Hyperparameter space analysis via interpretable meta-modeling

Miroslav Cibula, Kristína Malinovská, Igor Farkaš

Faculty of Mathematics, Physics and Informatics
Comenius University Bratislava
cibula25@uniba.sk, kristina.malinovska@fmph.uniba.sk, igor.farkas@fmph.uniba.sk

Abstract

The performance of many machine learning models is highly dependent on the setup of their hyperparameters. The search for the best-performing hyperparameter configuration can be performed using various uninformed or informed search methods. In the present study, our aim is to showcase an analytical method that reveals the individual effects of given hyperparameters on the performance of a model and to use this information to guide the search for an optimal hyperparameter configuration. We implement the proposed method by meta-modeling the model's performance as a function of the hyperparameter configuration using a multilayer perceptron network and subsequently analyzing the meta-model using feature-attribution methods. To demonstrate the feasibility of our method, we apply it to the hyperparameter space analysis of the bio-inspired UBAL model, where the setup of its unique hyperparameters is absolutely crucial for its performance in a given task.

1 Introduction and related work

In the area of supervised machine learning (ML), the primary objective of all methods is to estimate some mapping $f\colon \mathcal{X} \to \mathcal{Y}$ from a dataset of observations $\mathcal{D} = \left[\left(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}\right), \ldots, \left(\mathbf{x}^{(n)}, \mathbf{y}^{(n)}\right)\right]$, with $\mathbf{x}^{(i)} \in \mathcal{X}$ and $\mathbf{y}^{(i)} \in \mathcal{Y}$, using the inductive learning process. An ML method can be formally defined as an inducer function (Bischl et al., 2023) $\mathcal{I}\colon (\mathcal{D}, \boldsymbol{\zeta}) \mapsto \hat{f}_\theta$, taking training dataset $\mathcal{D}$, configured by a set of hyperparameters (HP) $\boldsymbol{\zeta}$, and producing an estimator $\hat{f}_\theta \in \mathcal{H}$, with θ denoting its parametrization and $\mathcal{H}$ a hypothesis space. Thus, $\boldsymbol{\zeta}$ determines which subspace of $\mathcal{H}$ is explored and significantly affects the performance of the produced estimator $\hat{f}_\theta$. The optimal configuration for solving a task specified by a dataset is not usually trivial to infer and must be found empirically. This problem is referred to as an HP optimization (HPO) problem.

The HPO methods generally consist of sampling some set of HP configurations and empirically evaluating them based on the performance of the respective generated models. This is the case with early methods such as a grid or random search (Bergstra and Bengio, 2012). For an overview of more sophisticated HPO methods, refer to the recent work of Bischl et al. (2023).

However, in this research, we do not aim to develop an HPO method per se; instead, we propose a means to analyze an HP space in terms of the importance of individual HPs. Since individual HPs are usually not equally important for finding the best estimator, identifying their individual importance can help guide the HPO and reduce the search space.

This problem was researched by Hutter et al. (2014), who proposed the identification of HP importance by meta-modeling the performance of a model using random forests trained on observations from a prior Bayesian optimization (Hutter et al., 2011) of the model and analyzing the predictions of the meta-model using functional ANOVA. Similarly, Sun et al. (2019) assess the HP importance of ML models by analyzing observations from a Bayesian optimization using their proposed N-RReliefF algorithm.

In the present paper, we introduce a method for assessing the importance of individual HPs by meta-modeling the performance of an ML model as a function of its HP configuration using a multilayer perceptron (MLP) and analyzing it using the SHAP method (Lundberg and Lee, 2017), a feature attribution method (Section 3). We conduct a case study applying this method to our biologically motivated UBAL model and analyzing its HP space (Section 4).

2 UBAL model

UBAL, or Universal Bidirectional Activation-based Learning model (Malinovská et al., 2019), is a biologically motivated alternative to classical error-backpropagation learning, which is known to be biologically implausible (O'Reilly et al., 2012). It is mainly inspired by the recirculation algorithm and by Generalized Recirculation, which is an adaptation of Contrastive Hebbian Learning (CHL) (O'Reilly et al., 2012). UBAL is a heteroencoder model that maintains separate weight matrices $\boldsymbol{W}$ and $\boldsymbol{M}$ for two different directions of activation propagation between inputs and outputs, as is the case in the brain. It also involves a unique *echo* mechanism that bounces back internal activations within the model, enabling unsupervised and self-supervised learning. Since UBAL is essentially a heteroassociator with self-loops, it is able to master various tasks, such as association (memory), denoising,

