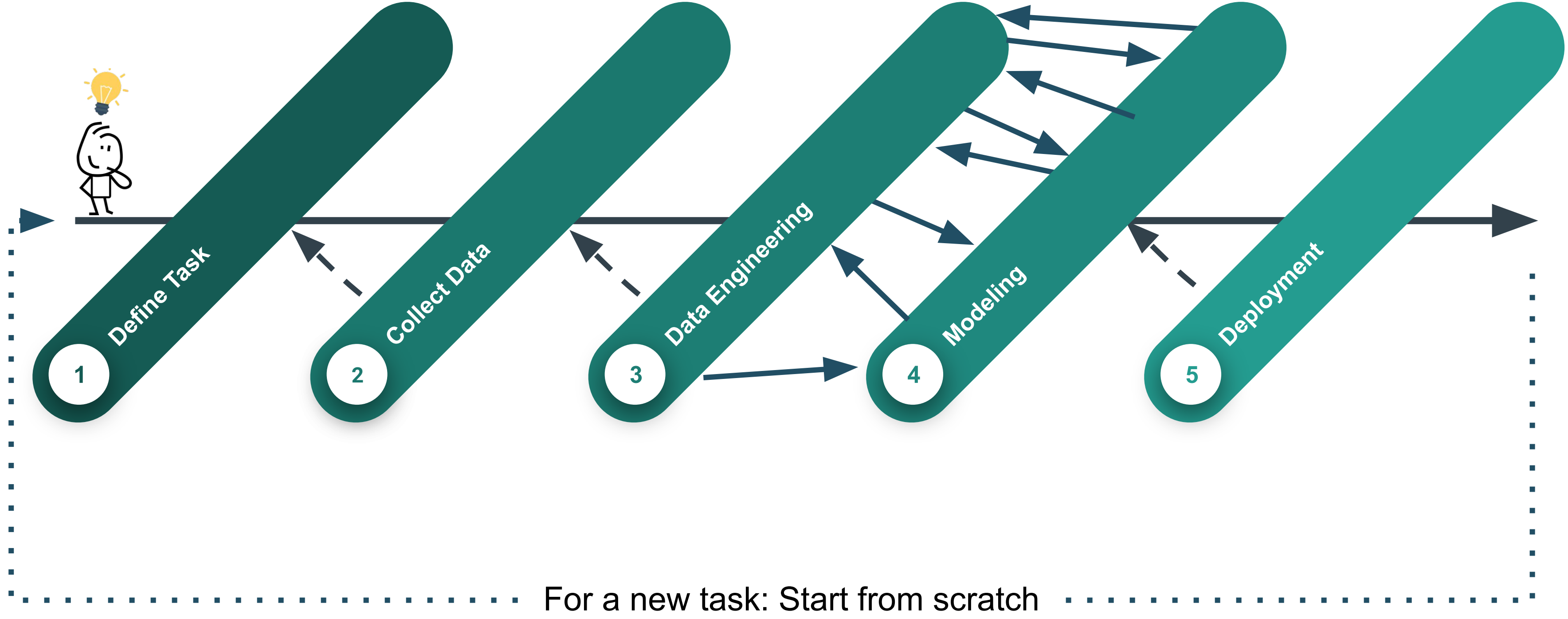
AutoML:
From Full Automation to A Human-Centric Approach

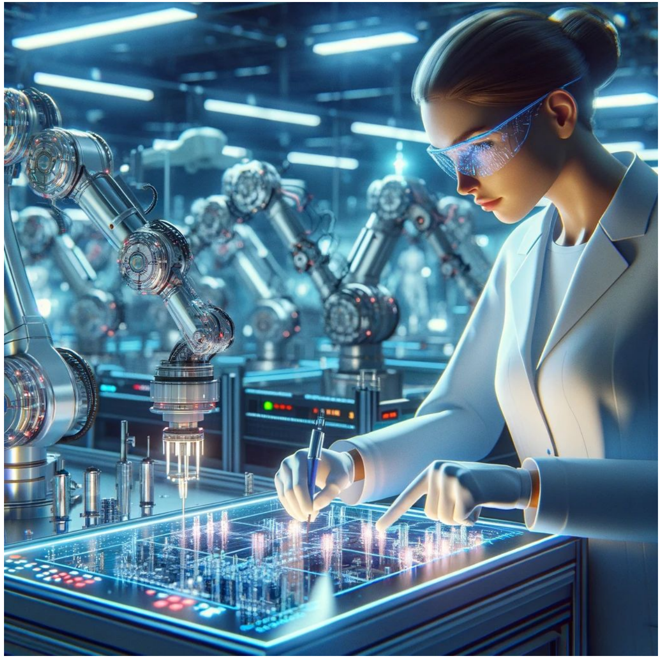
Prof. Marius Lindauer


Feb’2011
Ich
Frank
Hutter
Holger
Kevin
Leyton-Brown

Why does ML development take a lot of time?

