightest the highest ranked and the darker the lowest ranked. 66

3.1 Directed acyclic graphs for the three model specifications. Variables in squares account for observations: Y : presence of a taxon of interest (e.g. species) and X : presence of sample. Circles in blue correspond to latent variables while circles in grey correspond to parameters. Variables P_X and P_Y correspond to the latent processes of the sampling effort and ecological suitability, variables R_X and R_Y correspond to the random effect for the sampling effort and the ecological suitability processes respectively. Variables β_X and β_Y represent the parameters of the fixed effects (linear components) of the latent processes P_X and P_Y respectively. Squares in salmon colour indicate environmental (d_Y) and anthropic (d_X) explanatory variables. The variables inside the dark grey block define the random effects component; different in the three models. Variables S, S_X and S_Y describe the spatial component defined as Gaussian Markov Random Fields, while variables Z_X and Z_Y represent unstructured variability within an area. 105

3.2 A map showing the study area (overlaid semicircular polygon) over central Mexico. Important cities are shown as grey polygons scattered across the area. Greener areas represent higher vegetation cover. The basemap used as background was obtained from the ESRI topographic tiling service. 106

3.3 Comparison of models I, II and III against the maximum entropy algorithm (bottom left panel). The maps displayed here corresponds to the posterior mean probability for the three models using observations of pines as presence (panels on left) and botanical records (panels on right) as the sampling process. The bottom right panel shows the observations used to fit the models.115

3.4 Area under the receiver operating characteristic curve (AUC-ROC) for the different models of the pines example (left panel) and the birds example (right panel). The dots in colours represent a MaxEnt models using different parameters of regularisation (x-axis) and feature type (vertical legend). The values in the y-axis correspond to the resulting AUC-ROC value according to that specific pair of parameters. The AUC-ROC values of models I (red), II (green) and III (blue) are shown as horizontal lines. Solid lines represent the mean AUC-ROC values for models I, II and III, while dotted and dashed lines represent their respective lower and upper (95%) confidence intervals. 116

3.5 Comparison of models I, II and III against the maximum entropy algorithm (bottom left panel). The maps displayed here corresponds to the posterior mean probability for the three models using observations of flycatchers as presence (panels on left) and observations of birds records (panels on right) as the sampling process. The bottom right panel shows the observations used to fit the models. 120

3.6 Mean probability and 95% C.I. for Presence, Sample, and Joint presence and sample for Models I, II and III predicting presence of Pines (Class: Pinopsida) using Plants (Kingdom: Plantae) as sample. 138

3.7 Latent variable P_Y (Presence) for Models I, II and III predicting presence of Pines. The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively. . . . 139

3.8 Spatial random effect S_Y. The Gaussian Markov random field (GMRF) corresponding to the latent variable P_Y (Presence) for Models I, II and III predicting presence of Pines. The central column corresponds to the mean value, The column on the left and right corresponds to quantiles: 0.025 and 0.975, respectively. 140

3.9 Latent variable P_X (Sample) for Models I, II and III predicting presence of Pines using all plants as sample. The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively. 141

3.10 Spatial random effect S_X. The Gaussian Markov random field (GMRF) corresponding to the latent variable S_X (Sample) for Models I, II and III predicting presence of Pines. The central column corresponds to the mean value. The column on the left and right corresponds to quantiles: 0.025 and 0.975, respectively. 142

3.11 Area under the receiver operating characteristic curve (AUC-ROC) for the different models of Pines. The three models (b,c and d) perform significantly better than MaxEnt. 143

3.12 Mean probability and 95% C.I. for Presence, Sample, and Joint presence and sample for Models I, II and III predicting presence of flycatchers (Family: Tyrannidae) using birds (Class: Aves) as sample. 146

3.13 Latent variable P_Y (Presence) for Models I, II and III predicting presence of flycatchers (Family: Tyrannidae). The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively. 147

3.14 Spatial random effect S_Y. The Gaussian Markov random field (GMRF) corresponding to the latent variable P_Y (Presence) for Models I, II and III predicting presence of flycatchers (Family: Tyrannidae). The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively. 148

3.15 Latent variable P_X (Sample) for Models I, II and III predicting presence of flycatchers (Tyrannidae) using all birds as sample. The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively. 149

3.16 Spatial random effect S_X. The Gaussian Markov random field (GMRF) corresponding to the latent variable P_X (Sample) for Models I, II and III predicting presence of flycatchers (Tyrannidae). The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively. 150

3.17 Area under the receiver operating characteristic curve (AUC-ROC) for MaxEnt and models I, II and III of flycatchers. MaxEnt and models I and III achieved low AUC. Although, on average models I and III outperformed MaxEnt, their variances show that these models are not appropriate when the proportion of missing data is significantly higher than the presences. See the discussion section for a more detail explanation. 151

4.1 Directed acyclic graph representing the multi-species model. Nodes in squares indicate data. Blue squares are observed records. y_i are the presences of species i and y^c are the complementary records of $\bigcup_{i=1}^{n} y_i$, i.e. the records that are not from species $1, \ldots$ nor n, relative to the available dataset and an arbitrary taxonomic branch (Ω). Orange squares are covariates, e_i for environmental based and a_s for anthropological based, associated with the sampling effort (dark blue block). Nodes in blue circles represent latent variables where: Q_i is the mix between the sampling effort S and the corresponding ecological suitability process P_i. The node G represents the spatial random effect (CAR) shared between both the sampling effort components (dark blue block) and the ecological components (green block). Circular grey nodes represent the parameters used by the latent variables. 163

4.2 Comparison of each ecological suitability (E.S.) processes P_i (a,b,c,h and g) and its corresponding mixing effect (M.E.) Q_i (d,e,f,j and k) in the study area (l). Q_i is the convex combination of P_i with the sampling effort process S (i). All figures show the respective mean posterior on each unit element of the lattice $\mathbb{W}$ in the study area (l). 176

4.3 Receiver operator characteristic curve (ROC) obtained from the 10-fold cross-validation of the model applied to the case-study data. The orange solid line represents the ROC curve of the mean posterior prediction while the red and blue dashed lines represent the credible interval at 2.5% and 97.5% resp. The diagonal (identity) line represents the ROC curve of random classification with constant 50% true positive rate. 177

4.4 Errors of the spatial random effect G calculated as the difference between the mean posterior sample and the simulated data. Left panel (a) shows the spatial arrangement while right panel shows the histogram of the errors. The posterior mean was obtained through 1000 iterations fitted with Gaussian observations $\hat{Y}_i$. 193

4.5 Adjancency matrix W of the latttice $\mathbb{W}$. The precision matrix ($Q = (D - \alpha W)$) and its corresponding Covariance matrix Q^{-1} (right) corresponding to a simulated gaussian markov random field (GMRF). The figures are restricted to the upper section of the matrices covering the first 100 entries.Simulation, sampled from a multivariate normal with mean 0 and covariance matrix Q^{-1} 201

LIST OF TABLES

2.1 Principal software components of the Biospytial Knowledge Engine System 27

2.2 Output for environmental variables. Here showing only mean values for some variables on a single record. 59

2.3 Corresponding URLs for source code and container images for the Biospytial engine. The modules and the source code do not include data. These should be installed separately or loaded independently. 69

3.1 Definitions of the used terms and symbols 103

3.2 Comparison of the presence-only models: Independent Spatial Components (Model 1), Common Spatial Component (Model 2), Correlated Spatial Components (Model 3) and Maximum Entropy (MaxEnt) for the presence of Pines (class *Pinopsida*) using botanical records (kingdom: *Plantae*) as sample effort. A 7-fold cross validation was performed to calculate the area under the receiver-operating characteristic curve (ROC-AUC) as a measure of quality for each model. Models with the $\star$ symbol were fitted using only missing data from X (sample), i.e. treatment *ii*. 113

3.3 Comparison of the presence-only models: Independent Spatial Components (Model 1), Common Spatial Component (Model 2), Correlated Spatial Components (Model 3) and Maximum Entropy (MaxEnt) for the presence of the family *Tyrannidae* using birds as sample (class: *Aves*). A 7-fold cross validation was performed to calculate the area under the receiver-operating characteristic curve (ROC-AUC) as a measure of quality for each model. Models with the $\star$ symbol were fitted using only missing data from X (sample), i.e. treatment *ii*. 118

3.1 Posterior summaries of all the parameters in Model I with the associated 95% credible intervals for the example of pines. Parameters τ_Y^2 and τ_X^2 correspond to the variance of the spatial effects of the presence (Y) and the sample process (X) (i.e. S_Y and S_X) respectively. Likewise, σ_Y^2 and σ_X^2 correspond to the variance of the unstructured processes Z_Y and Z_X respectively. Significant parameters are shown in **bold**. For further information see section: 3.3 136

3.2 Posterior summaries of all the parameters in Model II with the associated 95% credible intervals for the example of pines. The parameter τ^2 represents the variance of the common spatial effect. Parameters σ^2 and σ^2 correspond to the variance of the unstructured process Z_Y and Z_X. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3 136

3.3 Posterior summaries of all the parameters in Model III with the associated 95% credible intervals for the example of pines. Parameters σ_Y^2 and σ_X^2 correspond to the variance for the presence (Y) and the sample (X). The term $\text{corr}_{X,Y}$ indicates the correlation between these two processes. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3 137

3.4 Posterior summaries of all the parameters in model I with the associated 95% credible intervals for the example of flycatchers. Parameters τ_Y^2 and τ_X^2 correspond to the variance of the spatial effects of the presence and the sample process (S_Y and S_X) respectively. Likewise, σ_Y^2 and σ_X^2 correspond to the variance of the unstructured processes Z_Y and Z_X respectively. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3 144

3.5 Posterior summaries of all the parameters in Model II with the associated 95% credible intervals for the example of flycatchers. The parameter τ^2 represents the variance of the common spatial effect. Parameters σ^2 and σ^2 correspond to the variance of the unstructured process Z_Y and Z_X. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3 144

3.6 Posterior summaries of all the parameters in Model III with the associated 95% credible intervals for the example of flycatchers. Parameters σ_Y^2 and σ_X^2 correspond to the variance for the presence (Y) and the sample (X). The term $\text{corr}_{X,Y}$ indicates the correlation between these two processes. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3 145

4.1 Posterior means, 95% credible intervals and convergence diagnostic $\hat{R}$ for the case-study of biodiversity records in the eastern part of Mexico. Ecological Suitability and Sampling Effort corresponds to the processes P and S defined in the main text. The Contribution to Ecological Suitability row describes the parameter α_i defined in the mixing process Q_i, for each taxon i 175

4.2 Chosen values used for simulating Q_i, P_i and S, given a matrix of covariates (sampled from a normal distribution) and a random effect G defined as a proper Gaussian Markov random field. 189

4.3 **[Continuous observations]** Comparison between simulated and inferred parameters sampled from the posterior joint probability distribution (see equations 4.1 for normal (continuous) observations ($\hat{Y}_i$). The inference was obtained by MCMC following 1000 iterations with a burn-in of 500. The β parameters correspond to the *fixed effects* of the species P_i for covariates 1 and 2. The parameters α correspond to the mixture between the P_i (probability of occurrence of species i and the sampling effort S). The parameters related to the variance are τ^2 for the spatial random effect and σ_q^2 for the unstructured random effect. All simulated parameters are within the 95% credible intervals 191

4.4 **[Binary observations]** Comparison between simulated and inferred parameters sampled from the posterior joint probability distribution (see equation 4.1 for binary observations (presence / absence) distributed as independent Bernoulli variables when conditioned to the latent random effect G. The inference was obtained from MCMC following 40000 iterations with a burn-in of 20000. The β parameters correspond to the *fixed effects* of the species P_i for covariates 1 and 2. The parameters α correspond to the mixture between the P_i (probability of occurrence of specie i and the sampling effort S). The parameters related to the variance are τ^2 for the spatial random effect. . . . 192

LIST OF SYMBOLS

- G : spatial autocorrelated process (latent variable) (chapter 4)
- P_X : latent variable for the sampling effort process (chapter 3)
- P_Y : latent variable for the ecological process (chapter 3)
- P_i : ecological suitability process for taxon (chapter 4)
- Q_i : A process that mixes the ecological suitability for taxon i and the sampling effort process (chapter 4)
- R_X : random effect of the sampling effort process (chapter 3)
- R_Y : random effect for the ecological process (chapter 3)
- S : sampling effort process (chapter 4)
- S_X : spatial random effect for the sampling effort process (chapter 3)
- S_Y : spatial random effect for the ecological process (chapter 3)
- X_k : observation of the sampling effort in the cell k (chapter 3)
- Y_k : observation of an occurrence (presence) in the cell k (chapter 3)
- Z_X : unstructured random effect for the sampling effort process (chapter 3)
- Z_Y : unstructured random effect for the ecological process (chapter 3)

- Ω : the totality of occurrences in a given database or, theoretically, all organisms in Earth.ToL : the (taxonomic) tree of Life (chapter 4)
- $\hat{R}$: A convergence diagnostic for Markov chains (Gelman et al., 1992). (chapter 4)
- $\mathbb{W}$: the spatial lattice. Its corresponding adjancency matrix is denoted as W. (chapter 3 and 4)
- $\widetilde{N_y}$: all the taxonomic nodes corresponding to a set of occurrences. That is, if the occurrences are the leaves of the ToL, N_y will include all the internodes from the root of the tree to each of the occurrences. (chapter 4)
- d_{a_s} : vector of covariates for the sampling effort (chapter 4)
- d_{e_i} : vector of environmental covariates (chapter 4)
- x_k : cell k on the spatial lattice $\mathbb{W}$ (chapter 4)
- $y^i_{x_k}$: presence observations at cell x_k (chapter 4)

List of acronyms

- AIC : Akaike information criteria (chapter 3)
- AUC : area under the curve (of the ROC curve)
- BAM : biotic, abiotic, movement diagram (chapter 3)
- BCE: Biospytial Computing Engine (chapter 2)
- BLOB: binary large object (chapter 2)
- CAR : conditional autoregressive model
- CI : credible interval
- CONABIO: National Commission for the Knowledge and Use of Biodiversity (chapter 2)
- CONACyT: National Council for Science and Technology (Mexico) (chapter 2)
- CRS: coordinate reference system (chapter 2)
- CSV: comma separated value (chapter 2)
- CmSC : common spatial component, eq. to model II (chapter 3)
- CrSC : correlated spatial component, eq. to model III (chapter 3)
- DAG: directed acyclic graph (chapter 2)

- DEM: digital elevation model (chapter 2)
- DIC : deviance information criteria (chapter 3)
- EBVs: Essential Biodiversity Variables (chapter 2)
- EPSG: European Petroleum Survey Group (chapter 2)
- GAM : generalised additive model (chapter 3)
- GBIF: Global Biodiversity Information Facility (chapter 2)
- GDAL: Geospatial Data Abstraction software Library (chapter 2)
- GLM : generalised linear model (chapter 3)
- GLMM : generalised linear mixed model (chapter 4)
- GMRF : Gaussian Markov random field
- GSPU: Graph Storage and Processing Unit (chapter 2)
- HMCMC : Hamiltonian Markov chain Monte Carlo (chapter 4)
- ICAR : intrinsic conditional autoregressive model
- ISC : independent spatial components, eq. to model I (chapter 3)
- JSDMs : joint species distribution models (chapter 4)
- MALA : Metropolis Adaptive Langevin Algorithm (chapter 3)
- MASL : meters above sea level (chapter 4)
- MCAR : multivariate conditional autoregressive model (chapter 3 and 4)
- MCMC : Markov chain Monte Carlo

- MPI: Message Passing Interface (chapter 2)
- MVN : multivariate normal distribution
- MaxEnt : maximum entropy algorithm (chapter 3 and 4)
- NUTS : No U-turn sampler, a MCMC sampling method (chapter 4)
- OGM: object-graph mapping (chapter 2)
- ORM: object-relational mapping (chapter 2)
- RDBMS: Relational Database Management System (chapter 2)
- RGU: Relational Geoprocessing Unit (chapter 2)
- ROC : receiver operating characteristic (curve) (chapter 3 and 4)
- SCAR : stationary conditional autoregressive model (chapter 4)
- SDI: spatial data infrastructure (chapter 2)
- SDM : species distribution model
- ToI : taxon of interest
- ToL: Tree of Life (chapter 2)
- WGS84 : world geodetic system of 1984, a geodetic datum. (chapter 4)
- WKT: well known text (chapter 2)
- X : sampling effort observations (chapter 3)
- Y : occurrence observations (chapter 3)
- dx : vector of covariates for the sampling effort (chapter 3)

- dy : vector of covariates for the ecological process (chapter 3)
- model I : independent spatial effects (chapter 3)
- model II : common spatial effect (chapter 3)
- model III : correlated spatial effects (chapter 3)

CHAPTER 1

INTRODUCTION

The research integrated in this thesis is an attempt to contribute to the development of computational and statistical methods for answering an old ecological question:

How is life distributed on Earth?

This question has been a central problem in ecology at least since the beginning of the XIX century. Some answers to this question appeared in the natural history voyages of von Humboldt and Bonpland (1807) and in the influential evolutionary theories of Darwin (1859) and Wallace (1876). During the mid XX century, novel ideas like the continental drift theory (Wegener, 1923) and the classification of climates (Köppen, 1918) inspired early works to explain the adaptations and needs of organisms, given an environmental context. Examples of these are the paleontological works of Simpson (1953), the descriptions of botanical distributions of Cain (1944) and, specially, the influential theory of island biogeography by MacArthur and Wilson (1967). The influx of these novel ideas allowed the synthesis of a study field in its own right. As such, the study of how organisms relate to each other and their environment, and how these relationships explain spatial statistical patterns of diversity, abundance, richness, evenness and community composition, became known as *Macroecology* (Brown et al., 1995).

Since the industrialization era of the early XIX century, the global natural environment has been altered profoundly by human activities (Stocker et al., 2013). The accelerating rate of land and ecosystems degradation induced by anthropogenic changes on land cover are provoking fast increasing rates in species extinctions (Foley, 2005). A consequence of this is the observed loss in biodiversity, a truly irreversible environmental change that Earth faces today (Dirzo and Raven, 2003). At the current pace, and without significant changes in society's standards for development and growth, humankind will witness the sixth mass extinction in the history of the planet (Ceballos et al., 2015). This will have devastating effects on life-sustaining processes on global biogeochemical cycles (Cavicchioli et al., 2019; Díaz et al., 2009; Ehrlich and Ehrlich, 2013; Hooper et al., 2012; Millennium Ecosystem Assessment, 2005; Pereira et al., 2010). It has been demonstrated that the current human-dominated epoch is impacting the global environment at such a magnitude that its effects will persist in the geologic record. Some authors like (Lewis and Maslin, 2015; Steffen et al., 2011) had promoted the use of the name *Antropocene* to refer to this contemporary geologic epoch. As such, it is a priority to produce methods and techniques to accurately describe the spatial distribution of the species using all the available information.

The growth of research in macroecology has been influenced by the need to advance the knowledge in applications related to the management, adaptation and mitigation of the society's impacts on the biosphere (Ferrier et al., 2016; Foden and Young, 2016; Intergovernmental Panel on Climate Change, 2014). With the development of formal ecological and statistical methods, aided with computational advances, predicting the spatial distribution of species have become an active research field, both theoretical and applied. See Elith and Leathwick (2009); Guisan et al. (2017); Guisan and Zimmermann (2000) and (Araújo et al., 2019) for a review. Sound statistical models for regression and classification have been developed for these purposes. Nevertheless, they often require

ad-hoc sampling designs to account for unbiased observations of the species under study. That is, these methods often rely on presence-absence or abundance (count) data, thus, limiting the synthesis and use of datasets to particular sites, research questions and objectives. As such, this thesis also tries to answer the question of:

How to integrate biodiversity data from different sources to infer species geographic distributions?

Paradoxically, the anthropocene has diversified technologically, opening opportunities to find solutions to some of the environmental problems it is causing. In particular, the revolution of information technologies (IT) has expanded the capacity to compute, store and transfer massive amounts of data. Environmental sciences have been benefited of this with the expansion of reliable and diverse data to measure natural phenomena, covering a wide range of *essential biodiversity variables* (EBVs) (Kissling et al., 2017) across diverse spatial and temporal scales. Examples of these are the remote sensing imagery from Earth observation systems like NASA's *Joint Polar Satellite System* (National Aeronautics and Space Administration et al., 2020) and the ESA's *Copernicus* programme (European Space Agency, 2014), for weather forecasting, natural disaster monitoring, etc. This IT era is opening opportunities for collecting data in many different forms. For example, ubiquitous Internet connectivity has made possible the transfer of data across large distances in a short time; and the multifunctional capabilities of mobile and *smart* devices has enabled the management and deployment of collaborative surveys at low costs with the collective help of communities of enthusiasts and volunteers. Geospatial citizen science is now possible with the use of methodologies for collecting, annotating and curating these new sources of spatial data (Goodchild, 2007), (Heipke, 2010) and (Kamel Boulos et al., 2011). Examples of these are (*crowd-based*) platforms like OpenStreetMap (OpenStreetMap Contributors, 2019) for geographic maps and the platform iNaturalist (ina, 2020) for registering species occurrences. This exponential growth of data brings

new challenges for storage, access, integration and analysis. To solve this, new theoretical methods and technologies (defined under the umbrella term of *Big Data*) are being developed to extract meaningful information from very large, complex and heterogeneous data collections (Chen et al., 2014) and (Mikalef et al., 2018). See (Li et al., 2016) for a review on theoretical and practical challenges involving big geospatial data. The advances in computational methods for big data and the expansion of environmental data had open the opportunity for integrating multiple sources of geospatial data. The challenge to develop novel data structures for representing ecologically meaningful data to support the analyses of species distributions is still open. As such, the integration and synthesis of heterogeneous environmental and biological data requires answering the question on:

How to formalise a comprehensive data structure to unify and synthesise hetereogeneous data sources ?

1.0.1 A constructivistic representation of knowledge

To answer the above question, a formal computational approach for representing knowledge with semantic and spatial networks is proposed. This approach is inspired on Piaget's constructivistic epistemology (1952) to model the acquisition of knowledge as a aggregated and interrelated concepts. In this theory, new concepts are built from the confrontation of previously recalled *schemes* with reality. A scheme, in this sense, is a network of associations between previous experiences and conceptual models of reality. As such, knowledge is the collection of schemes that associates concepts with real objects; a network of relationships between learned concepts. According to Piaget (1952), the acquisition of new knowledge can assimilated by established by knowledge schemes. However, when the real nature of the object contradicts the internal knowledge structures previously constructed, a critical process called *disequilibrium* occurs. When this happens, new evidence and cognitive processes need to occur in order to create a new knowledge scheme which can

interpret the new object, a process known as *accommodation*. Piaget's constructivistic epistemology is in some way similar to Popper's "Three Worlds" paradigm (1966), his views on *open theory* and falsifiability in science. For a theory to be *open* it must allow continuous testing and modifications when its previous propositions contradict the reality. Falsifiability in science is, therefore, the property of being subjected to critical analyses and rejecting false knowledge schemes when *new* evidence is presented.

Relational structures for representing knowledge have been used before in information sciences to analyse and integrate information from heterogeneous datasets through the concept of ontologies (Chandrasekaran et al., 1999; Guarino, 1995). Ontologies standardise knowledge schemes from a common domain of knowledge. For example, Bard and Rhee (2004) and (Smith et al., 2007) reviewed the applications and challenges for unifying standardised ontologies in the biological and medical sciences, while Madin et al. (2008) reviewed several proposals in ecology for integrating heterogeneous datasets. The research on these areas involves multiple organisation scales, going as far as phylogenetics in plant sciences (Walls et al., 2012).

The knowledge specifications developed in this research supports the use of standardised ontologies, as they are also defined as semantic networks. However, from a constructivistic approach (sensu Piaget (1952)), the term knowledge scheme is more appropriate than ontology, as our proposed schemes (networks) are non-definite. That is, the knowledge schemes are subject to constant redefinitions caused by the *disequilibrium* with previously assimilated schemes and the *accommodation* of new information. In this sense, the knowledge schemes are always subject to change in light of new evidence (data but also theories), rather than describe an inherent structure of a domain of knowledge, as the philosophical definition of ontology suggests (see (Guarino, 1995; Wielinga et al., 1993) for a similar argument).

1.1 Aim of the thesis

The research in this thesis tries to apply the aboved-mentioned ideas into concrete computational and statistical frameworks. First, by representing biodiversity, geospatial and environmental data as a unified connected network of knowledge schemes, called *knowledge graph*; and second, using these schemes to test ecological hypotheses using formal statistical methods. The thesis is a compendium of three research articles. The basic concepts for understanding the thesis are explained in this section, leaving further explanations, proofs, computer programs and discussions to the following research chapters.

The central assumption in the presented research is that the available recorded evidence of biological occurrences contribute to inform, to some extent, about the ecological mechanisms that drive organisms to occupy a given geographic area. We test the hypothesis that *meaningful* observations, together with appropriate spatial model-based specifications, describe the stochastic processes that generate these observations. Moreover, the characterisation of these processes can help to describe the species' responses to their environment and, ultimately, the presence (and absence) of other species. To formally address this hypothesis we must elaborate precise definitions of concrete statistical and computational specifications and their corresponding implementations. We begin with the definition of *biological diversity* agreed by the United Nations' Commission for Biological Diversity :

> *"The variability among living organisms from all sources, including, inter-alia, terrestrial, marine, and other aquatic ecosystems, and the ecological complexes of which they are part: this includes diversity within species, between species and of ecosystems"*
>
> – UN-CBD,(1992)

A biodiversity *occurrence* is, therefore, any recorded observation of an organism on a given time and location frame. In this thesis, an occurrence is considered to be a point in space-time that has associated attributes like: unique id, species name and collection id. By definition, an occurrence represents *only* the presence of a single organism. We can generalise the concept of species to include other taxonomic ranges (also called levels) like: genus, family, order, class, phylum (or division in case of plants) and kingdom. In this thesis, the term *taxon* (plural *taxa*) is used to denote an arbitrary taxonomic group. As such, an occurrence is a type of *presence-only* datum that has information about, when (t) and where (x, y) a member of a certain species, or any other taxon, (s) has been recorded.

The phrase *available recorded evidence* of biodiversity data implies an arbitrary collection of occurrences composed of independent surveys taken at different times, locations and sampling designs. The occurrence data is represented as a node in the knowledge graph with spatio-temporal coordinates. As said before, occurrence data only give information about presences of taxa. However, the statistical framework requires presence and absence observations for total identifiability of the model (Ward et al., 2009). One of the most fundamental concepts of the proposed statistical framework is the use of knowledge schemes to model absences using the informative occurrences (i.e. presence-only observations).

1.2 Structure of the thesis

This thesis is structured in two parts. Part I describes the knowledge engine, that is, the theoretical specification and software implementation of the knowledge graph for representing knowledge schemes. Part II describes the statistical frameworks for modelling the species distributions using the observations extracted from the knowledge schemes implemented by the knowledge engine. The frameworks use these knowledge schemes

to model absences and informative background data, necessary elements to identify the presence-only models.

1.2.1 Part I: the knowledge system

The knowledge graph is built on simple data structures represented as nodes. The nodes are linked with each other using relationships. These relationships have meaning and they link different classes of nodes. For example, a node of the type *occurrence* is linked to a node of the type *cell* to represent the concept of an *occurrence is located in cell*. As such, nodes represent atomic data-objects and these objects are linked through semantic predicates. In computational terms, a knowledge scheme is the abstraction of a pattern through the knowledge graph. This abstraction, also called a *graph traversal*, selects and visits nodes according to a given specification. A knowledge scheme is, therefore, a computational *type* of a conceptual model that gives semantic meaning to a graph traversal. It abstracts a scientific concept (e.g *ecological niche*) into a network of nodes and predicates extracted from the knowledge graph (i.e. the universe of discourse). In this way, these schemes are capable of integrating data generated (acquired) independently into a complex structure that has scientific meaning, properties and actions. An example of this is the generation of a local taxonomic tree constrained to the area determined by a cell. The graph traversal receives as input the cell node. It selects the associated occurrences located within the polygon determined by the chosen cell. The procedure starts in the occurrence node and traverses the knowledge graph following the relationship `Is_member_of` until it reaches the root of the tree of life. The resulting selection corresponds to the taxonomic tree constrained to the occurrences located within the given cell.

Chapter 2: The knowledge engine

The full description of the engine together with a tutorial on how to use the software for basic analysis and data queries is located in Chapter 2. It presents how taxonomic and topological predicates used to link biodiversity and environmental data. That is, the graph traversals are specified to match heterogeneous data to create complex data structures with taxonomic and spatio-temporal attributes. Examples of these schemes are the taxonomic tree defined within a geographic area, the selection of neighbouring areas given a central region, or the complementary groups given a set of taxa of interest. These schemes were later used in a Bayesian statistical framework to model the distribution of single to multiple taxa of interest (part II). This chapter was published as a research article format in the journal GigaScience with the following reference: GigaScience,Volume 9, Issue 5, May 2020,giaa039 DOI:10.1093/gigascience/giaa039 See Escamilla Molgora et al. (2020a).

1.2.2 Part II: The statistical framework

The second part of the this thesis addresses the problem of modelling the spatial distribution of single and multiple taxa (e.g. species). The main assumption of this framework is that the occurrence generating process is the joint effect of two processes, one of ecological interest, driven by environmental explanatory variables (described as *ecological suitability*) and the other related to the sampling effort, driven by anthropological explanatory variables. As such, for an occurrence *event* to happen, three other events need to occur simultaneously:

1. taxon s is present in an area L, where $(x, y) \in L$,
2. Someone has gone to location L during time t and recorded the occurrence, and
3. the organism has been recognised as a member of taxon s.

Events 1 and 2 constitute fundamental concepts of the presented statistical models. Further explanations and discussions of these two events are addressed in chapters 3 and 4, when modelling the *presence-only* hierarchical logistic regression. Event 3 implies that the classification process for the occurrence is perfect (i.e. there is no error in the classification of an occurrence). We are aware that this is in practice impossible, as the natural system of classification is subject to constant updates (De Queiroz and Gauthier, 1990; Futuyma, 2005; Woese et al., 1990). This assumption may not be valid for certain groups of organisms. However, the error decreases with respect to the taxonomic range. It is highest at the species rank and lowest at phylum. Nevertheless, for the purposes of this research, the effect of incorrect classification of taxa is not relevant. Unless it is said explicitly, we assume that this error does not exist.

The spatial random effect is another important component of the framework. Here the hypothesis was that the effect of other species interactions in the occurrence of the target species, can be statistically accounted with a spatial random effect. The test of this hypothesis and its associated modelling framework is contained in chapter 3 while chapter 4 explores a generalisation of this framework for multiple species. In both cases, the spatial effect serves to exchange information between the ecological suitability process and the sampling effort process.

Chapter 3

This chapter presents a modelling framework with three different spatial configurations for accounting the joint effect of the ecological suitability process and the sampling effort. To test the above-mentioned hypothesis, a comparison was done between the spatial models and maximum entropy (MaxEnt) algorithm (Phillips et al., 2006a), a popular method for predicting species occurrences using presence-only data. The results showed that the proposed spatial framework is superior in terms of its *goodness of fit* measured by the

receiver operator characteristic (ROC) curve and its area under the curve (AUC) (Fielding and Bell, 1997). This chapter was submitted to the journal Ecography in September 2020 with positive feedback from the reviewers. At the moment (December 2020) the article is under revision.

Chapter 4

Chapter 4 presents a generalisation of the modelling framework from chapter 3 to allow multiple taxa. An important difference is that, instead of using an arbitrary chosen informative sample as proxy for the *sampling effort*, the natural taxonomic classification of life was used to define an *intrinsic* structure from which to model the absences. This is based on previous results on joint species distribution models, where common species are often used to inform about the presence of less frequent or observed species (Ferrier and Guisan, 2006).

The proposed multiple species model is a novel and unique multilevel logistic hierarchical specification for modelling joint species distributions. It was implemented in the STAN programming language (Carpenter et al., 2017) and tested using simulated data. Additionally, a case study was presented showing results of ecological importance. This chapter was submitted for publication in the journal *Methods in Ecology and Evolution* in December 2020. The article is currently under review.

Chapters 5

This chapter discusses the general implications and limitations of this research, as well as possible research lines for continuing in future.

Chapter 6

This chapter contains the general conclusions of the thesis.

Part I

The knowledge engine

CHAPTER 2

BIOSPYTIAL: SPATIAL GRAPH-BASED COMPUTING FOR ECOLOGICAL BIG DATA

Chapter published in:

GigaScience,Volume 9, Issue 5, May 2020,giaa039

DOI:10.1093/gigascience/giaa039

Published: 11 May 2020

Received: 19 July 2019

Accepted: 02 April 2020

Biospytial: spatial graph-based computing for ecological big data

Juan M. Escamilla Molgora[a,b,1,*], Luigi Sedda[b,2], Peter M. Atkinson[c,3]

[a] *Lancaster Environment Center, Lancaster University, Lancaster LA14YQ, UK*
[b] *Centre for Health Informatics, Computing and Statistics (CHICAS), Lancaster Medical School, Faculty of Health and Medicine, Lancaster University, Lancaster LA1 4YQ, UK*
[c] *Faculty of Science and Technology, Lancaster University, Lancaster LA1 4YR, UK*
[d] *Lancaster Medical School, Faculty of Health and Medicine, Lancaster University, Lancaster LA1 4YQ, UK*

Abstract

Background The exponential accumulation of environmental and ecological data together with the adoption of open data initiatives bring opportunities and challenges for integrating and synthesising relevant knowledge that need to be addressed, given the ongoing environmental crises.

Findings Here we present Biospytial, a modular open source knowledge engine designed to import, organise, analyse and visualise big spatial ecological datasets using the power of graph theory. The engine uses a hybrid graph-relational approach to store and access information. A graph data structure uses linkage relationships to build semantic structures (objects and predicates) to answer scientific questions represented as complex data structures stored in a graph database, while tabular and geospatial data are stored in an efficient spatial relational database management system. We provide an application using information on species occurrences, their taxonomic classification and climatic datasets. With this, we built a knowledge graph of the Tree of Life embedded in an environmental and geographical grid to perform an analysis on threatened species co-occurring with jaguars (*Panthera onca*) across all Mexico.

Conclusion The Biospytial approach reduces the complexity of joining datasets using multiple tabular relations, while its scalable design eases the problem of merging datasets from different sources. Its modular design makes it possible to run and distribute several instances simultaneously, allowing fast and efficient handling of big and complex ecological datasets. The provided example demonstrates the engine's capabilities in performing basic graph (taxonomic trees) manipulation, analysis and visualizations of taxonomic groups co-occurring in space. The example shows potential avenues for performing novel ecological analyses, biodiversity syntheses and species distribution models aided by the interconnected network of taxonomic and spatial relationships.

Keywords: spatial data infrastructure, biodiversity informatics, ecological knowledge engine, big ecological data, open science

*Corresponding author
Email addresses: j.escamillamolgora@lancaster.ac.uk (Juan M. Escamilla Molgora), l.sedda@lancaster.ac.uk (Luigi Sedda), pma@lancaster.ac.uk (Peter M. Atkinson)
[1] https://orcid.org/0000-0002-3682-9828
[2] https://orcid.org/0000-0002-9271-6596
[3] https://orcid.org/0000-0002-5489-6880

2.1 Introduction

The IT revolution has created the opportunity to compute, store and transfer massive amounts of information. It is estimated that the volume of all digital information will surpass 175 Zettabytes (ZB) (1 ZB = 10^{21} bytes) by 2020 (Reinsel et al., 2018). In addition, the growth in data follows an exponential curve that doubles in volume every two years ((Kurzweil, 2004), (Hilbert and López, 2011) and (Gantz and Reinsel, 2011)). Moreover, this expansion in data production has occurred in all human activities, including the environmental sciences. Novel approaches for measuring natural processes are being applied, adding more reliable and diverse data, and environmental measurements cover a wide range of spatial and temporal scales ranging, for example, from long-term ecological experimental plots (Weigelt et al., 2010), (Borer et al., 2014) to near-real time imagery from Earth observation satellites systems like NASA's *Joint Polar Satellite System* (National Aeronautics and Space Administration et al., 2020) and ESA's *Copernicus* programme (European Space Agency, 2014). This IT era is opening new opportunities for greater understanding of nature. For example, pervasive Internet connectivity has made possible the transfer of data across large distances in a short time; and the multifunctional capabilities of mobile and *smart* devices has enabled the management and deployment of collaborative surveys at low marginal costs. Geospatial sciences have benefited in particular. Methodologies for collecting, annotating and curating these new sources of spatial data have been proposed by (Goodchild, 2007), (Heipke, 2010) and (Kamel Boulos et al., 2011) under the term *citizen-science*; where data are collectively assembled by a community of enthusiasts and volunteers. Some iconic examples of these (*crowd-based*) platforms are OpenStreetMap (OpenStreetMap Contributors, 2019) for geographic maps and the *Global Biodiversity Information Facility* (GBIF), an international consortium of

research and governmental institutions that gathers and publishes information of all types of biodiversity occurrences (GBIF Secretariat, 2015).

The exponential growth of data imposes new challenges for storage, access, integration and analysis. In recent years, new theoretical methods and technologies are being developed to tackle these problems. The name *Big Data* is now an umbrella term for methods dealing with huge, complex, and heterogeneous datasets that cannot be handled with traditional methods. See (Chen et al., 2014) and (Mikalef et al., 2018) for a review of the field and (Li et al., 2016) for theoretical and practical challenges involving big geospatial data.

A fundamental goal in ecology is the understanding of the relationships between living beings and the environment. A requirement to achieve this goal is the integration of independent studies and measurements to validate hypotheses on potential causal relations. To test the existence of these causalities, a substantial number of inputs in terms of theory, methods and data is needed. Moreover, reliable, reproducible, and easy to access methods are especially important given the urgency in addressing ongoing environmental crises (e.g. rapid ecosystem degradation, global climate change, accelerated extinctions and biodiversity loss) (Stocker et al., 2013),(Brondizio et al., 2019). Ecology is thus adapting rapidly to these critical challenges and is starting to adopt and develop novel theoretical and computational methods to answer a central problem: *How to synthesise and integrate ecological theory with big ecological data?* Answering this question requires an interdisciplinary approach that touches many fields, including: theoretical ecology, mathematical modelling, statistics, computer science and information sciences. For example, (Loreau, 2010) proposed a conceptual framework for integrating ecological theory by centering evolution as the link to unify ecology; and (Pavoine and Bonsall, 2011) proposed a semantic and mathematical formalization for unifying traits, species and phylogenetic diversity. The two approaches exemplify how evolutionary (ancestry) relationships between biologi-

cal objects constitute a solid base to unify distant branches of ecology. From a statistical perspective, meta-analysis has been effective in synthesizing research evidence across independent studies, including unveiling general relations through a statistically sound framework(Koricheva et al., 2013).

Geospatial data constitute a crucial component for data fusion and harmonization; see (Wiemann and Bernard, 2016) for a review of methods for heterogeneous spatial big data fusion, and (Wang et al., 2016) in order to remove bias by using spatial data stratification methods. A clear example of geospatial data fusion is the building of Essential Biodiversity Variables (EBVs) to identify biodiversity and ecosystem change (Pereira et al., 2010). EBVs constitute a minimal set of critical variables aimed to standardize and harmonize global biodiversity variables. Originally proposed by the Group on Earth Observations Biodiversity Observation Network (GEO BON) to assess biodiversity change globally (Navarro et al., 2017); EBVs are now being used to predict global species distributions and potential scenarios for policy options (Pereira et al., 2013). EBVs integrate data in a standardised framework that describes spatial, temporal and biological organization (Schmeller et al., 2017). Recently, methodologies for building EBVs are drawing the attention of interdisciplinary research for reliability and data quality (Kissling et al., 2018). System designs and infrastructures for integrating heterogeneous big ecological data are emerging. Examples of these are the *citizen-based* bird observation network (eBird (Sullivan et al., 2009)), the TRY database for plant traits (Kattge et al., 2011), the PREDICTS project (Projecting Responses of Ecological Diversity In Changing Terrestrial Systems) (Hudson et al., 2014) and the Botanical Information and Ecology Network (Enquist et al., 2016). Despite the data heterogeneity and biased information against real absences (a consequence of opportunistic sampling), these types of infrastructures are able to collect sufficient quantities of data to perform statistical inference ((Hartig et al., 2012) and (Kelling et al., 2015)). The use of high performance computational technologies with novel statistical methods for

representing and modelling big ecological data can provide deeper understanding of biodiversity evolution and its dynamics in a changing world (La Salle et al., 2016) , (Navarro et al., 2017) and (Schmeller et al., 2017). Moreover, its implications can be extended to other branches of ecology and Earth sciences. For example, a process-based approach by (Scheiter et al., 2013) showed how community assemblages can be integrated into dynamic vegetation models to increase the precision of climatic and Earth System models.

From a technical perspective, environmental and ecological data often come in matrix form such that they can be stored and analysed efficiently with a relational database management systems (RDBMS) or other tabular data structure. RDBMS are reliable and sophisticated tools. An important feature is the possibility to extend their functionality with programming languages such as: C, Java, Python, R-Cran, etc.. This allows the combined use of an efficient data management system with a broad range of statistical libraries and programming methodologies. An example of this is the integration of spatial analysis tools into the RDBMS through the Postgis project (Ramsey et al., 2018); a set of compiled functions written in the Postgresql Procedural Language (PostgresPL) that interfaces with high level geospatial libraries (e.g. (GDAL/OGR Contributors, 2018), (Geometry Engine Open Source (Contributors), 2019) and (PROJ Contributors, 2019)). Postgis adds GIS capabilities to the database engine, giving superior performance for querying information with geometric and topological features in space.

Integrating large datasets using only relational methods is computationally intensive. For example, matching data by a common feature involves the definition of join clauses plus computing the joined lookup between the pair of tables. The resulting product is often stored in volatile memory, a limiting factor when integrating large datasets. In a typical database design, table indices cost $O(log(n))$ in time, where $O(\cdot)$ is the classic *Big O*, a measure of computational complexity and n the size of the input dataset. A query involving multiple joins (from multiple data tables) can involve reverse and recursive

lookups, that can increase the load from $O(n)$ to $O(n^k)$, where k is the number of data tables to join. Although this issue may be addressed with database design techniques such as normalization (Harrington, 2009) or caching (Altinel et al., 2002), the solution likely obfuscates the comprehension of the relational schema by adding unintuitive tables and other auxiliary information. It also requires a learning curve and expertise for implementation as well as increasing complexity when more datasets are added.

Data structures based on direct acyclic graphs (DAGs) are advantageous in relation to the above approaches. Traversing a relationship in a graph database has constant cost ($O(1)$) (Celko, 2014) if the relations are defined explicitly for every node. Whenever a new dataset is added, a new link can be created to relate it with an existing record. Graph databases, however, are not as efficient at processing geospatial queries or handling simultaneous queries (Vicknair et al., 2010). In this sense, hybrid data management systems, capable of handling both paradigms (relational tables and DAGs), were proposed to overcome the limitations of both systems. However, to the best of our knowledge, these proposals have not been yet implemented (Grund et al., 2013), their code is closed (van Iersel et al., 2010) or their scope is not suited for environmental and spatial datasets, as is the case of the Reactome Database (Fabregat et al., 2018).

In this paper we propose an implementation of an open source knowledge engine (i.e. a hybrid database system) that stores, accesses and processes geospatial and temporal information, to integrate, analyse and visualise heterogeneous environmental, EBVs and big ecological data. The engine, named *Biospytial* (composed by the words *biodiversity, Python* and spatial and pronounced *Biospatial*) incorporates semantic relations that integrate data in a web of semantic knowledge able to represent complex graph (network) data structures.

Biospytial can be considered a component of traditional Spatial Data Infrastructure (SDI) because we simplify access and analysis of big datasets while satisfying the need of

producing information for scientists and policy makers, among others (Hendriks et al., 2012). This is possible due to the engine's capability to identify intrinsic and extrinsic relationships within environmental and socio-economic processes. Therefore, the developed engine is aimed to serve SDI-based decision making frameworks, as for example the European project INSPIRE.

The engine serves as a multi-purpose platform for modelling complex and heterogeneous data relationships using the power of graph theory. The current implementation uses the occurrences data from the GBIF and their updated systematic classification (GBIF Secretariat, 2017) to build the acyclic graph of the *Tree of Life*. To exemplify the geospatial capabilities, some EBVs like: mean monthly temperature, elevation and mean monthly precipitation are also included in the engine. The paper is structured as follows: The specification and general description of the engine is given in section 2. Section 3 proposes a methodology and software implementation for accessing biodiversity records arranged in a taxonomic tree. The graph of the *Tree of Life* is explained with examples for traversing and extracting spatial and taxonomic sub-networks. Section 4 explores the capabilities of the engine with a practical demonstration. It shows the syntax and discusses ways to interpret and traverse the knowledge graph. Finally, section 5 includes general conclusions, and future research directions.

2.2 An *Open Source* graph-based engine for geospatial analysis

The engine is able to import, organise, analyse and visualise big ecological datasets using the power of graph theory. It performs geospatial and temporal computations to synthesise information in different forms. The data can be queried and aggregated according to customised specifications defined by structural patterns called *graph traversals* (Ro-

driguez, 2015). The software has been developed with object-relational and object-graph mappings (ORM and OGM, respectively) that use the object-oriented paradigm to abstract interrelated data into class instances (Celko, 2014; Juneau, 2018). In this sense, every record is represented as an instance of a certain class with its attributes mapped one-to-one to entries in a particular table (if it is stored in a relational database) or in a key:value hash table (if it is stored in a graph-based database). This approach allows the building of complex and persistent data structures that can represent different aspects of the knowledge base. It also allows the assemblage of automatic methods for exploring, filtering, aggregating and storing information.

2.2.1 System architecture

The engine is composed of three interconnected modules : i) A *Relational Geoprocessing Unit* (RGU), ii) the *Biospytial Computing Engine* (BCE) and iii) a *Graph Storage and Processing Unit* (GSPU) (see figure 2.1). Each module is arranged in virtual containers isolated as standalone applications (Docker Inc., 2019) running a common Linux image (Debian 8) as the base operating system. The virtual container technology creates a common environment for each module disregarding the complications of working with heterogeneous computer infrastructures (Pahl and Lee, 2015). Its design allows the replication of several instances of the same module in a single computer or in a distributed network. Containerised applications are easier to replicate and migrate compared to large data volumes and databases, which often involve resource intensive tasks in terms of energy, computing, network bandwidth and management. The idea behind containerization is: *move the processes not the data* and especially in the geospatial context, to perform spatial analysis where the data is located.

The Relational Geoprocessing Unit (RGU)

The RGU module undertakes the storage and raster-vector processing. It relies on high-level abstractions that represent geospatial data stored in relational tables. The supported geometric features are (multi)points, (multi)lines, (multi)polygons and multiple band raster data. It features a fully operational Postgresql (9.4.9) server (port: 5241) with geospatial extension (Postgis 2.3.1)(Ramsey et al., 2018) and libraries for handling geospatial data (GDAL, OGR 1.10.1)(GDAL/OGR Contributors, 2018), transformation between different geographic projections (PROJ 4.8, (PROJ Contributors, 2019)), and computation of geometric operations (GEOS 3.6)(Geometry Engine Open Source (Contributors), 2019) (figure 2.1 b). The RGU image can be downloaded from:
https://hub.docker.com/r/molgor/postgis_biospytial/

The Graph Storage and Processing Unit (GSPU)

This module hosts a graph database that stores data on nodes and their relations in a network structure called the knowledge-base (figure 2.1 a). The graph database system is an instance of Neo4J (3.1.3), an open source ACID-compliant transactional database management system with native graph storage and processing (Celko, 2014). It includes a web-based interface located in http://<urlofhost>:7474. The interface allows the inspection and visualisation of queries (subgraphs) using the Cypher interpreter (a No-SQL type declarative language for interrogating graph databases). The module also includes a plugin for spatial and topological lookups[1] and the *Awesome Procedures on Cypher* (APOC) [2]. ; an extension library with more than 300 procedures for data integration, graph algorithms or format conversion procedures. The GSPU image can be downloaded from: https://hub.docker.com/r/molgor/neo4j_biospytial/.

[1] https://neo4j-contrib.github.io/spatial/0.24-neo4j-3.1/index.html
[2] https://neo4j-contrib.github.io/neo4j-apoc-procedures/index31.html

The Biospytial Computing Engine (BCE)

This module provides the interface and processing toolbox for accessing, exploring and analysing data structures through the *Object Mapping* design. The container hosts a virtual environment and an *Anaconda* package manager (ANACONDA, 2016) that includes all the dependencies required by the engine. The core code of the engine is contained in a new Python package called *Biospytial*[3] (figure 2.1 c). The engine structure includes a `drivers` module to communicate with the graph database, the modules for accessing each dataset in the relational database; the module for graph traversals, data ingestion, griding systems, vector sketching, Jupyter notebooks; and external plugins like `spystats`, a Python port of GeoR (Diggle et al., 2002). The image can be downloaded from: https://hub.docker.com/r/molgor/biospytial/

Other features

Scalable The implementation includes scripts for automating the engine's deployment in a single host or in cluster mode. This mode provides a granular configuration for the allocation of resources and services in a distributed manner. For example, The BCE module can be hosted in a computer with high performance architectures or multiprocessing (e.g. MPI) capabilities.

Message broker The engine includes a messaging service (Redis (Labs, 2012)) that delivers information between the different components. It also serves as an in-memory data structure storage and message broker. The storage is useful for interchanging data between different platforms and languages. For example, it allows export of the results into intermediary files (e.g. CSV or DBF) for use in other software (e.g. (Team and R Development Core Team, 2016) and (Hornik, 2012)).

[3]https://github.com/molgor/biospytial

Open Source - Open Contributions The software used in all the modules has been released with Open Source and Free Software licenses which allow users to reproduce, modify and publish their research source code. The engine was developed using best practices for scientific computing (Wilson et al., 2014a),data transparency and reproducibility (Perkel, 2018).

Access to the engine

There are two ways of accessing the engine. One is through a command-line interpreter based on the `iPython` console (Perez et al., 2007). The other is with an online Jupyter notebook server (Kluyver et al., 2016) (localhost:8888). The Jupyter notebook is a web-based interactive Python interpreter that renders MarkDown documents, plots and images in the browser . Analysts can create files in a *notebook* format (.ipdb) and share the results on-line. Peers can visit the notebook's url, read the document, run the code, replicate the analysis, access the variables, import other libraries, modify the analysis and export it into different formats (e.g. PDF, Latex or HTML).

2.2.2 Knowledge representation

The engine uses two database paradigms to store and represent data: a relational system with tables connected by primary and foreign keys and directed acyclic graphs (DAGs) where the data are stored as nodes (with associated attributes) and edges representing relations between nodes. Each node can belong to one or many classes. In our implementation, the relationships are semantic phrases that refer to location (e.g. *"IS IN"*), ancestry (*"IS PARENT OF"*) or topological features (*"IS CONTAINED IN"* or *"IS NEIGHBOUR OF"*). Thus, the engine uses explicit semantic relations between nodes to build a network of semantic information. The union of all these relationships is what we call *knowledge graph.*

Table 2.1 Principal software components of the Biospytial Knowledge Engine System

Software name	Version	Description
Biospytial Computing Unit	Debian GNU/Linux 8.6	Container OS image
Conda	4.3.30	Package manager optimized for Data Science
Python	2.7.11	Programming language (scheduled update for v.3.x)
R-base	3.2	Language and software environment for statistical computing
Jupyter	1.0.0	Interactive web application for reproducible computational workflows
Scipy	1.01	Python library for numerical and scientific computation
Pandas	0.19	Python library for data structures and data analysis
Geopandas	0.3	Extension of Pandas to support geospatial data
GDAL	2.1	Library for converting and processing geospatial data
Shapely	1.5.16	Python library for manipulation and analysis of geometric objects in the Cartesian plane
Django	1.8.4	ORM, web framework and standalone server
Py2neo	3.11	A client python library and toolkit for working with Neo4j
Pymc3	3.4,1	A Python based Probabilistic Programming Framework
Patsy	0.4.1	A Python library for describing statistical models
Relational Geoprocessing Unit	Debian GNU/Linux 8.6	Container OS image
Postgresql	9.4.9	Relational database management system
Postgis	2.3	Spatial extension for Postgresql
GDAL	1.10.1	Library for converting and processing geospatial data
GEOS	3.6	Geometric and Topological library
Proj4	4.8	Coordinate transformation software
Graph Stor. and Process. Unit	Alpine Linux 3.5	Container OS image
OpenJDK	IcedTea 3.3	Open Source Java compiler and virtual machine
Neo4J	3.1.3 (C.E)	Graph Database Management System
APOC	3.1.3	Utilities, graph algorithms and common procedures for Neo4j
Message Broker	Redis 5.0.3	a Key-value data structure store

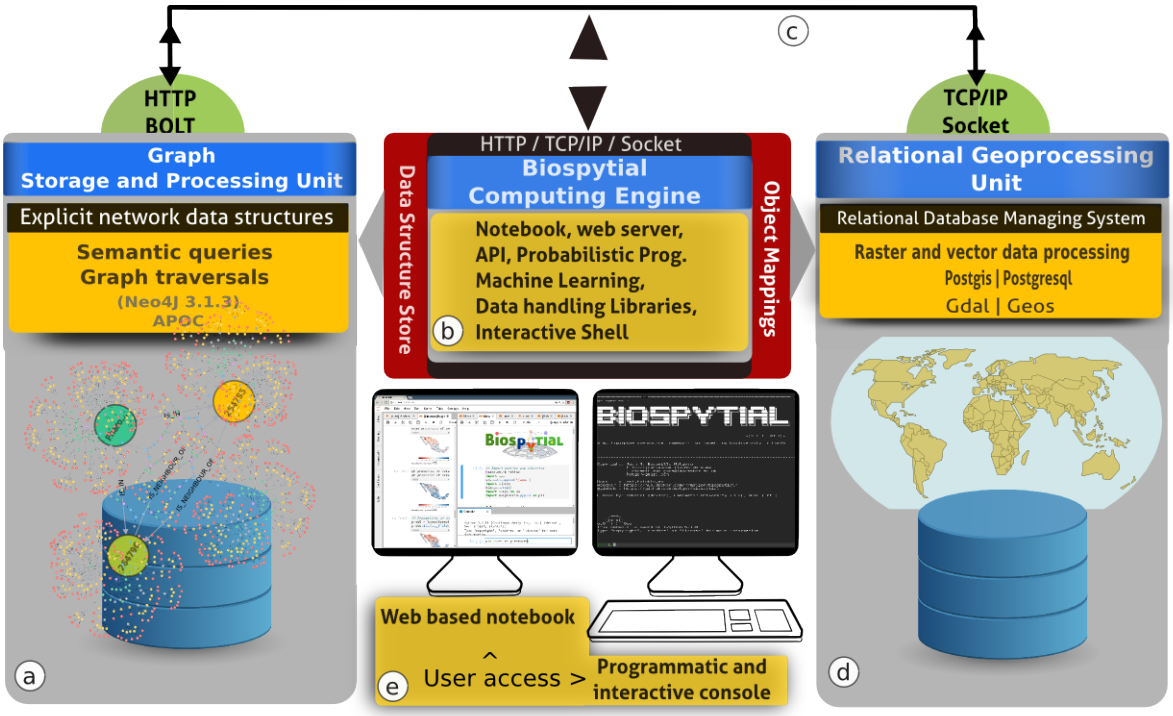


Fig. 2.1 The Biospytial System with the three interconnected modules. a) The GSPU, where semantic queries and graph traversals take place. b) The BCE, where object mappings, web services and the modelling framework takes place. It includes several libraries for performing exploratory analysis as well as Bayesian statistical inference and prediction using the probabilistic programming language: PYMC3; c) All the components can be allocated in the cloud and are connected using virtual and physical networks. d) The RGU, where the geoprocessing and spatial indexing occurs, storing efficiently any raster and vector data sources. e) Interactive access is possible in two ways: using an online web notebook (Jupyter) or an interactive console (iPython).

The event of a species s being recorded at location l can be represented as a node of the class *Species* connected to a node l of class *Cell* using the relation *IS_IN*. The *Cell* nodes are contained in a regular lattice (grid) and are instantiated by a class that implements a geospatial type defined by a polygon that acts as a geometric border. As an example, figure 2.2 shows this diagram for the bird family of quetzales (Trogonidae) found in southeast Mexico. The node in red represents the species: *Pharomachrus mocinno*. The nodes in blue are two *Cell* types that associate the locations where *P.mocinno* was found. The arrows indicate the directional relationships between the nodes. The graph database allows easy manipulation of these nodes, their relations and combinations. At the same time, the

selected pattern can be filtered by chosen attribute values to generate customized design matrices.

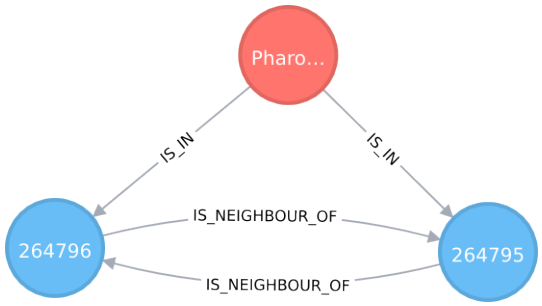


Fig. 2.2 The graph showing the connection between a *Species* node and two *Cell* nodes. Here: the species is *Pharomachrus mocinno* (Quetzal) and the number shown in each Cell node is its respective ID number. This is an actual visualisation taken from data stored in our Knowledge Graph.

2.2.3 Integrating data with graph structures and object mappings

The *Object Mapping* approach serves to communicate different database management systems (relational or graph-based). A high level Python-based Object Relational Mapping (ORM) library (`Django` (dja, 2018)) was used to communicate with the RDBMS and the other components of the engine. It includes a high level interface to translate sentences from the SQL declarative language into method calls from the object-oriented paradigm. Vector and raster operations are possible via the Open Source Geographic Information System (OSGIS) for Postgresql (Postgis (Ramsey et al., 2018)). Currently, all the spatial and tabular data are stored in the RDBMS.

The *object mapping* on the graph database system is achieved with `py2neo`, a client library and toolkit for communicating with the Neo4j database management system [4] within the Python programming language(Small, 2017). Topological information like neighbouring cells and nodes contained within cells are stored as semantic relations.

[4] https://neo4j.com

Some preprocessed information is stored in the knowledge graph. This includes some parameter estimates, aggregated data, summary statistics and associated raster metadata.

The procedure for adding data into the engine varies according to the data format (tables or linked data) and requires a new class to be created. The class is responsible for accessing and managing data in both database systems. It includes specifications for storage, conversion between formats and analysis. A simple implementation would include: the name and type of the attributes; the name of the table (for the case of RDBMS), the node type and incoming and outgoing relations between nodes (for graph-based datasets). Detailed information on all these procedures is given in the supplementary materials.

2.2.4 Graph Traversals

As explained above, the *Knowledge Graph* is the totality of nodes and relationships stored in the database. Each node represents a type (defined by a class) of data or a more abstract concept that generalises certain sets of data. Each node has associated edges to other nodes, as well as a list of attributes. In the example given in figure 2.2, the node is of type *Species* and one of its attributes is *name* with the associated value *P.mocinno*.

The graph engine can search and extract information from the knowledge graph using recursive rules based on semantic predicates. Typically, the search selects one, or several, nodes and continues visiting (traversing) other connected nodes that match the specified criteria until the relationship is exhausted or a depth threshold has been reached. The resulting selection of relationships and nodes is a subgraph of the knowledge graph. We call this structure a *pattern* and the set of rules that select a pattern is a *graph traversal*.

Graph traversals can be translated into data matrices that can be analysed within the scope of model-based geostatistics (Diggle et al., 2002) or areal unit modelling in lattice systems using Gaussian Markov Random Fields (Besag, 1974; Besag et al., 1991;

Rue and Held, 2005). Also, they can be analysed with network theory to answer questions about resilience, connectedness, modularity or invariants across scales. The objects are compatible with the open source libraries for statistical inference and network analysis. Libraries already included in the engine are: NetworkX (Hagberg et al., 2008), StatsModels (Seabold and Perktold, 2010) and PyMC3 (Salvatier et al., 2016).

Complex queries

Our implementation enforces the use of *lazy evaluations*, in which the evaluation of an expression is delayed until the value is needed and not directly upon the instantiation (Hudak and Paul, 1989). This helps in the creation of data primitives that can be composed into higher level graph traversals without the need to load in all the data. The design allows the request on demand of partial evaluations for a given traversal. This abstraction helps to explore, design and automate the discovery of relevant patterns and structures. A concrete example of this design is showed in section 2.3 with the analysis of local taxonomic trees, when the tree object is instantiated, it exists only as an abstract data container with no data requested to the database. As such, if an analyst is interested in studying the different species of bats (*Order:Chiroptera*) within this tree, she will need only to consider the descendant (children) nodes of the node *Chiroptera* of type *Order* (See section 2.5.1 for a practical example).

Some traversals are exclusive of certain node classes and, therefore, have associated special methods. This is the case for nodes of type *Cell* which include a method for extracting neighbouring cells. Figure 2.3 shows an example of this where a selection of cells was obtained first by requesting all the occurrences of the Family *Culicidae* and then traversing through the associated cells and their corresponding neighbours using the method getNeighbouringCells() twice.

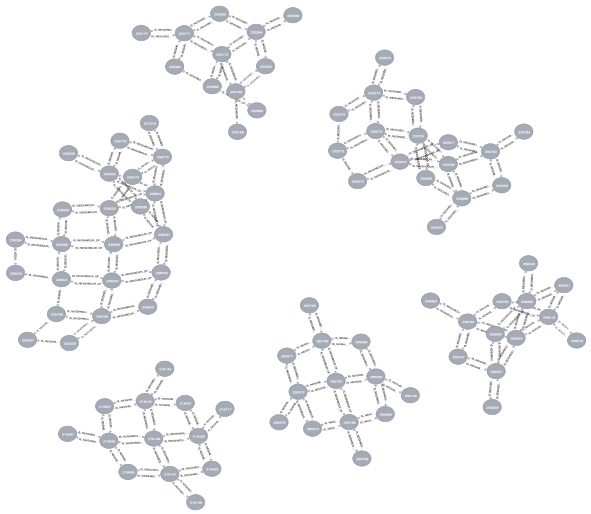


Fig. 2.3 A subgraph from the Knowledge Engine that shows the second order degree of neighbouring cells where at least one occurrence of any type of mosquito (family *Culicidae*) was registered. This query exemplifies the use of recursive lookups. In this case the relationship "IS_NEIGHBOUR_OF" is traversed twice.

2.2.5 Geospatial management and processing

The engine supports and processes geospatial information using the GDAL/OGR library (GDAL/OGR Contributors, 2018).The default Coordinate Reference System (CRS) is the WGS84 with geographic coordinates. However, it is possible to use and reproject the data into any other CSR. This feature is supported by the *Proj4* library (PROJ Contributors, 2019). See section 2.5.8 for a concrete example of this.

Vector data

Vector data are represented with tabular data structures. These tables should include the following information: at least one column with a unique identifier (id) for each record, one column for each type of feature, and at least one geographic column to represent the geometric shape of each record. The available geometric types are: points, multiple points, polylines, multiple polylines, polygons and multiple polygons. Each type of dataset corresponds to both a vector layer and a table in the RDBMS. A mapping between the

table structure and the engine needs to be created in the same way as described in section 2.2.3. For large datasets the engine uses indexing methods for optimal performance on accessing and querying the data. Additional information is provided in the supplementary material. 2.14.4

Raster data

Raster data are represented as a table stored in the RDBMS together with its corresponding metadata. The table has three columns: a primary key (id); a Binary Large Object (BLOB) data type (encoding a stack of matrices) that represent a multiband image; and a reference to a file where the metadata is stored. The metadata includes: projection type, affine parameters, datatype for entries (binary, integer, float) and other information related to provenance.

Ingesting raster data into the engine involves two steps, i) the dataset is partitioned in to regular tiles; and ii) each tile is converted into a BLOB string and inserted into the table. Data ingestion scripts can be found in the supplementary materials.2.14.7

The *Object Mapping* design is used to specify the definition of a *RasterData* type and its associated operations. The implemented class includes methods for clipping, downscaling, aggregating, exporting to image formats (Geotif and PNG), visualising, intersecting vector data, extracting metadata and conversion to arrays. An extended class for Digital Elevation Models (DEM) is also implemented to generate *on the fly* aspect, slope and shaded relief (figure 2.4), without requiring the datasets (derived DEM products) to be stored directly in memory.

On instantiation, a *RasterData* object requires the definition of a boundary object passed as argument. This object should be a polygon type `django.gis.contrib.GEOS.Polygon` or a text string defining a polygon in the *Well Known Text* (WKT) format. The resulting selection can be transformed to a dataframe or n-array for statistical modelling. As in the

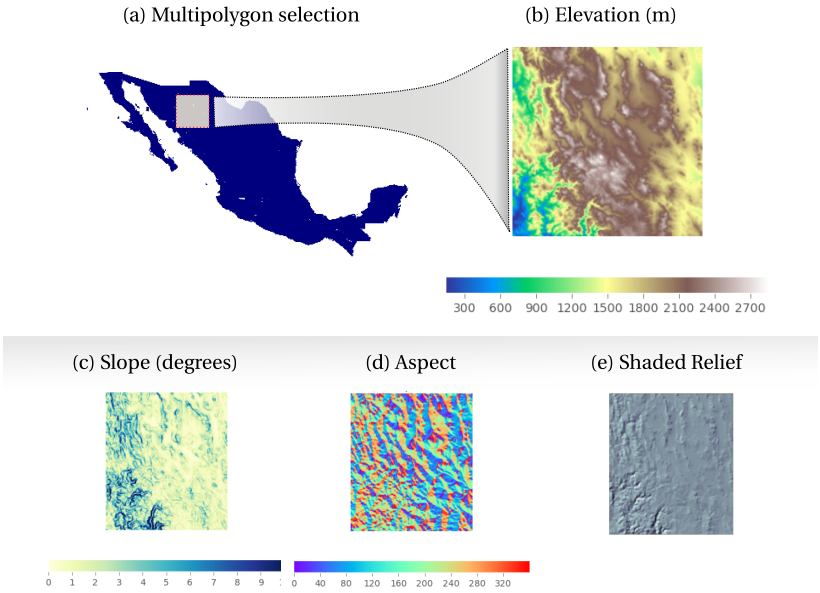


Fig. 2.4 Raster manipulation in the knowledge engine. a) a multipolygon selection corresponding to Mexico, an instance from the class `Country` that maps into the *WorldBorders* dataset. b) An `Elevation` object (class `RasterData`) instantiated with a customized polygon, in this case a subregion of the object Mexico. c), d) and e) are `RasterData` objects derived from the Elevation object. The data and visualisations were produced using the engine's raster API. The code for generating these figures are in supplementary materials.

other data structures, whenever a new raster model is added a new model class should be included (See Supplementary Materials) 2.14.7.

2.3 Using Biospytial to analyse the Tree of Life

In this section we propose a process for integrating spatio-temporal data together with graph traversals to represent tree structures using taxonomic and topological relationships within the knowledge engine. The graph traversals use biodiversity occurrences and environmental data to build complex structures to analyse, visualise and caracterize biological occurrences in different forms. The structure restricted to the taxonomic classification is an acyclic graph (tree) in which all the species occurrences constitute leaf

nodes. We call this structure the *Tree of Life* (ToL) and propose a set of graph traversals to retrieve subsets of the ToL constrained to arbitrary taxonomic groups, spatial regions or temporal ranges. Several class definitions for handling taxonomic trees are implemented, making it possible to automate tasks for unveiling patterns. For a detailed definition of terms and computational structures see suplementary materials II.

2.3.1 Study Area

The study site selected was restricted to Mexico since (i) Mexico is in the list of Megadiverse countries (UNEP/CBD, 2002, 2016); (ii) the territory contains a diverse range of the world's climatic regions (Rzedowski, 2006; Vidal Zepeda, 2005); (iii) the country has policies for publishing open environmental data, including centralized repositories of curated data related to biodiversity, conservation, ecosystem services, land cover and satellite sensor imagery (Sarukhán et al., 2009). The data in the study area provide a concrete example of the engine's capabilities.

2.3.2 Data used

The species occurrences were obtained from a snapshot taken from the global GBIF database on September 2016 (GBIF Secretariat, 2015). The data was filtered to only include the occurrences located within the Mexican borders. The total number of occurrences is 3,242,746 distributed in 54,828 species, 10,781 genera, 2,300 families, 543 orders, 113 classes and 42 phyla, with acquisition years ranging from 1819 to 2016. The taxonomic classification was taken from the GBIF Taxonomy Backbone (GBIF Secretariat, 2017). Each occurrence record has information of species name, location (point coordinates in WGS84) and acquisition date, and represents the observed presence of a certain species, therefore it is only based on presence-only records.

The digital elevation model (DEM) *ETOPO1 1 Arc-Minute Global Relief Model* (Amante and Eakins, 2009) was used at a spatial resolution of 1 minute. Precipitation, temperature (maximum, mean and minimum), solar radiation, wind speed and vapor pressure were obtained from the World Climatic Data *WorldClim* version 2 dataset (Fick and Hijmans, 2017). Each variable is a 12 band raster model with $1\ km^2$ spatial resolution that aggregates monthly average values from the years 1970 to 2000 per month, each band corresponding to each month. The data license for *WorldClim* restricts the redistribution of the data. Therefore, users need to download it and import it into the engine via an automated script:

```
raster_api.bash_raster_tools.migrateToPostgis.bash
```

The engine includes functions for generating grid systems at different spatial resolutions. When the grid system is created it stores a vector representation in the RGU and a network representation in the GSPU. The functions for generating the grid systems are located in the library: `mesh.tools.py`.

2.3.3 Traversals on the Knowledge Graph

The taxonomic tree structure was built with the relation: `IS_PARENT_OF`[5] following the taxonomic classification of the occurrence data and the GBIF *Backbone Taxonomy*(GBIF Secretariat, 2017). Each occurrence had a location attribute matched with environmental data (e.g. elevation or WorldClim) using a *point in polygon* query to the RGU. The spatial structure was built using the relations `IS_IN` and `IS_CONTAINED_IN` in accordance with topological relationships based on the DE-9IM model (Clementini et al., 1993; Egenhofer and Franzosa, 1991) (standardised by (Herrig, 2011)).

The main travesal structure is defined in the *TreeNeo* class. Each instance comprised of an area defined by a spatial polygon and a list of occurrences contained on it. The graph traversal was built recursively using the systematic classification of organisms, starting

[5] Conversely, `Has_Children`

from the GBIF occurrences as leaf nodes and progressing through the parent nodes until the traversal reaches the node with no parent. That is, it begins by the species level and finalises in the root node. On each step, the algorithm fetches the available nodes and group them by their corresponding parent node, generating a set of parent nodes and their associated children. Each of these duples (parent,children) are incorporated into a `LocalTree` object that parses the relevant information into several attributes. This process is applied recursively on each derived parent node of the previous step. The recursion is terminated when the set of parent nodes is empty, generating the desired tree data structure. When this happen the `LocalTree` object is wrapped into a *TreeNeo* instance that extends some additional methods like: manipulating and querying trees, nodes and multiple taxonomic groups as well as graph analysis and exportation to common exchange formats (e.g. graphml, data frames, png, geotif or shapefiles). In addition, all the spatial structures were implemented with Open Source Geospatial(OSGEO) standards (Kemp and Haklay, 2014) to facilitate the migration to other language and platforms. A visualisation of this traversal is showed in figure 2.5.

