Natural Environment Valuation Online Tool

Technical Documentation

Version 1.0, June 2019

Chapter 5b: Biodiversity Emulation

Land, Environment, Economics and Policy (LEEP) Institute

University of Exeter

Corresponding authors

Nathan Owen n.e.owen@exeter.ac.uk

Authors

Brett Day

Amy Binner

Ian Bateman

Greg Smith

Patrick Collings

Louis Haddrell

Lorena Liuzzo

Carlo Fezzi

Collaborators

Forestry Commission / Forest Research

UCL

JNCC

University of Aberdeen

Introduction

The report on biodiversity in NEVO outlined how JNCC have implemented a modelling framework for 100 priority species in Great Britain. The framework uses an ensemble of species distribution models to link the presence of a species to environmental variables such as climate, topology, soil and land use. This report describes why it is necessary in NEVO to have a fast-running version of the JNCC model, known as a statistical emulator, and outlines the methodology for building such an emulator.

In the NEVO tool, we wish to use the JNCC models to investigate the impact of land cover change on biodiversity, measured using species richness. If a user makes a change to the land cover in a 2km grid cell or a region, we need to re-run the JNCC model for this configuration behind the scenes, before returning the output to the user on the NEVO tool. However, re-running the JNCC models in this way poses a number of issues. Firstly, the JNCC models are slow to run and in some cases the users may have to wait some time before getting results returned to them. NEVO has been designed to be a fast-running decision support tool so this is not ideal. For example, running the species distribution models for all 100 species for Great Britain in a single year takes approximately 2 hours. In NEVO we present results across a 40 year future period, meaning that the computational burden becomes very expensive. Secondly, a more technical issue is that the JNCC models run in the statistical programming language R and rely on a large number of external packages which would have to be installed on the Linux server architecture of NEVO. Furthermore, the models themselves are stored in large data files – approximately 250GB in total – which again would have to be saved on the server. Therefore, for the purposes of the NEVO tool we would benefit from replacing the JNCC models with a fast-running alternative, known as a statistical emulator. This report outlines our methodology for building such an emulator.

The key concept of emulation is that a slow-running model is replaced with an alternative model which is orders of magnitude faster, for a reduction in the prediction accuracy. However, the loss in prediction accuracy can be controlled with enough information about the model itself, and the associated uncertainty can be quantified (Smith, 2013). There are many types of emulator, ranging from simple linear regression models to more complex formulations such as Gaussian process emulators (Rasmussen and Williams, 2006; O'Hagan, 2006) or polynomial chaos expansions (Ghanem and Spanos, 1990; Xiu and Karniadakis, 2002). In this work, we will keep it simple and use linear regression models.

JNCC Model: An Overview

In this section, we will give a short overview of the JNCC biodiversity modelling framework, focussing on aspects which are relevant for building a statistical emulator. More detailed information about the JNCC model can be found in the report on biodiversity in NEVO.

Data

The data used in building the JNCC models can be split into two categories: species presence data (the response variable in each model) and environmental data (used as explanatory variables in the models). All data is collected at, or has been processed to, 2km^2 grid cell resolution across Great Britain. There are 57,230 grid cells in Great Britain, a number which is reduced to 37,861 since currently NEVO is a tool for England & Wales.

In terms of the species data, JNCC selected a set of 100 species for inclusion in the modelling framework for NEVO. These species lie across 6 different taxonomic groups: mammals (14), bird (17), plants (38), invertebrates (25), lichens (5), and herptiles (1). The criteria for species selection can be found in the report on biodiversity in NEVO, along with the full list of species names (Table 1).

For each species in each cell, a binary variable is recorded to indicate whether the species is present (1) or absent (0). Based on this data, the JNCC modelling framework (to be summarised shortly) actually predicts the probability of occurrence for each species. This predicted probability then feeds through as data for building the emulators.

The environmental data includes variables on climate, topology, land use and soil, and comes from a variety of sources. There are 39 variables in total, the full list is documented in the report on biodiversity in NEVO.

Modelling framework

To link the species presence data to the environmental variables, JNCC use an ensemble of species distribution models: Bioclim, boosted regression trees, generalised linear model, generalised additive model, kernel support vector machine, Maxent, and random forest. A JNCC model for a single species is built as follows. Using 75% of the data for training and 25% for validation, the best of the above models is chosen according to the area under the Receiver Operating Characteristic Curve (AUC) statistic. This process is repeated 100 times using a different 75% of the data, selected randomly, with the best model saved for each iteration. Each of these 100 models provides a way of predicting the probability of occurrence for the species based on the data. The final model is an ensemble average of these 100 predictions of probability of occurrence.

