[Preprint]
A deep learning approach to Species Distribution Modelling

Christophe Botella[1,2,3,5], Alexis Joly[1], Pierre Bonnet[3,4], Pascal Monestiez[5], and François Munoz[6]

[1]INRIA Sophia-Antipolis - ZENITH team, LIRMM - UMR 5506 - CC 477, 161 rue Ada, 34095 Montpellier Cedex 5, France.
[2]INRA, UMR AMAP, F-34398 Montpellier, France.
[3]AMAP, Univ Montpellier, CIRAD, CNRS, INRA, IRD, Montpellier, France.
[4]CIRAD, UMR AMAP, F-34398 Montpellier, France.
[5]BioSP, INRA, Site Agroparc, 84914 Avignon, France.
[6]Université Grenoble Alpes, 621 avenue Centrale, 38400 Saint-Martin-d'Hères, France.

December 1, 2017

Accepted in : Multimedia Tools and Applications
Title of Book: Multimedia Technologies for Environmental & Biodiversity Informatics

Abstract

Species distribution models (SDM) are widely used for ecological research and conservation purposes. Given a set of species occurrence, the aim is to infer its spatial distribution over a given territory. Because of the limited number of occurrences of specimens, this is usually achieved through environmental niche modeling approaches, *i.e.* by predicting the distribution in the geographic space on the basis of a mathematical representation of their known distribution in environmental space (= realized ecological niche). The environment is in most cases represented by climate data (such as temperature, and precipitation), but other variables such as soil type or land cover can also be used. In this paper, we propose a deep learning approach to the problem in order to improve the predictive effectiveness. Non-linear prediction models have been of interest for SDM for more than a decade but our study is the first one bringing empirical evidence that deep, convolutional and multilabel models might participate to resolve the limitations of SDM. Indeed, the main challenge is that the realized ecological niche is often very different from the theoretical fundamental niche, due to environment perturbation history, species propagation constraints and biotic interactions. Thus, the realized abundance in the environmental feature space can have a very irregular shape that can be difficult to capture with classical models. Deep neural networks on the other side, have been shown to be able to learn complex non-linear transformations in a wide variety of domains. Moreover, spatial patterns in environmental variables often contains useful information for species distribution but are usually not considered in classical models. Our study shows empirically how convolutional neural networks efficiently use this information and improve prediction performance.

1 Introduction

1.1 Context on species distribution models

Species distribution models (SDM) have become increasingly important in the last few decades for the study of biodiversity, macro ecology, community ecology and the ecology of conservation. An accurate

knowledge of the spatial distribution of species is actually of crucial importance for many concrete scenarios including the landscape management, the preservation of rare and/or endangered species, the surveillance of alien invasive species, the measurement of human impact or climate change on species, etc. Concretely, the goal of SDM is to infer the spatial distribution of a given species based on a set of geo-localized occurrences of that species (collected by naturalists, field ecologists, nature observers, citizen sciences project, etc.). However, it is usually not possible to learn that distribution directly from the spatial positions of the input occurrences. The two major problems are the limited number of occurrences and the bias of the sampling effort compared to the real underlying distribution. In a real-world dataset, the raw spatial distribution of the observations is actually highly correlated to the preference and habits of the observers and not only to the spatial distribution of the species. Another difficulty is that in most cases, we only have access to presence data but not to absence data. In other words, occurrences inform that a species was observed at a given location but never that it was not observed at a given location. Consequently, a region without any observed specimen in the data remains highly uncertain. Some specimens could live there but were not observed, or no specimen live there but this information is not recorded. Finally, knowing abundance in space doesn't give information about the ecological determinants of species presence.
For all these reasons, SDM is usually achieved through *environmental niche modeling* approaches, *i.e.* by predicting the distribution in the geographic space on the basis of a representation in the environmental space. This environmental space is in most cases represented by climate data (such as temperature, and precipitation), but also by other variables such as soil type, land cover, distance to water, etc. Then, the objective is to learn a function that takes the environmental feature vector of a given location as input and outputs an estimate of the abundance of the species. The main underlying hypothesis is that the abundance function is related to the *fundamental ecological niche* of the species, in the sense of Hutchinson (see Hutchinson [1957]). That means that in theory, a given species is likely to live in a single privileged ecological niche, characterized by an unimodal distribution in the environmental space. However, in reality, the abundance function is expected to be more complex. Many phenomena can actually affect the distribution of the species relative to its so called *abiotic* preferences. For instance, environment perturbations, or geographical constraints, or interactions with other living organisms (including humans) might have encourage specimens of that species to live in a different environment. As a consequence, the *realized ecological niche* of a species can be much more diverse and complex than its hypothetical fundamental niche.

1.2 Interest of deep and convolutional neural networks for SDM

Notations: When talking about environmental input data, there could be confusions between their different possible formats. Without precisions given, x will represent a general input environmental variable which can have any format. When a distinction is made, x will represent a vector, while an array is always noted X. To avoid confusions on notations for the differents index kinds , we note the spatial **site** index as superscript on the input variable (x^k or X^k for k^{th} site) and the component index as subscript (so x_j^k for the j^{th} component of k^{th} site vector $x_k \in \mathbb{R}^p$, or for the array $X^k \in \mathcal{M}_{d,e,p}(\mathbb{R})$, $X^k_{.,j,.}$ is the j^{th} matrix slice taken on its second dimension). When we denote an input associated with a precise **point location** taken in a continuous spatial domain, the point z is noted as argument: $x(z)$.

Classical SDM approaches postulate that the relationship between output and environmental variables is relatively simple, typically of the form:

$$g(\mathbb{E}[y|x]) = \sum_j f_j(x_j) + \sum_{j,j'} h_{j,j'}(x_j, x_{j'}) \qquad (1)$$

where y is the response variable targeted, a presence indicator or an abundance in our case, the x_j's are components of a vector of environmental variables given as input for our model, f_j are real monovariate functions of it, $h_{j,j'}$ are bivariate real functions representing pairwise interactions effects between inputs, and g is a link function that makes sure $\mathbb{E}[y|x]$ lies in the space of our response variable y. State-of-the-art classification or regression models used for SDM in this way include GAM (Hastie & Tibshirani [1986]), MARS (Friedman [1991]) or MAXENT (Phillips *et al.* [2004],Phillips *et al.* [2006]). Thanks to f_j, we can isolate and understand the effect of the environmental factor x_j on the response. Often, pairwise effects form of $h_{j,j'}$ is restricted to products, like it is the case in the very popular model MAXENT. It facilitates the interpretation and limits the dimensionality of model parameters. However, it sets a strong prior constraint without a clear theoretical founding as the explanatory factors of a species presence can be related to complex environmental patterns.

To overcome this limitation, deep feedforward neural networks (NN) (Goodfellow *et al.* [2016]) are good candidates, because their architecture favor high order interactions effects between the input variables, without constraining too much their functional form thanks to the depth of their architecture. To date, deep NN have shown very successful applications, in particular image classification (Krizhevsky *et al.* [2012]). Until now, to our knowledge, only one-layered-NN's have been tested in the context of SDM (*e.g.* in Lek *et al.* [1996] or Thuiller [2003]). If they are able to capture a large panel of multivariate functions when they have a large number of neurons, their optimization is difficult, and deep NN have been shown empirically to improve optimization and performance (see section 6.4.1 in Goodfellow *et al.* [2016]). However, NN overfit seriously when dealing with small datasets, which is the case here (≈ 5000 data), for this reason we need to find a way to regularize those models in a relevant way. An idea that is often used in SDM (see for example Leathwick *et al.* [2006]) and beyond is to mutualize the heavy parametric part of the model for many species responses in order to reduce the space of parameters with highest likelihood. To put it another way, a NN that shares last hidden layer neurons for the responses of many species imposes a clear constraint: the parameters must construct high level ecological concepts which will explain as much as possible the abundance of all species. These high-level descriptors, whose number is controlled, should be seen as environmental variables that synthesize the most relevant information in the initial variables.

Another limitation of models described by equation (1) is that they don't capture spatial autocorrelation of species distribution, nor the information of spatial patterns described by environmental variables which can impact species presence. In the case of image recognition, where the explanatory data is an image, the variables, the pixels, are spatially correlated, as are the environmental variables used in the species distribution models. Moreover, the different channels of an image, RGB, can not be considered as being independent of the others because they are conditioned by the nature of the photographed object. We can see the environmental variables of a natural landscape in the same way as the channels of an image, noting that climatic, soil, topological or land use factors have strong correlations with others, they are basically not independent of each other. Some can be explained by common mechanisms as is the case with the different climatic variables, but some also act directly on others, as is the case for soil and climatic conditions on land use in agriculture, or the topology on the climate. These different descriptors can be linked by the concept of ecological environment. Thus, the heuristic that guides our approach is that the ecological niche of a species can be more effectively associated with high level ecological descriptors that combine non linearly the environmental variables on one hand, and the identification of multidimensional spatial patterns of images of environmental descriptors on the other hand. Convolutional neural networks (CNN, see LeCun *et al.* [1989]) applied to multi-dimensional spatial rasters of environmental variables can theoretically capture those, which makes them of particular interest.

1.3 Contribution

This work is the first attempt in applying deep feedforward neural networks and convolutional neural networks in particular to species distribution modeling. It introduces and evaluates several architectures based on a probabilistic modeling suited for regression on count data, the Poisson regression. Indeed, species occurrences are often spatially degraded in publicly available datasets so that it is statistically and computationally more relevant to aggregate them into counts. In particular, our experiments are based on the count data of the National Inventory for Nature Protection (INPN[1]), for 50 plant species over the metropolitan French territory along with various environmental data. Our models are compared to MAXENT, which is among the most used classical model in ecology. Our results first show how mutualizing model features for many species prevent deep NN to overfit and finally allow them to reach a better predictive performance than the MAXENT baseline. Then, our results show that convolutional neural networks performed even better than classical deep feed-forward networks. This shows that spatially extended environmental patterns contain relevant extra information compared to their punctual values, and that species generally have a highly autocorrelated distribution in space. Overall, an important outcome of our study is to show that a restricted number of adequately transformed environmental variables can be used to predict the distribution of a huge number of species. We believe the study of the high-level environmental descriptors learned by the deep NNs could help to better understand the co-abundance of different species, and would be of great interest for ecologists.

2 A Deep learning model for SDM

2.1 A large-scale Poisson count model

In this part, we introduce the statistical model which we assume generates the observed data. Our data are species observations without sampling protocol and spatially aggregated on large spatial quadrat cells of 10x10km. Thus, it is relevant to see them as counts.

To introduce our proposed model, we first need to clarify the distinction between the notion of "obsvered abundance" and "probability of presence". Abundance is a number of specimens relatively to an area. In this work, we model species *observed abundance* rather than *probability of presence* because we work with presence only data and without any information about the sampling process. Using presence-absence models, such as logistic regression, could be possible but it would require to arbitrarily generate absence data. And it has been shown that doing so can highly affect estimation and give biased estimates of total population Ward *et al.* [2009]. Working with observed abundance doesn't bias the estimation as long as the space if homogeneously observed and we don't look for absolute abundance, but rather relative abundance in space.