2.4 Worked examples

This section is a case study for analysing the frequency of coexistent taxonomic groups in all the available datasets restricted to arbitrarily chosen branches of the Tree of Life (ToL) and included in a list of threatened species. These types of analyses are important in conservation studies, where the characterisation of umbrella (or other surrogate) species constitute the basis for protecting a significant number of associated species (Andelman and Fagan, 2000; Drever et al., 2019). To account for this effect, we chose the jaguar (*Panthera onca*) as the species of interest. This due to its preference for undisturbed ecosystems (Thornton et al., 2016) and its wide geographic required range; $181 \pm 4 km^2$ for females and $431 \pm 152 km^2$ males (de la Torre et al., 2017).

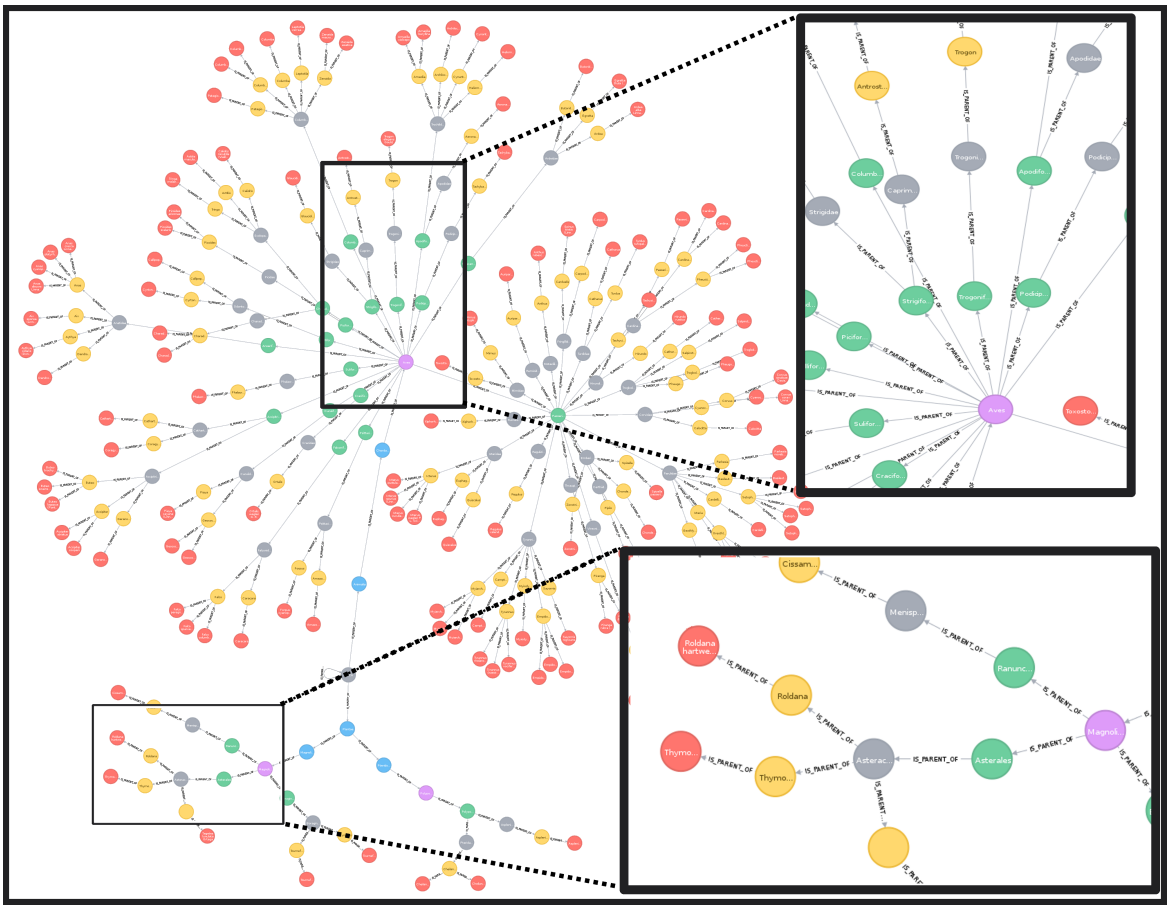


Fig. 2.5 A visualisation of a Local Taxonomic Tree built with the relationship: `IS_PARENT_OF`. The rectangles show zoomed areas in different sections of the tree (upper region for Birds (Order Aves), lower for plants (Order Magnoliopsida)). Colored nodes indicate distinct taxonomic levels (red : species, yellow: genera, grey: families, green orders, purple: classes).

2.4.1 Additional data used

We use the IUCN Red List of Threatened Species (Red List) (IUCN, 2019) in Mexico to account for the proportion of species (critically endangered, endangered or vulnerable) associated with the presence of jaguars. For aggregating the data into taxonomic trees (i.e `TreeNeo` objects), as well as for extracting their corresponding environmental covariates, we used a 0.05° (c. $5km$) resolution grid intersected with the terrestrial regions of Mexico and Central America. The used grid is included in the default installation of the engine and therefore, all the analysis performed in this example is reproducible.

2.4.2 Methodology

We first obtain the grid cells with at least one occurrence of jaguar. As these cells are `Cell` objects, it is possible to extract associated neighbouring cells using the method: `getNeighbours`. We can apply the same method recursively four times to obtain a list of neighbouring cells within a 4 degree neighbourhood. For each cell, we obtain the local taxonomic tree. The resulting trees are merged into a single tree that contains the union of all the nodes of all the local trees. Therefore, the aggregated tree contains all the known co-occurrences of jaguar in a neighbourhood of degree 4. The resulting tree is filtered to select only the nodes that match the Red List of threatened species. A new tree object is created using the selected nodes, an operation know as *trimming*.

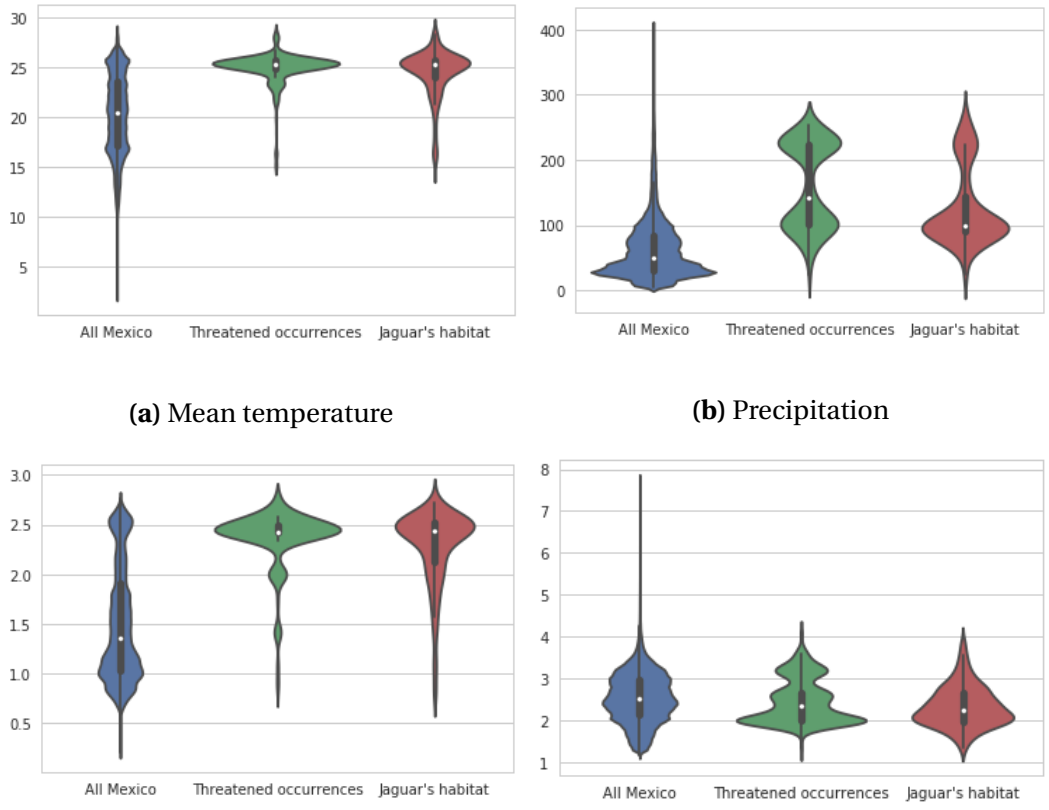
To provide an estimate of which nodes co-occur more often with jaguars, we rank all the nodes in the merged tree using the frequency of presence of each node at each neighboring cell. To show the raster querying capabilities, we contrast these results with environmental ranges of: jaguars, threated species and the entire country using the `raster_api` module. Finally, we provide methods for interactive visualisations of the extracted spatial data and the network structure.

2.4.3 Results of the worked example

The taxonomic analysis of the most abundant families across all neighbouring cells where: Muridae (rodents, 29%), Phyllostomidae (a family of bats, 23%) and Cervidae (deers, 15%) for the case of mammals. For of parrots (Order Psittaciformes) the most frequent species was *Ara militaris* (military macaws, 2%) and several species of the genus Amazona, accounting for 16% in total. Although the order Psittaciformes was abundant (23%) in the group of vertebrates, the most abundant taxon (*A. militaris*) only co-occurred 2% of the time with the jaguar's neighbouring cells. This result shows the great diversity of species within the group of parrots. This is consistent with natural history records, where it has been described that these species inhabit humid forests, wooded foothills and canyons in elevation ranges between 500 and 1,500 metres above sea level EOL.

The same analysis applied to plants showed that the most abundant genera were: the epiphyte Tillandsia (19%), the *Coussapoa oligocephala* (6%) , Pouteria (several species, 9%), *Cedrela odorata* (3.2%), which are tropical trees, and other trees not typical from tropical rain forests like Oreopanax (9%) and Quercus (6%). Longer lists of the most abundant taxa detailed in the worked example as well as their interactive version in the Jupyter notebook are provided in the file `examples/Official Demo Co-occurrences.ipynb` located in the Biospytial repository. A visualization of the threatened taxa tree is shown in figure 2.9 for: kingdoms, phyla, classes and orders.

From an environmental perspective there is a clear concordance between jaguars' habitat and threatened taxa, when compared to all Mexico, for mean temperature (fig 2.6a), annual rainfall (fig 2.6b and wind speed (fig 2.6d). In fact, threatened species and jaguars show environmental modalities distinct from all Mexico. To create the plots we used the library seaborn. Detailing the process for creating these graphs is out of the scope of the present tutorial. However, the snippet has been included in the interactive notebook.



(a) Mean temperature

(b) Precipitation

(c) Water vapor pressure

(d) wind speed

Fig. 2.6 Comparison of mean annual environmental ranges between treatments: All Mexico, threatened taxa and cells with occurrences of jaguars. See next section for more details.

2.4.4 Discussion

The latter example gives a glimpse of the general capabilities of the system in terms of: traversing the knowledge graph, extracting relations and performing geospatial queries and processing. The analysis was performed on a grid of approximately 4km of spatial resolution, a sensible choice, given the jaguars' body size and nomadic habits. However, it is a justified concern the applicability of a similar grid in analyses where the species under consideration are either too small or too large for that chosen spatial resolution. Take for example studies on ant colonies or even soil microbiota. Therefore, an adequate spatial resolution is critical to the research question and, consequently, any multi-purpose spatial

knowledge engine (as the one proposed in this chapter) should allow the implementation of various spatial grids at arbitrary resolutions to satisfy the research needs.

In the current implementation, the 4 km grid is the one with the finest resolution. However, the design of the system supports the integration of grids with finer resolutions. This is achieved by an abstract implementation of the geospatial data structures that define the grids as a connected acyclic graph (i.e. tree) that links cells through different scales (i.e spatial resolutions), similarly to the taxonomic tree structure.

In the spatial case, each unit area (i.e cell) of a grid can be subdivided in n (currently $n = 4$) different areas. This process induces a tree structure in which the original cell (i.e parent cell) assigns the link (`IS_CONTAINED_IN`) to each of the newly formed cells (children). This process can be executed recursively to fit any arbitrary spatial resolution. As such, each cell in the system is connected to other cells by two relationships (`IS_CONTAINED_IN` and `IS_NEIGHBOUR_OF`. This method for specifying grids of arbitrary size is an implementation of a Hierarchical Discrete Global Grid (Sahr et al., 2003). Additionally, the complete hierarchical structure that accounts for all grids at multiple spatial resolutions is part of the engine's knowledge graph. In this sense, the root of this spatial aspect of the knowledge graph corresponds to a cell that covers the whole Earth. The current version of the engine needs the existence of the specific grid cell (i.e. location and spatial resolution) in the database for performing operations. In the future we would like to allow the creation of new spatial grids *on demand* (i.e dynamically) to satisfy the user needs.

Clustered occurrences

Another aspect for consideration is the case where the research question involves the analysis of organisms at an organisation level distinct from the species level. That is, studies where the taxonomic aggregation of *species* is not enough to capture the nuances required for answering the research question involved. Examples of this could be: the

selection of different herds of elephants within the same species or the analysis of different populations of sahuaros *Carnegiea gigantea (Britton and Rose)* across the Sonoran desert.

New aggregation levels could be added into the taxonomic tree by linking them to the existing hierarchical structure, in a similar approach as the global spatial grid mentioned before. This added information is relevant for metapopulation studies, where the main concern is the prevalence and dynamics of spatially separated populations of the same species. Ecological synthesis can also be benefited by this approach. With this aid, meta-analysis studies can group occurrences by type of survey or collection to represent random effects.

2.5 Tutorial

The time for executing the following example varies considerably depending on the group of interest, the size of the neighbourhood and the computer platform. A quick workaround to speed up the processes is to reduce the number of neighbouring cells (order of the neighbourhood), for example a degree of 1.

A reproducible version of this tutorial is included in the Biospytial source code (inside the folder `examples/`) in an interactive jupyter notebook file named:

```
Official Demo Co-occurrences_jaguar.ipynb
```

The following section is a static version and is subject to minor modification to fit the layout and format of this PDF version.

2.5.1 Selecting the node *Jaguar*

We begin by selecting the node in the ToL corresponding to the genus *Panthera*. This node is linked to some `Species` and `Family` type nodes and also has links to `Occurrence`

nodes, where the information of location and time is stored. To start the traversal we need to first select this node. To do so we use the function `pickNode` using the following syntax:

```
pickNode(<Type of Node>,'name of the node')
```

In the example below we see how to load the `pickNode` function and the appropriate node class (in this case `Genus`).

```
from drivers.graph_models import Genus, pickNode
jaguars = pickNode(Genus,"Panthera")
```

The variable *jaguars* is now an instance of the class **Genus**. As such, it has associated attributes and methods. Its string representation is the following:

```
jaguars :  <TreeNode type: Genus id = 2435194 name: Panthera>
```

We proceed to traverse through all the cells where any occurrence of the *Panthera* genus was registered. To do so we call the attribute *cells*. This attribute is abstracted with *lazy evaluation*. Therefore, to fetch all the associated data we need to convert the object into a list (or a partial list using an iterator).

```
cells = list(jaguars.cells)
print("cells has %s elements"%len(cells))
```

```
cells has 62 elements
```

The resulting list has cell instances, each one connected to other cells by the relation: 'IS NEIGHBOUR OF'. Accessing their related cells is achieved by the method:

```
cell.getNeighbours(with_center=[Boolean],order=[Int])
```

where the parameter `with_center` returns the center of the neighborhood, and the parameter `order` the size (in number of cells) of the neighborhood (this value can be reduced to 1 for faster computation). In our case, we apply this method for each cell using a `map` function with a `lambda` expression.

```
neighbours = map(lambda cell :
                 cell.getNeighbours(with_center=True,order=4),
                 cells)
```

Lambda expressions are part of the Python syntax and are used to create anonymous functions. The *map-lambda* technique allows the definition of statements that are applied to all the elements of a list, returning a new list of objects obtained by evaluating the lambda expression on every element of the given list. Along this tutorial, the use of the *map-lambda* technique is frequently used. Whenever this expression comes it is recommended to read the form:

```
map(lambda x : <something involving x> , some_list)
```

As, "for all *x* in `some_list`, do *something involving* `x`". In the example above, the object `neighbours` is a list of neighbouring cells obtained from the method `getNeighbours` available on each cell instance (i.e. each element of the `cells` list).

As this list is composed of list-type elements (i.e. it is a nested list), we need to reduce it into a single list composed of only cell instances, a process known as flattening. To do this simply reduce the list as this.

```
# the + operator between two list instances merges them together.
neighbours = reduce(lambda list_a , list_b : list_a + list_b, neighbours)
```

The *reduce* function is a Python standard function that receives a two parameter function (in this case a lambda expression receiving parameters `list_a` and `list_b`) and the

nested list `neighbours`. The *reduce* function applies the lambda expression to the first pair of elements of the list and iteratively applies the result to the next element. As the sum operation between lists (+) merges the elements of both lists into a single list, performing this operation across the entire nested list `neighbours` result in a flattened list.

The resulting `neighbours` list now has 2497 Cell nodes. In the current implementation the name of the Grid (where all the Cells are contained) is called *mex4km*. We can display the first three elements as:

```
neighbours[:3]
```

```
[< Cell-mex4km id = 234686 >,
 < Cell-mex4km id = 234685 >,
 < Cell-mex4km id = 234684 >]
```

2.5.2 Converting cells to local taxonomic trees

We obtain the ToL inside each Cell node by extracting the occurrences inside each cell (using the method `occurrencesHere`) and plugging them into the *TreeNeo* constructor. The name *TreeNeo* is used because the storage backend is the Neo4j graph database.

```
from drivers.tree_builder import TreeNeo
cell_1 = neighbours[1]
tree_1 = TreeNeo(cell_1.occurrencesHere())
print(tree_1)
```

```
<LocalTree Of Life | Root: LUCA - n.count : 1062- >
```

The `n.count` value indicates the number of total occurrences. We can generate all the trees iteratively using a mapping from the `TreeNeo(cell.occurrencesHere())` through all neighbouring cells. This may take some time depending on the number of cells and occurrences on each cell. For reducing this time go to subsection 2.5.1.

```
sample_trees = map(lambda cell : TreeNeo(cell.occurrencesHere()),neighbours)
```

As in the last example, we can see basic information as object description. Here the first four elements are shown.

```
sample_trees[:4]
```

```
[<LocalTree Of Life | Root: LUCA - n.count : 3- >,
 <LocalTree Of Life | Root: LUCA - n.count : 1062- >,
 <LocalTree Of Life | Root: LUCA - n.count : 151- >,
 <LocalTree Of Life | No record available: - n.count : 0- >]
```

The value `n.count` indicates the number of occurrences found for the present node. It is possible to have empty trees, when no occurrences were found. This is shown with the text `No record available`.

2.5.3 Exploratory analysis on a single Tree

We select a tree in this example and explore informative data.

```
tree = sample_trees[1]
```

The object `tree` wraps the entire tree structure. All `tree` objects have as their starting node the root of the Taxonomic Tree, representing all known life.

```
root = tree.node
```

`root` node is similar to `Family` node, `Genus` node, etc. They all belong to the class: `TreeNode`. We can access a specific child node with the prefix `to_[name of taxon]`. For example, accessing the node 'Animalia' can be done with:

```
animalia = root.to_Animalia
animalia
```

```
<LocalTree | Kingdom: Animalia - n.count : 742- | AF: 0.05>
```

Traverse by children nodes

We can concatenate this method until the children attribute is empty. If running Biospytial in an interactive session (like a Jupyter notebook or iPython) we can use the key [TAB] to autocomplete and show the available nodes. For example, the family of rodents *Muridae*.

```
root.to_Animalia.to_Chordata.to_Mammalia.to_Rodentia.to_Muridae
```

```
<LocalTree | Family: Muridae - n.count : 34- | AF: 0.05>
```

Tree traversal by taxonomic level

The taxonomic levels (e.g., families, orders, etc.) are stored as atributes of the `TreeNeo` class. For example, to see the available `phyla` in this tree do:

```
print(tree.phyla)
```

```
[<LocalTree | Phylum: Chordata - n.count : 740- | AF: 0.05 >,
 <LocalTree | Phylum: Arthropoda - n.count : 2- | AF: 0.05 >,
 <LocalTree | Phylum: Bryophyta - n.count : 99- | AF: 0.05 >,
 <LocalTree | Phylum: Magnoliophyta - n.count : 175- | AF: 0.05 >,
 <LocalTree | Phylum: Mycetozoa - n.count : 46- | AF: 0.05 >]
```

and for some families inside this tree:

```
print(tree.families[:5]
```

```
[<LocalTree | Family: Menispermaceae - n.count : 3- | AF: 0.05 >,
 <LocalTree | Family: Piperaceae - n.count : 7- | AF: 0.05 >,
 <LocalTree | Family: Lauraceae - n.count : 2- | AF: 0.05 >,
 <LocalTree | Family: Acanthaceae - n.count : 7- | AF: 0.05 >,
 <LocalTree | Family: Plantaginaceae - n.count : 1- | AF: 0.05 >]
```

2.5.4 Tree operations

Tree objects allow symbolic operations for adding (merging) and intersecting other tree objects. These operations are currently implemented as `sum (+)` and `intersection (&)`. These operations can be applied to arbitrary number of trees and it is useful in comparative studies that require the calculus of (α, β, γ)-diversity using a combination of these operations (Whittaker, 1972). Mathematically, these operations are equivalent theoretic *set* operations acting at the occurrence level. As an example consider the following: let `t1` and `t2` be two trees from the list of `sampled_trees`, i.e.

```
t1 = sample_trees[1]
t2 = sample_trees[2]
```

Addition

Adding trees is equivalent to merging them. That is, making the union of all the nodes (inter nodes and leaves). The tree objects (`TreeNode` and `TreeNeo` classes) allow the use of the + operation. For example, the merge tree of `t1` and `t2` is obtained with:

```
t3 = t1 + t2
```

We can see the effect of this by selecting the nodes of a certain taxonomic level, for example, the classes of `t1` and `t2` are:

```
print(t1.classes)
```

```
[<LocalTree | Class: Myxomycetes - n.count : 46- | AF: 0.05 >,
 <LocalTree | Class: Bryopsida - n.count : 99- | AF: 0.05 >,
 <LocalTree | Class: Amphibia - n.count : 1- | AF: 0.05 >,
 <LocalTree | Class: Aves - n.count : 667- | AF: 0.05 >,
 <LocalTree | Class: Reptilia - n.count : 2- | AF: 0.05 >,
```

```
 <LocalTree | Class: Mammalia - n.count : 70- | AF: 0.05 >,
 <LocalTree | Class: Liliopsida - n.count : 36- | AF: 0.05 >,
 <LocalTree | Class: Magnoliopsida - n.count : 139- | AF: 0.05 >,
 <LocalTree | Class: Insecta - n.count : 2- | AF: 0.05 >]
```

```
print(t2.classes)
```

```
[<LocalTree | Class: Protosteliomycetes - n.count : 2- | AF: 0.05 >,
 <LocalTree | Class: Myxomycetes - n.count : 112- | AF: 0.05 >,
 <LocalTree | Class: Agaricomycetes - n.count : 4- | AF: 0.05 >,
 <LocalTree | Class: Liliopsida - n.count : 8- | AF: 0.05 >,
 <LocalTree | Class: Magnoliopsida - n.count : 25- | AF: 0.05 >]
```

```
print(t3.classes)
```

```
[<LocalTree | Class: Protosteliomycetes - n.count : 2- | AF: 0.05 >,
 <LocalTree | Class: Myxomycetes - n.count : 158- | AF: 0.05 >,
 <LocalTree | Class: Agaricomycetes - n.count : 4- | AF: 0.05 >,
 <LocalTree | Class: Bryopsida - n.count : 99- | AF: 0.05 >,
 <LocalTree | Class: Amphibia - n.count : 1- | AF: 0.05 >,
 <LocalTree | Class: Aves - n.count : 667- | AF: 0.05 >,
 <LocalTree | Class: Reptilia - n.count : 2- | AF: 0.05 >,
 <LocalTree | Class: Mammalia - n.count : 70- | AF: 0.05 >,
 <LocalTree | Class: Liliopsida - n.count : 44- | AF: 0.05 >,
 <LocalTree | Class: Magnoliopsida - n.count : 164- | AF: 0.05 >,
 <LocalTree | Class: Insecta - n.count : 2- | AF: 0.05 >]
```

Intersection

Intersection is applied through the `&` operation and it is equivalent to the intersection of sets with the *difference* that it is only applied to the leaf nodes, that is, the **Occurrence**

nodes. Once the leaf nodes are selected, the algorithm propagates through the parent nodes until it reaches the root node. To see the formalization of the data structure go to supplementary materials II. To obtain the intersection of two trees do:

```
t = t1 & t2
print(t)
```

```
<LocalTree Of Life | No record available: - n.count : 0- >
```

In this case, the intersection is empty because the Occurrences are overlaid in a regular lattice that partitions the space (i.e. the cells are disjoint). See supplementary materials II for a formal definition.

Efficient addition of trees from a list of cells

We can use the sum iteratively in a folding sum to obtain a Tree object representing all the areas defined in a list of Cells.

```
big_tree = reduce(lambda a , b : a+b , sample_trees)
```

However, this method is not efficient. In each step, a new tree is created and the internal logic to generate the union of all the intermediate nodes can result in redundant calculations. It is much faster to select first the occurrences for all the trees inside a list and then plug them into the TreeNeo constructor, as in the example below.

```
# Faster version
ocs = map(lambda s : s.occurrences,sample_trees)
## ocs is a nested list.
## We need to flatten this into  a single list of occurrences
ocs = reduce(lambda a,b : a + b, ocs)
big_tree = TreeNeo(ocs)
print(big_tree)
```

```
<LocalTree Of Life | Root: LUCA - n.count : 374731- >
```

The resulting tree could be very large. In this case, the obtained tree (`big_tree`) comprises 374731 occurrences. Remember that this tree is the resulting union of all the local taxonomic trees obtained from the neighbourhood of degree 4 around the cells where jaguars occurred.

2.5.5 Selecting nodes from the Red List

We filter the *Species* nodes from the `big_tree` that are present in the Red List of threatened species. To do this we simply match the names using regular expressions. Using more sophisticated methods for data matching are out of the scope of the present example. We assume that the Red List data (a CSV file) have been loaded into a data frame with the name `redlist`.

```
## Filter critically endangered species
critical_sps = redlist[
                        (redlist.redlistCategory == 'Critically Endangered')
                       | (redlist.redlistCategory == 'Endangered')
                       | (redlist.redlistCategory == 'Vulnerable')
                      ].scientificName.apply(str.lower)

protected_by_jaguar = map(lambda critical_sp :
                          filter(lambda sp : critical_sp in sp.name.lower(),
                          big_tree.species),
                          critical_sps)

## Remove empty lists
```

```
protected_by_jaguar = filter(lambda l :
                                l != [], protected_by_jaguar)
## flatten lists
threatened_species = reduce(lambda a,b : a + b ,protected_by_jaguar)
## remove species repetitions
threatened_species = list(set(threatened_species))
## Extract all corresponding occurrences and flatten list
t_ocs = reduce(lambda l1,l2 : l1 + l2 ,
               map(lambda l : l.occurrences, threatened_species))
## Instantiate new tree
threatened_tree = TreeNeo(t_ocs)
```

The `threatened_tree` is now a taxonomic tree that includes only the occurrences that match the species names of the Red List. To calculate the percentage of threatened species contained in the selected tree we can do:

```
## total number of critical endangered species
ncrit = len(critical_sps)
len(threatened_tree.species) / float(ncrit) * 100
```

```
13.49 %
```

That is, 13.49% of the threatened species are contained in the neighbouring regions where jaguars had been registered. To see if this result is relevant we calculate the percentage of the covered area with respect to the whole country. Before doing so, it is convenient to transform the selected geometries in a projected coordinate system with metric units.

Reprojecting data

The default coordinate reference system (crs) in the data used is in geographic coordinates with WGS84 datum (EPSG:4326). The units of this crs is in degrees, therefore the calculated

area is defined in squared degrees. In order to account for areas and distances in meters (or kilometers) we need to project the selected geometries into an appropriate projected coordinate system. To achieve this, we need to import some extra functions.

```
from shapely.ops import transform
from shapely import wkt,wkb
import pyproj
from functools import partial
```

Here we used the *Alberts Equal Area Conic projection* to account for an accurate area representation. This projection is specified in a string using the Proj4 syntax.

```
projection_string = """+proj=aea +lat_1=14.5 +lat_2=32.5 +lat_0=24
                    +lon_0=-105 +x_0=0 +y_0=0 +ellps=GRS80
                    +datum=NAD83 +units=m +no_defs;
                    """
mex_eq_area_proj = pyproj.Proj(projection_string)
## The WGS84 crs is defined as EPSG:4326
proj_in = pyproj.Proj(init='epsg:4326')
## function to project using the parameters of the
## original projection and the mexican equal area.
project = partial(
    pyproj.transform,
    proj_in,
    mex_eq_area_proj)
## Transform all cells to calculate area.
projected_neighbours_cells = map(lambda cell :
                                 transform(project,
```

```
                                    cell.polygon_shapely),
                                    neighbours)
```

For calculating the average cell size and the total area in square kilometers (1,000,000 m^2) we do:

```
tokm2 = 1000000 # to convert to sq. kilometers
areas = map(lambda cell : cell.area,
            projected_neighbours_cells)
total_cell_area = sum(areas)
## calculate the mean
np.mean(areas) / tokm2
## standard deviation
np.std(areas)/ tokm2
```

The calculated average area of all cells is 27 ± 3 km^2 and the total area is $8,509.81$ km^2.

2.5.6 Trimming trees

In certain situations we need to select a particular branch of a tree. We can cut (*trim*) this branch by simply selecting a node and converting it into a `TreeNeo` instance to produce a full feature tree. The method (function) for converting a TreeNode into a full feature tree is: `plantTreeNode`. We focus our attention on four branches of the `threatened` tree that co-occurrs with the presence of jaguars. These branches are: mammals (class *Mammalia*), parrots (order *Psittaciformes*) amphibians (class *Amphibia*) and plants (kingdom: Plantae).

Select the branch of interest

Trimming the tree is achieved by first selecting the nodes of interest and then converting all the descendant branches into fully featured trees. There is no restriction for selecting

the taxonomic type of the node (`mammals` and `amphibians` are `Class` type while `parrots` are `Order` type).

```
mammals = threatened_tree.to_Animalia.to_Chordata.to_Mammalia
parrots = threatened_tree.to_Animalia.to_Chordata.to_Aves.to_Psittaciformes
amphibians = threatened_tree.to_Animalia.to_Chordata.to_Amphibia
plants = threatened_tree.to_Plantae
```

The method `plantTreeNode()` converts the `TreeNode` and resulting descendants into a full featured tree (TreeNeo object).

```
mammals = mammals.plantTreeNode()
birds = birds.plantTreeNode()
amphibians = amphibians.plantTreeNode()
plants = plants.plantTreeNode()
```

We can add all these trees together using the sum operation.

```
vertebrates = mammals + parrots + amphibians
```

However, as explained earlier, an optimized version for summing more than two trees is achieved by instantiating a `TreeNeo` with all the occurrences.

```
vertebrates = TreeNeo(mammals.occurrences +
                      parrots.occurrences +
                      amphibians.occurrences)
print(vertebrates)
```

The total number of occurrences contained in the `vertebrates` tree is:

```
<LocalTree Of Life | Root: LUCA - n.count : 2056- >
```

Ranking the most frequent nodes in the selected list of cells

We proceed now to rank some groups according to their frequency of occurrence within the cells of the study area (i.e. the jaguar's neighbouring cells). The ranking analysis calculates this frequency for each node in a tree given a referential list of trees. That is, assuming that we have *n* different trees (e.g. one per cell), and a tree of interest (in this case `threatened_tree`) how frequently each node appears in the global tree (e.g `threatened_trees`) with respect to the list of *n* trees? Figure 2.9 shows these frequencies visualised as the size of each node. In our implementation, this analysis is performed with the method: `countNodesFrequenciesOnList(list_of_trees)` That is:

```
vertebrates.countNodesFrequenciesOnList(list_of_trees=sample_trees)
mammals.countNodesFrequenciesOnList(list_of_trees=sample_trees)
parrots.countNodesFrequenciesOnList(list_of_trees=sample_trees)
amphibians.countNodesFrequenciesOnList(list_of_trees=sample_trees)
plants.countNodesFrequenciesOnList(list_of_trees=sample_trees)
```

We can therefore rank by taxonomic level. In this example we show the procedure for *family* and *species* level in the different branches. Here, we show the corresponding top five nodes.

```
mammals.rankLevels()
mammals.families[:5]
```

```
[<LocalTree | Family: Muridae - n.count : 8 | AF: 0.30>,
 <LocalTree | Family: Phyllostomidae - n.count : 8 | AF: 0.29>,
 <LocalTree | Family: Cervidae - n.count : 14 | AF: 0.16>,
 <LocalTree | Family: Heteromyidae - n.count : 3 | AF: 0.15>,
 <LocalTree | Family: Tayassuidae - n.count : 158
 | AF: 0.15>]
```

```
parrots.rankLevels()
parrots.species[:5]
```

```
[<LocalTree | Specie: Ara militaris (Linnaeus, 1766) - n.count : 27->,
 <LocalTree | Specie: Amazona finschi (P. L. Sclater, 1864) - n.count : 23- >,
 <LocalTree | Specie: Amazona auropalliata (Lesson, 1842) - n.count : 3- >,
 <LocalTree | Specie: Amazona oratrix Ridgway, 1887 - n.count : 2- >,
```

```
amphibians.rankLevels()
amphibians.families[:3]
```

```
[<LocalTree | Family: Hylidae - n.count : 128- | AF: 0.083>,
 <LocalTree | Family: Plethodontidae - n.count :
 160 | AF: 0.05>,
 <LocalTree | Family: Eleutherodactylidae -
 n.count : 1- | AF: 0.016>]
```

```
plants.rankLevels()
plants.genera[:3]
```

```
[<LocalTree | Genus: Tillandsia - n.count : 3- | AF: 0.2>,
 <LocalTree | Genus: Lonchocarpus - n.count : 5- | AF: 0.18>,
 <LocalTree | Genus: Eugenia - n.count : 1- | AF: 0.15>]
```

2.5.7 Associated raster (environmental) information

Here, we demonstrate how to access raster data associated with a taxonomic tree `TreeNeo`. The raster data used are related to environmental variables stored in the RGU. Currently there are two forms for accessing this information: *i)* as a table with columns corresponding to environmental variables and rows defined by each occurrence (a point-based

method); *ii)* as a raster object sampled from the associated geometry of each tree or, in general, any (multi) polygon object. The raster object features methods for visualisation, geoprocessing and data exchange.

Extracting raster information as table

To extract the data in this format use the method (function):

```
TreeNeo.associatedData.getEnvironmentalVariablesPoints()
```

The output is a *Pandas* dataframe with the associated values of climatic covariates. See the following example:

```
table = vertebrates.associatedData.getEnvironmentalVariablesPoints()
print(table[:1])
```

Here we only show the first record.

Table 2.2 Output for environmental variables. Here showing only mean values for some variables on a single record.

	MinTemperature	...	Precipitation	Vapor	SolarRadiation	WindSpeed
0	22.25	...	21.16	1.33	16466.25	2.33

The geometric object of each tree is determined by the `Occurrence` nodes of the tree. In the graph database, each `Occurrence` node is linked to the `Cell` node that geographically contains the occurrence's location. One of the attributes of the `Cell` object is the geographic polygon that defines its border. The union of all the corresponding `Cell` nodes is what determines the geometric feature of the tree `TreeNeo`. As such, the raster extraction process is performed on each of the tree's associated cells.

Extracting Raster objects from TreeNeo instances

To extract the associated raster object of a `TreeNeo` instance use the method (function):

```
TreeNeo.associatedData.getAssociatedRasterAreaData([name of variable])
```

To obtain several environmental variables use:

```
associatedData.getEnvironmentalVariablesCells()
```

For example, information for a single variable can be obtained with:

```
meantemp_data = vertebrates.associatedData.
                              getAssociatedRasterAreaData(
                              'MeanTemperature')
```

The raster object is automatically added to the `TreeNeo` object after the method is called. The raster objects are appended to the attribute `associatedData`.

2.5.8 Extracting raster objects from arbitrary polygons

The extraction of raster objects is performed by the `raster_api` library, a Biospytial module for reading, writing and processing raster objects using the RGU as backend.

The `raster_api` can use natively any object stored in the knowledge engine that has at least a two dimensional geometric feature (attribute). This includes the basic operations for querying, reading and writing. For using external geometric objects like *Shapefiles, GeoPackages, GeoJSON, etc* the objects need to be transformed to their corresponding WKT or WKB (*Well Known Binary*) representation. Examples of these are described extensively in the Jupyter notebooks and in the documentation.

In this example we use the polygon defined by the border of Mexico to extract several raster objects (RasterData instances) using the *raster_api* module. We use these objects to compare the environmental ranges of: the threatened species, the Jaguars' habitat and the entire area of the country to conclude if the environmental niche of the threated species are covered by the habitat of the Jaguars' and how these ranges are different with respect to the whole country.

Importing the polygon for Mexico

The first step in this is to import the polygon for Mexico. The default installation of Biospytial includes the WorldBorders dataset (https://thematicmapping.org). Assuming that this dataset is installed, we can import the polygon of Mexico with the API provided by the class `Country` located in `sketches.models`. Country is a vector dataset stored in the RDBMS. The geometric feature is stored as the `geom` column.

```
from sketches.models import Country
## The syntax follows the Django Query Set API
mexico = Country.objects.filter(name='Mexico').first()
mex_area = mexico.geom.area


## For reprojecting the area of Mexico we similarly do:
mex_shapely = wkt.loads(mexico.geom.wkt)
mex_projected= transform(project,mex_shapely)
```

To calculate the percentage of area covered by all the cells with respect with the total area of Mexico we can do:

```
total_cell_area / mex_projected.area * 100
```

```
3.42%
```

For example, we can display simple visualisations invoking the method: `display_field()`. See figure 2.7.

```
vertebrates.associatedData.raster_MeanTemperature.display_field()
```

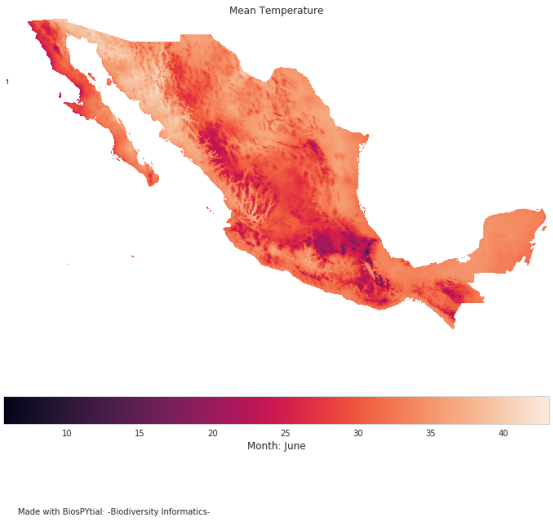


Fig. 2.7 The output of the method: `display_field()`, an easy way to visualise RasterData objects.

Interactive visualisation

As an alternative, we can export the raster object as an *xarray* (http://xarray.pydata.org) instance for interactive visualisation using the *Geoviews* (http://geoviews.org) package. To export the associated raster data to an *xarray* object do:

```
meantemp = vertebrates.associatedData.raster_MeanTemperature.to_xarray()
```

The following code gives an example on how to generate an interactive visualisation using the vertebrates' associated mean temperature data and the locations of the observed threatened species associated with the presence of Jaguars. We used the elevation data for Mexico (extracted before) as basemap. Figure 2.8 shows this visualisation at two different scales.

```
import geoviews as gv
from cartopy import crs
import geoviews.feature as gf
from geoviews import opts
gv.extension('bokeh')
```

```
sample_pt = gv.Points((env_threated_occurrences.x,env_threated_occurrences.y),
            label='ocurrences').opts(
            fill_color = 'orange',
            line_color = 'black',
            line_width = 0.5,
            line_alpha = 0.4,
            fill_alpha = 1.0,
            size = 5,
            )

elevation = all_mex_datasets[0].to_xarray()
elevds = gv.Dataset(elevation,crs=crs.PlateCarree())
elevimg = gvds.to(gv.Image,['Longitude','Latitude']
                  ).opts(cmap=plt.cm.gist_earth)

temp = meantemp.where(((meantemp.Longitude > -95) &
                       (meantemp.Longitude < -89) &
                       (meantemp.Latitude > 15) &
                       (meantemp.Latitude < 19)),
                       drop=True)

temp.name = meantemp.name
tempds = gv.Dataset(temp,crs=crs.PlateCarree())
tempimg = tempds.to(gv.Image,['Longitude','Latitude']).opts(cmap=plt.cm.magma)
```

```
## Display the map

map_ = (elevimg * gf.ocean * gf.coastline * gf.borders * tempimg * sample_pt )
```

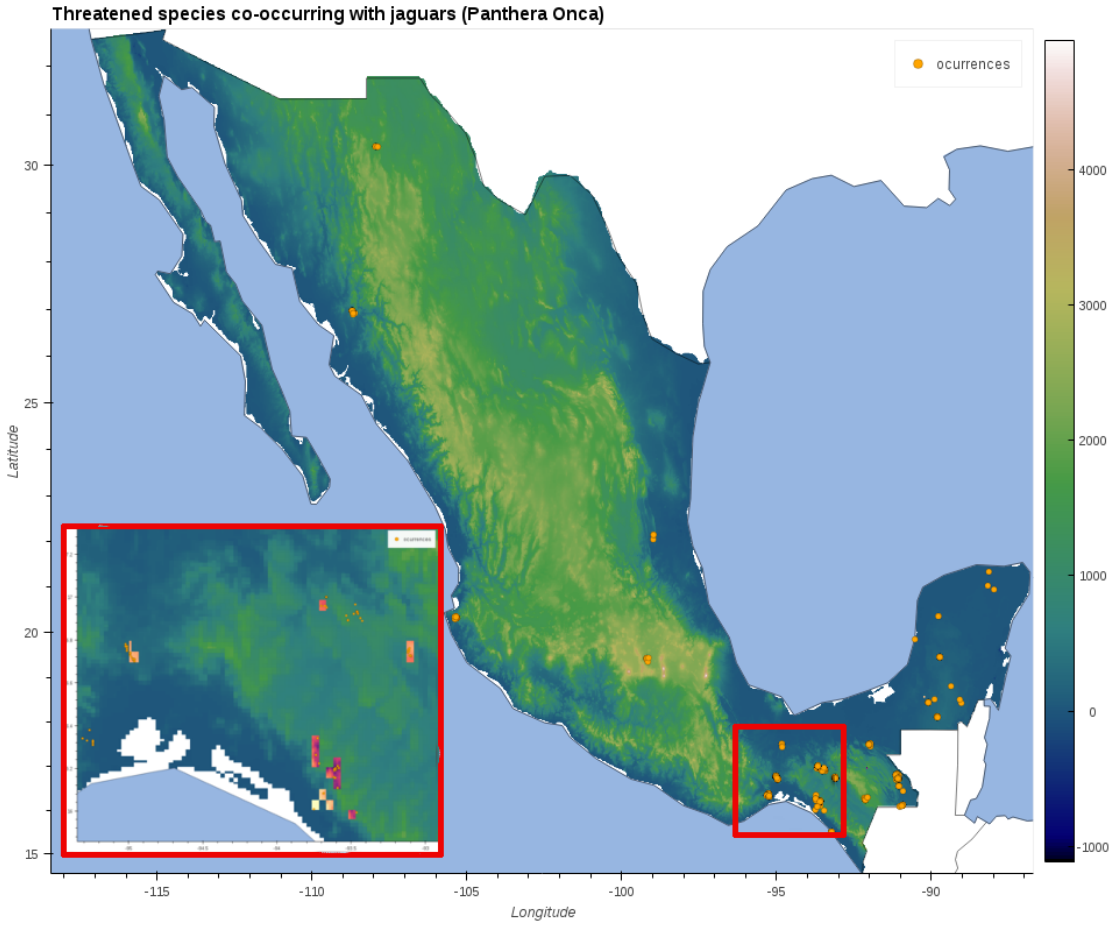


Fig. 2.8 A composite figure showing two states of the interactive visualisation. Orange dots represent occurrences of threated species associated with the presence of jaguars (*P. Onca*). The inland red square shows the zoomed-in area depicted in the left side of the figure. The colored squares in the zoomed area shows the mean temperature associated with threatened vertebrates (phylum Chordata). The base map shows the elevation for all the country. See section 2.3.2 for information regarding the data used.

2.5.9 Network visualisation and analysis

Each *tree* instance induces an acyclic graph. We can convert the tree into a networkx object to visualise and analyse its network properties. To do this, we simply need to use

the method: `tree.toNetworkx(depth_level=[k])` where *k* is the taxonomic level to reach in the tree, 0 for root 7 for species level.

Visualisation

A method for interactive visualisation has been developed using the *Holoviews* (https://holoviews.org) framework. To do this we need to invoke the method:

```
## Plot the Tree
from drivers.tools import to_interactivePlot
network = to_interactivePlot(threatened_tree,label_depth=8)
```

The output is a dictionary with two key-items: one for labels and the other for the actual graph (nodes and edges). To plot the whole graph we need to overlay both items.

```
network['labels'] * network['graph']
```

Analysis with standard graph algorithms

The `TreeNeo` structures are particular cases of graph traversals. As such, they can be analysed with graph theoretic methods. The library NetworkX (https://networkx.github.io/) is a Python package designed for analysing structure, dynamics and functions of complex networks. It includes standard graph algorithms and analysis measures as well as tools for import and export to other standard formats. We can convert a `TreeNeo` using the method: `toNetworkx(depth_level )`. where depth_level is the depth of the graph to be generated. In the next example we convert the threatened_tree to a `NetworkX` object and use this to calculate its corresponding adjacency matrix.

```
threatened_graph = threatened_tree.toNetworkx(depth_level=7)
from networkx import adjacency_matrix
```

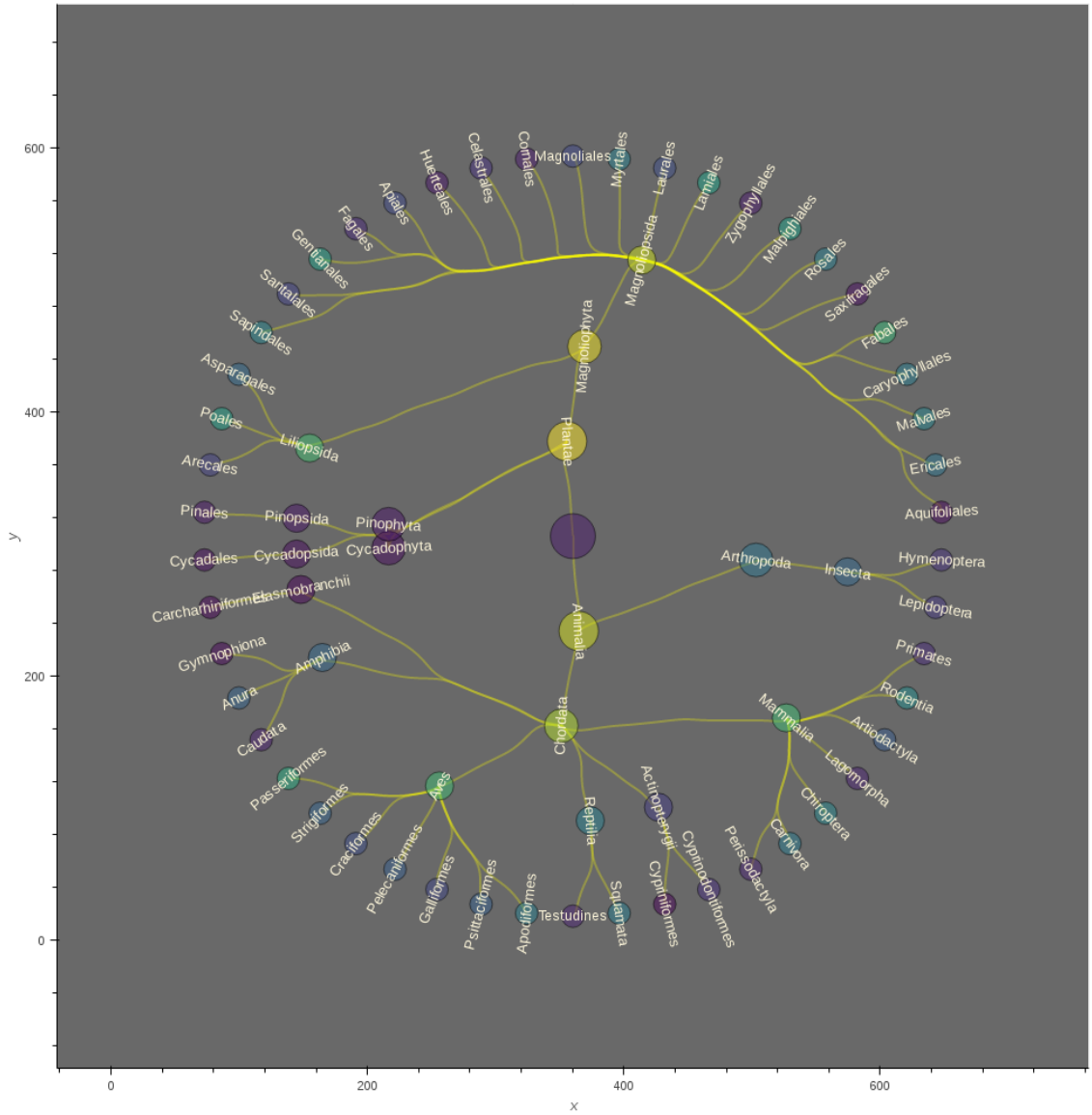


Fig. 2.9 A tree visualization for the merged tree corresponding to threatened taxa, showing up to *Order* level. The size of the nodes is proportional to the taxonomic level (the largest is the root of the tree, the smallest are orders). The node colouring indicates the frequency of occurrence with respect to all the neighbouring cells (neighbours of jaguars) being the brightest the highest ranked and the darker the lowest ranked.

```
M = adjacency_matrix(threatened_graph)
# uncomment this to plot the matrix
#plt.imshow(M.todense())
```

Representing `TreeNeo` objects into NetworkX graphs brings new possibilities for analysis and modelling. We hope this example will awake the spirit of the reader to explore the potential of representing data as complex graph structures.

2.6 Conclusions

Biospytial uses open source standards to integrate geospatial ecological big data as a tool for ecological niche modelling and the analysis of species distributions. This integration creates a complex network of data with enormous potential for data mining, information retrieval and visualisation. At the core, a web of semantic-wise relationships constitutes a corpus of taxonomic and environmental knowledge that opens up new ways to query and unveil complex ecological relations. To our knowledge, there is no other Open Source system with the design and capacity of achieving this including: i) storing information in a hybrid relational-graph system and ii) performing geospatial processes in vector and raster scalable databases.

A practical example provided a glimpse into how to query and manipulate taxonomic tree structures, as well as how to extract data, conduct frequency analysis and visualise results. The example demonstrated a new procedure to rank co-occurring taxonomic groups in an arbitrary size neighbourhood of pixels.

The GBIF occurrence data includes information only on location and taxonomy and in this sense the data are limited. However, the engine's design allows the capture, extension and exploration of semantic interpretation of the data by adding other types of relations. For example, linking information on trophic networks to the taxonomic backbone can

help in analysing spatial patterns of trophic groups and dependant species, a key question in conservation biology.

The development of Biospytial has followed best practices in scientific programming (Wilson et al., 2014b). We recognise that spatial analyses are often not generalisable and therefore replicable. However replicability and reproducibility can be enhanced by increasing openness and documentation transparency and completeness (Barba, 2019; Shannon and Walker, 2018; Teytelman, 2018). In fact, Biospytial's source code is open and can be accessed at: https://github.com/molgor/biospytial.git while this manuscript is Open Access. In the future, Biospytial can be further developed into a system not only for integration and distribution of datasets, but also as a tool for collaboration, experimentation, validation and reproduction of results in the era of Open Science, satisfying also the requisites of second generation SDI.

2.7 Availability of supporting source code and requirements

- Project name: Biospytial
- Project home page: https://github.com/molgor/biospytial
- Operating System(s): Platform independent (not tested in Windows)
- Other requirements: Docker 1.13 or higher
- License: GNU General Public License version 3.0 (GPLv3)
- Memory requirements: 40GB in HD for installing the database and at least 16GB in RAM for running the example.
- RRID:SCR_018226
- biotools:biospytial

The current example is located inside the folder `examples` with the name: `[Official Demo] Co-occurrences_Jaguar.ipynb`. The example has been modified only in the neighbourhood order, changing from 4 to 1. This modification reduces the data to process and the executing time.

2.8 Availability of supporting data

Snapshots of our code and other supporting data are openly available in the *GigaScience* repository, GigaDB (Escamilla Molgora et al., 2020b). The container images can be downloaded automatically using the script `installEngine.sh` . Instructions for installing and running the engine are located in the project's homepage.

Table 2.3 Corresponding URLs for source code and container images for the Biospytial engine. The modules and the source code do not include data. These should be installed separately or loaded independently.

Module name	**URL**
Graph Storage and Processing Unit	https://hub.docker.com/r/molgor/postgis_biospytial
Biospytial Computing Engine	https://hub.docker.com/r/molgor/biospytial
Relational Geoprocessing Unit	https://hub.docker.com/r/molgor/neo4j_biospytial
Source code	https://github.com/molgor/biospytial
Data	http://dx.doi.org/10.5524/100723

2.9 Abbreviations

BCE: Biospytial Computing Engine; BLOB: binary large object; CONABIO: National Commission for the Knowledge and Use of Biodiversity; CRS: coordinate reference system; CSV: comma separated value; DAG: directed acyclic graph; DEM: digital elevation model; EBVs: Essential Biodiversity Variables; EPSG: European Petroleum Survey Group; GBIF: Global Biodiversity Information Facility; GDAL: Geospatial Data Abstraction software Library; GSPU: Graph Storage and Processing Unit; MPI: Message Passing Interface; OGM:

object-graph mapping; ORM: object-relational mapping; RDBMS: Relational Database Management System; RGU: Relational Geoprocessing Unit; SDI: spatial data infrastructure; ToL: Tree of Life; WKT: well known text;

2.10 Funding

This project was jointly sponsored by the Doctoral Scholarships Program from the Mexican Science and Technology Council (CONACYT), the Faculty of Science and Technology from Lancaster University (FST-LU) and the GBIF Consortium through the GBIF Young Researchers Award (2016).

2.11 Authors' contributions

J.E. and P.A. conceived the original idea, which was further refined by all authors. The semantic structures and graph traversals were designed by J.E. with the mentorship of L.S. for integrating datasets. The software and system's design was developed by J.E. under the supervison of P.A. and L.S. The writing of the original draft was done by J.E with reviewing and editing from P.A. and L.S.

2.12 Competing interests

The authors declare that they have no competing interests.

2.13 Acknowledgments

We thank the effort of many researchers, students, public servants and citizen scientists that had contributed to sample, register and curate all the biodiversity occurrences data

contained in the GBIF database. We want to thank specially Raúl Jiménez Rosenberg from CONABIO for facilitating a complete snapshot of the GBIF database (2016) and the Free and Open Source Software community whose effort in developing software made possible the creation of this software.

Supplementary material I for *Biospytial: spatial graph-based computing for ecological big data*

2.14 Adding data in Biospytial

Biospytial is a Knowledge Engine that merges different data using graph theory in order to model ecological big datasets using geostatistical, graph and other frameworks. Biospytial has reached a snapshop stage for initial release and will undergo further development.

2.14.1 Aims of this tutorial

This tutorial provides a simple guide on how to install new data sources. As an example, two data sources are installed: a vector-based data source called: `global_ecoregions` and raster based data source: `World Population for Latin America`.

2.14.2 Assumptions

A fully installed and running Biospytial Suite. This mean the three modules are running.

- Geoprocessing-Backend (GBP)
- Graph-Computing-Engine (GCE)
- Biospytial-Client. (BPE)

In addition, the datasources are downloaded and allocated in an accessible path from the Biospytial Client.

2.14.3 Converting the data to a Django Model

For data handling, Biospytial uses the ORM model for accessing geospatial data stored in the Geoprocessing-Backend. To achieve this, a Class called Model is specified using a given datasource. That is, each datasource has a class specification for communicating with the Relational Database manager.

2.14.4 Vector data

We make use of the tool `ogrinspect` to generate the model definition for a shapefile file and follow these steps.

1. Login to Biospytial-Client session (the bash shell and not the iPython environment).
2. Locate the path where the data are stored. In this case we are interested in adding the datasource 'terr-ecoregions-TNC' which has an ESRI-Shapefile format.

Ingest the shapefile into the GPB

We make use of the `LayerMapping` utility. Use the tool `ogrinspect` described in the manage.py module inside the folder `apps` where all the Biospytial sources are located. The general syntax of this command is:

| python manage.py ogrinspect [options] [options]|

For this example:

```
python manage.py ogrinspect path_to/tnc_terr_ecoregions.shp TerrEcoregions \
                            --srid=4326 --mapping --multi
```

where the:

- `-srid` option sets the SRID for the geographic field.
- `-mapping` option tells ogrinspect to also generate a mapping dictionary for use with LayerMapping.
- `-multi` option is specified so that the geographic field is a MultiPolygonField instead of just a PolygonField.

More information is provided in: (https://docs.djangoproject.com/en/2.0/ref/contrib/gis/tutorial/)

The command prints in the standard output format the class definition for this dataset. If we decided to use the `-mapping` option a dictionary is also included with a standarized format for the column names.

2.14.5 Export Shapefile into the Database (Geoprocessing Container)

We use the `LayerMapping` utility to make this process faster. The first action is to edit or create the file `load_shapefiles.py` inside the `ecoregions` app.

We define here the mapping names dictionary (see above) and the necessary code to insert the shapefile into the database.

This is the content of the file `load_shapefile.py`

```
!/usr/bin/env python
-- coding: utf-8 --

from future import absolute_import, division, print_function, unicode_literals
import os from django.contrib.gis.utils
import LayerMapping from .models
import TerrEcoregions from biospytial
import settings

""" Functions for exporting shapefiles into the Postgis Database. """
```

```
author = "Juan Escamilla Molgora"
copyright = "Copyright 2018, JEM"
license = "GPL"
mantainer = "Juan"
email ="molgor@gmail.com"

#Generated by ogrinspect

terrecoregions_mapping = { 'eco_id_u' : 'ECO_ID_U',
                           'eco_code' : 'ECO_CODE',
                           'eco_name' : 'ECO_NAME',
                           'eco_num' : 'ECO_NUM',
                           'ecode_name' : 'ECODE_NAME',
                           'cls_code' : 'CLS_CODE',
                           'eco_notes' : 'ECO_NOTES',
                           'wwf_realm' : 'WWF_REALM',
                           'wwf_realm2' : 'WWF_REALM2',
                           'wwf_mhtnum' : 'WWF_MHTNUM',
                           'wwf_mhtnam' : 'WWF_MHTNAM',
                           'realmmht' : 'RealmMHT',
                           'er_update' : 'ER_UPDATE',
                           'er_date_u' : 'ER_DATE_U',
                           'er_ration' : 'ER_RATION',
                           'sourcedata' : 'SOURCEDATA',
                           'geom' : 'MULTIPOLYGON', }

file_shp = os.path.abspath( os.path.join(settings.PATH_RAWDATASOURCES,
                                         'terr-ecoregions-TNC',
                                         'tnc_terr_ecoregions.shp'), )

def run(verbose=True):
    lm = LayerMapping( TerrEcoregions, file_shp,
                       terrecoregions_mapping, transform=False, )
    lm.save(strict=True, verbose=verbose)
```

To load the layer, one must log into the Biospytial iPython environment with:

| python manage.py shell |

Inside the BCE module (e.g. ssh) and using the iPython console, run the following:

```
from ecoregions import load_shapefiles

load_shapefiles.run()
```

2.14.6 Example 2: Adding vector data

Download the roads shapefile from: http://www.conabio.gob.mx/informacion/gis/maps/geo/carre1mgw.zip

Using the `ogronspect` tool we have the following:

```
#This is an auto-generated Django model module created by ogrinspect.

from django.contrib.gis.db import models

class MexRoads(models.Model):
    fnode_field = models.BigIntegerField()
    tnode_field = models.BigIntegerField()
    lpoly_field = models.BigIntegerField()
    rpoly_field = models.BigIntegerField()
    length = models.FloatField()
    cov_field = models.BigIntegerField()
    cov_id = models.BigIntegerField()
    geom = models.MultiLineStringField(srid=4326)

#Auto-generated LayerMapping dictionary for MexRoads model

mexroads_mapping = { 'fnode_field' : 'FNODE_',
 'tnode_field' : 'TNODE_',
 'lpoly_field' : 'LPOLY_',
 'rpoly_field' : 'RPOLY_',
 'length' : 'LENGTH',
 'cov_field' : 'COV_',
 'cov_id' : 'COV_ID',
 'geom' : 'MULTILINESTRING'
 }
```

2.14.7 Add raster data

As before, this process involves two steps: *i)* loading the datasource into the database and *ii)* creating a Class definition for the datasource, intepreted by the engine.

Add the data to the database

We use the raster support from Postgis. We use the script: `migrateToPostgis.bash` located in: `/apps/raster_api/bash_raster_tools/bash_scripts`

However, the tools for ingesting data into the database are stored in the Geospatial Processing Container. We need to log into this container and run the above file. You can copy the `bash_raster_tools` inside this container and run the command `migrateToPostgis.bash`.

Example Running the following line will load the dataset into the database.

| migrateToPostgis.bash [RasterData.tif] |

Create a class definition for Raster Data

We need to add the Model Class definition inside the file: `raster_api/models.py`

The base class is GenericRaster. We need to extend this class into a new definition according to the type of data we are loading.

The following code describes a generic template for creating a class definition.

```
class myNewModel(GenericRaster):
    """
    ..
    Description of the model in plain words.
    Attributes
```

```
    ==========
    Default attributes given by the raster2pgsql
    id : int Unique primary key
        This is the id number of each element in the mesh.

    """
    number_bands = 1
    neo_label_name = 'name of node class'(optional)
    link_type_name = 'name of associated edges'(optional)
    units = 'The measurment units name'

    class Meta:
        managed = False
        db_table = 'name of table in DB'

    def __str__(self):
        c = "< String representation: %s >"
        return c
```

The last step is to add this new model into the `raster_models_dic` in the `settings.py` file.

```
raster_models_dic = {
'WindSpeed' : raster_models[7],
'Elevation' : raster_models[0],
'Vapor' : raster_models[6],
'MaxTemperature' : raster_models[5] ,
'MinTemperature' : raster_models[4] ,
'MeanTemperature' : raster_models[3] ,
'SolarRadiation' : raster_models[2],
'Precipitation' : raster_models[1],
'WorldPopLatam2010' : raster_models[8] ,
'myNewModel' : raster_models[9],
}
```

Suplementary materials II for the paper: *Biospytial: spatial graph-based computing for ecological big data*

2.15 Mathematical formalisms

This section gives a brief description of the mathematical and biological terms used in the paper. It also includes formalization of the data specification and some conceptual and theoretical consequences.

2.15.1 Mathematical definitions

Definition 1 (Equivalent class) *Let Ω be a set. An equivalent relation on Ω is a subset $R \subseteq \Omega \times \Omega$ that satisfies the following three properties:*

- Reflexivity*: For all $x \in \Omega, (x, x) \in R$*
- Symmetry*: For all $x \in \Omega$ and $y \in \Omega$, if $(x, y) \in R$ then $(y, x) \in R$*
- Transitivity*: For all $x, y, z \in \Omega$ if $(x, y) \in R$ and $(y, z) \in R$ then $(x, z) \in R$*

The equivalent class of an element $x \in \Omega$ is denoted as the set:

$$[x]_R = \{x \in \Omega | (x, y) \in R, y \in \Omega\} \tag{2.1}$$

Given that x and y are elements of Ω it follows that if $(x, y) \in R$ then $[x]_R \subseteq \Omega$.

Definition 2 (Partition) *Let Ω be a set and $\mathscr{A} = \{A_1, A_2, ..., A_n\}$. $\mathscr{A}$ is called a partition of Ω if and only if:*

- $\cup_{i=1}^{n} A_i = \Omega$

- $A_i \neq \emptyset$
- $A_i \cup A_j = \emptyset$ *for all* $i \neq j$

Definition 3 (Modulus) *Let* $\mathscr{F} = \{[x]_R | x \in \Omega\}$ *that is, the family of all equivalent classes in* Ω *defined by the relationship* R*. This set (*$\mathscr{F}$*) is denoted as* $\Omega \backslash R$ *and is called the quotient set of* Ω *by* R *or* Ω modulo R*.*

$\Omega \backslash R$ is a partition of Ω if and only if R is an equivalence relation. Therefore, any pair of elements A_i, A_j in $\Omega \backslash R$ (subsets of Ω) are mutually exclusive. A feature that, with the right caveats, eases the computation of probabilities using the rule of total probability. For example conditional autoregressive models use spatial lattices that partitions space in mutually exclusive areas, the aggregated measurements on each area simplifies the computing of spatial correlations in large areas (Besag, 1974).

Definition 4 (Graph or Network) *Let* $V(G)$ *be a set and* $E(G) \subseteq V(G) \times V(G)$*. A graph* G *is a duple given by* $(V(G), E(G))$*.* $V(G)$ *is the set of vertices of the graph and* $E(G)$ *is the set of edges. An example of a graph is drawn in figure: 2.1.*

Definition 5 (Subgraph) *Let* G *be a graph.* G' *is a subgraph of* G *(*$G' \subseteq G$*) if and only if* $V(G') \subseteq V(G)$ *and* $E(G') \subseteq E(G)$*.*

Definition 6 (Connected and acyclic graph) *If for every* $u, v \in V(G)$ *there exists a path that connects them, then* G *is said to be connected. If that path is unique for every* u, v *then* G *is acyclic (without cycles).*

Definition 7 (Tree) *A graph* T *which is connected and non-cyclic is called a* Tree*. An example is given in figure 2.2.*

Definition 8 (Subtree) *Let* T *be a tree. A subtree* T' *is a subgraph of* T *such that is also a tree (i.e. contains no cycles).*

2.15.2 Biological definitions

Definition 9 (Biological Species) *The following definitions are equivalent:*

- *Groups of actually or potentially interbreeding natural populations which are reproductively isolated from other such groups ((Mayr, 1940)).*
- *An inclusive Mendelian population; it is integrated by the bonds of sexual reproduction and parentage ((Dobzhansky and Dobzhansky, 1970): 354).*
- *A species is a group of interbreeding natural populations that is reproductively isolated from other such groups ((Mayr and Ashlock, 1991))*

Definition 10 (Taxonomic concept of species) '... a species consists of all the specimens which are, or would be, considered by a particular taxonomist to be members of a single kind as shown by the evidence or the assumption that they are as alike as their offspring or their hereditary relatives within a few generations. When there is no evidence of the hereditary relationship, the taxonomist will rely on distinctions that have been found to be effective in segregating species among other groups'. ((Blackwelder, 1967) : 164)

The concept of species is mostly biased by the data used. In the practical case is based in natural museum records around the world (See section on Data used and GBIF page: 35). Therefore, a more restrictive definition should be used in order to support further argumentations on evolution and ecology.

2.16 Theoretical consequences

Lemma 1 *There is a unique Taxonomic Tree of all life on Earth. This tree is called The Tree of Life.*

Proof 1 *All organisms have Common Ancestor. Because of this is possible to build taxonomic relationships based on this comparison. The Uniqueness of this common ancestor and the existence of LUA implies that: i) there is just one path that connects any pair of species (vertices) and ii) the graph is connected.*

Lemma 2 (Local Tree) *For any area in Earth it is possible to derive a unique Taxonomic Tree.*

Proof 2 *Because Life is Conspicuous it is possible to find organisms in any place. By the axioms of Common Ancestor and Taxonomic Relationship it is possible to build a taxonomic hierarchy between the group of organisms within that place. Because Axiom of LUA there is only one tree that represents these taxonomic /ancestry relationships.*

Proposition 1 *For a given area*[6] *in Earth, the taxonomic tree derived from it is a subtree of the Tree of Life.*

Proof 3 *Let T be the Tree of Life and $T(A)$ the local tree in the area A. $A \subseteq Earth$. $T(A)$ is a tree because of lemma 1.14. $T(A)$ is based on the same taxonomy given by the species in A (which are leaves in the tree) therefore all the edges of $T(A)$ are in T. The species in A is a subset of all the species in the $Earth$ otherwise the $Earth$ would not be the $Earth$ and there exist another greater set that could be called $Earth$.*

Corollary 1 *If $A = Earth$ then $T(A) =$ Tree of Life.*

Proof 4 *Let $A = Earth$. This implies that all species in A are in $Earth$ and vice versa. $V(T(Earth)) = V(Treeof Life)$ and the taxonomic chain (path) of $V(T(Earth))$ is the same as in $V(Treeof Life)$ because it is unique. Therefore, $Treeof Life = T(Earth)$*

[6] Any open set contained in the surface Earth. Earth can be considered as a compact surface embedded in $\mathbb{R}^3$

2.17 Formal data specification

This section explains the mathematical formalities of the model. For the purposes of this treatment we will call Ω the total sample. In the current implementation the GBIF dataset is the only source of information for occurrences, therefore Ω = GBIF for an arbitrary chosen snapshot (version). In general, $\Omega \subset \mathscr{B}$ where $\mathscr{B}$ is the totality of living beings in Earth (the biosphere) for a given time t [7].

Raw Occurrence Data Let $o \in \Omega$ be called an Occurrence. o has attached a set of properties $\mathscr{P}(o)$. In the case of the GBIF database, $\mathscr{P}(o)$ consists (but not exclusively) of:

- Species
- Genus
- Family
- Order
- Class
- Phylum (or Division)
- Kingdom
- Location (lat/long) (point)
- time-stamp of collection
- Unique Id

The first eight properties are called **taxonomic properties**.

[7] If it would be necessary to clarify further we will write this as Ω_t

Towards integrated modelling

The concept of *equivalence class* is foundational because the set of properties $\mathscr{P}$ give a direct classification for living beings. In any ecological study, the sample (e.g. GBIF) will always be a subset of the universal set of *Life in Earth*. Each element in the sample has certain properties like acquisition time, location and, of course, the ontological properties of each particular study (e.g. individuals within a population; plant traits within an ecosystem; pollinators and plants, vectors and diseases, etc.)

A general modelling of properties derived by *equivalence relations* can model different representations of the same phenomenon in a generic way. For example, all occurrences have the attribute *Species Name*. If the relation (x, y) is: *x is the same species as y*; we have that the relation is indeed an **equivalence relation**. Continuing through this line of thought we have that the following relations are **equivalent relations** and each one defines as well a quotient set.

Relation	Quotient Set (notation)
x:has_the_same_id_as:y	$[Id]$
x:is_the_same_species_as:y	$[Sp]$
x:is_the_same_genus_as:y	$[Gns]$
x:is_the_same_family_as:y	$[Fam]$
x:is_the_same_order_as:y	$[Ord]$
x:is_the_same_class_as:y	$[Cls]$
x:is_the_same_phylum_as:y	$[Phy]$
x:is_the_same_kingdom_as:y	$[Kng]$
x:is_a_living_being_as:y	$[Root]$

By recursion, if Ω is a partition of a larger set say, Γ, any partition (equivalence relation) within Ω is also a partition of Γ. The models for Ω will be valid for Γ also.