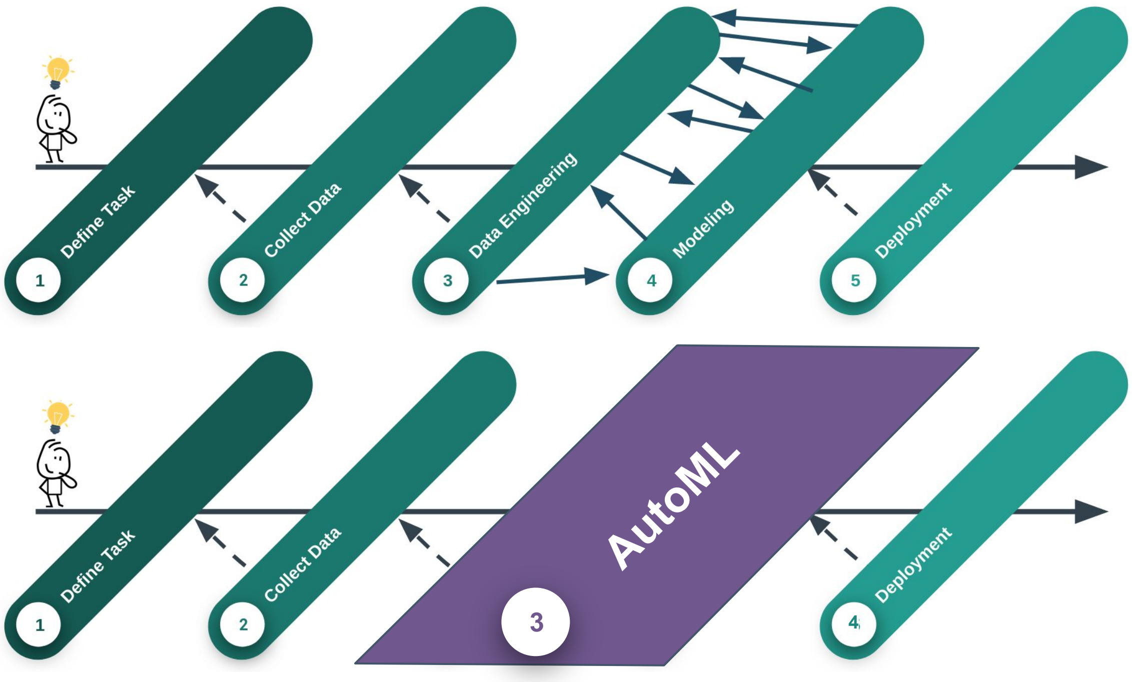


From ML Alchemy to Science

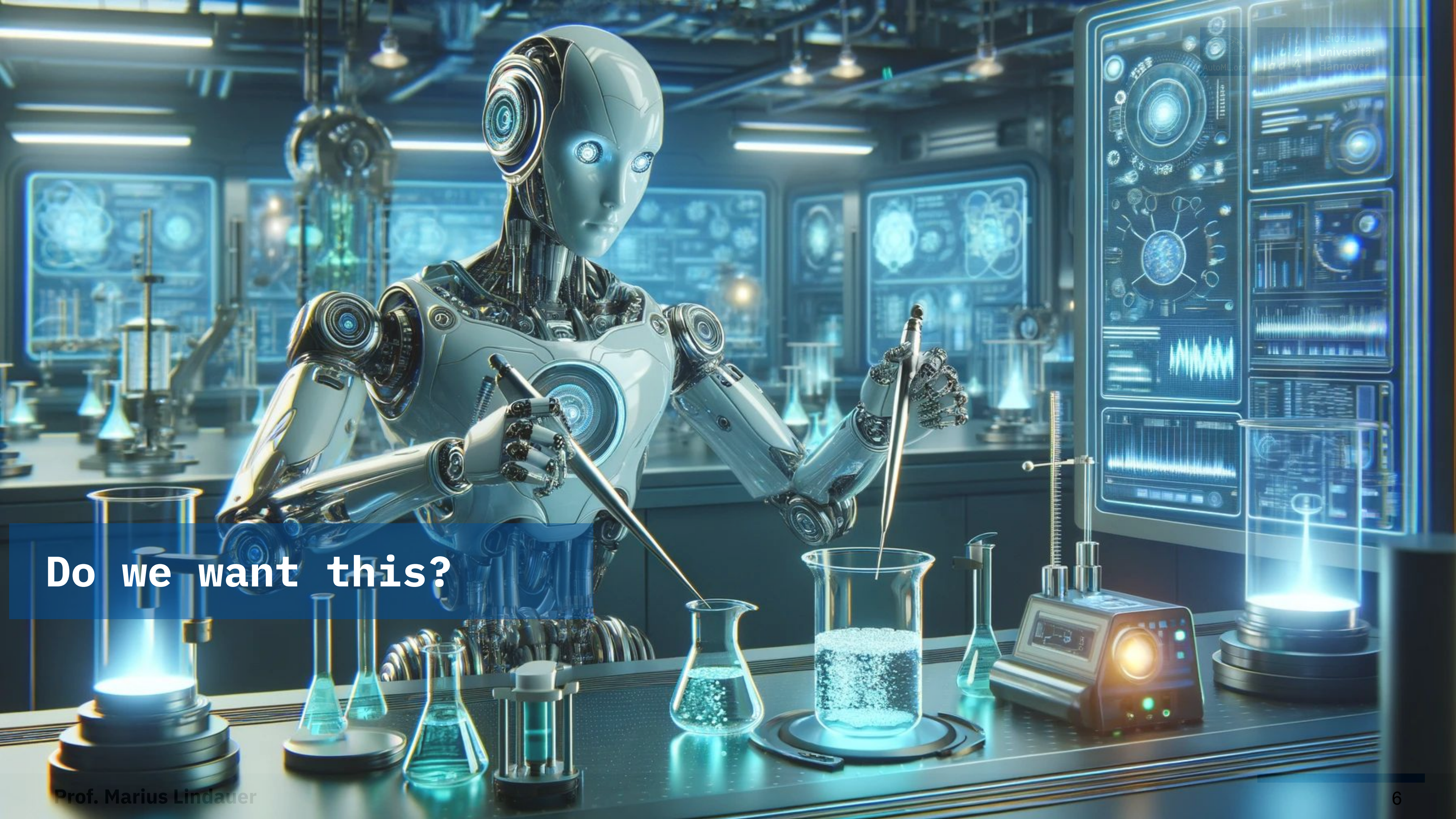


“You can teach an old dog new tricks” [Ruffinelli et al. 2020]
→ Hyperparameter optimization might not be the only required