The observed abundance, *i.e.* the number of specimens of a plant species found in a spatial area, is very often modeled by a Poisson distribution in ecology: when a large number of seeds are spread in the domain, each being independent and having the same probability of growing and being seen by someone, the number of observed specimens in the domain will behave very closely to a Poisson distribution. Furthermore, many recent SDM models, especially MAXENT as we will see later, are based on inhomogeneous Poisson point processes (IPP) to model the distribution of species specimens in an heterogeneous environment. However, when geolocated observations are aggregated in spatial quadrats ($\approx$ 10km x 10km each in our case), observations must be interpreted as count per quadrats. If we consider K quadrats named $(s_1, ..., s_K)$ (we will call them sites from now), with empty intersection, and we consider observed specimens are distributed according to $\mathcal{IPP}(\lambda)$, where λ is a positive function defined on $\mathbb{R}^p$ and integrable over our study domain D (where x is known everywhere), we obtain the following equation :

$$\forall k \in [|1, K|], N(s_k) \sim \mathcal{P}\left(\int_{s_k} \lambda(x(z))dz\right) \tag{2}$$

Now, in a parametric context, for the estimation of the parameters of λ, we need to evaluate the integral by computing a weighted sum of λ values taken at quadrature points representing all the potential variation of λ. As our variables x are constant by spatial patches, we need to compute λ on every point with a unique value of x inside s_k, and to do this for every $k \in [|1, K|]$. This can be very computationally and memory expensive. For example, if we take a point per square

[1] http://https://inpn.mnhn.fr/

km (common resolution for environmental variables), it would represent 518,100 points of vector, or patch, input to extract from environmental data and to handle in the learning process. At the same time, environmental variables are very autocorrelated in space, so the gain estimation quality can be small compared to taking a single point per site. Thus, for simplicity, we preferred to make the assumption, albeit coarse, that the environmental variables are constant on each site and we take the central point to represent it. Under this assumption, we justify by the following property the Poisson regression for estimating the intensity of an IPP.

Property: The inhomogeneous Poisson process estimate is equivalent to a Poisson regression estimate with the hypothesis that $x(z)$ is constant over every site.

Proof: We note $z_1, ..., z_N \in D$ the N species observations points, K the number of disjoints sites making a partition of D, and assumed to have an equal area. We write the likelihood of $z_1, ..., z_N$ according to the inhomogeneous poisson process of intensity function $\lambda \in (\mathbb{R}^+)^D$:

$$p(z_1, ..., z_N|\lambda) = p(N|\lambda)\prod_{i=1}^{N} p(z_i|\lambda)$$

$$= \frac{(\int_D \lambda)^N}{N!}\exp\left(-\int_D \lambda\right)\prod_{i=1}^{N}\frac{\lambda(x(z_i))}{\int_D \lambda}$$

$$= \frac{\exp\left(-\int_D \lambda\right)}{N!}\prod_{i=1}^{N}\lambda(x(z_i))$$

We transform the likelihood with the logarithm for calculations commodity:

$$\log(p(z_1, ..., z_N|\lambda)) = \sum_{i=1}^{N}\log\left(\lambda(x(z_i))\right) - \int_D \lambda - \log(N!)$$

We leave the $N!$ term, as it has no impact on the optimisation of the likelihood with respect to the parameters of λ :

$$\sum_{i=1}^{N}\log\left(\lambda(x(z_i))\right) - \int_D \lambda = \sum_{i=1}^{N}\log\left(\lambda(x(z_i))\right) - \sum_{k\in\text{Sites}}\frac{|D|}{K}\lambda(x^k)$$

$$= \sum_{k\in\text{Sites}} n_k \log\left(\lambda(x^k)\right) - \frac{|D|}{K}\lambda(x^k)$$

Where n_k is the number of species occurrences that fall in site k. We can aggregate the occurrences that are in a same site because x is the same for them. We can now factorize $|D|/K$ on the whole sum, which brings us, up to the factor, to the the poisson regression likelihood with pseudo-counts $Kn_k/|D|$.

$$= \frac{|D|}{D}\sum_{k\in\text{Sites}}\frac{Dn_k}{|D|}\log\left(\lambda(x^k)\right) - \lambda(x^k)$$

So maximizing this log-likelihood is exactly equivalent to maximizing the initial Poisson process likelihood. □

Proof uses the re-expression of the IPP likelihood, inspired from Berman & Turner [1992], as that of the associated Poisson regression. In the following parts, we always consider that, for a given species, the number y of specimens observed in a site of environmental input x is as follows:

$$y \sim \mathcal{P}(\lambda_{m,\theta}(x)) \tag{3}$$

Where m is a model architecture with parameters θ.
From equation **(3)**, we can write the likelihood of counts on K different sites $(x_1, ..., x_K)$ for N independently distributed species with abundance functions $(\lambda_{m_i,\theta_i})_{i\in[|1,N|]} \in (\mathbb{R}^+)^{\mathbb{R}^p}$, respectively determined by models $(m_i)_{i\in[|1,N|]}$ and parameters $(\theta_i)_{i\in[|1,N|]}$:

$$p\left((y_k^i)_{i\in[|1,N|],k\in[|1,K|]}|(\lambda_{m_i,\theta_i})_{i\in[|1,N|]}\right) = \prod_{i=1}^{N}\prod_{k=1}^{K}\frac{(\lambda_{m_i,\theta_i}(x_k))^{y_k^i}}{y_k^i!}\exp(-\lambda_{m_i,\theta_i}(x_k))$$

Which gives, when eliminating $\log(y_k^i)!$ terms (which are constant relatively to models parameters), the following negative log-likelihood :

$$\mathcal{L}\left((y_k^i)_{i\in[|1,N|],k\in[|1,K|]}|(\lambda_{m_i,\theta_i})_{i\in[|1,N|]}\right) := \sum_{i=1}^{N}\sum_{k=1}^{K}\lambda_{m_i,\theta_i}(x_k) - y_k^i\log(\lambda_{m_i,\theta_i}(x_k)) \tag{4}$$

Following the principle of maximum likelihood, for fitting a model architecture, we minimize the objective function given in equation (4) relatively to parameters θ.

2.2 Links with MAXENT

For our experiment, we want to compare our proposed models to a state of the art method commonly used in ecology. We explain in the following why and how we can compare the chosen reference, MAXENT, with our models.

MAXENT (Phillips *et al.* [2004],Phillips *et al.* [2006]) is a popular SDM method and related software for estimating relative abundance as a function of environmental variables from presence only data points. This method has proved to be one of the most efficient in prediction P Anderson *et al.* [2006], while guaranteeing a good interpretability thanks to the simple elementary form of its features and its variable selection procedure. The form of the relative abundance function belongs to the class described in Equation 1. More specifically:

$$\log\left(\lambda_{MAX,\theta}(x)\right) = \alpha + \sum_{j=1}^{p}\sum_{s=1}^{S} f_j^s(x_{(j)}) + \sum_{j<j'}\beta_{j,j'}x_j x_j' \tag{5}$$

where $x_{(j)}$ is the j^{th} component of vector x. The link function is a logarithm, and variables interactions effects are product interactions. If x_j is a quantitative variable the functions $(f_s)_{s\in[|1,S|]}$ belongs to 4 categories: linear, quadratic, threshold and hinge. One can get details on the hinges functions used in MAXENT in Phillips & Dudík [2008]. If x_j is categorical, then f_j takes a different value for every category, with one zero category.
It has been shown that MAXENT method is equivalent to the estimation of an IPP intensity function with a specific form and a weighted L1 penalty on its variables Fithian & Hastie [2013]. Let's call $\lambda_{MAX,\theta}(x)$ the intensity predicted by MAXENT with parameters θ at x. Last property says that on any given dataset, $\hat{\theta}$ estimated from a Poisson regression (aggregating observations as counts per site) is the same as the one of the IPP (each observation is an individual point, even when there are several at a same site). In our experiments, we ran MAXENT using the `maxnet` package in R Phillips *et al.* [2017], with the default regularization, and giving to the function :

1. A positive point per observation of the species.

2. A pseudo-absence point per site.

MAXENT returns only the parameters of the $(f_j^s)_{s,j}$ and the $(\beta_{j,j'})_{j<j'}$, but not the intercept α, as it is meant to only estimate the absolute abundance. We don't aim at estimating absolute abundance either, however, we need the intercept to measure interesting performance metrics across all the compared models. To resolve this, for each species, we fitted the following model using the `glm` package in R as a second step:

$$y \sim \mathcal{P}\left(\exp(\alpha + \log(p))\right)$$

Where α is our targeted intercept, p is the relative intensity prediction given by MAXENT at the given site, and y is the observed number of specimens at this site.

2.3 SDM based on a fully-connected NN model

We give in the following a brief description of the general structure of fully-connected NN models, and how we decline it in our tested deep model architecture.

General introduction of fully-connected NN models. A deep NN is a multi-layered model able to learn complex non-linear relationship between an input data, which in our case will be a vector $x \in \mathbb{R}^p$ of environmental variables that is assumed to represent a spatial site, and output variables $y_1, ..., y_N$, which in our case is species counts in the spatial site. The classic so called **fully-connected** NN model is composed of one or more **hidden layer(s)**, and each layer is composed of one or more **neuron(s)**. We note $n(l,m)$ the number of neurons of layer l in model architecture m. m parameters are stored in θ. In the first layer, each neuron is the result of a parametric linear combination of the elements of x, which is then transformed by an **activation function** a. So for a NN m, $a_m^{1,j}(x,\theta) := a(x^T\theta_j^1)$ is called **the activation** of j^{th} neuron of the first hidden layer of m when it is applied to x. Thus, on the l^{th} layer with $l > 1$, the activation of the j^{th} neuron is $a((\theta_j^l)^T a_m^{l-1,.})$. Now, we understand that the neuron is the unit that potentially combines every variables in x, and, its activation inducing a non-linearity to the parametric combination, it can be understood as a particular basis function in the p dimensional space of x. Thus, the model is able to combine as many basis functions as there are neurons in each layer, and the basis functions become more and more complex when going to further layers. Finally, these operations makes m theoretically able to closely fit a broad range of functions of x.

Learning of model parameters is done through optimization (minimization by convention) of an objective function that depends on the prediction goal. Optimization method for NN parameters θ is based on stochastic gradient descent algorithms, however, the loss function gradient is approximated by the back-propagation algorithm Rumelhart *et al.* [1988].