For example: suppose that every occurrence is an organism. Every organism is constituted by cells. If Γ is the set of all cells then clearly Ω will be a partition under the equivalence relation: *x is a cell of the same organism as y*.

The above formalization of *taxonomic objects* can continue indefinitely. An unbounded object like this will always be in a state of definition but not fully defined. A theory or methodological framework needs to be able to add-up new possible properties in which the objects could be partitioned.

Adding more properties

Suppose that a new property P is added to each element of Ω. The new property P could be any type, e.g. binary, categorical or continuous, and determines a new equivalence relation such that a new quotient set $\Omega \backslash P$ can be derived. Any new property that splits Ω in a partition is an equivalence relation.

Partial orders and semi-lattice systems

The hierarchical ordering of: *kingdom, phylum, class, order, family, genus* and *species* is based on the *natural system.* If this order acts on the entire set of species on Earth (the biosphere $\mathscr{B}$), with the inclusion of LUA (Axiom 1.5) it defines a partial order set [8].

A consequence of being a **partial order set** is that, for every species s there exists a unique chain of ordered elements that join s with a genus gn, a family f, ..., a kingdom k.e.g., The species *Homo sapiens (L. 1758)* has an ordered chain of: *H. sapiens* $\leqslant$ Homo $\leqslant$ Hominidae $\leqslant$ Primates $\leqslant$ Mammalia $\leqslant$ Chordata $\leqslant$ Animalia. A partial order set induces a semi-lattice data structure compatible with ontology specifications and the spatial lattices framework. Using both types of relations is a first approach to define graph traversals based on spatial and evolutionary relationships. This can help to analyse

[8] Ergo, the $biosphere$ is a partial ordered set. For formal definition see: (Skornyakov, 2014)

species distributions, co-occurrence relationships and statistical modelling of ecological properties.

Part II

The statistical framework

Chapter 3

A joint distribution framework to improve presence-only species distribution models by exploiting opportunistic surveys

State of publication

The following chapter is a facsimile of the draft reviewed and approved by all the co-authors. Its target journal is *Methods in Ecology an Evolution*, indexed in the Q1 Ecological Modelling catalog of Scimago Journal Rank (SJR) (2018).

Keywords: Species Distribution Models, Presence-only data, Model-based spatial autocorrelation models, Multivariate conditional autorregresive models

A joint distribution framework to improve presence-only species distribution models by exploiting opportunistic surveys

Juan M. Escamilla Molgora[a,b,1,*], Luigi Sedda[c,1], Peter Diggle[b,1], Peter M. Atkinson[d,1]

[a] *Lancaster Environment Center, Lancaster University, Lancaster LA14YQ, UK*
[b] *Centre for Health Informatics, Computing and Statistics (CHICAS), Lancaster Medical School, Faculty of Health and Medicine, Lancaster University, Lancaster LA1 4YQ, UK*
[c] *Lancaster Medical School, Faculty of Health and Medicine, Lancaster University, Lancaster LA1 4YQ, UK*
[d] *Faculty of Science and Technology, Lancaster University, Lancaster LA1 4YR, UK*

Abstract

Species distribution models (SDM)s are essential tools for predicting biodiversity loss and selecting areas for habitat restoration or conservation. Their reliability depends on the available presence and absence data. While presence data is widely available, absence data is difficult and expensive to obtain. SDMs for presence-only data have been designed to address this problem. However, they require additional assumptions about absences, which are often unrealistic or not flexible enough to reduce the bias given by the presence-only observations.
Here we propose a Bayesian framework for SDMs using presence-only biodiversity occurrences obtained from historical opportunistic surveys. The framework defines a bivariate spatial process separable into ecological and sampling effort processes that jointly generate occurrence observations of biodiversity records. Presence-only data are conceived as incomplete observations where some presences have been filtered out. A choosing principle is used to separate out presences, missing data and absences relative to the species of interest and the sampling observations. The framework provides three alternatives for modelling the spatial autocorrelation structure: independent latent variables (model I); common latent spatial random effect (model II); and correlated latent spatial random effects (model III).
To demonstrate the performance of the framework we compared it against the popular presence-only SDM model, Maximum Entropy (MaxEnt) in two examples: one for prediction of pines (Class: Pinopsida) using botanical records as sampling observations and another for prediction of Flycatchers (Family: Tyranidae) using bird sightings as sampling records. In both examples, all models achieved higher predictive accuracy than MaxEnt. Model III fit best when the sampling effort signal was informative, while model II was more suitable in cases with non-informative samples.
Our approach provides a flexible methodology for presence-only SDMs aided by a sampling effort process informed by the accumulated observations of independent and heterogeneous surveys.

Keywords: species distribution models, presence-only data, opportunistic sampling, multivariate conditional autoregressive models, model-based statistical ecology,

* Corresponding author
Email addresses: j.escamillamolgora@lancaster.ac.uk (Juan M. Escamilla Molgora), l.sedda@lancaster.ac.uk (Luigi Sedda), p.diggle@lancaster.ac.uk (Peter Diggle), pma@lancaster.ac.uk (Peter M. Atkinson)

3.1 Introduction

Species distribution models (SDMs) are statistical and computational methods for characterising the distribution of organisms across space (Elith and Leathwick, 2009; Guisan and Zimmermann, 2000). The predictive capabilities of these models allow forecasting changes in species distribution under different environmental scenarios, providing meaningful insights with which to assess biodiversity loss (Pereira et al., 2010), adaptation to climate change (Wiens et al., 2009), ecosystem management and conservation (Navarro et al., 2017) and the risk of invasive species (Jiménez-Valverde et al., 2011), amongst others. Predicting the spatial distribution of species subject to different environmental conditions is crucial for developing strategies for the management, adaptation and mitigation of human-induced impacts to the biosphere (Ferrier et al., 2016; Foden and Young, 2016; Intergovernmental Panel on Climate Change, 2014). Although this field is relatively new (see Elith and Leathwick (2009); Guisan and Zimmermann (2000) and Guisan et al. (2017) for a review) it has developed quickly in both theoretical and applied studies (Araújo et al., 2019).

SDMs use occurrence observations as the response variable(s) and environmental features (covariates) as explanatory variables. The methodological frameworks for prediction are, however, diverse. For example, generalised linear models (GLMs) and generalised additive models (GAMs) have been demonstrated to characterise natural distributions accurately if presence-absence records are available (Guisan et al., 2002) and (Keating and Cherry, 2004).

Methods based in machine learning, specifically supervised classification algorithms, have also been used to model species distributions (e.g Elith et al. (2006); Peterson et al. (2011); Segurado and Araújo (2004)). These methods include boosted regression trees (BRT, Friedman (2001)), multivariate adaptive regression spline (MARS, Friedman (1991))

and artificial neural networks (ANN, Rosenblatt (1958)). The R package `sdm` includes an exhaustive list of machine learning methods for fitting species distribution models. One of the critiques of using machine learning methods in SDMs is that they reduce the species' ecological processes into a mere classification problem and does not describe the stochastic process that generates the observations, limiting their scientific interpretability (Gelfand and Shirota, 2019; Haegeman and Loreau, 2008).

In this sense, model-based statistical methods are better fit to describe the underlying mechanisms of species distributions. In particular, joint stochastic modelling and hierarchical Bayesian models have recently been proposed to account for uncertainties in the parameters estimations and for defining more flexible random effects. For example, in cases where spatial autocorrelation is present, the use of Gaussian Processes (Golding and Purse, 2016) or Gaussian Markov Random Fields (GMRF) (Illian et al., 2013) have been shown to increase predictive accuracy. Although these models are statistically sound, their major limitation is their reliance on presence-absence data, which generally are not available. In cases where the goal is the modelling of species distributions across large geographic regions, the creation of presence-absence records requires a careful sampling design with possibly hundreds of experts deployed in the field for data collection. Surveys of this kind are atypical and usually are developed by governments or similar sized institutions that can afford full inventory or census data (e.g. forest Inventory and analysis (Smith, 2002) and Inventario Nacional Forestal (CONAFOR, 2018)).

The widespread use of opportunistic observations has been favoured by citizen science initiatives and the availability of large and open repositories like: The Global Biodiversity Information Facility GBIF (GBIF Secretariat, 2015), eBird for bird sightings (Hudson et al., 2014) and the PREDICTS database (Sullivan et al., 2009)). These records are often derived from museums, herbaria collections or unstructured citizen observations. As such, the data are often limited to presence-only observations and, therefore, do not include infor-

mation on where or when a given species was *not* found (i.e. absences). In addition, the information related to sampling design is frequently lost, or does not exist, and the data itself are prone to several sources of bias in space, time, and detectability among species and habitats (Beck et al., 2014; Dickinson et al., 2010; Franklin et al., 2016; Isaac and Pocock, 2015). Despite the inevitable problem of their sampling bias, presence-only observations contain valuable information about species distributions and, therefore, several modelling frameworks for presence-only data have been proposed for such purposes.

With the exception of some unrealistic assumptions about the absences on presence-only models (e.g. assuming that absence of evidence is equivalent to evidence of absence), estimating the probability for species occurrence using solely presence-only observations involves a problem of model identification (Ward et al., 2009). That is, the model has multiple solutions and is not possible to make reliable inferences. This problem has lead to recognise the importance of incorporating other sources of information into SDMs based on presence-only data.

One of the earliest methods is the Maximum Entropy (MaxEnt) algorithm (Phillips et al., 2006b) for predicting occurrences based on the density of environmental covariates conditional to the known species presences using background data that serves as pseudo-absences. The MaxEnt algorithm reduces predictions to an optimal density distribution calculated with a constrained optimization algorithm, denying accountability for uncertainties related to the optimised distribution and the specification of other random effects. Despite this, it has shown to perform well in practice (Elith et al., 2006) and is still one of the most widely used methods for predicting species distributions (> 2600 articles in Web of Science at the time of writing).

Phillips et al. (2009) recognised the effect of the sampling bias in presence-only distribution models and proposed the use of occurrence records of other species that are have been collected using the similar methods (called a "target group" in the sense of Phillips

et al. (2009)). In their work, they proposed a joint model for accounting the sampling bias and implemented their methodology in three generic types of models: GAMs, MARS, BRTs and Maxent. Their conclusion was that using and informed background data (one that potentially shares same characteristics of the sampling process) significantly improves the models' accuracy.

The use of joint modelling methods for accounting sampling bias has been addressed by other authors. For example, the expectation maximization algorithm for estimating underlying presence-absence processes (Ward et al., 2009) aims to infer the underlying presence-absence logistic signal of the data used as presence-only observations. This approach does not account for spatial dependencies. The occupancy model proposed by Royle and Kéry (2007) specifies a hierarchical Bayesian model for accounting the joint effect of two components, one for partially observed occupancy and other for the observations conditional on that process. Inconveniently, their model is suited for longitudinal data (i.e. time series) and does not account for any spatial effect.

In this regard, the framework developed by Pacifici et al. (2017) accounts spatial dependencies in both components, one for presence-only data and other based on presence-absence. Both proposals do not allow the explicit modelling of the preferential sampling (i.e sampling effort process), with fixed and random effects. Another modelling framework that integrates sampling effort and an ecological process was proposed by Croft et al. (2019) to model future scenarios of distribution models. These models had advance the presence-only SDMs in many aspects. However, a unified spatial statistical framework for species distributions using presence-only data for spatial lattices (i.e. data aggregated on a grid), has not been proposed yet. We consider that a framework of this kind with the capability for jointly modelling the sampling effort and the ecological processes using a flexible design for defining missing data can contribute to a greater predictive accuracy by exploiting citizen science effort.

We present a statistical framework for modelling species distributions using presence-only data. We assume that the registered occurrences of a taxon of interest (ToI) are incomplete observations of a bivariate process that includes information about the ecological suitability (i.e. where the ToI can live) and complementary occurrence data that serve as a proxy for sampling effort, providing information on how the observations were recorded. The framework specifies three hierarchical bayesian models that jointly specifies the ecological and sampling processes. The approach provides a full description of the data generating process, giving a more direct interpretation of the parameters as well as giving explicit estimates of their uncertainties. The presented model assumes that the species populations are static in time and in equilibrium with the environment (in the sense of Guisan and Zimmermann (2000)). Therefore, this model does not differentiate between sink populations or populations with sustained growth.

The paper is structured as follows. Section 3.2 describes the general specification of the frameworks. Here, we develop a logistic hierarchical model defined as a bivariate process that accounts for spatial random effects. Our most general model (full description in appendix: 3.9.3) includes a latent bivariate spatial process with correlated components. We also consider two extreme special cases: in model I (appendix: 3.9.3) the two component processes are independent; in model II (appendix: 3.9.3) they are proportional. In section 3.3 we propose two study cases for predicting presences of Pines (class: *Pinopsida*) and Flycatchers (family: *Tyrannidae*). The prediction analysis is described in sections 3.4.1 and 3.4.2, respectively. We compared the framework using the three models with the MaxEnt algorithm as a standard benchmark. Finally, section 3.5 discusses the methodology, caveats and future research.

3.2 Materials and Methods

As presence-only data lack real absences, there exists no knowledge on whether the absence of data is due to the inaccessibility of a potential sampling location or the real absence of the taxon of interest (ToI). This ambiguity suggests that presence-only data provide incomplete evidence of two underlying processes acting together. A process P_Y that generates the ecological phenomenon of a taxon's occurrence, and a process P_X associated with the sampling effort or survey. As such, locations with no records of the ecological phenomenon or sampling effort indicates incomplete or missing information. Our proposal is an attempt to model these two processes using a hierarchical Bayesian framework with the aim to predict probability of occurrence for a ToI using presence-only data under different configurations of the spatial autocorrelation of X and Y.

3.2.1 Model summary

In general, the framework specifies a Bayesian hierarchical model that accounts for the joint effect of two components; an ecological process (P_Y), that drives the occurrence of species of interest in the study region, and a sampling effort process (P_X) that models how the occurrence data were sampled. Each stochastic process include a structural component (fixed effect) and a random effect that includes the specification of spatial autocorrelation. The model is defined in a discrete spatial lattice. Consequently the estimations are also discrete and are defined in each area element of the lattice. The support of the model is the area element.

The presence-only data is assumed to represent realizations of a bivariate stochastic binary process (Bernoulli) separable in two components: one relative to an ecological process P_Y that drives the environmental suitability for the ToI, and another process P_X related to the sampling effort. P_X and P_Y are modelled according to the following

equations:

$$\log\left(\frac{p_y}{1-p_y}\right) = d_Y^t \beta_Y + r_y \tag{3.1}$$

$$\log\left(\frac{p_x}{1-p_x}\right) = d_X^t \beta_X + r_x \tag{3.2}$$

where d_X and d_Y represent vectors of explanatory variables and r_X and r_Y the random effects for X and Y, respectively. Specifically, d_Y is suited for environmental variables of ecological importance, while d_X should account for variables that help explain the sampling process.

The data used to fit both processes includes information on known occurrences of the ToI, the sampling effort and missing observations. To predict the probability for sites with missing data, we use the *data augmentation* scheme proposed by Tanner and Wong (1987) and implemented by Lee (2013) in the R-Cran package *CARBayes*. The approach generates posterior samples of X and Y as well as the latent variables related to processes P_Y and P_X in all locations, including the ones with missing observations (i.e. $\widetilde{X}$ and $\widetilde{Y}$).

The full model specification is explained in the supplementary materials 3.9.

Three models for spatial variation

The proposed framework assumes that the ecological process P_Y and the anthropogenic sampling process P_X are conditionally independent given the random effects R_Y and R_X. Figure 3.1 show the model structure while a detailed description of the framework specification is in the supplementary materials 3.9.

The spatial random effect are described by components S_Y (ToI) and S_X (sampling effort). The only source of dependency between R_Y and R_X is the dependency between these spatial components. In addition, each random effect incorporates an independent component for modelling unstructured variation, namely variables Z_Y and Z_X, corre-

sponding to R_Y and R_X respectively. The framework assumes that the observations of presence for the ToI and the existence of the survey (sampling) are independent when conditioned to the spatial effect. As such, the spatial autocorrelation structure is responsible for informing both processes. To test for this effect we designed three possible models in which the spatial processes S_Y and S_X inform R_Y and R_X. Model I where S_Y and S_X are independent, model II with one shared spatial process ($S_X = S_Y$) and model III where S_X and S_Y are correlated components. Schematics of the directed acyclic graphs (DAG) describing the three models are reported in figure 3.1, while the full description of the framework is described in supplementary materials 3.9.

We are aware that estimating real probability of occurrence using presence-only data is not possible given the inherently sampling bias of these type of data (e.g Guillera-Arroita et al. (2014)). Along this text, we refer to *ecological suitability* as the spatial variation across space that determines a species to live, settle or occupy a given area. This definition disregards the scale of the given value for a particular area. In other situations, we use the term *probability of occurrence* to account for the spatial variation of the ecological process (i.e. ecological suitability) in a probabilistic context, that is, where the spatial variation ranges in values from 0 to 1. To exemplify this compare the range in values of the latent variable S_Y (spatial effect) to those of the ecological process P_Y. Values in P_Y are range only within the $[0,1]$ interval.

Selection of explanatory variables

Our framework is based on the Grinnellian definition of ecological niche, that is, a niche defined by non-interactive and non-consumable (scenopoetic) variables with environmental conditions changing smoothly and coarsely in space (Soberón, 2007). The selection of these explanatory variables (covariates) are crucial for the interpretability of the model and, although, the general specifications for P_X and P_Y are mathematically similar (eqs.

3.10 and 3.11), they describe very different processes. P_Y models the ecological suitability for a ToI to occupy the area under study. Therefore, its associated explanatory variables (d_Y) should be of ecological interest. Examples of these variables are: temperature, precipitation, evapotranspiration, elevation, slope and vegetation cover. On the other hand, P_X models the probability of a ToI to be sampled, given that it has been observed. This process is assumed to be independent from the ecological suitability and it is fully determined by anthropic variables such as: distance to closest road, population density, infrastructures, political borders or land use type. The selection of covariates depends on the nature and specificities of each problem and research question. Therefore, the classification between anthropic and ecological variables is not necessarily mutually exclusive.

3.2.2 A Choosing Principle for obtaining presences, relative absences and missing observations

Estimating the probability of occurrence using solely presence-only observations necessarily requires additional assumptions about non-existent absences (Ward et al., 2009). Thus, any non recorded presence of the taxon of interest (ToI) can potentially be a real absence (i.e. the area is not inhabited by the ToI) or an unobserved presence (i.e. the ToI inhabits the area but there is not record about it). The fundamental concept of this work is to use occurrence records of other taxa that are considered to share a similar sampling pattern as the ToI. These occurrences are used to model a sample effort process that informs about the presence and absence of the taxon of interest.

Models I, II and III specify a joint bivariate process that uses two vectors of observations as inputs; one (Y) for fitting the ecological process (P_Y) and other (X) for fitting the associated sampling effort process (P_X). These input vectors (hereafter called *response vectors*) are composed of k entries, one for each area element of the spatial lattice. Each entry has assigned one of three possible values: 1, for defining the *presence*, 0 for defining

an absence, relative to a surrogate taxa that informs about the sampling effort (hereafter called *relative absence*), and (N.A) for *missing data* (also called *missing observations*); where there is no information about the presence of the ToI nor the surrogate taxa. As such, each of these values correspond to a presence-absence state on each area element of the spatial lattice.

As we are using exclusively occurrence data we need an algorithm for deriving response vectors X and Y from presence-only records. We call this algorithm the *choosing principle* and receives two lists as inputs: *target* ($\dot{\boldsymbol{t}}$) and *background* ($\dot{\boldsymbol{b}}$)). These lists are obtained by checking the existence of an occurrence on each area element of the spatial lattice. That is, if on a given area, there exists at least one record inside, assign a 1, otherwise assign a 0. This procedure is repeated on all the areas of the spatial lattice, therefore $\dot{\boldsymbol{t}}$, $\dot{\boldsymbol{b}}$ and consequently, X and Y have k elements. The lists $\dot{\boldsymbol{t}}$ and $\dot{\boldsymbol{b}}$ are transformed by the choosing principle into a response vector with presence, relative absences and missing observations. The resulting response vector (X or Y) would depend on the selection of $\dot{\boldsymbol{t}}$, $\dot{\boldsymbol{b}}$. To put it simply, the choosing principle defines the missing data for X and Y, given a list of presences and absences of records.

There are many possibilities to define a choosing principle. Here we use one that assigns: missing data (N.A.) to locations where neither the background nor target observations are present, 0 to locations where there is no presence of a target observation (i.e. $\boldsymbol{t}_i = 0$) but has a background observation (i.e. $\boldsymbol{b}_i = 1$), and 1 to locations where presences of both, target and background exist. Algorithm 1 describes this particular case of the *choosing principle*.

Obtaining response variables X and Y

The response vector X is obtained, first, by defining the list of occurrences to be used as the target list in algorithm 1. That is, the observations of the surrogate taxa that inform

Algorithm 1 Choosing principle: Obtaining a response vector R using background $\dot{\boldsymbol{b}}$ and target observations $\dot{\boldsymbol{t}}$ over a spatial lattice composed of K area elements. Binary values are: 1 if there is at least one registered occurrence, and 0 otherwise. The symbol $N.A$ (*Not a number*) is assigned to missing values.

Require: $\dot{\boldsymbol{b}}$ and $\dot{\boldsymbol{t}}$
  for ($i := 1$ to $i == K$; $i++$) **do**
    if $\dot{b}[i] == 1$ **then**
      if $\dot{t}[i] == 1$ **then**
        $R[i] \leftarrow 1$
      else
        $R[i] \leftarrow 0$
      end if
    else
      $R[i] \leftarrow$ NaN
    end if
  end for

about the sampling effort of the taxon of interest (ToI). We define this list of observations as the *informative sample* ($\dot{\boldsymbol{x}}$). It accounts for the presence of a taxon (or group of taxa) different from the ToI but known to be associated with its presence. The informative sample should be chosen accordingly to the particularities of the ToI, that is, one that gives meaningful information related to the real presence of the ToI. The *background* observations for X (i.e. $\dot{\boldsymbol{b}}$, input of algorithm 1) are defined as all known presence-only records of any taxonomic group in the spatial lattice. In this sense, X also supports missing data, corresponding to areas that have never been sampled.

The response vector Y is obtained similarly by assigning the presence observations of the ToI ($\dot{\boldsymbol{y}}$) to the target $\dot{\boldsymbol{t}}$ and using the informative sample of X (i.e. $\dot{\boldsymbol{x}}$) as background ($\dot{\boldsymbol{b}}$). Along this text we refer to $\dot{\boldsymbol{y}}$ and $\dot{\boldsymbol{x}}$ to differentiate between the target lists (i.e $\dot{\boldsymbol{t}}$) used by Y and X respectively. Additionally, we refer to set of missing data for X and Y with the symbols $\widetilde{X}$ and $\widetilde{Y}$. Table **??** includes definitions of all the terms and symbols used in the methods and application sections.

The selected choosing principle is reasonable from an ecological view. If, on average, the existence of X informs the occurrence of Y, we can argue that: if a site i has no

background information, the probability of X and Y is unknown and it is informed only by nearby sites. If on the other hand, the background information exists, but there is no known occurrence (i.e. a *relative absence*) of Y at area i, the probability of occurrence for Y will depend on the presence of X as well as its nearby areas. In this sense, the probability of occurrence of a taxon (e.g. species) depends on the presence, its relative absence, its sampling effort and the nearby areas where the taxon is present. The next section shows two practical examples.

3.3 Applications

To show the capabilities of the framework we chose two examples for predicting presences. The first involves predicting the presence of pines, that is, occurrences of the class *Pinopsida* as the process P_Y (*Pines*) using the available botanical records and occurrences of the kingdom *Plantae* as the sampling process P_X (*Plants*). The second example predicts the presence of a relatively abundant family of flycatchers (family: *Tyrannidae*) as the process P_Y (*Tyranids*), using the available records of birds (class *Aves*) as the sampling process P_X (*Birds*). In both cases we chose *Elevation* and *Precipitation* as the scenopoetic variables for process P_Y and *Distance to roads* and *Population density* as the anthropological variables for process P_X. Following the model specification in equations 3.10 and 3.11 (supplementary materials 3.9) The model for the examples of *Pines* and *flycatchers* is defined as the joint Bernoulli process.

$$\begin{cases} \text{logit}(\text{ToI})_k = \beta_{Y_0} + \beta_{Y_1}(\text{Elevation})_k + \beta_{Y_2}(\text{Precipitation})_k + S_Y + Z_Y \\ \text{logit}(\text{Sample})_k = \beta_{X_0} + \beta_{X_1}(\text{Population density})_k + \beta_{X_2}(\text{Distance to roads})_k + S_X + Z_X \end{cases} \tag{3.3}$$

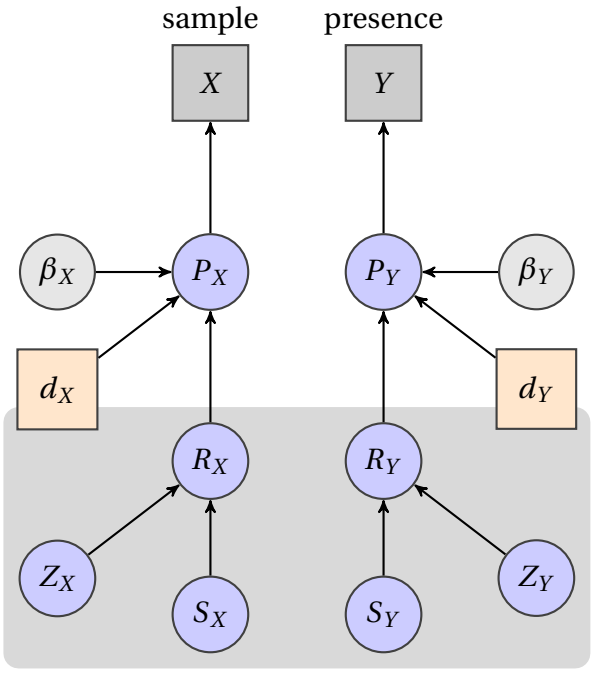
Table 3.1 Definitions of the used terms and symbols

Symbol / term	Definition
response vector	vector input, each entry could be a presence, absence or missing data
occurrence	a presence entry (1) in a response vector
relative absence	entry for absence (0), relative to the presence of an external response vector
missing observation	an entry (N.A) in a response vector with no information about presence or relative absence
Y	response vector of the taxon of interest
X	response vector of sample observations
$\widetilde{Y}$	missing observations contained in the response vector (Y). These values are parameters and are sampled by the MCMC procedure
$\widetilde{X}$	missing observations contained in the response vector (X). These values are parameters and are sampled by the MCMC procedure
P_Y	latent variable for ecological process
P_X	latent variable for sampling effort process
r_Y or (R_Y)	random effect (latent process) for the ecological process
r_X or (R_X)	random effect (latent process) for the sampling process
S	spatial process, a component of the random effect
Z	unstructured random effect, normal distributed
target ($\dot{\boldsymbol{t}}$)	input (presence-only) data, used by the choosing principle to derive the response vector of the ecological process (Y)
informative sample ($\dot{\boldsymbol{x}}$)	input (presence-only) data, used by the choosing principle to derive the response vector of the sample process (X)
background ($\dot{\boldsymbol{b}}$)	input (presence-only) data used by the choosing principle to define entries of relative absence or missing data

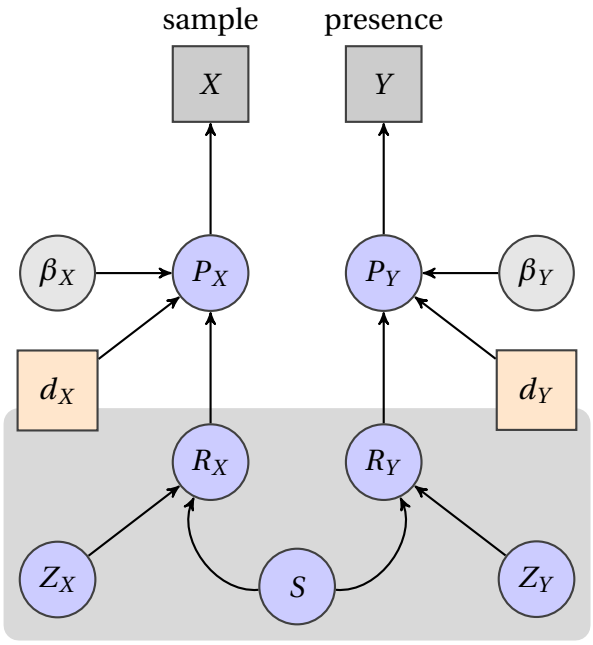
Where ToI indicates that the equation is used for the taxon of interest (i.e. pines or flycatchers) and Sample indicates that the equation is valid for the sampling effort (i.e. plants or birds).

3.3.1 Study region

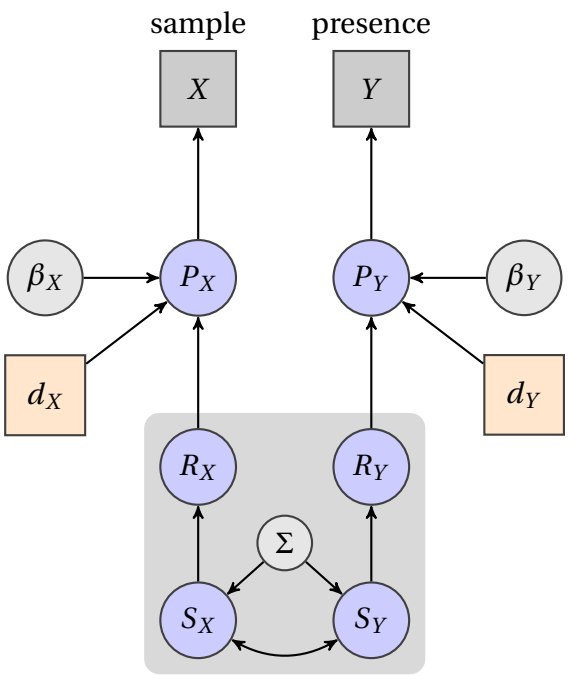
Both models were fitted to data from the same study region. The region comprises the inland area of a circular polygon centered in central-eastern Mexico at 19N −97E with radius of 2° (ca.~ 200 km). The area covers approximately 112,000 km^2 and intersects several Mexican states including: Veracruz, Puebla, Tlaxcala, Hidalgo, Mexico City, Morelos and Oaxaca (see figure 3.2 (i)). It includes heterogeneous landscapes with variability in biodiversity, geomorphological and climatic features. The region also includes distinct biomes such as: coastal dunes, chaparrales, mesophyl forests, evergreen rainforest, grasslands, mangroves, broad leaf forests and coniferous forests (Rzedowski, 2006) and (INEGI, 2015). The circular polygon was intersected on a grid of 4 km spatial resolution to obtain a lattice $\mathbb{W}$ composed of 4061 areal units. This lattice was used to define the spatial structure in models I, II and III.



(a) Model I: Independent processes



(b) Model II: Common spatial effect



(c) Model III: Correlated spatial effects

Fig. 3.1 Directed acyclic graphs for the three model specifications. Variables in squares account for observations: Y : presence of a taxon of interest (e.g. species) and X : presence of sample. Circles in blue correspond to latent variables while circles in grey correspond to parameters. Variables P_X and P_Y correspond to the latent processes of the sampling effort and ecological suitability, variables R_X and R_Y correspond to the random effect for the sampling effort and the ecological suitability processes respectively. Variables β_X and β_Y represent the parameters of the fixed effects (linear components) of the latent processes P_X and P_Y respectively. Squares in salmon colour indicate environmental (d_Y) and anthropic (d_X) explanatory variables. The variables inside the dark grey block define the random effects component; different in the three models. Variables S, S_X and S_Y describe the spatial component defined as Gaussian Markov Random Fields, while variables Z_X and Z_Y represent unstructured variability within an area.

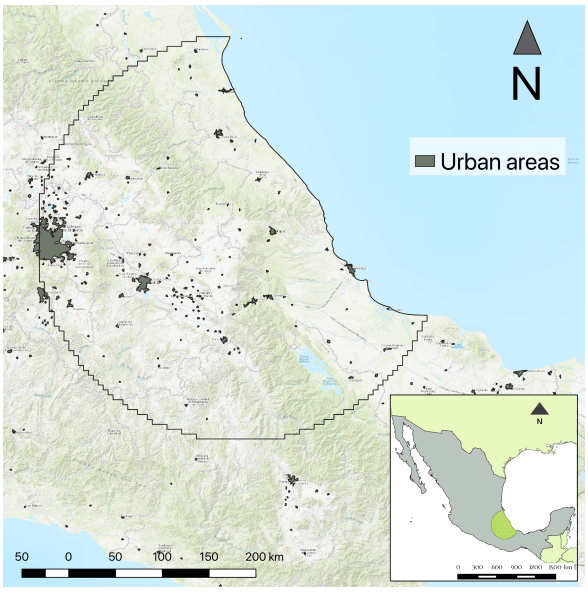


Fig. 3.2 A map showing the study area (overlaid semicircular polygon) over central Mexico. Important cities are shown as grey polygons scattered across the area. Greener areas represent higher vegetation cover. The basemap used as background was obtained from the ESRI topographic tiling service.

3.3.2 Occurrence data

For the presence-only data we used the available GBIF occurrence data (GBIF Secretariat, 2015) registered before January 2015, constrained to the region $\mathbb{W}$. The raw data was downloaded from the GBIF portal with the catalog id: DOI:10.15468/dl.oflvla . Upon downloading, we performed a minimal data cleansing to remove records with missing information in any of the seven taxonomic ranks (i.e. kingdom, phylum, class, order, family, genus and species), acquisition date and collection code. We kept occurrences with identical coordinates as, historically, these occurrences might represent distinct different records collected in a common study area. Further information of this dataset, including all data attributions can be found in (GBIF.org, 2016).

We aggregated the occurrence data following the *choosing principle* described in subsection 3.2.2 to obtain response variables $\dot{\boldsymbol{y}}, \dot{\boldsymbol{x}}$ according to each example. The aggregation was by the class *Pinopsida* and kingdom *Plantae*, in the *Pines* example and, by the family *Tyrannidae* and class *Birds* for the *Tyrannids* case. Both examples used all known living records (*Life*) as background signal $\dot{\boldsymbol{b}}$. The taxonomic classification structure used was the GBIF Taxonomic Backbone (GBIF Secretariat, 2017).

3.3.3 Treatments for missing data

To assess the impact of using missing information in the prediction accuracy of the framework, we established two different treatments for fitting each model on each example. These treatments are defined as follows:

- treatment *i*: response vectors for the ToI (Y) and the sample (X) have missing data (i.e. $\widetilde{X} \neq \emptyset \neq \widetilde{Y}$).
- treatment *ii*: only the sample response vector (X) has missing data. That is, $\widetilde{X}$ is the only source of missing information.

The motivation of using treatments is that they can serve as a middle hypothesis to assess the performance of the framework under scenarios with different proportions of missing data. The recommended scenario for use in practical applications is to use treatment *i*. The chosen measure of model performance within treatments is the ROC-AUC estimate. Using this estimate as an absolute measure between models may lead to wrong conclusions. For example, treatment *ii* implies that all the absences of Y are real and the sample X provides no information in the data augmentation methodology and therefore resulted in lower variance. This may lead to the conclusion that treatment ii performed better, and has greater predictive accuracy than treatment i. This conclusion would be true only under the assumption that the absences of the sampling effort are in fact true absences, which, in the case of presence-only data is false. Therefore, the comparison of presence-only models using the AUC-ROC estimate is only valid as a relative measure within models that used the same data, as it penalises models that estimate potential distributions (e.g treating absences as missing information) whilst favouring those that model realised distributions those where absences are informative) (Jiménez-Valverde, 2012). Comparing the AUC makes sense only when they are conditioned to a specific treatment and not between treatments.

3.3.4 Explanatory variables

The elevation data used were obtained from the Global Relief Model *ETOPO1* at 1 arc-minute resolution (Amante and Eakins, 2009). The precipitation data were obtained from the World Climatic Data *WorldClim* version 2 (Fick and Hijmans, 2017). The data are composed in a 12 band raster model with c.a 1 km spatial resolution averaged in monthly values from the years 1970 to 2000 (each band corresponds to a month). Note that the cells in the spatial lattice do not have the same resolution as the climatic data and a process for data homogenisation was required. To get a scalar value on each grid cell of the lattice

we first, extracted all the raster values contained within the grid cell (a layer stack) and then performed the average across all 12 bands. A similar approach was used for the rest of the raster data. The distance to road dataset was generated in two steps. First we rasterised the National Road Network for Mexico (*Red Nacional de Caminos* (RNC) INEGI, Instituto Mexicano del Transporte and Gobierno de Mexico (2014), scale: 1 : 250000) at 1 km spatial resolution. Later, we used this raster dataset to calculate its proximity to the closest road (pixels flaged as road) using the function `gdal_proximity` delivered as a standalone command-line utility from (GDAL/OGR Contributors, 2018). The road network data were obtained from: Vázquez (2018). The population dataset was obtained from the WorldPop project (Sorichetta et al., 2015) for the year 2010. The dataset consists of population counts on each areal unit, each with a spatial resolution of 3 arc-seconds (c.a 100 m).

3.3.5 Data preprocessing

The occurrences, scenopoetic and anthropological data were spatially overlaid and aggregated on each areal unit of $\mathbb{W}$. The aggregation method differed according to the data type. Mean and standard deviation were used for continuous variables, mode for categorical variables and the logical AND for binary data ($\dot{\boldsymbol{y}}, \dot{\boldsymbol{x}}$ and $\dot{\boldsymbol{b}}$). The data pipeline for processing the data was undertaken with *Biospytial* (Escamilla Molgora et al., 2020a) a geospatial knowledge engine for processing environmental data https://github.com/molgor/biospytial.

3.3.6 Inference and prediction

We used a customised version of the R package *CarBayes* (Lee, 2013) and adapted it to fit models I, II and III. It includes a wrapper for easily fitting SDMs using one of the three models proposed using any type of fixed effects. The code is available from: https://github.

com/molgor/CARBayeSDM. The package fits the model with a Markov Chain Monte Carlo (MCMC) method using a combination of Gibbs sampling and the Metropolis-adjusted Langevin Method (MALA), (Roberts and Tweedie, 2006). The posterior distributions were sampled by running 10000 iterations (using 5000 for burn-in) and a thinning interval of 5. Prediction for sites with missing information was done by sampling the posterior distributions of $\widetilde{X}$ and $\widetilde{Y}$. This same configuration was used in models I, II and III.

3.3.7 Comparison between models

Models I, II and III were compared with the *Deviance Information Criterion* (DIC) (Spiegelhalter et al., 2002). The DIC accounts for the number of parameters used and the likelihood of the observed data, given the statistical model assumed to be generating the data. The DIC is a generalisation of the Akaike information criterion (AIC) for hierarchical models, both measure the quality of the models in terms of their accuracy and parsimony. The DIC also serves as a Bayesian-based model selection tool. Model A is preferred to model B if its DIC value is lower than the one for B (i.e $\text{DIC}_A < \text{DIC}_B$).

3.3.8 Comparison against Maxent

As mentioned in the introduction, we used the maximum entropy (MaxEnt) algorithm (Phillips et al., 2006b) as a benchmark to compare the prediction accuracy of the proposed models. Contrary to models I, II and III, MaxEnt does not have a hierarchical specification and, therefore, calculating a DIC for model comparison is not possible. To address this limitation, we used a seven-fold cross-validation methodology for measuring the quality of the predictions of all models. That is, on each fold, 1/7-th of the data was excluded from the fitting process and used as testing data to be compared against the corresponding predictions. This procedure was performed seven times, until every observation had a corresponding predicted value. We then used the *receiver operator characteristic* (ROC)

curve and its area under the curve (AUC) (Fielding and Bell, 1997) as a measure of prediction accuracy. The same seven-fold cross validation was performed for models I, II and III with the difference that the excluded data were treated as missing data. The ROC / AUC values were calculated with the R package pROC (Turck et al., 2011).

Recalling that the proposed models are based on a spatial lattice structure (i.e. a CAR-based model), the spatial variation is modelled on a finite set of areal units. In the following case studies, these units were defined as square cells on a regular grid of approximately 4 km of spatial resolution. To make a fair comparison, we used the same spatial resolution and environmental values for fitting the MaxEnt models. Additionally, the background data (i.e. *pseudo-absences* in the MaxEnt jargon) used for fitting MaxEnt were obtained from locations with sampling observations but with no record of the taxon of interest, similarly to the sample selection bias for background data proposed by (Phillips et al., 2009). In other words, the *choosing principle* was also applied to the MaxEnt models resulting in the same input for all models (only valid for component Y (presence) of models I, II and III).

MaxEnt optimisation

MaxEnt allows different configurations for model fitting. The most important are: the regularisation factor (reg) and the composition of mathematical transformations of the covariates, so-called *features* (see: Merow et al. (2013)). These features are equivalent to functions of the trend (i.e. they modify the fixed effect). To optimise the predictions of MaxEnt, we ran the 7-fold cross validation using different combinations of regularisation factors ($\text{reg} \in (0.1, 150)$) and feature functions. In the case of the features, we used single and paired combinations of each of the following types: linear (l), quadratic (q), product(p), threshold (t) and hinge (h). The total number of different combinations (i.e models) for MaxEnt was 2250. The model was fitted with the R package maxnet (Phillips et al., 2017).

3.4 Results

3.4.1 Presence of Pines

We performed the methods described in section 3.2.2 to obtain response variables for Pines (*Pines*) and the botanical sample (*Plants*) using a geographical lattice $\mathbb{W}$ composed of 4060 cells (or unit areas). For the presence observations, 341 (8.4%) cells have known occurrences (class *Pinopsida*), 2559 (63%) have relative absences and 1160 (28.6%) are unknown (locations with missing observations). For the sample observations (botanical records), 2900 (71.4%) cells have known occurrence, 430 (8.4%) have relative absence and 730 (18%) unknown information (missing data).

The optimal MaxEnt, in terms of its higher predictive accuracy measured by the AUC-ROC was the one with a hinge feature type (nknots=50) and regularisation factor of 0.5. This combination, however, achieved the lowest predictions AUC of 0.67 $\pm(0.64, 0.7)$95% confidence interval (CI), when compared with models I, II and III (see figure 3.4a). Results from the best MaxEnt model and Models I, II and III are described in table 3.2.

For the treatment *i* (i.e. with both sources of missing information, see section 3.3.3), Model III (the one with correlated spatial structures) resulted to be the best ranked, that is, it achieved the lowest *Deviance Information Criterion* (DIC of 3440.2, see table 3.2). The predictive accuracy of this model, measured as the area under the ROC curve (i.e. AUC-ROC) was the highest of all three models (see figure 3.4a). The AUC of the three models fell within a common 95% credible interval of [0.8,0.86], that is, the predictive accuracy of models I, II and III was not significantly different.

Treatment *ii* (i.e. the one with no missing data in the sample effort component) produced slightly different results. In this case, Model I (independent spatial effects) was the best ranked by achieving the lowest DIC value (3421.2). The AUC in all models was higher than those on treatment *i*. However, in a similar way all of these values fell

within a common 95% credible interval of [0.85, 0.89] (see supplementary materials fig: 3.11). Possible reasons for this effect are explained in the next section. Additionally, the ROC curves in all models show similar variance described as the envelope of the ROC curve. Figures of this has been left to the supplementary materials (fig: 3.11). The

Table 3.2 Comparison of the presence-only models: Independent Spatial Components (Model 1), Common Spatial Component (Model 2), Correlated Spatial Components (Model 3) and Maximum Entropy (MaxEnt) for the presence of Pines (class *Pinopsida*) using botanical records (kingdom: *Plantae*) as sample effort. A 7-fold cross validation was performed to calculate the area under the receiver-operating characteristic curve (ROC-AUC) as a measure of quality for each model. Models with the $\star$ symbol were fitted using only missing data from X (sample), i.e. treatment *ii*.

	DIC	ROC-AUC	95% C.I	DIC*	ROC-AUC*	95% C.I$\star$
Model I	3517.6	0.835	[0.81, 0.86]	**3421.2**	0.874	[0.85,0.89]
Model II	3665.9	0.826	[0.8,0.85]	3647.9	0.877	[0.86, 0.89]
Model III	**3440.2**	0.832	[0.80,0.85]	3505.9	0.876	[0.86,0.89]
MaxEnt	–	–	–	–	0.67	[0.64,0.7]

framework allows testing the significance the model's parameters, in the same form as a Bayesian linear regression. In this sense, the variable *distance to road* was found to be the only significant covariate common to models I, II and III. That is, the zero is out of the 95% credible intervals (CI) of its posterior distribution. The scenopoetic variables (elevation and precipitation) were only significant in Model II. The selection of these specific covariates was based solely to demonstrate the capabilities of the model. As such, other covariates with stronger significance may be used further applications.

Spatial results

Figure 3.3 shows the mean predicted latent surfaces for the presence of Pines P_Y and sampling effort P_X in all three models (left and right columns resp.). P_X shows higher probability of occurrence than P_Y across all the region. This is consistent in the three models. In contrast, the presence P_Y revealed clustered patterns of high probability (figure 3.3). Of particular interest is the central zone that shows a high probability of occurrence. This area corresponds to the contact between the Eastern Sierra Madre and the Volcanic

Axis and is of high elevation and high precipitation. In contrast, the MaxEnt model (fig: 3.3, bottom left panel) produced a smoother surface. The orographic features are more defined and the clustered patterns for presence are lost. Visual comparison between the models is difficult because of their similarity. However, in treatment ii (only one source of missing observations), Model II shows the compromise of estimating the sample P_X to satisfy a common spatial component with P_Y. In Model III, the median correlation obtained from the cross variance (Σ), between the presence of pines (P_Y) and the sampling effort (P_X), was 0.97 with $(0.9, 0.99)$ 95% credible interval. This result is consistent with the fact that the taxon of interest (i.e. pines) is totally contained in the sampling effort (i.e. plants). The complete estimates summary can be checked in supplementary section 3.10.

3.4.2 Results for the Presence of Flycatchers (family *Tyrannidae*)

This example was performed in the same study region (i.e., across the lattice $\mathbb{W}$). However, the data availability was significantly different and, therefore, the results were also different. In this example we obtained 596 (14.6%) cells with known occurrences of flycatchers, 368 (9.1%) with relative absences and 3096 (76.2%) of unknown or missing information. The occurrences for the sample (birds in general) was composed of: 990 (24.4%) known occurrences, 2340 (57.6%) relative absences and 730 (18%) missing data.

The optimal MaxEnt, in terms of its higher predictive accuracy measured by the AUC-ROC was the one with a combination of feature type of linear and threshold (nknots=50), and a regularisation factor of 0.7. The resulting optimal combination achieved a ROC-AUC of 0.61 $\pm(0.59, 0.63)$95% confidence interval (CI). The optimal parameter combination resulted to be equivalent to models I and III in terms of its predictive accuracy. That is, all the MaxEnt models are covered by the 95% confidence intervals of the ROC-AUC estimation for models I, II and III. Nevertheless, Model II (the one with a common spatial random effect) resulted to be significantly more accurate than the rest of the models.

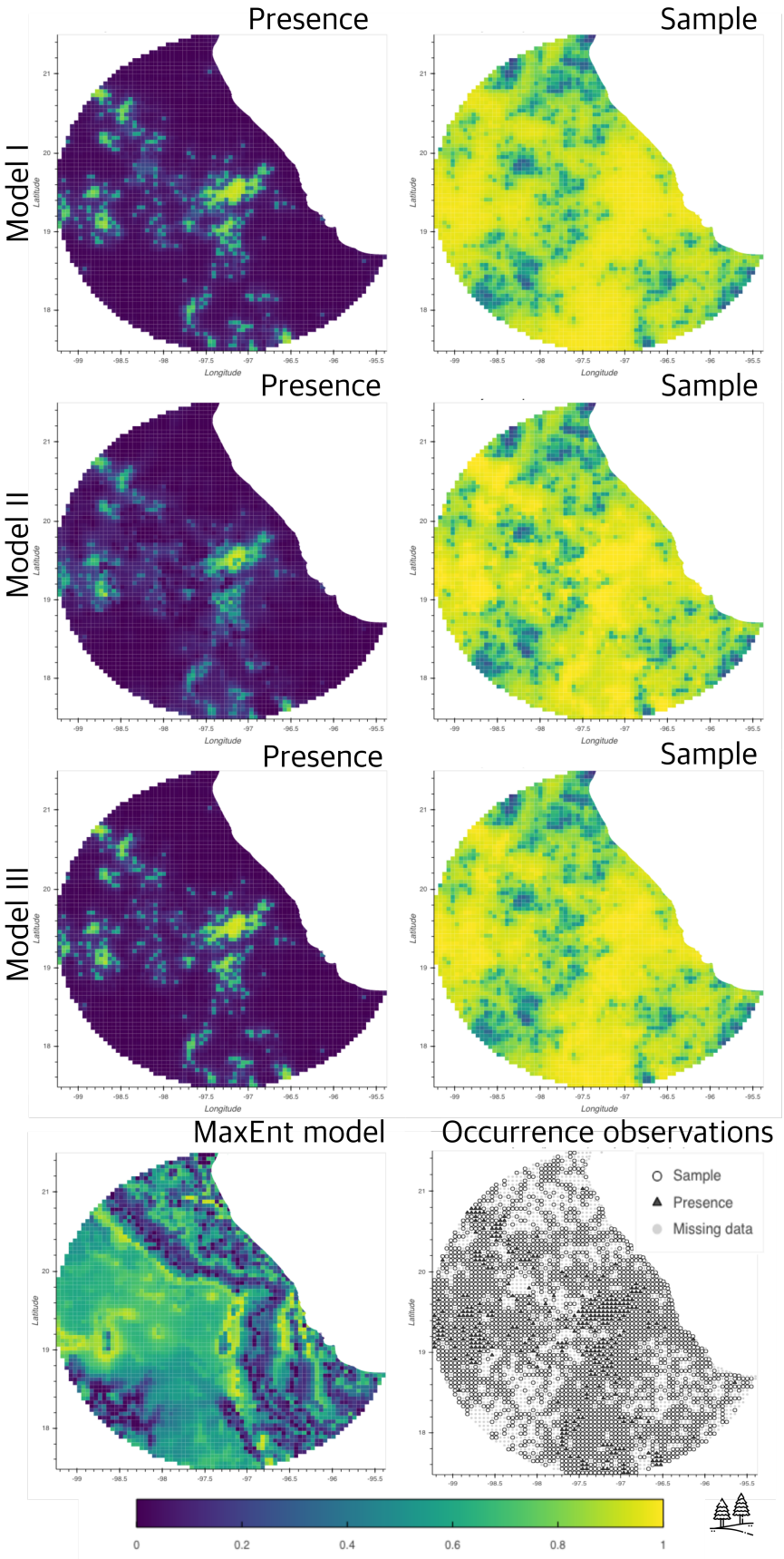
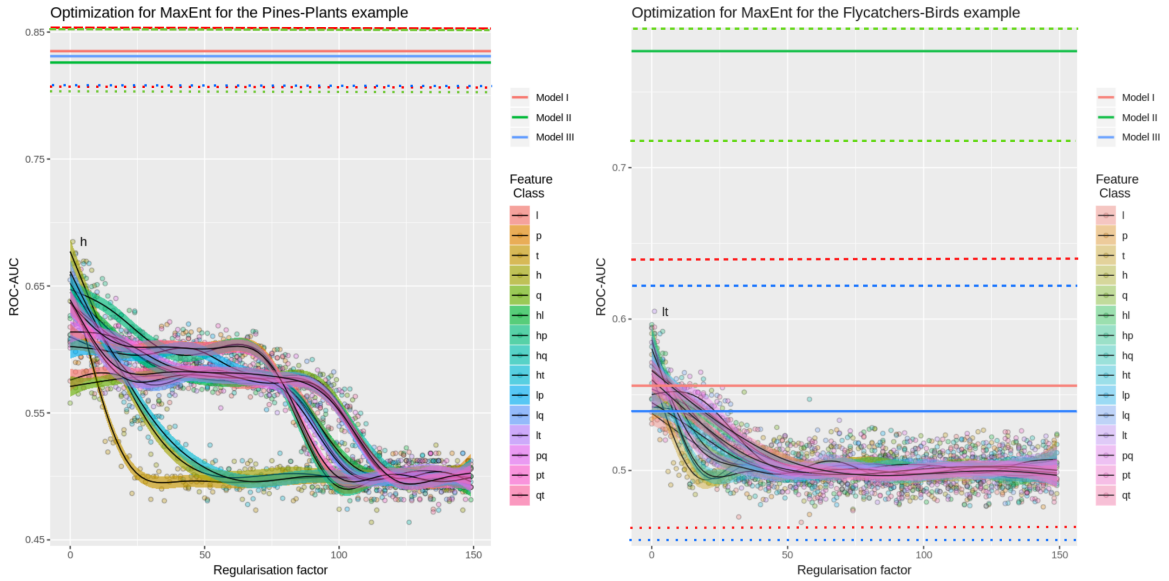


Fig. 3.3 Comparison of models I, II and III against the maximum entropy algorithm (bottom left panel). The maps displayed here corresponds to the posterior mean probability for the three models using observations of pines as presence (panels on left) and botanical records (panels on right) as the sampling process. The bottom right panel shows the observations used to fit the models.



(a) Pines example **(b)** Birds example

Fig. 3.4 Area under the receiver operating characteristic curve (AUC-ROC) for the different models of the pines example (left panel) and the birds example (right panel). The dots in colours represent a MaxEnt models using different parameters of regularisation (x-axis) and feature type (vertical legend). The values in the y-axis correspond to the resulting AUC-ROC value according to that specific pair of parameters. The AUC-ROC values of models I (red), II (green) and III (blue) are shown as horizontal lines. Solid lines represent the mean AUC-ROC values for models I, II and III, while dotted and dashed lines represent their respective lower and upper (95%) confidence intervals.

Figure 3.4b shows a comprehensive view of the aforementioned results. Additionally, a quantitative summary of these results is described in table 3.3.

In treatment *i* (i.e. missing data in both response vectors, the one for presence and the one for sample), Model III (correlated spatial components between the ecological process and the sampling effort) was the best ranked, achieving the lowest DIC value (3905), similarly to the Pines example. However, its accuracy in terms of ROC-AUC was close to random classification, reaching an AUC of 0.54 with $\pm(0.45, 0.62)$ at 95% CI. Model I (independent spatial effect for the ecological and the sampling components) obtained similar values of ROC-AUC ($0.56 \pm (0.47, 0.64)$ at 95% CI). In contrast, Model II obtained the highest predictive accuracy ($0.77 \pm (0.71, 0.84)$) with a DIC of 3905, second in rank. (see figure 3.4b); In addition, models I and III achieved a low predictive power compared to the benchmark model (MaxEnt).

Treatment *ii*, (i.e only one response vector (X) with missing information) showed contrasting results. Although model III (correlated components) ranked best, in terms of a lowest DIC (3331.1), its AUC was $0.95 \pm (0.94, 0.96)$. Model I (independent spatial components) followed with an AUC of $0.89 \pm (0.88, 0.91)$. Model II, could not obtain valid posterior distributions, as its log-likelihood diverged to $-\infty$. We discuss possible reasons and circumventing strategies in the next section.

All results are shown in table 3.3. Based solely on the DIC, Model III was ranked first in both treatments. However, in cases with large proportions of missing data (as in treatment *i* with 76.2% cells) the prediction accuracy (ROC-AUC) was low. This effect highlights the importance of selecting informative missing data as well as the type of model to use. These issues are explored further in the discussion section.

The covariate *Distance to roads* was found to be significant in models I and III. The rest (elevation, precipitation and population count) were not significant in all three models.

The selection of these specific covariates was based solely to demonstrate the capabilities of the model. As such, other covariates with stronger significance may be used.

Table 3.3 Comparison of the presence-only models: Independent Spatial Components (Model 1), Common Spatial Component (Model 2), Correlated Spatial Components (Model 3) and Maximum Entropy (MaxEnt) for the presence of the family *Tyrannidae* using birds as sample (class: *Aves*). A 7-fold cross validation was performed to calculate the area under the receiver-operating characteristic curve (ROC-AUC) as a measure of quality for each model. Models with the $\star$ symbol were fitted using only missing data from X (sample), i.e. treatment *ii*.

	DIC	ROC-AUC	95% C.I	DIC*	ROC-AUC*	95% C.I$\star$
Model I	4445.8	0.556	[0.47, 0.64]	5607.3	0.89	[0.88 ,91]
Model II	4251.1	**0.77**	[0.71, 0.84]	N.A.	N.A.	N.A.
Model III	**3905.0**	0.54	[0.45, 0.62]	**3331.1**	**0.95**	[0.94,0.96]
MaxEnt	–	–	–	–	0.61	[0.59,0.63]

Spatial results

Figure 3.5 shows the mean predicted latent surfaces for the presence of flycatchers P_Y (*Tyranids*) and relative sample P_X (*Birds*) in all the three models (left and right columns resp.). Model I presents a clear difference between P_Y and P_X (figure 3.5, first row). In this case, P_Y appears more smooth with patches of lower probability, although always with probability higher than 0.2. The surface P_X in model I (fig: 3.5, top right panel) has clear shaped patterns with contrasting probabilities between interior regions (*pocket shapes*). This feature is present in both surfaces of model II (fig:3.5, second row) and model III (fig:3.5, third row) The fixed effects (covariates) for P_X and P_Y are close to zero, therefore, the spatial variation is driven only by the common structure S. In the case of model III, the sample surface P_X presents greater connectivity and higher probabilities in places with known observations. Both surfaces, however, present a similar structure in shapes and patterns.

In contrast, the MaxEnt prediction lacks the random spatial effect component. The resulting probability surface is determined exclusively by the features used by the covariates. Although is possible to distinguish spatial patterns within the region, the predicted

probability is in general close to uniform random classification (i.e. 0.5). This effect is supported by the obtained AUC-ROC value of the cross-validation analysis (0.6) (fig: 3.4b (a)). In Model III, the median correlation, obtained from the cross variance (Σ)between the presence of flycatchers (P_Y) and the sampling effort (P_X), was 0.996 with $(0.993, 0.998)$ 95% credible interval. As in the latter example, this result is consistent with the fact that the taxon of interest (i.e. flycatchers) is totally contained in the sampling effort (i.e.birds). The complete estimates' summary can be checked in 3.11.

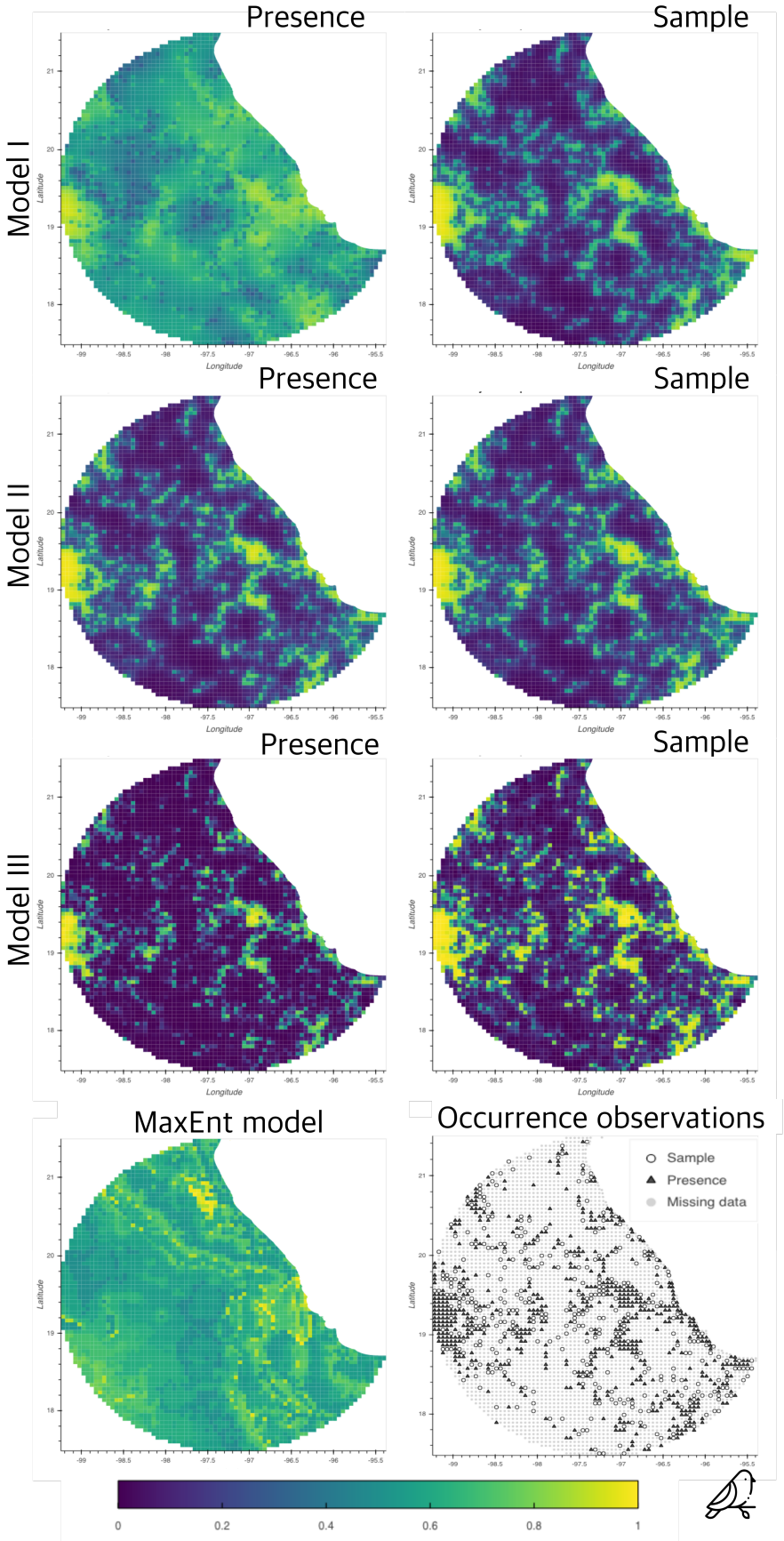


Fig. 3.5 Comparison of models I, II and III against the maximum entropy algorithm (bottom left panel). The maps displayed here corresponds to the posterior mean probability for the three models using observations of flycatchers as presence (panels on left) and observations of birds records (panels on right) as the sampling process. The bottom right panel shows the observations used to fit the models.

3.5 Discussion

We proposed a framework for predicting the probability of occurrence of a given taxon using presence-only data. Our contribution is the design of a bivariate CAR framework that uses an additional source of information, apart from the presences of the target species. This extra information comes from sampling observations related to other species and other taxa that, according to the modeller, give complementary information relative to the occurrence of the taxon of interest (ToI). The framework relies on three fundamental concepts: *i)* the sampling effort as complementary information for inferring the probability of presence, *ii)* the spatial autocorrelation structure for determining the variability and occurrences likelihood across the landscape, and *iii)* the *choosing principle*, a mechanism for determining presences, relative absences and missing data from presence-only records. Both examples showed that, at least one of the three proposed models outperformed MaxEnt. The results in tables 3.2 and 3.3 show that the models' goodness-of-fit statistic (i.e. DIC) and predictive accuracy increased in treatment *ii*, that is, when the absence of records were treated as real absences. This is expected because assuming missing data as real absences reduces uncertainty.

These results show that the proportion of missing data plays a fundamental role in the predictive capability of the model. This effect is recognised in the flycatchers example, where the proportion of missing observations is much higher (76% of the total number of regions) compared to presences and relative absences. In this case, models I and III produced low predictive accuracy, similarly to MaxEnt, with an AUC-ROC of near 0.6 (i.e., close to random classification). In contrast, model II, although ranked second in terms of DIC, achieved the highest predictive accuracy (AUC-ROC). This result is also supported by by the high number of missing data (increased uncertainty) and reduced number of spatial parameters to fit. In terms of models' parsimony, one shared spatial latent effect

(model II) has less parameters to fit compared with two spatial effects in the case of models I and II.

The three proposed models impose different restrictions on how the spatial autocorrelation structure affects the probability of a species to occur. The more complex the spatial structure is, the more presence-only observations (and less missing data) are needed. This can be modulated by the amount of missing data with respect to the relative absences determined by the sampling effort observations and the choosing principle. Consequently, using an appropriate informative sample becomes crucial for obtaining accurate inferences and predictions. This finding highlights interesting paths for future research: one related to the selection of informative observations for the sampling effort process, and the other for different choosing principles.

Model II may be a better alternative for taxa with sparse spatial distributions and large proportion of missing data. Nevertheless, model II presented problems with identifiability in treatment *ii* (i.e. missing data only in the ToI observations and assumed real absences in the sampling process). A possible reason is that the inference method could not find a suitable compromise in accounting for a common spatial effect that had two constraints. One, the accountability of residuals of both processes (P_Y and P_X) and two, the restrictions imposed by the intrinsic CAR model specification. That is, the sum of the random effect on all the lattice areas should sum one. A possibility to circumvent this last restriction is to specify, instead, a proper CAR model (e.g (Leroux et al., 2000)). The package CARBayes (Lee, 2013) allows this specification. We recommend the practitioner to compare the three models accordingly to fit specific needs.

3.5.1 The role of the choosing principle

When presence-only data are used, any choosing principle is inevitably a source of potential bias. Thus, the research question and the selection of the sampling effort observations

play a fundamental role in determining the accuracy of predictions. The way relative absences and missing data are derived implies ecological assumptions that should be kept in mind when one tries to model species (taxon) distributions. For example, following the *biotic, abiotic, movements* (BAM) diagram proposed (Soberon and Nakamura, 2009), if the objective is to model the *realised distribution*, (i.e., places where the species lives in reality) absences become informative. If on the other hand, the objective is to model the species' *potential distribution* (i.e. places where it can survive and thrive due to suitable environmental conditions) absences may constitute missing data. See equivalent concepts from a SDM approach Jiménez-Valverde et al. (2008).

In our framework, we used the sample observations X together with the *choosing principle* to discriminate between informative absences and missing data. If the sampling effort is chosen to be informative it can increase significantly the accuracy of predictions (see table 3.2).

The current choosing principle assumes that for every location k, if the ToI (e.g. species) is not present, but the sample observation exists ($X_k = 1$), then the ToI is assumed to be absent (Y_k =0). In some applications this assertion may be incorrect and, if the sample observations X consist as well of presence-only data, the bias in false absences can propagate in both processes. This problem is present in all presence-only methods that tries to account for the sampling bias using pseudo-absences (e.g. target-background approach of Phillips et al. (2009)), given the intrinsic bias of the collected data. Ideally, the best way to rank distinct choosing principles, given a ToI, is using presence-absence data. The proposed choosing principle is not intended to be a general rule for all species and problems. An it is worth for the modeller to consider other choosing principle in which relative absences and missing data can be specified from presence-only data. For example, another type of choosing principle can incorporate information on other species features. For example movement, since the accessibility of an area can be indicative of

poor sampling and its use has been shown to reduce bias in occurrence data (Monsarrat et al., 2018).

We would like also to explore further the role of the taxonomic structure in determining informative samples. In the examples we used broad and generic groups, jumping from class *Pinopsida* to kingdom *Plantae*, in the case of Pines, and from family *Tyranidae* to class *Aves*, in the case of the flycatchers. We hypothesise that using the immediate parent node of the ToI, according to its taxonomical classification, could give more accurate models for certain groups. An example of this could be the use of the family (of the ToI) as sample, if the ToI is a type of genus.

In recent years, spatial point process (SPP) models have been proposed to model presence-only occurrences (see Velázquez et al. (2016) for review).

This is a sensible choice of modelling giving that these models are able to represent discrete events in a continuous space. Recently, authors like (Renner et al., 2015, 2019) proposed a combined likelihood approach for modelling the spatial dependence using a latent log Gaussian Cox process (Møller et al., 1998). Although these models are sound and have been used satisfactory, the assumptions about the required sample design restrict their application to only specific cases (Gelfand et al. (2013), Chp. 20). Additionally, in SPP models, all information is contained in the location of the occurrences and separating the sampling effort from the ecological process, can lead to confounding and identifiability problems. In our opinion the use of spatial lattices (i.e. Gaussian Markov random fields) for modelling spatial autocorrelation presents a more appropriate alternative for modelling generic species.

3.5.2 Advantages in using this framework

The model is defined in a spatial lattice. The observations occurred on a given area element can be aggregated to reflect presences or abundances. That is, the model support

repeated measurements within areas. In addition, the probabilities for presence in areas that have not been sampled can be inferred by the neighbouring areas. The method is able to infer places where data availability is limited. The model specifies a Bayesian hierarchical model and accounting uncertainties of the parameters is possible. This brings the possibility to perform hypotheses testing on the posterior sample. As it is a hierarchical model it is possible to perform model selection using the DIC statistic. The structural components of the models, that is, the ecological process and the sampling effort can be explicitly modelled using different covariates and even feature classes, as the ones used by MaxEnt. Lastly, the choosing principle provides a flexible form to assign absences and missing data.

3.5.3 Limitations

Manipulating the spatial random component of the model implies greater computational complexity on the order of $O(n^3)$ (in its worse scenario). Although, the matrix is sparse and the inference uses optimised numerical methods that can reduce the computational complexity, the numerical methods involved are more intensive than MaxEnt or other models that are not based on hierarchical Bayesian inference. This is a limitation for studies that requires extended regions involving hundreds of thousands of area elements.

Another limitation is that the specification of the spatial effect is based on discrete spatial distributions. This implies that, once the model is fitted, it is not possible to make predictions on observed regions or data (as opposed to geostatistical models). Also, depending on the specification, a modeler may need the spatial random effect to be continuous in space, instead of over a discrete lattice. If this is the case we recommend the use of SPP-based models like (Renner et al., 2015, 2019).

3.6 Conclusions

The proposed framework was demonstrated to be superior than the benchmark model (i.e. MaxEnt). All models within the framework achieved better predictive accuracy when compared to MaxEnt. Additionally, as this is a statistical model it accounts for true parameter uncertainties allowing robust statistical analysis; a missing feature in the algorithmic MaxEnt approach. Additionally, MaxEnt does not account for missing data and spatial autocorrelation structure. This highlights the importance of modelling these sources of variation to increase the accuracy of the spatial prediction of species.

In the studied cases, the likelihood of the sampling effort informed the presence process P_Y in all models. This was true even for the independent model I, where the information from the sampling effort is informed only by the *choosing principle* and relative absences. In both examples, at least one of the three models achieved high predictive accuracy (AUC ≥ 0.7); bringing attention to the use of informative observations in the likelihood of the sampling effort and its role in determining accurate predictions.

Overall, model III (i.e. correlated spatial effects) obtained the highest *goodness of fit* measure, that is, it was the one that lost the least information. Model II, however, was more suitable in terms of accuracy when the missing data were significantly larger than the relative sample. In cases like this, where the proportion of missing information is large, it is appropriate to consider a single source of spatial autocorrelation for both processes.

3.7 Data and source code availability

Currently the code and data are stored in the following repository: https://github.com/molgor/CARBayeSDM. We intend to put the code and data in a long term curated repository such as Dryad or FigShare.

3.8 Declarations

3.8.1 Funding

This project was jointly sponsored by the Mexican Science and Technology Council (CONACyT) under the doctoral program: *Becas al Extranjero* and the Faculty of Science and Technology from Lancaster University. Icons for birds and pines made by Freepik from www.flaticon.com.

3.8.2 Authors' contributions

All authors developed the general framework and provided critical feedback in all the stages of this work. More specifically, PD proposed the three model specifications. PA proposed the choosing principle. LS and JEM designed the modelling and simulations strategies. JEM prepared the data, implemented the models, performed the analysis and visualizations and wrote the manuscript with inputs and edits from all co-authors. PA, LS and PD supervised the project.

