Guidelines for the supervised learning of species interactions

Timothée Poisot [1,2]

[1] Université de Montréal; [2] Québec Centre for Biodiversity Sciences

Correspondance to:
Timothée Poisot — timothee.poisot@umontreal.ca

Keywords:
species interaction networks
binary classifiers
machine learning
regression
supervised learning

1. The prediction of species interaction networks is gaining momentum as a way to circumvent limitations in data volume. Yet, ecological networks are challenging to predict because they are typically small and sparse. Dealing with extreme class imbalance is a challenge for most binary classifiers, and there are currently no guidelines as to how predictive models can be trained.
2. Using simple mathematical arguments and numerical experiments in which a variety of classifiers (for supervised learning) are trained on simulated networks, we develop a series of guidelines related to the choice of measures to use for model selection, and the degree of unbiasing to apply to the training dataset.
3. Classifier accuracy and the ROC-AUC are not informative measures for the performance of interaction prediction. PR-AUC is a fairer assesment of performance. In some cases, even standard measures can lead to selecting a more biased classifier because the effect of connectance is strong. The amount of correction to apply to the training dataset depends as a function of the classifier and the network connectance.
4. These results reveal that training machines to predict networks is a challenging task, and that in virtually all cases, the composition of the training set needs to be experimented on before performing the actual training. We discuss these consequences in the context of the low volume of data.

example on diagnostic test: rare events are hard to detect even with really good models

summary of model challenges for networks - Strydom et al. (2021) importance of drawing on traits + validation is challenging - Whalen et al. (2021) machine learning from genomics

Binary classifiers are usually assessed by measuring properties of their confusion matrix, *i.e.* the contingency table reporting true/false positive/negative hits. The same approach is used to evaluate *e.g.* species occurrence models (Allouche et al., 2006). A confusion matrix is laid out as

$$\begin{pmatrix} \text{tp} & \text{fp} \\ \text{fn} & \text{tn} \end{pmatrix},$$

wherein tp is the number of interactions predicted as positive, tn is the number of non-interactions predicted as negative, fp is the number of non-interactions predicted as positive, and fn is the number of interactions predicted as negative. Almost all measures based on the confusion matrix express rates of error or success as proportions, and therefore the values of these components matter in a *relative* way.

At a coarse scale, a classifier is *accurate* when the trace of the matrix divided by the sum of the matrix is close to 1, with other measures focusing on different ways in which the classifer is wrong.

list of problems to solve - baseline values and response to bias - effect of training set bias on performance - which models need the least amount of interactions to work

summary of the results

1

Baseline values

1.1. Confusion matrix with skill and bias In this section, we will assume a network of connectance ρ, *i.e.* having ρS^2 interactions (where S is the species richness), and $(1-\rho)S^2$ non-interactions. Therefore, the vector describing the *true* state of the network is a column vector $\mathbf{o}^T = [\rho(1-\rho)]$ (we can safely drop the S^2 terms, as we will work on the confusion matrix, which ends up expressing *relative* values).

In order to write the values of the confusion matrix for a hypothetical classifier, we need to define two characteristics: its skill, and its bias. Skill, here, refers to the propensity of the classifier to get the correct answer (*i.e.* to assign interactions where they are, and to not assign them where they are not). A no-skill classifier guesses at random, *i.e.* it will guess interactions with a probability ρ. The predictions of a no-skill classifier can be expressed as a row vector $\mathbf{p} = [\rho(1-\rho)]$. The confusion matrix $\mathbf{M}$ for a no-skill classifier is given by the element-wise product of these vectors $\mathbf{o} \odot \mathbf{p}$, *i.e.*

$$\mathbf{M} = \begin{pmatrix} \rho^2 & \rho(1-\rho) \\ (1-\rho)\rho & (1-\rho)^2 \end{pmatrix}.$$