Learning a NN model lead to a lot of technical difficulties that have been progressively dealt with during last decade, and through many different techniques. We present some that have been of particular interest in our study. A first point is that there are several types of activation functions, the first one introduced being the sigmoid function. However, the extinction of its gradient when $x^T\theta_j^1$ is small or big, has presented a serious problem for parameters optimization in the past. More recently, the introduction of the ReLU (Nair & Hinton [2010]) activation function helped made an important step forward in NNs optimization. A second point is that when we train a NN model, simultaneous changes of all the parameters lead to important change in the distribution (across the dataset) of each activation of the model. This phenomenon is called internal covariate shift, and perturbs learning importantly. Batch-Normalization (Ioffe & Szegedy [2015]) is a technique that significantly reduces internal covariate shift and help to regularize our model as well. It consists of a parameterized centering and reduction of pre-activations. This facilitates optimization and enables to raise the learning rate leading to a quicker convergence. At the same time, it has a regularization effect because the centering and reduction of a neuron activation is linked to the mini-batch statistics. The mini-batch selection being stochastic at every iteration, a neuron activation is stochastic itself, and the model will not rely on it when it has no good effect on prediction.

Models architecture in this study. For a given species i, When we know the model parameter θ, we can predict the parameter of the Poisson distribution of the random response variable $y_i \in \mathbb{N}$, *i.e.* the count of species i, conditionally on its corresponding input x, with the formula :

$$\lambda_{m,\theta}(x) = \exp(\gamma_i^T a_m^{N_h,.}(x, \theta)) \tag{6}$$

For this work, we chose the logarithm as link function g mentioned in **1.2**. It is the conventional link function for the generalized linear model with Poisson family law, and is coherent with MAXENT. $\gamma_i \in \mathbb{R}^{n(N_h,m)}$ is included in θ. It does the linear combinations of last layer neurons activations for the specific response i. If we set $n(N_h, m) := 200$ as we do in the following experiments, there are only 200 parameters to learn per individual species, while there are a lot more in the shared part of the model that builds $a_m^{N_h,.}(x, \theta)$. Now for model fitting, we follow the method of the maximum likelihood, **the objective function** will be a negative-loglikelihood, but it could otherwise be some other prediction error function. Note that we will rather use the term **loss function** than negative loglikelihood for simplicity. We chose **the ReLU as activation function**, because it showed empirically less optimization problems and a quicker convergence. Plus, we empirically noticed the gain in optimization speed and less complications with the learning rate initialization when using Batch-Normalization. For this reason, Batch-Normalization is applied to every pre-activation (before applying the ReLU) to every class of NN model in this paper, even with CNNs. We give a general representation of the class of NN models used in this work in Figure 1.

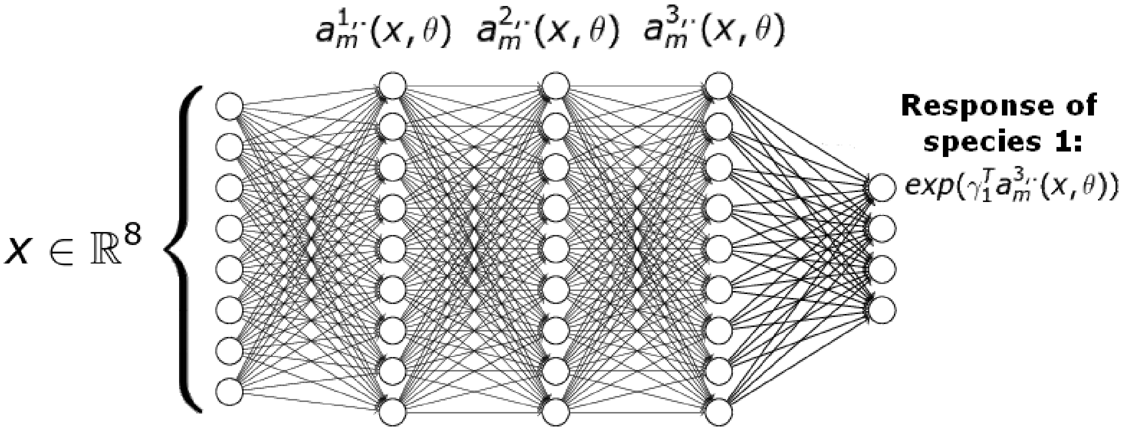


Figure 1: A schematic representation of fully-connected NN architecture. Except writings, image comes from Michael®Nielsen[3]

2.4 SDM based on a convolutional NN model

A convolutional NN (CNN) can be seen as a extension of NN that are particularly suited to deal with certain kind of input data with very large dimensions. They are of particular interest in modeling species distribution, because they are able to capture the effect of spatial environmental patterns. Again, we will firstly describe the general form of CNN before going to our modeling choices.

General introduction of CNN models. CNN is a form of neural network introduced in LeCun *et al.* [1989]. It aims to efficiently apply NN to input data of large size (typically 2D or 3D arrays, like images) where elements are spatially auto-correlated. For example, using a fully-connected neural network with 200 neurons on an input RGB image of dimensions 256x256x3 would imply around $4 * 10^7$ parameters only for the first layer, which is already too heavy computationally to optimize on a standard computer these days. Rather than applying a weight to every pixel of an input array, CNN will apply a **parametric discrete convolution**, based on a kernel of reasonable size (3/3/p or 5/5/p are common for N/N/p input arrays) on the input arrays to get an intermediate feature map (2D). The convolution is applied with a moving windows as illustrated in Figure 2 -B. Noting $\mathbf{X} \in \mathcal{M}_{d,d,p}$ an input array, we simplify notations in all that follows by writing $\mathcal{CV}(X, k_\gamma(c))$ the resulting feature map from applying the convolution with (c, c, p) kernel of parameters $\gamma \in \mathbb{R}^{c^2 p}$. If the convolution is applied directly on $\mathbf{X}$, the sliding window will pass its center over every $X_{i,j,.}$ from the up-left to the bottom-right corner and produce a feature map with a smaller size than the input because $c > 1$. The **zero-padding operation** removes this effect by adding $(c-1)/2$ layers of 0 on every side of the array. After a convolution, there can be a Batch-Normalization and an activation function is generally applied to each pixel of the features maps. Then, there is a synthesizing step made by the **pooling** operation. Pooling aggregates groups of cells in a feature map in order to reduce its size and introduce invariance to local translations and distortions. After having composed these operations several times, when the size of feature maps is reasonably small (typically reaching 1 pixel), a **flattening** operation is applied to transform the 3D array containing all the feature maps into a vector. This features vector will then be given as input to a fully-connected layer as we described in last part. The global concept underlying convolution layers operations is that first layers act as low level interpretations of the signal, leading to activations for salient or textural patterns. Last layers, on their side, are able to detect more complex patterns, like eyes or ears in the case of a face picture. Those high levels features have much greater sense regarding predictions we want to make. Plus, they are of much smaller dimension than the input data, which is more manageable for a fully-connected layer.

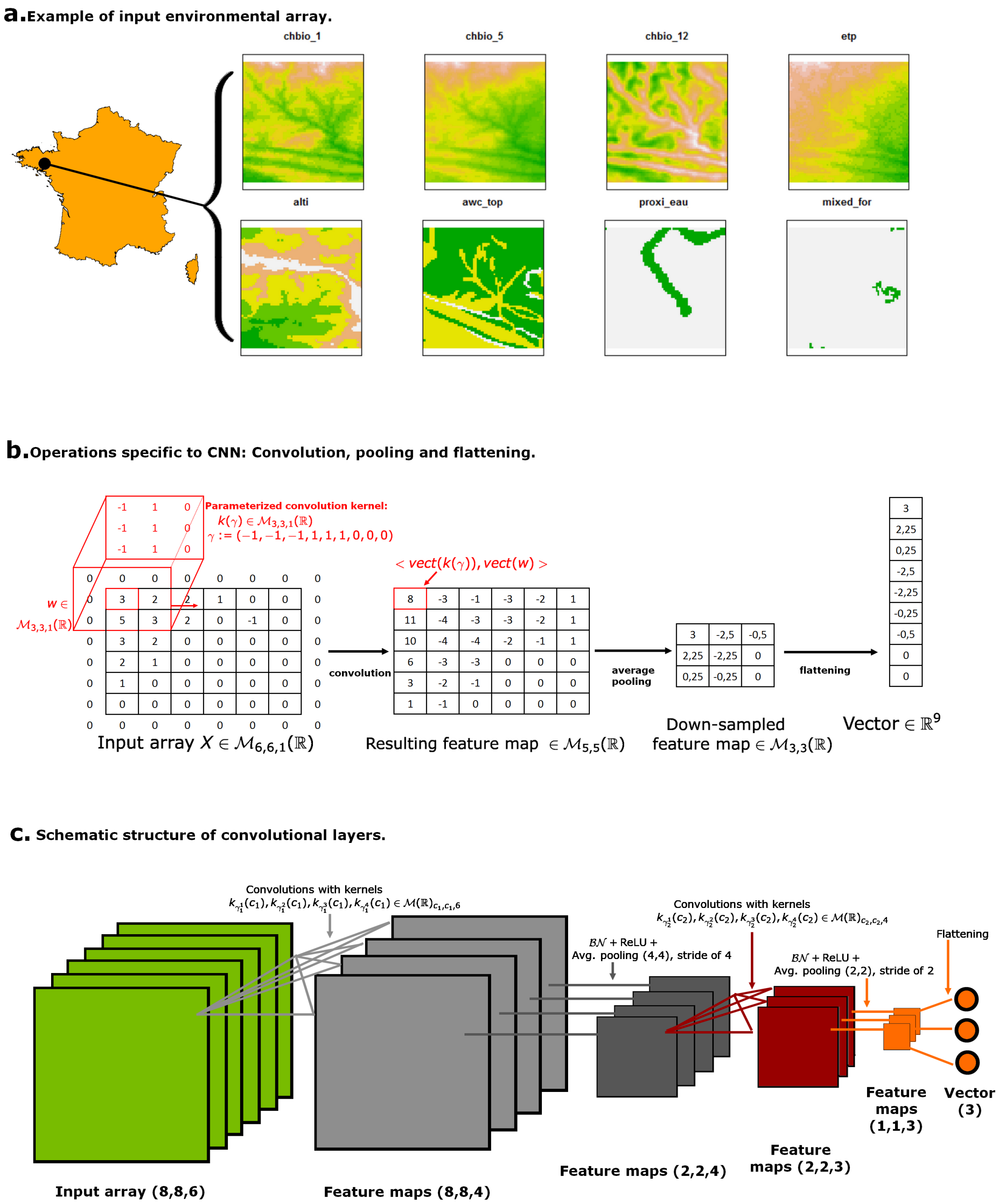
Constitution of a CNN model for SDM. The idea which pushes the use of CNN models for SDM is that complex spatial patterns like a water network, a valley, etc., can affect importantly the species abundance. This kind of pattern can't be really deducted for punctual values of environmental variables. Thus, we have chosen to build a SDM model which takes as input an array with a map of values for each environmental variable that is used in the other models. This way, we will be able to conclude if there is extra relevant information in environmental variables spatial patterns to predict better species distribution. In 2 -A, we show for a single site a subsample of environmental variables maps taken as input by our CNN model. To provide some more detail about the model architecture, the input array X is systematically padded such that the feature map resulting from the convolution is of same size as 2 first dimensions of the input ($(c-1)/2$ cells of 0 after on the sides of the 2 dimensions). To illustrate that, our padding policy is the same as the one illustrated in the example given in Figure 2 -B. However, notice that the kernel size can differ and the third dimension size of input array will be the number of input variables or feature maps. For an example of For the reasons described in **2.3**, **we applied a Batch-Normalization** to each feature map (same normalization for every pixels of a map) before the activation, which **is still a ReLU**. For the pooling opreation, we chose the **average pooling** which seems intuitively more relevant to evaluate an abundance (=concentration). The different kinds of operations and their succession in our CNN model are illustrated in the Figure 2 -C.