3.8.3 Conflicts of interest

The authors declare no conflict of interest.

3.9 Supplementary materials I: Framework specification

We begin by defining a grid inside a region of interest located somewhere on the Earth's surface. Mathematically this is a spatial lattice $\mathbb{W} = \{k_1, ..., k_K\}$ that partitions a compact set $A \subset \mathbb{S}^2 \subset \mathbb{R}^3$ into K non-overlapping compact subregions. Let $X = \{x_k | k \in \mathbb{W}\}$ be the recorded presence of a certain sample (or survey) and $Y = \{y_k | k \in \mathbb{W}\}$ the presence of a taxon (e.g. species) of interest (ToI). As such, x_k and y_k are two binary random variables corresponding to the events of: *a sample x_k has been registered in location k* and *taxon y_k is present at location k*. Missing observations are defined in the same lattice as: $\widetilde{X} = \{\tilde{x}_k | k \in \mathbb{W} \wedge \mathscr{R}_x(k)\}$ where $\mathscr{R}_x(k)$ is the predicate of: *there is no recorded evidence of x in k* and similarly, $\widetilde{Y} = \{\tilde{y}_k | k \in \mathbb{W} \wedge \mathscr{R}_y(k)\}$ where $\mathscr{R}_y(k)$ is the predicate of: *there is no recorded evidence of the presence of y in k*. The data augmentation methodology (Tanner and Wong, 1987) implemented in CARBayes (Lee, 2013) generates posterior samples of $\widetilde{X}$ and $\widetilde{Y}$. We opted to omit any further specification for the variables $\widetilde{X}$ and $\widetilde{Y}$ here, to simplify the description of the framework.

The general specification of the framework factorises the joint probability distribution in the following form:

$$
\begin{aligned}
[Y, X, P_Y, P_X, R_Y, R_X, \beta_Y, \beta_Y; d_Y, d_X, \mathbb{W}] &= [Y|P_Y][X|P_X] && (3.4)\\
&\quad [P_Y|R_Y, \beta_Y][P_X|R_X, \beta_X] && (3.5)\\
&\quad [\beta_Y; d_Y][\beta_X; d_X] && (3.6)\\
&\quad [R_Y, R_X; \mathbb{W}] && (3.7)
\end{aligned}
$$

Equations 1 to 3 are consistent across the framework while the specification for equation 4 (i.e. *random effects*) vary according to three different assumptions of spatial autocorrelation; independent components (model I), a common spatial component (model II) and

correlated spatial components (model III). We start by defining equations 1 and 2. That is, the probability of presence for a ToI (Y_k) given the latent variable $P_Y(k)$ in a cell k and similarly, the probability of a sample X_k to be present given its respective latent variable $P_X(k)$. These binary random variables are modelled as following:

$$[Y|P_Y = p_y] \sim \text{Bernoulli}(p_y) \tag{3.8}$$

$$[X|P_X = p_x] \sim \text{Bernoulli}(p_x) \tag{3.9}$$

3.9.1 Latent variables P_Y and P_X

We assume that the presence-only data represent realizations of a joint stochastic process separable in two components: one relative to an ecological process P_Y that drives the environmental suitability for the ToI, and another process P_X related to the sampling effort. We, therefore, model $[P_Y = p_y|R_Y = r_y, \beta_Y; d_Y]$ and $[P_X = p_x|R_X = r_x, \beta_X; d_X]$ (eqs. 3.5) according to the following specification:

$$\log\left(\frac{p_y}{1-p_y}\right) = d_Y^t \beta_Y + r_y \tag{3.10}$$

$$\log\left(\frac{p_x}{1-p_x}\right) = d_X^t \beta_X + r_x \tag{3.11}$$

where d_X and d_Y represent vectors of explanatory variables and r_X and r_Y the random effects for X and Y respectively. Specifically, d_Y is suited for environmental variables of ecological importance, while d_X should account for variables that help explain the sampling process. The prior distributions for β_Y and β_X (eq: 3.6) are defined, as default, as uninformative zero-mean normal distributions with default variance 100,000. We acknowledge that the use of uninformative priors can yield to skewed parameter estimates and negate the advantage of using Bayesian methods over frequentist analyses (Gelman and Shalizi, 2013; Hobbs and Hooten, 2015). These hyperparameter values are default

options in CarBayes (Lee, 2013) and, consequently, in our modelling framework. As such, they can be changed according to the user needs. See (Lemoine, 2019) for a concise guide on using informative and weakly informative priors in ecological models. In the following section we present the three alternatives for modelling R_X and R_Y.

3.9.2 Random effects

The general form of the random effects component for P_Y (and P_X) is defined as an independent zero-mean random variable R_Y (R_X). This variable accounts for the combined effect of a spatial process S_Y (S_X) that models the spatial variation across the lattice $\mathbb{W}$ and an independent normally distributed random variable Z_Y (Z_X) with variance σ_Y^2 (σ_X^2) that accounts for unstructured noise inside each cell of the lattice.

Specifically, these random effects are defined as follows:

$$\begin{aligned} R_Y &= S_Y + Z_Y \\ R_X &= S_X + Z_X \end{aligned} \tag{3.12}$$

where $Z_Y \sim N(0, \sigma_Y)$ and $Z_X \sim N(0, \sigma_X)$ and the spatial components S_Y and S_X are modelled as *intrinsic conditional autoregressions* (ICAR) (Besag, 1974; Besag et al., 1991) with parameters τ_Y^2 and τ_X^2 respectively, over the lattice $\mathbb{W}$. In the rest of this work we represent $\mathbb{W}$ in its matrix form, that is, the adjacency matrix W of its graph representation; defined as a $k \times k$ symmetric matrix with entries: $w_{i,j} = 1 = w_{j,i}$ if cells i and j are neighbours, otherwise $w_{i,j} = 0$. Modelling the spatial autocorrelation as an ICAR eases significantly the computation of W^{-1} with the aid of optimised methods for sparse matrix algebra (Rue and Held, 2005). This approach simplifies significantly the inference, prediction and posterior sampling, a great advantage in applications with large datasets.

3.9.3 Three models for spatial autocorrelation

The proposed framework assumes that the ecological process P_Y and the anthropogenic sampling process P_X are independent when conditioned to the random effects R_Y and R_X (see figure 3.1 and eq: 3.5). This assumption implies that the only source of dependency between R_Y and R_X is the dependency between the spatial effects S_Y and S_X, this by the assumption of independence between variables Z_Y and Z_X. Moreover, the framework assumes that the observations of presence for the ToI and the existence of the survey (sampling) are independent when conditioned to the spatial effect. As such, the spatial autocorrelation structure is the component responsible for informing both processes. In order to test for this we designed three possible models in which the spatial processes S_Y and S_X inform R_Y and R_X. Model I in which the spatial components S_Y and S_X are independent, Model II with a unique spatial component shared between both processes P_X and P_Y (i.e. $S_X = S_Y$) and Model III in which the spatial components S_X and S_Y are correlated. Below we give the full description of each model.

Model I: Independent Spatial Components (ISC)

This model assumes that the spatial random effects on both processes (R_X, R_Y) are independent. By equations 3.12 the joint distribution is given by

$$[R_Y, R_X; \mathbb{W}] = [S_Y, S_X, Z_X, Z_Y, \tau_Y^2, \tau_X^2, \sigma_Y^2, \sigma_X^2; W]$$

and, given the assumptions on independence, it can be factorised into:

$$\begin{aligned} [S_Y, S_X, Z_X, Z_Y, \tau_Y^2, \tau_X^2, \sigma_Y^2, \sigma_X^2; W] &= [S_Y|\tau_Y^2; W][S_X|\tau_X^2; W] && (3.13)\\ &\quad [Z_X|\sigma_X][Z_X, \sigma_X^2] && (3.14)\\ &\quad [\tau_Y^2][\tau_X^2][\sigma_Y^2][\sigma_X^2] && (3.15) \end{aligned}$$

where the term $[S_l|\tau_l^2; W]$ (l being X or Y) is modelled as an ICAR (Besag, 1974; Besag et al., 1991) with a full conditional form of:

$$[S_{l_k}|S_{l_{-k}}, \tau_l^2; W] \sim N\left(\frac{\sum_{i=1}^{K} w_{k,i} S_{l_i}}{\sum_{i=1}^{K} w_{k,i}}, \frac{\tau_l^2}{\sum_{i=1}^{K} w_{k,i}}\right) \tag{3.16}$$

for each process $l \in \{Y, X\}$ on each cell k (i.e. S_{l_k}). The prior distributions for parameters τ_l^2 and σ_l^2 are defined as inverse gamma(1,0.01), default values in the package *CARBayes*. Figure 3.1a (in the main text) shows a general DAG structure for this model.

Model II: Common Spatial Component (CSC)

This model assumes that the random effects R_X and R_Y share the same spatial component S (i.e. $S_X = S_Y$). By equations 3.12 the joint distribution is given by $[R_Y, R_X; W] = [S, Z_Y, Z_X, \tau^2, \sigma_Y^2, \sigma_X^2; W]$ and, given the assumptions on independence, it can be factorised as:

$$\begin{aligned} [S, Z_Y, Z_X, \tau^2, \sigma_Y^2, \sigma_X^2; W] &= [S|\tau^2; W] && (3.17)\\ & [Z_Y|\sigma_Y^2][Z_X|\sigma_X^2] && (3.18)\\ & [\sigma_Y^2][\sigma_X^2] && (3.19) \end{aligned}$$

Similarly to model I, the spatial effect $[S|\tau^2; W]$ is modelled as an ICAR (Besag, 1974; Besag et al., 1991) in full conditional form on each cell $k \in \mathbb{W}$.

$$[S_k|S_{-k}, \tau^2; W] \sim N\left(\frac{\sum_{i=1}^{K} w_{k,i} S_i}{\sum_{i=1}^{K} w_{k,i}}, \frac{\tau^2}{\sum_{i=1}^{K} w_{k,i}}\right) \tag{3.20}$$

The prior distributions for parameters τ_l^2 and σ_l^2 are defined as inverse gamma(1,0.01), default values in the package *CARBayes*. Figure 3.1b (in the main text) shows a general DAG structure for this model. Model II is specified as a two-level model where each areal

unit k has two response variables, X_k and Y_k. The individual level variation is split into two groups: Z_X and Z_Y. Figure 3.1b shows the DAG describing the model.

Model III: Correlated Spatial Components (CSC)

This model specifies the joint random effect $[R_Y, R_X; W]$ as a combined effect of the spatial processes, S_Y and S_X. To model this effect, both spatial effects are ensembled as a bivariate conditional autoregresive (BCAR) process that accounts for both S_Y and S_Y simultaneously. To improve the identifiability of the model, the unstructured random effect (i.e. Z_X and Z_Y in models I and II) is integrated into the spatial effect using a more relaxed specification of the spatial autocorrelation structure. This specification, proposed by Leroux et al. (2000), adds a new parameter ρ that models the strength of the spatial dependency. When $\rho = 1$ the spatial dependency is maximum and the spatial process is equivalent to an intrinsic CAR model. On the other hand, if $\rho = 0$ there is no evidence of spatial autocorrelation and therefore, the observations are spatially independent. To make the comparison between models I and II consistent, we have restricted $\rho = 1$. However, this restriction can be removed according to the needs of the users. Following the equations 3.12 and the DAG specification shown in figure 3.1c (in the main text) the joint distribution $[R_Y, R_X; W]$ can be factorised as:

$$[R_Y, R_X; W] \quad = \quad [S_{YX}|\Sigma, \rho; W][\Sigma][\rho] \tag{3.21}$$

The combined random effect S_{YX} is defined as the Kronecker product between the Leroux et al. (2000) CAR model and a 2×2 covariance matrix Σ that accounts for the cross variable effect between both processes. The correlation between both variables can be calculated as:

$$Corr(X, Y) \quad = \quad \frac{\Sigma_{1,2}}{\Sigma_{1,1}\Sigma 2, 2} \tag{3.22}$$

The BCAR model is a particular case of the multivariate model (MCAR) proposed by Gelfand and Vounatsou (2003) and it has been implemented in the R package CARBayes (Lee, 2013) following the proposal of Kavanagh et al. (2016). S_{YX} is a realization of the following multivariate normal distribution:

$$S_{YX} \sim N\left(0, \left[Q(W,\rho) \otimes \Sigma^{-1}\right]^{-1}\right) \tag{3.23}$$

The autocorrelation function $Q(W,\rho)$ is defined by the precision matrix:

$$Q(W,\rho) = \rho[D - W] + (1-\rho)I \tag{3.24}$$

where D is a $k \times k$ diagonal matrix in which each entry $d_{i,i}$ is equal to the number of neighbours of each unit area $i \in \{1,..,k\}$. The prior for Σ is distributed as Inverse-Wishart$(3,\Omega)$ with three degrees of freedom and $\Omega = I_{2x2}$ as scale matrix. The prior $[\rho]$ is a non-informative uniform (0,1) distribution. The DAG describing the model is described in figure 3.1c.

3.10 Supplementary materials II

This section contains the summary statistics of the fitted posterior distributions of the parameters corresponding to models I, II and III, described in summary in the main text (section: 3.2) and extensively in the supplementary materials 3.9. The summary statistics corresponding to the presence of pines (using plants as sampling effort) is showed first. The second case study is showed in the next section. The structure of every table is the same for all models in both examples. The rows describe the parameters corresponding to each model (on each table). The first three columns describe the median, upper and lower bounds of the 95% credible intervals. The `n.effective` column indicates an estimate for the size of independent samples (taking into account autocorrelations within each chain of the MCMC sampler). The column `% accepted` refers to the proportion of times a proposed value was accepted by the Metropolis updating step as a new value of the posterior sample (see (Lee, 2013)). The column `Geweke.diag` refers to Geweke's convergence diagnostic (Geweke, 1992) which compares the means calculated from distinct parts of the Markov chain to test for convergence of the stationary distribution (default first 10% and last 50%). If the chains reached a stationary distribution, then the two means are equal and Geweke's statistic has an asymptotically standard normal distribution. All models can be fitted in CARBayes (Lee, 2013), which uses the R package Coda (Plummer et al., 2006) for calculating `n.effective` and `Geweke.diag`.

3.10.1 Estimates for the predicted presence of Pines using botanical records as sample

Table 3.1 Posterior summaries of all the parameters in Model I with the associated 95% credible intervals for the example of pines. Parameters τ^2_Y and τ^2_X correspond to the variance of the spatial effects of the presence (Y) and the sample process (X) (i.e. S_Y and S_X) respectively. Likewise, σ^2_Y and σ^2_X correspond to the variance of the unstructured processes Z_Y and Z_X respectively. Significant parameters are shown in **bold**. For further information see section: 3.3

	Median	2.5%	97.5%	n.sample	%accept	n.effective	Geweke.diag
(Intercept of Y)	-1.1871	-4.0872	0.9928	10000	64.2	16.0	-7.8
Elevation	0.0002	-0.0002	0.0006	10000	64.2	299.9	-2.0
Precipitation	0.0002	-0.0001	0.0005	10000	64.2	206.4	0.4
τ^2_Y	19.6638	13.2754	45.1344	10000	-	8.5	-1.3
σ^2_Y	0.3658	0.0357	0.7923	10000	-	3.1	1.8
(**Intercept of** X)	3.0309	2.4178	3.9749	10000	61	24.3	-0.9
Dist. to road	-0.0002	-0.0004	-0.0001	10000	61	1294.1	0.5
Population	0.0000	-0.0001	0.0001	10000	61	1320.2	0.4
τ^2_X	5.2708	2.7058	9.5806	10000	-	8.7	-1.1
σ^2_X	0.1818	0.0637	0.3250	10000	-	7.9	-1.1

Table 3.2 Posterior summaries of all the parameters in Model II with the associated 95% credible intervals for the example of pines. The parameter τ^2 represents the variance of the common spatial effect. Parameters σ^2 and σ^2 correspond to the variance of the unstructured process Z_Y and Z_X. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3

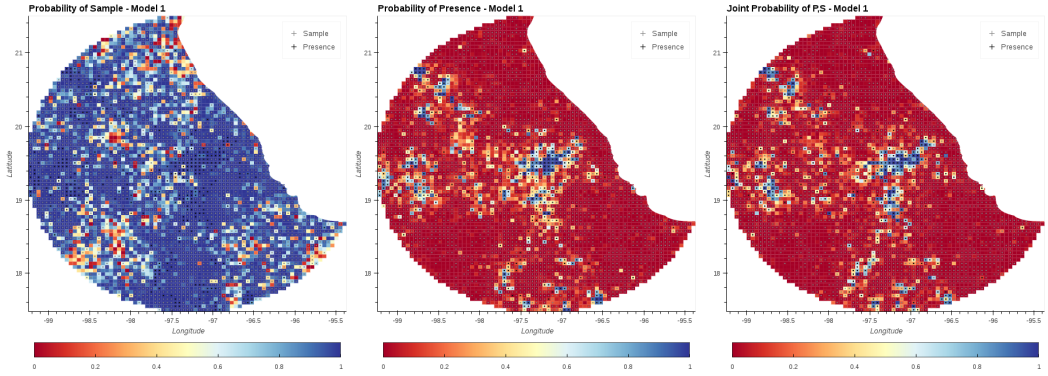
	Median	2.5%	97.5%	n.sample	%accept	n.effective	Geweke.diag
(**Intercept**)	-0.7085	-1.0766	-0.3426	5000	51.6	80.5	-4.9
Dist. to road	-0.0002	-0.0004	-0.0001	5000	51.6	170.9	-1.2
Population	0.0000	-0.0001	0.0001	5000	51.6	150.2	-0.2
Elevation	0.0002	0.0000	0.0004	5000	51.6	79.7	1.6
Precipitation	0.0003	0.0001	0.0004	5000	51.6	85.9	3.5
τ^2	6.8838	4.7169	11.8695	5000	-	5.5	5.1
σ^2	9.7797	2.8682	72.7988	5000	-	5000.0	1.1

Table 3.3 Posterior summaries of all the parameters in Model III with the associated 95% credible intervals for the example of pines. Parameters σ^2_Y and σ^2_X correspond to the variance for the presence (Y) and the sample (X). The term $\text{corr}_{X,Y}$ indicates the correlation between these two processes. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3

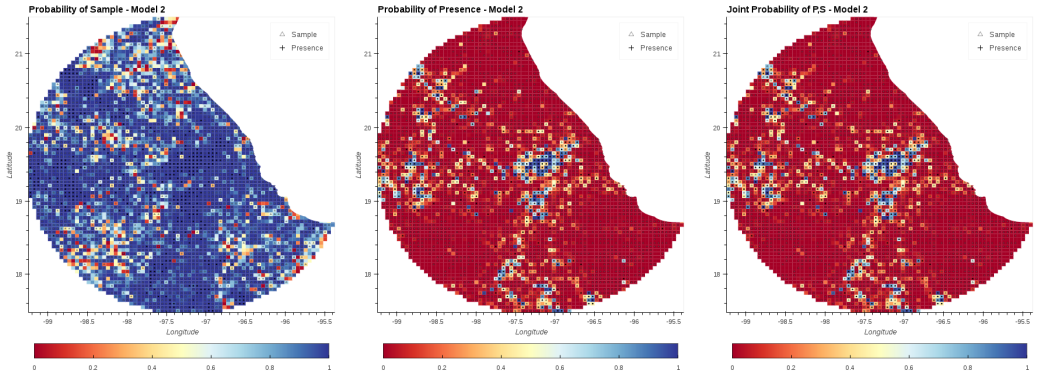
	Median	2.5%	97.5%	n.sample	%accept	n.effective	Geweke.diag
(**Intercept of** Y)	-7.7938	-9.2851	-6.3099	5000	55.6	60.5	6.4
Elevation Y	0.0003	-0.0001	0.0007	5000	55.6	102.6	-3.0
Precipitation Y	0.0002	-0.0002	0.0005	5000	55.6	82.7	0.7
(**Intercept of** X)	3.4115	2.7572	4.4384	5000	55.6	58.4	5.7
Dist. to road X	-0.0002	-0.0004	-0.0001	5000	55.6	387.9	-3.3
Population X	0.0000	-0.0001	0.0002	5000	55.6	437.5	-0.3
σ^2_Y	31.8726	21.3638	44.6661	5000	-	8.2	-3.5
σ^2_X	6.8778	4.3181	15.4775	5000	-	5.1	2.2
$\text{corr}_{Y,X}$	0.972	0.906	0.994	-	-	-	-

3.10.2 Maps of posterior variables for the presence of Pines

(a) Model I



(b) Model II



(c) Model III

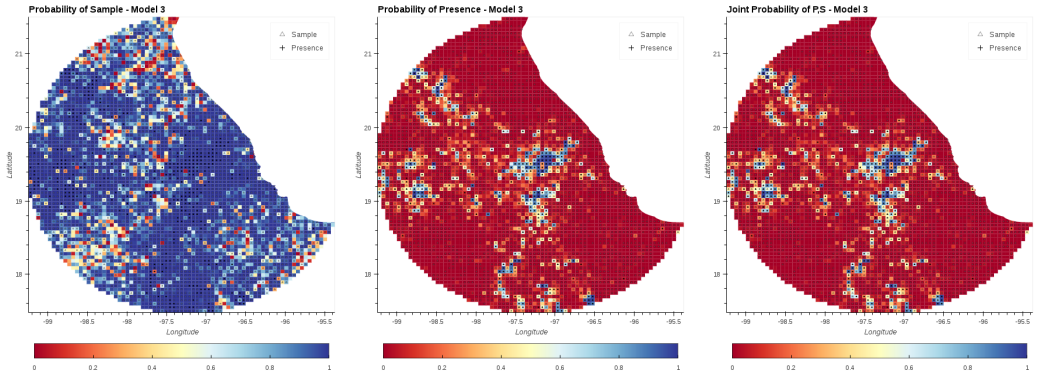
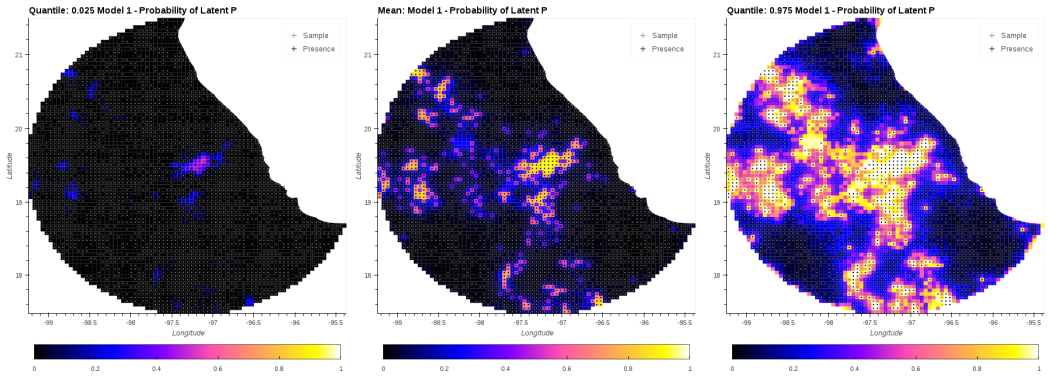
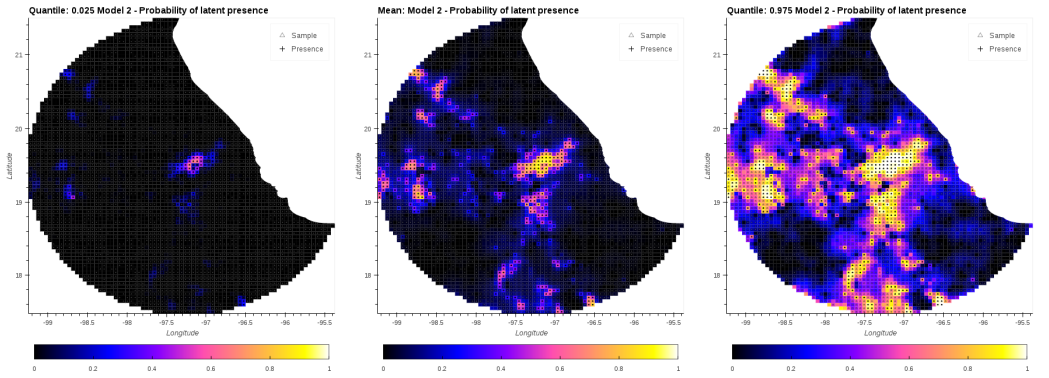


Fig. 3.6 Mean probability and 95% C.I. for Presence, Sample, and Joint presence and sample for Models I, II and III predicting presence of Pines (Class: Pinopsida) using Plants (Kingdom: Plantae) as sample.

(a) Model I



(b) Model II



(c) Model III

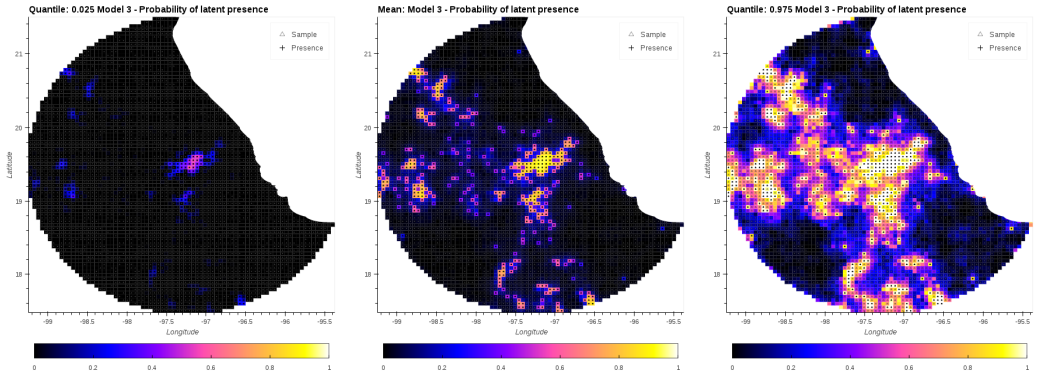
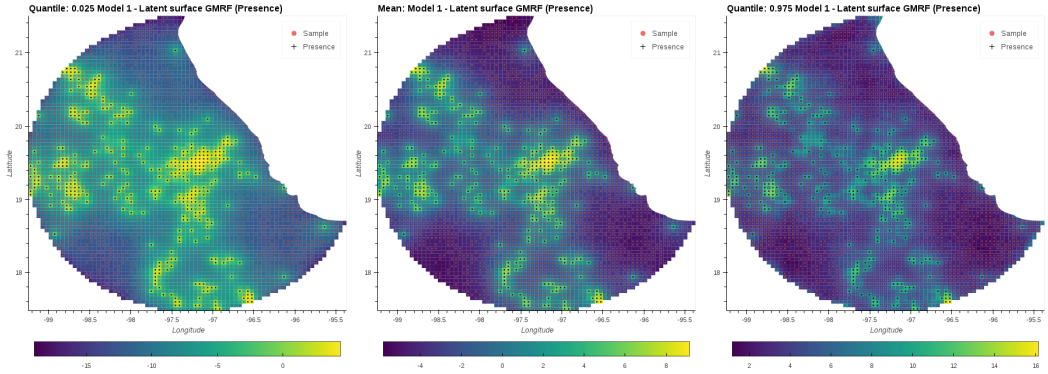
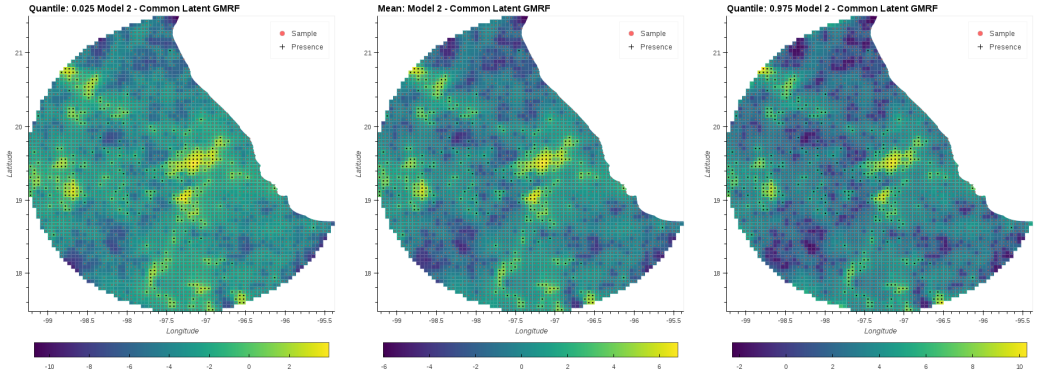


Fig. 3.7 Latent variable P_Y (Presence) for Models I, II and III predicting presence of Pines. The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively.

(a) Model I



(b) Model II



(c) Model III

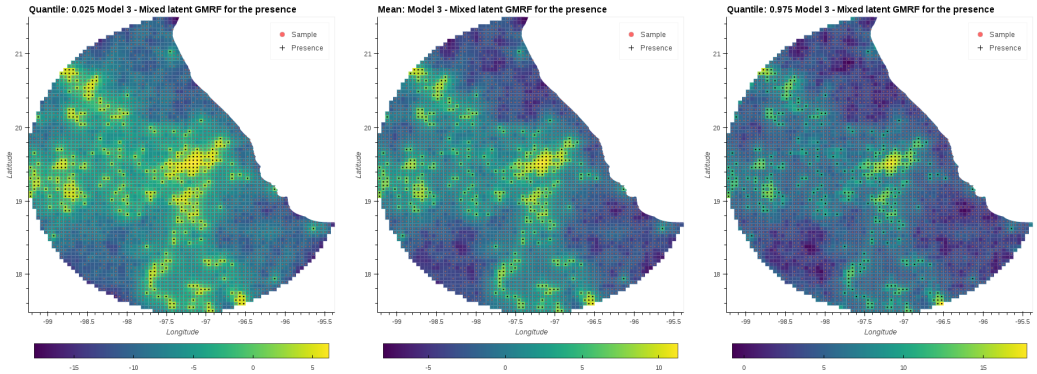
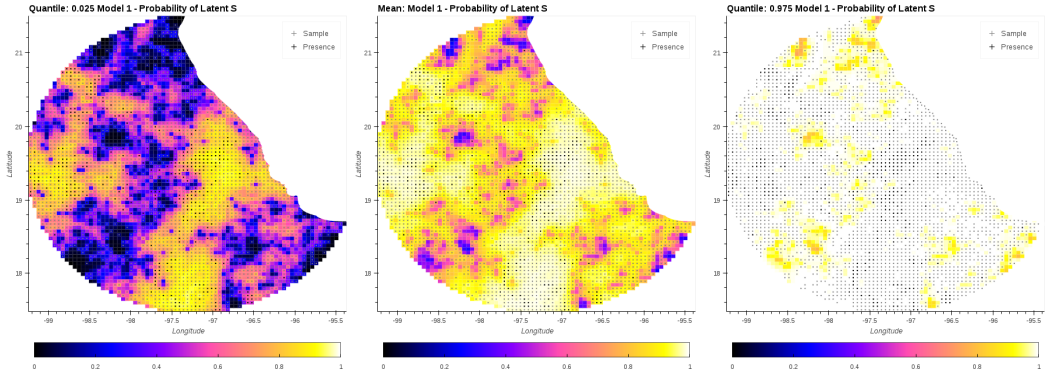
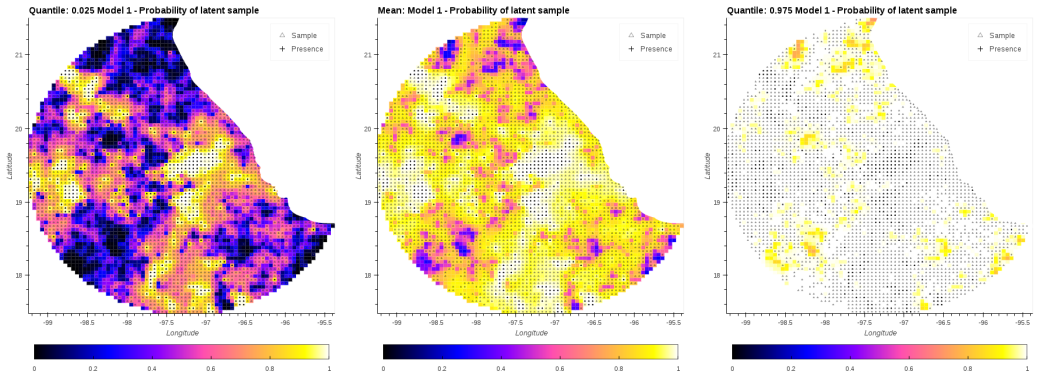


Fig. 3.8 Spatial random effect S_Y. The Gaussian Markov random field (GMRF) corresponding to the latent variable P_Y (Presence) for Models I, II and III predicting presence of Pines. The central column corresponds to the mean value, The column on the left and right corresponds to quantiles: 0.025 and 0.975, respectively.

(a) Model I



(b) Model II



(c) Model III

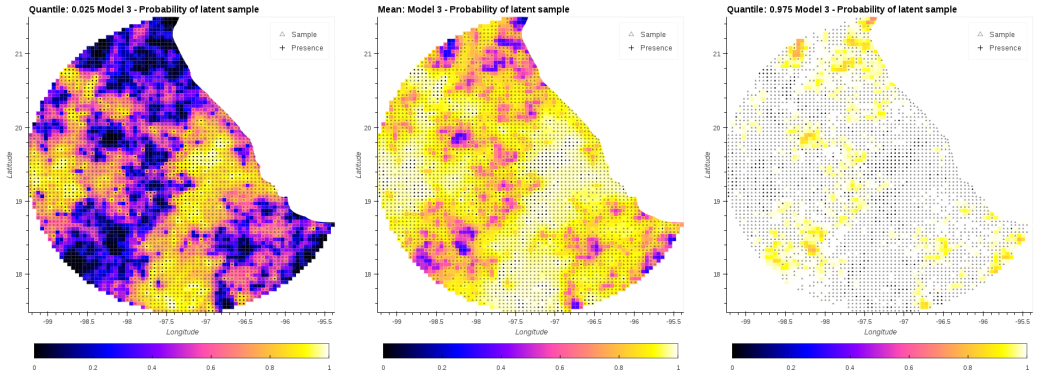
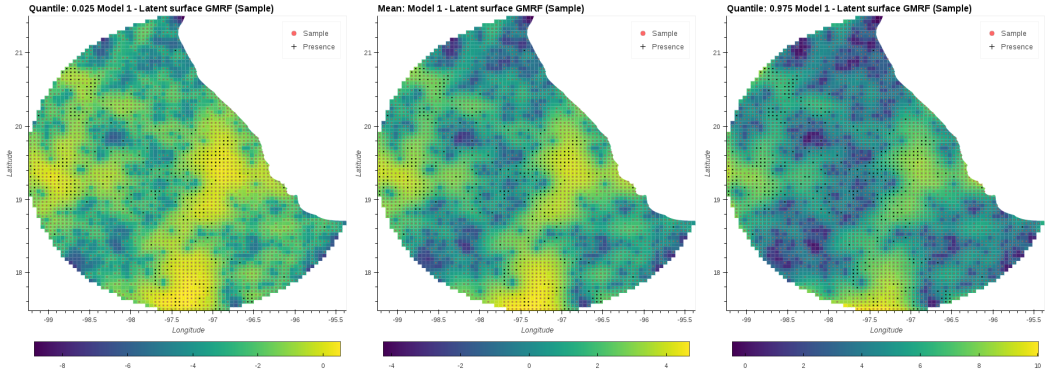
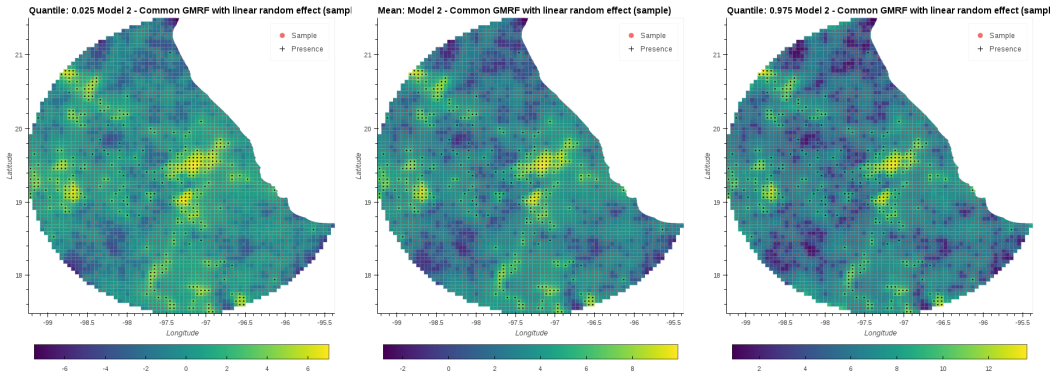


Fig. 3.9 Latent variable P_X (Sample) for Models I, II and III predicting presence of Pines using all plants as sample. The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively.

(a) Model I



(b) Model II



(c) Model III

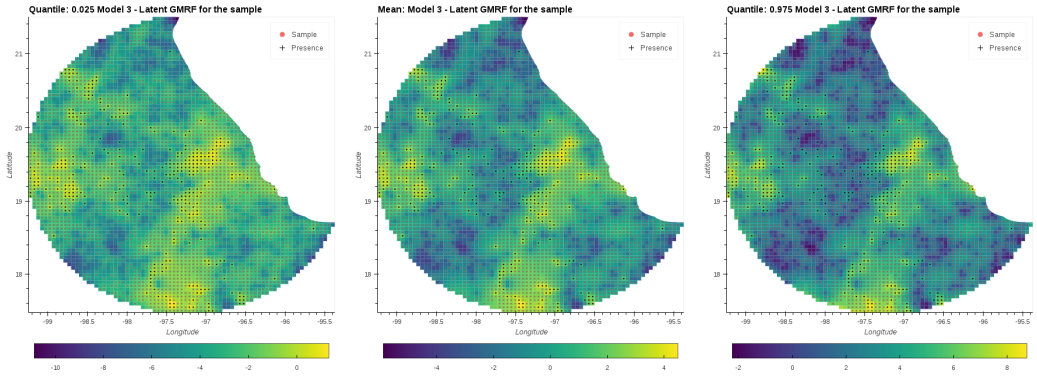
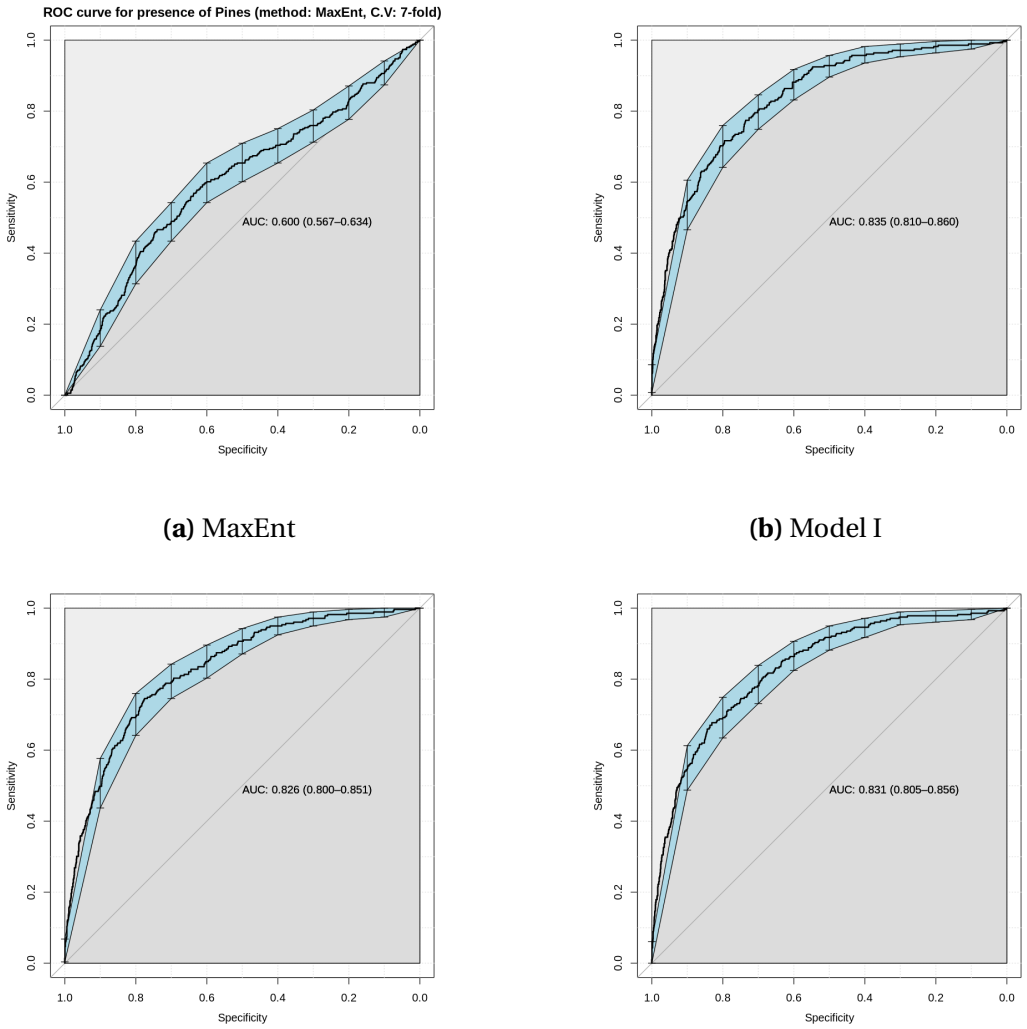


Fig. 3.10 Spatial random effect S_X. The Gaussian Markov random field (GMRF) corresponding to the latent variable S_X (Sample) for Models I, II and III predicting presence of Pines. The central column corresponds to the mean value. The column on the left and right corresponds to quantiles: 0.025 and 0.975, respectively.



(a) MaxEnt **(b)** Model I

(c) Model II **(d)** Model III

Fig. 3.11 Area under the receiver operating characteristic curve (AUC-ROC) for the different models of Pines. The three models (b,c and d) perform significantly better than MaxEnt.

3.11 Estimates for the predicted presence of tyranids using birds records as sample

Table 3.4 Posterior summaries of all the parameters in model I with the associated 95% credible intervals for the example of flycatchers. Parameters τ_Y^2 and τ_X^2 correspond to the variance of the spatial effects of the presence and the sample process (S_Y and S_X) respectively. Likewise, σ_Y^2 and σ_X^2 correspond to the variance of the unstructured processes Z_Y and Z_X respectively. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3

	Median	2.5%	97.5%	n.sample	%accept	n.effective	geweke.diag
(Intercept X)	-1.2410	-2.7526	0.0656	10000	59	7.7	3.0
Dist.to road	-0.0001	-0.0002	0.0000	10000	59	1329.3	1.7
Population	0.0000	-0.0001	0.0001	10000	59	1242.7	0.1
τ_Y^2	9.8274	5.3185	13.8716	10000	100	13.2	0.0
σ_X^2	0.0063	0.0014	0.0196	10000	100	4.3	6.4
(Intercept Y)	-0.4842	-1.4833	0.6361	10000	57.9	20.3	8.6
Elevation	0.0000	-0.0002	0.0002	10000	57.9	309.5	0.5
Precipitation	0.0001	-0.0001	0.0003	10000	57.9	143.8	-3.4
τ_Y^2	1.9098	1.0779	3.6263	10000	-	8.6	-0.4
σ_Y^2	0.5745	0.0867	1.8564	10000	-	3.4	-4.8

Table 3.5 Posterior summaries of all the parameters in Model II with the associated 95% credible intervals for the example of flycatchers. The parameter τ^2 represents the variance of the common spatial effect. Parameters σ^2 and σ^2 correspond to the variance of the unstructured process Z_Y and Z_X. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3

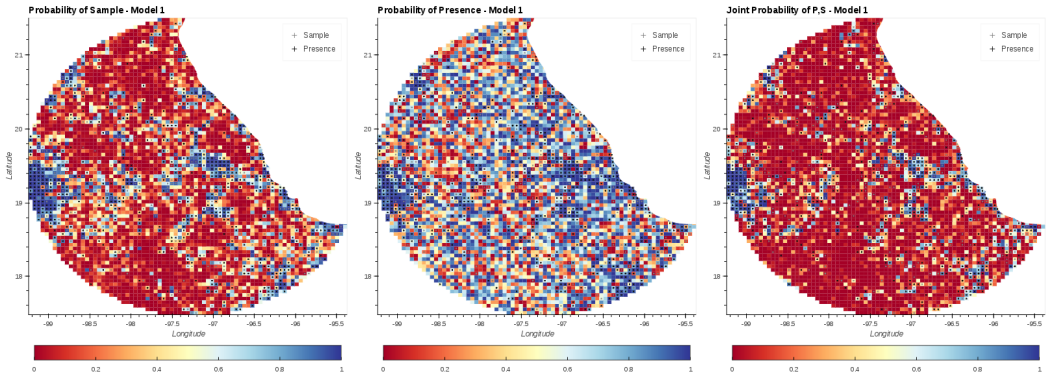
	Median	2.5%	97.5%	n.sample	%accept	n.effective	Geweke.diag
(Intercept)	-1.6937	-2.1358	-1.3629	10000	47.6	68.7	4.7
Dist to road	-0.0001	-0.0002	0.0001	10000	47.6	443.7	-0.8
Population	0.0000	-0.0001	0.0001	10000	47.6	300.6	-1.4
Elevation	-0.0001	-0.0003	0.0001	10000	47.6	175.3	1.6
Precipitation	0.0000	-0.0001	0.0002	10000	47.6	192.1	2.4
τ^2	10.1800	7.3033	14.9518	10000	-	18.8	-3.8
σ^2	0.0089	0.0022	0.0829	10000	-	1552.6	0.4

Table 3.6 Posterior summaries of all the parameters in Model III with the associated 95% credible intervals for the example of flycatchers. Parameters σ^2_Y and σ^2_X correspond to the variance for the presence (Y) and the sample (X). The term $\text{corr}_{X,Y}$ indicates the correlation between these two processes. Significant parameters for the fixed effect are shown in **bold**. For further information see section: 3.3

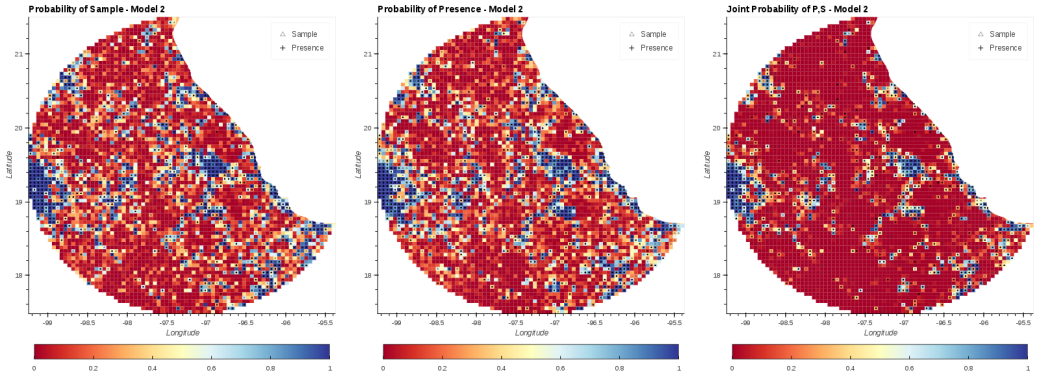
	Median	2.5%	97.5%	n.sample	%accept	n.effective	Geweke.diag
(Intercept Y)	-0.9374	-1.6520	-0.2057	5000	53.3	110.0	1.0
Elevation	0.0000	-0.0002	0.0002	5000	53.3	88.5	-1.2
Precipitation	0.0001	-0.0001	0.0003	5000	53.3	150.2	-2.0
(Intercept X)	-1.4153	-1.9346	-0.9441	5000	53.3	85.2	0.4
Dist. to road	-0.0001	-0.0002	0.0000	5000	53.3	523.5	0.5
Population	0.0000	-0.0001	0.0001	5000	53.3	232.1	-1.0
σ^2_Y	3.5179	2.7614	6.0832	5000	-	5.6	-0.7
σ^2_X	7.3840	5.9431	12.1276	5000	-	7.1	-0.6
$\text{corr}_{Y,X}$	-	-	-	-	-	-	-

3.11.1 Maps of posterior probabilities for Tyranids

(a) Model I



(b) Model II



(c) Model III

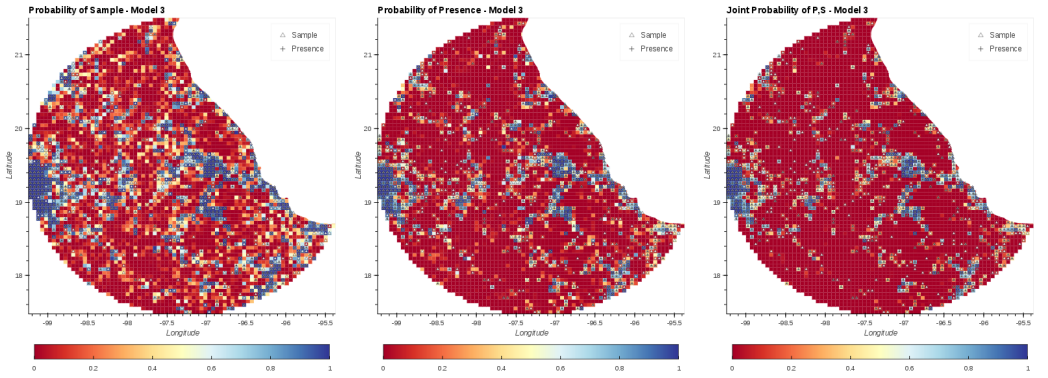
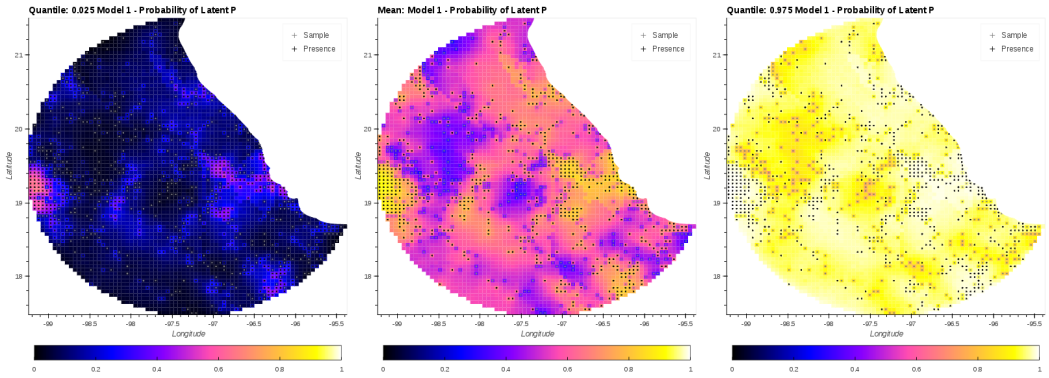
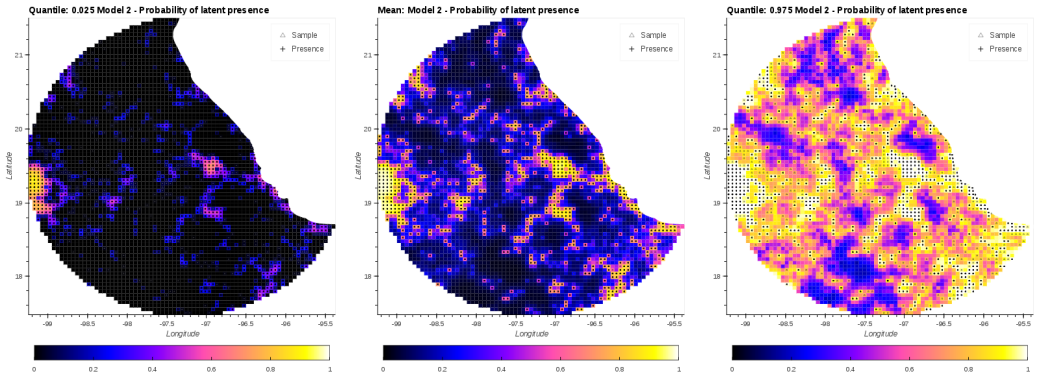


Fig. 3.12 Mean probability and 95% C.I. for Presence, Sample, and Joint presence and sample for Models I, II and III predicting presence of flycatchers (Family: Tyrannidae) using birds (Class: Aves) as sample.

(a) Model I



(b) Model II



(c) Model III

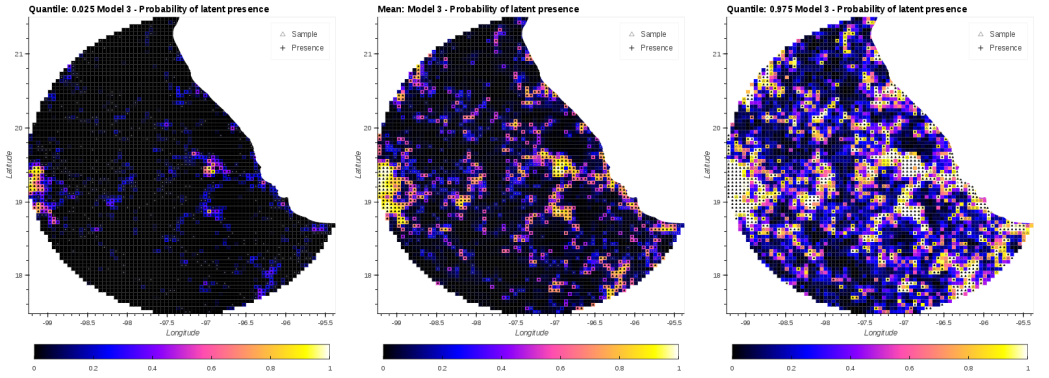
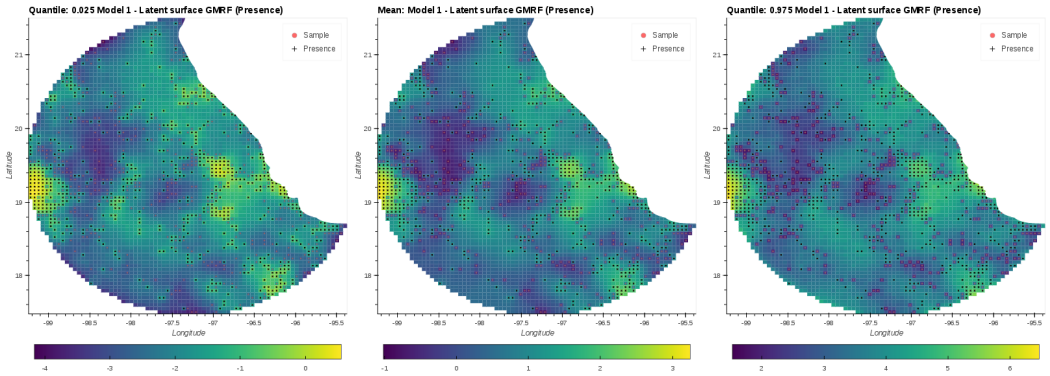
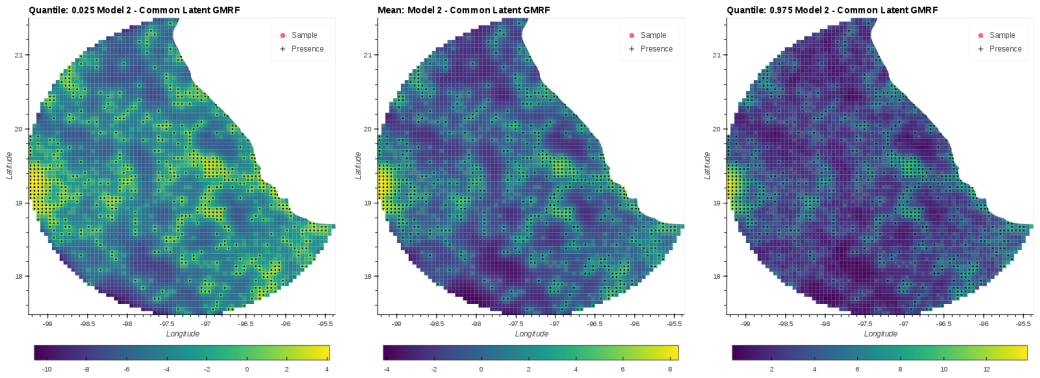


Fig. 3.13 Latent variable P_Y (Presence) for Models I, II and III predicting presence of flycatchers (Family: Tyrannidae). The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively.

(a) Model I



(b) Model II



(c) Model III

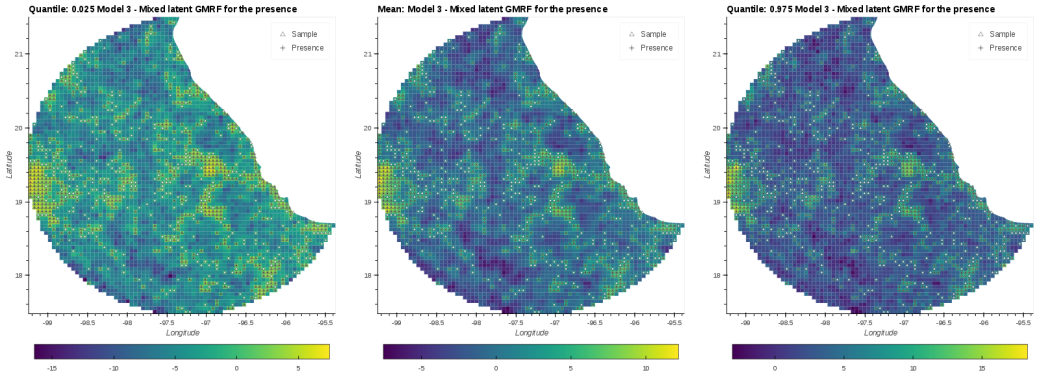
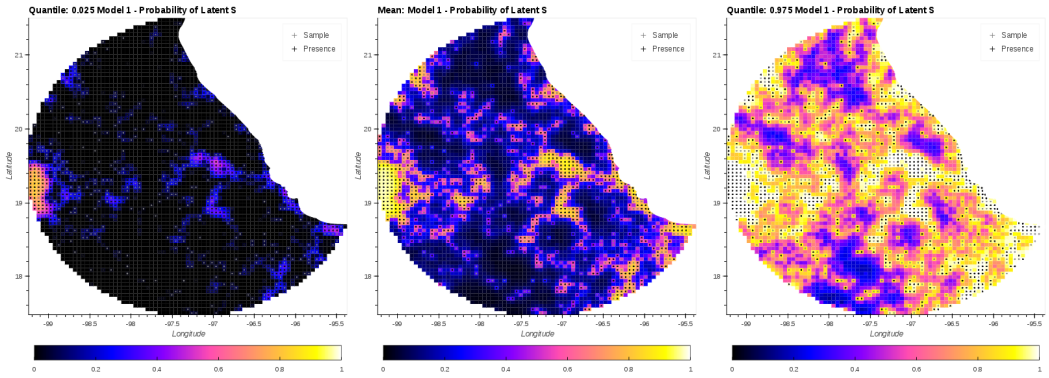
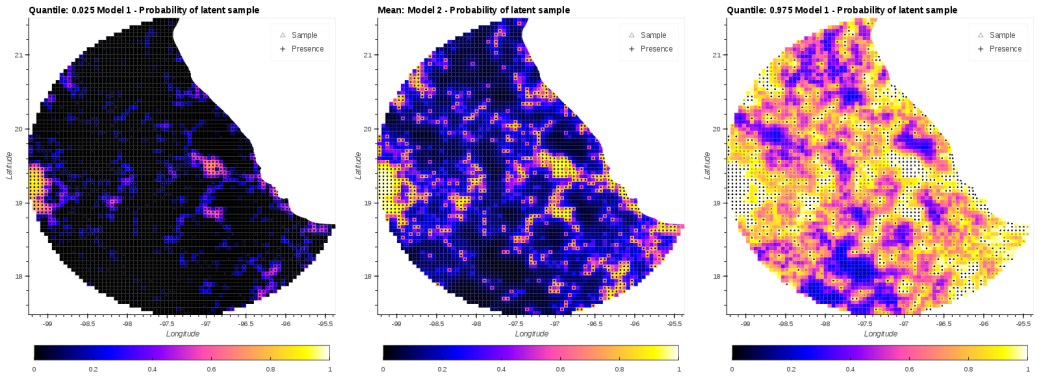


Fig. 3.14 Spatial random effect S_Y. The Gaussian Markov random field (GMRF) corresponding to the latent variable P_Y (Presence) for Models I, II and III predicting presence of flycatchers (Family: Tyrannidae). The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively.

(a) Model I



(b) Model II



(c) Model III

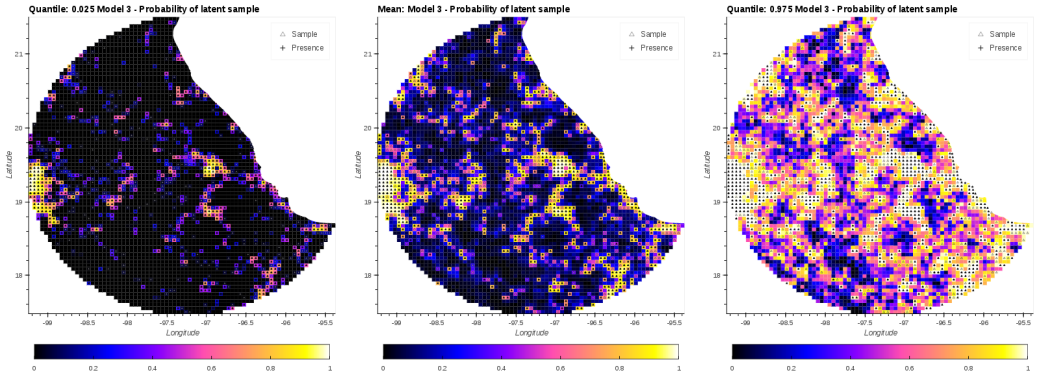
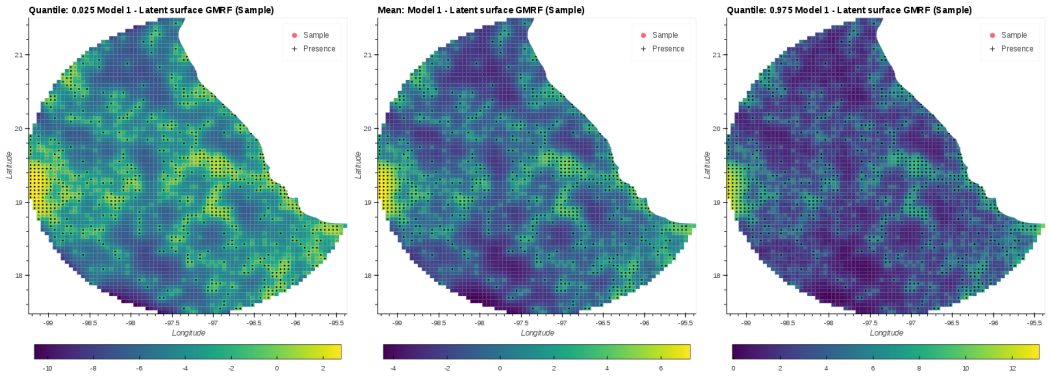
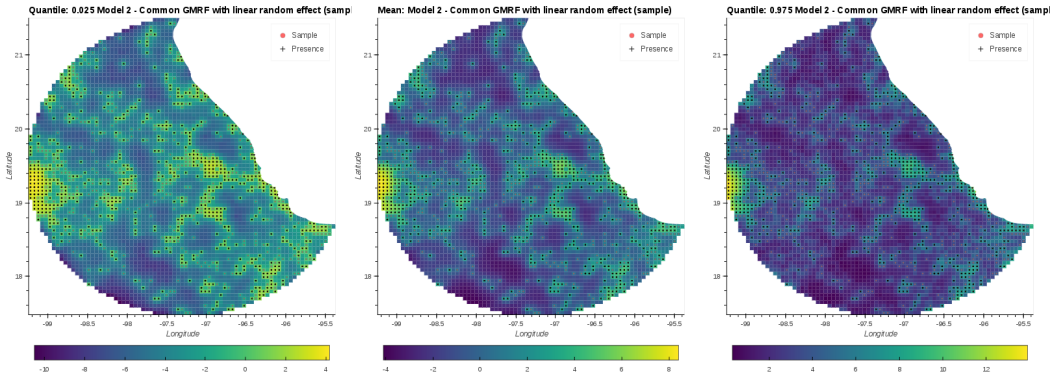


Fig. 3.15 Latent variable P_X (Sample) for Models I, II and III predicting presence of flycatchers (Tyrannidae) using all birds as sample. The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively.

(a) Model I



(b) Model II



(c) Model III

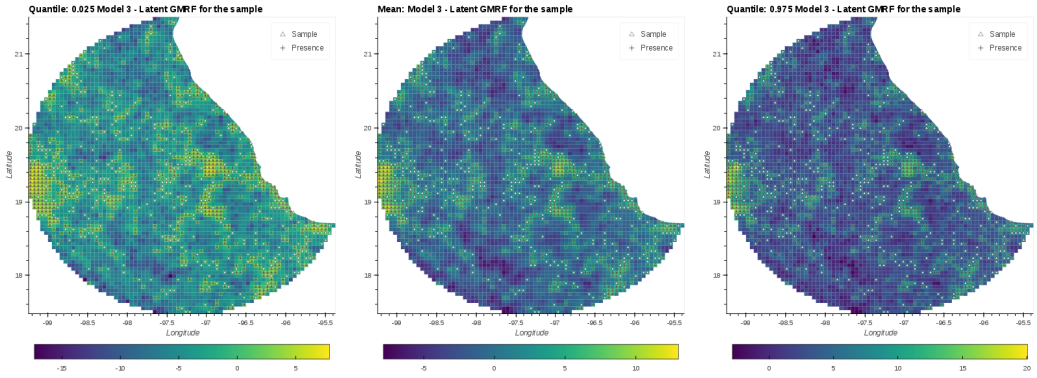
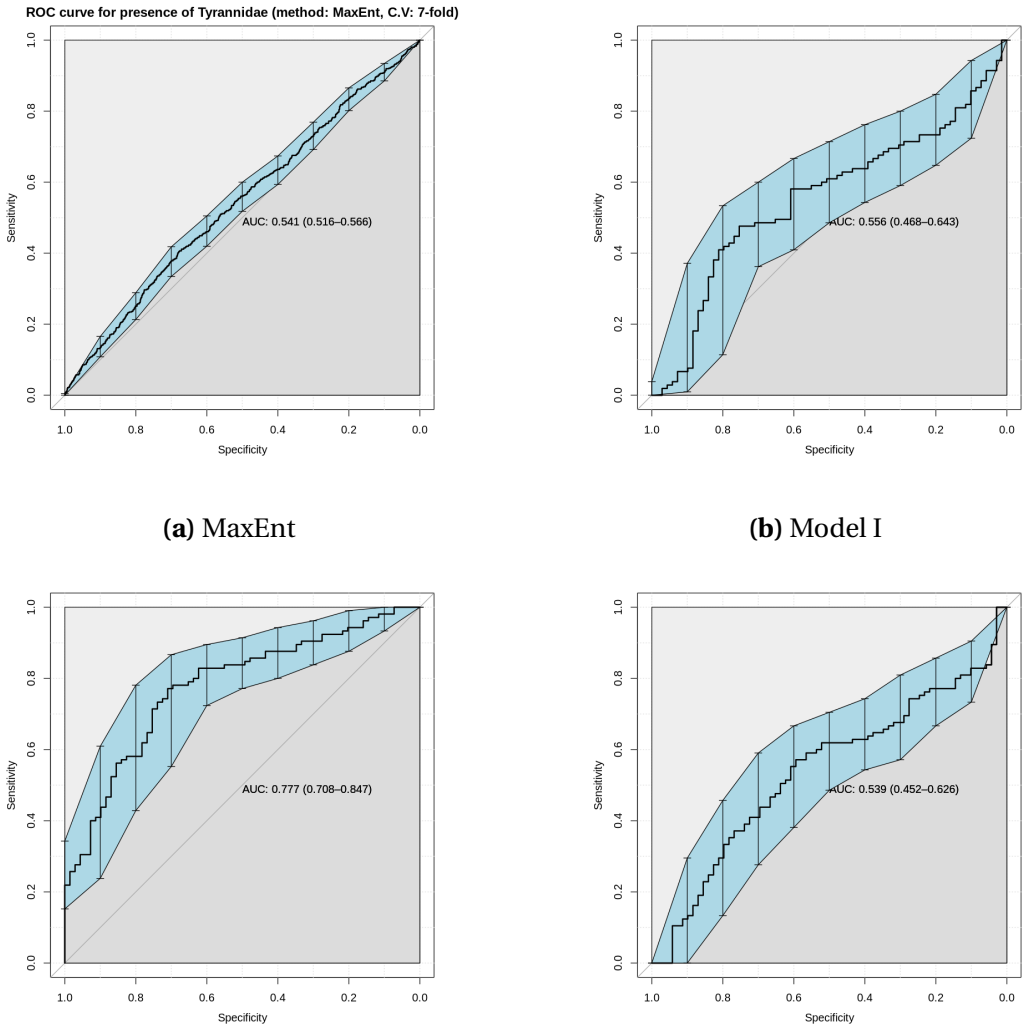


Fig. 3.16 Spatial random effect S_X. The Gaussian Markov random field (GMRF) corresponding to the latent variable P_X (Sample) for Models I, II and III predicting presence of flycatchers (Tyrannidae). The central column corresponds to the mean value. The columns on the left and right correspond to quantiles: 0.025 and 0.975, respectively.



(a) MaxEnt

(b) Model I

(c) Model II

(d) Model III

Fig. 3.17 Area under the receiver operating characteristic curve (AUC-ROC) for MaxEnt and models I, II and III of flycatchers. MaxEnt and models I and III achieved low AUC. Although, on average models I and III outperformed MaxEnt, their variances show that these models are not appropriate when the proportion of missing data is significantly higher than the presences. See the discussion section for a more detail explanation.

CHAPTER 4

A TAXONOMIC-BASED JOINT SPECIES DISTRIBUTION MODEL FOR PRESENCE-ONLY DATA

State of publication

The following chapter is a facsimile of the draft reviewed by all the co-authors. Its target journal is *Methods in Ecology an Evolution* or *Ecography*, indexed in the Q1 Ecological Modelling catalog of Scimago Journal Rank (SJR) (2018).

