Towards Explaining HPO via Partial Dependence Plots

Julia Moosbauer*, Julia Herbinger*, Giuseppe Casalicchio, Marius Lindauer, Bernd Bischl

{julia.moosbauer,julia.herbinger}@stat.uni-muenchen.de

Introduction

AutoML systems mainly return well-performing configurations, but leave users without insights into the decisions of the optimization process. This lack of insights makes it difficult to trust and understand the automated process and the results.

Interpretable machine learning (IML) methods can be used to gain insights from experimental data obtained during HPO. Efficient optimizers like Bayesian Optimization tend to focus on promising regions with potential high-performance configurations and thus introduce a sampling bias. Therefore, IML techniques like *Partial Dependence Plots* (PDPs) carry the risk of generating biased interpretations.

Our Contributions:

- We study the problem of sampling bias in experimental data produced by HPO systems and assess its implications on PDPs.
- We derive an uncertainty measure for PDPs of probabilistic surrogate models.
- Based on this uncertainty measure, we propose to partition the hyperparameter space to obtain more confident and reliable PDPs in relevant sub-regions.

Background

Partial Dependence Plots:
Let $S \subset \{1, 2, ..., d\}$ denote an index set of hyperparameters, and let $C = \{1, 2, ..., d\} \setminus S$ be its complement. The PDP [1] of $\hat{c} : \Lambda \to \mathbb{R}$ for a sample $\left(\boldsymbol{\lambda}_C^{(i)}\right)_{i=1,...,n} \sim \mathbb{P}(\boldsymbol{\lambda}_C)$ and hyperparameter(s) S is defined as

$$\hat{c}_S : \Lambda_S \to \mathbb{R}, \qquad \hat{c}_S(\boldsymbol{\lambda}_S) = \frac{1}{n}\sum_{i=1}^{n} \hat{m}\left(\boldsymbol{\lambda}_S \boldsymbol{\lambda}_C^{(i)}\right), \tag{1}$$

with $\hat{m} : \Lambda \to \mathbb{R}$ denoting the posterior mean.

PDP as average over ICE curves:
For a fixed i, $\hat{m}\left(\boldsymbol{\lambda}_S, \boldsymbol{\lambda}_C^{(i)}\right) : \Lambda_S \to \mathbb{R}$ is called the i-th *Individual Conditional Expectation* (ICE) curve [2]. The PDP shows the average marginal contribution by averaging over all ICE curves.

Problem Statement

Sampling & Model Bias: Bayesian optimization tends to exploit promising regions of the hyperparameter space while other regions are less explored. Consequently, predictions of surrogate models (and thus also ICE curves) are usually more accurate with less uncertainty in well-explored regions and vice versa.

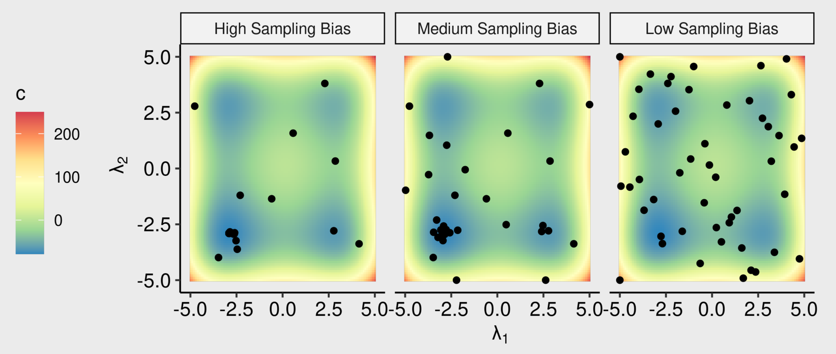

Figure 1: Illustration of the sampling bias when optimizing the 2D Styblinski Tang function with BO and the Lower Confidence Bound (LCB) acquisition function $a(\boldsymbol{\lambda}) = \hat{m}(\boldsymbol{\lambda}) + \tau \cdot \hat{s}(\boldsymbol{\lambda})$ for $\tau = 0.1$ (left) and $\tau = 2$ (middle) vs. data sampled uniformly at random (right).