3 Data and methods

3.1 Observations data of INPN

This paper is based on a reference dataset composed of count data collected and validated by French expert naturalists. This dataset, referred as INPN[4] for "national inventory of natural heritage"

[4] https://inpn.mnhn.fr



a. Example of input environmental array.

chbio_1
chbio_5
chbio_12
etp
alti
awc_top
proxi_eau
mixed_for

b. Operations specific to CNN: Convolution, pooling and flattening.

Parameterized convolution kernel:
$k(\gamma) \in \mathcal{M}_{3,3,1}(\mathbb{R})$
$\gamma := (-1,-1,-1,1,1,1,0,0,0)$
$w \in \mathcal{M}_{3,3,1}(\mathbb{R})$
$< vect(k(\gamma)), vect(w) >$
convolution
average pooling
flattening
Input array $X \in \mathcal{M}_{6,6,1}(\mathbb{R})$
Resulting feature map $\in \mathcal{M}_{5,5}(\mathbb{R})$
Down-sampled feature map $\in \mathcal{M}_{3,3}(\mathbb{R})$
Vector $\in \mathbb{R}^9$

-1	1	0
-1	1	0
-1	1	0

0	0	0	0	0	0	0	0
0	3	2	2	1	0	0	0
0	5	3	2	0	-1	0	0
0	3	2	0	0	0	0	0
0	2	1	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

8	-3	-1	-3	-2	1
11	-4	-3	-3	-2	1
10	-4	-4	-2	-1	1
6	-3	-3	0	0	0
3	-2	-1	0	0	0
1	-1	0	0	0	0

3	-2,5	-0,5
2,25	-2,25	0
0,25	-0,25	0

3
2,25
0,25
-2,5
-2,25
-0,25
-0,5
0
0

c. Schematic structure of convolutional layers.

Convolutions with kernels
$k_{\gamma_1^1}(c_1), k_{\gamma_1^2}(c_1), k_{\gamma_1^3}(c_1), k_{\gamma_1^4}(c_1) \in \mathcal{M}(\mathbb{R})_{c_1,c_1,6}$
$\mathcal{BN}$ + ReLU +
Avg. pooling (4,4), stride of 4
Convolutions with kernels
$k_{\gamma_2^1}(c_2), k_{\gamma_2^2}(c_2), k_{\gamma_2^3}(c_2), k_{\gamma_2^4}(c_2) \in \mathcal{M}(\mathbb{R})_{c_2,c_2,4}$
$\mathcal{BN}$ + ReLU +
Avg. pooling (2,2), stride of 2
Flattening
Input array (8,8,6)
Feature maps (8,8,4)
Feature maps (2,2,4)
Feature maps (2,2,3)
Feature maps (1,1,3)
Vector (3)

Figure 2: (a) Examples of input environmental data (b) for convolution, pooling and flattening process in our (c) Convolutional Neural Network architecture

Dutrève B. [2016], comes from the GBIF portal[5]. It provides access to occurrences data collected in various contexts including Flora and regional catalogs, specific inventories, field note books, and prospections carried out by the botanical conservatories. In total, the INPN data available on the GBIF contains 20,999,334 occurrences, covering 7,626 species from which we selected 1000 species. The assets of this data are the quality of their taxonomic identification (provided by an expert network), their volume and geographic coverage. Its main limitation, however, is that the geolocation of the occurrences was degraded (for plant protection concerns). More precisely, all geolocations were

[5] `https://www.gbif.org/`

aggregated to the closest central point of a spatial grid composed of 100 km2 quadrat cells (*i.e.* sites of 10×10km). Thus, the number of observations of a species falling in a site gives a count.
In total, our study is based on 5,181 sites, which are split in 4,781 training sites for fitting models, and 400 test sites for validating and comparing models predictions.

3.2 Species selection

For the genericity of our results and to make sure they are not biased by the choice of a particular category of species, we have chosen to work with a high number of randomly chosen species. From the 7,626 initial species, we selected species with more than 300 observations. We selected amongst those a random subset of 1000 species to constitute an ensemble E_{1000}. Then, we randomly selected 200 species amongst E_{1000} to constitute E_{200}, and finally randomly selected 50 in E_{200} which gave E_{50}. E_{50} being the main dataset used to compare our model to the baselines, we provide in Figure 1 the list of species composing it. The full dataset with species of E_{1000} contains 6,134,016 observations in total (see Table 1 for the detailed informations per species).

3.3 Environnemental data

In the following, we denote by p the number of environmental descriptors. For this study, we gathered and compiled different sources of environmental data into $p = 46$ geographic rasters containing the pixel values of environmental descriptors presented in the table 2 with several resolutions, nature of values, but having a common cover all over the metropolitan French territory. We chose some typical environmental descriptors for modeling plant distribution that we believe carry relevant information both as punctual and spatial representation. They can be classified as bioclimatic, topological, pedologic hydrographic and land cover descriptors. In the following, we briefly describe the sources, production method, and resolution of initial data, and the contingent specific post-process for reproducibility.

3.3.1 Climatic descriptors: Chelsea Climate data 1.1

Those are raster data with worldwide coverage and 1km resolution. A mechanistical climatic model is used to make spatial predictions of monthly mean-max-min temperatures, mean precipitations and 19 bioclimatic variables, which are downscaled with statistical models integrating historical measures of meteorologic stations from 1979 to today. The exact method is explained in the reference papers Karger *et al.* [2016b] and Karger *et al.* [2016a]. The data is under Creative Commons Attribution 4.0 International License and downloadable at (`http://chelsa-climate.org/downloads/`).

3.3.2 Potential Evapotranspiration : CGIAR-CSI ETP data

The CGIAR-CSI distributes this worldwide monthly potential-evapotranspiration raster data. It is pulled from a model developed by Antonio Trabucco (Zomer *et al.* [2007], Zomer *et al.* [2008]). Those are estimated by the Hargreaves formula, using mean monthly surface temperatures and standard deviation from WorldClim 1:4 (`http://www.worldclim.org/version1`), and radiation on top of atmosphere. The raster is at a 1km resolution, and is freely downloadable for a nonprofit use at `http://www.cgiar-csi.org/data/global-aridity-and-pet-database#description`.

3.3.3 Pedologic descriptors : The ESDB v2 - 1kmx1km Raster Library

The library contains multiple soil pedology descriptor raster layers covering Eurasia at a resolution of 1km. We selected 11 descriptors from the library. More precisely, those variables have ordinal format, representing physico-chemical properties of the soil, and come from the PTRDB. The PTRDB variables have been directly derived from the initial soil classification of the Soil Geographical Data Base of Europe (SGDBE) using expert rules. SGDBE was a spatial relational data base relating spatial

Taxon name	Total # obs.	Prevalence
Alisma plantago-aquatica L.	15324	56.3
Alopecurus geniculatus L.	5703	31.5
Antennaria carpatica (Wahlenb.) Bluff & Fingerh.	1780	4.0
Anthriscus sylvestris (L.) Hoffm.	27381	64.9
Astragalus hypoglottis L.	1901	5.7
Berteroa incana (L.) DC.	3966	11.2
Biscutella brevicaulis Jord.	450	1.0
Campanula spicata L.	544	1.7
Carduus vivariensis Jord.	1577	7.4
Carex ericetorum Pollich	538	1.8
Carlina acanthifolia All.	6214	10.6
Centranthus angustifolius (Mill.) DC.	2755	5.9
Cladanthus mixtus (L.) Chevall.	637	5.3
Coronilla coronata L.	325	0.9
Cynoglossum creticum Mill.	1470	9.2
Cytisus villosus Pourr.	562	1.0
Dianthus pyrenaicus Pourr.	392	0.8
Epilobium alpestre (Jacq.) Krocker	1197	3.5
Euphorbia dendroides L.	747	0.5
Festuca cinerea Vill.	3795	5.3
Galium lucidum All.	3204	11.7
Galium timeroyi Jord.	1362	6.6
Helictotrichon sedenense (Clarion ex DC.) Holub	8498	5.4
Hieracium lawsonii Vill.	629	3.2
Hieracium praecox Sch.Bip.	998	4.7
Iris lutescens Lam.	2537	6.6
Juncus trifidus L.	3570	3.9
Lathyrus niger (L.) Bernh.	2474	13.8
Myrtus communis L.	2054	1.9
Meconopsis cambrica (L.) Vig.	1291	3.8
Oxalis corniculata L.	5628	37.5
Oxytropis fetida (Vill.) DC.	315	1.0
Persicaria vivipara (L.) Ronse Decraene	11122	5.9
Phleum alpinum L.	7267	6.3
Potamogeton coloratus Hornem.	813	5.5
Potentilla pusilla Host	655	1.7
Primula latifolia Lapeyr.	1268	1.8
Psilurus incurvus (Gouan) Schinz & Thell.	597	4.2
Ranunculus parnassifolius L.	371	1.0
Ranunculus repens L.	76346	83.0
Reseda lutea L.	16756	49.0
Rorippa pyrenaica (All.) Rchb.	2169	9.2
Rubus ulmifolius Schott	14523	35.5
Thalictrum aquilegifolium L.	2855	8.8
Thalictrum alpinum L.	581	1.0
Trifolium micranthum Viv.	767	8.0
Valerianella rimosa Bast.	1518	13.8
Vicia onobrychioides L.	1602	6.3
Viola lactea Sm.	520	4.7
Viscaria vulgaris Bernh.	781	3.2

Table 1: List of species in E_{50} with the total number of observations and prevalence in the full database.