Keywords: Joint Species Distribution Models, Presence-only data, Model-based spatial autocorrelation models, Multivariate conditional autorregresive models

A taxonomic-based joint species distribution model for presence-only data

Juan M. Escamilla Molgora[a,b,1,*], Luigi Sedda[c,2], Peter Diggle[b,3], Peter M. Atkinson[d,4]

[a]*Lancaster Environment Center, Lancaster University, Lancaster LA14YQ, UK*
[b]*Centre for Health Informatics, Computing and Statistics (CHICAS), Lancaster Medical School, Faculty of Health and Medicine, Lancaster University, Lancaster LA1 4YQ, UK*
[c]*Lancaster Medical School, Faculty of Health and Medicine, Lancaster University, Lancaster LA1 4YQ, UK*
[d]*Faculty of Science and Technology, Lancaster University, Lancaster LA1 4YR, UK*

Abstract

- Species distribution models (SDMs) are essential tools for assessing a variety of ecological problems (e.g. mapping species, assessments of biodiversity loss and conservation planning). In general, SDMs assume that the presence of a species is determined fully by environmental conditions, disregarding dependencies between different species. Modelling these dependencies is achieved by joint species distribution models (JSDMs) that simultaneously inform multiple species with their associated occurrences. Recently, several frameworks for JSDMs have been proposed. However, all of them require information on species presence and absence. No extension for JSDM using presence-only data has been published.
- Here, we propose a JSDM for presence-only data using a multilevel hierarchical model that combines two components: an ecological suitability process that models the probability of a given taxon to occur within certain environmental conditions, and a sampling effort process that models the probability of a site to be sampled based on anthropological covariates. Both components inform each other through a common spatial random effect that captures the unobserved effect of biotic interactions. The absences of the target species were modelled with a taxonomic-based algorithm that uses surveyed places unlikely to contain the taxa of interest.
- The model was validated with simulated data and later applied to a case study in central Mexico focusing on five taxa: oaks (genus Quercus), pines (family: Pinacea), leadtrees (genus Leucena), leafnose bats (family: Phyllostomidae) and woodpeckers (family: Picidae)
- The model was able to identify (and remove) the influence of the sampling effort from the ecological process that determines the presence of a taxon. The removal of this effect was surprisingly effective in urban areas, were the high abundance of observations bias the estimates of presence. Additionally, the spatial patterns of the ecological process were consistent with the theoretical biogeographical patterns of the investigated taxa. Our model can be

*Corresponding author
Email addresses: j.escamillamolgora@lancaster.ac.uk (Juan M. Escamilla Molgora), l.sedda@lancaster.ac.uk (Luigi Sedda), p.diggle@lancaster.ac.uk (Peter Diggle), pma@lancaster.ac.uk (Peter M. Atkinson)
[1]https://orcid.org/0000-0002-3682-9828
[2]https://orcid.org/0000-0002-9271-6596
[3]https://orcid.org/0000-0003-3521-5020
[4]https://orcid.org/0000-0002-5489-6880

used to infer the presence of multiple species across space in cases were the only available information are species occurrences. It can also determine the degree of contribution of the sampling effort in the overall estimation of the species presence.

Keywords: Species Distribution Models, Presence-only data, Tree of life, Multivariate conditional autorregresive models,

4.1 Introduction

Estimating the geographic distribution of species, conditioned to their ecological niche is crucial for risk assessments of species extinctions, conservation planning, habitat restoration and forecasting the effects of climate change on biodiversity (Benito et al., 2009; Elith et al., 2006). Species distribution models (SDMs) are quantitative tools designed for these purposes and have become essential tools for decision and policy-making in regional to global biodiversity assessments (Araújo et al., 2019).

SDMs have been shown to be effective in characterising the natural distributions of species when the sampling has been properly designed to fit the model's assumptions (Elith and Leathwick, 2009; Guisan and Zimmermann, 2000). SDMs are often limited to the use of presence-absence observations to predict single species under two theoretical assumptions: *i)* that the probability for a target species S occupies a given area is independent from other species (Gelfand et al., 2006; Guisan and Thuiller, 2005) and *ii)* S is at *equilibrium* with their environment. That is, the species S occurs in all environmentally suitable areas and is absent in unsuitable environments (Hutchinson, 1957).

Although existing SDMs are statistically sound, their reliance on presence-absence observations reduces the application to surveys where there is certainty about absences which represent unsuitable environments for the target species to exist. As such, obtaining presence-absence data is a hard and expensive task. Additionally, the content and uncertainties depends on several factors like the study design, its extension and species of interest, among others. In this regard, presence-only occurrence data are far more widely available and accessible. An increasing number of centralised and open repositories such as: the Global Biodiversity Information Facility GBIF (GBIF Secretariat, 2015), eBird for bird sightings (Hudson et al., 2014),the PREDICTS global database on terrestrial biodiver-

sity (Sullivan et al., 2009) and the Disease Vector Database (Moffett et al., 2009), have been released to aid the endeavour of mapping life in Earth.

Although valuable ecological information exists in presence-only records, they are prone to several problems related to heterogeneous sampling design, bias in space and time, and detectability among species (Beck et al., 2014; Dickinson et al., 2010; Franklin et al., 2016; Isaac and Pocock, 2015). Specifically, presence-only observations do not inform on real absences or missing observations. Thus, failure in specifying absences adequately can result in models that are not identifiable (see Phillips and Elith (2013); Ward et al. (2009)). Approaches to solve this problem for single (univariate) SDMs involves the use of informative *background* data as a proxy to model absences (see (Croft et al., 2019; Renner et al., 2019; Ward et al., 2009)).

Another limitation of SDMs relates to the assumption of being in *equilibrium with the environment*. Extrapolating species habitat to new environmental conditions (e.g. affected by climate change or by invasive species) violates the assumption of equilibrium (Elith and Leathwick, 2009). This implies that, using only environmental predictors for modelling species distributions without the support of existing knowledge of the species' ecology is not enough to extrapolate the targeted species distribution, exposing the limitations of using single SDMs to infer the probability of occurrence outside the observed environmental range.

Araújo et al. (2005); Chase and Myers (2011); De Marco et al. (2008); Dormann (2007); Elith and Leathwick (2009) have shown the importance of integrating elements of ecological community theory, such as: the effect of biotic interactions, intrinsic phenotipic plasticity and evolutionary relationships in a new generation of SDMs capable of supporting multiple species out of environmental equilibrium. A primer for this was reviewed by Ferrier and Guisan (2006) in an influential synthesis. They described three modelling strategies for combining data from multiple species to produce information on spatial

distributions integrated in a collective community-level model. Broadly, these strategies are: *assemble first, predict later* where biodiversity data are first analysed and processed to obtain a community-level attribute used for prediction (e.g. a measure of richness); *predict first, assemble later* where species occurrences are first predicted using classic univariate SDMs and then integrated as a predicted output as a community-level variable; and *assemble and predict together* in which all the species of interest are modelled simultaneously using community-level information. From a statistical perspective, this last strategy is achieved, with the modelling of the joint probability distribution of multiple species occurrences and is therefore called: *joint species distribution models* (JSDM).

Disregarding the effect of species interactions can lead to inconsistent results Clark et al. (2014). In contrast, JSDMs can use the information contained in the implicit dependencies between species, considering other effects not explained by the predictor variables. This is has been shown to be effective in the modelling of rare species, where the observations of other more conspicuous co-occurring species inform the likelihood of the rarest ones (Hui et al., 2013; Ovaskainen and Soininen, 2011). For these reasons, JSDMs provide more flexibility for simultaneously assembling multiple species supported in ecological theory, resulting in models with greater inferential and predictive power (Warton et al., 2015).

Multivariate generalised linear mixed models (GLMMs) and latent variable models (LVM) have been used to model a wide range of JSDMs (see review by Warton et al. (2015)) From a statistical perspective, both GLMMs and LVMs can be specified as hierarchical models, using different levels of random effects to capture correlations between distinct taxa and ecological relationships, demonstrating to be effective in modelling uncertainty (Cressie et al., 2009). For example, Aderhold et al. (2012) proposed a model for reconstructing species interaction networks using Bayesian changepoint frameworks, and Jamil

et al. (2013) used GLMMs to incorporate the effect of species traits in response to the environment and other species occurrences.

Spatial autocorrelation is acknowledged to be an important random effect that captures the geographic variation not explained by environmental predictors (Elith and Leathwick, 2009; Legendre, 1993). Several spatial modelling approaches have been proposed in the literature (Gelfand et al., 2006; Golding and Purse, 2016; Illian et al., 2013; Lichstein and Simons, 2002).

Recent advances in high performance computing and computational statistics have opened the possibility for inferring complex statistical models using Markov chain Monte Carlo (MCMC) methodologies. These advances have led to the development of novel methods in JSDMs. One of the first attempts was developed by Latimer et al. (2009) using a hierarchical approach for binary responses (presence-absence) and a geostatistical model for co-regionalization (Wackernagel, 2003) to model a spatial effect per species. Later, Clark et al. (2014) proposed a hierarchical model for abundances and presence-absence for multispecies using a zero-inflated Poisson process to account for the bias in the number of zeros related to abundance data. An approach by Thorson et al. (2015) and later independently by Ovaskainen et al. (2016) used latent factors to model the whole community level, with a single parameter exponential spatial covariance function for each latent factor. This research has been improved recently in Tikhonov et al. (2020) using Gaussian predictive processes a nearest neighbour Gaussian process as spatial latent factors. Although, these approaches provide deeper understanding of the ecological processes at different scales (i.e from community to species), they do not support presence-only data. To our knowledge, joint species distributions models for presence-only data have received relatively little attention.

Here, we propose a hierarchical multilevel model for multiple species distributions using presence-only data. Our approach uses the taxonomic tree of the taxa of interest

to obtain an *intrinsic* informative sample (i.e. independent from external information) to inform the likelihood of all taxa jointly. The intrinsic informative sample serves as background information (in the sense of Ward et al. (2009)) to define an identifiable JSDM based on presence-only data using the evolutionary (taxonomic) structure of the natural classification (De Queiroz and Gauthier, 1990).

The paper is structured as follows. A description of the model is presented in section 4.2. Section 4.3 tests the model performance on simulated data. The model was applied to a case study using occurrence data from central Mexico in section 4.4. Finally, section 4.5 discusses the implementation, findings in the study case, and future research which builds on the fundamental approach introduced here.

4.2 Methods

Inference on species distribution models using presence-only data require the use of additional data to specify an identifiable model (Ward et al., 2009). Here, we propose the use of taxonomically-related observations as auxiliary data for modelling a preferential sampling process (an extended case of (Escamilla Mólgora et al., 2020) for multiple species). Preferential sampling arises when the assumption of independence between the sampling process and the process of interest (in this case the ecological suitability) is not valid. That is, either the sampling process (S) is stochastically dependent on the ecological process (P) or, vice versa, the ecological process is influenced by the sampling. In statistical terms, preferential sampling implies that the joint process $[P,S] \neq [P][S]$. As we are dealing with presence-only observations the information of the sampling effort and the ecological process is intertwined in the data. As such, the data provide no evidence for assuming independence between S and P and, therefore, the presence-only data present, intrinsically, a preferential sampling.

Our model assumes that the observed records of organisms are determined by the joint effect of an ecological suitability process for each taxa (e.g. species) to settle (or occupy) a place, and a preferential sampling process that biases the occurrence of observed records with respect to their true ecological occurrences. To account for this effect, the model defines explicitly a mixed latent variable that expresses the proportional effect of the ecologically suitability for presence and the preferential sampling process. The model is able to infer the presence of multiple taxa simultaneously. Multiple species models involve extensive computational power.

Another important aspect to account for is the variability across space. This phenomenon is commonly known as the *First Law of Geography* and states that: *"everything is related to everything else, but near things are more related than distant things"* (Tobler, 1970). This *law* refers to the empirical evidence that nature varies differently across space. Acknowledging that it is the norm, rather than the exception, that nearby observations are more correlated to each other than to the more distant ones. Statistically speaking, the ubiquity of variability in space implies that the assumption of homoskedasticity (i.e. constant variance between all observations) is not satisfied and, by the Gauss-Markov theorem, the linear estimators of the model are not guaranteed to be unbiased, making the use of ordinary linear regression inappropriate. To account for this effect, we introduce a spatial process (G) that assigns correlations between observations based on neighbouring relationships to characterise the residual spatial heterogeneity (spatial random effect). The spatial process incorporates a parameter (λ_G) that modulates the proportion of the spatial variability This parameter give the capability to identify spatial and unstructured variability that arises from interactions between taxa and bias within the preferential sampling.

To reduce the computational complexity of the model, we specified a unique spatial process common to all taxa, including the preferential sampling process. Additionally,

the process is specified as a proper conditional autoregressive model (CAR) (Besag, 1974). In this sense, the inference is performed efficiently with sparse numerical methods that greatly reduce the processing power for matrix inversion. For more information of how this model is specified refer to the supplementary materials 4.11.

The model introduces a novel approach for addressing the problem of preferential sampling. It uses the natural taxonomic classification of life, a classification based on the evolutionary relationships between organisms, to determine a set of informative taxa. This set, called *complementary* taxa, and their corresponding observation records (hereafter called *intrinsic* sample observations) constitute an informative sample relative to the closest common ancestral node of the taxa of interest. The intrinsic sample observations are different from the observations of the taxa of interest and constitute informative records used for fitting the preferential sampling process. In this sense, the likelihood of the sampling effort uses observations derived from the union of the complementary groups relative to the selected taxa. The directed acyclic graph representing the model is pictured in figure 4.1.

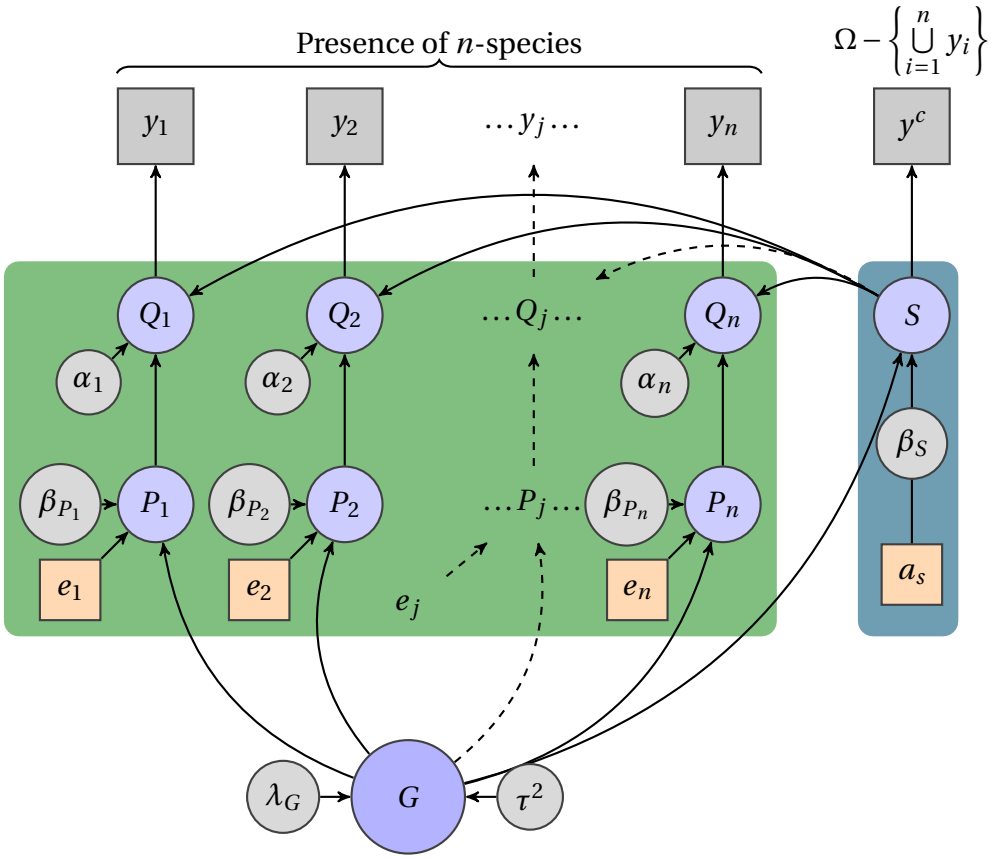


Fig. 4.1 Directed acyclic graph representing the multi-species model. Nodes in squares indicate data. Blue squares are observed records. y_i are the presences of species i and y^c are the complementary records of $\bigcup_{i=1}^{n} y_i$, i.e. the records that are not from species $1, \dots$ nor n, relative to the available dataset and an arbitrary taxonomic branch (Ω). Orange squares are covariates, e_i for environmental based and a_s for anthropological based, associated with the sampling effort (dark blue block). Nodes in blue circles represent latent variables where: Q_i is the mix between the sampling effort S and the corresponding ecological suitability process P_i. The node G represents the spatial random effect (CAR) shared between both the sampling effort components (dark blue block) and the ecological components (green block). Circular grey nodes represent the parameters used by the latent variables.

The model specification in figure 4.1 factorises the joint probability distribution in:

$$[\mathbf{y}, Q, P, S, G, \alpha, \beta_p, \beta_s, \tau^2, \lambda_G; d_e, d_a, \mathbb{W}] = \prod_{i=1}^{n} [y_i|Q_i][Q_i|P_i, S, \alpha_i] \tag{4.1}$$

$$[P_i|G, \beta_{p_i}; d_{e_i}][S|G, \beta_s; d_a][\beta_{p_i}][\beta_s] \tag{4.2}$$

$$[G|\lambda_G, \tau^2; \mathbb{W}][\lambda_G][\tau^2] \tag{4.3}$$

Data-related variables are written in lower case letters while latent variables are written in upper case. In this sense, **y** represents the observations of all (n) taxa of interest. Latent variables Q, P, S and G correspond to the mixing process, ecological suitability, sampling effort and spatial random effect, respectively. The terms in 4.1 refer to the distributions of likelihood and mixture (Q_i) between the latent variables P and S. Terms in 4.2 refer to these two processes and the corresponding prior distribution of their parameters (βs). Terms in 4.3 define the spatial autocorrelation process (G). The parameters in Greek letters are properties of the latent variables. The variables d_e, d_a and $\mathbb{W}$ represent data for environmental and anthropological covariates and the spatial lattice structure, respectively. The full description of the model is detailed in the supplementary materials 4.11.

4.2.1 Support for missing data

The model allows inference on locations where information about the presence or absence of a given taxa (or sampling effort) is unknown. This approach treats missing observations as additional parameters to the model. The approach is similar to the data augmentation scheme proposed by Tanner and Wong (1987). It uses the information provided by the latent variables (Q, P, S and G) to sample posterior distributions at the locations with missing information. The scheme is incorporated into the MCMC-based inference, along with the rest of the parameters. Aided by the spatial autocorrelation structure (G) and the data from nearby areas, the fitted posterior distribution can provide information about the presence (or absence) at sites with missing data. Refer to supplementary materials 4.11 for a formal specification of the posterior distribution on sites with missing data.

4.2.2 Obtaining the sampling effort with complementary taxa

We propose to use the systematic classification of life to derive informative observations for the *sampling effort* process. This methodology aims to generate information related to absence and, thereby, reduce the bias in presence-only data by including taxonomically related taxa sampled in the same region. The informative sample is obtained from the taxonomic tree of life (ToL) and represents an *intrinsic* informative sample that models the likelihood of the sampling effort (S) and the ecological suitability (P) processes by providing complementary information relative to the taxa of interest and the rest of the available dataset. The response vector corresponding to the intrinsic informative sample is called the *complementary sample* and it is a binary (presence-absence) vector aggregated per location (unit area). In the reminder of this research we assume that there is only one taxonomic tree of life (unique) and it does not change in space-time (static).

Obtaining the complementary sample

Let N_{y_i} be a node in the ToL corresponding to taxon y_i, for example, a particular type of family (Asteracea) or phylum (Chordata). In this sense, $N_y = \{N_{y_1}, ..., N_{y_n}\}$ is the set of all nodes corresponding to each taxon of interest y_i. By the properties of the ToL (Chapter 2, supplementary materials), with the exception of the root node (i.e. all life), there exists a node $\widetilde{N_y} \in$ ToL which is an ancestor to all nodes in N_y. If we consider the subtree generated [1] by $\widetilde{N_y}$ (hereafter defined as $T(\widetilde{N_y})$), then N_y is contained in the nodes of $T(\widetilde{N_y})$ (i.e. $N_y \subseteq \text{Nodes}(T(\widetilde{N_y}))$). On the other hand, it is possible that $\text{Nodes}(T(\widetilde{N_y}))$ includes nodes that are not elements of N_y. We denote this set of nodes as N_y^c, that is, the complement of N_y. Although N_y and N_y^c have no common nodes, all of them are taxonomically related; by construction all share the same ancestor $\widetilde{N_y}$. The derived

[1] All the paths connecting $\widetilde{N_y}$ to the leaves of the ToL.

taxonomic subtree $T(\widetilde{N_y})$ contains the nodes of interest N_y and its complement N_y^c given the taxonomic structure.

From a practical perspective, the content of $T(\widetilde{N_y})$ is subject to the available data, hereafter referred as Ω (data universe). In this sense, the content of $T(\widetilde{N_y})$ depends on each user and case study. For example, if the user has access to a local database with a handful of taxa; Ω will be small, whereas, if the user has access to a massive repository of, say, biodiversity records, the size of Ω could potentially reach millions of nodes.

For large datasets Ω and, therefore, $T(\widetilde{N_y})$ could be previously filtered by any relevant restriction R such as: the spatial extent of the area under study (i.e. the lattice $\mathbb{W}$), dates of data acquisition or ecological relationships like membership to an ecosystem type, community or functional group. In this sense, the nodes of $T(\widetilde{N_y})$ constitute a total set with respect to the taxa of interest, N_y and Ω_R (i.e. the Ω restricted by R). N_y^c can be defined as:

$$N_y^c = \{n_i \in \text{Nodes}(T(\widetilde{N_y}))|n_i \notin N_y \wedge R(N_y)\} \tag{4.4}$$

That is, the nodes of the subtree generated by $\widetilde{N_y}$ (ancestor of all the taxa of interest) that are not taxa of interest and satisfy the restriction $R(\cdot)$. In this sense, the observations (records) of N_y^c can support the likelihood of the sampling effort S with informative data derived intrinsically by the taxonomic classification of the natural classification. Algorithm 2 describes the procedure for obtaining an *intrinsic* complementary sample y^c given Ω_R and N_y.

4.2.3 Model Implementation

The model was implemented in STAN (Carpenter et al., 2017), a Turing complete probabilistic programming language for specifying statistical models. STAN performs full Bayesian inference using Markov chain Monte Carlo methods such as Hamiltonian Monte Carlo sampling. It also includes the $\hat{R}$ statistic (Gelman et al., 1992) as a robust diagnos-

Algorithm 2 Obtaining the complementary sample y^c from an intrinsic response vector (`y_c`) for inferring the likelihood of the *sampling effort* process (S). For notation refer to section: 4.2.2

Require: $N_y, \Omega_R, \text{ToL}(\Omega)$
  Get the ancestor node from all nodes in N_y
  $\widetilde{N_y} \leftarrow \text{getAncestor}(N_y)$
  Generate a subtree from the ancestral node
  $T(\widetilde{N_y}) \leftarrow \text{generateTree}(\widetilde{N_y})$
  Assign all the nodes of the subtree to a list
  $\text{nodesTN} \leftarrow \text{nodes}(T(\widetilde{N_y}))$
  Find the lowest taxonomic level from N_y
  $\text{lower\_level} \leftarrow \min\{\text{n.level} \mid \forall n \in N_y\}$
  Filter nodes with same taxonomic level as lower_level
  `complement` $\leftarrow$ [ ] {init. empty list}
  for ($i := 0$; $i :=$ size(nodesTN);i++) **do**
    Add the corresponding vector of records given the spatial lattice $\mathbb{W}$
    if (nodesTN[i].level == lower_level) **and** (nodesTN[i] **not in** N_y) **then**
      `complement[ i ]` $\leftarrow$ getRecords(nodesTN[i],$\mathbb{W}$))
    end if
  end for
  Aggregate complementary observations
  for ($i := 0$; $i :=$ size(`complement`); i++) **do**
    if sum(`complement[i]`) ≥ 1 **then**
      `y_c[i]` $\leftarrow 1$ {Assign 1 if there is at least one record}
    else
      `y_c[i]` $\leftarrow 0$
    end if
  end for

tic for chain convergence. The implementation code is located in the supplementary materials 4.1.

4.3 Validation with simulated data

To validate the model implementation we generated a synthetic dataset following the specification of the model. The model was fitted using Hamiltonian Monte Carlo approach that resulted in posterior samples for each model's parameter. All the parameters used for generating the synthetic dataset were inside the 95% credible interval of their corresponding fitted posterior sample. The complete specification, analysis, results of the simulation and work related to data acquisition are described in the supplementary materials 4.12.

Geographic lattice $\mathbb{W}$

The lattice $\mathbb{W}$ used in the simulation was obtained from a polygon intersected on a geographical grid of approximately 4 km spatial resolution. The region comprises the inland area of a circular polygon centered in central-eastern Mexico at 19N −97E with radius of 2° (ca.~ 200 km). The area covers approximately 112,000 km^2 and is composed of 4061 areal units (see figure 4.2l in supplementary materials I). To derive the associated adjacency matrix W we performed a topological analysis on the grid to determine the corresponding neighbours for each areal unit. To ease the processing work on data acquisition and transformation to the adjacency matrix representation we used *Biospytial*, a spatial graph-based computing engine for ecological data (Escamilla Molgora et al., 2020a). The engine was also used to generate the complementary samples as it implements several methods for selecting nodes and subtrees of the ToL as well as filtering queries by taxa and geographic location. The same geographical lattice used in the simulation was also used in the study case using real occurrence data (see section: 4.4).

4.4 Application to biodiversity occurrences in eastern Mexico

We now apply the multi-species model in a climatically and topographically diverse region in central-eastern Mexico using a selection of five taxa obtained from the Global Biodiversity Information Facility (GBIF).

4.4.1 Study region

The studied area has the same geographic extent and latticed tessellation as the one for the simulation (see subsection: 4.3). The area covers approximately 112,000 km^2 and intersects several Mexican states (e.g. Veracruz, Puebla, Hidalgo, Mexico City and Oaxaca). It includes heterogeneous landscapes with variability in geomorphological and climatic features as well as distinct biomes such as coastal dunes, chaparrales, mesophyl forests, evergreen rainforest, grasslands, mangroves, broad-leaf forests and coniferous forests (Rzedowski, 2006) and (INEGI, 2015). Figure 4.2-l shows the region over Mexico. The region under study is of ecological importance due to the confluence of the two biogeographic realms in the American continent; neartic and neotropical (Udvardy, 1975). Consequently, the region is rich in biodiversity at several taxonomic levels and, therefore, of high scientific interest. Additionally, the collection of GBIF records is highly abundant facilitating the acquisition of informative complementary samples.

4.4.2 Explanatory variables

The covariates used as explanatory variables for the ecological process were elevation and annual mean precipitation. The elevation data were obtained from the Global Relief Model *ETOPO1* at 1 arc-minute resolution (Amante and Eakins, 2009). The precipitation

data were obtained from the World Climatic Data *WorldClim* version 2 (Fick and Hijmans, 2017). These data are distributed as a 12 band raster model with c.a 1 km spatial resolution aggregated by monthly average values from the years 1970 to 2000. The anthropological covariates used to explain the sample process were: distance to the closest road and population counts. The distance to the closest road dataset was generated in two steps. First, we rasterised the National Road Network for Mexico (*Red Nacional de Caminos* (RNC) INEGI, Instituto Mexicano del Transporte and Gobierno de Mexico (2014), scale: $1:250000$) at 1 km spatial resolution. Later, we used this raster dataset to calculate its proximity to the closest road (pixels flagged as road) using the function `gdal_proximity` (GDAL/OGR Contributors, 2018). The road network data were obtained from: Vázquez (2018). The population dataset was obtained from the WorldPop project (Sorichetta et al., 2015) for the year 2010. The dataset consists of population counts on each areal element, each with a spatial resolution of 3 arc-seconds (c.a 100 m).

4.4.3 Occurrence and taxonomic data

The biodiversity data used were all the available GBIF occurrences (GBIF Secretariat, 2015) registered before January 2015, constrained to the studied region (4.3). The raw data were downloaded from GBIF (DOI:10.15468/dl.oflvla). For further information of this dataset, including all data attributions see GBIF.org (2016). Each GBIF record includes information on species name, location (coordinates in WGS84) and acquisition date. We parsed all the occurrences contained inside each area into a taxonomic-tree structure using the taxonomic classification of the GBIF Taxonomic Backbone (GBIF Secretariat, 2017). Therefore, for the 4061 area elements in the lattice ($\mathbb{W}$) we obtained the same number of taxonomic trees (hereafter referred as local taxonomic trees). To obtain the *complementary sample* (section 4.2.2) all the local taxonomic trees were merged into a single regional tree. The complementary sample was generated by applying algorithm

2 using as input: all the local trees, the regional tree and the taxa of interest. Within this framework, the taxa of interest are particular nodes of the regional tree.

4.4.4 Selection of taxa

We designed a *gold standard* selection of the taxa of interest by following the criteria : *i)* Each taxon should be abundant and distributed widely across the region, this to guarantee a fair number of observations. *ii)* The selected taxa should respond to known environmental factors, in particular elevation and precipitation. *iii)* Documented mutualistic relationships between the taxa exist. *iv)* Taxonomic diversity of the taxa should be preferred to ensure a diverse complementary sample.

To find a balance between the criteria, we decided to constrain the taxa at family or genus ranks. An exploratory analysis showed that the most abundant genera and families that satisfied the rest of the mentioned criteria were: Leadtrees (Genus: Leucaena, Family: Fabaceae), a type of shrub associated with tropical semi-deciduous forests and seasonal lowland forest. Leadtrees develop between sea level to 1400 m above sea level (MASL) (Niembro-Rocas et al., 2010). Oaks (Genus: Quercus, Family: Fagales), this group comprises trees and shrubs. They are frequently distributed between 1200 to 2800 MASL and between 600 to 1200 mm of precipitation per year. Oaks are associated with Pines in *mixed* forests. However, it is also common to find them in mesophyl forests, grasslands and woodlands (Rzedowski, 2006). Pines (Family: Pinacea), this group has, in general, affinity between temperate-to-cold dry climates, moderately moist and acid soils. They are associated with Oaks in *mixed* forests, as well as cypresses and spruces. Pines develop between 1500 to 3600 MASL and tolerate a wide range of precipitation conditions. Depending on the species this restriction can range from 350 mm to 1000 m (Rzedowski, 2006). Leafnose bats (Family: Phyllostomidae), this group constitutes the most diverse family of bats (Order: Chiroptera) and includes frugivorous, insectivorous and haematophagous.

Bats, in particular Phyllostomidae, is a taxon of high ecological importance. They provide key ecosystem functions like pollination, seed dispersal, nutrient cycling and arthropod suppression (Kasso and Balakrishnan, 2013; Kunz et al., 2011). Woodpeckers (Family: Picidae), this group was among the most abundant groups of birds. The selection of this group was its strong association with woodland forests, in particular with Oaks and Pines. The three groups of plants respond to gradients of elevation and precipitation while the two selected animals respond to ecological relationships with the associated biomes where the chosen plants are abundant. All the taxa have implicit ecological relevance as they shelter and give life support to other species.

4.4.5 Data preprocessing

The explanatory variables were spatially overlaid and aggregated by mean on each areal element. To obtain the vector of observations (Y_i) for each taxon of interest i a *point in polygon* test was performed. That is: $Y_i(x) = 1$ if taxon i is present in areal element x, otherwise $Y_i(x) = 0$. The test was applied to all taxa across all areal elements in the lattice. The data processing pipeline, as well as the generation of local and regional taxonomic trees was also undertaken with *Biospytial* (Escamilla Molgora et al., 2020a).

4.4.6 Model fitting

The response vectors together with their respective covariates were arranged in a design matrix with shape $(4061 * 6) \times (2 * 2)$, where 4061 are the number of areal elements of the lattice $\mathbb{W}$ and 6 corresponding to the five taxa plus the sample. The $2 * 2$ columns correspond to two columns for the ecological covariates and two columns for the anthropological covariates.

For fitting the model we used our implementation in the STAN language (see suplementary materials 4.1). We obtain posterior samples through MCMC using the NUTS

sampler on four independent chains with default parameters of step size and tree depth. The posterior sample was run for 100,000 iterations with a burn-in size of 50,000 and no thinning. The prior distributions for $\beta_{i\in\{1,\dots,n\}}$ are distributed $N(0, 10000)$. The prior distribution for parameters α_i (mixing process Q_i) and λ_G (proper CAR model) are beta$(5,5)$ and the parameter τ^2 is distributed as Inv. Gamma$(1, 0.01)$.

4.4.7 Cross validation with occurrence observations

We used k-fold ($k = 10$) cross-validation (Liu and Özsu, 2016) for evaluating the model's accuracy. This method partitions the data in $k = 10$ disjoint sets. On each iteration (fold), $k-1$ sets are used to fit the model, while the remaining one is used to assess the model's discrepancies between its predicted outcomes (scores) and the excluded (*missed*) observations. For a qualitative and quantitative assessment of the model's performance we used, respectively, the *receiver operator characteristic curve (ROC)* and its area under the curve (AUC), as it is a standard validation method in SDMs (Fielding and Bell, 1997).

4.4.8 Results

The posterior means and credible intervals of the model's parameters are shown in table 4.1. In all the MCMC chains (4), all parameters converged ($\hat{R} < 1.05$ (Vehtari et al., 2019)). Analysis among taxa showed that Leadtrees (Leucaena) obtained significant negative correlation with elevation and precipitation while leafnose bats (Phyllostomidae) showed positive correlation for precipitation and negative correlation for elevation. Oaks (Quercus) and pines (Pinacea) showed significant preference for higher elevations below the tree line. however, pines also showed significant preference for precipitation. Woodpeckers (Picidae) showed preference for higher precipitation but no significant preference for elevation.

The sampling effort was found to be significant for both covariates (i.e. distance to closest road and population density) with an increasing probability for getting samples in places close to roads (negative correlation) and with high population density (positive correlation).

In relation to the ecological suitability associated to each taxon, we found that pines obtained the largest contribution with respect to the sampling effort (mean 0.58 with 0.42, 0.78 at 95% CI). This was followed by oaks (mean 0.51 with 0.35, 0.65 at 95% CI). Leadtrees and bats obtained similar results (mean 0.44, 0.28,0.62 at 95% CI) while woodpeckers obtain the smallest value (mean 0.2, 0.1, 0.3 at 95% CI).

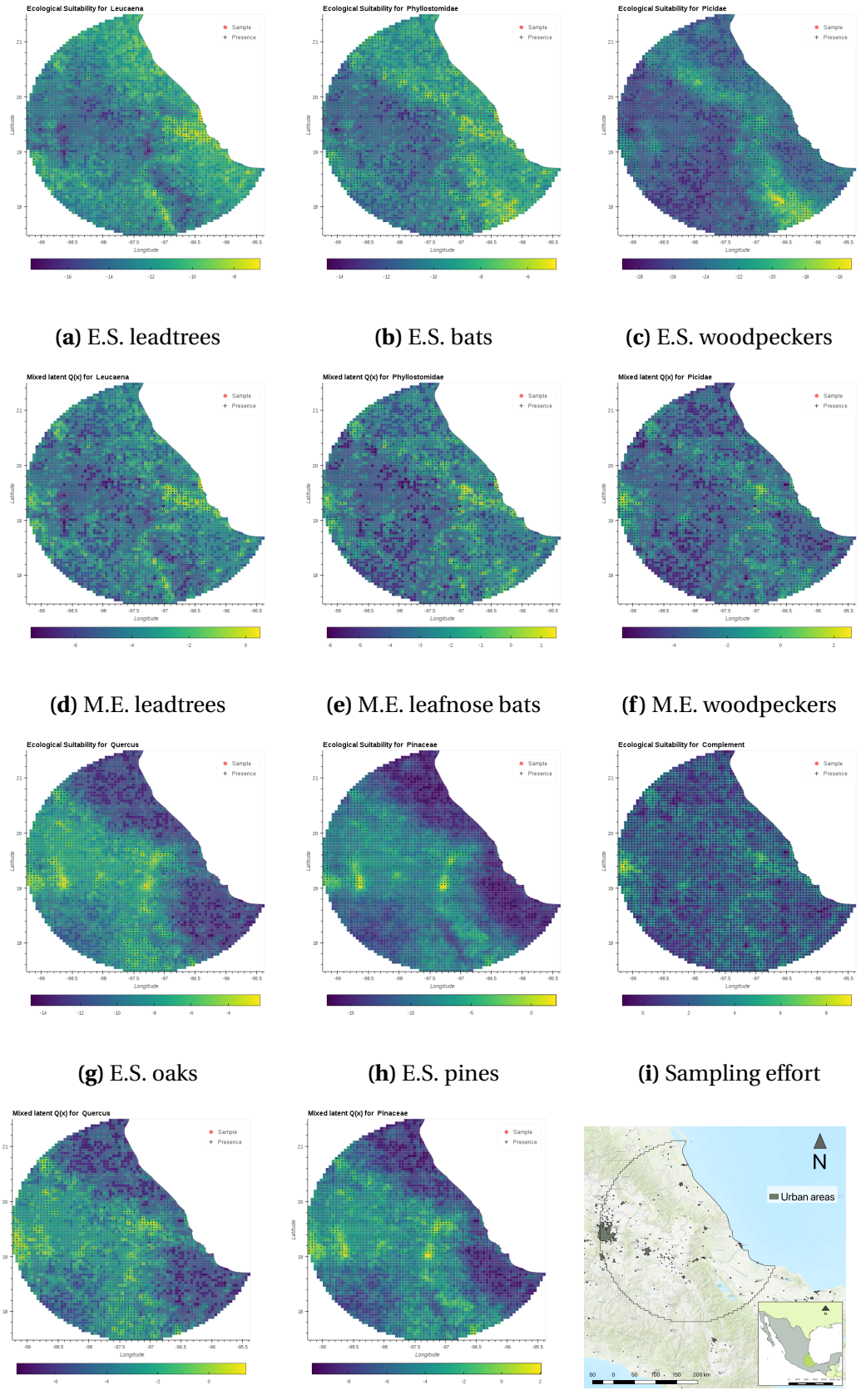
The posterior means for the spatial effect showed low spatial dependence (λ_G) of 0.13 ranged from 0.05 to 0.23 at 95% CI with an overall precision (τ^{-2}) of 0.1 ranged from 0.09 to 0.12 at 95% CI.

To show the model's capability to discriminate between the sampling effort process (S) and the ecological suitability (P_i) of each taxon, we compared side-by-side the spatial process P_i with its corresponding mixed process (Q_i, figure 4.2). It is remarkable that all the ecological suitability processes show smoother (less noisy patterns) than their corresponding mixed processes. Additionally, the probability of occurrences in urban areas, specifically the metropolitan area of Mexico City (see largest grey polygon in figure 4.2l) are attenuated in the ecological process. This effect is different for each taxa and is discussed in the next section. Leadtrees, bats and woodpeckers are mostly distributed on the eastern side of the mountain ridge (Sierra Madre Oriental) while oaks and pines overlap and are distributed in higher areas of the mountain ridges.

In the case of the simulated data, the posterior distribution of the estimated parameter converged ($\hat{R} < 1.01$). In all cases, the value used for generating the synthetic data was contained within the credible interval (95%) range. See 4.12 for more information on analysis and results.

Table 4.1 Posterior means, 95% credible intervals and convergence diagnostic $\hat{R}$ for the case-study of biodiversity records in the eastern part of Mexico. Ecological Suitability and Sampling Effort corresponds to the processes P and S defined in the main text. The Contribution to Ecological Suitability row describes the parameter α_i defined in the mixing process Q_i, for each taxon i

				Credible intervals				
			mean	2.5%	50%	97.5%	n.eff	$\hat{R}$
Ecological Suitability (P)	Intercept	Leucaena	-9.06	-15.39	-8.64	-5.23	725	1.0
		Phyllostomidae	-10.13	-16.42	-9.67	-6.39	900	1.01
		Picidae	-30.35	-57.26	-28.07	-15.59	218	1.02
		Quercus	-11.84	-17.67	-11.46	-8.16	1176	1.0
		Pinacea	-18.0	-26.15	-17.48	-12.82	1172	1.0
	Elevation	Leucaena	-1.7e-3	-2.9e-3	-1.6e-3	-9.3e-4	1185	1.0
		Phyllostomidae	-8.0e-4	-1.4e-3	-7.7e-4	-3.6e-4	1675	1.0
		Picidae	4.9e-4	-3.8e-4	4.5e-4	1.6e-3	1068	1.0
		Quercus	1.9e-3	1.3e-3	1.9e-3	2.8e-3	1453	1.0
		Pinacea	3.5e-3	2.5e-3	3.5e-3	5.1e-3	1194	1.0
	Precipitation	Leucaena	-0.01	-0.03	-0.01	-4.0e-3	2770	1.0
		Phyllostomidae	9.8e-3	3.1e-3	9.5e-3	0.02	1487	1.0
		Picidae	0.04	0.02	0.03	0.07	243	1.01
		Quercus	-3.6e-3	-0.01	-3.5e-3	3.8e-3	4234	1.0
		Pinacea	0.03	0.02	0.03	0.04	1375	1.0
	Sampling effort (S)	Intercept	2.56	2.37	2.56	2.76	309	1.02
		Distance to road	-1.5e-4	-1.9e-4	-1.5e-4	-1.1e-4	976	1.01
		Population density	4.3e-4	3.2e-4	4.3e-4	5.5e-4	858	1.01
	Contribution to Ecological suitability (α)	Leucaena	0.44	0.28	0.44	0.62	866	1.0
		Phyllostomidae	0.44	0.29	0.44	0.6	1031	1.01
		Picidae	0.2	0.1	0.2	0.33	331	1.01
		Quercus	0.51	0.36	0.51	0.65	1350	1.0
		Pinacea	0.58	0.42	0.59	0.74	1520	1.0
	Spatial effect	λ_g	0.13	0.05	0.12	0.23	1149	1.0
		τ^2	0.1	0.09	0.1	0.12	113.0	1.03



(a) E.S. leadtrees **(b)** E.S. bats **(c)** E.S. woodpeckers

(d) M.E. leadtrees **(e)** M.E. leafnose bats **(f)** M.E. woodpeckers

(g) E.S. oaks **(h)** E.S. pines **(i)** Sampling effort

(j) M.E. oaks **(k)** M.E. pines **(l)** Study area

Fig. 4.2 Comparison of each ecological suitability (E.S.) processes P_i (a,b,c,h and g) and its corresponding mixing effect (M.E.) Q_i (d,e,f,j and k) in the study area (l). Q_i is the convex combination of P_i with the sampling effort process S (i). All figures show the respective mean posterior on each unit element of the lattice $\mathbb{W}$ in the study area (l).

4.4.9 Cross-validation

The k-fold cross-validation resulted in high predictive accuracy with an AUC of 91.6% (91.14%–91.16%). The resulting ROC curves are shown in figure 4.3.

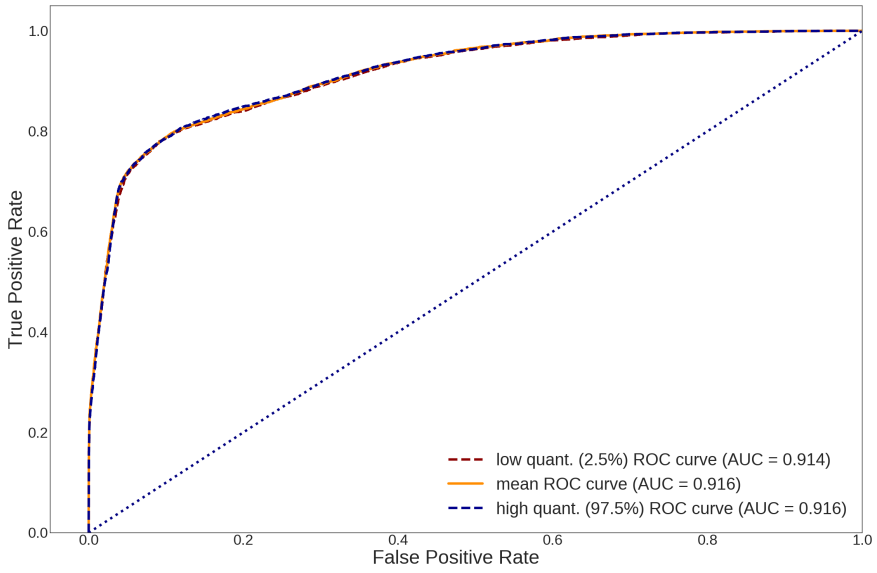


Fig. 4.3 Receiver operator characteristic curve (ROC) obtained from the 10-fold cross-validation of the model applied to the case-study data. The orange solid line represents the ROC curve of the mean posterior prediction while the red and blue dashed lines represent the credible interval at 2.5% and 97.5% resp. The diagonal (identity) line represents the ROC curve of random classification with constant 50% true positive rate.

4.5 Discussion

In this paper, we presented a model for predicting the occurrence of multiple taxa in a spatial unit-area setting (i.e. lattice) using presence-only data and an *intrinsic* complementary sample derived from the taxonomic structure of the selected taxa of interest. The model was tested using simulated data for binary and continuous observations with all parameters contained between the corresponding 95% credible intervals (CIs) of the

posterior distribution. The binary simulation (table: 4.4 in supplementary materials 4.12) showed larger ranges of the CIs than for the Gaussian observations (4.3 sup. mat. 4.12). Additionally, the number of iterations needed for reaching convergence of the Markov chains during the MCMC inference (measured with the $\hat{R}$ diagnostic) was significantly lower for the continuous case (1000 iterations) than the binary case (c.a 40,000 iterations). The analyses suggest that the variance of the model (driven by the parameters τ^2 and λ_g) is a determining factor for the precision of the model (range of the credible intervals) and the greater the values of the parameters, the harder it is to identify accurately the spatial effect. Although the results presented correspond to a single simulation, they indicate the precision that may be expected for real data.

The findings in the case study showed that Leadtrees (Leucaena) have negative preference for elevation and precipitation. This result is consistent with ecological theory as it is a plant that thrives in warm and semi-arid environments (Niembro-Rocas et al., 2010; Rzedowski, 2006). Interestingly, pines and oaks obtained similar estimates for their contribution to their respective ecological suitability (and sampling effort). These two taxa are often considered a single type of vegetation due to their complex network of dependencies and similar ecological niches (Rzedowski, 2006). The fact that both taxa have similar contributions as well as an overlapping geographic space (i.e. middle to high elevated areas of the Sierra Madre Oriental) is indicative of their common ecological niche and suggests that the presence of one taxon is informative of the other.

Leafnose bats have a preference to roost in warm subtropical regions (Ceballos, 2013). Consistent with ecological theory, areas with higher ecological suitability for bats were the subtropical regions between the coast and the foothills of the Sierra Madre Oriental (figure 4.2b). Moreover, constrained to the study area, the highest levels of ecological suitability coincide remarkably with the neotropical realm, suggesting that the model is capable of capturing macroecological patterns.

The fixed effect for the common sampling effort was significant for both covariates (i.e. distance to closest road and population density) supporting the idea that these two covariates are important for characterising the sampling process.

The low contribution to ecological suitability for Leadtrees, bats and woodpeckers, in particular, suggests that the observations of these taxa are mostly biased by the sampling process. This is consistent with the fact that birds and mammals are among the most overrepresented taxa in biodiversity occurrence data (Troudet et al., 2017). Leadtrees, however, have many agricultural and industrial uses such as: shade for coffee plantations, forage for cattle seeds and resin for perfumes and soaps (Niembro-Rocas et al., 2010). Their extensive use explains the strong contribution of the sampling effort, mostly driven by anthropological covariates.

An interesting result was the attenuation of the ecological suitability process in large urban areas, despite the mixed process (Q) showing higher probability. This effect can be seen in all taxa (see figure: 4.2). The model suggests that the signal of scientific interest (i.e. the ecological suitability) can be recovered from observations affected by the mixed joint effect of the ecological and sampling signals. The signal recovery varies depending on the taxon. For example, woodpeckers (fig: 4.2c) was less likely to occur in the metropolitan area of Mexico City (middle western part of the study area), once the signal from the sampling process was removed (see 4.2f for comparison). This result is consistent with the reported preference for woodpeckers and other forest birds to nest in natural and semi-urban areas rather than urban and densely populated areas (Sandström et al., 2006).

There are several potential routes for future research. An immediate limitation of the model is that the spatial random effect (G) is shared between all the taxa of interest and the sampling effort. Despite this reducing the model complexity while capturing interactions between taxa, it is a strong assumption. We think that these interactions could be more ecologically meaningful if the spatial effect is modelled jointly, accounting

for correlated spatial components between the sampling process and one (or multiple) ecological suitability(ies). A way to move forward in this direction is to specify a multivariate CAR (MCAR) model (Gelfand et al., 2003) for the spatial effect (G) to account for joint and marginal spatial effects for each P_i and S.

Another aspect to consider is that the selection of the taxa of interest (see section: 4.4.4) was focused on the available data and the capabilities of the model. As such, we gave priority to widely distributed taxa whilst keeping ecological interest and diversity across different branches of the tree of life. These criteria should not be considered pre-requisites for the model as other taxa with greater scientific interest can be analysed with our model.

We acknowledge the limitations of the algorithm 2 for deriving the complementary sample. Here, we assumed that the known presences of taxonomically related taxa increase the likelihood for presences of the taxa of interest. Although this is a sensible approach to model occurrences with presence-only data (see Croft et al. (2019); Ward et al. (2009) for modelling examples), the assumption is not true in general. A central principle in community ecology is that of "interspecific competition" in which closely phylogenetically related species are more prone to share similar ecological niches resulting in stronger competition between each other (Darwin, 1859; Elton, 1946) and (Hardin, 1960).

The above approach is problematic in two forms: If the taxa of interest are too phylogenetically distant, the complementary sample may not be representative. For example, an ecologist surveying for woodpeckers may have strictly zero interest in any kind of tree and thereby the presence of a woodpecker provides zero information on the presence or absence of any kind of tree. Measuring this effect is difficult due to the high complexity of phylogenetic, phenotypic and environmental relationships (See Aschehoug et al. (2016) for a review and Cahill et al. (2008) for a direct test for measurement). As such, we suggest caution when applying the complementary sample algorithm mindlessly on any group of

taxa without analysing previously possible relations of spatial exclusion due to potential competition.

Increasing the taxonomic resolution to subranks (e.g subfamilies, tribes, subgenera) can be a first step in obtaining more informative complementary samples. However, an improved algorithm aimed to integrate phylogenetic relationships in theoretically consistent, assemblages of ecological communities should account for both spatial and phylogenetic effects, of overdispersion and clustering (see Webb et al. (2002) for a conceptual definition and Pavoine and Bonsall (2011) for a semantic methodology).

4.6 Conclusion

Here, we proposed a model for predicting the occurrence of multiple taxa using presence-only data. The multivariate model separates the observations between two sets: taxa of interest, modelled with an ecological suitability process of scientific interest, and a complementary sample, modelled with a sampling effort process. The complementary sample is used as a proxy for modelling the absences in the presence-only data and it is based on the associated complementary taxonomic tree available in the region under study. Although the model reduces the ecological relationships in a shared spatial random effect, it is still capable of removing the sampling effort signal while recovering the one of scientific focus. In this regard, the application showed preferential areas and macroecological patterns relative to each taxon, showing its potential for future ecological applications.

4.7 Acknowledgments

This project was jointly sponsored by the Mexican Science and Technology Council (CONACyT) under the doctoral program: *Becas al Extranjero* and the Faculty of Science and

Technology from Lancaster University. We thank Bruno Barrales Alcalá and Juan Carlos González Rodríguez for their suggestions of the arboreal taxa used in the case example.

4.8 Data and source code availability

The lattice $\mathbb{W}$ in Esri shapefile format as well as its adjacency matrix representation W is located in main repository of the project (see section 4.9.

The implementation of the code in STAN as well as the simulation procedures and interactive Jupyter (Kluyver et al., 2016) notebooks are located in <Repo URL>

4.9 Data and source code availability

Currently the code and data are stored in the following repository: http://git.holobio.me/juan/paper3code.git. We intend to put the code and data in a long term curated repository such as Dryad or FigShare.

4.10 Authors' contributions

All authors developed the general model and provided critical feedback in all stages of this research. PD proposed the specific statistical model with a shared spatial random effect. JEM proposed the multilevel component. PA proposed the *complementary sample* concept. LS and JEM designed the complementary sample algorithm and the simulated dataset. JEM developed the simulations, implemented the models, performed the analysis, created the figures and wrote the manuscript with inputs and edits from all co-authors. PA, LS and PD supervised the project.

4.11 Appendix: Model description

Let $\mathbf{y}_{x_k} = \{y^1_{x_k}, y^2_{x_k}, ..., y^n_{x_k}\}$ be the presence observations for n different taxa (e.g. species) at location x_k in the spatial lattice $\mathbb{W}$. The model is suited for aggregated data defined on an areal lattice system $\mathbb{W}$ in the sense of Besag (1974). That is, on each cell $x_k \in \mathbb{W}$, an observation for species i (i.e. $y^i_{x_k}$) is the realisation of a binary-valued random variable $Y_i(x_k)$, Bernoulli distributed and independent when conditioned to a mixing effect Q_i (not introduced yet). The conditional distribution of this variable is:

$$[Y_i(x_k) = y^i_{x_k} | Q_i(x_k)] \sim \text{Bernoulli}(Q_i(x_k)) \tag{4.5}$$

$y^i_{x_k} = 1$ represents the event of: "taxon i has been observed in site x_k", while, $y^i_{x_k} = 0$ represents the event of: "taxon i has not been observed in site x_k". Given that the model uses presence-only data, unobserved taxa (i.e. $y^i_{x_k} = 0$) imply two possibilities. Either the site x_k is not suitable for taxon i to be present, or taxon i has not been sampled in x_k. The role of the process Q_i is to account for these two possibilities through a mixture of two processes; one that conditions taxon i to live in location x_k, (hereafter called *ecological suitability*), and other, that represents a preferential sampling given surveying information derived from the taxa of interest. This last process is described here as the *sampling effort* process S. This process reduces the uncertainty for places with no available information related to the taxa of interest.

Mixing process Q_i

This process describes the quantifiable contributions of the *ecological suitability* (P_i) and the *sampling effort* (S). Mathematically, Q_i is defined as the mixing process between the P_i and S, for each taxon i. That is, Q_i is the convex combination between P_i and S and

has the form (eq: 4.6):

$$[Q_i(x_k)|P_i(x_k), S(x_k), \alpha_i] = \alpha_i P_i(x_k) + (1-\alpha_i)S(x_k) \tag{4.6}$$

where $0 \leq \alpha_i \leq 1$ and it is called the *contribution to ecological suitability* parameter ($i \in \{i, ..., n\}$) and $x_k \in \mathbb{W}$.

Ecological Suitability process P_i

The ecological suitability process P_i explains the presence of the taxon i independently from the sampling effort (see conditional dependencies in figure 4.1). P_i can be often considered as the process of scientific focus, as it accounts for the environmental pressure that determines the existence (or establishment) of a taxon (species) i in a given location. We decided to characterise each P_i as a process with two components: a structural *fixed effect* component that models the effect of environmental covariates (scenopoetic variables); and a common *spatial random effect* G that depends on the spatial correlation on the areal data. Recalling that the occurrences are measured as binary outcomes, the latent processes should represent valid probability values (i.e. real values on the unit interval). As such, the distribution of each P_i conditional to G (the common spatial random effect) has a logistic form (see eq. 4.7).

$$\text{logit}\left([P_i(x_k)|G(x_k), \beta_i; d_{e_i}]\right) = \beta_{P_i}^t d_{e_i}(x_k) + G(x_k) \tag{4.7}$$

where $\beta_{P_i} \in \mathbb{R}^k$ is a vector of linear coefficients for the fixed effect and $d_{e_i}(x_k) \in \mathbb{R}^k$ the k-dimensional vector of covariates corresponding to location x_k. In general, any pair of taxa i, j may have different covariates and, thus, $d_{e_i}(x_k)$ and $d_{e_j}(x_k)$ are not necessarily the same.

Preferential Sampling Effort S

The sampling effort is modelled similarly to the *ecological suitability* process and mathematically both types of process are equivalent. From a conceptual perspective these models are rather different. While the likelihood of each P_i depends directly on the presences of a specific taxon i, the likelihood of S relies on aggregated observations from the complementary taxa. A full description on how to obtain *intrinsic* observations from these taxa is explained in section 4.2.2, whereas equation 4.8 is a decomposition into fixed and random effects.

$$\text{logit}\left([S(x_k)|G(x_k),\beta_s;d_a]\right) = \beta_s^t d_a(x_k) + G(x_k) \tag{4.8}$$

where $\beta_s \in \mathbb{R}^k$ is a vector of linear coefficients for the fixed effect of the sample and $d_a(x_k) \in \mathbb{R}^k$ a k-dimensional vector of anthropological covariates corresponding to location x_k. In general, $d_a(x_k)$ and $d_{e_j}(x_k)$ are likely to be different covariates.

Common spatial random effect G

To model the spatial interactions between the areal elements of the lattice ($\mathbb{W}$), we propose a stationary conditional autoregressive (SCAR) model (Gelfand and Vounatsou, 2003) and (Rue and Held, 2005) common to each P_i and S. In this sense, $P_i(x_k)$ and $S(x_k)$ have the same spatial effect for a given location of the lattice (i.e. $x_k \in \mathbb{W}$). The SCAR model is a generalisation of the intrinsic CAR (ICAR) model (Besag, 1974; Besag et al., 1991) that enhances the scientific interpretability with an additional parameter λ_G that accounts, for the proportional strength of the spatial dependence in relation to non-informative or unstructured variability. In contrast to the ICAR model, the stationary CAR is specified with a proper probability distribution and, therefore, no further constraints on its values are needed, making the model fully identifiable.

Here, we specify G in its full conditional form. Let $(G_{x_1},...G_{x_m})^T$ be a vector representation of G across the lattice $\mathbb{W}$. That is, each areal element $x_k \in \mathbb{W}$ is mapped one-to-one with an element G_{x_k}. The conditional distribution of G_{x_k} given the rest of the areal elements (G_{-x_k}), an unknown variance ($\tau^2_{x_k}$) and a spatial dependence parameter (λ_G), is defined as a normal distribution of the form:

$$[G_{x_k}|G_{-x_k},\lambda_G,\tau^2_{x_k};\mathbb{W}] \sim N\left(\lambda_G \sum_{j=1}^{m} b_{k,j}G_{x_k},\tau^{-2}_{x_k}\right) \tag{4.9}$$

Where: G_{x_k} is equivalent to $G(x_k)$, G_{-x_k} is a notational term for defining the rest of areal elements, that it, $\{G(x'),x' \in \mathbb{W}|x' \neq x_k\}$, λ_G is the spatial dependence parameter and $b_{k,j}$ is a weight related to each areal element. The spatial dependency parameter λ_G varies from $[0,1]$. If $\lambda_G = 0$, G has no spatial autocorrelation while if $\lambda_G = 1$, the random effect is fully described by an intrinsic CAR model (Besag, 1974; Besag et al., 1991).

To simplify the model (eq. 4.9) we assumed that $\tau^2_{x_k}$ and $b_{k,j}$ are quantities normalised by the number of neighbours of each unit area $x_k \in \mathbb{W}$ That is: $\tau^2_{x_k} = \frac{\tau^2}{\sum_{j=1}^{n} w_{k,j}}$ and $\sum_{j=1}^{m} b_{k,j} = \frac{\sum_{j=1}^{m} w_{k,j}G_j}{\sum_{j=1}^{m} w_{k,j}}$. The term $w_{k,j}$ is the (k,j)-entry of the adjacency matrix of the lattice $\mathbb{W}$, hereafter referred to W. Using Brook's lemma (Besag, 1974; Brook, 1964), the full conditional specification in eq. (4.9) is equivalent to a zero-centred multivariate normal distribution (MVN) of the form:

$$[G,\tau^2,\lambda_G;W] \sim \text{MVN}\left(0,\tau^2(D-\lambda_G W)^{-1}\right) \tag{4.10}$$

Where D is a mxm diagonal matrix built with the reciprocal of the number of neighbours for each areal unit (i.e. $D_k = (\sum_{j=1}^{m} w_{k,j})^{-1}$).

4.11.1 Support for missing data

The model allows inference on locations where information about the presence or absence of a given taxa (or sampling effort) is unknown. This approach treats missing observations as additional parameters to the model. That is, let $\hat{Y}_i(x_k)$ be a Bernoulli random variable corresponding to the missing observation of taxa i in location x_k, an areal element of the spatial lattice $\mathbb{W}$. The posterior distribution of $[\hat{Y}_i(x_k)|Q_i(x_k), Y_i(x_k) = y^i_{x_k}]$ is obtained by marginalisation. That is:

$$[\hat{Y}_i(x_k)|Q_i(x_k)] \quad = \quad [Y_i(x_k) = 1|Q_i(x_k)][Q_i(x_k)] + [Y_i(x_k) = 0|Q_i(x_k)][Q_i(x_k)] \quad (4.11)$$

4.12 Appendix: Simulation study

To validate the model implementation we generated a synthetic dataset following the specification of the model. That is, we generated a random realisation of the processes Q_i, P_i, S and G as well as two types of observations: Continuous $\hat{Y}_i$ (Gaussian) outcomes, to validate the model at the last hierarchical latent surface, and binary Y_i (i.e. presence-absence) outcomes, to validate at the last level of observations. The observations were sampled according to the distributions:

$$\hat{Y}_i \sim \text{Normal}(Q_i(x_k), \sigma_q^2) \tag{4.12}$$

$$Y_i \sim \text{Bernoulli}(Q_i(x_k)) \tag{4.13}$$

The simulated dataset consisted of five synthetic taxa and one sampling process. First we simulated G, with a proper CAR model with parameters: $\lambda_G = 0.7$ and $\tau^2 = 2.0$. The realization was obtained by sampling a MVN distribution, zero centered and with covariance matrix given by equation: 4.10. The resulting realisation of this process is pictured in figure 4.4a.

The ecological suitability processes were simulated by sampling $n = 4061 \times 5 \times 2$ independent standard normal values x_i; where 4061 are the number of regions in the lattice $\mathbb{W}$, 5 the number of different taxa (levels) and 2 the number of covariates. In a similar way, the sampling effort process was sampled from 4061×2 independent standard normal values. A design matrix of dimensions 4061×2 was arranged for each level (i.e. ecological suitability plus sampling effort). Each design matrix was multiplied by an arbitrarily chosen vector of coefficients to obtain the *fixed effects* for P_i and S. The fixed effect of each level was summed to the spatial random effect G to account for the total variation (as defined in eqs: 4.7 and 4.8). The values used as coefficients for simulating the fixed effects are shown in table 4.2.

Table 4.2 Chosen values used for simulating Q_i, P_i and S, given a matrix of covariates (sampled from a normal distribution) and a random effect G defined as a proper Gaussian Markov random field.

Taxon	1	2	3	4	5	Sample
Covariate (β_1)	1.0	2.0	3.0	4.0	5.0	6.0
Covariate (β_2)	7.0	8.0	9.0	10.0	11.0	12.0
Mixture (α)	0.35	0.5	0.123	0.75	1.0	–

Having obtained all the values for P_i and S we proceeded to simulate the mixing processes Q_i using equation 4.6. To do this, we chose different values for α_i, one for each Q_i and computed the corresponding convex combination. The values (α) used in this stage are shown in table 4.2.

Finally, we proceeded to sample observations Y_i according to equations 4.12 and 4.13 (with $\sigma_q^2 = 00.1$ for the continuous case). We performed the inference on both types of observations: binary and continuous to assess the model more generally.

4.12.1 Model fitting

For fitting the model we used the no-U-turn sampler (NUTS) (Hoffman and Gelman, 2014), an adaptive variant of the Hamiltonian Markov chain Monte Carlo (HMCMC) method. The NUTS algorithm was used with default parameters of step size and tree depth. The posterior sample was run for 1000 iterations with a burn-in size of 500 for $\hat{Y}_i$ and 15000 iterations with 7500 of burn-in for the binary observation Y_i. In both examples we did not use any thinning. As we are working in a Bayesian setting, we defined prior distributions for each parameter $\beta_{i\in\{1,\dots,n\}} \sim N(0, 10000)$. The parameters α_i of the mixing process Q_i and λ_G of the proper CAR model were defined with non informative prior distribution of beta$(1,1)$ (equivalent to Uniform$(0,1)$). + The variance parameters σ_q^2 and τ^2 were sampled from an inverse-gamma$(1, 0.01)$. The inverse-gamma family is conjugate with the conditional posterior distribution and it is frequently used to model variance parameters, making the calculation of the posterior distribution easier.

4.12.2 Results

The parameters used in the simulation were estimated accurately on both simulations (i.e. Y_i and $\hat{Y}$), as all posterior means were almost identical to the real parameters and located within the 95% credible intervals (CI). The summary statistics of the corresponding marginal posterior distributions are shown in table 4.3 for the case of continuous observations ($\hat{Y}_i$) and table 4.4 for the binary ones Y_i.

The inference in the Gaussian responses for the single simulation of $\hat{Y}_i$ gave accurate estimations in all parameters. In this case, the parameters used to generate the simulated data were contained in the corresponding 95% credible intervals (CI)s of the posterior sample. The range of theses CIs was short, suggesting good fit, despite the relatively small number of iterations (1000). The parameters of the fixed effects gave a maximum range of 0.05 for all processes, that is, ecological suitability β_{P_i} and sampling effort β_S. The posterior distribution of τ^2 produced a small error range of 0.19, while the rest of the parameters produced a CI less than 0.009. See table 4.3 for a complete list of parameters and their corresponding summary statistics. A comparison between the simulated random effect G and the mean surface of the posterior distribution of G is shown in figure 4.4. The errors are close and centered in zero showing that the model is able to recreate the landscape with a high level of accuracy. For the case of the Bernoulli simulation, we found longer ranges of the CIs. Despite this, all the simulated parameters fell within the 95% CIs of their corresponding posterior marginal distribution. The fixed effects estimates obtained maximum and minimum CI ranges of 5.75 and 1.02, with an average size of 2.31 and a standard deviation of 1.41. The mixing proportions (α_i) between the ecological suitability P_i and the preferential sampling S, obtained 95% CI of (0.15, 0.12, 0.16, 0.06 and 0.02) for each $(P_1, ..., P_5)$, respectively. The ranges of the CI for the parameters of the spatial random effect were: 3.43 for τ^2 and 0.54 for λ_G. The complete list of parameters with their corresponding summary statistics is described in table 4.4. Computing the

posterior sample took approximately 25 minutes for the Gaussian observations $\hat{Y}_i$ and 5 hours 30 minutes for the Bernoulli (binary) observations Y_i using a 4-core Intel(R) Xeon(R) CPU E5-2690 v2 at 3.00GHz and sampling two chains simultaneously.

Table 4.3 [Continuous observations] Comparison between simulated and inferred parameters sampled from the posterior joint probability distribution (see equations 4.1 for normal (continuous) observations ($\hat{Y}_i$). The inference was obtained by MCMC following 1000 iterations with a burn-in of 500. The β parameters correspond to the *fixed effects* of the species P_i for covariates 1 and 2. The parameters α correspond to the mixture between the P_i (probability of occurrence of species i and the sampling effort S). The parameters related to the variance are τ^2 for the spatial random effect and σ_q^2 for the unstructured random effect. All simulated parameters are within the 95% credible intervals

	Simulation	Summary of posterior sample							
		mean	2.5%	25%	50%	75%	97.5%	n.eff	$\hat{R}$
$\beta_{P_1,1}$	1.0	1.0	0.99	0.99	1.0	1.0	1.01	1300.0	1.0
$\beta_{P_2,1}$	2.0	2.01	2.0	2.01	2.01	2.01	2.01	691.0	1.0
$\beta_{P_3,1}$	3.0	3.03	3.0	3.02	3.03	3.04	3.05	1279.0	1.0
$\beta_{P_4,1}$	4.0	4.0	3.99	4.0	4.0	4.0	4.0	124.0	1.02
$\beta_{P_5,1}$	5.0	5.01	5.0	5.0	5.01	5.01	5.01	59.0	1.03
$\beta_{S,1}$	6.0	6.01	6.0	6.01	6.01	6.01	6.01	50.0	1.04
$\beta_{P_1,2}$	7.0	7.01	7.0	7.0	7.01	7.01	7.02	221.0	1.0
$\beta_{P_2,2}$	8.0	8.01	8.0	8.01	8.01	8.01	8.02	66.0	1.03
$\beta_{P_3,2}$	9.0	9.02	8.98	9.01	9.02	9.03	9.06	1155.0	1.0
$\beta_{P_4,2}$	10.0	10.01	10.0	10.01	10.01	10.01	10.02	60.0	1.04
$\beta_{P_5,2}$	11.0	11.02	11.01	11.02	11.02	11.02	11.03	44.0	1.04
$\beta_{S,2}$	12.0	12.01	12.01	12.01	12.01	12.02	12.02	39.0	1.06
α_{P_1}	0.35	0.35	0.35	0.35	0.35	0.35	0.35	87.0	1.02
α_{P_2}	0.5	0.5	0.5	0.5	0.5	0.5	0.5	45.0	1.04
α_{P_3}	0.123	0.12	0.12	0.12	0.12	0.12	0.12	429.0	1.0
α_{P_4}	0.75	0.75	0.75	0.75	0.75	0.75	0.75	43.0	1.05
α_{P_5}	1.0	1.0	1.0	1.0	1.0	1.0	1.0	34.0	1.06
τ^2	2.0	1.96	1.86	1.93	1.96	1.99	2.05	1644.0	1.0
λ_G	0.7	0.72	0.67	0.7	0.72	0.73	0.77	1699.0	1.0
σ_q^2	0.1	0.1	0.1	0.1	0.1	0.1	0.1	1313.0	1.0

The difference on precision between the continuous and binary cases can be explained, in by the difference in their signal-to-noise ratio. The results on the simulation suggest a higher ratio for the continuous case compared with the binary one. While the variance in $\hat{Y}$ is constant (i.e. $\text{VAR}[\hat{Y}_i(x_k)|Q_i(x_k)] = \sigma_q^2 = 0.1$), the variance of the binary case depends

Table 4.4 [Binary observations] Comparison between simulated and inferred parameters sampled from the posterior joint probability distribution (see equation 4.1 for binary observations (presence / absence) distributed as independent Bernoulli variables when conditioned to the latent random effect G. The inference was obtained from MCMC following 40000 iterations with a burn-in of 20000. The β parameters correspond to the *fixed effects* of the species P_i for covariates 1 and 2. The parameters α correspond to the mixture between the P_i (probability of occurrence of specie i and the sampling effort S). The parameters related to the variance are τ^2 for the spatial random effect.

	Simulation	Summary of posterior sample							
		mean	2.5%	25%	50%	75%	97.5%	n.eff	$\hat{R}$
$\beta_{P_1,1}$	1.0	0.62	0.08	0.43	0.62	0.81	1.2	5512.0	1.0
$\beta_{P_2,1}$	2.0	2.38	1.92	2.22	2.37	2.54	2.85	4592.0	1.0
$\beta_{P_3,1}$	3.0	2.71	1.52	2.25	2.68	3.13	4.13	5030.0	1.0
$\beta_{P_4,1}$	4.0	4.19	3.77	4.04	4.18	4.33	4.63	3950.0	1.0
$\beta_{P_5,1}$	5.0	5.17	4.49	4.92	5.16	5.41	5.9	4157.0	1.0
$\beta_{S,1}$	6.0	5.99	5.54	5.83	5.99	6.14	6.45	3685.0	1.0
$\beta_{P_1,2}$	7.0	6.65	5.68	6.3	6.63	6.97	7.69	4248.0	1.0
$\beta_{P_2,2}$	8.0	8.38	7.52	8.07	8.37	8.68	9.3	3638.0	1.0
$\beta_{P_3,2}$	9.0	9.66	7.43	8.71	9.53	10.47	12.73	5020.0	1.0
$\beta_{P_4,2}$	10.0	10.21	9.44	9.93	10.2	10.48	11.05	3007.0	1.0
$\beta_{P_5,2}$	11.0	10.95	9.75	10.51	10.92	11.38	12.28	3898.0	1.0
$\beta_{S,2}$	12.0	12.36	11.56	12.07	12.35	12.64	13.2	3165.0	1.0
α_{P_1}	0.35	0.34	0.31	0.33	0.34	0.35	0.38	5014.0	1.0
α_{P_2}	0.5	0.48	0.45	0.47	0.48	0.49	0.51	4302.0	1.0
α_{P_3}	0.2	0.19	0.15	0.17	0.19	0.2	0.23	5106.0	1.0
α_{P_4}	0.75	0.76	0.74	0.75	0.76	0.77	0.78	4821.0	1.0
α_{P_5}	0.45	0.45	0.42	0.44	0.45	0.46	0.48	4720.0	1.0
λ_G	0.7	0.71	0.63	0.68	0.71	0.71	0.77	4912.0	1.0
τ^2	0.01	0.0097	0.0084	0.0092	0.0096	0.01	0.01	2537	1.0

on each value of $Q_i(x_k)$. As $[Y_i(x_k)|Q_i(x_k)] \sim$ Bernoulli$(q_i(x_k))$, its variance ranges from 0 to 0.25, reaching its maximum at $Q_i(x_k) = 0.5$ when $P_i(x_k) = 0$ and $S(x_k) = 0$. In other words, for the binary case, the variance associated with the error (noise) depends on the each value of the latent variable (i.e $Q_i(x_k)$, for each $x_k \in \mathbb{W}$), while in the continuous case, this variance is constant. A possible explanation is that the relatively large variance of the spatial parameters τ^2 and λ_g made it difficult to identify accurately the effect of the spatial effect.