solution, but without it, it will also be hard.

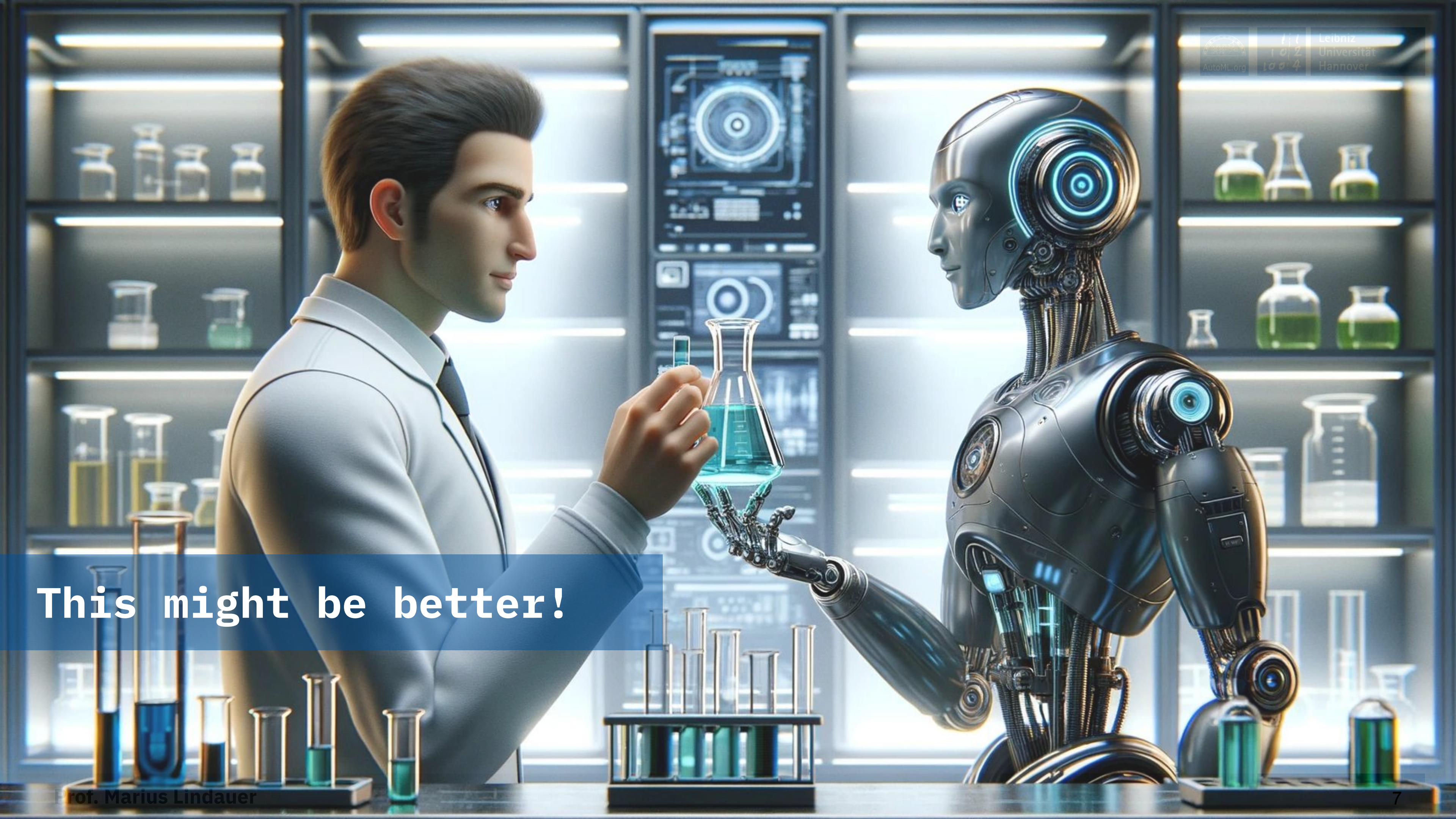
ML vs AutoML



Define Task
Collect Data
Data Engineering
Modeling
Deployment
1
2
3
4
5
Define Task
Collect Data
AutoML
Deployment
1
2
3
4



Do we want this?