Name	Description	Nature	Values	Resolution
CHBIO_1	Annual Mean Temperature	quanti.	[-10.6,18.4]	30
CHBIO_2	Mean of monthly max(temp)-min(temp)	quanti.	[7.8,21.0]	30
CHBIO_3	Isothermality (100*chbio_2/chbio_7)	quanti.	[41.2,60.0]	30
CHBIO_4	Temperature Seasonality (std. dev.*100)	quanti.	[302,778]	30
CHBIO_5	Max Temperature of Warmest Month	quanti.	[36.4,6.2]	30
CHBIO_6	Min Temperature of Coldest Month	quanti.	[-28.2,5.3]	30
CHBIO_7	Temp. Annual Range (5- 6)	quanti.	[16.7,42.0]	30
CHBIO_8	Mean Temp. of Wettest Quarter	quanti.	[-14.2,23.0]	30
CHBIO_9	Mean Temp. of Driest Quarter	quanti.	[-17.7,26.5]	30
CHBIO_10	Mean Temp. of Warmest Quarter	quanti.	[-2.8,26.5]	30
CHBIO_11	Mean Temp. of Coldest Quarter	quanti.	[-17.7,11.8]	30
CHBIO_12	Annual Precipitation	quanti.	[318,2543]	30
CHBIO_13	Precip. of Wettest Month	quanti.	[43.0,285.5]	30
CHBIO_14	Precip. of Driest Month	quanti.	[3.0,135.6]	30
CHBIO_15	Precip. Seasonality (Coef. of Var.)	quanti.	[8.2,26.5]	30
CHBIO_16	Precipitation of Wettest Quarter	quanti.	[121,855]	30
CHBIO_17	Precipitation of Driest Quarter	quanti.	[20,421]	30
CHBIO_18	Precip. of Warmest Quarter	quanti.	[19.8,851.7]	30
CHBIO_19	Precip. of Coldest Quarter	quanti.	[60.5,520.4]	30
etp	Potential Evapo Transpiration	quanti.	[133,1176]	30
alti	Elevation	quanti.	[-188,4672]	3
awc_top	Topsoil available water capacity	ordinal	$\{0, 120, 165, 210\}$	30
bs_top	Base saturation of the topsoil	ordinal	$\{35, 62, 85\}$	30
cec_top	Topsoil cation exchange capacity	ordinal	$\{7, 22, 50\}$	30
crusting	Soil crusting class	ordinal	[|0, 5|]	
dgh	Depth to a gleyed horizon	ordinal	$\{20, 60, 140\}$	30
dimp	Depth to an impermeable layer	ordinal	$\{60, 100\}$	30
erodi	Soil erodibility class	ordinal	[|0, 5|]	30
oc_top	Topsoil organic carbon content	ordinal	$\{1, 2, 4, 8\}$	30
pd_top	Topsoil packing density	ordinal	$\{1, 2\}$	30
text	Dominant surface textural class	ordinal	[|0,5|]	30
proxi_eau	<50 meters to fresh water	bool.	$\{0, 1\}$	30
arti	Artificial area: clc $\in \{1, 10\}$	bool.	$\{0, 1\}$	30
semi_arti	Semi-artificial area: clc $\in \{2, 3, 4, 6\}$	bool.	$\{0, 1\}$	30
arable	Arable land: clc $\in \{21, 22\}$	bool.	$\{0, 1\}$	30
pasture	Pasture land: clc $\in \{18\}$	bool.	$\{0, 1\}$	30
brl_for	Broad-leaved forest: clc $\in \{23\}$	bool.	$\{0, 1\}$	30
coni_for	Coniferous forest: clc $\in \{24\}$	bool.	$\{0, 1\}$	30
mixed_for	Mixed forest: clc $\in \{25\}$	bool.	$\{0, 1\}$	30
nat_grass	Natural grasslands: clc $\in \{26\}$	bool.	$\{0, 1\}$	30
moors	Moors: clc $\in \{27\}$	bool.	$\{0, 1\}$	30
sclero	Sclerophyllous vegetation: clc $\in \{28\}$	bool.	$\{0, 1\}$	30
transi_wood	Transitional woodland-shrub: clc $\in \{29\}$	bool.	$\{0, 1\}$	30
no_veg	No or few vegetation: clc $\in \{31, 32\}$	bool.	$\{0, 1\}$	30
coastal_area	Coastal area: clc $\in \{37, 38, 39, 42, 30\}$	bool.	$\{0, 1\}$	30
ocean	Ocean surface: clc $\in \{44\}$	bool.	$\{0, 1\}$	30

Table 2: Table of 46 environmental variables used in this study.

units to a diverse pedological attributes of categorical nature, which is not useful for our purpose. For more details, see Panagos [2006], Panagos *et al.* [2012] and Van Liedekerke *et al.* [2006]. The data is maintained and distributed freely for scientific use by the European Soil Data Centre (ESDAC) at `http://eusoils.jrc.ec.europa.eu/content/european-soil-database-v2-raster`.

3.3.4 Altitude : USGS Digital Elevation data

The Shuttle Radar Topography Mission achieved in 2010 by Endeavour shuttle managed to measure digital elevation at 3 arc second resolution over most of the earth surface. Raw measures have been post-processed by NASA and NGA in order to correct detection anomalies. The data is available from the U.S. Geological Survey, and downloadable on the Earthexplorer (`https://earthexplorer.usgs.gov/`). One can refer to `https://lta.cr.usgs.gov/SRTMVF` for more informations.

3.3.5 Hydrographic descriptor: BD Carthage v3

BD Carthage is a spatial relational database holding many informations on the structure and nature of the french metropolitan hydrological network. For the purpose of plants ecological niche, we focus on the geometric segments representing watercourses, and polygons representing hydrographic fresh surfaces. The data has been produced by the *Institut National de l'information Géographique et forestière* (IGN) from an interpretation of the BD Ortho IGN. It is maintained by the SANDRE under free license for non-profit use and downloadable at `http://services.sandre.eaufrance.fr/telechargement/geo/ETH/BDCarthage/FX`. From this shapefile, we derived a raster containing the binary value of variable `proxi_eau`, i.e. proximity to fresh water, all over France. We used qgis to rasterize to a 12.5 meters resolution, with a buffer of 50 meters, the shapefile `COURS_D_EAU.shp` on one hand, and the polygons of `SURFACES_HYDROGRAPHIQUES.shp` with attribute NATURE="Eau douce permanente" on the other hand. We then created the maximum raster of the previous ones (So the value of 1 correspond to an approximate distance of less than 50 meters to a watercourse or hydrographic surface of fresh water).

3.3.6 Land cover : Corine Land Cover 2012, version 18.5.1, 12/2016

It is a raster layer describing soil occupation with 48 categories across Europe (25 countries) at a resolution of 100 meters. This classification is the result of an interpretation process from earth surface high resolution satellite images. This data base of the European Union is freely accessible online for all use at `http://land.copernicus.eu/pan-european/corine-land-cover/clc-2012` and commonly used for the purpose of plant distribution modeling. For a need of meaningfull variables at our scale and reduced memory consumption, we reduced the number of categories to 14 following mainly the procedure of They eliminate some categories of few interest, too rare or inaccurate, and groups categories that are associated with similar plant communities. In addition, we introduce a category "Semi artificial surfaces", which regroups perturbed natural areas, interesting for the study of alien invasive species. We keep the Corine Land Cover category called "Sea and ocean" that can be an important contextual variable for the convolutional neural network model, and . The final categories groups are detailed in the table 2. for each of the retain categories, we created a raster of the same resolution as the original one, where the value 1 means the pixel belongs to the category, or the value is 0 otherwise.

3.3.7 Environmental variables extraction and format

When creating the p global GeoTIIF rasters, as the original coordinate system of the layer vary among sources, we change it if necessary to WGS84 using `rgdal` package on R, which is the coordinate system INPN occurrences databases. As explained previously, for computational reasons considering the scale, and simplicity, we chose to represent each site by a single geographic point, and chose the center of the site. We are going to compare two types of models. For a site k, the first takes as input a vector of p elements which values are those of the environmental variables taken at the geolocation of the center of the site k, while the other takes p rasters of size (d,d) cropped (with package `raster`) from the global raster of each environmental descriptors and centered at the center of k. If we denote $res_{\text{lon},j}$ the spatial resolution in longitude of global raster of the j_th environmental descriptor, and $res_{\text{lat},j}$ its resolution in latitude, the spatial extent of $X^k_{.,.,j}$ is $(d.res_{\text{lat},j} \times d.res_{\text{lon},j})$. As a consequence, the extents are heterogeneous across environmental descriptors. In this study, we experimented the method with $d = 64$, so the input data items X^k learned by our convolutional model is of dimension $64 \times 64 \times 46$.

3.4 Detailed models architectures and learning protocol

MAXENT is learned independently on every species of E_{50}. Similarly, we fit a classic loglinear model to give a naive reference. Then, two architectures of NN are tested, one with a single hidden layer (SNN), one with six hidden layers (DNN). Those models take a vector of environmental variables x^k as input. As introduced previously, we want to evaluate if training a multi-response NN model, *i.e.* a NN predicting several species from a single $a_m^{N_h(m)}(x, \theta)$, can prevent overfitting. One architecture of CNN is tested, which takes as input an array X^k. Hereafter, we described more precisely the architecture of those models.

3.4.1 Baseline models

• **LGL** Considering a site k, and its environmental variables vector x^k, the output function λ_{LGL} of the loglinear model parametrized by $\beta \in \mathbb{R}^p$ is simply the exponential of a scalar product between x^k and β :

$$\lambda_{LGL}(x^k, \beta) = \exp\left(\beta^T x^k\right)$$

As LGL has no hidden layer, we learned a multi-response model, which is equivalent to fitting the 50 mono-response models independently.

• **MAXENT**.

3.4.2 Proposed models based on NN

• **SNN** has only 1 hidden layer ($N_h = 1$) with 200 neurons ($|a^1_{SNN}| = 200$) all batch-normalized and the activation function is ReLU. As the architecture is not deep, it makes a control example to evaluate when stacking more layers. SNN is tested in 3 multi-response versions, on E_{50}, E_{200} or E_{1000}.

• **DNN** is a deep feedforward network with $N_h = 6$ hidden layers and $n(l, \text{DNN}) = 200, \forall l \in [|1, 6|]$. Every pre-activation is Batch-normalized and has a ReLU activation. DNN is tested in 4 versions, the mono-response case fitted independently on each species of E_{50} like MAXENT and LGL, and the multi-response fitted on E_{50}, E_{200} or E_{1000}.

• **CNN** is composed of two hidden convolutional layers and one last layer fully connected with 200 neurons, exactly similar to previous ones. The first layer is composed of 64 convolution filters of kernel size $(3, 3)$ and 1 line of 0 padding. The resulting feature maps are batch-normalized (same normalization for every pixels of a feature map) and transformed with a Relu. Then, an average pooling with a $(8, 8)$ kernel and $(8, 8)$ stride is applied. The second layer is composed of 128 convolution filters of kernel size $(5, 5)$ and 2 lines of padding, plus Batch-Normalization and ReLU. After, that a second average pooling with a $(8, 8)$ kernel and $(8, 8)$ kernel and $(8, 8)$ stride reduces size of the 128 feature maps to one pixel. Those are collected in a vector by a flattening operation preceding the fully connected layer. This architecture is not very deep. However, considered the restricted number of samples, a deep CNN would be very prone to over fitting. CNN is tested in multi-responses versions on E_{50}, E_{200} and E_{1000}.