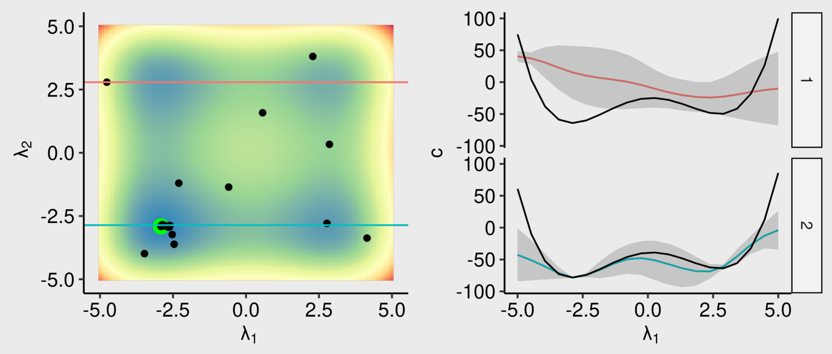

Figure 2: The two horizontal cuts (left) yield two ICE curves (right) showing the mean prediction and uncertainty band against λ_1 for $\hat{c}$ with $\tau = 0.1$ on the 2D Styblinski-Tang function. The upper ICE curve deviates more from the true effect (black) and shows a higher uncertainty.

Unreliable PD estimates: ICE curves may be biased and less confident if they are computed in poorly learned regions (upper curve) and may obfuscate well-learned effects of ICE curves belonging to other regions (lower curve) when they are aggregated to a PDP.

Quantifying Uncertainty in PDPs

Based on the posterior variance of the probabilistic surrogate model, we derived the following uncertainty estimate for the PDP estimate

$$\hat{s}_S^2(\boldsymbol{\lambda}_S) = \mathbb{V}_{\hat{c}}\left[\hat{c}_S(\boldsymbol{\lambda}_S)\right] = \mathbb{V}_{\hat{c}}\left[\frac{1}{n}\sum_{i=1}^{n} \hat{c}\left(\boldsymbol{\lambda}_S, \boldsymbol{\lambda}_C^{(i)}\right)\right] = \frac{1}{n^2}\mathbf{1}^\top \hat{\boldsymbol{K}}(\boldsymbol{\lambda}_S)\ \mathbf{1}, \tag{2}$$

with $\hat{\boldsymbol{K}}(\boldsymbol{\lambda}_S) := \left(\hat{k}\left(\left(\boldsymbol{\lambda}_S, \boldsymbol{\lambda}_C^{(i)}\right), \left(\boldsymbol{\lambda}_S, \boldsymbol{\lambda}_C^{(j)}\right)\right)\right)_{i,j=1,...,n}$ denoting the (posterior) covariance. This uncertainty estimate can be shown as confidence intervals around the PDP estimate.

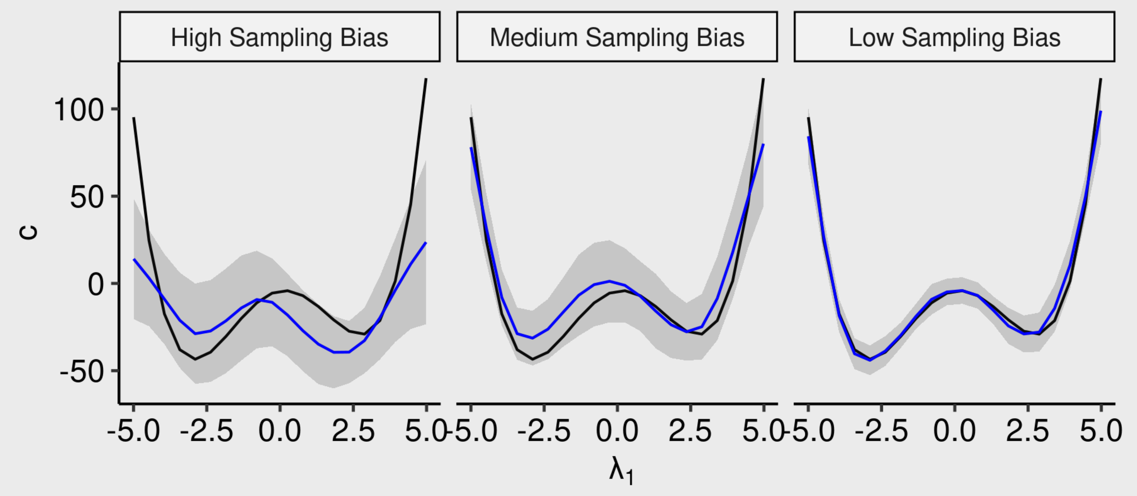

Figure 3: PDPs (blue) with confidence bands for surrogates trained on data created by BO and LCB with $\tau = 0.1$ (left), $\tau = 1$ (middle) and uniform i.i.d. dataset (right) vs. the true PD (black).