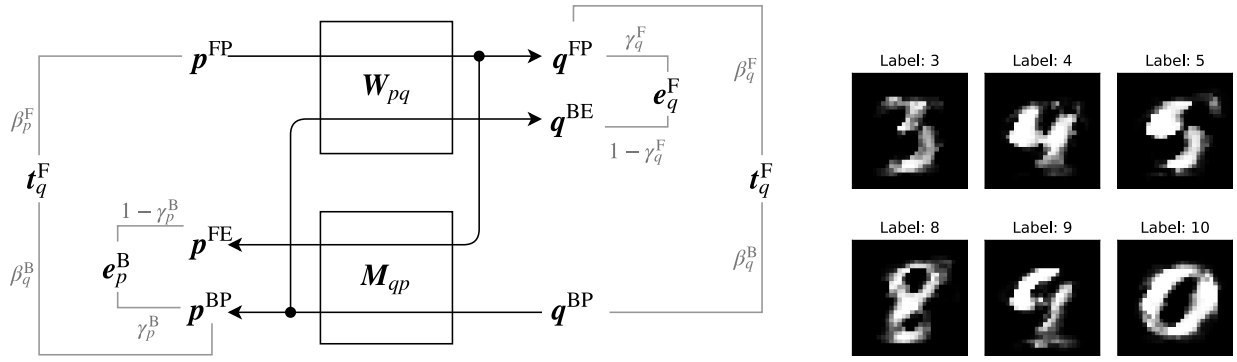

Fig. 1: Left: activation propagation and learning rule terms for connected layers $\boldsymbol{p}$ and $\boldsymbol{q}$ in UBAL. The activation variables in the net are created in 4 different activation phases: forward prediction (FP), forward echo (FE), backward prediction (BP), and backward echo (BE). Right: MNIST digits generated by UBAL.

and classification, unlike other neural network models, which require specifically designed learning rules to perform different tasks.

Learning in the UBAL model takes a form similar to GeneRec and CHL, with a Hebbian component (the product of pre-synaptic and post-synaptic activations) and an anti-Hebbian component as a difference between the network's estimate and the introduced target-related activations. In UBAL, however, these terms are formed by its internal variables as well as external inputs propagated through the network. In short, the β and γ HPs control the proportion of supervised and self-supervised components in the model's learning. Fig. 1 (left) illustrates these HPs and their influence on the model in two connected layers $\boldsymbol{p}$ and $\boldsymbol{q}$.

UBAL has been shown to perform well compared to related models in the classical handwritten digit classification benchmark (MNIST) (LeCun et al., 1998), without using any regularization techniques (Malinovská et al., 2019). One of the most intriguing properties of the model is that it demonstrates generative properties as an emergent phenomenon (Malinovská and Farkaš, 2021). As a heteroencoder, apart from classifying the digits, UBAL also makes projections of those digits in its input layer, without this being a training objective. This could be understood as the network imagination of the learned classes, as shown in Fig. 1 (right). Our preliminary results suggest that these images vary among network initializations and differ from the computed averages of all images in the dataset.

3 Our method

Given a dataset characterizing an ML task and an ML method providing estimators for this task, to analyze the HP space, we first obtain observations of the performance of every trained estimator configured from a sampled set of HP configurations, similarly to all the previous methods listed in Section 1. The sampling can be done by any HP search method. After training an

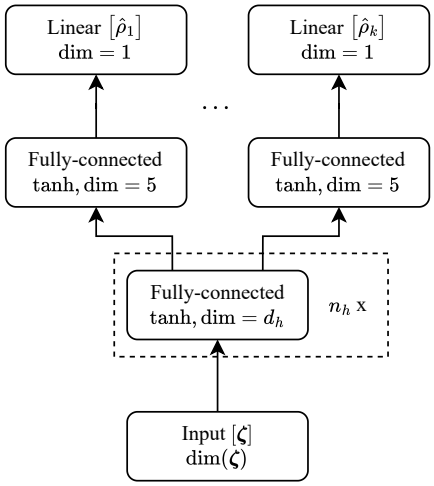

Fig. 2: General architecture of the meta-model implemented as an MLP. n_h and d_h denote the number of hidden common layers and their dimensionality, respectively.

estimator, its prediction is evaluated using a chosen performance metric, ρ. Subsequently, we construct a meta-model $\mathcal{M}\colon \boldsymbol{\zeta}_i \mapsto r$ modeling the performance based on HP configuration with r denoting the performance according to a performance metric ρ. The meta-model $\mathcal{M}$ is implemented as an MLP regressor (Fig. 2) and is trained on the observations collected in the first stage.