3.4.3 Models optimization

Our experiments were conducted using the `R` framework (version 3.3.2), on a Windows 10 machine with 2 CPUs with 2.60 GHz and 4 cores each, and one GPU NVIDIA Quadro M1000M. `mxnet` (Chen *et al.* [2015]) is a convenient C++ library for learning deep NN models and is deployed as an R package. It integrates a high level symbolic language for quickly building customized models and loss functions, and automatically distributes calculations under CPUs or GPUs.
We fit the MAXENT model for every species of E_{50} with the recently released R package `maxnet` Phillips *et al.* [2017] and the vector input variables.
The LGL model was fitted with the package `mxnet`. The loss being convex, we used a simple **gradient descent algorithm** and stopped when the gradient norm was close to 0. The learning took around 2 minutes.
SNN, DNN and CNN models are fitted with the package `mxnet`: All model parameters were initialized with a uniform distribution $U(-0.03, 0.03)$, then we applied a **stochastic gradient descent algorithm with a momentum** of 0.9, a batch-size of 50 (batch samples are randomly chosen at each iteration), and an initial learning rate of 10^-8. The choice of initial learning rate was critical for a good optimization behavior. A too big learning rate can lead to training loss divergence, whereas when it is too small, learning can be very slow. We stopped when the average slope of the training mean loss had an absolute difference to 0 on the last 100 epochs inferior to 10^-3. The learning took approximately 5 minutes for SNN, 10 minutes for DNN, and 5 hours for CNN (independently of the version).

3.5 Evaluation metrics

Predictions are made for every species of E_{50} and several model performance metrics are calculated for each species and for two disjoints and randomly sampled subsets of sites: A train set (4781 sites) which is used for fitting all models and a test set (400 sites) which aims at testing models generalization capacities. Then, train and test metrics are averaged over the 50 species. The performance metrics are described in the following.

Mean loss Mean loss, just named loss in the following, is an important metric to consider because it is relevant regarding our ecological model and it is the objective function that is minimized during model training. The Mean loss of model m on species i and on sites $1, ..., K$ is:

$$\text{Loss}(m, i, \{1, ..., K\}) = \frac{1}{K} \sum_{k=1}^{K} \lambda_{m,\theta_i}(x_k) - y_k^i \log(\lambda_{m,\theta_i}(x_k))$$

In Table 3, the loss is averaged over species of E_{50}. Thus, in the case of a mono-response model, we averaged the metric over the 50 independently learned models. In the multi-response case, we averaged the metric over each species response of the same model.

Root Mean Square Error (Rmse). The root mean square error is a general error measure, which, in contrary to the previous one, is independent of the statistical model:

$$\text{Rmse}(m, i, \{1, ..., L\}) = \sqrt{\frac{1}{K} \sum_{k=1}^{K} \left(y_k^i - \lambda_{m,\theta_i}(x_k)\right)^2}$$

In Table 3, the average of the **Rmse** is computed over species of E_{50}. Mono-response models are treated as explained previously.

Accuracy on 10% densest quadrats (A10%DQ). It represents the proportion of sites which are in the top 10% of all sites in term of both real count and model prediction. This is a meaningful metric for many concrete scenarios where the regions of a territory have to be prioritized in terms of decision or actions related to the ecology of species. However, we have to define the last site ranked in the top 10% for real counts, which is problematic for some species, because of ex-aequo sites. That is why we defined the following procedure which adjust for each species the percentage of top cells, such that the metrics can be calculated and the percentage is the closest to 10%. Denoting y the vector of real counts over sites and $\hat{y}$ the model prediction :

$$A10\%DQ(\hat{y}, y) := \frac{N_{p\&c}(\hat{y}, y)}{N_c(y)} \tag{7}$$

Where $N_{p\&c}(\hat{y}, y)$ is the number of sites that are contained in the $N_c(y)$ highest values of both y and $\hat{y}$.

Calculation of $N_c(y)$: We order the sites by decreasing values of y and note C_k the value of the k^{th} site in this order. Noting $d := \text{round}(\dim(y)/10) = \text{round}(\dim(\hat{y})/10)$, as we are interested in the sites ranked in the 10% highest, if $C_d > C_{d+1}$ we simply set $N_c(y) = d$. Otherwise, if $C_d = C_{d+1}$ (ex-aequo exist for d^{th} position), we note **Sup** the position of the last site with value C_{d+1} and **Inf** the position of the first site with count C_d. The chosen rule is to take $N_c(y)$ such that $N_c(y) = \min(|\mathbf{Sup} - d|, |\mathbf{Inf} - d|)$.

4 Results

In the first part we describe and comment the main results obtained from performance metrics. Then, we illustrate and discuss qualitatively the behavior of models from the comparison of their predictions maps to real counts on some species.

4.1 Quantitative results analysis.

Table 3 provides the results obtained for all the evaluated models according to the 3 evaluation metrics. The four main conclusions that we can derive from that results are that (i) performances of LGL and mono-response DNN are lower than the one of MAXENT for all metrics, (ii) multi-response DNN outperforms SNN in every version and for all metrics, (iii) multi-response DNN outperforms MAXENT in test Rmse in every version, (iv) CNN outperforms all the other models, in every versions (CNN50, 200, 1000), and for all metrics.

According to these results, MAXENT shows the best performance amongst mono-response models. The low performance of the baseline LGL model is mostly due to underfitting. Actually, the evaluation metrics are not better on the training set than the test set. Its simple linear architecture is not able to exploit the complex relationships between environmental variables and observed abundance. DNN shows poor results as well in the mono-response version, but for another reason. We can see that its average training loss is very close to the minimum, which shows that the model is overfitting, *i.e.* it adjusts too much its parameters to predict exactly the training data, loosing its generalization capacity on test data.

However, for multi-responses versions, DNN performance increases importantly. DNN50 shows better results than MAXENT for the test Loss and test Rmse, while DNN200 and DNN1000 only show better Rmse. To go deeper, we notice that average and standard deviation of test rmse across E_{50} species goes down from DNN1 to DNN1000, showing that model becomes less sensitive to species data. Still, test loss and A10%DQ decrease, so there seems to be a performance trade-off between the different metrics as a side effect of the number of responses.

Whatever is the number of responses for SNN, the model is under-fitting and its performance are stable, without any big change between SNN50, 200, and 1K. This model doesn't get improvement from the use of training data on a larger number of species. Furthermore, its performance is always lower than DNN's, which shows that stacking hidden layers improves the model capacity to extract relevant features from the environmental data, keeping all others factors constant.

The superiority of the CNN whatever the metric is a new and important result for species distribution modeling community. Something also important to notice, as for DNN, is the improvement of its performance for te.Loss and te.Rmse when the number of species in output increases. Those results suggest that the multi-response regularization is efficient when the model is complex (DNN) or the input dimensionality is important (CNN) but has no interest for simple models and small dimension input (SNN). There should be an optimal compromise to find between model complexity, in term of number of hidden layers and neurons, and the number of species set as responses.

For the best model CNN1000, it is interesting to see if the performance obtained on E_{50} could be generalized at a larger taxonomic scale. Therefore, we computed the results of the CNN1000 on the 1,000 plant species used in output. Metrics values are :

- Test Loss = -1.275463 (minimum=-1.95)
- Test Rmse = 2.579596
- Test A10%DQ = 0.58

These additional results show that the average performance of CNN1000 on E_{1000} remains close from the one on E_{50}. Furthermore, one can notice the stability of performance across species. Actually, the test Rmse is lower than 3 for 710 of the 1000 species. That means that the learned environmental features are able to explain the distribution of a wide variety of species. According to the fact that French flora is compound of more than 6,000 plant species, the potential of improvement of CNN predictions based on the use of this volume of species could be really important and one of the first at the country level (which is costly in terms of time with classical approaches).

We can go a bit deeper in the understanding of model performances in terms of species types. Figure 3 provides for CNN1000 and MAXENT the test Rmse as a function of the species percentage of presence sites. It first illustrates the fact that all SDMs are negatively affected by an higher percentage of presence sites, even the best, which is a known issue amongst species distribution modelers. Actually, the two models have quite similar results for species with high percentage of presence sites. Moreover, CNN1000 is better for most species compared to Maxent, and especially for species with low percentage of presence sites. For those species, we also notice that CNN's variance of Rmse is much smaller than MAXENT: there is no hard failing for CNN.

4.2 Qualitative results analysis

As metrics are only summaries, visualization of predictions on maps can be useful to make a clearer idea of the magnitude and nature of models errors. We took a particular species with a spatially restricted distribution in France, *Festuca cinerea*, in order to illustrate some models behavior that we have found to be consistent across this kind of species in E_{50}. The maps of real counts and several models predictions for this species are shown on Figure 4. As we can note on map A of, *Festuca cinerea* was only observed in the south east part of the French territory. When we compare the different models prediction, CNN1000 (B) is the closest to real counts though DNN50 (C) and MAXENT (E) are not far. Clearly, DNN1000 (E) and LGL (F) are the models that over estimate the most the species presence over the territory. Another thing relative to DNN behavior can be noticed regarding Figure 4. DNN1000 has less peaky punctual predictions than DNN50, it looks weathered. This behavior is consistent across species and could explain that the A10%DQ metric is weak for DNN1000 (and DNN200) compared to DNN50: A contraction of predicted abundance values toward the mean will imply less risk on prediction errors but predictions on high abundance sites will be less distinguished from others.

# species in output	Archi.	Loss on E_{50}		Rmse on E_{50}		A10%DQ on E_{50}	
		tr.(min:-1.90)	te.(min:-1.56)	tr.	te.	tr.	te.
1	MAX	-1.43	**-0.862**	2.24	**3.18**	0.641	**0.548**
	LGL	-1.11	**-0.737**	3.28	**3.98**	0.498	**0.473**
	DNN	-1.62	**-0.677**	3.00	**3.52**	0.741	**0.504**
50	SNN	-1.14	**-0.710**	3.14	**3.05**	0.494	**0.460**
	DNN	-1.45	**-0.927**	2.94	**2.61**	0.576	**0.519**
	CNN	-1.82	**-0.991**	1.18	**2.38**	0.846	**0.607**
200	SNN	-1.09	**-0.690**	3.25	**3.03**	0.479	**0.447**
	DNN	-1.32	**-0.790**	5.16	**2.51**	0.558	**0.448**
	CNN	-1.59	**-1.070**	2.04	**2.34**	0.650	**0.594**
1K	SNN	-1.13	**-0.724**	3.27	**3.03**	0.480	**0.455**
	DNN	-1.38	**-0.804**	3.86	**2.50**	0.534	**0.467**
	CNN	-1.70	**-1.09**	1.51	**2.20**	0.736	**0.604**

Table 3: Train and test performance metrics averaged over all species of E_{50} for all tested models. For the single response class, the metric is averaged over the models learnt on each species.