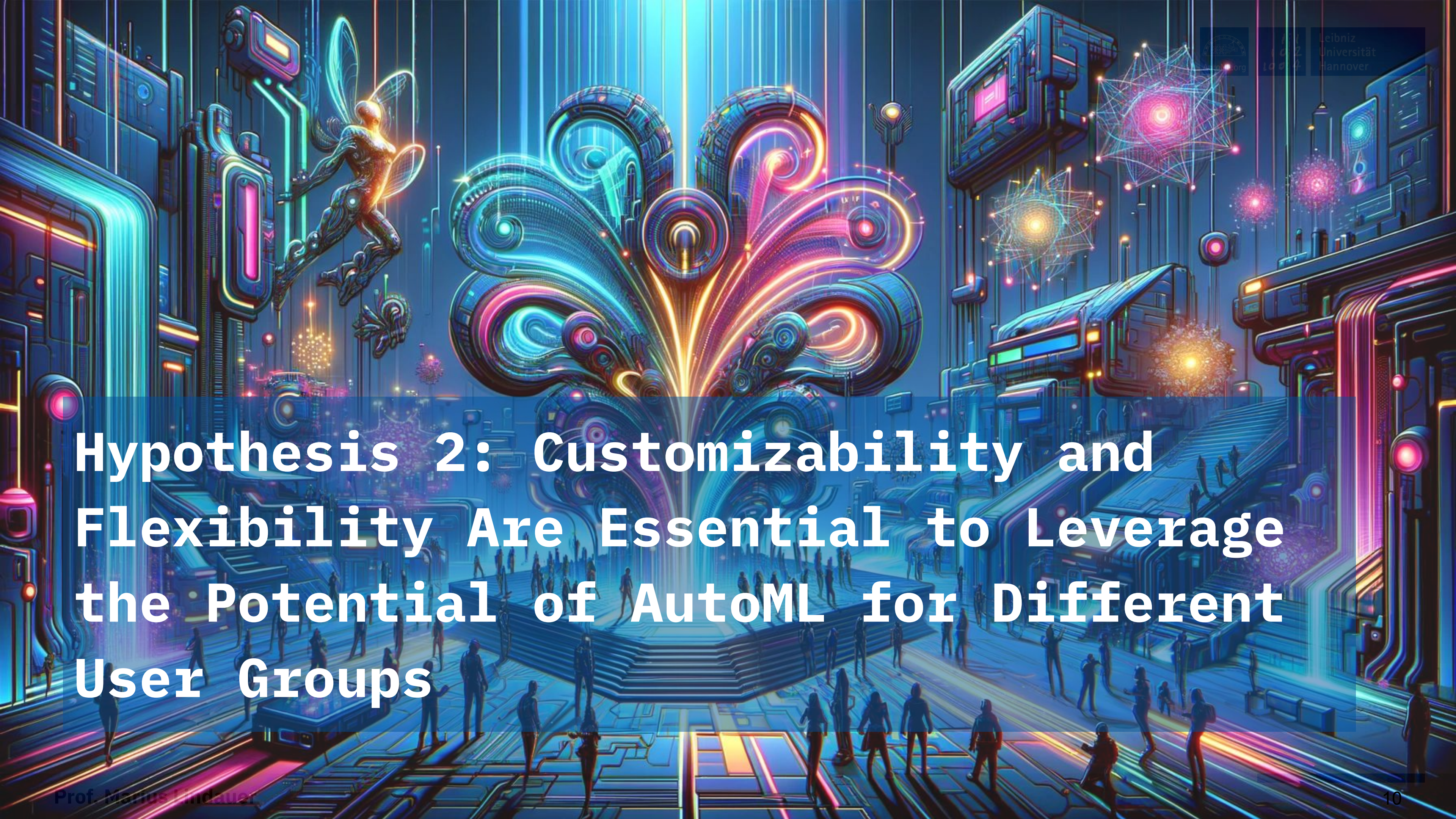


This might be better!