Partial Dependence Plots on Sub-regions

Aim: Find sub-regions of the hyperparameter space in which the PDP can be estimated with high confidence (2); separate those from regions where it cannot be estimated reliably.

Method: We propose a tree-based partitioning procedure that partitions the hyperparameter space Λ in disjoint and interpretable sub-regions. To receive sub-regions with confident PDP estimates, ICE curves are splitted according to the similarity of their uncertainty. We propose the splitting criterion $\mathcal{R}_{L2}(\mathcal{N}') = \sum_{g=1}^{G} L(\boldsymbol{\lambda}_S^{(g)}, \mathcal{N}')$, which is based on the loss

$$L(\boldsymbol{\lambda}_S, \mathcal{N}') = \sum_{i\in\mathcal{N}'}\left(\hat{s}^2\left(\boldsymbol{\lambda}_S, \boldsymbol{\lambda}_C^{(i)}\right) - \hat{s}^2_{S|\mathcal{N}'}(\boldsymbol{\lambda}_S)\right)^2 \tag{3}$$

with $\hat{s}^2_{S|\mathcal{N}'}(\boldsymbol{\lambda}_S) := \frac{1}{|\mathcal{N}'|}\sum_{i\in\mathcal{N}'} \hat{s}^2\left(\boldsymbol{\lambda}_S, \boldsymbol{\lambda}_C^{(i)}\right)$.

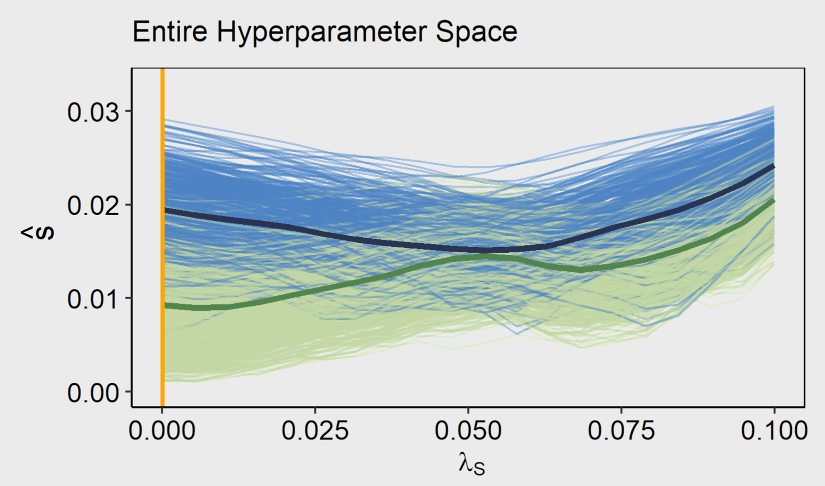

Figure 4: ICE curves of the uncertainty estimate of $\boldsymbol{\lambda}_S$ for the left (green) and right (blue) sub-region after the first split. The darker lines represent the respective PDPs. The orange vertical line marks the value λ_S of the optimal configuration.

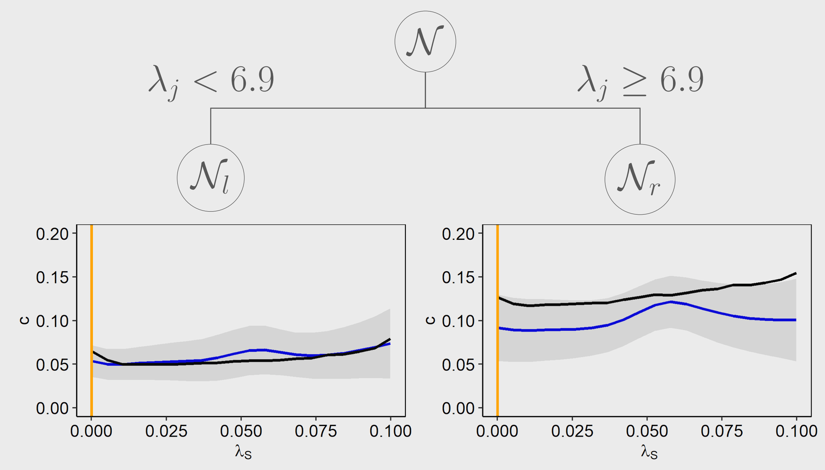

Figure 5: Example of two estimated PDPs (blue line) and 95% confidence bands after one partitioning step. The orange vertical line is the value of λ_S from the optimal configuration, the black curve is the true PD estimate.