Mathematically, the probability of occurrence for a species s, denoted p_s, is related to the 39 variables $\boldsymbol{x} = (x_1, \dots, x_{39})$ in the following way:

$$p_s = \frac{1}{100}\sum_{k=1}^{100} f_k(\boldsymbol{x})$$

where $f_k(\boldsymbol{x})$ denotes one of the 100 best models described above. For the purposes of building a statistical emulator, we simplify this relationship to:

$$p_s = f(\boldsymbol{x}) \tag{1}$$

i.e. we treat the JNCC model for a single species as a black box: given values for the 39 variables in $\boldsymbol{x}$, a prediction of the probability of occurrence for the species is returned.

Summary Statistics

In terms of the NEVO tool, it is not feasible to present outputs from the JNCC model for all 100 individual species. Instead, we choose to present some summary statistics. The main summary statistic presented is the expected species richness (ESR), i.e. the number of species present out of the total 100. This is approximated simply by summing the probability of occurrence across the 100 species:

$$ESR = \frac{1}{100}\sum_{s=1}^{100} p_s$$

We also present the expected species richness within each taxonomic group. Given a set of species in a taxonomic group T with $|T|$ species, this is computed as:

$$ESR_T = \frac{1}{|T|}\sum_{s \in T} p_s$$

Emulator Methodology

To reiterate, a statistical emulator is a fast-running approximation to a model. In terms of the JNCC modelling framework, we seek to find an approximation to the relationship in Equation (1), and we will denote this approximation as $p_s \approx \hat{f}(\boldsymbol{x})$. To aid the modelling of species probability of occurrence, we also apply the logit transformation such that the relationship is $\log(p_s/(1-p_s)) \approx \hat{f}(\boldsymbol{x})$. When predicting at a new value of the environmental, say $\boldsymbol{x}^*$, we invert this relationship to find the predicted species probability of occurrence: $p_s^* = 1/(1+\exp(-\hat{f}(\boldsymbol{x}^*)))$.

For the functional form of $\hat{f}(\boldsymbol{x})$, we choose a regression model including linear, quadratric and first order interactions in the 39 variables:

$$\hat{f}(\boldsymbol{x}) = \beta_0 + \sum_{i=1}^{39} \alpha_i x_i + \sum_{i<j} \sum_{j=1}^{39} \alpha_{ij} x_i x_j + \sum_{i=1}^{39} \beta_i x_i^2 \qquad (2)$$

This formulation is much simpler than the JNCC modelling framework, which takes an average of an ensemble of 7 machine learning approaches. However, it shares some likeness to some of the individual machine learning approaches (e.g. generalised linear model, generalised additive model), and it is flexible enough to mimic a wide range of behaviours. As we will show, it proves to be an accurate representation of the JNCC modelling framework. Moreover, it provides a simple relationship between the probability of occurrence of a species and the environmental variables, which can be evaluated very quickly in the NEVO tool.

Results

The emulator function in Equation (2) is estimated for each of the 100 species chosen by JNCC, using all available data for England & Wales at the 2km^2 grid cell scale. The coefficients are estimated using the lm function in the statistical programming language R (R Core Team, 2019).

Once an emulator has been built, it is important to validate it to test whether it is doing a good job of replicating the model. In this report, we do this by comparing the fitted values of the emulator to the corresponding values predicted by the JNCC model. In other words, we compare the predicted probability of occurrence for each species for the JNCC model and emulator at the baseline set of environmental data. In this way, we can compare the predictions of probability of occurrence spatially on a map of Great Britain.

In Figures 1, 2, and 3 we have plotted this comparison for the three of the 100 species: European hare, white-letter hairstreak butterfly, and string-of-sausage lichen respectively. We also include the difference between the emulator and the JNCC model predictions to assess the goodness of fit. We can see that in all three cases, the emulator provides an accurate representation of the JNCC model of the species. The difference plot shows us that the emulator is not perfect – there are areas where the emulator overestimates or underestimates the probability of occurrence – but in general we do a very good job with the simple emulator function in Equation (2). Crucially, the emulator function can be evaluated at a fraction of the cost of the original JNCC model.