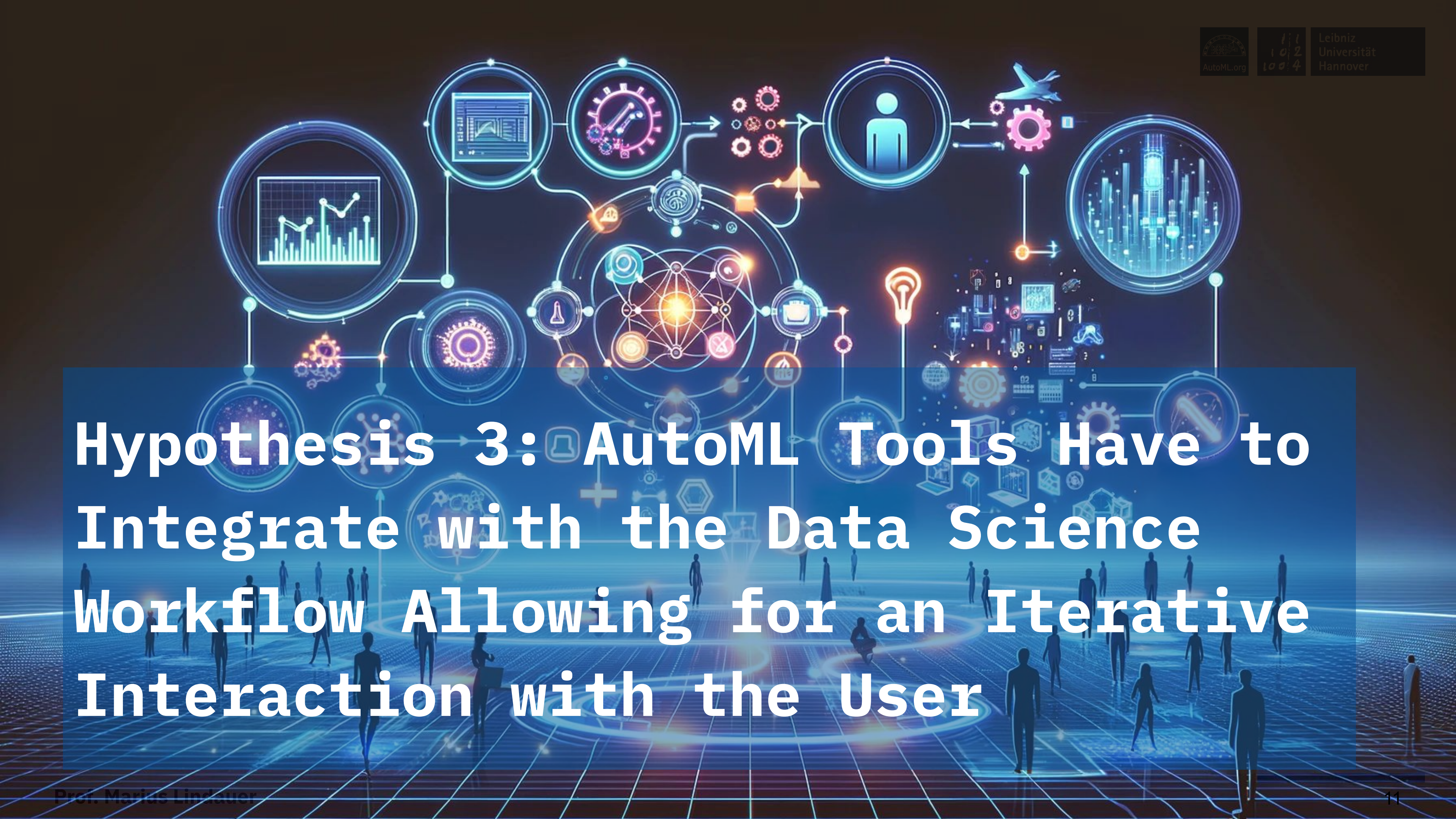
5 Hypotheses for Human-Centered AutoML

[Lindauer, Karl, Klier, Moosbauer, Tornede, Müller, Hutter, Feurer, Bischl. Submission to ICML’24]

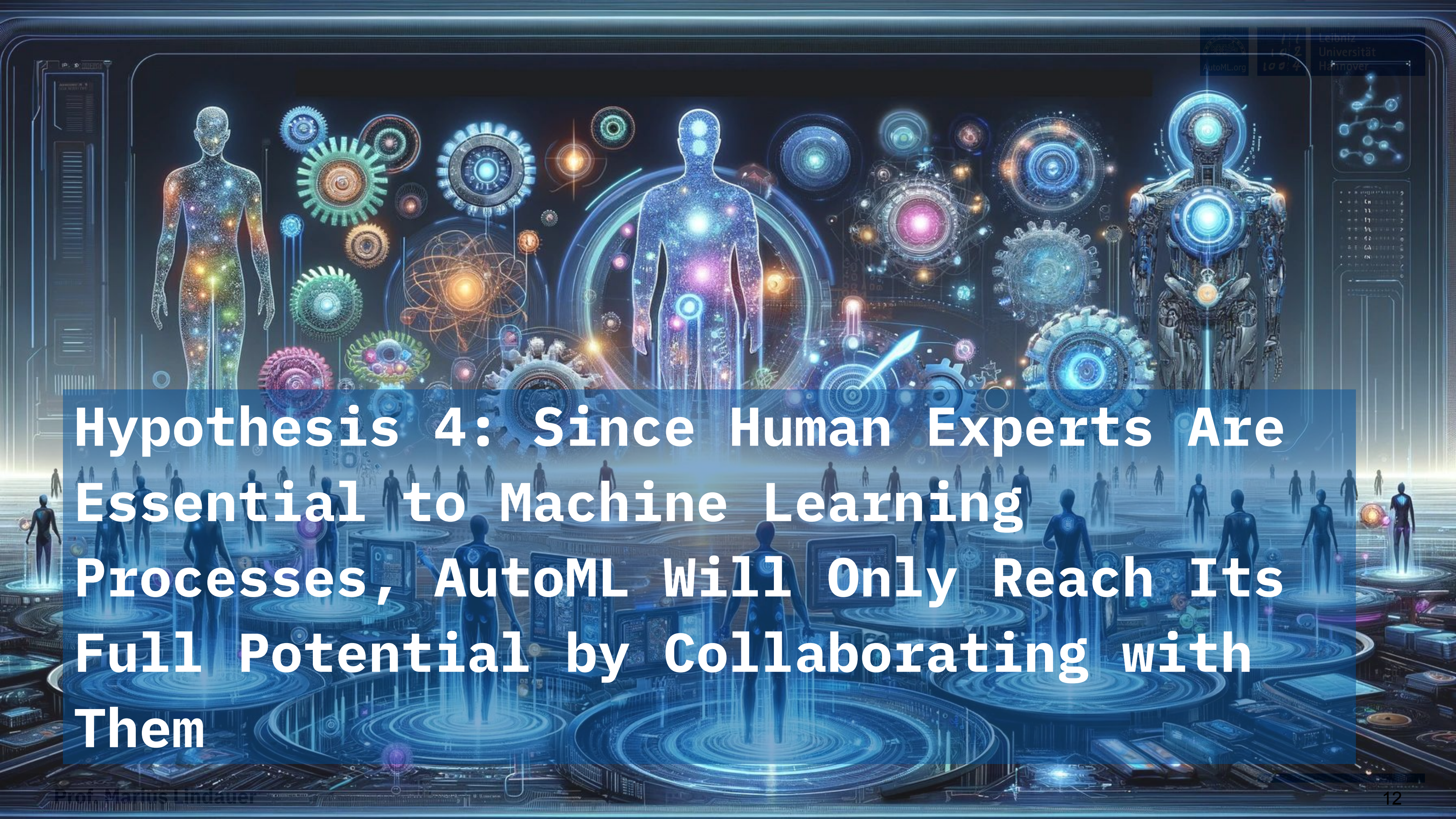
Hypothesis 1: Transparency and Interpretability are Key for ML and AutoML in Many Applications and on Many Levels