Results

Benchmark setup: A surrogate benchmark based on 35 datasets of the LCBench [3], data was set up by training a random forest as empirical performance model based on the datasets. Model-based optimization with a GP surrogate model was run on each of the 35 tasks, and the final surrogate model was analyzed by our proposed methods.

Evaluation: We evaluate the performance of the method with regards to two main criteria

- **Reliability** of a PDP estimate measured by the Negative-log-likelihood (NLL) of PDP estimate compared to true PDP
- **Confidence**: Mean confidence (MC) over the entire range of $\boldsymbol{\lambda}_S$ and pointwise confidence at optimal configuration (OC)

We evaluate those measures in the sub-region containing the optimal configuration that we receive after 6 splits and compare it against the global estimates on the entire hyperparameter space.

Empirical Results:

- Confidence measures improve on average by at least 31 percent (on average higher improvement close to the optimal configuration)
- NLL improves on average by at least 12 percent

The analysis of individual examples showed that PDPs on the entire hyperparameter space can result in completely misleading interpretations while PDPs in confident sub-regions received by the splitting procedure reflect the true learned effect (see Figure 6).

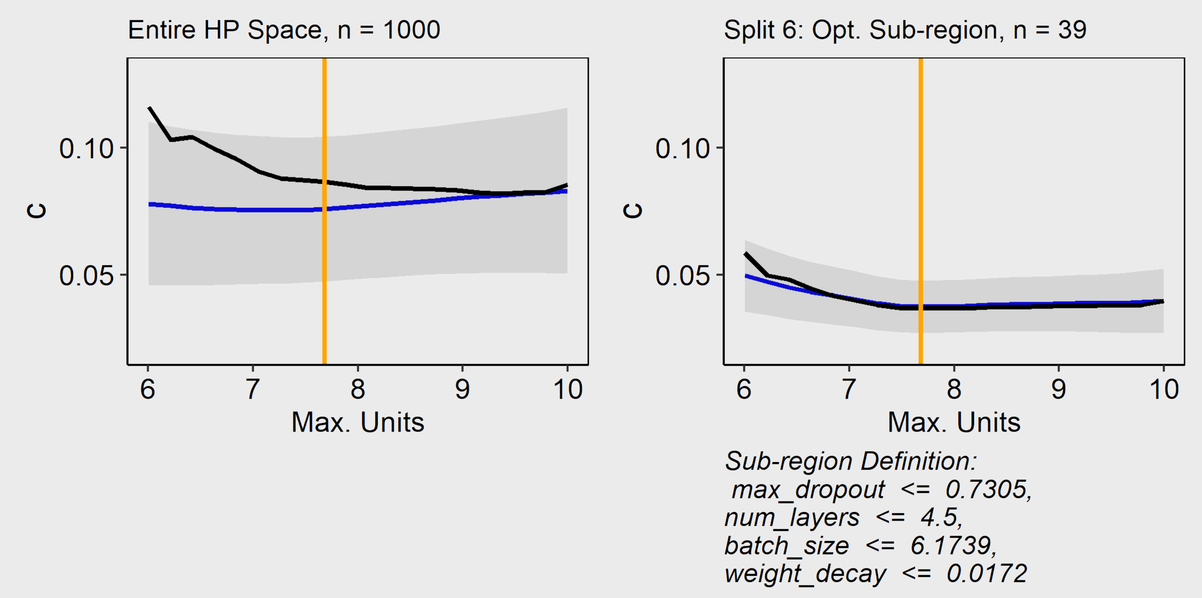

Figure 6: PDP (blue) and confidence band (grey) of the GP for hyperparameter *max. number of units*. The black line shows the PDP of the meta surrogate model representing the true PDP estimate. The orange vertical line marks the optimal configuration.

References

[1] Jerome H Friedman. Greedy function approximation: A gradient boosting machine. *Annals of Statistics*, pages 1189–1232, 2001.

[2] Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65, 2015.

[3] Lucas Zimmer, Marius Lindauer, and Frank Hutter. Auto-pytorch tabular: Multi-fidelity metalearning for efficient and robust autodl. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–12, 2021. To appear.