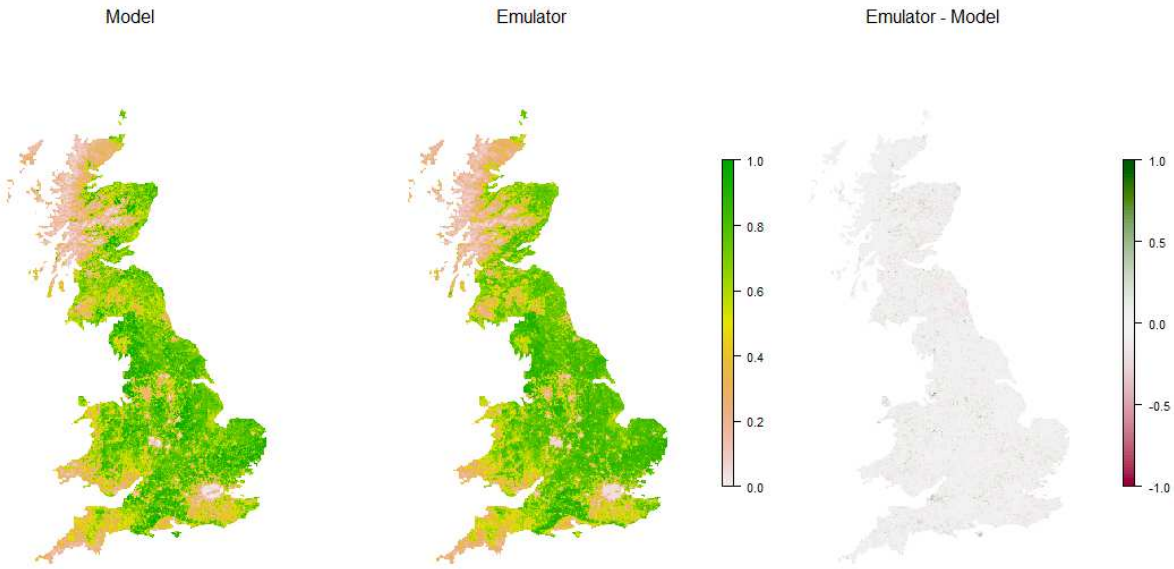

Figure 1: Probability of occurrence for the European hare (Lepus europaeus) as predicted by the JNCC model (left) and the emulator (middle), as well as the difference between the two (right).

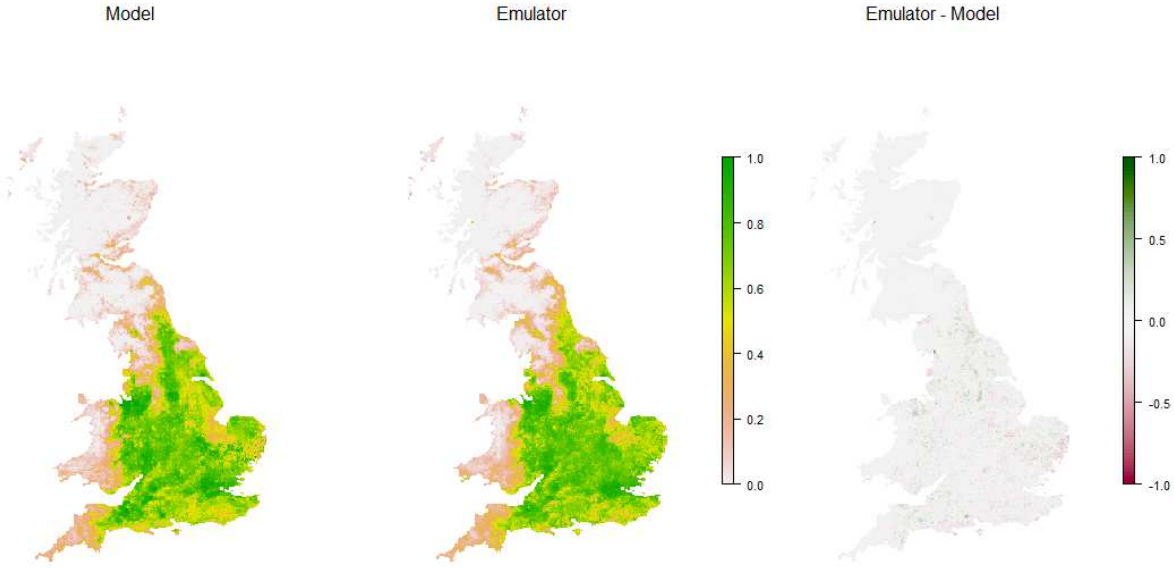

Figure 2: Probability of occurrence for the white-letter hairstreak butterfly (Satyrium w-album) as predicted by the JNCC model (left) and the emulator (middle), as well as the difference between the two (right).