In order to regulate the skill of this classifier, we can define a skill matrix $\mathbf{S}$ with diagonal elements equal to s, and off-diagonal elements equal to $(1-s)$, and re-express the skill-adjusted confusion matrix as $\mathbf{M} \odot \mathbf{S}$, *i.e.*

$$\begin{pmatrix} \rho^2 & \rho(1-\rho) \\ (1-\rho)\rho & (1-\rho)^2 \end{pmatrix} \odot \begin{pmatrix} s & (1-s) \\ (1-s) & s \end{pmatrix}.$$

Note that when $s = 0$, $\text{Tr}(\mathbf{M}) = 0$ (the classifier is *always* wrong), when $s = 0.5$, the classifier is no-skill and guesses at random, and when $s = 1$, the classifier is perfect.

The second element we can adjust in this hypothetical classifier is its bias, specifically its tendency to over-predict interactions. Like above, we can do so by defining a bias matrix $\mathbf{B}$, where interactions are over-predicted with probability b, and express the final classifier confusion matrix as $\mathbf{M} \odot \mathbf{S} \odot \mathbf{B}$, *i.e.*

$$\begin{pmatrix} \rho^2 & \rho(1-\rho) \\ (1-\rho)\rho & (1-\rho)^2 \end{pmatrix} \odot \begin{pmatrix} s & (1-s) \\ (1-s) & s \end{pmatrix} \odot \begin{pmatrix} b & b \\ (1-b) & (1-b) \end{pmatrix}.$$

The final expression for the confusion matrix in which we can regulate the skill and the bias is

$$\mathbf{C} = \begin{pmatrix} s \times b \times \rho^2 & (1-s) \times b \times \rho(1-\rho) \\ (1-s) \times (1-b) \times (1-\rho)\rho & s \times (1-b) \times (1-\rho)^2 \end{pmatrix}.$$

1.2. What are the baseline values of performance measures? In this section, we will change the values of b abd s for a given value of ρ, and see how the values of common performance measures for binary classification are affected. Specifically, we will focus on four quantities: the accuracy ((tp + tn)/(tp + tn + fp + fn)), the *balanced* accuracy (tp/(2(tp + fn)) + tn/(2(tn + fp))), Youden's J (tp/(tp + fn) + tn/(tn + fp) − 1), and the F_1 score (2tp/(2tp + fp + fn)).

Justification of why these 4

Assuming a no-skill unbiased classifier (*i.e.* $\mathbf{C} = \mathbf{M}$), the accuracy is $\rho^2 + (1-\rho)^2$, the balanced accuracy is 0.5, Youden's J is 0, and $F_1 = \rho$. In other words, given a connectance $\rho = 0.05$, we expect that a

classifier guessing at random would still achieve an accuracy of 0.905; for a connectance of $\rho = 0.01$, this accuracy *increases* to over 0.98. In other words, networks with fewer interactions have inherently higher accuracy, because it is easy to predict the overwheling majority of non-interactions right.

In order to examine how these values change w.r.t. the skill and bias, we performed a grid exploration of the values of s (from 0 to 1), and of logit(b), and visualize the result for a connectance of **TODO**.

1.3. What are the baseline values of thresholding performance?

ROC-AUC v. PR-AUC, vary s to simulate a classifier ranging from no-skill to perfect

2 Numerical experiments

In the following section, we will generate random networks, and train four binary classifiers (as well as an ensemble model using the sum of the outputs) on 30% of the interaction data. Networks are generated by picking random generality g and vulnerability v traits for $S = 200$ species uniformly on the unit interval, and assigning an interaction from species i to species j if $0.2g_i - \xi \leq v_j \leq 0.2g_i + \xi$, where ξ is a constant regulating the connectance of the networks, and varies uniformly in $[5\times10^{-3}, 10^{-1}]$. This model gives fully interval networks that are close analogues to the niche model (Williams & Martinez, 2000), but has the benefit of only relying on two features (g_i, v_j), and having the exact same rule for all interactions. It is, therefore, a simple case which most classifiers should be able to learn.