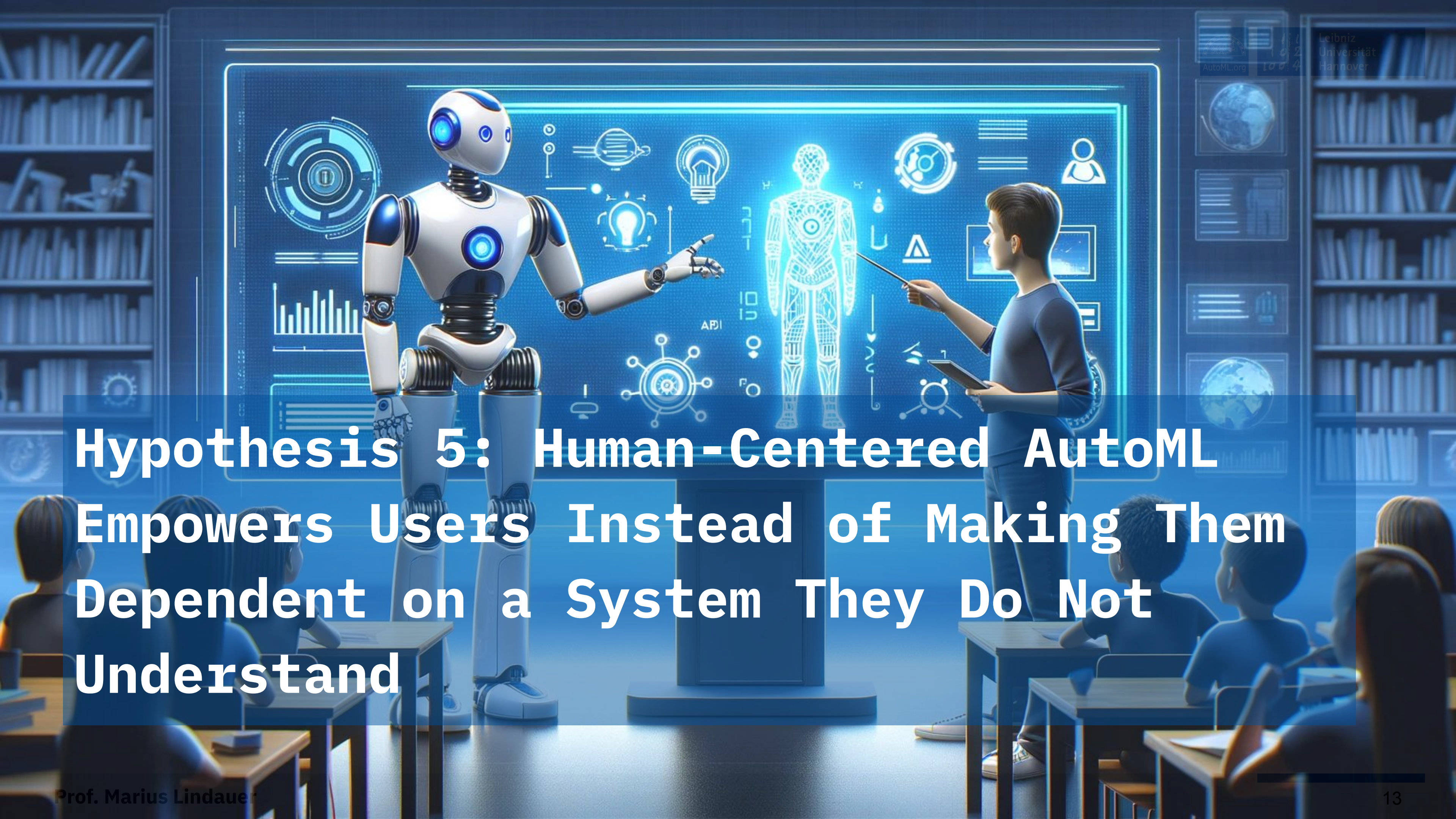
Hypothesis 2: Customizability and Flexibility Are Essential to Leverage the Potential of AutoML for Different User Groups