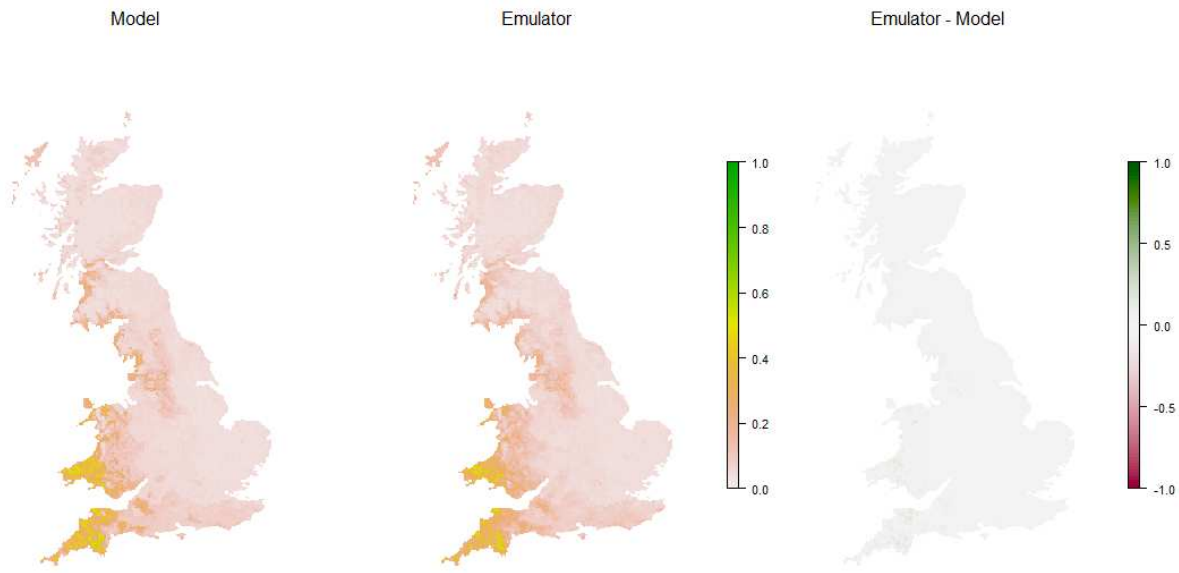

Figure 3: Probability of occurrence for the string-of-sausage lichen (Usnea articulate) as predicted by the JNCC model (left) and the emulator (middle), as well as the difference between the two (right).

Summary

In this report, we have shown how the JNCC models for priority species can be replaced with fast running emulators. The JNCC modelling framework for a species comprised a model ensemble of 7 machine learning approaches, providing a link between species presence and a range of environmental variables. We found that this could be accurately represented with a linear regression function which mapped the logit of the species probability of occurrence to the environmental variables using linear, quadratic and first order interaction terms.

In terms of the speed up the emulator provides over the JNCC model we revisit our example given in the Introduction. Using the emulator, predicting the probability of occurrence for all 100 species for the time period 2020-2059 takes approximately 2-3 seconds – a huge reduction in the computational cost and time taken to run the JNCC model.

References

Ghanem, R. and Spanos, P.D., 1990. Polynomial chaos in stochastic finite elements. *Journal of Applied Mechanics*, *57*(1), pp.197-202.

O'Hagan, A., 2006. Bayesian analysis of computer code outputs: A tutorial. *Reliability Engineering & System Safety*, *91*(10-11), pp.1290-1300.

R Core Team, 2019. R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rasmussen, C.E. and Williams, C.K., 2006. *Gaussian processes for machine learning* (Vol. 2, No. 3, p. 4). Cambridge, MA: MIT Press.

Smith, R.C., 2013. *Uncertainty quantification: theory, implementation, and applications* (Vol. 12). Siam.

Xiu, D. and Karniadakis, G.E., 2002. The Wiener--Askey polynomial chaos for stochastic differential equations. *SIAM journal on scientific computing*, *24*(2), pp.619-644.