The training sample is composed of 30% of the 4×10^4 possible entries in the network, *i.e.* $n = 12000$. Out of these interactions, we pick a proportion ν (the training set bias) to be positive, so that the training set has νn interactions, and $(1-\nu)n$ non-interactions. We vary ν uniformly in $]0, 1[$. This allows to evaluate how the measures of binary classification performance respond to artificially rebalanced dataset for a given network connectance. Note that both ξ and ν are sampled from a distribution rather than being picked on a grid; this is because there is no direct relationship between the value of ξ and the connectance of the simulated network, and therefore the precise value of ξ is not relevant for the analysis of the results.

The dataset used for numerical experiments is composed of 20000 such (ξ, ν) pairs, on which four learners are trained: a decision tree regressor, a boosted regression tree, a ridge regressor, and a random forest regressor. All models were taken from the `MLJ.jl` package (Blaom et al., 2020; Blaom & Vollmer, 2020) in Julia 1.7 (Bezanson et al., 2017). In order to pick the best adjacency matrix for a given learner, we performed a thresholding approach using 500 steps on predictions from the testing set, and picking the threshold that maximized Youden's informedness, which is usually the optimized target for imbalanced classification. During the thresholding step, we measured the area under the receiving-operator characteristic (ROC-AUC) and precision-recall (PR-AUC) curves, as measures of overall performance over the range of returned values. We report the ROC-AUC and PR-AUC, as well as a suite of other measures as introduced in the next section, for the best threshold. The ensemble model was generated by summing the predictions of all component models on the testing set (ranged in $[0, 1]$), then put through the same thresholding process. The complete code to run the simulations is given as an appendix.

After the simulations were completed, we removed all runs (*i.e.* pairs of ξ and ν) for which at least one of the following conditions was met: the accuracy was 0, the true positive or true negative rates were 0, the connectance was larger than 0.2. This removes both the obviously failed model runs, and the networks that are more densely connected compared to the connectance of empirical food webs (and are therefore less difficult to predict, being less imbalanced).

2.1. Effect of training set bias on performance

2.2. Required amount of positives to get the best performance

3

Guidelines for prediction

References

Allouche, O., Tsoar, A., & Kadmon, R. (2006). Assessing the accuracy of species distribution models: Prevalence, kappa and the true skill statistic (TSS). *Journal of Applied Ecology*, *43*(6), 1223–1232. https://doi.org/10.1111/j.1365-2664.2006.01214.x

Bezanson, J., Edelman, A., Karpinski, S., & Shah, V. (2017). Julia: A Fresh Approach to Numerical Computing. *SIAM Review*, *59*(1), 65–98. https://doi.org/10.1137/141000671

Blaom, A. D., Kiraly, F., Lienart, T., Simillides, Y., Arenas, D., & Vollmer, S. J. (2020). MLJ: A Julia package for composable machine learning. *Journal of Open Source Software*, *5*(55), 2704. https://doi.org/10.21105/joss.02704

Blaom, A. D., & Vollmer, S. J. (2020, December 31). *Flexible model composition in machine learning and its implementation in MLJ*. http://arxiv.org/abs/2012.15505

Strydom, T., Catchen, M. D., Banville, F., Caron, D., Dansereau, G., Desjardins-Proulx, P., Forero-Muñoz, N. R., Higino, G., Mercier, B., Gonzalez, A., Gravel, D., Pollock, L., & Poisot, T. (2021). A roadmap towards predicting species interaction networks (across space and time). *Philosophical Transactions of the Royal Society B: Biological Sciences*, *376*(1837), 20210063. https://doi.org/10.1098/rstb.2021.0063

Whalen, S., Schreiber, J., Noble, W. S., & Pollard, K. S. (2021). Navigating the pitfalls of applying machine learning in genomics. *Nature Reviews Genetics*, 1–13. https://doi.org/10.1038/s41576-021-00434-9

Williams, R., & Martinez, N. (2000). Simple rules yield complex food webs. *Nature*, *404*, 180–183. http://userwww.sfsu.edu/