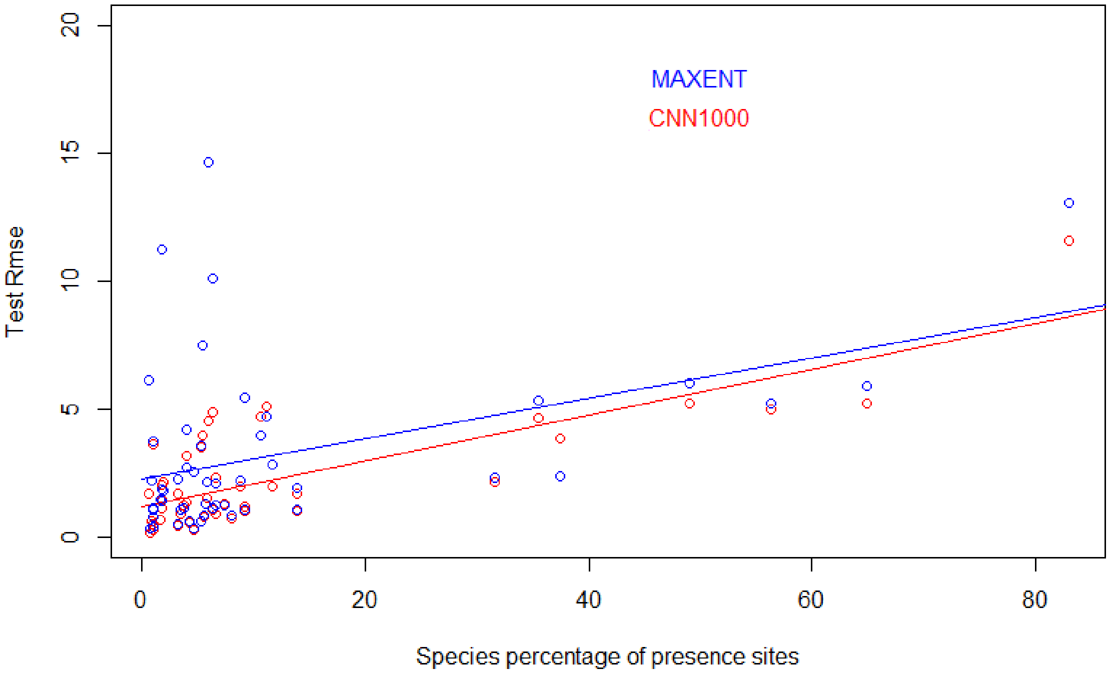


Figure 3: Test Rmse plotted versus percentage of presence sites for every species of E_{50}, with linear regression curve, in blue with Maxent model, in red with CNN1000.

Good results provided in Table 3 can hide bad behavior of the models for certain species. Indeed, when we analyze, on Figure 5, the distribution predicted by Maxent and CNN1000 for widespread species, such as *Anthriscus sylvestris* (L.) and *Ranunculus repens* L., we can notice a strong divergence with the INPN data. These 2 species, with the most important number of observation and percentage of presence sites in our experiment (see Table 1), are also the less well predicted by all models. For both species, MAXENT shows very smooth variations of predictions in space, which is sharply different from their real distribution. If CNN1000 seems to better fit to the presence area, it has still a lot of errors.

As last interesting remark, we note that a global maps analysis, on more species than the ones illustrated here, shows a consistent stronger false positive ratio for models under-fitting the data or with too much regularization (high number of responses in output).

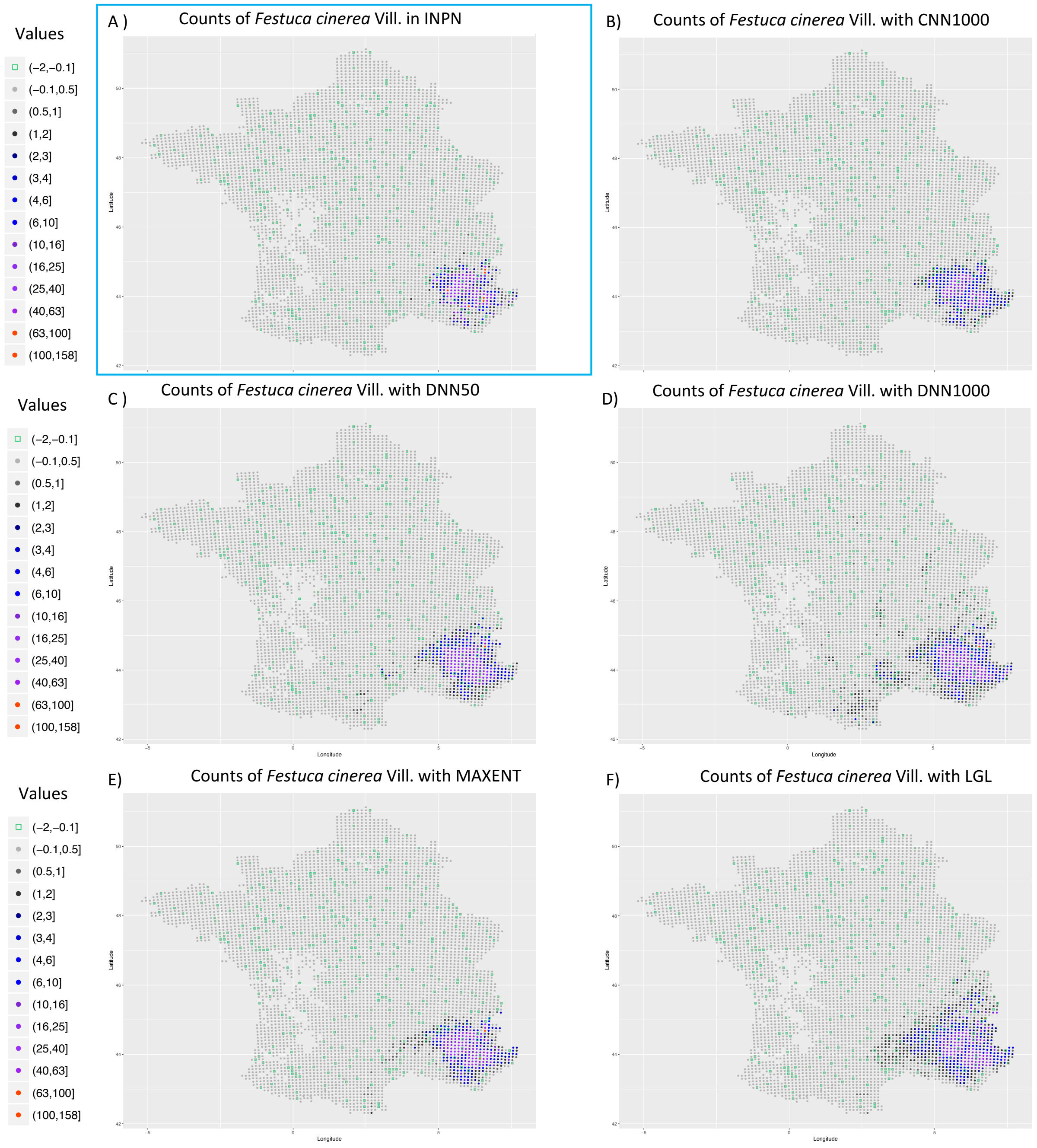

Values
(−2,−0.1]
(−0.1,0.5]
(0.5,1]
(1,2]
(2,3]
(3,4]
(4,6]
(6,10]
(10,16]
(16,25]
(25,40]
(40,63]
(63,100]
(100,158]
A) Counts of *Festuca cinerea* Vill. in INPN
B) Counts of *Festuca cinerea* Vill. with CNN1000
C) Counts of *Festuca cinerea* Vill. with DNN50
D) Counts of *Festuca cinerea* Vill. with DNN1000
E) Counts of *Festuca cinerea* Vill. with MAXENT
F) Counts of *Festuca cinerea* Vill. with LGL
Latitude
50
48
46
44
42
Longitude
−5
0
5

Figure 4: Real count of *Festuca cinerea* Vill. and prediction for 5 different models. Test sites are framed into green squares. A) Number of observations in INPN dataset, and geographic distribution predicted with B) CNN1000, C)DNN50, D)DNN1000, E) Maxent, F)LGL.

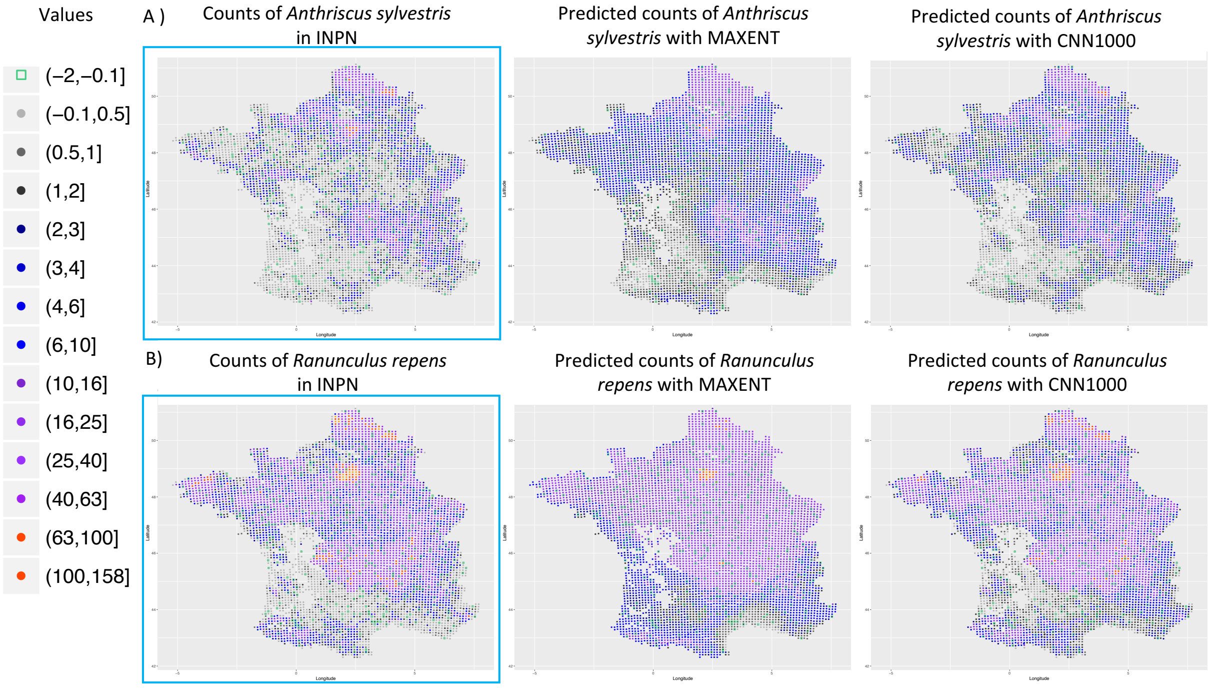

Values
(−2,−0.1]
(−0.1,0.5]
(0.5,1]
(1,2]
(2,3]
(3,4]
(4,6]
(6,10]
(10,16]
(16,25]
(25,40]
(40,63]
(63,100]
(100,158]
A)
Counts of *Anthriscus sylvestris* in INPN
Predicted counts of *Anthriscus sylvestris* with MAXENT
Predicted counts of *Anthriscus sylvestris* with CNN1000
B)
Counts of *Ranunculus repens* in INPN
Predicted counts of *Ranunculus repens* with MAXENT
Predicted counts of *Ranunculus repens* with CNN1000
Longitude
Latitude

Figure 5: A) Species occurrences in INPN dataset, and geographic distribution predicted with Maxent and CNN1000 for *Anthriscus sylvestris* (L.) Hoffm., B) Species occurrences in INPN dataset, and geographic distribution predicted with Maxent and CNN1000 for *Ranunculus repens* L.