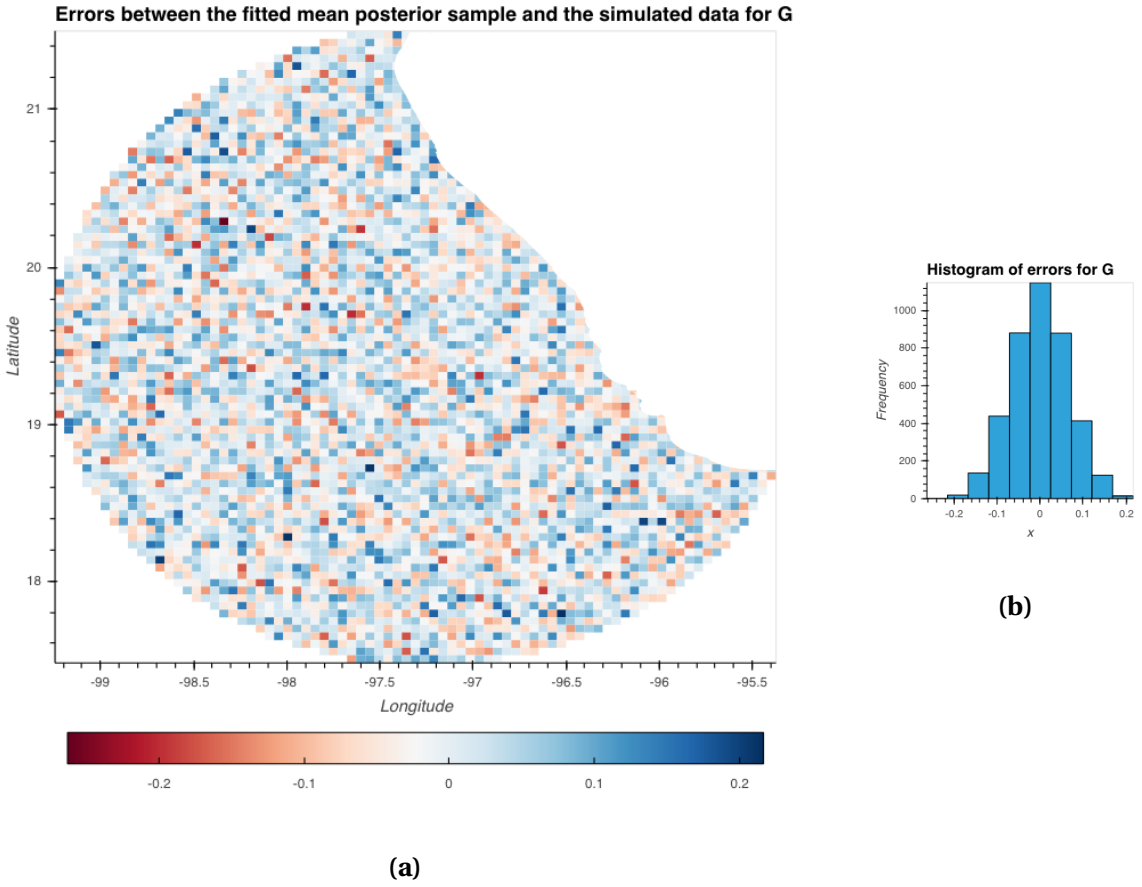


Fig. 4.4 Errors of the spatial random effect G calculated as the difference between the mean posterior sample and the simulated data. Left panel (a) shows the spatial arrangement while right panel shows the histogram of the errors. The posterior mean was obtained through 1000 iterations fitted with Gaussian observations $\hat{Y}_i$.

4.13 Appendix: Implementation of the model in STAN

Listing 4.1 The multispecies model implementation in STAN

```
/**
 The Multispecies model with common random effect

 This is an implementation of the Multispecies model with mixing components for
 ecological suitability and sampling effort using a common proper CAR model as
 spatial autocorrelation.

 This implementation assumes that the observations [Y_i | Q_i ] are normal
 distributed with variance $\sigma_q$. This was done to fit simulated
 data described in the section: methods.

 To fit presence-absence observation change the likelihood [Y_i | Q_i ]
 accordingly, e.g. a bernoulli distribution.

 Author: Juan Escamilla Molgora
 Date: 05/10/2020

 Note: The 'sparse_car_lpdf' function and the transformation of the adjacency
 matrix into a sparse representation were adapted from Max Joseph's tutorial on
 CAR model in STAN: https://mc-stan.org/users/documentation/case-studies/mbjoseph-
    CARStan.html
*/
    functions {
      /**
      * Return the log probability of a proper conditional autoregressive (CAR)
    prior
      * with a sparse representation for the adjacency matrix
      *
      * @param phi Vector containing the parameters with a CAR prior
      * @param tau Precision parameter for the CAR prior (real)
      * @param alpha Dependence (usually spatial) parameter for the CAR prior (
    real)
      * @param W_sparse Sparse representation of adjacency matrix (int array)
      * @param n Length of phi (int)
      * @param W_n Number of adjacent pairs (int)
      * @param D_sparse Number of neighbors for each location (vector)
      * @param lambda Eigenvalues of D^{-1/2}*W*D^{-1/2} (vector)
      *
      * @return Log probability density of CAR prior up to additive constant
      */
      real sparse_car_lpdf(vector phi, real tau, real alpha,
        int[,] W_sparse, vector D_sparse, vector lambda, int n, int W_n) {
          row_vector[n] phit_D; // phi' * D
          row_vector[n] phit_W; // phi' * W
          vector[n] ldet_terms;
```

```
      phit_D = (phi .* D_sparse)'; //';
      phit_W = rep_row_vector(0, n);
      for (i in 1:W_n) {
        phit_W[W_sparse[i, 1]] = phit_W[W_sparse[i, 1]] + phi[W_sparse[i, 2]];
        phit_W[W_sparse[i, 2]] = phit_W[W_sparse[i, 2]] + phi[W_sparse[i, 1]];
      }

      for (i in 1:n) ldet_terms[i] = log1m(alpha * lambda[i]);
      return 0.5 * (n * log(tau) + sum(ldet_terms) - tau * (phit_D * phi -
alpha * (phit_W * phi)));
  }
}

data {
  int<lower=0> N;        // num obs.
  int<lower=0> J;       // number of levels
  int<lower=0> N_ecological_covariates; // number of covariates for the eco.
suit process.
  int<lower=0> N_sample_covariates; // number of covariates for the sample
effort.
  row_vector[N_ecological_covariates + N_sample_covariates] x[N];// Size of
design matrix
  int<lower=1,upper=J> level[N];  // type of level (spec)
  int<lower=0,upper=2> y[N];              // observations, in this case is
binary.
// data for the spatial structure
  int<lower=0> N_areas; // number of areas in the region.
  int<lower=0> N_edges;  // Number of pairs
  matrix<lower = 0, upper = 1>[N_areas, N_areas] W; // adjacency matrix of
lattice

  int<lower=0> N_miss; // Number of missing information
  //int<lower=0> Y_miss_array[N_miss]; // array of indexed missing
observations

}

transformed data {
 // rename variables for better usage
   int L = N_ecological_covariates;
   int M = N_sample_covariates;
   int K = L + M;
// Sparse representation of W
  int<lower=0> W_n = N_edges; // just to make it compliant with the rest of
the models
  int W_sparse[W_n, 2];   // adjacency pairs
  vector[N_areas] D_sparse;     // diagonal of D (number of neigbors for each
site)
  vector[N_areas] lambda;       // eigenvalues of invsqrtD * W * invsqrtD

  { // generate sparse representation for W
  int counter;
```

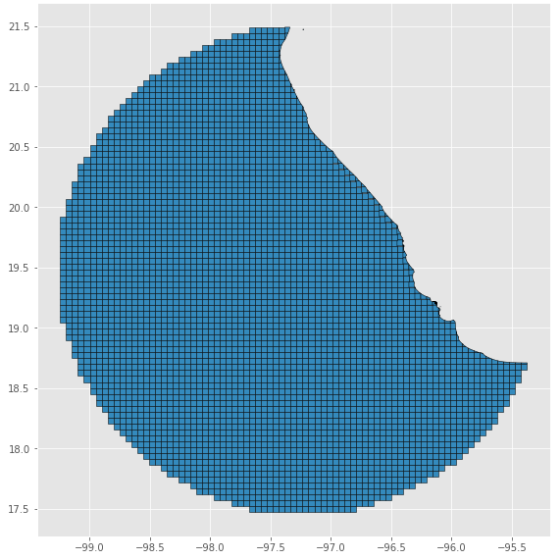
```
  counter = 1;
  // loop over upper triangular part of W to identify neighbor pairs
    for (i in 1:(N_areas - 1)) {
      for (j in (i + 1):N_areas) {
        if (W[i, j] == 1) {
          W_sparse[counter, 1] = i;
          W_sparse[counter, 2] = j;
          counter = counter + 1;
        }
      }
    }
  }
  for (i in 1:N_areas) D_sparse[i] = sum(W[i]);
  {
    vector[N_areas] invsqrtD;
    for (i in 1:N_areas) {
      invsqrtD[i] = 1 / sqrt(D_sparse[i]);
    }
    lambda = eigenvalues_sym(quad_form(W, diag_matrix(invsqrtD)));
  }


}


parameters {
  // Multilevel fixed effect

  // The splitted betas for the ecological processes
  vector[L + M] beta_eco[J];

  // Spatial effect
  vector[N_areas] G;          // spatial effects
  real<lower = 0> tau;
  real<lower = 0, upper =1> alpha_car;

  // Mixing effect for Q
  // The alpha parameter, one per level.
  simplex[2] alpha_1[J - 1];


}


transformed parameters {
  simplex[2] alpha[J];
  vector[K] beta[J]; // Each level has an assigend beta of K dimension.



  // Define the last level (sampling effort) with no mixing effect
    for (j in 1:J - 1){
```

```
            alpha[j] = alpha_1[j];
        }
        alpha[J][1] = 0.0;
        alpha[J][2] = 1.0;

      for (j in 1:J - 1){ // Do this for the multispecies level
        for (i in 1:L){
            beta[j][i] = beta_eco[j][i];
      }
        for(i in L + 1: L + M){
                beta[j][i] = 0.0;
      }
      }
      // Assign values to covariates of the sample.
        for (i in 1:K){
             if (M == 0 && i <= L){
                beta[J][i] = beta_eco[J][ i ];
            }

            else if ( i > L ){
                //beta[J][i] = beta_samp[ i - L ];
                beta[J][i] = beta_eco[J][ i ];
                }

        // if number of covariates for sample effort is 0 then assume both process
     have
        // the same covariates
            else {
               beta[J][i] = 0.0;
            }
      }



      }

    model {
        // def variable
      vector[N] S;
      vector[N] P;
      vector[N] Q;

        // Priors for multilevel fixed effects

        // Priors for the stationary CAR
            tau ~ inv_gamma(1, 0.1);
            // a very informative one
            alpha_car ~ beta(1,1);

        // Spatial prior
                G ~ sparse_car(tau,alpha_car,W_sparse, D_sparse, lambda, N_areas,
    W_n);
```

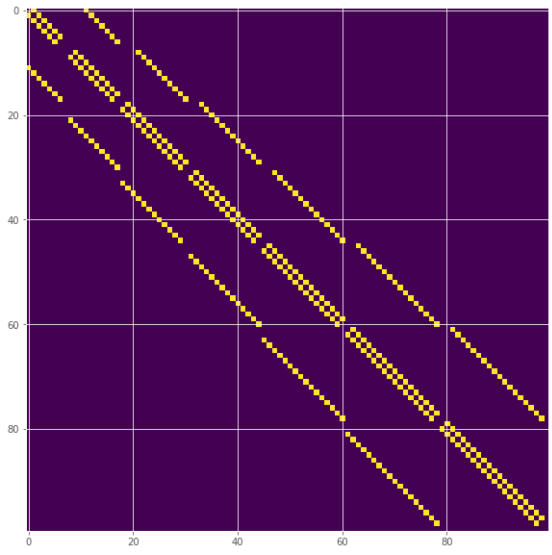
```
    // Model for priors in the mixing Qs
    // For the betas in the multilevel
    for (j in 1:J - 1){
            // betas for ecological process
            beta_eco[j] ~ normal(0,10000);
            alpha_1[j] ~ beta(5,5);
    }
    // parameters for the sample (J is the last number of the level)
    beta_eco[J] ~ normal(0,10000);

    // The Qs
    for (i in 0:N - 1){  //starts with 0 because we are using modulus
        // P and S with spatial random effect.
        P[i + 1] = x[i + 1] * beta[level[i + 1]] + G[ (i % N_areas)+1 ]; //
This because the N_areas is half N, this assures common component
        S[i + 1] = x[(i % N_areas + 1) + (N - N_areas)] * beta[J] + (G[ (i %
N_areas)+1 ]);
        Q[i +1] = (alpha[level[i + 1],1] *  P[i + 1]) + (alpha[level[i + 1],2]
 * S[i + 1]);
        }


  // The parsing element to support missing data. (prototype: missing-data =
2)
  for (i in 1:N){
      if (y[i] > 1) {

      // The marginal of y
      target += log_mix(inv_logit(Q[i]),bernoulli_logit_lpmf(1 | Q[i]),
                                        bernoulli_logit_lpmf(0 | Q[i]));
      }
      else {
      target += bernoulli_logit_lpmf(y[i] | Q[i]);
       //Y[i] = y[i];
      }
  }


    // y ~ bernoulli_logit(Q);
    // target += normal_lpdf(y | Q, sigma_q);

}


generated quantities {
 vector[N_areas] S;
 vector[N] P;
 vector[N] Q;
 int y_imp[N_miss];
 int k = 1;
```

```
    for (i in 0:N_areas - 1 ){
            S[i + 1] = x[(N - N_areas) + (i + 1) ] * beta[J] + (G[ i + 1 ]);
    }

    // The Qs
        for (i in 0:N - 1){
            P[i + 1] = x[i + 1] * beta[level[i + 1]] + G[ (i % N_areas)+1 ];
            Q[i +1] = (alpha[level[i + 1],1] *  P[i + 1]) + (alpha[level[i + 1],2]
     * S[(i % N_areas)+1]);
     }

    // The missing values
        for (i in 0:N - 1){
         if (y[ i + 1 ] > 1) {
             y_imp[k] = bernoulli_logit_rng(Q[i + 1]);
             k += 1;
         }
     }
    }
```

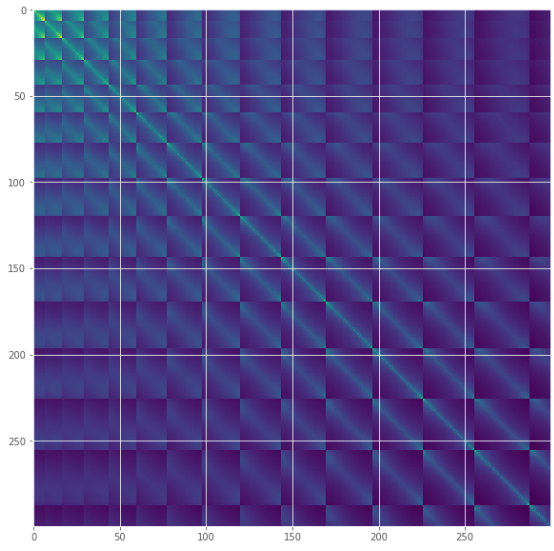
4.14 Appendix: Visualisations of simulated data and associated spatial structures



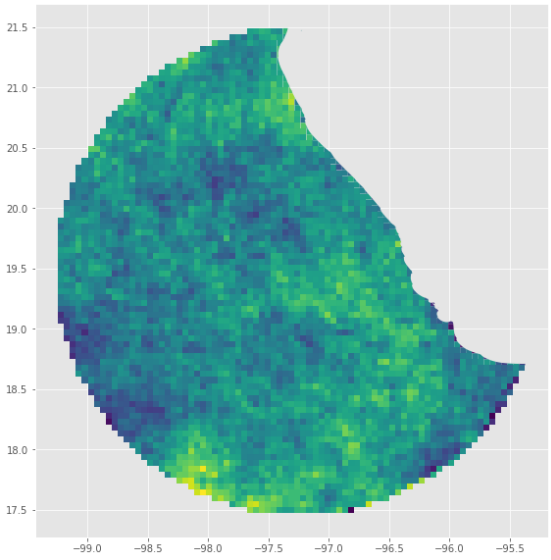
(a) Lattice $\mathbb{W}$



(b) Adjancency matrix W



(c) Covariance matrix Q^{-1}



(d) Simulated ϕ

Fig. 4.5 Adjancency matrix W of the latttice $\mathbb{W}$. The precision matrix ($Q = (D - \alpha W)$) and its corresponding Covariance matrix Q^{-1} (right) corresponding to a simulated gaussian markov random field (GMRF). The figures are restricted to the upper section of the matrices covering the first 100 entries.Simulation, sampled from a multivariate normal with mean 0 and covariance matrix Q^{-1}

CHAPTER 5

GENERAL DISCUSSIONS AND RECOMMENDATIONS FOR THE FUTURE

The research presented in this thesis responded to the problem of inferring the probability of biological taxa (e.g. species) to be present in geographical space using a large corpus of environmental and biodiversity data. These data are *presence-only* observations and, therefore, the information related to real absences is missing. The aim of the research was the development of novel methodologies for modelling species distributions using all available sources of information. Its achievement involved the development of two lines of research. The first was the design and implementation of a knowledge-engine to store, integrate and synthesise large volumes of heterogeneous biodiversity, environmental and geospatial data. The second line of research developed statistical methods to design comprehensive model-based specifications of the stochastic processes that have generated the observed biodiversity occurrences; as opposed to using machine-learning-based *black boxes* that are, generally, complicated to interpret scientifically (Rudin, 2019).

To answer the first central question in the thesis (i.e *How to formalise a comprehensive data structure to unify and synthesise heterogeneous data sources?*) I developed knowledge-based computational methods for integrating heterogeneous environmental and biodiversity data. There are several more advanced proposals for integrating environmental

data. Examples of these are Australia living Atlas (https://www.ala.org.au/), Ecocommons virtual labs (La Salle et al., 2016) and especially, the Earth Observations Biodiversity Observation Network (GEOBON) platform for data homogenization and standarisation of essential biodiversity variables (EVB)s (Navarro et al., 2017; Pereira et al., 2013; Scholes et al., 2012). We contributed to this international effort presenting an open source engine capable of storing information in a knowledge graph and abstracting complex ecological concepts as *knowledge schemes* for traversing the knowledge engine. We applied these concepts to assemble taxonomic trees distributed in space. The taxonomic tree includes information on the ecological communities linked with environmental spatio-temporal information. The object-oriented approach allowed the efficient abstraction of taxonomic trees into complex data structures with algebraic operational support.

The idea of merging multiple taxonomic structures and phylogenies into a global tree of life is not new. A successful example is the project *Open Tree of Life* (Hinchliff et al., 2015) that synthesized published phylogenies, together with taxonomic classifications to reconstruct a comprehensive global tree of life. Thus, the work is not an attempt to compete with these projects but to show the potential to incorporate similar structures to better inform species distribution models. In this respect, an innovative feature of the engine is that instances of taxonomic trees can be algebraically operated, bringing the possibility of concatenating intersections, differences or unions between other tree instances. These operations make it possible to express complex data combinations for selecting taxa and merging collections of taxonomic trees to study variation in scale and time. A direct application is the aggregation of trees into biodiversity measures to analyse changes in α, β and γ diversity (see Magurran (2004) and a full mathematical explanation in appendix A.1).

The implemented spatial lattice is another aspect that deserves attention. It is composed of nodes of the type *cell* linked with other cell nodes by the relationship `IS_NEIGH-`

BOUR_OF. Each cell node has a geometric attribute that defines an area bounded by a polygon. The complete set of nodes, conditioned to the same scale, tessellates the surface as a regular grid (see chapter 2 figure 2.3). The tessellation and the neighbouring relations induce a spatial lattice (2.15.1) that is used to define the spatial autocorrelation structure (Chapters 3 and 4). That is, local taxonomic trees (i.e constrained by its linked cell node) together with either: the *choosing principle* (Chp. 3 sec.3.2.2) or the *complementary sample* (Chp. 3 sec. 4.2.2), specifies contextual background data by aggregating the associated occurrences into binary responses [1].

The second part of the thesis was the integration of the knowledge-engine with statistical frameworks to answer the question of, "*How to integrate biodiversity data from different sources to infer species geographic distributions*?" The data integration was obtained through the modelling of informative background data using semantic graph traversals, while the inference of species distributions was carried out with the specification (and implementation in the case of chapter 4) of multilevel hierarchical logistic models for describing the joint effect of an ecological suitability process and the sampling effort. As discussed in chapter 3, several methodologies for modelling single species distributions using presence-only records have been proposed. For example, the work of Renner et al. (2019) suggested the use of a spatial latent factor to jointly model sampling effort and ecological processes. Additionally, Croft et al. (2019) used an intrinsic conditional autoregressive (ICAR) model as the spatial dependency. This part of the research explored and discussed the above proposals. It was found that they were limited to a single spatial specification and a rigid use of the sampling effort process.

The research in this part of the thesis contributed to the field with a generalisation of the spatial specification into three different types of spatial structures: independent components (model I), common spatial component (model II) and correlated components (model III). Additionally, this work proposed a flexible scheme to model absences through

[1] with missing data in the case of Chapter 2

the use of the *choosing principle*. The development of this work brought novel statistical and computational methods (increased presence-only single species distribution model accuracy, compared with the popular model MaxEnt), and gave ideas for extending the presence-only framework to model multiple species (taxa) simultaneously.

In the proposed framework, all the models (i.e. I, II and III) performed more accurately than benchmark (i.e. MaxEnt). However, the results given by the two examples showed that the size of the informative sample plays an important role in choosing the appropriate model accordingly with its spatial autocorrelation structure. That is, model III resulted to achieve the highest predictive accuracy when the informative sample has a low proportion of missing data, while model II was better suited for cases where the proportion of missing data of the informative sample is larger than the presences or relative absences.

Single species distribution models have been criticised for not including interaction between species (e.g Elith and Leathwick (2009)). In fact, species interactions have been described as fundamental forces for determining the likelihood of organisms to occupy an ecological niche (Chase and Myers, 2011; Leibold et al., 2004; Wisz et al., 2013). To account for this effect, we decided to specify and implement a joint multispecies model for presence-only occurrences. The field of joint species distribution modelling has been actively growing in recent years. An earlier and influential work was the one developed by Latimer et al. (2009) using a hierarchical approach for binary responses (presence-absence). Later, Wackernagel (2003) used a geostatistical model for co-regionalization to account for a multi-spatial effect. To account for the bias in the number of absences related to abundance data, Clark et al. (2014) proposed a hierarchical model for multispecies abundances using a zero-inflated Poisson process. Despite these efforts, to the best of these authors' knowledge, a joint species distribution model for presence-only data has not been published yet.

Chapter 4 describe an attempt to innovate in this respect by integrating the knowledge engine with a multispecies presence-only spatial framework. To achieve this, a knowledge scheme that traversed the taxonomic tree of co-occurring taxa in neighbouring regions was proposed, to obtain a complementary sample given the selected taxa of interest. That is, without the need of explicitly specifying an *ad-hoc* sample.

One of the main contributions of this thesis is the use of evolutionary-based structures in the form of local taxonomic trees to model background data. In contrast to generic *one-fits-all* heuristics for obtaining background observations (e.g. maximum entropy (MaxEnt)), these structures have the potential to algorithmically generate knowledge-based background information, adapted to the taxa of interest and their geospatial context. This is done with graph traversals that select data following semantic patterns, for example, the relationships of ancestry (i.e `IS_PARENT_OF`) and relationships of neighbouring cells (`IS_NEIGHBOUR_OF`). The obtained background data determine the selected graph pattern, that is, the context of the taxa of interest and their spatial distribution, given associate data linked by semantic relationships. This methodology demonstrated to have a greater predictive capacity than the generic alternative, MaxEnt (see chapter 3).

5.1 Spatial point processes as an alternative

Species distribution models have been a fertile field for the application and development of spatial statistical models (e.g. Cressie et al. (2009); Fortin et al. (2012); Lichstein and Simons (2002)). As it was explained in the introduction, several modelling approaches have been used to address ecological phenomena with presence-only data. Considering that the central objective of this thesis is the inference and prediction of species occurrences in space, it is sensible to assume that these occurrences could be realisations of a random process occurring continuously in space. In contrast to use aggregated observations on a fixed grid of locations across the study area, as in the case of spatial lattices (i.e. Gaussian

random Markov fields), the occurrences could be understood as point-wise realisations of the process itself. This conceptualisation suits the specification of a spatial point model, another branch of spatial statistics (*sensu* Cressie (1993)).

Spatial point processes (SPP) have gain popularity in ecological studies, particularly in modelling species distribution using presence-only records (see Velázquez et al. (2016) for review). The main components of SPP models are: the *intensity function*; the expected density of points per unit area (i.e. a first moment quantity), and the interaction between points; the spatial autocorrelation between the occurrences (i.e. second moment quantity). Although these components are equivalent to the fixed and random effect of the models proposed in chapters 3 and 4, their specification involves fundamental differences in terms of their assumptions, computational complexity and model specification, specially resolving confounding and identifiability conflicts.

An convenient approach for modelling SPP is the use of non-parametric methods for testing hypotheses about spatial clustering or inhibition (Diggle, 2013). Although this approach have showed great value in the ecological sciences, in this research we were more interested in developing integrative frameworks on parametric models that could account for inference, prediction and interpretability. In this regard, we consider the Bayesian framework a fundamental base for such endeavours.

Certainly, methodologies for SPP that allow a Bayesian specification of parametric models exist. An example is the log Gaussian Cox process (Møller et al., 1998) that specifies a (log) Gaussian process as a latent variable for modulating the intensity function. Another example is the approach by (Hengl et al., 2009) that uses density estimations and geostatistical methods. Although these models have been satisfactory used in the ecology, the assumptions about the required study design restrict their application to specific cases, hindering the application to generic taxonomic groups and data collections. The study design assumptions for standard SPP models are:

1. The locations of points are measured exactly.

2. No two points lie at exactly the same location.

3. The survey is exhaustive within the study region. That is, there are no errors in detecting the presence of points of the random process within W.

Clearly, these assumptions restrict the possibility to model a variety of species, specially in a multiple species scenario. For example, in relation to the first assumption, it is impossible for the vast majority of terrestrial animals to have an exact location, as they move constantly within an occupancy area. The second assumption is unsuitable for jointly modelling overlapped species like parasites and hosts or birds perching in trees. In this sense, both groups would probably share the same location (coordinates) and, although, there are multivariate adaptations for SPP, their specification is complex and may encounter methodological problems (Baddeley et al., 2015). Additionally, some records locate the observations in a common geographical coordinate. This means that the locations lack precision and are reliable only within a certain scale. It is very common to find this feature in historical collections like museums and herbaria or where the species under consideration have a high conservation status and their precise location is confidential.

Lastly, the third assumption restricts the concept of modelling jointly the sampling effort and the ecological process that determines the occupancy of an area. In SPP models, all information is contained in the location of the occurrences and it is not possible to separate the sampling effort from the ecological process, leading to confounding and identifiability problems (Gelfand et al. (2013), Chp. 20).

Consequently, the use of spatial lattices (i.e. aggregated data by unit area) for modelling spatial autocorrelation presents a more appropriate alternative. From an epistemological aspect, it can be used to model a wider variety of organisms on an area. From a pragmatic

aspect, it allows a broader use of the data by allowing the use of occurrences with common coordinates, a problem consistent in old datasets or species that, for conservation purposes, their precise location is confidential.

From a methodological aspect, the spatial lattice is indeed a subgraph of the entire knowledge-graph. As such, any selection of it can be represented directly as the adjacency matrix of the selected subgraph. In this way, the spatial autocorrelation of the CAR models (Besag, 1974) is by the adjacency matrix. As such, the complete definition of the input data can be obtained from graph traversals; substructures of the knowledge graph. Lastly, from a computational aspect, the spatial lattice has a great advantage in terms of its computational efficiency, due to its sparse adjacency matrix representation. In this type of matrix, most of the elements are zero. This property can be exploited by sparse numerical methods to invert the matrices in a much reduced computational time (Trefethen and Bau, 1997). This is a great advantage compared to other spatial statistical methods, like geostatistical models (including the log Gaussian Cox process) where the inversion of dense covariance matrices involves a high computational cost; exponential ($O(n^3)$) with respect to the number of data points.

5.2 Limits and recommendations for the future

5.2.1 Absence of rare species

The models presented in chapter 3 and 4 proposed two different sources of background information, both based on the same principle: the use of other occurrences to model the sampling effort associated with the taxa of interest. The modelling of absences, either by the informative sample (chapter 3) or the complementary sample (chapter 4) assume that the taxon of interest is absent when an occurrence of an informative sample is present.

This assumption can be a limitation for modelling rare species, leading to possible biased estimations of their prevalence.

Rare species are difficult to model using solely the proposed models because of their low density across the landscape. As such, applying indiscriminately any *choosing principle* to rare species may hinder the inference of its likely prevalence. This is problematic for analyses where absent species are the study objects. For example, studies related to *dark diversity* (Pärtel et al., 2011), where the taxa expected to be present in the regional species pool are in reality absent in the study area.

5.2.2 Limitations on the single species framework

The aggregated observations associated to the sampling effort process were derived by applying a *choosing principle* to an informative set of sampled observations. The framework requires an informative sample to be defined by the practitioner and, therefore, the model's fitness depends on the particular selection of the sample. Although this feature gives flexibility to reduce the bias in presence-only SDMs, it may be troublesome for studies where the *informative sample* is unknown, difficult to define or impractical to handle by the researchers. In the future, I would like to explore the effect of different informative samples on the inference of the ecological process. We would like to also explore other relations for deriving informative samples. For example, the use of co-occurring general taxa, disregarding the taxonomic proximity.

5.2.3 Limitations of the multiple species model

In chapter 4, the knowledge scheme (graph traversal) was used to generate a *complementary sample*, given a set of taxa of interest. Although it is certainly an advancement from models I, II and III (chapter 3) it does not provide all the flexibility of these models. Specifically, this model lacks: support of missing observations and multivariate spatial

random effects. Contrary to model III on single species, the multiple species model includes only a single spatial random effect, shared between the rest of the taxa and the sampling effort. An implementation of the data augmentation scheme, similarly to the one used in chapter 3 (i.e. (Tanner and Wong, 1987)), and the specification of multivariate conditional autoregresive model (MCAR) (Gelfand and Vounatsou, 2003)) for modelling multiple correlated spatial effects are left for future work. Arguably, the correlations between the random effects could give insights of ecological importance between taxa as demonstrated by Thorson et al. (2015) and later, independently by Ovaskainen et al. (2016) to model a whole community level with a combination of latent factors spatial covariance function for each latent factor.

5.2.4 Computational limitations

The application of the statistical framework to large datasets (several thousands of areas) involves a high computational cost, despite the reduction in complexity aided by the sparse numerical methods used in the inference of the spatial random effect (CAR model). The multispecies model is particularly costly. Its complexity increments proportionally accordingly to the number of species and covariates.

This could be a practical limitation for global or fine spatial resolution studies or analysis that involve the simultaneous modelling of hundred of species. This problem opens new research lines for using other computational methods, for example, approximation methods like INLA or variational inference. A more pragmatic alternative is to change the areal-based autocorrelation structure (CAR model) and use geostatistical models for large datasets like: nearest neighbour Gaussian processes (Datta et al., 2016) or adaptive Gaussian predictive processes (Guhaniyogi et al., 2011), although these methods are not based on graphical models.

5.2.5 Random effects as graph traversals

An interesting research line to develop in the future is the specification of other knowledge schemes to add different random effects. These effects can be specified as graph traversals selecting other relationships of interest, for example, ecological (e.g trophic networks) and evolutionary relations (e.g. phylogenetic trees). The effect of scale can also be modelled in a similar way.

As mentioned in chapter 2, neighbouring cells can be aggregated in a hierarchical taxonomy using the topological relationship: `Is_contained_in` that induces another tree structure. This time a 3-dimensional quad tree (Bentley, 1975; Worboys and Duckham, 2004). The network representations of the knowledge schemes can be unified using probabilistic graphical models (Bishop, 2013). To continue expanding this research the implementation could also be represented by the data generating process as a graph embedded in the knowledge engine. The models presented in chapters 3 and 4 have a functional parametric specification. That is, the data generating process is defined as a family of parametric models. The addition of other sources of random effects and correlation structures would change the structure of the model. As such, a different implementation of the sampler should be built for each model specification. This would result in different implementations for different arrangements of random effects. This is the case for the models in chapter 3, implemented in CARBayes (Lee, 2013) and the multi-species joint model (chapter 4), implemented with STAN (Carpenter et al., 2017)

An interesting route for future research is the extension of the knowledge-engine to fully store the probabilistic graphical models directly in the knowledge graph with automatic implementation of the posterior sampler. This can be achieved with the use of differential programming languages and application programming interfaces (API)s with

automatic differentiation and graph-based numerical computations like TensorFlow [2], PyTorch [3] or PyMC3 (Salvatier et al., 2016).

5.2.6 The temporal dimension

Although the knowledge-engine supports the data querying and extraction of spatio-temporal data, the statistical framework does not account (yet) for any correlation based on time. The decision was taken to ease the specification and implementation of the models. The need to also account for, say, seasonal to long-term variation across time is real and important. Nevertheless, adding another dimension to the modelling framework represents a serious technical challenge. From a computational aspect, the *curse of dimensionality* shows the need to increase substantially the number of data required for fitting the model properly. Consequently, as the data needed to fit is significantly larger, further assumptions to the model need to be imposed to reduce the computational complexity. Additionally, fitting the model with standard MCMC methods requires a much higher computational cost than the spatial-only models. Some research lines following this direction have been developed in recent years. Of importance to our research is the use of integrated Laplace approximation (INLA (Lindgren and Rue, 2015)) to infer spatio-temporal models based on Markov random fields (MRF) (Blangiardo et al., 2013).

[2] https://www.tensorflow.org/
[3] pytorch.org

CHAPTER 6

CONCLUSIONS

This thesis described novel computational and statistical methods for mapping the geographical distributions of organisms using synthesised information of presence-only species records gathered from multiple biodiversity collections.

The knowledge-engine demonstrated the capability to homogenise and synthesise heterogeneous environmental and biodiversity datasets. This was achieved with three interconnected modules that integrate geospatial (raster and vector), temporal and tabular data into graph structures. These structures store and connect different types of data using semantic relationships between the data nodes, allowing the representation and retrieval of data based on concepts. Contrary to other knowledge-engines, the presented engine, supports geoprocessing functions, spatial analyses and efficient spatial querying; as well as, graph-based algorithms for querying and analysing large network structures. The system is also scalable to several terabytes of information. It has been released with an open source license (GPLv.3), allowing its free use and contributions from the community.

The presented statistical framework for modelling species distributions contributed a new approach based on the concept that presence-only occurrences are the joint effect of two stochastic processes, one driven by environmental conditions (ecological suitability) and the other by anthropological factors (sampling effort). In the single species setting

(chapter 3), the framework showed superior predictive accuracy than MaxEnt, one of the most popular methods for species distributions. Contrary to the algorithmic methodology of MaxEnt, the presented framework is a statistical model and, therefore, accounts for parameter uncertainties. The single species model served as a basis for generalisation to the multiple taxa model. The multi-taxa model (chapter 4) represents a new contribution in joint species distribution models. It is the first of its kind to jointly model multiple species using presence-only data. This novel model involves the use of a *complementary sample* for generating, automatically, the sampling effort. The approach benefits from a full integration with the knowledge engine. It traverses the taxonomic tree of the selected taxa to derive observations of the sampling effort based on the taxonomic and spatial context of the selected taxa. The results from the case study showed promising results. In particular, the ecological suitability process reduced the noise and bias influenced by the sampling effort. Also, the ecological process was able to capture clear patterns of macroecological importance. Opening exciting opportunities in future applications.

This thesis tried to contribute to the fields of macroecology and spatial statistics by proposing an integrative approach for extract the intrinsic value of opportunistic (presence-only) biodiversity records into a congruent knowledge-based graphical modelling framework. It also showed the challenges and opportunities involved in the use of big ecological data aimed at synthesising ecological knowledge. The use of knowledge graphs to jointly model taxa using Markov graphical models (e.g. CAR models) showed great potential for integrating other types of ecological processes. There is tremendous potential to include other data sources like trophic webs, pollination or trait networks to reach other fields of ecological importance while improving the modelling of species distributions. ■

REFERENCES

(2018). Django [Computer Software].

(2020). iNaturalist.

Aderhold, A., Husmeier, D., Lennon, J. J., Beale, C. M., and Smith, V. A. (2012). Hierarchical Bayesian models in ecology: Reconstructing species interaction networks from non-homogeneous species abundance data. *Ecological Informatics*, 11:55–64.

Altinel, M., Altinel, M., Luo, Q., Krishnamurthy, S., Mohan, C., and Pirahesh, H. (2002). Dbcache: Database caching for web application servers. *SIGMOD*, 2002:612.

Amante, C. and Eakins, B. (2009). ETOPO1 1 Arc-Minute Global Relief Model: Procedures, Data Sources and Analysis. Technical Report March.

ANACONDA (2016). vers. 2-2.4.0, Anaconda Software Distribution. Computer software.

Andelman, S. J. and Fagan, W. F. (2000). Umbrellas and flagships: Efficient conservation surrogates or expensive mistakes? *Proceedings of the National Academy of Sciences*, 97(11):5954–5959.

Araújo, M. B., Anderson, R. P., Márcia Barbosa, A., Beale, C. M., Dormann, C. F., Early, R., Garcia, R. A., Guisan, A., Maiorano, L., Naimi, B., O'Hara, R. B., Zimmermann, N. E., and Rahbek, C. (2019). Standards for distribution models in biodiversity assessments. *Science Advances*, 5(1):eaat4858.

Araújo, M. B., Pearson, R. G., and Rahbek, C. (2005). Equilibrium of species' distributions with climate. *Ecography*, 28(5):693–695.

Aschehoug, E. T., Brooker, R., Atwater, D. Z., Maron, J. L., and Callaway, R. M. (2016). The Mechanisms and Consequences of Interspecific Competition Among Plants. *Annual Review of Ecology, Evolution, and Systematics*, 47(1):263–281.

Baddeley, A., Rubak, E., and Turner, R. (2015). *Spatial Point Patterns*. Chapman and Hall/CRC.

Barba, L. A. (2019). Praxis of Reproducible Computational Science. *Computing in Science and Engineering*, 21(1):73–78.

Bard, J. B. L. and Rhee, S. Y. (2004). Ontologies in biology: design, applications and future challenges. *Nature Reviews Genetics*, 5(3):213–222.

Beck, J., Böller, M., Erhardt, A., and Schwanghart, W. (2014). Spatial bias in the GBIF database and its effect on modeling species' geographic distributions. *Ecological Informatics*, 19:10–15.

Benito, B. M., Martínez-Ortega, M. M., Muñoz, L. M., Lorite, J., and Peñas, J. (2009). Assessing extinction-risk of endangered plants using species distribution models: A case study of habitat depletion caused by the spread of greenhouses. *Biodiversity and Conservation*, 18(9):2509–2520.

Bentley, J. L. (1975). Multidimensional Binary Search Trees Used for Associative Searching. *Communications of the ACM*, 18(9):509–517.

Besag, J. (1974). Spatial Interaction and the Statistical Analysis of Lattice Systems. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):192–236.

Besag, J., York, J., and Mollié, A. (1991). Bayesian image restoration, with two applications in spatial statistics. *Annals of the Institute of Statistical Mathematics*, 43(1):1–20.

Bishop, C. M. (2013). Model-based machine learning. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*.

Blackwelder, R. E. (1967). *Taxonomy: a text and reference book*. Wiley.

Blangiardo, M., Cameletti, M., Baio, G., and Rue, H. (2013). Spatial and spatio-temporal models with R-INLA. *Spatial and Spatio-temporal Epidemiology*, 4(1):33–49.

Borer, E. T., Harpole, W. S., Adler, P. B., Lind, E. M., Orrock, J. L., Seabloom, E. W., and Smith, M. D. (2014). Finding generality in ecology: A model for globally distributed experiments. *Methods in Ecology and Evolution*, 5(1):65–73.

Brondizio, E. S., Settele, J., Díaz, S., Ngo, H. T., and (editors) (2019). IPBES. 2019 Global assessment report on biodiversity and ecosystem services of the Intergovernmental Science- Policy Platform on Biodiversity and Ecosystem Services. Technical report, Bonn, Germany.

Brook, D. (1964). On the Distinction Between the Conditional Probability and the Joint Probability Approaches in the Specification of Nearest-Neighbour Systems. *Biometrika*, 51(3/4):481.

Brown, I., Martinelli, L. a., Thomas, W., Moreira, M. Z., Cid Ferreira, C., and Victoria, R. a. (1995). Uncertainty in the biomass of Amazonian forests: An example from Rondônia, Brazil. *Forest Ecology and Management*, 75(94):175–189.

Cahill, J. F., Kembel, S. W., Lamb, E. G., and Keddy, P. A. (2008). Does phylogenetic relatedness influence the strength of competition among vascular plants? *Perspectives in Plant Ecology, Evolution and Systematics*, 10(1):41–50.

Cain, S. A. (1944). *Foundations of Plant Geography*. Harper and Brothers, New York and London.

Carpenter, B., Gelman, A., Hoffman, M. D., Lee, D., Goodrich, B., Betancourt, M., Brubaker, M. A., Guo, J., Li, P., and Riddell, A. (2017). Stan: A probabilistic programming language. *Journal of Statistical Software*, 76(1):1–32.

Cavicchioli, R., Ripple, W. J., Timmis, K. N., Azam, F., Bakken, L. R., Baylis, M., Behrenfeld, M. J., Boetius, A., Boyd, P. W., Classen, A. T., Crowther, T. W., Danovaro, R., Foreman, C. M., Huisman, J., Hutchins, D. A., Jansson, J. K., Karl, D. M., Koskella, B., Mark Welch, D. B., Martiny, J. B. H., Moran, M. A., Orphan, V. J., Reay, D. S., Remais, J. V., Rich, V. I., Singh, B. K., Stein, L. Y., Stewart, F. J., Sullivan, M. B., van Oppen, M. J. H., Weaver, S. C., Webb, E. A., and Webster, N. S. (2019). Scientists' warning to humanity: microorganisms and climate change. *Nature Reviews Microbiology*, 17(9):569–586.

Ceballos, G. (2013). *Mammals of Mexico*. The Johns Hopkins University Press, Baltimore.

Ceballos, G., Ehrlich, P. R., Barnosky, A. D., García, A., Pringle, R. M., and Palmer, T. M. (2015). Accelerated modern human–induced species losses: Entering the sixth mass extinction. *Science Advances*.

Celko, J. (2014). *Graph Databases*.

Chandrasekaran, B., Josephson, J. R., and Benjamins, V. R. (1999). What Are Ontologies , and Why Do We Need Them ? *IEEE INTELLIGENT SYSTEMS*.

Chase, J. M. and Myers, J. A. (2011). Disentangling the importance of ecological niches from stochastic processes across scales. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 366(1576):2351–2363.

Chen, M., Mao, S., and Liu, Y. (2014). Big data: A survey. In *Mobile Networks and Applications*, volume 19, pages 171–209.

Clark, J. S., Gelfand, A. E., Woodall, C. W., and Zhu, K. (2014). More than the sum of the parts: forest climate response from joint species distribution models. Technical Report 5.

Clementini, E., Felice, P., and Oosterom, P. (1993). A small set of formal topological relationships suitable for end-user interaction. pages 277–295. Springer, Berlin, Heidelberg.

Clifford, A. H. and Preston, G. B. (1961). *The algebraic theory of semigroups. Vol. I.*, volume 7. American Mathematical Society (AMS), Providence, RI.

CONAFOR (2018). Inventario Nacional Forestal y de Suelos 2009 - 2014. *Inventario Nacional de Suelos*, page 432.

Cressie, N., Calder, C. A., Clark, J. S., Ver Hoef, J. M., and Wikle, C. K. (2009). Accounting for uncertainty in ecological analysis: The strengths and limitations of hierarchical statistical modeling. *Ecological Applications*, 19(3):553–570.

Cressie, N. A. C. (1993). Statistics for Spatial Data. In *Statistics for Spatial Data*, Wiley Series in Probability and Statistics, pages 1–26. John Wiley & Sons, Inc., Hoboken, NJ, USA.

Croft, S., Ward, A. I., Aegerter, J. N., and Smith, G. C. (2019). Modeling current and potential distributions of mammal species using presence only data: A case study on British deer. *Ecology and Evolution*, page ece3.5424.

Darwin, C. (1859). *On the Origin of Species By Means of Natural Selection*. John Murray, London, first edition.

Datta, A., Banerjee, S., Finley, A. O., and Gelfand, A. E. (2016). Hierarchical Nearest-Neighbor Gaussian Process Models for Large Geostatistical Datasets. *Journal of the American Statistical Association*, 111(514):800–812.

de la Torre, J. A., Núñez, J. M., and Medellín, R. A. (2017). Spatial requirements of jaguars and pumas in Southern Mexico. *Mammalian Biology*, 84:52–60.

De Marco, P., Diniz-Filho, J. A. F., and Bini, L. M. (2008). Spatial analysis improves species distribution modelling during range expansion. *Biology Letters*, 4(5):577–580.

De Queiroz, K. and Gauthier, J. (1990). Phylogeny as a central principle in taxonomy: Phylogenetic definitions of taxon names. *Systematic Zoology*, 39(4):307–322.

Díaz, S., Wardle, D. A., and Hector, A. (2009). Incorporating biodiversity in climate change mitigation initiatives. In Naeem, S., Bunker, D. E., Hector, A., Loreau, M., and Perrings, C., editors, *Biodiversity, Ecosystem Functioning, and Human Wellbeing: An Ecological and Economic Perspective*, chapter 11. Oxford University Press.

Dickinson, J. L., Zuckerberg, B., and Bonter, D. N. (2010). Citizen Science as an Ecological Research Tool: Challenges and Benefits. *Annual Review of Ecology, Evolution, and Systematics*, 41(1):149–172.

Diggle, P. J. (2013). *Statistical Analysis of Spatial and Spatio-Temporal Point Patterns*. Chapman and Hall/CRC.

Diggle, P. J., Tawn, J. A., and Moyeed, R. A. (2002). Model-based geostatistics. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 47(3):299–350.

Dirzo, R. and Raven, P. H. (2003). Global State of Biodiversity and Loss. *Annual Review of Environment and Resources*, 28(1):137–167.

Dobzhansky, T. and Dobzhansky, T. G. (1970). *Genetics of the Evolutionary Process*. Columbia University Press.

Docker Inc. (2019). Enterprise Application Container Platform | Docker.

Dormann, C. F. (2007). Promising the future? Global change projections of species distributions. *Basic and Applied Ecology*, 8(5):387–397.

Drever, C. R., Hutchison, C., Drever, M. C., Fortin, D., Johnson, C. A., and Wiersma, Y. F. (2019). Conservation through co-occurrence: Woodland caribou as a focal species for boreal biodiversity. *Biological Conservation*, 232(January):238–252.

Egenhofer, M. J. and Franzosa, R. D. (1991). Point-set topological spatial relations. *International Journal of Geographical Information Systems*, 5(2):161–174.

Ehrlich, P. R. and Ehrlich, A. H. (2013). Can a collapse of global civilization be avoided? *Proceedings of the Royal Society B: Biological Sciences*, 280(1754):1–9.

Elith, J., Graham, C. H., Anderson, R. P., Dudik, M., Ferrier, S., Guisan, A., Hijmans, R. J., Huettmann, F., Leathwick, J. R., Lehmann, A., Li, J., Lohmann, L. G., Loiselle, B. A., Manion, G., Moritz, C., Nakamura, M., Nakazawa, Y., Overton, J. M., Peterson, A. T., Phillips, S. J., Richardson, K., Scachetti-Pereira, R., Schapire, R. E., Soberon, J., Williams, S., Wisz, M. S., and Zimmermann, N. E. (2006). Novel methods improve prediction of species' distributions from occurrence data. *Ecography*, 29(2):129–151.

Elith, J. and Leathwick, J. R. (2009). Species Distribution Models: Ecological Explanation and Prediction Across Space and Time. *Annual Review of Ecology, Evolution, and Systematics*, 40(1):677–697.

Elton, C. (1946). Competition and the Structure of Ecological Communities. *The Journal of Animal Ecology*.

Enquist, B. J., Condit, R. R., Peet, R. K., Schildhauer, M., and Thiers, B. M. (2016). The Botanical Information and Ecology Network (BIEN): Cyberinfrastructure for an integrated botanical information network to investigate the ecological impacts of global climate change on plant biodiversity. *PeerJ*.

Escamilla Molgora, J. M., Sedda, L., and Atkinson, P. M. (2020a). Biospytial: spatial graph-based computing for ecological Big Data. *GigaScience*, 9(5).

Escamilla Molgora, J. M., Sedda, L., and Atkinson, P. M. (2020b). Supporting data for "Biospytial: spatial graph-based computing engine for ecological big data".

Escamilla Mólgora, J. M., Sedda, L., Diggle, P., and Atkinson, P. (2020). A joint distribution framework to improve presence-only species distribution models by exploiting citizen sampling effort.

European Space Agency (2014). Copernicus.

Fabregat, A., Korninger, F., Viteri, G., Sidiropoulos, K., Marin-Garcia, P., Ping, P., Wu, G., Stein, L., D'Eustachio, P., and Hermjakob, H. (2018). Reactome graph database: Efficient access to complex pathway data. *PLOS Computational Biology*, 14(1):e1005968.

Ferrier, S. and Guisan, A. (2006). Spatial modelling of biodiversity at the community level. *Journal of Applied Ecology*, 43(3):393–404.

Ferrier, S., Ninan, K., Leadley, P., Alkemade, R., Acosta, L., Akcakaya, H., Brotons, L., Cheung, W., Christensen, V., Harhash, K., Kabubo-Mariara, J., Lundquist, C., Obersteiner, M., Pereira, H., Peterson, G., Pichs-Madruga, R., Ravindranath, N., Rondinini, C., and Wintle, B. A., editors (2016). *IPBES: The methodological assessment report on Scenarios and Models of Biodiversity and Ecosystem Services*. ecretariat of the Intergovernmental Science-Policy Platform on Biodiversity and Ecosystem Services, Bonn, Germany.

Fick, S. and Hijmans, R. (2017). Worldclim 2: New 1-km spatial resolution climate surfaces for global land areas. *International Journal of Climatology*.

Fielding, A. H. and Bell, J. F. (1997). A review of methods for the assessment of prediction errors in conservation presence/absence models. *Environmental Conservation*, 24(1):38–49.

Foden, W. B. and Young, B. E. (2016). IUCN SSC Guidelines for Assessing Species' Vulnerability to Climate Change. Technical report, Cambridge, United Kingdom.

Foley, J. A. (2005). Global Consequences of Land Use. *Science*, 309(5734):570–574.

Fortin, M. J., James, P. M., MacKenzie, A., Melles, S. J., and Rayfield, B. (2012). Spatial statistics, spatial regression, and graph theory in ecology. *Spatial Statistics*.

Franklin, J., Serra-Diaz, J. M., Syphard, A. D., and Regan, H. M. (2016). Big data for forecasting the impacts of global change on plant communities. *Global Ecology and Biogeography*, pages 6–17.

Friedman, J. (2001). Greedy Function Approximation : A Gradient Boosting Machine Author (s): Jerome H . Friedman Source : The Annals of Statistics , Vol . 29 , No . 5 (Oct ., 2001), pp . 1189-1232 Published by : Institute of Mathematical Statistics Stable URL : http://www. *The Annals of Statistics*, 29(5):1189–1232.

Friedman, J. H. (1991). Multivariate Adaptive Regression Splines. *The Annals of Statistics*, 19(1):1–67.

Futuyma, D. J. (2005). *Evolution*, volume 1. Sunderland, MA.

Gantz, J. and Reinsel, D. (2011). Extracting Value from Chaos. Technical report.

GBIF Secretariat (2015). Global Biodiversity Infrastructure.

GBIF Secretariat (2017). GBIF Backbone Taxonomy.

GBIF.org (2016). GBIF Occurrence Download.

GDAL/OGR Contributors (2018). GDAL/OGR - Geospatial Data Abstraction software Library.

Gelfand, A. E., Diggle, P. J., Fuentes, M., and Guttorp, P. (2013). *Handbook of Spatial Statistics*, volume 53.

Gelfand, A. E., Schmidt, A. M., and Sirmans, C. (2003). Multivariate spatial process models: Conditional and unconditional Bayesian approaches using coregionalization.

Gelfand, A. E. and Shirota, S. (2019). Preferential sampling for presence/absence data and for fusion of presence/absence data with presence-only data. *Ecological Monographs*, page e01372.

Gelfand, A. E., Silander, J. A., Wu, S., Latimer, A., Lewis, P. O., Rebelo, A. G., and Holder, M. (2006). Explaining species distribution patterns through hierarchical modeling. *Bayesian Analysis*, 1(1 A):41–92.

Gelfand, A. E. and Vounatsou, P. (2003). Proper multivariate conditional autoregressive models for spatial data analysis. *Biostatistics*, 4(1):11–15.

Gelman, A., Rubin, D. B., Gelman, A., and Rubin, D. B. (1992). Inference from Iterative Simulation Using Multiple Sequences Linked references are available on JSTOR for this article : Inference from Iterative Simulation Using Multiple Sequences. *Statistical Science*, 7(4):457–472.

Gelman, A. and Shalizi, C. R. (2013). Philosophy and the practice of Bayesian statistics. *British Journal of Mathematical and Statistical Psychology*, 66(1):8–38.

Geometry Engine Open Source (Contributors) (2019). Geometry Engine Open Source.

Geweke, J. (1992). Evaluating the Accuracy of Sampling-Based Approaches to the Calculation of Posterior Moments. *Bayesian Statistics*, 4:1–31.

Golding, N. and Purse, B. V. (2016). Fast and flexible Bayesian species distribution modelling using Gaussian processes. *Methods in Ecology and Evolution*, 7(5):598–608.

Goodchild, M. F. (2007). Citizens as sensors: the world of volunteered geography. *GeoJournal*, 69(4):211–221.

Grund, M., Cudre-Mauroux, P., Krueger, J., and Plattner, H. (2013). Hybrid graph and relational query processing in main memory. In *Proceedings - International Conference on Data Engineering*, pages 23–24.

Guarino, N. (1995). Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human-Computer Studies*, 43(5-6):625–640.

Guhaniyogi, R., Finley, A. O., Banerjee, S., and Gelfand, A. E. (2011). Adaptive Gaussian predictive process models for large spatial datasets. *Environmetrics*, 22(8):997–1007.

Guillera-Arroita, G., Lahoz-Monfort, J. J., and Elith, J. (2014). Maxent is not a presence-absence method: A comment on Thibaud et al. *Methods in Ecology and Evolution*, 5(11):1192–1197.

Guisan, A., Edwards, T. C., and Hastie, T. (2002). Generalized linear and generalized additive models in studies of species distributions: Setting the scene. *Ecological Modelling*, 157(2-3):89–100.

Guisan, A. and Thuiller, W. (2005). Predicting species distribution: Offering more than simple habitat models. *Ecology Letters*, 8(9):993–1009.

Guisan, A., Thuiller, W., and Zimmermann, N. E. (2017). *Habitat suitability and distribution models: With applications in R*.

Guisan, A. and Zimmermann, N. E. (2000). Predictive habitat distribution models in ecology. *Ecological Modelling*, 135(2-3):147–186.

Haegeman, B. and Loreau, M. (2008). Limitations of entropy maximization in ecology. *Oikos*.

Hagberg, A. A., Schult, D. A., and Swart, P. J. (2008). Exploring Network Structure, Dynamics, and Function using NetworkX. In G Varoquaux, T Vaught, and J Millman, editors, *Proceedings of the 7th Python in Science conference (SciPy 2008)*, pages 11–15.

Hardin, G. (1960). The competitive exclusion principle. *Science*.

Harrington, J. L. (2009). *Relational Database Design and Implementation*.

Hartig, F., Dyke, J., Hickler, T., Higgins, S. I., O'Hara, R. B., Scheiter, S., and Huth, A. (2012). Connecting dynamic vegetation models to data - an inverse perspective. *Journal of Biogeography*, 39(12):2240–2252.

Heipke, C. (2010). Crowdsourcing geospatial data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 65(6):550–557.

Hendriks, P. H., Dessers, E., and van Hootegem, G. (2012). Reconsidering the definition of a spatial data infrastructure. *International Journal of Geographical Information Science*, 26(8):1479–1494.

Hengl, T., Sierdsema, H., Radović, A., and Dilo, A. (2009). Spatial prediction of species' distributions from occurrence-only records: combining point pattern analysis, ENFA and regression-kriging. *Ecological Modelling*, 220(24):3499–3511.

Herrig, J. R. (2011). Simple Feature Access - Part 1: Common Architecture | OGC. Technical report, Open Geospatial Consortium Inc.

Hilbert, M. and López, P. (2011). The world's technological capacity to store, communicate, and compute information. *Science (New York, N.Y.)*, 332(6025):60–5.

Hinchliff, C. E., Smith, S. A., Allman, J. F., Burleigh, J. G., Chaudhary, R., Coghill, L. M., Crandall, K. A., Deng, J., Drew, B. T., Gazis, R., Gude, K., Hibbett, D. S., Katz, L. A., Dail Laughinghouse, H., McTavish, E. J., Midford, P. E., Owen, C. L., Ree, R. H., Rees, J. A., Soltisc, D. E., Williams, T., and Cranston, K. A. (2015). Synthesis of phylogeny and taxonomy into a comprehensive tree of life. *Proceedings of the National Academy of Sciences of the United States of America*, 112(41):12764–12769.

Hobbs, N. T. and Hooten, M. B. (2015). *Bayesian models: A statistical primer for ecologists*. Princeton University Press.

Hoffman, M. D. and Gelman, A. (2014). The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15:1593–1623.

Hooper, D. U., Adair, E. C., Cardinale, B. J., Byrnes, J. E., Hungate, B. A., Matulich, K. L., Gonzalez, A., Duffy, J. E., Gamfeldt, L., and Connor, M. I. (2012). A global synthesis reveals biodiversity loss as a major driver of ecosystem change. *Nature*, 486(7401):105–108.

Hornik, K. (2012). The Comprehensive R Archive Network.

Hudak, P. and Paul (1989). Conception, evolution, and application of functional programming languages. *ACM Computing Surveys*, 21(3):359–411.

Hudson, L. N., Newbold, T., Contu, S., Hill, S. L., Lysenko, I., De Palma, A., Phillips, H. R., Senior, R. A., Bennett, D. J., Booth, H., Choimes, A., Correia, D. L., Day, J., Echeverría-Londoño, S., Garon, M., Harrison, M. L., Ingram, D. J., Jung, M., Kemp, V., Kirkpatrick, L., Martin, C. D., Pan, Y., White, H. J., Aben, J., Abrahamczyk, S., Adum, G. B., Aguilar-Barquero, V., Aizen, M. A., Ancrenaz, M., Arbeláez-Cortés, E., Armbrecht, I., Azhar, B., Azpiroz, A. B., Baeten, L., Báldi, A., Banks, J. E., Barlow, J., Batáry, P., Bates, A. J.,

Bayne, E. M., Beja, P., Berg, Å., Berry, N. J., Bicknell, J. E., Bihn, J. H., Böhning-Gaese, K., Boekhout, T., Boutin, C., Bouyer, J., Brearley, F. Q., Brito, I., Brunet, J., Buczkowski, G., Buscardo, E., Cabra-García, J., Calviño-Cancela, M., Cameron, S. A., Cancello, E. M., Carrijo, T. F., Carvalho, A. L., Castro, H., Castro-Luna, A. A., Cerda, R., Cerezo, A., Chauvat, M., Clarke, F. M., Cleary, D. F., Connop, S. P., D'Aniello, B., da Silva, P. G., Darvill, B., Dauber, J., Dejean, A., Diekötter, T., Dominguez-Haydar, Y., Dormann, C. F., Dumont, B., Dures, S. G., Dynesius, M., Edenius, L., Elek, Z., Entling, M. H., Farwig, N., Fayle, T. M., Felicioli, A., Felton, A. M., Ficetola, G. F., Filgueiras, B. K., Fonte, S. J., Fraser, L. H., Fukuda, D., Furlani, D., Ganzhorn, J. U., Garden, J. G., Gheler-Costa, C., Giordani, P., Giordano, S., Gottschalk, M. S., Goulson, D., Gove, A. D., Grogan, J., Hanley, M. E., Hanson, T., Hashim, N. R., Hawes, J. E., Hébert, C., Helden, A. J., Henden, J. A., Hernández, L., Herzog, F., Higuera-Diaz, D., Hilje, B., Horgan, F. G., Horváth, R., Hylander, K., Isaacs-Cubides, P., Ishitani, M., Jacobs, C. T., Jaramillo, V. J., Jauker, B., Jonsell, M., Jung, T. S., Kapoor, V., Kati, V., Katovai, E., Kessler, M., Knop, E., Kolb, A., Korösi, Á., Lachat, T., Lantschner, V., Le Féon, V., Lebuhn, G., Légaré, J. P., Letcher, S. G., Littlewood, N. A., López-Quintero, C. A., Louhaichi, M., Lövei, G. L., Lucas-Borja, M. E., Luja, V. H., Maeto, K., Magura, T., Mallari, N. A., Marin-Spiotta, E., Marshall, E. J., Martínez, E., Mayfield, M. M., Mikusinski, G., Milder, J. C., Miller, J. R., Morales, C. L., Muchane, M. N., Muchane, M., Naidoo, R., Nakamura, A., Naoe, S., Nates-Parra, G., Navarrete Gutierrez, D. A., Neuschulz, E. L., Noreika, N., Norfolk, O., Noriega, J. A., Nöske, N. M., O'Dea, N., Oduro, W., Ofori-Boateng, C., Oke, C. O., Osgathorpe, L. M., Paritsis, J., Parra-H, A., Pelegrin, N., Peres, C. A., Persson, A. S., Petanidou, T., Phalan, B., Philips, T. K., Poveda, K., Power, E. F., Presley, S. J., Proença, V., Quaranta, M., Quintero, C., Redpath-Downing, N. A., Reid, J. L., Reis, Y. T., Ribeiro, D. B., Richardson, B. A., Richardson, M. J., Robles, C. A., Römbke, J., Romero-Duque, L. P., Rosselli, L., Rossiter, S. J., Roulston, T. H., Rousseau, L., Sadler, J. P., Sáfián, S., Saldaña-Vázquez, R. A., Samnegård, U., Schüepp, C., Schweiger, O., Sedlock, J. L., Shahabuddin, G., Sheil, D., Silva, F. A., Slade, E. M., Smith-Pardo, A. H., Sodhi, N. S., Somarriba, E. J., Sosa, R. A., Stout, J. C., Struebig, M. J., Sung, Y. H., Threlfall, C. G., Tonietto, R., Tóthmérész, B., Tscharntke, T., Turner, E. C., Tylianakis, J. M., Vanbergen, A. J., Vassilev, K., Verboven, H. A., Vergara, C. H., Vergara, P. M., Verhulst, J., Walker, T. R., Wang, Y., Watling, J. I., Wells, K., Williams, C. D., Willig, M. R., Woinarski, J. C., Wolf, J. H., Woodcock, B. A., Yu, D. W., Zaitsev, A. S., Collen, B., Ewers, R. M., Mace, G. M., Purves, D. W., Scharlemann, J. P., and Purvis, A. (2014). The PREDICTS database: A global database of how local terrestrial biodiversity responds to human impacts. *Ecology and Evolution*, 4(24):4701–4735.

Hui, F. K. C., Warton, D. I., Foster, S. D., and Dunstan, P. K. (2013). of mixture models vs . separate species distribution models R eports R eports. *Ecology*, 94(9):1913–1919.

Hutchinson, G. (1957). Population studies — animal ecology and demography — Concludig remarks. *Cold Spring Harbor Symposia on Quantitative Biology*, 22:415–427.

Illian, J. B., Martino, S., Sørbye, S. H., Gallego-Fernández, J. B., Zunzunegui, M., Esquivias, M. P., and Travis, J. M. (2013). Fitting complex ecological point process models with integrated nested Laplace approximation. *Methods in Ecology and Evolution*, 4(4):305–315.

INEGI (2015). Conjunto de datos vectoriales de la carta de usoUso del suelo y vegetación, escala 1:250000, serie V (continuo nacional).

Instituto Mexicano del Transporte and Gobierno de Mexico (2014). Red Nacional de Caminos.

Intergovernmental Panel on Climate Change (2014). Climate Change 2014: Impacts, Adaptation and Vulnerability. Summary for Policy Makers. *Climate Change 2014: Impacts, Adaptation and Vulnerability - Contributions of the Working Group II to the Fifth Assessment Report*, pages 1–32.

Isaac, N. J. and Pocock, M. J. (2015). Bias and information in biological records. *Biological Journal of the Linnean Society*, 115(3):522–531.

IUCN (2019). The IUCN Red List of Threatened Species. Version 2013.2. *International Union for Conservation of Nature*, page Available at http://www.iucnredlist.org.

Jamil, T., Ozinga, W. A., Kleyer, M., and Ter Braak, C. J. (2013). Selecting traits that explain species-environment relationships: A generalized linear mixed model approach. *Journal of Vegetation Science*, 24(6):988–1000.

Jiménez-Valverde, A. (2012). Insights into the area under the receiver operating characteristic curve (AUC) as a discrimination measure in species distribution modelling. *Global Ecology and Biogeography*, 21(4):498–507.

Jiménez-Valverde, A., Lobo, J. M., and Hortal, J. (2008). Not as good as they seem: the importance of concepts in species distribution modelling. *Diversity and Distributions*, 14(6):885–890.

Jiménez-Valverde, A., Peterson, A. T., Soberón, J., Overton, J. M., Aragón, P., and Lobo, J. M. (2011). Use of niche models in invasive species risk assessments. *Biological Invasions*, 13(12):2785–2797.

Juneau, J. (2018). Object-Relational Mapping. In *Java EE 8 Recipes*, pages 395–439. Apress, Berkeley, CA.

Kamel Boulos, M. N., Resch, B., Crowley, D. N., Breslin, J. G., Sohn, G., Burtner, R., Pike, W. A., Jezierski, E., and Chuang, K.-Y. (2011). Crowdsourcing, citizen sensing and sensor web technologies for public and environmental health surveillance and crisis management: trends, OGC standards and application examples. *International Journal of Health Geographics*, 10(1):67.

Kasso, M. and Balakrishnan, M. (2013). Ecological and Economic Importance of Bats (Order Chiroptera). *ISRN Biodiversity*, 2013:1–9.

Kattge, J., Diaz, S., Lavorel, S., Prentice, I. C., Leadley, P., Bönisch, G., Garnier, E., Westoby, M., Reich, P. B., Wright, I. J., and Others (2011). TRY–a global database of plant traits. *Global change biology*, 17(9):2905–2935.

Kavanagh, L., Lee, D., and Pryce, G. (2016). Is Poverty Decentralizing? Quantifying Uncertainty in the Decentralization of Urban Poverty. *Annals of the American Association of Geographers*, 106(6):1286–1298.

Keating, K. A. and Cherry, S. (2004). Use and Interpretation of logistic regression in habitat-selection studies. *Journal of Wildlife Management*, 68(4):774–789.

Kelling, S., Fink, D., La Sorte, F. A., Johnston, A., Bruns, N. E., and Hochachka, W. M. (2015). Taking a 'Big Data' approach to data quality in a citizen science project. *Ambio*.

Kemp, K. and Haklay, M. (2014). Open Source Geospatial Foundation (OSGF). In *Encyclopedia of Geographic Information Science*.

Kissling, W. D., Ahumada, J. A., Bowser, A., Fernandez, M., Fernández, N., García, E. A., Guralnick, R. P., Isaac, N. J., Kelling, S., Los, W., McRae, L., Mihoub, J. B., Obst, M., Santamaria, M., Skidmore, A. K., Williams, K. J., Agosti, D., Amariles, D., Arvanitidis, C., Bastin, L., De Leo, F., Egloff, W., Elith, J., Hobern, D., Martin, D., Pereira, H. M., Pesole, G., Peterseil, J., Saarenmaa, H., Schigel, D., Schmeller, D. S., Segata, N., Turak, E., Uhlir, P. F., Wee, B., and Hardisty, A. R. (2018). Building essential biodiversity variables (EBVs) of species distribution and abundance at a global scale. *Biological Reviews*, 93(1):600–625.

Kissling, W. D., Ahumada, J. A., Bowser, A., Fernandez, M., Fernández, N., García, E. A., Guralnick, R. P., Isaac, N. J. B., Kelling, S., Los, W., McRae, L., Mihoub, J.-B., Obst, M., Santamaria, M., Skidmore, A. K., Williams, K. J., Agosti, D., Amariles, D., Arvanitidis, C., Bastin, L., De Leo, F., Egloff, W., Elith, J., Hobern, D., Martin, D., Pereira, H. M., Pesole, G., Peterseil, J., Saarenmaa, H., Schigel, D., Schmeller, D. S., Segata, N., Turak, E., Uhlir, P. F., Wee, B., and Hardisty, A. R. (2017). Building essential biodiversity variables (EBVs) of species distribution and abundance at a global scale. *Biological Reviews*.

Kluyver, T., Ragan-Kelley, B., Pérez, F., Granger, B., Bussonnier, M., Frederic, J., Kelley, K., Hamrick, J., Grout, J., Corlay, S., Ivanov, P., Avila, D., Abdalla, S., Willing, C., and Team, J. D. (2016). Jupyter Notebooks – a publishing format for reproducible computational workflows. In *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, pages 87 – 90.

Köppen, W. (1918). Klassifikation der Klimate nach Temperatur, Niederschlag und Jahresablauf. *Petermanns Geographische Mitteilungen*.

Koricheva, J., Gurevitch, J., and Mengersen, K. L. (2013). *Handbook of meta-analysis in ecology and evolution*. Princeton University Press.