Hypothesis 3: AutoML Tools Have to Integrate with the Data Science Workflow Allowing for an Iterative Interaction with the User



Hypothesis 4: Since Human Experts Are Essential to Machine Learning Processes, AutoML Will Only Reach Its Full Potential by Collaborating with Them

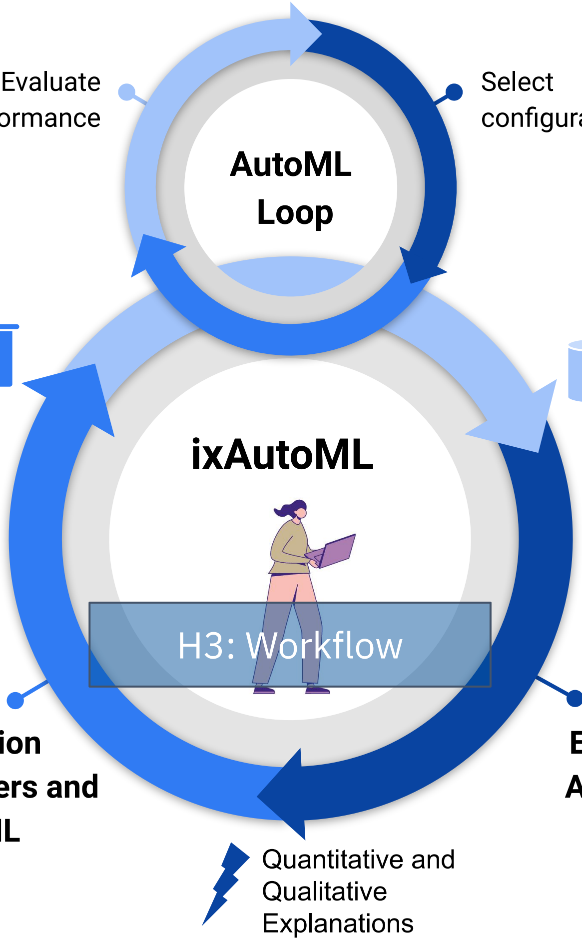


Hypothesis 5: Human-Centered AutoML Empowers Users Instead of Making Them Dependent on a System They Do Not Understand

ixAutoML: interactive and explainable AutoML (for Data Scientists and ML researchers)