5 Discussion

The performance increase with multi-responses models shows that multi-responses architecture are an efficient regularization scheme for NNs in SDM. It could be interesting to evaluate the performance impact of going multi-response on rare species where data are limited. We have systematically noticed false predicted presence for species that are not in the Mediterranean region. It could be due to a high representativity of species from this region in France. In the multi-response modeling, the Mediterranean species could favor prediction in this area through neurons activations rather than other areas where few species are present, inducing bias. Thus, the distributions complementarity between selected species could be an interesting subject for further research.

Even if our study presents promising results, there are still some open problems. A first one is related to the bias in the sampling process that is not taken into account in the model. Indeed, even if the estimation of bias in the learning process is difficult, this could strongly improve our results. Bias can be related to the facts that (i) some regions and difficult environments are clearly less inventoried than others (this can be seen with "empty region" in South western part of the country in Figure 4 and 5) ; (ii) some regions are much more inventoried than others, according to the human capacities of the National botanical conservatories, which have very different sizes ; (iii) some common and less attractive species for naturalists are not recorded, even if they are present in prospected areas, which is a bias due to the use of opportunistic observations rather than exhaustive count data.

In the NN models learning, there is still work to be done on quick automated procedure for tuning optimization hyper-parameters, especially the initial learning rate, and we are looking for a more suited stopping rule. On the other hand, in the case of models of species distributions, we can imagine to minimize the number of not null connections in the network, to make it more interpretable, and introduce an L1-type penalty on the network parameters. This is a potential important perspective of future works.

One imperfection in our modeling approach that induces biased distribution estimate is that the representation (vector or array of environmental variables) of a site is extracted from its geographic center. MAXENT, SNN and DNN models typically only integrate the central value of the environmental variables on each site, omitting the variability within the site. Instead of that, an unbiased data generation would sample for each site many representations uniformly in its spatial domain and in number proportional to its area. This way, it would provide richer information about sites and at the same time prevent NN model over-fitting by producing more data samples.

A deeper analysis of the behavior of the models according to the ecological preferences of the species could be of a strong interest for the ecological community. This study could allow to see dependences of the models to particular spatial patterns and/or environmental variables. Plus, it would be interesting to check if NN perform better when the species environmental niche is in the intersection of variables values that are far from their typical ranges into the study domain, which is something that MAXENT cannot fit.

Another interesting perspective for this work is the fact that, new detailed fine-scale environmental data become freely available with the development of the open data movement, in particular thanks to advances in remote sensing methods. Nevertheless, as long as we only have access to spatially degraded observations data at kilometer scales like here, it is difficult to consistently estimate the effect of variables that vary at high frequency in space. For example, the informative link between species abundance and land cover, proximity to fresh water or proximity to roads, is very blurred and almost lost. To overcome this difficulty, there is much hope in the high flow of finely geolocated species observations produced by citizen sciences programs for plant biodiversity monitoring like **Tela Botanica** [6] , **iNaturalist** [7] , **Naturgucker** [8] or **Pl@ntNet** [9]. From what we can see on the **GBIF** [10], the first three already have high resolution and large cover observation capacity: they have accumulated around three hundred thousand finely geolocated plant species observations just in France during last decade. Citizen programs in biodiversity sciences are currently developing worldwide. We expect them to reach similar volumes of observations to the sum of national museums, herbaria and conservatories in the next few years, while still maintaining a large flow of observations for the future. With good methods for dealing with sampling bias, those fine precision and large spatial scale data will make a perfect context for reaching the full potential of deep learning SDM methods. Thus, NN methods could be a significant tool to explore biodiversity data and extract new ecological knowledge in the future.

6 Conclusion

This study is the first one evaluating the potential of the deep learning approach for species distributions modeling. It shows that DNN and CNN models trained on 50 plant species of French flora clearly overcomes classical approaches, such as Maxent and LGL, used in ecological studies. This result is promising for future ecological studies developed in collaboration with naturalists expert. Actually, many ecological studies are based on models that do not take into account spatial patterns in environmental variables. In this paper, we show for a random set of 50 plant species of the French flora, that CNN and DNN, when learned as multi-species output models, are able to automatically learn

[6] http://www.tela-botanica.org/site:accueil

[7] https://www.inaturalist.org/

[8] http://naturgucker.de/enjoynature.net

[9] https://plantnet.org/en/

[10] https://www.gbif.org/

non-linear transformations of input environmental features that are very relevant for every species without having to think a priori about variables correlation or selection. Plus, CNN can capture extra information contained in spatial patterns of environmental variables in order to surpass other classical approaches and even DNN. We also did show that the models trained on higher number of species in output (from 50 to 1000) stabilize predictions across species or even improve them globally, according to the results that we got for several metrics used to evaluate them. This is probably one the most important outcome of our study. It opens new opportunities for the development of ecological studies based on the use of CNN and DNN (e.g. the study of communities). However, deeper investigations regarding specific conditions for models efficiency, or the limits of interpretability NN predictions should be conducted to build richer ecological models.

References

Berman, Mark, & Turner, T Rolf. 1992. Approximating point process likelihoods with GLIM. *Applied Statistics*, 31–38.

Chen, Tianqi, Li, Mu, Li, Yutian, Lin, Min, Wang, Naiyan, Wang, Minjie, Xiao, Tianjun, Xu, Bing, Zhang, Chiyuan, & Zhang, Zheng. 2015. Mxnet: A flexible and efficient machine learning library for heterogeneous distributed systems. *arXiv preprint arXiv:1512.01274*.

Dutrève B., Robert S. 2016. INPN - Données flore des CBN agrégées par la FCBN. Version 1.1. *SPN - Service du Patrimoine naturel, Muséum national d'Histoire naturelle, Paris.*

Fithian, William, & Hastie, Trevor. 2013. Finite-sample equivalence in statistical models for presence-only data. *The annals of applied statistics*, **7**(4), 1917.

Friedman, Jerome H. 1991. Multivariate adaptive regression splines. *The annals of statistics*, 1–67.

Goodfellow, Ian, Bengio, Yoshua, & Courville, Aaron. 2016. *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Hastie, Trevor, & Tibshirani, Robert. 1986. Generalized Additive Models. *Statistical Science*, **1**(3), 297–318.

Hutchinson, G Evelyn. 1957. Cold spring harbor symposium on quantitative biology. *Concluding remarks*, **22**, 415–427.

Ioffe, Sergey, & Szegedy, Christian. 2015. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *Pages 448–456 of: International Conference on Machine Learning.*

Karger, Dirk N, Conrad, Olaf, Böhner, Jürgen, Kawohl, Tobias, Kreft, Holger, Soria-Auza, Rodrigo W, Zimmermann, Niklaus E, Linder, H Peter, & Kessler, Michael. 2016a. CHELSA climatologies at high resolution for the earth\'s land surface areas (Version 1.1).

Karger, Dirk Nikolaus, Conrad, Olaf, Böhner, Jürgen, Kawohl, Tobias, Kreft, Holger, Soria-Auza, Rodrigo Wilber, Zimmermann, Niklaus, Linder, H Peter, & Kessler, Michael. 2016b. Climatologies at high resolution for the earth's land surface areas. *arXiv preprint arXiv:1607.00217*.

Krizhevsky, Alex, Sutskever, Ilya, & Hinton, Geoffrey E. 2012. Imagenet classification with deep convolutional neural networks. *Pages 1097–1105 of: Advances in neural information processing systems.*

Leathwick, JR, Elith, J, & Hastie, T. 2006. Comparative performance of generalized additive models and multivariate adaptive regression splines for statistical modelling of species distributions. *Ecological modelling*, **199**(2), 188–196.

LeCun, Yann, *et al.* . 1989. Generalization and network design strategies. *Connectionism in perspective*, 143–155.

Lek, Sovan, Delacoste, Marc, Baran, Philippe, Dimopoulos, Ioannis, Lauga, Jacques, & Aulagnier, Stéphane. 1996. Application of neural networks to modelling nonlinear relationships in ecology. *Ecological modelling*, **90**(1), 39–52.

Nair, Vinod, & Hinton, Geoffrey E. 2010. Rectified linear units improve restricted boltzmann machines. *Pages 807–814 of: Proceedings of the 27th international conference on machine learning (ICML-10).*

P Anderson, Robert, Dudík, Miroslav, Ferrier, Simon, Guisan, Antoine, J Hijmans, Robert, Huettmann, Falk, R Leathwick, John, Lehmann, Anthony, Li, Jin, G Lohmann, Lucia, *et al.* . 2006. Novel methods improve prediction of species' distributions from occurrence data. *Ecography*, **29**(2), 129–151.

Panagos, Panos. 2006. The European soil database. *GEO: connexion*, **5**(7), 32–33.

Panagos, Panos, Van Liedekerke, Marc, Jones, Arwyn, & Montanarella, Luca. 2012. European Soil Data Centre: Response to European policy support and public data requirements. *Land Use Policy*, **29**(2), 329–338.

Phillips, Steven J, & Dudík, Miroslav. 2008. Modeling of species distributions with Maxent: new extensions and a comprehensive evaluation. *Ecography*, **31**(2), 161–175.

Phillips, Steven J, Dudík, Miroslav, & Schapire, Robert E. 2004. A maximum entropy approach to species distribution modeling. *Page 83 of: Proceedings of the twenty-first international conference on Machine learning.* ACM.

Phillips, Steven J, Anderson, Robert P, & Schapire, Robert E. 2006. Maximum entropy modeling of species geographic distributions. *Ecological modelling*, **190**(3), 231–259.

Phillips, Steven J, Anderson, Robert P, Dudík, Miroslav, Schapire, Robert E, & Blair, Mary E. 2017. Opening the black box: an open-source release of Maxent. *Ecography.*

Rumelhart, David E, Hinton, Geoffrey E, Williams, Ronald J, *et al.* . 1988. Learning representations by back-propagating errors. *Cognitive modeling*, **5**(3), 1.

Thuiller, Wilfried. 2003. BIOMOD–optimizing predictions of species distributions and projecting potential future shifts under global change. *Global change biology*, **9**(10), 1353–1362.

Van Liedekerke, M, Jones, A, & Panagos, P. 2006. ESDBv2 Raster Library-a set of rasters derived from the European Soil Database distribution v2. 0. *European Commission and the European Soil Bureau Network, CDROM, EUR*, **19945**.

Ward, Gill, Hastie, Trevor, Barry, Simon, Elith, Jane, & Leathwick, John R. 2009. Presence-only data and the EM algorithm. *Biometrics*, **65**(2), 554–563.

Zomer, Robert J, Bossio, Deborah A, Trabucco, Antonio, Yuanjie, Li, Gupta, Diwan C, & Singh, Virendra P. 2007. *Trees and water: smallholder agroforestry on irrigated lands in Northern India.* Vol. 122. IWMI.

Zomer, Robert J, Trabucco, Antonio, Bossio, Deborah A, & Verchot, Louis V. 2008. Climate change mitigation: A spatial analysis of global land suitability for clean development mechanism afforestation and reforestation. *Agriculture, ecosystems & environment*, **126**(1), 67–80.