Kunz, T. H., de Torrez, E. B., Bauer, D., Lobova, T., and Fleming, T. H. (2011). Ecosystem services provided by bats. *Annals of the New York Academy of Sciences*, 1223(1):1–38.

Kurzweil, R. (2004). The Law of Accelerating Returns. In *Alan Turing: Life and Legacy of a Great Thinker*, pages 381–416. Springer Berlin Heidelberg, Berlin, Heidelberg.

La Salle, J., Williams, K. J., and Moritz, C. (2016). Biodiversity analysis in the digital era. *Philosophical Transactions of the Royal Society B: Biological Sciences*.

Labs, R. (2012). Redis, an in-memory data structure store.

Latimer, A. M., Banerjee, S., Sang, H., Mosher, E. S., and Silander, J. A. (2009). Hierarchical models facilitate spatial analysis of large data sets: A case study on invasive plant species in the northeastern United States. *Ecology Letters*, 12(2):144–154.

Lee, D. (2013). CARBayes : An R Package for Bayesian Spatial Modeling with Conditional Autoregressive Priors. *Journal of Statistical Software*, 55(13):1–24.

Legendre, P. (1993). Spatial autocorrelation: trouble or new paradigm? *Ecology*, 74(6):1659–1673.

Leibold, M. A., Holyoak, M., Mouquet, N., Amarasekare, P., Chase, J. M., Hoopes, M. F., Holt, R. D., Shurin, J. B., Law, R., Tilman, D., Loreau, M., and Gonzalez, A. (2004). The metacommunity concept: A framework for multi-scale community ecology. *Ecology Letters*, 7(7):601–613.

Lemoine, N. P. (2019). Moving beyond noninformative priors: why and how to choose weakly informative priors in Bayesian analyses. *Oikos*, 128(7):912–928.

Leroux, B. G., Lei, X., and Breslow, N. (2000). Estimation of Disease Rates in Small Areas: A new Mixed Model for Spatial Dependence. pages 179–191.

Lewis, S. L. and Maslin, M. A. (2015). Defining the Anthropocene. *Nature*, 519(7542):171–180.

Li, S., Dragicevic, S., Castro, F. A., Sester, M., Winter, S., Coltekin, A., Pettit, C., Jiang, B., Haworth, J., Stein, A., and Cheng, T. (2016). Geospatial big data handling theory and methods: A review and research challenges. *ISPRS Journal of Photogrammetry and Remote Sensing*, 115:119–133.

Lichstein, J. and Simons, T. (2002). Spatial autocorrelation and autoregressive models in ecology. *Ecological Monographs*, 72(3):445–463.

Lindgren, F. and Rue, H. (2015). Bayesian spatial modelling with R-INLA. *Journal of Statistical Software*, 63(19):1–25.

Liu, L. and Özsu, M. T. (2016). *Encyclopedia of Database Systems*. Springer New York, New York, NY.

Loreau, M. (2010). Linking biodiversity and ecosystems: towards a unifying ecological theory. *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, 365(1537):49–60.

MacArthur, R. H. and Wilson, E. O. (1967). *The theory of island biogeography*, volume 1. Princeton University Press.

Madin, J. S., Bowers, S., Schildhauer, M. P., and Jones, M. B. (2008). Advancing ecological research with ontologies. *Trends in Ecology & Evolution*, 23(3):159–168.

Magurran, A. E. (2004). *Measuring biological diversity*. Blackwell Pub.

Mayr, E. (1940). Speciation Phenomena in Birds. *American Naturalist*, 74(752).

Mayr, E. and Ashlock, P. D. (1991). *Principles of Systematic Zoology*. McGraw-Hill.

Merow, C., Smith, M. J., and Silander, J. A. (2013). A practical guide to MaxEnt for modeling species' distributions: What it does, and why inputs and settings matter. *Ecography*, 36(10):1058–1069.

Mikalef, P., Pappas, I. O., Krogstie, J., and Giannakos, M. (2018). Big data analytics capabilities: a systematic literature review and research agenda. *Information Systems and e-Business Management*.

Millennium Ecosystem Assessment (2005). *Ecosystems and Human Well-being: Synthesis*. Island Press.

Moffett, A., Strutz, S., Guda, N., González, C., Ferro, M. C., Sánchez-Cordero, V., and Sarkar, S. (2009). A global public database of disease vector and reservoir distributions. *PLoS Neglected Tropical Diseases*, 3(3):1–3.

Møller, J., Syversveen, A. R., and Waagepetersen, R. P. (1998). Log Gaussian Cox processes. *Scandinavian Journal of Statistics*, 25(3):451–482.

Monsarrat, S., Boshoff, A. F., and Kerley, G. I. H. (2018). Accessibility maps as a tool to predict sampling bias in historical biodiversity occurrence records. *Ecography*.

National Aeronautics and Space Administration, Administration, N. O., and Atmospheric (2020). Joint Polar Satellite System.

Navarro, L. M., Fernández, N., Guerra, C., Guralnick, R., Kissling, W. D., Londoño, M. C., Muller-Karger, F., Turak, E., Balvanera, P., Costello, M. J., Delavaud, A., El Serafy, G. Y., Ferrier, S., Geijzendorffer, I., Geller, G. N., Jetz, W., Kim, E. S., Kim, H. J., Martin, C. S., McGeoch, M. A., Mwampamba, T. H., Nel, J. L., Nicholson, E., Pettorelli, N., Schaepman, M. E., Skidmore, A., Sousa Pinto, I., Vergara, S., Vihervaara, P., Xu, H., Yahara, T., Gill, M., and Pereira, H. M. (2017). Monitoring biodiversity change through effective global coordination. *Current Opinion in Environmental Sustainability*, 29:158–169.

Niembro-Rocas, A., Vázquez-Torres, M., and Sánchez-Sánchez, O. (2010). *Árboles de Veracruz. 100 especies para la reforestación estratégica*. Secretaría de Educación del Estado de Veracruz, Veracruz.

OpenStreetMap Contributors (2019). OpenStreetMap (OSM).

Ovaskainen, O., Roy, D. B., Fox, R., and Anderson, B. J. (2016). Uncovering hidden spatial structure in species communities with spatially explicit joint species distribution models. *Methods in Ecology and Evolution*, 7(4):428–436.

Ovaskainen, O. and Soininen, J. (2011). Making more out of sparse data: Hierarchical modeling of species communities. *Ecology*, 92(2):289–295.

Pacifici, K., Reich, B. J., Miller, D. A., Gardner, B., Stauffer, G., Singh, S., McKerrow, A., and Collazo, J. A. (2017). Integrating multiple data sources in species distribution modeling: A framework for data fusion. *Ecology*, 98(3):840–850.

Pahl, C. and Lee, B. (2015). Containers and clusters for edge cloud architectures-A technology review. In *Proceedings - 2015 International Conference on Future Internet of Things and Cloud*, pages 379–386.

Pärtel, M., Szava-Kovats, R., and Zobel, M. (2011). Dark diversity: Shedding light on absent species. *Trends in Ecology and Evolution*, 26(3):124–128.

Pavoine, S. and Bonsall, M. B. (2011). Measuring biodiversity to explain community assembly: a unified approach. *Biol Rev Camb Philos Soc*, 86(4):792–812.

Pereira, H. M., Ferrier, S., Walters, M., Geller, G. N., Jongman, R. H., Scholes, R. J., Bruford, M. W., Brummitt, N., Butchart, S. H., Cardoso, A. C., Coops, N. C., Dulloo, E., Faith, D. P., Freyhof, J., Gregory, R. D., Heip, C., Höft, R., Hurtt, G., Jetz, W., Karp, D. S., McGeoch, M. A., Obura, D., Onoda, Y., Pettorelli, N., Reyers, B., Sayre, R., Scharlemann, J. P., Stuart, S. N., Turak, E., Walpole, M., and Wegmann, M. (2013). Essential biodiversity variables.

Pereira, H. M., Leadley, P. W., Proença, V., Alkemade, R., Scharlemann, J. P., Fernandez-Manjarrés, J. F., Araújo, M. B., Balvanera, P., Biggs, R., Cheung, W. W., Chini, L., Cooper, H. D., Gilman, E. L., Guénette, S., Hurtt, G. C., Huntington, H. P., Mace, G. M., Oberdorff, T., Revenga, C., Rodrigues, P., Scholes, R. J., Sumaila, U. R., and Walpole, M. (2010). Scenarios for global biodiversity in the 21st century. *Science*, 330(6010):1496–1501.

Perez, S., Jandl, R., and Rubio, A. (2007). Modelización del secuestro de carbono en sistemas forestales: Efecto de la elección de especie. *Ecología*, 21:341–352.

Perkel, J. M. (2018). A toolkit for data transparency takes shape. *Nature*, 560(7719):513–515.

Peterson, A. T., Soberón, J., Pearson, R. G., Anderson, R. P., Martínez-Meyer, E., Nakamura, M., and Araújo, M. B. (2011). *Ecological Niches and Geographic Distributions (MPB-49)*. Princeton University Press.

Phillips, S. J., Anderson, R. P., Dudík, M., Schapire, R. E., and Blair, M. E. (2017). Opening the black box: an open-source release of Maxent. *Ecography*, 40(7):887–893.

Phillips, S. J., Anderson, R. P., and Schapire, R. E. (2006a). Maximum entropy modeling of species geographic distributions. *Ecological Modelling*, 190(3-4):231–259.

Phillips, S. J., Anderson, R. P., and Schapire, R. E. (2006b). Maximum entropy modeling of species geographic distributions. *Ecological Modelling*, 190(3-4):231–259.

Phillips, S. J., Dudík, M., Elith, J., Graham, C. H., Lehmann, A., Leathwick, J., and Ferrier, S. (2009). Sample selection bias and presence-only distribution models: implications for background and pseudo-absence data. *Ecological Applications*, 19(1):181–197.

Phillips, S. J. and Elith, J. (2013). On estimating probability of presence from use-availability or presence-background data. *Ecology*, 94(6):1409–1419.

Piaget, J. (1952). *The Origins of Intelligence in Children*.

Plummer, M., Best, N., Cowles, K., and Vines, K. (2006). CODA: Convergence Diagnosis and Output Analysis for MCMC. *R News*, 6(1):7 – 11.

Popper, K. (1966). Objective Knowledge: A Realist View of Logic, Physics, and History. *Oxford University Press*.

PROJ Contributors (2019). PROJ coordinate transformation software library.

Ramsey, P., Santilli, S., Obe, R., Cave-Ayland, M., and Park, B. (2018). PostGIS.

Reinsel, D., Gantz, J., and Rydning, J. (2018). The Digitization of the World - From Edge to Core. *IDC White Paper*, (US44413318).

Renner, I. W., Elith, J., Baddeley, A., Fithian, W., Hastie, T., Phillips, S. J., Popovic, G., and Warton, D. I. (2015). Point process models for presence-only analysis. *Methods in Ecology and Evolution*, 6(4):366–379.

Renner, I. W., Louvrier, J., and Gimenez, O. (2019). Combining multiple data sources in species distribution models while accounting for spatial dependence and overfitting with combined penalised likelihood maximisation. *Methods in Ecology and Evolution*, pages 2041–210X.13297.

Roberts, G. O. and Tweedie, R. L. (2006). Exponential Convergence of Langevin Distributions and Their Discrete Approximations. *Bernoulli*, 2(4):341.

Rodriguez, M. a. (2015). The Gremlin Graph Traversal Machine and Language. *Proc. 15th Symposium on Database Programming Languages*, pages 1–10.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*.

Royle, J. A. and Kéry, M. (2007). A Bayesian state-space formulation of dynamic occupancy models. *Ecology*, 88(7):1813–1823.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Rue, H. and Held, L. (2005). *Gaussian markov random fields: Theory and applications*. Chapman & Hall/CRC.

Rzedowski, J. (2006). *Vegetación de México*. Comisión Nacional para el Conocimiento y Uso de la Biodiversidad, Mexico, 1ra. edici edition.

Sahr, K., White, D., and Kimerling, A. J. (2003). Geodesic discrete global grid systems. *Cartography and Geographic Information Science*, 30(2):121–134.

Salvatier, J., Wiecki, T. V., and Fonnesbeck, C. (2016). Probabilistic programming in Python using PyMC3. *PeerJ Computer Science*, 2:e55.

Sandström, U. G., Angelstam, P., and Mikusiński, G. (2006). Ecological diversity of birds in relation to the structure of urban green space. *Landscape and Urban Planning*, 77(1-2):39–53.

Sarukhán, J., Koleff, P., Carabias, J., Soberón, J., Dirzo, R., Llorente-Bousquets, J., Halffter, G., González, R., March, I., Mohar, A., Anta, S., and de la Maza, J. (2009). Capital Natural de Mexico. Síntesis: Conocimiento actual y perspectivas de sustentabilidad. *Comisión Nacional para el Conocimiento y Uso de la Biodiversidad, México*.

Scheiter, S., Langan, L., and Higgins, S. I. (2013). Next-generation dynamic global vegetation models: Learning from community ecology. *New Phytologist*, 198(3):957–969.

Schmeller, D. S., Mihoub, J. B., Bowser, A., Arvanitidis, C., Costello, M. J., Fernandez, M., Geller, G. N., Hobern, D., Kissling, W. D., Regan, E., Saarenmaa, H., Turak, E., and Isaac, N. J. (2017). An operational definition of essential biodiversity variables. *Biodiversity and Conservation*, 26(12):2967–2972.

Scholes, R. J., Walters, M., Turak, E., Saarenmaa, H., Heip, C. H., Tuama, É. Ó., Faith, D. P., Mooney, H. A., Ferrier, S., Jongman, R. H., Harrison, I. J., Yahara, T., Pereira, H. M., Larigauderie, A., and Geller, G. (2012). Building a global observing system for biodiversity.

Seabold, S. and Perktold, J. (2010). Statsmodels: Econometric and Statistical Modeling with Python. *PROC. OF THE 9th PYTHON IN SCIENCE CONF.*

Segurado, P. and Araújo, M. B. (2004). An evaluation of methods for modelling species distributions. *Journal of Biogeography*, 31(10):1555–1568.

Shannon, J. and Walker, K. (2018). Opening GIScience: A process-based approach. *International Journal of Geographical Information Science*, 32(10):1911–1926.

Simpson, G. G. (1953). *Evolution and Geography. An Essay on Historical Biogeography, with Special Reference to Mammals*. Oregon State System of Public Education, Eugene.

Skornyakov, L. o. (2014). Partially ordered set. *Encyclopedia of Mathematics*, October.

Small, N. t. (2017). py2neo [Computer Software].

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., OBI Consortium, t. O., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Scheuermann, R. H., Shah, N., Whetzel, P. L., and Lewis, S. (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, 25(11):1251–5.

Smith, W. (2002). Forest inventory and analysis: a national inventory and monitoring program. *Environmental Pollution*, 116(SUPPL. 1):S233–S242.

Soberón, J. (2007). Grinnellian and Eltonian niches and geographic distributions of species. *Ecology Letters*, 10(12):1115–1123.

Soberon, J. and Nakamura, M. (2009). Niches and distributional areas: Concepts, methods, and assumptions. *Proceedings of the National Academy of Sciences*, 106(Supplement_-2):19644–19650.

Sorichetta, A., Hornby, G. M., Stevens, F. R., Gaughan, A. E., Linard, C., and Tatem, A. J. (2015). High-resolution gridded population datasets for Latin America and the Caribbean in 2010, 2015, and 2020. *Scientific Data*, 2(1):150045.

Spiegelhalter, D. J., Best, N. G., Carlin, B. P., and Van Der Linde, A. (2002). Bayesian measures of model complexity and fit. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 64(4):583–616.

Steffen, W., Grinevald, J., Crutzen, P., and Mcneill, J. (2011). The anthropocene: Conceptual and historical perspectives. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 369(1938):842–867.

Stocker, T. F., Qin, D., Plattner, G.-K., Tignor, M., Allen, S. K., Boschung, J., Nauels, A., Xia, Y., Bex, V., and Midgley, P. M. (2013). (IPCC) Climate Change 2013: The Physical Science Basis. Technical report, Intergovernmental Panel on Climate Change,.

Sullivan, B. L., Wood, C. L., Iliff, M. J., Bonney, R. E., Fink, D., and Kelling, S. (2009). eBird: A citizen-based bird observation network in the biological sciences. *Biological Conservation*.

Tanner, M. A. and Wong, W. H. (1987). The calculation of posterior distributions by data augmentation. *Journal of the American Statistical Association*, 82(398):528–540.

Team, R. D. C. and R Development Core Team, R. (2016). R: A Language and Environment for Statistical Computing. *R Foundation for Statistical Computing*, 1(2.11.1):409.

Teytelman, L. (2018). No more excuses for non-reproducible methods. *Nature*, 560(7719):411.

Thornton, D., Zeller, K., Rondinini, C., Boitani, L., Crooks, K., Burdett, C., Rabinowitz, A., and Quigley, H. (2016). Assessing the umbrella value of a range-wide conservation network for jaguars (Panthera onca). *Ecological Applications*, 26(4):1112–1124.

Thorson, J. T., Scheuerell, M. D., Shelton, A. O., See, K. E., Skaug, H. J., and Kristensen, K. (2015). Spatial factor analysis: A new tool for estimating joint species distributions and correlations in species range. *Methods in Ecology and Evolution*, 6(6):627–637.

Tikhonov, G., Duan, L., Abrego, N., Newell, G., White, M., Dunson, D., and Ovaskainen, O. (2020). Computationally efficient joint species distribution modeling of big spatial data. *Ecology*, 101(2):1–8.

Tobler, W. R. (1970). A Computer Movie Simulating Urban Growth in the Detroit Region. *Economic Geography*, 46:234.

Trefethen, L. N. and Bau, D. (1997). *Numerical Linear Algebra*. Society for Industrial and Applied Mathematics.

Troudet, J., Grandcolas, P., Blin, A., Vignes-Lebbe, R., and Legendre, F. (2017). Taxonomic bias in biodiversity data and societal preferences. *Scientific Reports*, 7(1):1–14.

Turck, N., Vutskits, L., Sanchez-Pena, P., Robin, X., Hainard, A., Gex-Fabry, M., Fouda, C., Bassem, H., Mueller, M., Lisacek, F., Puybasset, L., and Sanchez, J.-C. (2011). pROC: an open-source package for R and S+ to analyze and compare ROC curves. *BMC Bioinformatics*, 8:12–77.

Udvardy, M. D. F. (1975). A classification of the biogeographical provinces of the world. *IUCN Occasional Paper, International Union for Conservation of Nature and Natural Resources*.

UNEP/CBD (2002). Cancun declaration of like-minded megadiversity countries. In *United Nations Environmental Program-Convention on Biological Diversity (UNEP-CBD)*, page UNEP/CBD/COP/6/INF/33, The Hague, Netherlands.

UNEP/CBD (2016). Like-minded mega-diverse countries carta to achieve Aichi biodiversity Target 11. In *United Nations Environmental Program-Convention on Biological Diversity (UNEP-CBD)*, page UNEP/CBD/COP/13/INF/45, Cancún, México.

United Nations (1992). Convention on Biological Diversity.

van Iersel, M. P., Pico, A. R., Kelder, T., Gao, J., Ho, I., Hanspers, K., Conklin, B. R., and Evelo, C. T. (2010). The BridgeDb framework: Standardized access to gene, protein and metabolite identifier mapping services. *BMC Bioinformatics*, 11.

Vázquez, A. T. (2018). Portal de Información Geográfica - CONABIO.

Vehtari, A., Gelman, A., Simpson, D., Carpenter, B., and Bürkner, P.-C. (2019). Rank-normalization, folding, and localization: An improved R-hat for assessing convergence of MCMC.

Velázquez, E., Martínez, I., Getzin, S., Moloney, K. A., and Wiegand, T. (2016). An evaluation of the state of spatial point pattern analysis in ecology. *Ecography*, 39(11):1042–1055.

Vicknair, C., Macias, M., Zhao, Z., Nan, X., Chen, Y., and Wilkins, D. (2010). A comparison of a graph database and a relational database. In *Proceedings of the 48th Annual Southeast Regional Conference on - ACM SE '10*, page 1, New York, New York, USA. ACM Press.

Vidal Zepeda, R. (2005). *Las regiones climaticas de Mexico 1.2.2*. UNAM, Instituto de Geografia.

von Humboldt, A. and Bonpland, A. (1807). *Essai sur la géographie des plantes*. Paris.

Wackernagel, H. (2003). *Multivariate geostatistics: an introduction with applications*. Springer-Verlag Berlin Heidelberg.

Wallace, A. R. (1876). *The Geographical Distribution of Animals*. Harper and brothers, New York.

Walls, R. L., Athreya, B., Cooper, L., Elser, J., Gandolfo, M. A., Jaiswal, P., Mungall, C. J., Preece, J., Rensing, S., Smith, B., and Stevenson, D. W. (2012). Ontologies as integrative tools for plant science. *American journal of botany*, 99(8):1263–75.

Wang, J. F., Zhang, T. L., and Fu, B. J. (2016). A measure of spatial stratified heterogeneity. *Ecological Indicators*, 67:250–256.

Ward, G., Hastie, T., Barry, S., Elith, J., and Leathwick, J. R. (2009). Presence-Only Data and the EM Algorithm. *Biometrics*, 65(2):554–563.

Warton, D. I., Blanchet, F. G., O'Hara, R. B., Ovaskainen, O., Taskinen, S., Walker, S. C., and Hui, F. K. (2015). So Many Variables: Joint Modeling in Community Ecology. *Trends in Ecology and Evolution*, 30(12):766–779.

Webb, C. O., Ackerly, D. D., McPeek, M. A., and Donoghue, M. J. (2002). Phylogenies and Community Ecology. *Annual Review of Ecology and Systematics*, 33(1):475–505.

Wegener, A. (1923). *Die Entstehung der Kontinente und Ozeane*. Druck und Verlag von Friedr. Vieweg & Sohn.

Weigelt, A., Marquard, E., Temperton, V. M., Roscher, C., Scherber, C., Mwangi, P. N., Felten, S., Buchmann, N., Schmid, B., Schulze, E.-D., and Weisser, W. W. (2010). The Jena Experiment: six years of data from a grassland biodiversity experiment. *Ecology*.

Whittaker, R. H. (1972). Evolution and Measurement of Species Diversity. *Taxon*, 21(2/3):213.

Wielinga, B., Schreiber, G., and Breuker, J. (1993). Modelling expertise. In *KADS: A Principled Approach to Knowledge-Based System Development*, Knowledge-Based Systems. Elsevier Science.

Wiemann, S. and Bernard, L. (2016). Spatial data fusion in Spatial Data Infrastructures using Linked Data. *International Journal of Geographical Information Science*, 30(4):613–636.

Wiens, J. A., Stralberg, D., Jongsomjit, D., Howell, C. A., and Snyder, M. A. (2009). Niches, models, and climate change: Assessing the assumptions and uncertainties. *Proceedings of the National Academy of Sciences of the United States of America*, 106(SUPPL. 2):19729–19736.

Wilson, G., Aruliah, D. A., Brown, C. T., Chue Hong, N. P., Davis, M., Guy, R. T., Haddock, S. H. D., Huff, K. D., Mitchell, I. M., Plumbley, M. D., Waugh, B., White, E. P., and Wilson, P. (2014a). Best practices for scientific computing. *PLoS Biology*, 12(1):e1001745.

Wilson, G., Aruliah, D. A., Brown, C. T., Hong, N. P. C., Davis, M., Guy, R. T., Haddock, S. H. D., Huff, K. D., Mitchell, I. M., Plumbley, M. D., Waugh, B., White, E. P., and Wilson, P. (2014b). Best Practices for Scientific Computing. *Plos Biology*, 12(1).

Wisz, M. S., Pottier, J., Kissling, W. D., Pellissier, L., Lenoir, J., Damgaard, C. F., Dormann, C. F., Forchhammer, M. C., Grytnes, J. A., Guisan, A., Heikkinen, R. K., Høye, T. T., Kühn, I., Luoto, M., Maiorano, L., Nilsson, M. C., Normand, S., Öckinger, E., Schmidt, N. M., Termansen, M., Timmermann, A., Wardle, D. A., Aastrup, P., and Svenning, J. C. (2013). The role of biotic interactions in shaping distributions and realised assemblages of species: Implications for species distribution modelling. *Biological Reviews*, 88(1):15–30.

Woese, C. R., Kandler, O., and Wheelis, M. L. (1990). Towards a natural system of organisms: proposal for the domains Archaea, Bacteria, and Eucarya. *Proceedings of the National Academy of Sciences*, 87(12):4576–4579.

Worboys, M. F. and Duckham, M. (2004). *GIS: a computing perspective*. CRC press.

APPENDIX A

EXTRA MATHEMATICAL DEFINITIONS

A.1 Additional mathematical definitions

A.1.1 Network

Definition 11 (Graph or Network) *Let $V(G)$ be a set and $E(G) \subseteq V(G) \times V(G)$. A graph G is a duple given by $(V(G), E(G))$. $V(G)$ is the set of vertices of the graph and $E(G)$ is the set of edges. An example of a graph is draw in figure: 2.1.*

Definition 12 (Subgraph) *Let G be a graph. G' is a subgraph of G ($G' \subseteq G$) if and only if $V(G') \subseteq V(G)$ and $E(G') \subseteq E(G)$.*

Definition 13 (Connected and acyclic graph) *If for every $u, v \in V(G)$ there exist a path that connects them, then G is say to be connected. If that path is unique for every u, v then G is acyclic (without cycles).*

Definition 14 (Tree) *A graph T which is connected and non-cyclic is called* Tree. *An example in figure 2.2*

Definition 15 (Subtree) *Let T be a tree. A subtree T' is a subgraph of T such that is also a tree (i.e. contains no cycles).*

A.1.2 Algebraic operations

The following definitions will give algebraic structure to the model meaning that it will be possible to sum and take the difference of taxonomic tree structures in a similar way as integer, real numbers or matrices operate arithmetically.

Mathematically, the tree data-structure is a semi-lattice and therefore, a partial ordered set and a semi-group (Clifford and Preston, 1961). As such, it is possible to define arithmetic operations among trees to derive new trees. If, additionally, we allow a special tree as an empty tree, containing only the root of the tree of life (LUCA) that can act as an identity element, the set of all possible taxonomic trees and the arithmetic operation constitute a *monoid*.

Definition 16 (Semigroup) *Let T be a set and $m : T \times T \rightarrow T$ be an associative binary operation*[1]*. The duple (T, m) is called a semigroup and T is called the underlying set of the semigroup.*

In this work $s, t \in T$, $m(s, t)$ will be written $s + t$ and is called *sum* if m is defined as *Sum* or $s - t$ if m is defined as *Difference*.

Definition 17 (Identity element) *Let $e \in T$ and T a semigroup. e is called* identity element *if and only if $te = et$ for all $t \in T$. There can only be at most one identity element in a semigroup.*

Definition 18 (Monoid) *A* monoid *is a semigroup with an identity element.*

We will see that the *sum* and *difference* operator of taxonomic trees are monoids.

[1] Meaning that if $t, p, q \in T$ then $m(m(t, p), q) = m(t, m(p, q))$

A.2 Deprecated software and the future of the engine

I want to finish with an anecdote of an unfortunate decision. The idea of the engine was conceived during my masters in 2014. At that time, the design and development were erratic, driven mostly by random explorations while analysing a big database of biological occurrences; a single CSV file with more than 400,000,000 records. Due to time constraints, my proficiency in the language and a library that ended not being used, I decided to develop the engine in Python 2.

My masters dissertation ended with some interesting ideas worth to be continued in the PhD. Specifically, that of representing local taxonomic trees on a grid of cells. In the PhD, I continued working with the code developed before, this time with a more clear design. Soon after, I knew that Python 2 was going to be deprecated in early 2020. The limitations of time forced me to continue developing the engine, knowing that at some point, I would need to translate the code to Python 3.

It is now July 2020 and Python 2 is no longer maintained. Newer and faster libraries for data processing and numerical methods are constantly released for Python 3 and, unfortunately, these new technologies are incompatible with the engine. There is a great task ahead in revisiting the code to translate it to Python 3. If this is done, the implementation of unit-testing modules should be a priority, together with succinct and light container specifications, and a careful and standardised documentation. In my opinion, the knowledge-modelling-engine has potential to grow and become a mature and reliable platform. There is, though, a long way ahead ■

Part III

Published research article

GigaScience, 9, 2020, 1–25

doi: 10.1093/gigascience/giaa039
Technical Note

TECHNICAL NOTE

Biospytial: spatial graph-based computing for ecological Big Data

Juan M. Escamilla Molgora [1,2,*], Luigi Sedda[3] and Peter M. Atkinson[4]

[1]Lancaster Environment Centre, Lancaster University, Library Avenue, Lancaster, LA1 4YQ, UK; [2]Centre for Health Informatics, Computing and Statistics (CHICAS), Lancaster Medical School, Faculty of Health and Medicine, Furness Building, Lancaster University, Lancaster, LA1 4YQ, UK; [3]Lancaster Medical School, Faculty of Health and Medicine, Lancaster University, Furness Building, Lancaster, LA1 4YQ, UK and [4]Faculty of Science and Technology, Lancaster University, Old Engineering Building, Lancaster, LA1 4YQ, UK

*Correspondence address. Juan M. Escamilla Molgora, Lancaster University, Lancaster LA1 4YQ, UK. E-mail: j.escamillamolgora@lancaster.ac.uk http://orcid.org/0000-0002-3682-9828

Abstract

Background: The exponential accumulation of environmental and ecological data together with the adoption of open data initiatives bring opportunities and challenges for integrating and synthesising relevant knowledge that need to be addressed, given the ongoing environmental crises. **Findings:** Here we present Biospytial, a modular open source knowledge engine designed to import, organise, analyse and visualise big spatial ecological datasets using the power of graph theory. The engine uses a hybrid graph-relational approach to store and access information. A graph data structure uses linkage relationships to build semantic structures represented as complex data structures stored in a graph database, while tabular and geospatial data are stored in an efficient spatial relational database system. We provide an application using information on species occurrences, their taxonomic classification and climatic datasets. We built a knowledge graph of the Tree of Life embedded in an environmental and geographical grid to perform an analysis on threatened species co-occurring with jaguars (Panthera onca). **Conclusions:** The Biospytial approach reduces the complexity of joining datasets using multiple tabular relations, while its scalable design eases the problem of merging datasets from different sources. Its modular design makes it possible to distribute several instances simultaneously, allowing fast and efficient handling of big ecological datasets. The provided example demonstrates the engine's capabilities in performing basic graph manipulation, analysis and visualizations of taxonomic groups co-occurring in space. The example shows potential avenues for performing novel ecological analyses, biodiversity syntheses and species distribution models aided by a network of taxonomic and spatial relationships.

Keywords: spatial data infrastructure; biodiversity informatics; ecological knowledge engine; big ecological data; open science

Introduction

The IT revolution has created the opportunity to compute, store, and transfer massive amounts of information. It is estimated that the volume of all digital information will surpass 175 zettabytes (ZB) (1 ZB = 10^{21} bytes) by 2020 [1]. In addition, the growth in data follows an exponential curve that doubles in volume every 2 years ([2–4]). Moreover, this expansion in data production has occurred in all

Received: 19 July 2019; **Revised:** 6 March 2020; **Accepted:** 2 April 2020

human activities, including the environmental sciences. Novel approaches for measuring natural processes are being applied, adding more reliable and diverse data, and environmental measurements cover a wide range of spatial and temporal scales ranging, for example, from long-term ecological experimental plots [5, 6] to near-real time imagery from Earth observation satellite systems such as NASA's Joint Polar Satellite System [7] and ESA's Copernicus programme [8]. This IT era is opening new opportunities for greater understanding of nature. For example, pervasive Internet connectivity has made possible the transfer of data across large distances in a short time, and the multifunctional capabilities of mobile and smart devices have enabled the management and deployment of collaborative surveys at low marginal costs. Geospatial sciences have benefited in particular. Methodologies for collecting, annotating, and curating these new sources of spatial data have been proposed by [9–11] under the term "citizen science," where data are collectively assembled by a community of enthusiasts and volunteers. Some iconic examples of these (crowd-based) platforms are OpenStreetMap [12] for geographic maps and the Global Biodiversity Information Facility (GBIF), an international consortium of research and governmental institutions that gathers and publishes information of all types of biodiversity occurrences [13].

The exponential growth of data imposes new challenges for storage, access, integration, and analysis. Recent years have brought new theoretical methods and technologies that are being developed to tackle these problems. "Big Data" is now an umbrella term for methods dealing with huge, complex, and heterogeneous datasets that cannot be handled with traditional methods. See [14, 15] for a review of the field and [16] for theoretical and practical challenges involving big geospatial data.

A fundamental goal in ecology is the understanding of the relationships between living beings and the environment. A requirement to achieve this goal is the integration of independent studies and measurements to validate hypotheses on potential causal relations. To test the existence of these causalities, a substantial number of inputs in terms of theory, methods, and data is needed. Moreover, reliable, reproducible, and easy-to-access methods are especially important given the urgency in addressing ongoing environmental crises (e.g., rapid ecosystem degradation, global climate change, accelerated extinctions, and biodiversity loss) [17, 18]. Ecology is thus adapting rapidly to these critical challenges and is starting to adopt and develop novel theoretical and computational methods to solve a central problem: how to synthesize and integrate ecological theory with big ecological data. Answering this question requires an interdisciplinary approach that touches many fields, including theoretical ecology, mathematical modelling, statistics, computer science, and information sciences. For example, Loreau [19] proposed a conceptual framework for integrating ecological theory by centering evolution as the link to unify ecology; and Pavoine and Bonsall [20] proposed a semantic and mathematical formalization for unifying traits, species, and phylogenetic diversity. The 2 approaches exemplify how evolutionary (ancestry) relationships between biological objects constitute a solid base to unify distant branches of ecology. From a statistical perspective, meta-analysis has been effective in synthesizing research evidence across independent studies, including unveiling general relations through a statistically sound framework [21].

Geospatial data constitute a crucial component for data fusion and harmonization; see [22] for a review of methods for heterogeneous spatial Big Data fusion, and [23] in order to remove bias by using spatial data stratification methods. A clear example of geospatial data fusion is the building of essential biodiversity variables (EBVs) to identify biodiversity and ecosystem change [24]. EBVs constitute a minimal set of critical variables aimed to standardize and harmonize global biodiversity variables. Originally proposed by the Group on Earth Observations Biodiversity Observation Network (GEO BON) to assess biodiversity change globally [25], EBVs are now being used to predict global species distributions and potential scenarios for policy options [26]. EBVs integrate data in a standardized framework that describes spatial, temporal, and biological organization [27]. Recently, methodologies for building EBVs have been drawing the attention of interdisciplinary research for reliability and data quality [28]. System designs and infrastructures for integrating heterogeneous big ecological data are emerging. Examples of these are the citizen-based bird observation network (eBird [29]), the TRY database for plant traits [30], the PREDICTS project (Projecting Responses of Ecological Diversity in Changing Terrestrial Systems) [31], and the Botanical Information and Ecology Network [32]. Despite the data heterogeneity and biased information against real absences (a consequence of opportunistic sampling), these types of infrastructures are able to collect sufficient quantities of data to perform statistical inference ([33, 34]). The use of high-performance computational technologies with novel statistical methods for representing and modelling big ecological data can provide deeper understanding of biodiversity evolution and its dynamics in a changing world [25, 27, 35]. Moreover, its implications can be extended to other branches of ecology and earth sciences. For example, a process-based approach [36] showed how community assemblages can be integrated into dynamic vegetation models to increase the precision of climatic and earth system models.

From a technical perspective, environmental and ecological data often come in matrix form such that they can be stored and analysed efficiently with a relational database management system (RDBMS) or other tabular data structure. RDBMSs are reliable and sophisticated tools. An important feature is the possibility to extend their functionality with programming languages such as C, Java, Python, or R-Cran. This allows the combined use of an efficient data management system with a broad range of statistical libraries and programming methodologies. An example of this is the integration of spatial analysis tools into the RDBMS through the Postgis project [37], a set of compiled functions written in the Postgresql Procedural Language (PostgresPL) that interfaces with high-level geospatial libraries (e.g., [38–40]). Postgis adds GIS capabilities to the database engine, giving superior performance for querying information with geometric and topological features in space.

Integrating large datasets using only relational methods is computationally intensive. For example, matching data by a common feature involves the definition of join clauses plus computing the joined lookup between the pair of tables. The resulting product is often stored in volatile memory, a limiting factor when integrating large datasets. In a typical database design, table indices cost $O(\log(n))$ in time, where $O(\cdot)$ is the classic "Big O," a measure of computational complexity, and n is the size of the input dataset. A query involving multiple joins (from multiple data tables) can involve reverse and recursive lookups, which can increase the load from $O(n)$ to $O(n^k)$, where k is the number of data tables to join. Although this issue can be addressed with database design techniques such as normalization [41] or caching [42], the solution likely obfuscates the comprehension of the relational schema by adding unintuitive tables and other auxiliary information. It also requires a learning curve and expertise for implementation as well as increasing complexity when more datasets are added.

Data structures based on direct acyclic graphs (DAGs) are advantageous in relation to the above approaches. Traversing a relationship in a graph database has constant cost (O(1)) [43] if the relations are defined explicitly for every node. Whenever a new dataset is added, a new link can be created to relate it with an existing record. Graph databases, however, are not as efficient at processing geospatial queries or handling simultaneous queries [44]. In this sense, hybrid data management systems, capable of handling both paradigms (relational tables and DAGs), were proposed to overcome the limitations of both systems. However, to the best of our knowledge, these proposals have not been yet implemented [45], their code is closed [46], or their scope is not suited for environmental and spatial datasets, as is the case of the Reactome Database [47].

In this article we propose an implementation of an open source knowledge engine (i.e., a hybrid database system) that stores, accesses, and processes geospatial and temporal information, to integrate, analyse, and visualize heterogeneous environmental, EBV, and big ecological data. The engine, named "Biospytial" (composed of the words "biodiversity," "Python," and "spatial" and pronounced "Biospatial"), incorporates semantic relations that integrate data in a web of semantic knowledge able to represent complex graph (network) data structures.

Biospytial can be considered a component of traditional spatial data infrastructure (SDI) because we simplify access and analysis of big datasets while satisfying the need of producing information for scientists and policy makers, among others [48]. This is possible owing to the engine's capability of identifying intrinsic and extrinsic relationships within environmental and socioeconomic processes. Therefore, the developed engine is aimed to serve SDI-based decision-making frameworks, such as, e.g., the European project INSPIRE [49].

The engine serves as a multi-purpose platform for modelling complex and heterogeneous data relationships using the power of graph theory. The current implementation uses the occurrences data from the GBIF and their updated systematic classification [50] to build the acyclic graph of the Tree of Life (ToL). To exemplify the geospatial capabilities, some EBVs such as mean monthly temperature, elevation, and mean monthly precipitation are also included in the engine. The article is structured as follows: the specification and general description of the engine is described in the next section followed by the methodology and software implementation for accessing biodiversity records arranged in a taxonomic tree. The knowledge graph of the ToL is explained with examples for traversing and extracting spatial and taxonomic sub-networks. A tutorial explores the capabilities of the engine with a practical demonstration. This section shows the syntax and discusses ways to interpret and traverse the knowledge graph, ending with general conclusions and future research directions.

An Open Source Graph-Based Engine for Geospatial Analysis

The engine is able to import, organize, analyse, and visualize big ecological datasets using the power of graph theory. It performs geospatial and temporal computations to synthesize information in different forms. The data can be queried and aggregated according to customized specifications defined by structural patterns called "graph traversals" [51]. The software has been developed with object-relational and object-graph mappings (ORM and OGM, respectively) that use the object-oriented paradigm to abstract interrelated data into class instances [43,52]. In this sense, every record is represented as an instance of a certain class with its attributes mapped one-to-one to entries in a particular table (if it is stored in a relational database) or in a key:value hash table (if it is stored in a graph-based database). This approach allows the building of complex and persistent data structures that can represent different aspects of the knowledge base. It also allows the assembly of automatic methods for exploring, filtering, aggregating, and storing information.

System architecture

The engine is composed of 3 interconnected modules: (i) a Relational Geoprocessing Unit (RGU), (ii) the Biospytial Computing Engine (BCE), and (iii) a Graph Storage and Processing Unit (GSPU) (see Fig. 1). Each module is arranged in virtual containers isolated as standalone applications [53] running a common Linux image (Debian 8) as the base operating system. The virtual container technology creates a common environment for each module, enabling the user to disregard the complications of working with heterogeneous computer infrastructures [54]. Its design allows the replication of several instances of the same module in a single computer or in a distributed network. Containerized applications are easier to replicate and migrate compared to large data volumes and databases, which often involve resource-intensive tasks in terms of energy, computing, network bandwidth, and management. The idea behind containerization is to move the processes not the data and, especially in the geospatial context, to perform spatial analysis where the data are located.

The Relational Geoprocessing Unit

The RGU module undertakes the storage and raster-vector processing. It relies on high-level abstractions that represent geospatial data stored in relational tables. The supported geometric features are (multi)points, (multi)lines, (multi)polygons, and multiple-band raster data. It features a fully operational Postgresql (9.4.9) server (port: 5241) with geospatial extension (Postgis 2.3.1) [37] and libraries for handling geospatial data (GDAL, OGR 1.10.1) [38], transformation between different geographic projections (PROJ 4.8 [40]), and computation of geometric operations (GEOS 3.6) [39] (Fig. 1b). The RGU image can be downloaded from [55].

The Graph Storage and Processing Unit

This module hosts a graph database that stores data on nodes and their relations in a network structure called the knowledge base (Fig. 1a). The graph database system is an instance of Neo4J (3.1.3), an open source ACID-compliant transactional database management system with native graph storage and processing [43]. It includes a web-based interface located in http://< custom url >:7474 The interface allows the inspection and visualization of queries (subgraphs) using the Cypher interpreter (a No-SQL type declarative language for interrogating graph databases). The module also includes a plugin for spatial and topological lookups [56] and the Awe-

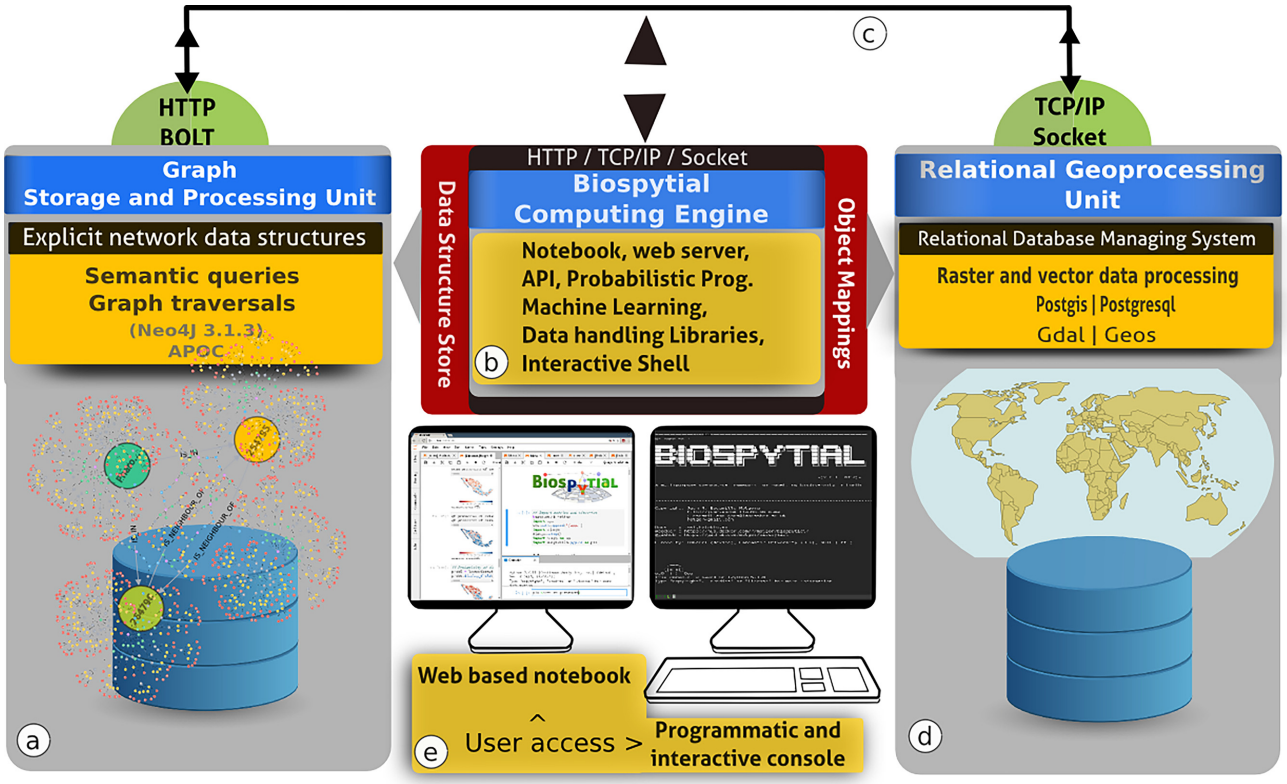


Figure 1: The Biospytial system with the 3 interconnected modules. (a) The GSPU, where semantic queries and graph traversals take place. (b) The BCE, where object mappings, web services, and the modelling framework take place. It includes several libraries for performing exploratory analysis as well as Bayesian statistical inference and prediction using the probabilistic programming language PYMC3. (c) All the components can be allocated in the cloud and are connected using virtual and physical networks. (d) The RGU, where the geoprocessing and spatial indexing occurs, storing efficiently any raster and vector data sources. (e) Interactive access is possible in 2 ways: using an online web notebook (Jupyter) or an interactive console (iPython).

some Procedures on Cypher (APOC) [57], an extension library with >300 procedures for data integration, graph algorithms, or format conversion procedures. The GSPU image can be downloaded from [58].

The Biospytial Computing Engine

This module provides the interface and processing toolbox for accessing, exploring, and analysing data structures through the Object Mapping design. The container hosts a virtual environment and an Anaconda package manager [59] that includes all the dependencies required by the engine. The core code of the engine is contained in a new Python package called Biospytial [60] (Fig. 1c). The engine structure includes a `drivers` module to communfoicate with the graph database; the modules for accessing each dataset in the relational database; the module for graph traversals, data ingestion, gridding systems, vector sketching, and Jupyter notebooks; and external plugins such as `spystats`, a Python port of GeoR [61]. The image can be downloaded from [62].

Other features

Scalable The implementation includes scripts for automating the engine's deployment in a single host or in cluster mode. This mode provides a granular configuration for the allocation of resources and services in a distributed manner. For example, the BCE module can be hosted in a computer with high-performance architectures or multiprocessing (e.g., MPI) capabilities.

Message broker The engine includes a messaging service (Redis [63]) that delivers information between the different components. It also serves as an in-memory data structure storage and message broker. The storage is useful for interchanging data between different platforms and languages. For example, it allows export of the results into intermediary files (e.g., CSV or DBF) for use in other software (e.g., [64, 65]).

Open Source—Open Contributions The software used in all the modules has been released with open source and free software licenses, which allow users to reproduce, modify, and publish their research source code. The engine was developed using best practices for scientific computing [66], data transparency, and reproducibility [67].

Access to the engine

There are 2 ways of accessing the engine. One is through a command line interpreter based on the `iPython` console [68]. The other is with an online Jupyter notebook server [69] (localhost:8888). The Jupyter notebook is a web-based interactive Python interpreter that renders MarkDown documents, plots, and images in the browser. Analysts can create files in a notebook format (.ipdb) and share the

Table 1: Principal software components of the Biospytial Knowledge Engine System

Software name	Version	Description
Biospytial Computing Unit	Debian GNU/Linux 8.6	Container OS image
Conda	4.3.30	Package manager optimized for data science
Python	2.7.11	Programming language (scheduled update for v.3.x)
R-base	3.2	Language and software environment for statistical computing
Jupyter	1.0.0	Interactive web application for reproducible computational workflows
Scipy	1.01	Python library for numerical and scientific computation
Pandas	0.19	Python library for data structures and data analysis
Geopandas	0.3	Extension of Pandas to support geospatial data
GDAL	2.1	Library for converting and processing geospatial data
Shapely	1.5.16	Python library for manipulation and analysis of geometric objects in the Cartesian plane
Django	1.8.4	ORM, web framework and stand-alone server
Py2neo	3.11	A client Python library and toolkit for working with Neo4j
Pymc3	3.4.1	A Python-based probabilistic programming framework
Patsy	0.4.1	A Python library for describing statistical models
Relational Geoprocessing Unit	Debian GNU/Linux 8.6	Container OS image
Postgresql	9.4.9	Relational database management system
Postgis	2.3	Spatial extension for Postgresql
GDAL	1.10.1	Library for converting and processing geospatial data
GEOS	3.6	Geometric and topological library
Proj4	4.8	Coordinate transformation software
Graph Storage and Processing Unit	Alpine Linux 3.5	Container OS image
OpenJDK	IcedTea 3.3	Open source Java compiler and virtual machine
Neo4J	3.1.3 (C.E)	Graph database management system
APOC	3.1.3	Utilities, graph algorithms, and common procedures for Neo4j
Message Broker	Redis 5.0.3	A key-value data structure store

results online. Peers can visit the notebook's url, read the document, run the code, replicate the analysis, access the variables, import other libraries, modify the analysis, and export it into different formats (e.g., PDF, LaTeX, or HTML).

Knowledge representation

The engine uses 2 database paradigms to store and represent data: a relational system with tables connected by primary and foreign keys and directed acyclic graphs (DAGs) where the data are stored as nodes (with associated attributes) and edges representing relations between nodes. Each node can belong to 1 or many classes. In our implementation, the relationships are semantic phrases that refer to location (e.g., "IS IN"), ancestry ("IS PARENT OF"), or topological features ("IS CONTAINED IN" or "IS NEIGHBOUR OF"). Thus, the engine uses explicit semantic relations between nodes to build a network of semantic information. The union of all these relationships is what we call a "knowledge graph."

The event of a species *s* being recorded at location *l* can be represented as a node of the class "Species" connected to a node *l* of class "Cell" using the relation "IS_IN". The Cell nodes are contained in a regular lattice (grid) and are instantiated by a class that implements a geospatial type defined by a polygon that acts as a geometric border. As an example, Fig. 2 shows this diagram for the bird family of quetzales (Trogonidae) found in southeast Mexico. The node in red represents the species: *Pharomachrus mocinno*. The nodes in blue are 2 Cell types that associate the locations where *P. mocinno* was found. The arrows indicate the directional relationships between the nodes. The graph database allows easy manipulation of these nodes, their relations, and combinations. At the same time, the selected pattern can be filtered by chosen attribute values to generate customized design matrices.

Integrating data with graph structures and object mappings

The object mapping approach serves to communicate different database management systems (relational or graph-based). A high-level Python-based Object Relational Mapping (ORM) library (`Django` [70]) was used to communicate with the RDBMS and the other components of the engine. It includes a high-level interface to translate sentences from the SQL declarative language into method calls

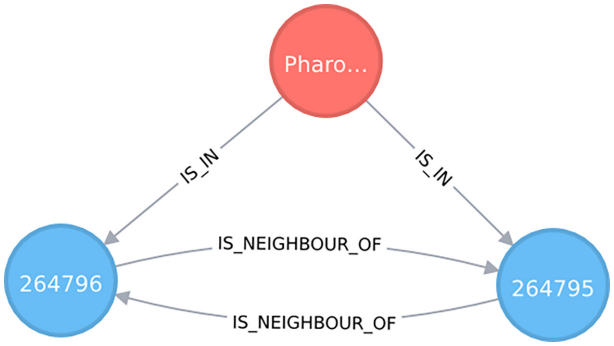


Figure 2: Graph showing the connection between a Species node and 2 Cell nodes. Here the species is *Pharomachrus mocinno* (quetzal) and the number shown in each Cell node is its respective ID number. This is an actual visualization taken from data stored in our knowledge graph.

from the object-oriented paradigm. Vector and raster operations are possible via the Open Source Geographic Information System (OSGIS) for Postgresql (Postgis [37]). Currently, all the spatial and tabular data are stored in the RDBMS.

The object mapping on the graph database system is achieved with `py2neo`, a client library and toolkit for communicating with the Neo4j database management system [71] within the Python programming language [72]. Topological information such as neighbouring cells and nodes contained within cells is stored as semantic relations. Some preprocessed information is stored in the knowledge graph. This includes some parameter estimates, aggregated data, summary statistics, and associated raster metadata.

The procedure for adding data into the engine varies according to the data format (tables or linked data) and requires a new class to be created. The class is responsible for accessing and managing data in both database systems. It includes specifications for storage, conversion between formats, and analysis. A simple implementation would include the name and type of the attributes, the name of the table (for the case of RDBMS), the node type, and incoming and outgoing relations between nodes (for graph-based datasets). Detailed information on all these procedures is given in the supplementary materials: "Adding data in Biospytial".

Graph traversals

As explained above, the knowledge graph is the totality of nodes and relationships stored in the database. Each node represents a type (defined by a class) of data or a more abstract concept that generalizes certain sets of data. Each node has associated edges to other nodes, as well as a list of attributes. In the example given in Fig. 2, the node is of type "Species" and one of its attributes is "name" with the associated value *P. mocinno*.

The graph engine can search and extract information from the knowledge graph using recursive rules based on semantic predicates. Typically, the search selects 1, or several, nodes and continues visiting (traversing) other connected nodes that match the specified criteria until the relationship is exhausted or a depth threshold has been reached. The resulting selection of relationships and nodes is a subgraph of the knowledge graph. We call this structure a "pattern," and the set of rules that select a pattern is a "graph traversal."

Graph traversals can be translated into data matrices that can be analysed within the scope of model-based geostatistics [61] or areal unit modelling in lattice systems using Gaussian Markov random fields [73–75]. Also, they can be analysed with network theory to answer questions about resilience, connectedness, modularity, or invariants across scales. The objects are compatible with the open source libraries for statistical inference and network analysis. Libraries already included in the engine are as follows: NetworkX [76], StatsModels [77], and PyMC3 [78].

Complex queries

Our implementation enforces the use of "lazy evaluations," in which the evaluation of an expression is delayed until the value is needed and not performed directly upon the instantiation [79]. This helps in the creation of data primitives that can be composed into higher level graph traversals without the need to load in all the data. The design allows the request on demand of partial evaluations for a given traversal. This abstraction helps to explore, design, and automate the discovery of relevant patterns and structures. A concrete example of this design is shown in the next section with the analysis of local taxonomic trees; when the tree object is instantiated, it exists only as an abstract data container with no data requested to the database. As such, if an analyst is interested in studying the different species of bats (Order: Chiroptera) within this tree, she will need only to consider the descendant (children) nodes of the node Chiroptera of type Order (see Tutorial section for a practical example).

Some traversals are exclusive of certain node classes and, therefore, have associated special methods. This is the case for nodes of type Cell, which include a method for extracting neighbouring cells. Fig. 3 shows an example of this where a selection of cells was obtained first by requesting all the occurrences of the family Culicidae and then traversing through the associated cells and their corresponding neighbours using the method getNeighbouringCells() twice.

Geospatial management and processing

The engine supports and processes geospatial information using the GDAL/OGR library [38]. The default coordinate reference system (CRS) is the WGS84 with geographic coordinates. However, it is possible to use and reproject the data into any other CRS. This feature is supported by the Proj4 library [40]. See Tutorial section for a concrete example of this.

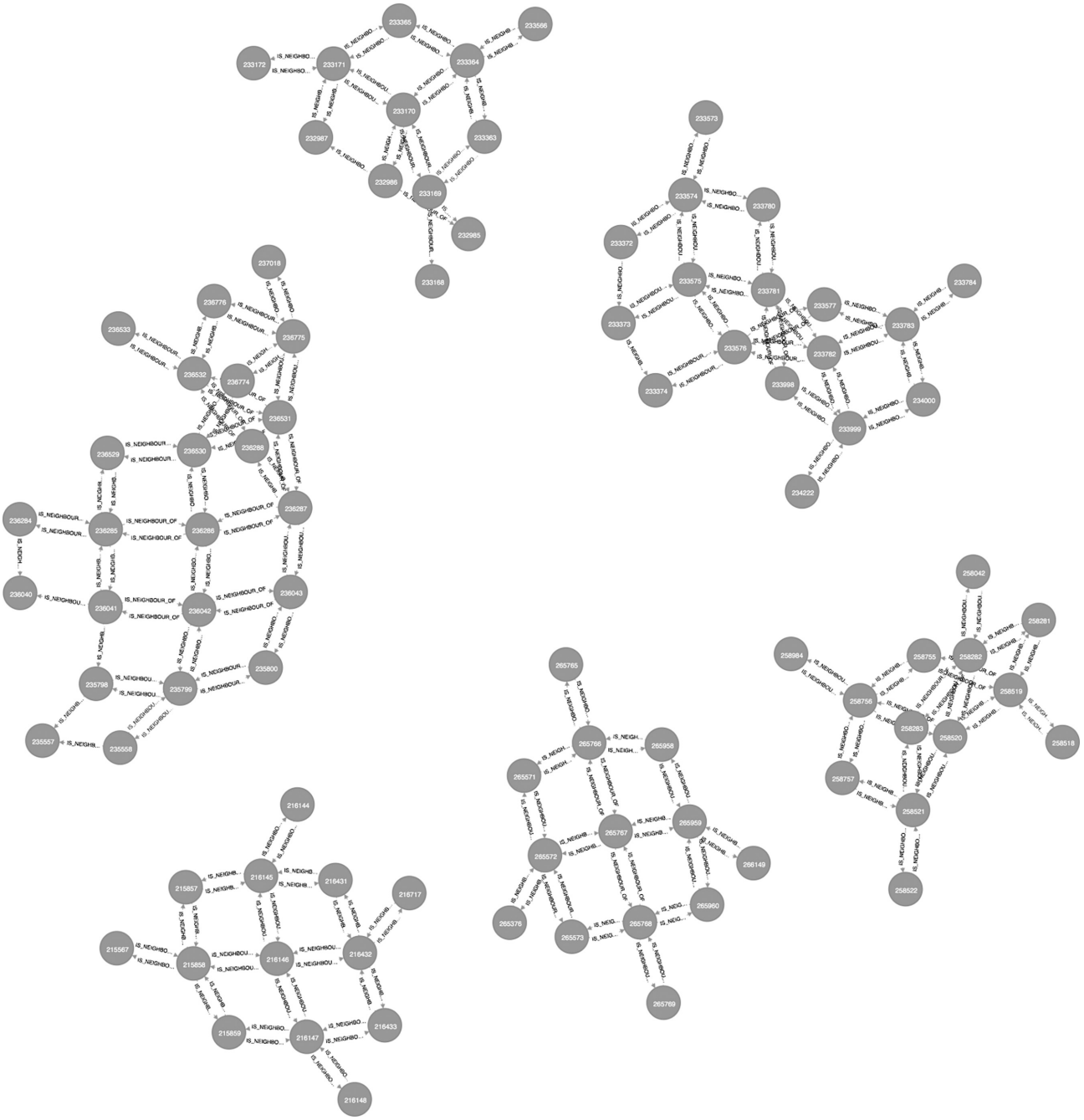


Figure 3: A subgraph from the knowledge engine that shows the second-order degree of neighbouring cells where ≥1 occurrence of any type of mosquito (family Culicidae) was registered. This query exemplifies the use of recursive lookups. In this case the relationship "IS_NEIGHBOUR_OF" is traversed twice.

Vector data

Vector data are represented with tabular data structures. These tables should include the following information: ≥1 column with a unique identifier (ID) for each record, 1 column for each type of feature, and ≥1 geographic column to represent the geometric shape of each record. The available geometric types are points, multiple points, polylines, multiple polylines, polygons, and multiple polygons. Each type of dataset corresponds to both a vector layer and a table in the RDBMS. A mapping between the table structure and the engine needs to be created in the same way as described in engine's specification section. For large datasets the engine uses indexing methods for optimal performance on accessing and querying the data. Additional information is provided in the supplementary material: "Adding data in Biospytial - Vector data".

Raster data

Raster data are represented as a table stored in the RDBMS together with its corresponding metadata. The table has 3 columns: a primary key (ID), a binary large object (BLOB) data type (encoding a stack of matrices) that represents a multiband image, and a

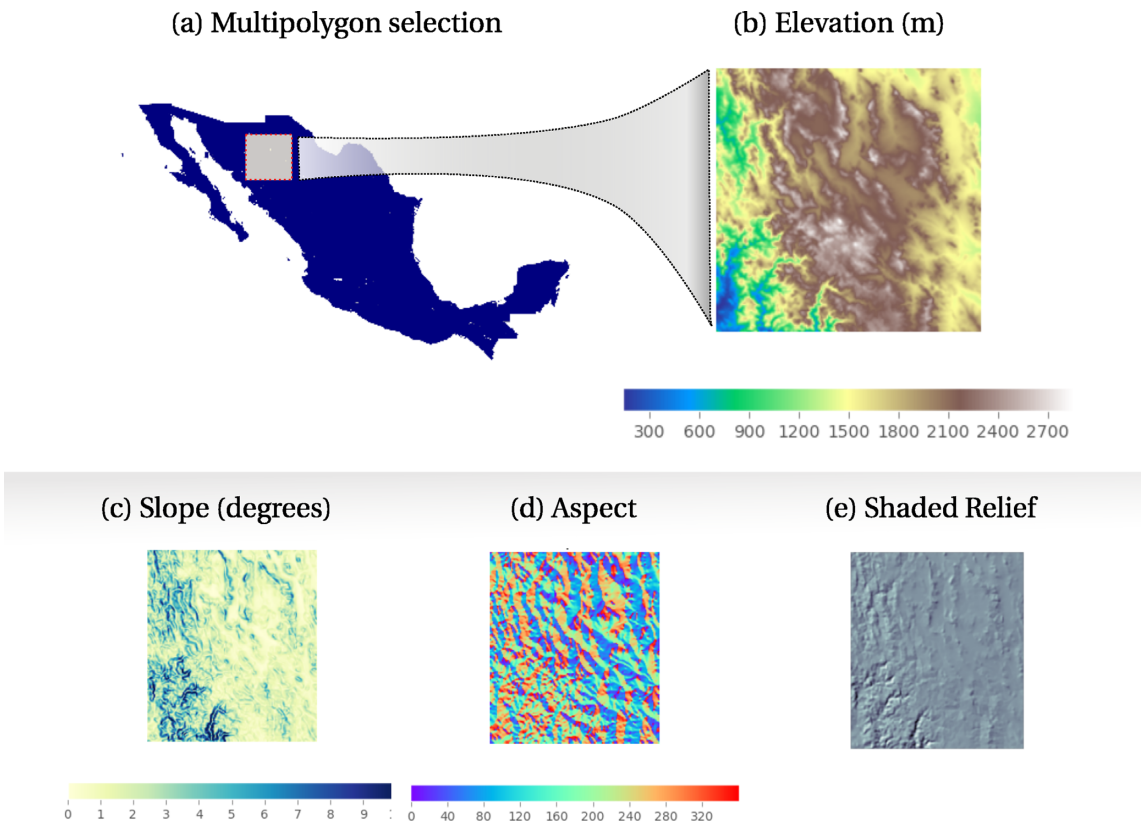


Figure 4: Raster manipulation in the knowledge engine. (a) A multipolygon selection corresponding to Mexico, an instance from the class `Country` that maps into the WorldBorders dataset. (b) An `Elevation` object (class `RasterData`) instantiated with a customized polygon, in this case a subregion of the object Mexico. (c–e) `RasterData` objects derived from the Elevation object. The data and visualizations were produced using the engine's raster API .

reference to a file where the metadata are stored. The metadata includes projection type, affine parameters, datatype for entries (binary, integer, float), and other information related to provenance.

Ingesting raster data into the engine involves 2 steps: (i) the dataset is partitioned into regular tiles, and (ii) each tile is converted into a BLOB string and inserted into the table. Data ingestion scripts can be found in the supplementary materials: "Adding data in Biospytial - Add raster data".

The object mapping design is used to specify the definition of a RasterData type and its associated operations. The implemented class includes methods for clipping, downscaling, aggregating, exporting to image formats (Geotif and PNG), visualizing, intersecting vector data, extracting metadata, and conversion to arrays. An extended class for Digital Elevation Models (DEM) is also implemented to generate on-the-fly aspect, slope, and shaded relief (Fig. 4), without requiring the datasets (derived DEM products) to be stored directly in memory.

On instantiation, a RasterData object requires the definition of a boundary object passed as argument. This object should be a polygon type `django.gis.contrib.GEOS.Polygon` or a text string defining a polygon in the Well Known Text (WKT) format. The resulting selection can be transformed to a dataframe or n-array for statistical modelling. As in the other data structures, whenever a new raster model is added a new model class should be included (see Supplementary Materials: "Adding data in Biospytial - Add Raster data").

Using Biospytial to Analyse the Tree of Life

In this section we propose a process for integrating spatiotemporal data together with graph traversals to represent tree structures using taxonomic and topological relationships within the knowledge engine. The graph traversals use biodiversity occurrences and environmental data to build complex structures to analyse, visualize, and characterize biological occurrences in different forms. The structure restricted to the taxonomic classification is an acyclic graph (tree) in which all the species occurrences constitute leaf nodes. We call this structure the ToL and propose a set of graph traversals to retrieve subsets of the ToL constrained to arbitrary taxonomic groups, spatial regions, or temporal ranges. Several class definitions for handling taxonomic trees are implemented, making it possible to automate tasks for unveiling patterns. For a detailed definition of terms and computational structures see supplementary materials: "Mathematical formalisms".

Study area

The study site selected was restricted to Mexico because (i) Mexico is on the list of megadiverse countries [80,81]; (ii) the territory contains a diverse range of the world's climatic regions [82,83]; and (iii) the country has policies for publishing open environmental data, including centralized repositories of curated data related to biodiversity, conservation, ecosystem services, land cover, and satellite sensor imagery [84]. The data in the study area provide a concrete example of the engine's capabilities.

Data used

The species occurrences were obtained from a snapshot taken from the global GBIF database in September 2016 [13]. The data were filtered to only include the occurrences located within the borders of Mexico. The total number of occurrences is 3,242,746 distributed in 54,828 species, 10,781 genera, 2,300 families, 543 orders, 113 classes, and 42 phyla, with acquisition years ranging from 1819 to 2016. The taxonomic classification was taken from the GBIF Taxonomy Backbone [50]. Each occurrence record has information of species name, location (point coordinates in WGS84), and acquisition date, and represents the observed presence of a certain species; therefore, it is entirely based on presence-only records.

The DEM "ETOPO1 1 Arc-Minute Global Relief Model" [85] was used at a spatial resolution of 1 minute. Precipitation, temperature (maximum, mean, and minimum), solar radiation, wind speed, and vapor pressure were obtained from the World Climatic Data WorldClim version 2 dataset [86]. Each variable is a 12-band raster model with 1 km^2 spatial resolution that aggregates monthly average values from the years 1970 to 2000 per month, each band corresponding to 1 month. The data license for WorldClim restricts the redistribution of the data. Therefore, users need to download it and import it into the engine via an automated script:

`raster_api.bash_raster_tools.migrateToPostgis.bash`

The engine includes functions for generating grid systems at different spatial resolutions. When the grid system is created it stores a vector representation in the RGU and a network representation in the GSPU. The functions for generating the grid systems are located in the library `mesh.tools.py`.

Traversals on the knowledge graph

The taxonomic tree structure was built with the relation `IS_PARENT_OF` (conversely, `Has_Children`) following the taxonomic classification of the occurrence data and the GBIF Backbone Taxonomy [50]. Each occurrence had a location attribute matched with environmental data (e.g., elevation or WorldClim) using a point-in-polygon query to the RGU. The spatial structure was built using the relations `IS_IN` and `IS_CONTAINED_IN` in accordance with topological relationships based on the DE-9IM model [87,88] (standardized by [89]).

The main traversal structure is defined in the TreeNeo class. Each instance comprises an area defined by a spatial polygon and a list of occurrences contained on it. The graph traversal was built recursively using the systematic classification of organisms, starting from the GBIF occurrences as leaf nodes and progressing through the parent nodes until the traversal reaches the node with no parent. That is, it begins at the species level and finalizes in the root node. At each step, the algorithm fetches the available nodes and groups them by their corresponding parent node, generating a set of parent nodes and their associated children. Each of these duples (parent, children) are incorporated into a `LocalTree` object that parses the relevant information into several attributes. This process is applied recursively on each derived parent node of the previous step. The recursion is terminated when the set of parent nodes is empty, generating the desired tree data structure. When this happens the `LocalTree` object is wrapped into a TreeNeo instance that extends some additional methods such as manipulating and querying trees, nodes, and multiple taxonomic groups as well as graph analysis and exportation to common exchange formats (e.g., graphml, data frames, png, geotif, or shapefiles). In addition, all the spatial structures were implemented with Open Source Geospatial (OSGEO) standards [90] to facilitate migration to other languages and platforms. A visualization of this traversal is shown in Fig. 5.

Worked Examples

This section is a case study for analysing the frequency of coexistent taxonomic groups in the entire available dataset restricted to arbitrarily chosen branches of the ToL, included in a list of threatened species. These types of analyses are important in conservation studies, where the characterization of umbrella (or other surrogate) species constitutes the basis for protecting a significant number of associated species [91,92]. To account for this effect, we chose the jaguar (*Panthera onca*) as the species of interest. This is due to its preference for undisturbed ecosystems [93] and its wide geographic required range: 181 ± 4 km^2 for females and 431 ± 152 km^2 males [94].

Additional data used

We use the International Union for Conservation of Nature Red List of Threatened Species (Red List) [95] in Mexico to account for the proportion of species (critically endangered, endangered, or vulnerable) associated with the presence of jaguars. For aggregating the data into taxonomic trees (i.e., `TreeNeo` objects), as well as for extracting their corresponding environmental covariates, we used a 0.05° (~5 km) resolution grid intersected with the terrestrial regions of Mexico and Central America. The grid used is included in the default installation of the engine, and therefore, all the analysis performed in this example is reproducible.