The trained meta-model is analyzed using the SHAP method (Lundberg and Lee, 2017). As a feature attribution method, SHAP identifies the effect (importance) of individual input features on the prediction of output features. Consequently, the SHAP analysis of the meta-model reveals the importance of an individual HP ζ_i (input feature) for the prediction of a performance metric ρ (output feature). If an HP ζ_i is of high importance for a performance metric ρ, its manipulation observed in the data and learned by $\mathcal{M}$ will cause a significant change in the value of ρ. This indicates that ζ_i contributes substantially to the prediction of ρ, and this

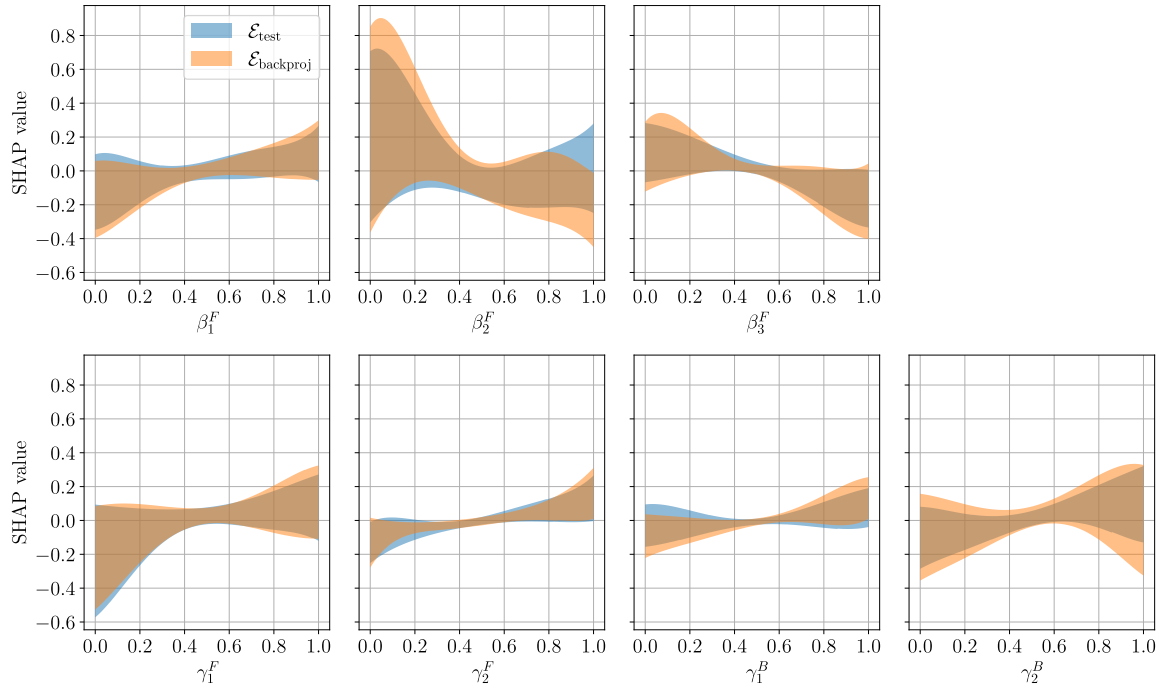

Fig. 3: Comparison of the SHAP values of the individual HPs as functions of their respective values inferred from the meta-models of the $\mathcal{E}_{\text{test}}$ and $\mathcal{E}_{\text{backproj}}$ ensembles.

contribution is quantified by the SHAP method as a high SHAP value of the ζ_i feature.

The presented method is analogous to the one proposed in our previous work (Cibula et al., 2024), in which we modeled causal relationships in a robotic environment from raw observations (analogous to modeling the behavior of an ML model) and extracted them via SHAP analysis.

4 Hyperparameters of UBAL

To verify the feasibility of the proposed method, we applied it to the analysis of the HP space of the UBAL model (Malinovská et al., 2019) for the MNIST task (LeCun et al., 1998). The base model consisted of one hidden layer with 1200 neurons and all layers being σ-activated. All weight matrices were initialized from $\mathcal{N}(0.0, 0.5)$. The learning rate of the model was set to $\lambda = 0.1$, and each model was trained for 30 epochs. We have considered the values of the HPs of β^F for each layer of the network, and γ^F and γ^B for each weight matrix $\boldsymbol{W}$ and $\boldsymbol{M}$. Since β^B HPs are equal to $1 - \beta^F$, we do not have to explore those, rendering 7 different special HPs to optimize. Other HPs of the model have been set based on previous rich and long-term experimentation with the UBAL model. It is important to note that β and γ are unique HPs of the UBAL model, and their setup is crucial for the model being able to perform a task. Their particular setups allow UBAL to master qualitatively different tasks, such as classification versus auto-association.

To collect a sufficient number of observations of the performance of HP configurations, we explored the HP space using a random search (Bergstra and Bengio, 2012). We evaluate the trained model instances in terms of testing and back-projection accuracy. The back-projection accuracy refers to the accuracy of the model for its own projected images, given the digit code (one-hot encoding). This performance measure indicates how well the model generates images, at least from its own point of view.