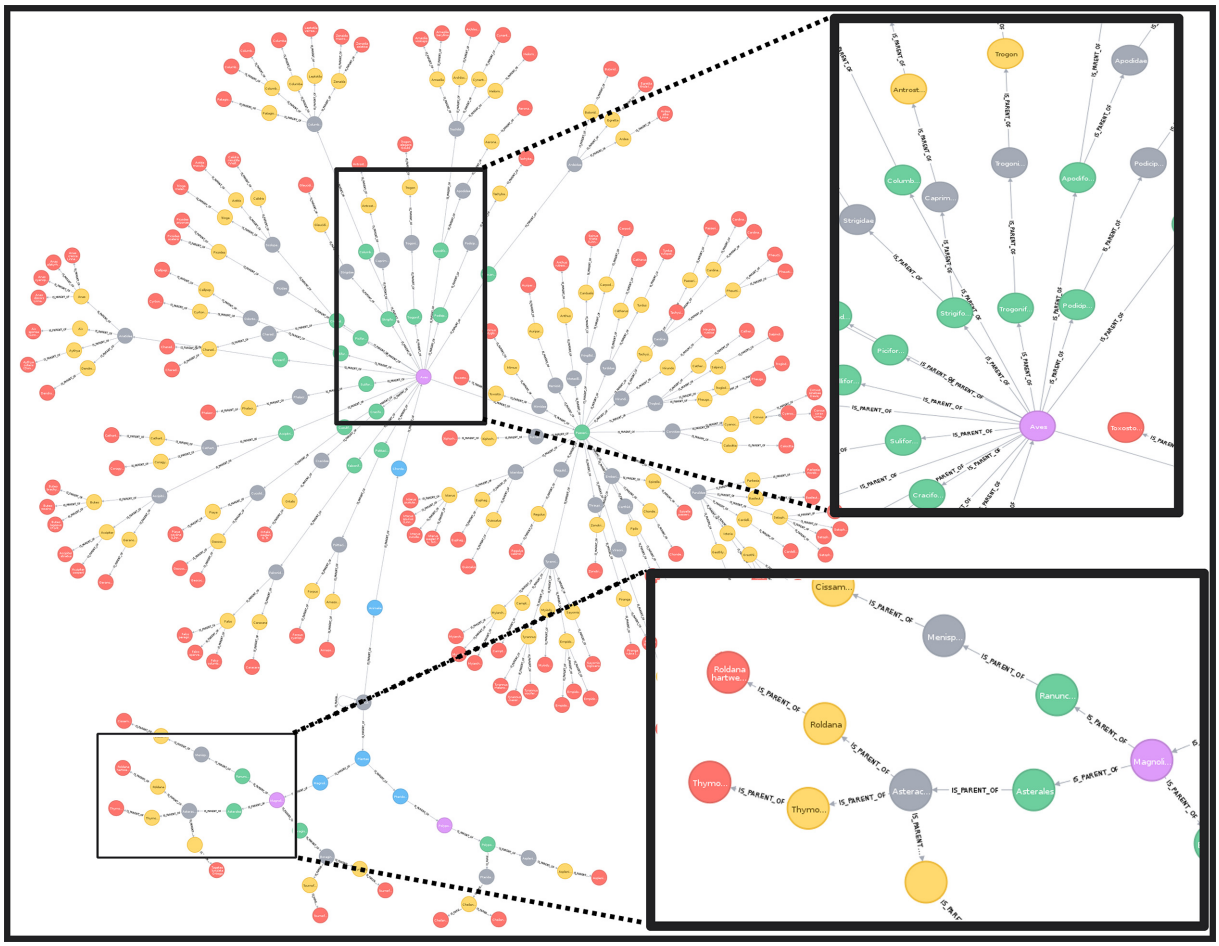


Figure 5: A visualization of a local taxonomic tree built with the relationship `IS_PARENT_OF`. The rectangles show zoomed-in areas in different sections of the tree (upper region for birds [Order Aves], lower for plants [Order Magnoliopsida]). Colored nodes indicate distinct taxonomic levels (red: species; yellow: genera; grey: families; green: orders; purple: classes).

Methodology

We first obtain the grid cells with ≥1 occurrence of jaguar. Because these cells are `Cell` objects, it is possible to extract associated neighbouring cells using the method `getNeighbours`. We can apply the same method recursively 4 times to obtain a list of neighbouring cells within a 4-degree neighbourhood. For each cell, we obtain the local taxonomic tree. The resulting trees are merged into a single tree that contains the union of all the nodes of all the local trees. Therefore, the aggregated tree contains all the known co-occurrences of jaguar in a neighbourhood of degree 4. The resulting tree is filtered to select only the nodes that match the Red List of threatened species. A new tree object is created using the selected nodes, an operation known as "trimming."

To provide an estimate of which nodes co-occur more often with jaguars, we rank all the nodes in the merged tree using the frequency of presence of each node at each neighbouring cell. To show the raster querying capabilities, we contrast these results with the environmental ranges of the following: jaguars, threatened species, and the entire country using the `raster_api` module. Finally, we provide methods for interactive visualizations of the extracted spatial data and the network structure.

Results of the worked example

The taxonomic analysis revealed that the most abundant families across all neighbouring cells were Muridae (rodents, 29%), Phyllostomidae (a family of bats, 23%), and Cervidae (deers, 15%) for the case of mammals. For parrots (Order Psittaciformes) the most frequent species was *Ara militaris* (military macaws, 2%) and several species of the genus *Amazona*, accounting for 16% in total. Although the order Psittaciformes was abundant (23%) in the group of vertebrates, the most abundant taxon (*A. militaris*) only co-occurred 2% of the time with the jaguar's neighbouring cells. This result shows the great diversity of species within the group of parrots. This is consistent with natural history records, where these species have been reported to inhabit humid forests, wooded foothills, and canyons in elevation ranges between 500 and 1,500 m above sea level [96].

The same analysis applied to plants showed that the most abundant genera and species were the epiphyte *Tillandsia* (19%), *Coussapoa oligocephala* (6%), *Pouteria* (several species, 9%), *Cedrela odorata* (3%), which are tropical trees, and other trees not typical from tropical rain forests such as *Oreopanax* (9%) and *Quercus* (6%). Longer lists of the most abundant taxa detailed in the worked example

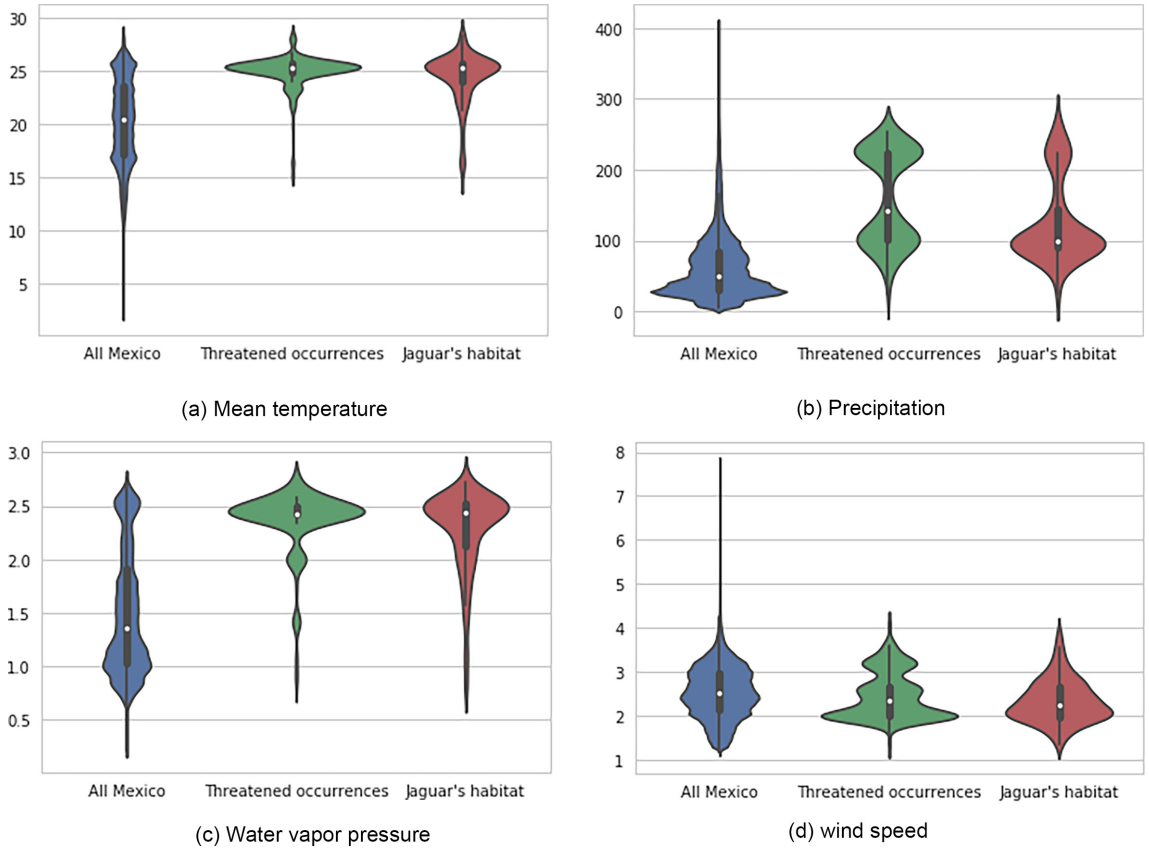


Figure 6: Comparison of mean annual environmental ranges between treatments: all Mexico, threatened taxa, and cells with occurrences of jaguars using violin plots. White dots indicate medians, black bars the interquartile range and stretched black lines lower and upper adjacent values.

as well as their interactive version in the Jupyter notebook are provided in the file `examples/Official Demo Co-occurrences.ipynb` located in the Biospytial repository. A visualization of the threatened taxa tree is shown in Fig. 9 for kingdoms, phyla, classes, and orders.

From an environmental perspective there is a clear concordance between jaguars' habitat and threatened taxa, when compared to all Mexico, for mean temperature (Fig. 6a), annual rainfall (Fig. 6b), and wind speed (Fig. 6d). In fact, threatened species and jaguars show environmental modalities distinct from all Mexico. To create the plots we used the Seaborn library [97]. Detailing the process for creating these graphs is out of the scope of the present tutorial. However, the snippet has been included in the interactive notebook.

Tutorial

The time for executing the following example varies considerably depending on the group of interest, the size of the neighbourhood, and the computer platform. A quick workaround to speed up the processes is to reduce the number of neighbouring cells (order of the neighbourhood). For example using a degree of 1.

A reproducible version of this tutorial is included in the Biospytial source code (inside the folder `examples/`) in an interactive Jupyter notebook file named :

`Official Demo Co-occurrences_jaguar.ipynb`

The following section is a static version and is subject to minor modifications to fit the layout and format of this version.

Selecting the node "Jaguar"

We begin by selecting the node in the ToL corresponding to the genus *Panthera*. This node is linked to some `Species` and `Family` type nodes and also has links to `Occurrence` nodes, where the information of location and time is stored. To start the traversal we need to first select this node. To do so we use the function `pickNode` using the following syntax:

```
pickNode(<Type of Node>,'name of the node')
```

In the next example we see how to load the `pickNode` function and the appropriate node class (in this case `Genus`).

```
from drivers.graph_models import Genus, pickNode
jaguars = pickNode(Genus,''Panthera'')
```

The variable "jaguars" is now an instance of the class Genus. As such, it has associated attributes and methods. Its string representation is the following:

```
jaguars: <TreeNode type: Genus id = 2435194 name: Panthera>
```

We proceed to traverse through all the cells where any occurrence of the *Panthera* genus was registered. To do so we call the attribute "cells." This attribute is abstracted with lazy evaluation. To fetch all the associated data we need to convert the object into a list (or a partial list using an iterator).

```
cells = list(jaguars.cells)
print(''cells has %s elements''%len(cells))
cells has 62 elements
```

The resulting list has cell instances, each one connected to other cells by the relation "IS NEIGHBOUR OF". Accessing their related cells is achieved by the method:

```
cell.getNeighbours(with_center=[Boolean],order=[Int])
```

where the parameter `with_center` returns the center of the neighbourhood, and the parameter `order` the size (in number of cells) of the neighbourhood (this value can be reduced to 1 for faster computation). In our case, we apply this method for each cell using a `map` function with a `lambda` expression.

```
neighbours = map(lambda cell:
                 cell.getNeighbours(with_center=True,order=4),
                 cells)
```

"Lambda expressions" are part of Python [98] and are used to create anonymous functions. The "map-lambda" technique allows the definition of statements that are applied to all the elements of a list, returning a new list of objects obtained by evaluating the lambda expression on every element of the given list. Along this tutorial, the map-lambda technique is frequently used. Whenever this expression comes it is recommended to read the form:

```
map(lambda x: <something involving x> , some_list)
```

as, ""for all *x* in `some_list`, do something involving *x*". In the example above, the object `neighbours` is a list of neighbouring cells obtained from the method `getNeighbours` , available on each cell instance (i.e., each element of the `cells` list).

Because this list is composed of list-type elements (i.e., it is a nested list), we need to reduce it into a single list composed of only cell instances, a process known as flattening. To do this simply reduce the list as follows.

```
# the + operator between 2 list instances merges them together.
neighbours = reduce(lambda list_a , list_b: list_a + list_b, neighbours)
```

The "reduce" function is a Python standard function that receives a 2-parameter function (in this case a lambda expression receiving parameters `list_a` and `list_b`) and the nested list `neighbours`. The reduce function applies the lambda expression to the first pair of elements of the list and iteratively applies the result to the next element. As the sum operation between lists (+) merges the elements of both lists into a single list, performing this operation across the entire nested list `neighbours` results in a flattened list.

The resulting `neighbours` list now has 2,497 Cell nodes. In the current implementation the name of the Grid (where all the Cells are contained) is called "mex4km". We can display the first 3 elements as:

```
neighbours[:3]
[< Cell-mex4km id = 234686 >,
 < Cell-mex4km id = 234685 >,
 < Cell-mex4km id = 234684 >]
```

Converting cells to local taxonomic trees

We obtain the ToL inside each Cell node by extracting the occurrences inside each cell (using the method `occurrencesHere`) and plugging them into the TreeNeo constructor. The name "TreeNeo" is used because the storage backend is the Neo4j graph database.

```
from drivers.tree_builder import TreeNeo
cell_1 = neighbours[1]
tree_1 = TreeNeo(cell_1.occurrencesHere())
print(tree_1)
<LocalTree Of Life | Root: LUCA - n.count: 1062- >
```

The `n.count` value indicates the number of total occurrences. We can generate all the trees iteratively using a mapping from the `TreeNeo(cell.occurrencesHere())` through all neighbouring cells. This may take some time depending on the number of cells and occurrences on each cell. For reducing this time see subsection: "Selecting the node Jaguar".

```
sample_trees = map(lambda cell: TreeNeo(cell.occurrencesHere()),neighbours)
```

As in the last example, we can see such basic information as object description. Here the first 4 elements are shown.

```
sample_trees[:4]
[<LocalTree Of Life | Root: LUCA - n.count: 3- >,
 <LocalTree Of Life | Root: LUCA - n.count: 1062- >,
 <LocalTree Of Life | Root: LUCA - n.count: 151- >,
 <LocalTree Of Life | No record available: - n.count: 0- >]
```

The value `n.count` indicates the number of occurrences found for the present node. It is possible to have empty trees, when no occurrences were found. This is shown with the text `No record available`.

Exploratory analysis on a single tree

We select a tree in this example and explore informative data.

```
tree = sample_trees[1]
```

The object `tree` wraps the entire tree structure. All `tree` objects have as their starting node the root of the Taxonomic Tree, representing all known life.

```
root = tree.node
```

`root` node is similar to `Family` node, `Genus` node, etc. They all belong to the class `TreeNode`. We can access a specific child node with the prefix `to_[name of taxon]`.

For example, accessing the node "Animalia" can be done as follows:

```
animalia = root.to_Animalia
print(animalia)
<LocalTree | Kingdom: Animalia - n.count: 742- | AF: 0.05>
```

Traverse by child nodes

We can concatenate this method until the children attribute is empty. If running Biospytial in an interactive session (like a Jupyter notebook or iPython), we can use the key [TAB] to autocomplete and show the available nodes. For example, the family of rodents Muridae:

```
print(root.to_Animalia.to_Chordata.to_Mammalia.to_Rodentia.to_Muridae)
<LocalTree | Family: Muridae - n.count: 34- | AF: 0.05>
```

Tree traversal by taxonomic level

The taxonomic levels (e.g., families, orders) are stored as attributes of the `TreeNeo` class. For example, to see the available `phyla` in this tree do the following:

```
print(tree.phyla)
[<LocalTree | Phylum: Chordata - n.count: 740- | AF: 0.05 >,
 <LocalTree | Phylum: Arthropoda - n.count: 2- | AF: 0.05 >,
 <LocalTree | Phylum: Bryophyta - n.count: 99- | AF: 0.05 >,
 <LocalTree | Phylum: Magnoliophyta - n.count: 175- | AF: 0.05 >,
 <LocalTree | Phylum: Mycetozoa - n.count: 46- | AF: 0.05 >]
```

and for some families inside this tree:

```
print(tree.families[:5]
[<LocalTree | Family: Menispermaceae - n.count: 3- | AF: 0.05 >,
 <LocalTree | Family: Piperaceae - n.count: 7- | AF: 0.05 >,
 <LocalTree | Family: Lauraceae - n.count: 2- | AF: 0.05 >,
 <LocalTree | Family: Acanthaceae - n.count: 7- | AF: 0.05 >,
 <LocalTree | Family: Plantaginaceae - n.count: 1- | AF: 0.05 >]
```

Tree operations

Tree objects allow symbolic operations for adding (merging) and intersecting other tree objects. These operations are currently implemented as `sum (+)` and `intersection (&)`. These operations can be applied to an arbitrary number of trees, and they are useful in comparative studies that require the calculus of (α, β, γ)-diversity using a combination of these operations [99]. Mathematically, these operations are equivalent to set operations acting at the occurrence level. As an example consider the following: let `t1` and `t2` be 2 trees from the list of `sampled_trees`, i.e.,

```
t1 = sample_trees[1]
t2 = sample_trees[2]
```

Addition

Adding trees is equivalent to merging them. That is, performingunion of all the nodes (internodes and leaves). The tree objects (`TreeNode` and `TreeNeo` classes) allow the use of the + operation. For example, the merged tree of `t1` and `t2` is obtained as follows:

```
t3 = t1 + t2
```

We can see the effect of this by selecting the nodes of a certain taxonomic level, e.g., the classes of `t1` and `t2` are as follows:

```
print(t1.classes)
[<LocalTree | Class: Myxomycetes - n.count: 46- | AF: 0.05 >,
 <LocalTree | Class: Bryopsida - n.count: 99- | AF: 0.05 >,
 <LocalTree | Class: Amphibia - n.count: 1- | AF: 0.05 >,
 <LocalTree | Class: Aves - n.count: 667- | AF: 0.05 >,
 <LocalTree | Class: Reptilia - n.count: 2- | AF: 0.05 >,
 <LocalTree | Class: Mammalia - n.count: 70- | AF: 0.05 >,
 <LocalTree | Class: Liliopsida - n.count: 36- | AF: 0.05 >,
 <LocalTree | Class: Magnoliopsida - n.count: 139- | AF: 0.05 >,
 <LocalTree | Class: Insecta - n.count: 2- | AF: 0.05 >]
print(t2.classes)
[<LocalTree | Class: Protosteliomycetes - n.count: 2- | AF: 0.05 >,
 <LocalTree | Class: Myxomycetes - n.count: 112- | AF: 0.05 >,
 <LocalTree | Class: Agaricomycetes - n.count: 4- | AF: 0.05 >,
 <LocalTree | Class: Liliopsida - n.count: 8- | AF: 0.05 >,
```

```
 <LocalTree | Class: Magnoliopsida - n.count: 25- | AF: 0.05 >]
print(t3.classes)
[<LocalTree | Class: Protosteliomycetes - n.count: 2- | AF: 0.05 >,
 <LocalTree | Class: Myxomycetes - n.count: 158- | AF: 0.05 >,
 <LocalTree | Class: Agaricomycetes - n.count: 4- | AF: 0.05 >,
 <LocalTree | Class: Bryopsida - n.count: 99- | AF: 0.05 >,
 <LocalTree | Class: Amphibia - n.count: 1- | AF: 0.05 >,
 <LocalTree | Class: Aves - n.count: 667- | AF: 0.05 >,
 <LocalTree | Class: Reptilia - n.count: 2- | AF: 0.05 >,
 <LocalTree | Class: Mammalia - n.count: 70- | AF: 0.05 >,
 <LocalTree | Class: Liliopsida - n.count: 44- | AF: 0.05 >,
 <LocalTree | Class: Magnoliopsida - n.count: 164- | AF: 0.05 >,
 <LocalTree | Class: Insecta - n.count: 2- | AF: 0.05 >]
```

Intersection

Intersection is applied through the `&` operation, and it is equivalent to the intersection of sets applied only to the leaf nodes, i.e., the "Occurrence" nodes. Once the leaf nodes are selected, the algorithm propagates through the parent nodes until it reaches the root node. The formalization of the data structure is presented in the supplementary materials: "Mathematical formalisms". To obtain the intersection of 2 trees do the following:

```
t = t1 & t2
print(t)
<LocalTree Of Life | No record available: - n.count: 0- >
```

In this case, the intersection is empty because the Occurrences are overlaid in a regular lattice that partitions the space (i.e., the cells are disjoint). See supplementary materials: "Mathematical formalisms" for a formal definition.

Efficient addition of trees from a list of cells

We can use the sum iteratively in a folding sum to obtain a Tree object representing all the areas defined in a list of Cells.

```
big_tree = reduce(lambda a , b: a+b , sample_trees)
```

However, this method is not efficient. In each step, a new tree is created and the internal logic to generate the union of all the intermediate nodes can result in redundant calculations. It is much faster to select first the occurrences for all the trees inside a list and then plug them into the TreeNeo constructor, as in the example below.

```
# Faster version
ocs = map(lambda s: s.occurrences,sample_trees)
## ocs is a nested list.
## We need to flatten this into  a single list of occurrences
ocs = reduce(lambda a,b: a + b, ocs)
big_tree = TreeNeo(ocs)
print(big_tree)
<LocalTree Of Life | Root: LUCA - n.count: 374731- >
```

The resulting tree could be very large. In this case, the obtained tree (`big_tree`) comprises 374,731 occurrences. Remember that this tree is the resulting union of all the local taxonomic trees obtained from the neighbourhood of degree 4 around the cells where jaguars occurred.

Selecting nodes from the Red List

We filter the "Species" nodes from the `big_tree` that are present in the Red List of threatened species. To do this we simply match the names using regular expressions. Using more sophisticated methods for data matching is out of the scope of the present example. We assume that the Red List data (a CSV file) have been loaded into a data frame with the name `redlist`.

```
## Filter critically endangered species
critical_sps = redlist[
                       (redlist.redlistCategory == 'Critically Endangered')
                      | (redlist.redlistCategory == 'Endangered')
                      | (redlist.redlistCategory == 'Vulnerable')
                     ].scientificName.apply(str.lower)

protected_by_jaguar = map(lambda critical_sp:
                          filter(lambda sp: critical_sp in sp.name.lower(),
                          big_tree.species),
                          critical_sps)

## Remove empty lists
protected_by_jaguar = filter(lambda l:
                             l != [], protected_by_jaguar)
## flatten lists
```

```
threatened_species = reduce(lambda a,b: a + b ,protected_by_jaguar)
## remove species repetitions
threatened_species = list(set(threatened_species))
## Extract all corresponding occurrences and flatten list
t_ocs = reduce(lambda l1,l2: l1 + l2 ,
               map(lambda l: l.occurrences, threatened_species))
## Instantiate new tree
threatened_tree = TreeNeo(t_ocs)
```

The `threatened_tree` is now a taxonomic tree that includes only the occurrences that match the species names of the Red List. To calculate the percentage of threatened species contained in the selected tree we can do the following:

```
## total number of critical endangered species
ncrit = len(critical_sps)
len(threatened_tree.species) / float(ncrit) * 100
13.49 %
```

That is, 13.49% of the threatened species are contained in the neighbouring regions where jaguars had been registered. To see whether this result is relevant, we calculate the percentage of the covered area with respect to the whole country. Before doing so, it is convenient to transform the selected geometries in a projected coordinate system with metric units.

Reprojecting data

The default CRS in the data used is in geographic coordinates with WGS84 datum (EPSG:4326). The units of this CRS are degrees; therefore, the calculated area is defined in squared degrees. To account for areas and distances in metres (or kilometres) we need to project the selected geometries into an appropriate projected coordinate system. To achieve this, we need to import some extra functions.

```
from shapely.ops import transform
from shapely import wkt,wkb
import pyproj
from functools import partial
```

Here we used the Alberts equal area conic projection to account for an accurate area representation. This projection is specified in a string using the Proj4 syntax.

```
projection_string = '''''''+proj=aea +lat_1=14.5 +lat_2=32.5 +lat_0=24
                    +lon_0=-105 +x_0=0 +y_0=0 +ellps=GRS80
                    +datum=NAD83 +units=m +no_defs;
                    '''''''
mex_eq_area_proj = pyproj.Proj(projection_string)
## The WGS84 crs is defined as EPSG:4326
proj_in = pyproj.Proj(init='epsg:4326')
## function to project using the parameters of the
## original projection and the mexican equal area.
project = partial(
    pyproj.transform,
    proj_in,
    mex_eq_area_proj)
## Transform all cells to calculate area.
projected_neighbours_cells = map(lambda cell:
                                 transform(project,
                                 cell.polygon_shapely),
                                 neighbours)
```

To calculate the average cell size and the total area in square kilometers (1,000,000 m^2) we do as follows:

```
tokm2 = 1000000 # to convert to sq. kilometers
areas = map(lambda cell: cell.area,
            projected_neighbours_cells)
total_cell_area = sum(areas)
## calculate the mean
np.mean(areas) / tokm2
## standard deviation
np.std(areas)/ tokm2
```

The calculated average area of all cells is 27 $\pm$ 3 km^2 and the total area is 8,509.81 km^2.

Trimming trees

In certain situations we need to select a particular branch of a tree. We can cut (trim) this branch by simply selecting a node and converting it into a `TreeNeo` instance to produce a full feature tree. The method (function) for converting a TreeNode into a full feature tree is `plantTreeNode`. We focus our attention on 4 branches of the `threatened` tree that co-occur with the presence of jaguars. These branches are mammals (class Mammalia), parrots (order Psittaciformes), amphibians (class Amphibia), and plants (kingdom Plantae).

Select the branch of interest

Trimming the tree is achieved by first selecting the nodes of interest and then converting all the descendant branches into fully featured trees. There is no restriction for selecting the taxonomic type of the node (`mammals` and `amphibians` are `Class` type while `parrots` are `Order` type).

```
mammals = threatened_tree.to_Animalia.to_Chordata.to_Mammalia
parrots = threatened_tree.to_Animalia.to_Chordata.to_Aves.to_Psittaciformes
amphibians = threatened_tree.to_Animalia.to_Chordata.to_Amphibia
plants = threatened_tree.to_Plantae
```

The method `plantTreeNode()` converts the `TreeNode` and resulting descendants into a full featured tree (TreeNeo object).

```
mammals = mammals.plantTreeNode()
birds = birds.plantTreeNode()
amphibians = amphibians.plantTreeNode()
plants = plants.plantTreeNode()
```

We can add all these trees together using the sum operation.

```
vertebrates = mammals + parrots + amphibians
```

However, as explained earlier, an optimized version for summing >2 trees is achieved by instantiating a `TreeNeo` with all the occurrences.

```
vertebrates = TreeNeo(mammals.occurrences +
                      parrots.occurrences +
                      amphibians.occurrences)
print(vertebrates)
```

The total number of occurrences contained in the `vertebrates` tree is:

```
<LocalTree Of Life | Root: LUCA - n.count: 2056- >
```

Ranking the most frequent nodes in the selected list of cells

We proceed now to rank some groups according to their frequency of occurrence within the cells of the study area (i.e., the jaguar's neighbouring cells). The ranking analysis calculates this frequency for each node in a tree given a referential list of trees. That is, assuming that we have *n* different trees (e.g., 1 per cell) and a tree of interest (in this case `threatened_tree`), how frequently does each node appear in the global tree (e.g., `threatened_trees`) with respect to the list of *n* trees? Fig. 9 shows these frequencies visualized as the size of each node. In our implementation, this analysis is performed with the following method: `countNodesFrequenciesOnList(list_of_trees)`. That is,

```
vertebrates.countNodesFrequenciesOnList(list_of_trees=sample_trees)
mammals.countNodesFrequenciesOnList(list_of_trees=sample_trees)
parrots.countNodesFrequenciesOnList(list_of_trees=sample_trees)
amphibians.countNodesFrequenciesOnList(list_of_trees=sample_trees)
plants.countNodesFrequenciesOnList(list_of_trees=sample_trees)
```

We can therefore rank by taxonomic level. In this example we show the procedure for family and species level in the different branches. Here, we show the corresponding top 5 nodes.

```
mammals.rankLevels()
mammals.families[:5]
[<LocalTree | Family: Muridae - n.count: 8 | AF: 0.30>,
 <LocalTree | Family: Phyllostomidae - n.count: 8 | AF: 0.29>,
 <LocalTree | Family: Cervidae - n.count: 14 | AF: 0.16>,
 <LocalTree | Family: Heteromyidae - n.count: 3 | AF: 0.15>,
 <LocalTree | Family: Tayassuidae - n.count: 158
 | AF: 0.15>]
parrots.rankLevels()
parrots.species[:5]
[<LocalTree | Specie: Ara militaris (Linnaeus, 1766) - n.count: 27->,
 <LocalTree | Specie: Amazona finschi (P. L. Sclater, 1864) - n.count: 23- >,
 <LocalTree | Specie: Amazona auropalliata (Lesson, 1842) - n.count: 3- >,
 <LocalTree | Specie: Amazona oratrix Ridgway, 1887 - n.count: 2- >,
amphibians.rankLevels()
amphibians.families[:3]
[<LocalTree | Family: Hylidae - n.count: 128- | AF: 0.083>,
 <LocalTree | Family: Plethodontidae - n.count:
 160 | AF: 0.05>,
 <LocalTree | Family: Eleutherodactylidae -
 n.count: 1- | AF: 0.016>]
plants.rankLevels()
plants.genera[:3]
[<LocalTree | Genus: Tillandsia - n.count: 3- | AF: 0.2>,
 <LocalTree | Genus: Lonchocarpus - n.count: 5- | AF: 0.18>,
 <LocalTree | Genus: Eugenia - n.count: 1- | AF: 0.15>]
```

Table 2: Output for environmental variables

	MinTemperature	Precipitation	Vapor	SolarRadiation	WindSpeed
0	22.25	21.16	1.33	16,466.25	2.33

Here showing only mean values for some variables on a single record.

Associated raster (environmental) information

Here, we demonstrate how to access raster data associated with a taxonomic tree `TreeNeo`. The raster data used are related to environmental variables stored in the RGU. Currently there are 2 forms for accessing this information: (i) as a table with columns corresponding to environmental variables and rows defined by each occurrence (a point-based method) and (ii) as a raster object sampled from the associated geometry of each tree or, in general, any (multi) polygon object. The raster object features methods for visualization, geoprocessing, and data exchange.

Extracting raster information as table

To extract the data in this format use the method (function):

`TreeNeo.associatedData.getEnvironmentalVariablesPoints()`

The output is a Pandas dataframe with the associated values of climatic covariates. See the following example:

```
table = vertebrates.associatedData.getEnvironmentalVariablesPoints()
print(table[:1])
```

Here we only show the first record.

The geometric object of each tree is determined by the `Occurrence` nodes of the tree. In the graph database, each `Occurrence` node is linked to the `Cell` node that geographically contains the occurrence's location. One of the attributes of the `Cell` object is the geographic polygon that defines its border. The union of all the corresponding `Cell` nodes is what determines the geometric feature of the tree `TreeNeo`. As such, the raster extraction process is performed on each of the tree's associated cells.

Extracting raster objects from TreeNeo instances

To extract the associated raster object of a `TreeNeo` instance use the following method (function):

`TreeNeo.associatedData.getAssociatedRasterAreaData([name of variable])`

To obtain several environmental variables use `associatedData.getEnvironmentalVariablesCells()`

For example, information for a single variable can be obtained with

```
meantemp_data = vertebrates.associatedData.
                            getAssociatedRasterAreaData(
                            'MeanTemperature')
```

The raster object is automatically added to the `TreeNeo` object after the method is called. The raster objects are appended to the attribute `associatedData`.

Extracting raster objects from arbitrary polygons

The extraction of raster objects is performed by the `raster_api` library, a Biospytial module for reading, writing, and processing raster objects using the RGU as back end.

The `raster_api` can use natively any object stored in the knowledge engine that has at least a 2D geometric feature (attribute). This includes the basic operations for querying, reading, and writing. For using external geometric objects such as Shapefiles, GeoPackages, or GeoJSON, the objects need to be transformed to their corresponding WKT or WKB (Well Known Binary) representation. Examples of these are described extensively in the Jupyter notebooks and in the documentation.

In this example we use the polygon defined by the border of Mexico to extract several raster objects (RasterData instances) using the raster_api module. We use these objects to compare the environmental ranges of the threatened species, the jaguars' habitat, and the entire area of the country to conclude whether the environmental niches of the threated species are covered by the habitat of the jaguars and how these ranges are different with respect to the whole country.

Importing the polygon for Mexico

The first step in this is to import the polygon for Mexico. The default installation of Biospytial includes the WorldBorders dataset [100]. Assuming that this dataset is installed, we can import the polygon of Mexico with the API provided by the class `Country` located in `sketches.models`. Country is a vector dataset stored in the RDBMS. The geometric feature is stored as the `geom` column.

```
from sketches.models import Country
## The syntax follows the Django Query Set API
mexico = Country.objects.filter(name='Mexico').first()
mex_area = mexico.geom.area

## For reprojecting the area of Mexico we similarly do:
mex_shapely = wkt.loads(mexico.geom.wkt)
mex_projected= transform(project,mex_shapely)
```

To calculate the percentage of area covered by all the cells with respect with the total area of Mexico we can do

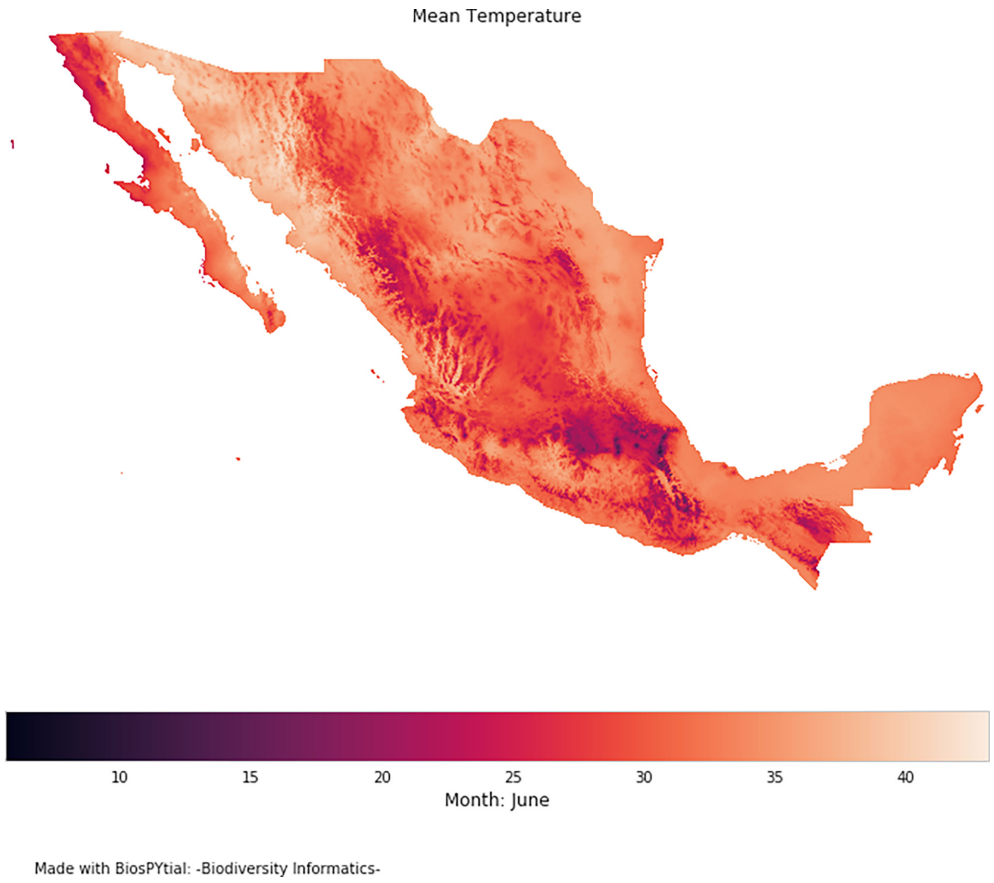


Figure 7: The output of the method `display_field()`, an easy way to visualize RasterData objects.

```
total_cell_area / mex_projected.area * 100
3.42%
```

For example, we can display simple visualizations invoking the method `display_field()`. See Fig. 7.

```
vertebrates.associatedData.raster_MeanTemperature.display_field()
```

Interactive visualization

As an alternative, we can export the raster object as an xarray [101] instance for interactive visualization using the Geoviews [102] package. To export the associated raster data to an xarray object do the following:

```
meantemp = vertebrates.associatedData.raster_MeanTemperature.to_xarray()
```

The following code gives an example of how to generate an interactive visualization using the vertebrates' associated mean temperature data and the locations of the observed threatened species associated with the presence of jaguars. We used the elevation data for Mexico (extracted before) as base map. Fig. 8 shows this visualization at 2 different scales.

```
import geoviews as gv
from cartopy import crs
import geoviews.feature as gf
from geoviews import opts
gv.extension('bokeh')

sample_pt = gv.Points((env_threated_occurrences.x,env_threated_occurrences.y),
            label='ocurrences').opts(
            fill_color = 'orange',
            line_color = 'black',
            line_width = 0.5,
            line_alpha = 0.4,
            fill_alpha = 1.0,
            size = 5,
            )

elevation = all_mex_datasets[0].to_xarray()
```

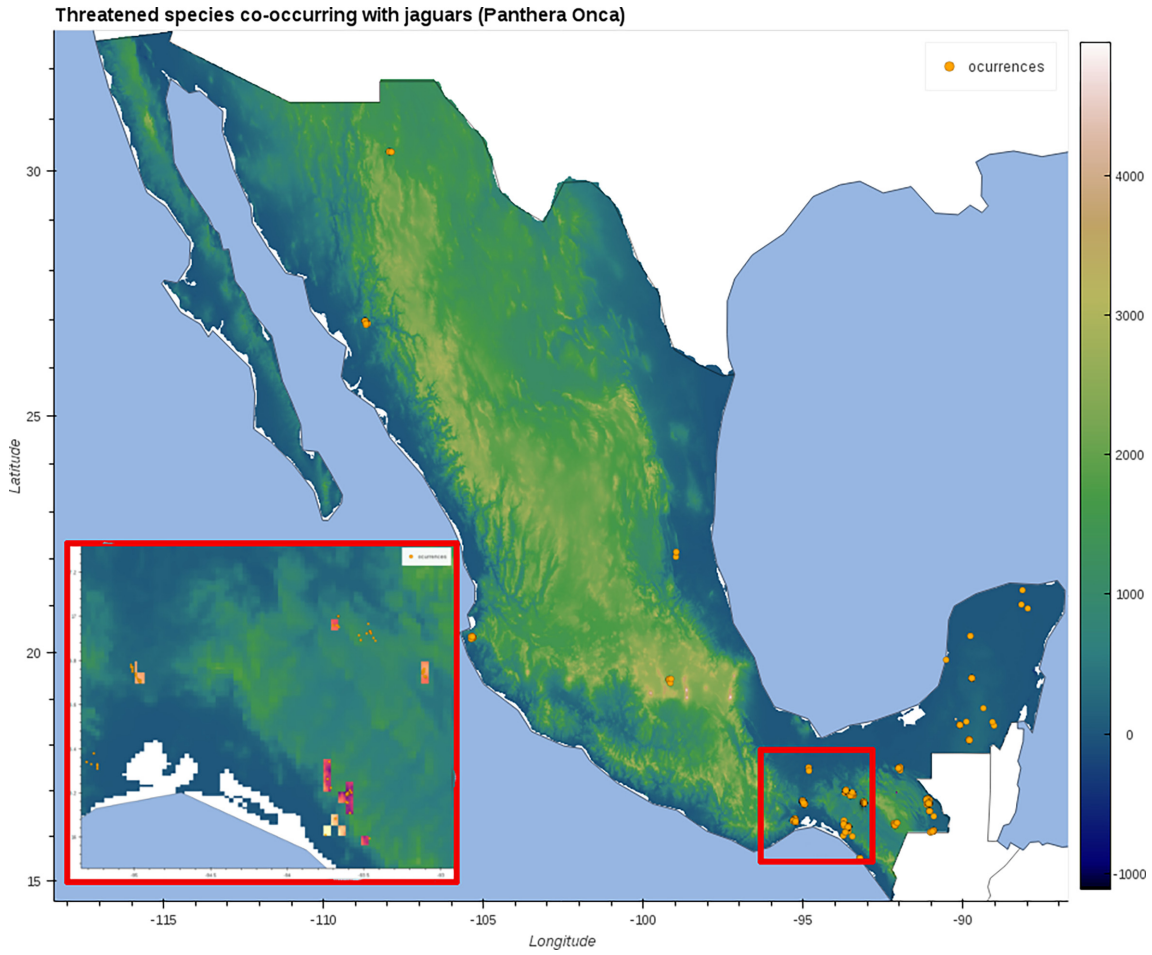


Figure 8: A composite figure showing 2 states of the interactive visualization. Orange dots represent occurrences of threated species associated with the presence of jaguars (*P. onca*). The inset shows the area inside the red square in the main map. The colored squares in the inset show the mean temperature associated with threatened vertebrates (phylum Chordata). The base map shows the elevation for all of Mexico. See section: "Data Used" for more information.

```
elevds = gv.Dataset(elevation,crs=crs.PlateCarree())
elevimg = gvds.to(gv.Image,['Longitude','Latitude']
                  ).opts(cmap=plt.cm.gist_earth)

temp = meantemp.where(((meantemp.Longitude > -95) &
                       (meantemp.Longitude < -89) &
                       (meantemp.Latitude > 15) &
                       (meantemp.Latitude < 19)),
                       drop=True)

temp.name = meantemp.name
tempds = gv.Dataset(temp,crs=crs.PlateCarree())
tempimg = tempds.to(gv.Image,['Longitude','Latitude']).opts(cmap=plt.cm.magma)
## Display the map
map_ = (elevimg * gf.ocean * gf.coastline * gf.borders * tempimg * sample_pt )
```

Network visualization and analysis

Each tree instance induces an acyclic graph. We can convert the tree into a networkx object to visualize and analyse its network properties. To do this, we simply need to use the method `tree.toNetworkx(depth_level=[k])`, where *k* is the taxonomic level to reach in the tree, 0 for root 7 for species level.

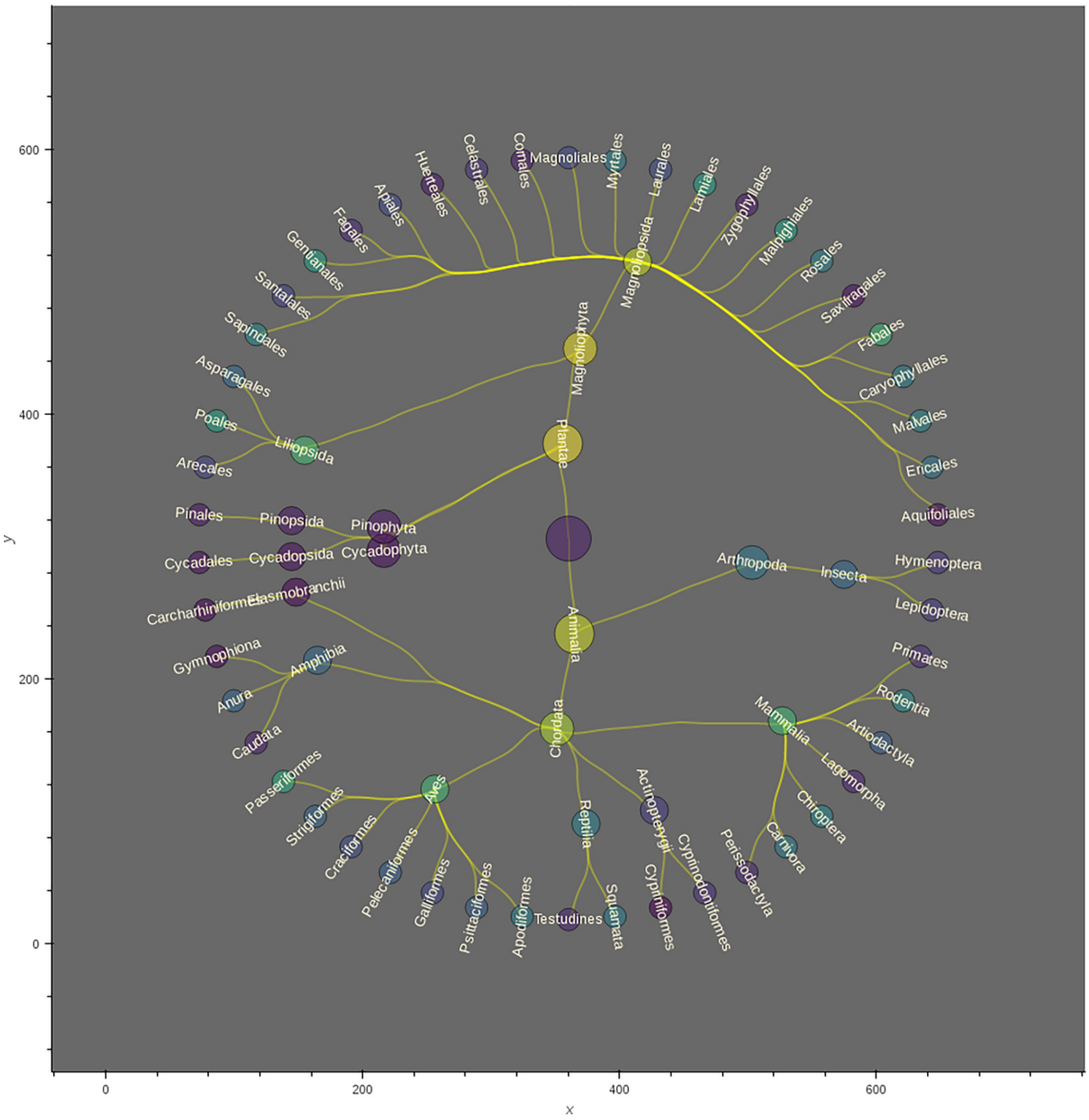


Figure 9: A tree visualization for the merged tree corresponding to threatened taxa, showing up to order level. The size of the nodes is proportional to the taxonomic level (the largest is the root of the tree, the smallest are orders). The node colouring indicates the frequency of occurrence with respect to all the neighbouring cells (neighbours of jaguars), ranging from brighter to darker for higher and lower ranked, respectively.

Visualization

A method for interactive visualization has been developed using the Holoviews [103] framework. To do this we need to invoke the following method:

```
## Plot the Tree
from drivers.tools import to_interactivePlot
network = to_interactivePlot(threatened_tree,label_depth=8)
```

The output is a dictionary with 2 key items: 1 for labels and the other for the actual graph (nodes and edges). To plot the whole graph we need to overlay both items.

```
network['labels'] * network['graph']
```

Analysis with standard graph algorithms

The `TreeNeo` structures are particular cases of graph traversals. As such, they can be analysed with graph theoretic methods. The library NetworkX [104] is a Python package designed for analysing the structure, dynamics, and functions of complex networks. It includes standard graph algorithms and analysis measures as well as tools for import and export to other standard formats. We can convert a `TreeNeo` using the method `toNetworkx(depth_level )`, where depth_level is the depth of the graph to be generated. In the next example we convert the threatened_tree to a `NetworkX` object and use this to calculate its corresponding adjacency matrix.

```
threatened_graph = threatened_tree.toNetworkx(depth_level=7)
```

Table 3: Corresponding URLs for source code and container images for the Biospytial engine.

Module name	URL
Graph Storage and Processing Unit	https://hub.docker.com/r/molgor/postgis_biospytial
Biospytial Computing Engine	https://hub.docker.com/r/molgor/biospytial
Relational Geoprocessing Unit	https://hub.docker.com/r/molgor/neo4j_biospytial
Source code	https://github.com/molgor/biospytial
Data	http://dx.doi.org/10.5524/100723

The modules and the source code do not include data. These should be installed separately or loaded independently.

```
from networkx import adjacency_matrix
M = adjacency_matrix(threatened_graph)
# uncomment this to plot the matrix
#plt.imshow(M.todense())
```

Representing `TreeNeo` objects into NetworkX graphs brings new possibilities for analysis and modelling. We hope this example will awaken the spirit of the reader to explore the potential of representing data as complex graph structures.

Conclusions

Biospytial uses open source standards to integrate geospatial ecological Big Data as a tool for ecological niche modelling and the analysis of species distributions. This integration creates a complex network of data with enormous potential for data mining, information retrieval, and visualization. At the core, a web of semantic-wise relationships constitutes a corpus of taxonomic and environmental knowledge that opens up new ways to query and unveil complex ecological relations. To our knowledge, there is no other open source system with the design and capacity to achieve this including (i) storing information in a hybrid relational-graph system and (ii) performing geospatial processes in vector and raster scalable databases.

A practical example provided a glimpse into how to query and manipulate taxonomic tree structures, as well as how to extract data, conduct frequency analysi,s and visualize results. The example demonstrated a new procedure to rank co-occurring taxonomic groups in an arbitrary size neighbourhood of pixels.

The GBIF occurrence data include information only on location and taxonomy, and in this sense the data are limited. However, the engine's design allows the capture, extension, and exploration of a semantic interpretation of the data by adding other types of relations. For example, linking information on trophic networks to the taxonomic backbone can help in analysing spatial patterns of trophic groups and dependant species, a key question in conservation biology.

The development of Biospytial has followed best practices in scientific programming [105]. We recognize that spatial analyses are often not generalizable and therefore replicable. However replicability and reproducibility can be enhanced by increasing openness and documentation transparency and completeness [106–108]. In fact, Biospytial's source code is open and can be accessed at [109] while this article is Open Access. In the future, Biospytial can be further developed into a system not only for integration and distribution of datasets but also as a tool for collaboration, experimentation, validation, and reproduction of results in the era of Open Science, satisfying also the requisites of second-generation SDI.

Availability of Supporting Source Code and Requirements

- Project name: Biospytial
- Project home page: https://github.com/molgor/biospytial
- Operating System(s): Platform independent (not tested in Windows)
- Other requirements: Docker 1.13 or higher
- License: GNU General Public License version 3.0 (GPLv3)
- Memory requirements: 40GB in HD for installing the database and ≥16 GB in RAM for running the example.
- RRID:SCR_018226
- biotools:biospytial

The current example is located inside the folder `examples` with the name `[Official Demo] Co-occurrences_Jaguar.ipynb`. The example has been modified only in the neighbourhood order, changing from 4 to 1. This modification reduces the data to process and the executing time.

Availability of Supporting Data and Materials

Snapshots of our code and other supporting data are openly available in the *GigaScience* repository, GigaDB [110]. The container images can be downloaded automatically using the script `installEngine.sh`. Instructions for installing and running the engine are located in the project's home page.

Additional Files

"Jupyter notebook for the tutorial section": [Official Demo]Co-ocurrences_Jaguar.ipynb

"Supplementary materials I": Adding data in Biospytial (pdf file)
"Supplementary materials II": Mathematical formalisms (pdf file)

Abbreviations

ACID: atomicity, consistency, isolation, durability; API: application programming interface; BCE: Biospytial Computing Engine; BLOB: binary large object; CONABIO: National Commission for the Knowledge and Use of Biodiversity; CRS: coordinate reference system; CSV: comma separated value; DAG: directed acyclic graph; DEM: digital elevation model; EBV: essential biodiversity variable; EPSG: European Petroleum Survey Group; ESA: European Space Agency; GBIF: Global Biodiversity Information Facility; GDAL: Geospatial Data Abstraction Software Library; GIS: Geographic Information Systems; GSPU: Graph Storage and Processing Unit; MPI: Message Passing Interface; NASA: National Aeronautics and Space Administration; OGM: object-graph mapping; ORM: object-relational mapping; RDBMS: Relational Database Management System; RGU: Relational Geoprocessing Unit; SDI: spatial data infrastructure; ToL: Tree of Life; WKB: Well Known Binary; WKT: Well Known Text.

Competing Interests

The authors declare that they have no competing interests.

Funding

J.M.E.M and the project were jointly sponsored by the Doctoral Scholarships Program from the Mexican Science and Technology Council (CONACYT, Becas al Extranjero), the Faculty of Science and Technology from Lancaster University (FST-LU) and the GBIF Consortium through the GBIF Young Researchers Award (2016). L.S and P.M.A are supported by the Engineering and Physical Sciences Research Council grant number EP/R01860X/1 "Data Science of the Natural Environment"

Authors' Contributions

J.M.E.M. and P.M.A. conceived the original idea, which was further refined by all authors. The semantic structures and graph traversals were designed by J.M.E.M. with the mentorship of L.S. for integrating datasets. The software and system's design was developed by J.M.E.M. under the supervison of P.M.A. and L.S. The writing of the original draft was done by J.M.E.M. with reviewing and editing from P.M.A. and L.S.

Acknowledgments

We thank the many researchers, students, public servants, and citizen scientists who contributed to sample, register, and curate all the biodiversity occurrences data contained in the GBIF database. We especially thank Raúl Jiménez Rosenberg from CONABIO for facilitating a complete snapshot of the GBIF database (2016) and the Free and Open Source Software community whose effort in developing software made possible the creation of this software.

References

1. Reinsel D, Gantz J, Rydning J. The Digitization of the World - From Edge to Core. IDC White Paper No. US44413318. 2018, International Data Comporation (IDC), Framingham, MA 01701 USA. . https://www.seagate.com/files/www-content/our-story/trends/files/idc-seagate-dataage-whitepaper.pdf. . Accessed data: 02/05/2020.
2. Kurzweil R. The law of accelerating returns. In: Teuscher C , ed. Alan Turing: Life and Legacy of a Great Thinker. Berlin, Heidelberg: Springer; 2004:381–416.
3. Hilbert M, López P. The world's technological capacity to store, communicate, and compute information. Science 2011;**332**(6025):60–5.
4. Gantz J, Reinsel D. Extracting Value from Chaos. 2011, IDC Digital Universe Study , 1142, 1–12.
5. Weigelt A, Marquard E, Temperton VM, et al. The Jena Experiment: six years of data from a grassland biodiversity experiment. Ecology 2010;**91**:930–1.
6. Borer ET, Harpole WS, Adler PB, et al. Finding generality in ecology: A model for globally distributed experiments. Methods Ecol Evol 2014;**5**(1):65–73.
7. National Aeronautics and Space Administration, Joint Polar Satellite System, Tecnical Documents. 2020, . https://www.jpss.noaa.gov/technical_documents.html . Accessed date: 02/05/2020.
8. European Space Agency. Copernicus. 2014. hhttps://www.esa.int/Our_Activities/Observing_the_Earth/Copernicus/Overview3.
9. Goodchild MF. Citizens as sensors: the world of volunteered geography. GeoJournal 2007;**69**(4):211–21.
10. Heipke C. Crowdsourcing geospatial data. ISPRS J Photogramm Remote Sens 2010;**65**(6):550–7.
11. Kamel Boulos MN, Resch B, Crowley DN, et al. Crowdsourcing, citizen sensing and sensor web technologies for public and environmental health surveillance and crisis management: trends, OGC standards and application examples. Int J Health Geog 2011;**10**(1):67.
12. OpenStreetMap Contributors. OpenStreetMap (OSM). 2019. https://www.openstreetmap.org. Accessed date: 03/12/2019.
13. GBIF Secretariat, Global Biodiversity Infrastructure. GBIF Consortium. 2015. https://www.gbif.org/the-gbif-network. Accessed date: 03/05/2020.

14. Chen M, Mao S, Liu Y. Big data: A survey. In: Mobile Networks and Applications, vol. **19**. Berlin, Heidelberg: Springer; 2014:171–209.
15. Mikalef P, Pappas IO, Krogstie J, et al. Big data analytics capabilities: a systematic literature review and research agenda. Inf Syst e-Bus Manag 2018;**16**:547–78.
16. Li S, Dragicevic S, Castro FA, et al. Geospatial big data handling theory and methods: A review and research challenges. ISPRS J Photogramm Remote Sens 2016;**115**:119–33.
17. Stocker TF, Qin D, Plattner GK, et al. (IPCC) Climate Change 2013: The physical science basis, Intergovernmental Panel on Climate Change. . 2013.
18. Brondizio ES, Settele J, Díaz S, et al. IPBES. 2019 Global assessment report on biodiversity and ecosystem services of the Intergovernmental Science- Policy Platform on Biodiversity and Ecosystem Services. Bonn, Germany; 2019.
19. Loreau M. Linking biodiversity and ecosystems: towards a unifying ecological theory. Philos Trans R Soc Lond B Biol Sci 2010;**365**(1537):49–60.
20. Pavoine S, Bonsall MB. Measuring biodiversity to explain community assembly: a unified approach. Biol Rev Camb Philos Soc 2011;**86**(4):792–812.
21. Koricheva J, Gurevitch J, Mengersen KL. Handbook of Meta-analysis in Ecology and Evolution. Princeton University Press; 2013.
22. Wiemann S, Bernard L. Spatial data fusion in spatial data infrastructures using linked Data. Int J Geog Inf Sci 2016;**30**(4):613–36.
23. Wang JF, Zhang TL, Fu BJ. A measure of spatial stratified heterogeneity. Ecol Indic 2016;**67**:250–6.
24. Pereira HM, Leadley PW, Proença V, et al. Scenarios for global biodiversity in the 21st century. Science 2010;**330**(6010):1496–501.
25. Navarro LM, Fernández N, Guerra C, et al. Monitoring biodiversity change through effective global coordination. Curr Opin Environ Sustain 2017;**29**:158–69.
26. Pereira HM, Ferrier S, Walters M, et al. Essential biodiversity variables. Science 2013;**339**(6117):277–8.
27. Schmeller DS, Mihoub JB, Bowser A, et al. An operational definition of essential biodiversity variables. Biodivers Conserv 2017;**26**(12):2967–72.
28. Kissling WD, Ahumada JA, Bowser A, et al. Building essential biodiversity variables (EBVs) of species distribution and abundance at a global scale. Biol Rev 2018;**93**(1):600–25.
29. Sullivan BL, Wood CL, Iliff MJ, et al. eBird: A citizen-based bird observation network in the biological sciences. Biol Conserv 2009:**144**:2282–92.
30. Kattge J, Diaz S, Lavorel S, et al. TRY–a global database of plant traits. Global Change Biol 2011;**17**(9):2905–35.
31. Hudson LN, Newbold T, Contu S, et al. The PREDICTS database: A global database of how local terrestrial biodiversity responds to human impacts. Ecol Evol 2014;**4**(24):4701–35.
32. Enquist BJ, Condit RR, Peet RK, et al. Cyberinfrastructure for an integrated botanical information network to investigate the ecological impacts of global climate change on plant biodiversity. PeerJ 2016. Retrieved from: https://peerj.com/preprints/2615/.
33. Hartig F, Dyke J, Hickler T, et al. Connecting dynamic vegetation models to data - an inverse perspective. J Biogeog 2012;**39**(12):2240–52.
34. Kelling S, Fink D, La Sorte FA, et al. Taking a 'Big Data' approach to data quality in a citizen science project. Ambio 2015;**44**(Suppl 4):601–11.
35. La Salle J, Williams KJ, Moritz C. Biodiversity analysis in the digital era. Philos Trans R Soc Lond B Biol Sci 2016;**371**, doi:10.1098/rstb.2015.0337.
36. Scheiter S, Langan L, Higgins SI. Next-generation dynamic global vegetation models: Learning from community ecology. New Phytol 2013;**198**(3):957–69.
37. Ramsey P, Santilli S, Obe R, et al. PostGIS. 2018. http://www.postgis.org/. Accessed date: 03/05/2020.
38. GDAL/OGR Contributors. GDAL/OGR - Geospatial Data Abstraction software Library. 2018. https://www.gdal.org/. Accessed date: 03/05/2020.
39. Geometry Engine Open Source (Contributors). Geometry Engine Open Source. 2019. https://trac.osgeo.org/geos. Accessed date: 03/05/2020.
40. PROJ Contributors. PROJ coordinate transformation software library. 2019. https://proj4.org/. Accessed date: 03/05/2020.
41. Harrington JL. Relational Database Design and Implementation, Fourth. 2009, Morgan Kauffman . https://www.sciencedirect.com/book/9780128043998/relational-database-design-and-implementation.
42. Altinel M, Altinel M, Luo Q, et al. Dbcache: Database caching for web application servers, Proceedings of the ACM SIGMOD Conference. 2002;**2002**:612, .
43. Celko J. Joe Celko's Complete Guide to NoSQL. 2014:27–46, doi:10.1016/B978-0-12-407192-6.00003-0.
44. Vicknair C, Macias M, Zhao Z, et al. A comparison of a graph database and a relational database. In: Proceedings of the 48th Annual Southeast Regional Conference - ACM SE '10 New York. New York, NY: ACM; 2010, doi:10.1145/1900008.1900067:.
45. Grund M, Cudre-Mauroux P, Krueger J, et al. Hybrid graph and relational query processing in main memory. In: Proceedings - International Conference on Data Engineering. 2013:23–4.
46. van Iersel MP, Pico AR, Kelder T, et al. The BridgeDb framework: Standardized access to gene, protein and metabolite identifier mapping services. BMC Bioinformatics 2010;**11**, doi:10.1186/1471-2105-11-5.
47. Fabregat A, Korninger F, Viteri G, et al. Reactome graph database: Efficient access to complex pathway data. PLoS Comput Biol 2018;**14**(1):e1005968.
48. Hendriks PHJ, Dessers E, van Hootegem G. Reconsidering the definition of a spatial data infrastructure. Int J Geog Inf Sci 2012;**26**(8):1479–94.
49. INSPIRE: Infrastructure for spatial information in Europe. https://inspire.ec.europa.eu/. Accessed date: 03/05/2020
50. GBIF Secretariat. GBIF Backbone Taxonomy. 2017. https://www.gbif.org/dataset/d7dddbf4-2cf0-4f39-9b2a-bb099caae36c, doi:10.15468/39omei.

51. Rodriguez MA. The Gremlin Graph Traversal Machine and Language. In: Proc 15th Symposium on Database Programming Languages. 2015, doi:10.1145/2815072.2815073.
52. Juneau J. Object-Relational Mapping. In: Java EE 8 Recipes . Berkeley, CA: Apress; 2018:395–439. http://link.springer.com/10.1007/978-1-4842-3594-2_8.
53. Docker. Enterprise Application Container Platform | Docker. 2019. https://www.docker.com/. Accessed date: 03/05/2020.
54. Pahl C, Lee B. Containers and clusters for edge cloud architectures-A technology review. In: Proceedings - 2015 International Conference on Future Internet of Things and Cloud; 2015:379–86.
55. Biospytial's Postgis container. https://hub.docker.com/r/molgor/postgis_biospytial/. Accessed date: 03/05/2020.
56. Neo4j Spatial v0.24-neo4j-3.1.4. https://neo4j-contrib.github.io/spatial/0.24-neo4j-3.1/index.html. Accessed date: 03/05/2020.
57. APOC User Guide 3.1.3.9. https://neo4j-contrib.github.io/neo4j-apoc-procedures/index31.html. Accessed date: 03/05/2020.
58. Biospytial's Neo4J container. https://hub.docker.com/r/molgor/neo4j_biospytial. Accessed date: 03/05/2020.
59. Anaconda, vers. 2-2.4.0, Anaconda Software Distribution. 2016. https://repo.anaconda.com/archive/. Accessed date: 03/05/2020.
60. Biospytial. https://github.com/molgor/biospytial. Accessed date: 03/05/2020.
61. Diggle PJ, Tawn JA, Moyeed RA. Model-based geostatistics. J R Stat Soc Ser C Appl Stat 2002;**47**(3):299–350.
62. Biospytial main container. https://hub.docker.com/r/molgor/biospytial/. Accessed date: 03/05/2020.
63. Redis, an in-memory data structure store. http://redis.io/. Accessed date: 03/05/2020.
64. R Development Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing. 2016. https://www.r-project.org. Accessed date: 03/05/2020.
65. Hornik K. The Comprehensive R Archive Network. 2012. https://cran.r-project.org. Accessed date: 03/05/2020, .
66. Wilson G, Aruliah DA, Brown CT, et al. Best practices for scientific computing. PLoS Biol 2014;**12**(1):e1001745.
67. Perkel JM. A toolkit for data transparency takes shape. Nature 2018;**560**(7719):513–5.
68. Perez S, Jandl R, Rubio A. Modelización del secuestro de carbono en sistemas forestales: Efecto de la elección de especie. Ecología 2007;**21**:341–52.
69. Kluyver T, Ragan-Kelley B, Pérez F, et al. Jupyter Notebooks – a publishing format for reproducible computational workflows. In: Positioning and Power in Academic Publishing: Players, Agents and Agendas. 2016:87–90.
70. Django, Version 1.10. Lawrence, KS. 2018. https://djangoproject.com. Accessed date: 03/05/2020.
71. Neo4J, Graph Database . https://neo4j.com. Accessed date: 03/05/2020.
72. Small NT. py2neo. 2017. https://py2neo.org/v3/index. Accessed date: 03/05/2020.
73. Besag J. Spatial Interaction and the Statistical Analysis of Lattice Systems. J R Stat Soc Ser B Methodol 1974;**36**(2):192–236.
74. Besag J, York J, Mollié A. Bayesian image restoration, with two applications in spatial statistics. Ann Inst Stat Math 1991;**43**(1):1–20.
75. Rue H, Held L. Gaussian Markov Random Fields: Theory and Applications. Chapman & Hall/CRC; 2005. https://www.crcpress.com/Gaussian-Markov-Random-Fields-Theory-and-Applications/Rue-Held/p/book/9781584884323.
76. Hagberg AA, Schult DA, Swart PJ. Exploring network structure, dynamics, and function using NetworkX. In: Varoquaux G, Vaught T, Millman J , eds. Proceedings of the 7th Python in Science conference (SciPy 2008). 2008:11–15. http://conference.scipy.org/proceedings/SciPy2008/paper_2/.
77. Seabold S, Perktold J. Statsmodels: Econometric and statistical modeling with Python. In: Proceedings of the 9th Python in Science Conference. 2010; http://conference.scipy.org/proceedings/scipy2010/pdfs/seabold.pdf.
78. Salvatier J, Wiecki TV, Fonnesbeck C. Probabilistic programming in Python using PyMC3. PeerJ Comput Sci 2016, doi.org/10.7717/peerj-cs.55.
79. Hudak P. Conception, evolution, and application of functional programming languages. ACM Comput Surv 1989;**21**(3):359–411.
80. UNEP/CBD. Cancun declaration of like-minded megadiversity countries. In: United Nations Environmental Program-Convention on Biological Diversity (UNEP-CBD). The Hague, Netherlands; 2002: UNEP/CBD/COP/6/INF/33.
81. UNEP/CBD. Like-minded mega-diverse countries carta to achieve Aichi biodiversity Target 11. In: United Nations Environmental Program-Convention on Biological Diversity (UNEP-CBD) Cancún, México; 2016: UNEP/CBD/COP/13/INF/45. https://www.cbd.int/doc/meetings/cop/cop-13/information/cop-13-inf-45-en.pdf.
82. Vidal -Zepeda R. Las regiones climaticas de Mexico. Instituto de Geografia, UNAM, 1.2.2; 2005. http://www.publicaciones.igg.unam.mx/index.php/ig/catalog/book/42.
83. Rzedowski J. Vegetacio´n de Me´xico . Primera edicion digital, Comisio´n Nacional para el Conocimiento y Uso de la Biodiversidad; 2006. https://www.biodiversidad.gob.mx/publicaciones/librosDig/pdf/VegetacionMx_Cont.pdf.
84. Sarukhán J, Koleff P, Carabias J, et al. Capital Natural de Mexico. Síntesis: Conocimiento actual y perspectivas de sustentabilidad. Comisión Nacional para el Conocimiento y Uso de la Biodiversidad, México; 2009.
85. Amante C, Eakins BW. ETOPO1 1 Arc-Minute Global Relief Model: Procedures, Data Sources and Analysis, NOAA Technical Memorandum NESDIS NGDC-24, NOAA, National Geophysical Data Center Marine Geology and Geophysics Division. 2009, https://www.ngdc.noaa.gov/mgg/global/relief/ETOPO1/docs/ETOPO1.pdf.
86. Fick SE, Hijmans RJ. Worldclim 2: New 1-km spatial resolution climate surfaces for global land areas. Int J Climatol 2017, doi:10.1002/joc.5086.
87. Egenhofer MJ, Franzosa RD. Point-set topological spatial relations. Int J Geog Inf Syst 1991;**5**(2):161–74.
88. Clementini E, Felice P, Oosterom P. A Small Set of Formal Topological Relationships Suitable for End-User Interaction. Berlin, Heidelberg: Springer; 1993:277–95.
89. Herrig JR. Simple Feature Access - Part 1: Common Architecture | OGC. Open Geospatial Consortium Inc.; 2011.
90. , Haklay M, Kemp K . Open Source Geospatial Foundation (OSGF). In: Encyclopedia of Geographic Information Science, SAGE Publications, Thousand Oaks, California. 2008.
91. Andelman SJ, Fagan WF. Umbrellas and flagships: Efficient conservation surrogates or expensive mistakes? Proc Natl Acad Sci U S A 2000;**97**(11):5954–9.

92. Drever CR, Hutchison C, Drever MC, et al. Conservation through co-occurrence: Woodland caribou as a focal species for boreal biodiversity. Biol Conserv 2019;**232**:238–52.
93. Thornton D, Zeller K, Rondinini C, et al. Assessing the umbrella value of a range-wide conservation network for jaguars (*Panthera onca*). Ecol Appl 2016;**26**(4):1112–24.
94. de la Torre JA, Núñez JM, Medellín RA. Spatial requirements of jaguars and pumas in Southern Mexico. Mammal Biol 2017;**84**:52–60.
95. IUCN. The IUCN Red List of Threatened Species. Version 2013.2. 2019. http://www.iucnredlist.org.
96. Psitaciformes, Encyclopedia of Life. https://eol.org/pages/1590. Accessed date: 03/05/2020.
97. Seaborn: statistical data visualization. https://seaborn.pydata.org. Accessed date: 03/05/2020.
98. Lambda syntax, Python documentation. https://docs.python.org/3/reference/expressions.html#lambdasyntax. Accessed date: 03/05/2020.
99. Whittaker RH. Evolution and measurement of species diversity. Taxon 1972;**21**(2/3):213.
100. World borders maps: https://thematicmapping.org/downloads/world_borders.php. Accessed date: 03/05/2020.
101. xarray: N-D labeled arrays and datasets in Python. http://xarray.pydata.org. Accessed date: 03/05/2020, .
102. GeoViews. http://geoviews.org. Accessed date: 03/05/2020.
103. HoloViews. https://holoviews.org. date: 03/05/2020.
104. Networkx: network analysis in Python . https://networkx.github.io/. Accessed date: 03/05/2020.
105. Wilson G, Aruliah DA, Brown CT, et al. Best practices for scientific computing. PLoS Biol 2014;**12**(1):e1001745.
106. Barba LA. Praxis of reproducible computational science. Comput Sci Eng 2019;**21**(1):73–78.
107. Teytelman L. No more excuses for non-reproducible methods. Nature 2018;**560**(7719):411.
108. Shannon J, Walker K. Opening GIScience: A process-based approach. Int J Geog Inf Sci 2018;**32**(10):1911–26.
109. Biospytial, project's repository:. https://github.com/molgor/biospytial.git. Accesssed date: 03/05/2020.
110. Escamilla Molgora JM, Sedda L, Atkinson PM. Supporting data for "Biospytial: spatial graph-based computing engine for ecological Big Data." GigaScience Database 2020. http://dx.doi.org/10.5524/100723.
111 Mayr E. Speciation phenomena in birds. Am Nat 1940;**74**(752, 249–278).
112 Dobzhansky T, Dobzhansky TG. Genetics of the Evolutionary Process. Columbia University Press; 1970.
113 Mayr E, Ashlock PD. Principles of Systematic Zoology. McGraw-Hill; 1991.
114 Blackwelder RE. Taxonomy: a Text and Reference Book. Wiley; 1967.
115 Skornyakov LA. Partially ordered set. In: Encyclopedia of Mathematics. 2014, Springer and the European Mathematical Society. http://www.encyclopediaofmath.org/index.php?title=Partially_ordered_set&oldid=33633.