The data collected was used to train two ensembles of meta-models: $\mathcal{E}_{\text{test}}$ for modeling the testing accuracy and $\mathcal{E}_{\text{backproj}}$ for back-projection accuracy. Each meta-model in both ensembles was analyzed by the Deep SHAP method (Lundberg and Lee, 2017). SHAP natively quantifies the importance of HPs as a function of their values (Fig. 3).

In Fig. 3, it can be observed that β_2^F is the HP with the most impact, as for $\beta_2^F \lesssim 0.4$ it significantly positively contributes to the prediction of both testing and back-projection accuracies. On the other hand, $\beta_2^F \gtrsim 0.4$ negatively affects the prediction of the back-projection accuracy. The influence of $\beta_2^F \gtrsim 0.4$ on the testing accuracy prediction slightly diverges in a positive trend from the trend of $\mathcal{E}_{\text{backproj}}$. This indicates that while the configurations of $\beta_2^F \gtrsim 0.4$ diminish the back-projection accuracy, they positively contribute to the testing accuracy. A similar deduction could be made for other HPs as well.

These findings suggest how the subsequent HPO of the model, with respect to testing accuracy, for instance,

could be constrained to reduce its complexity. HPs β_1^F, γ_1^B, and γ_2^B could be excluded from the HPO as their importance is low throughout their range. For the rest of HPs, the HPO could be limited only to the ranges yielding positive contributions (e.g., $\beta_2^F, \beta_3^F \lesssim 0.4$, $\gamma_1^F, \gamma_2^F \gtrsim 0.6$). Such a constrained HPO would further explore the subspaces with strongly interacting effects, and it would either yield an optimal HP configuration or the observations obtained during it could be analyzed using the method once again to explain the interacting effects further.

Although our method identifies the importance of individual HPs and is able to reveal partial dynamics of their effects on the defined performance metrics, it fails to fully decompose the interacting effects of multiple HPs. This limitation manifests itself as a relatively large variance in SHAP values of some HP (e.g., for $\beta_2^F \lesssim 0.4$). As the effects of some other HPs causally collide with a particular HP ζ_i, their manipulation results in different performance measurements for the same ζ_i values.

5 Conclusion

We introduce a method for analyzing the HP space of any supervised HP-configurable estimator by modeling its performance as a function of its HP configuration. We study this meta-model using feature attribution methods to extract functional relationships between individual HPs (input) and the performance (output). As demonstrated in the case study of the UBAL model, the method is able to infer the absolute importance of individual HPs with respect to the modeled metric, as well as their importance relative to their values. However, the presented method has some limitations; in particular, it does not completely decompose the interacting effects of multiple HPs. For future work, the performance and data efficiency of this method should also be explored when applied to a wider set of models.

Acknowledgement

Supported by the project VEGA 1/0373/23. Research results were partially obtained using the computational resources procured in the national EU-funded project 311070AKF2 National Competence Centre for High Performance Computing. We also thank the Slovak Society for Cognitive Science (SSKV)[1] for their support.

References

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(10):281–305.

Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Thomas, J., Ullmann, T., Becker, M., Boulesteix, A., Deng, D., and Lindauer, M. (2023). Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges. *WIREs Data Mining and Knowledge Discovery*, 13(2):e1484.

Cibula, M., Kerzel, M., and Farkaš, I. (2024). Learning low-level causal relations using a simulated robotic arm. In *Artificial Neural Networks and Machine Learning – ICANN 2024*, pages 285–298, Cham. Springer Nature Switzerland.

Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2011). Sequential model-based optimization for general algorithm configuration. In *Learning and Intelligent Optimization*, pages 507–523, Berlin, Heidelberg. Springer Berlin Heidelberg.

Hutter, F., Hoos, H. H., and Leyton-Brown, K. (2014). An efficient approach for assessing hyperparameter importance. In *Proceedings of the 31st International Conference on Machine Learning*, volume 32, pages 754–762. PMLR.

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, volume 30 of *NIPS'17*, pages 4768–4777.

Malinovská, K. and Farkaš, I. (2021). Generative properties of Universal Bidirectional Activation-based Learning. In *Artificial Neural Networks and Machine Learning – ICANN 2021*, pages 80–83. Springer Nature Switzerland AG.

Malinovská, K., Malinovský, Ľ., Krsek, P., Kraus, S., and Farkaš, I. (2019). UBAL: A universal bidirectional activation-based learning rule for neural networks. In *Proceedings of the 2019 2nd International Conference on Computational Intelligence and Intelligent Systems*, CIIS 2019, pages 57–62. ACM.

O'Reilly, R. C., Munakata, Y., Frank, M. J., Hazy, T. E., and Contributors (2012). *Computational Cognitive Neuroscience*. PediaPress.

Sun, Y., Gong, H., Li, Y., and Zhang, D. (2019). Hyperparameter importance analysis based on N-RReliefF algorithm. *International Journal of Computers Communications & Control*, 14(4):557–573.

[1] https://cogsci.fmph.uniba.sk/sskv/