H2: Customizability

H4: Collaboration



H1: Transparency

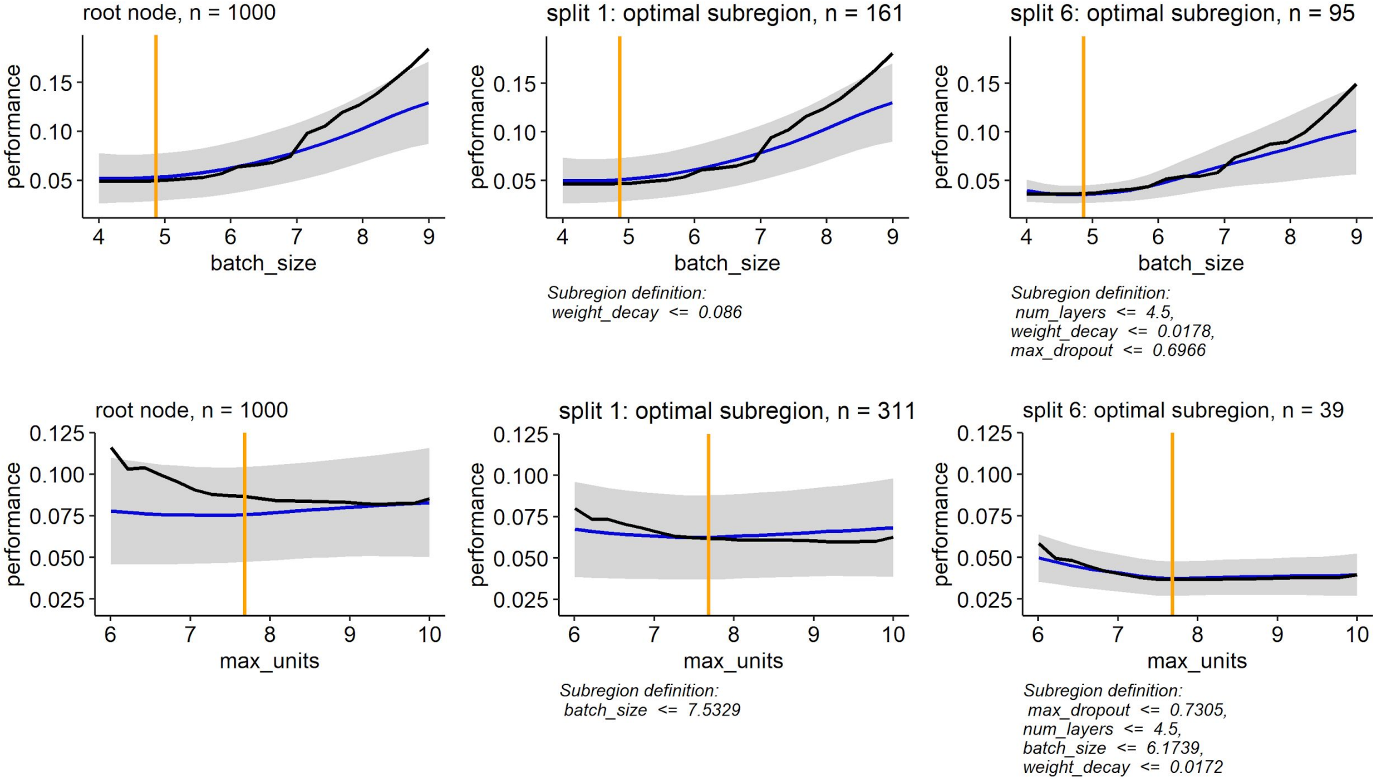
H5: Empowers Users

Explaining I: Partial Dependence Plots

Explaining Hyperparameter Effects via PDPs

[Moosbauer et al. NeurIPS'22]

Ground truth
PDP
incumbent



Partial Dependence Plots

[Moosbauer et al. NeurIPS'22]

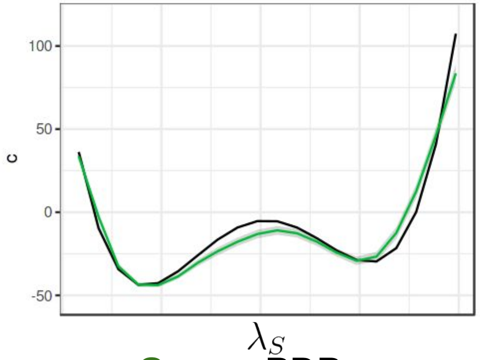
For, a subset S of the hyperparameters, the partial dependence function is:

$$c_S(\lambda_S) := \mathbb{E}_{\lambda_C}\left[c(\lambda)\right] = \int_{\Lambda_C} c(\lambda_S, \lambda_C) d\mathbb{P}(\lambda_C)$$

and can be approximated by Monte-Carlo integration on a surrogate model:

$$\hat{c}_S\left(\lambda_S\right) = \frac{1}{n}\sum_{i=1}^{n} \hat{m}\left(\lambda_S, \lambda_C^{(i)}\right)$$

where $\left(\lambda_C^{(i)}\right)_{i=1,\ldots,n} \sim \mathbb{P}(\lambda_C)$ and λ_S for a set of grid points.



Green: PDP
Black: Ground truth

$\rightarrow$ Average of ICE curves.

Partial Dependence Plots with Uncertainties

[Moosbauer et al. NeurIPS'22]

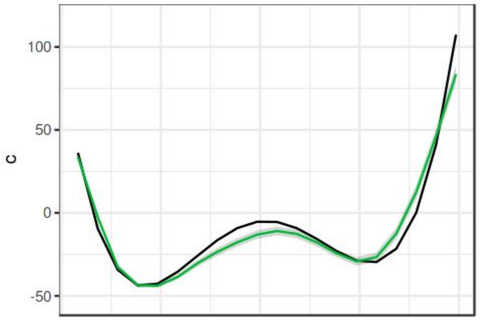
$$
\begin{aligned}
&\hat{s}^2_S(\lambda_S)\\
&= \mathbb{V}_{\hat{c}}\left[\hat{c}_S\left(\lambda_S\right)\right]\\
&= \mathbb{V}_{\hat{c}}\left[\frac{1}{n}\sum\nolimits_{i=1}^{n} \hat{c}\left(\lambda_S, \lambda_C^{(i)}\right)\right]\\
&= \frac{1}{n^2}1^\top \hat{K}\left(\lambda_S\right)\ 1.
\end{aligned}
$$

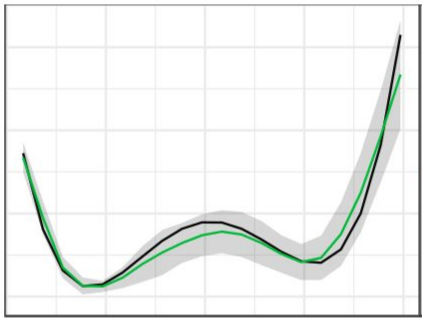
→ requires a kernel correctly specifying the covariance structure (e.g., GPs).

Approximation:

$$
\hat{s}^2_S\left(\lambda_S\right) \approx \frac{1}{n}\sum\nolimits_{i=1}^{n} \hat{K}\left(\lambda_S\right)_{i,i}
$$

→ Model-agnostic (local) approximation

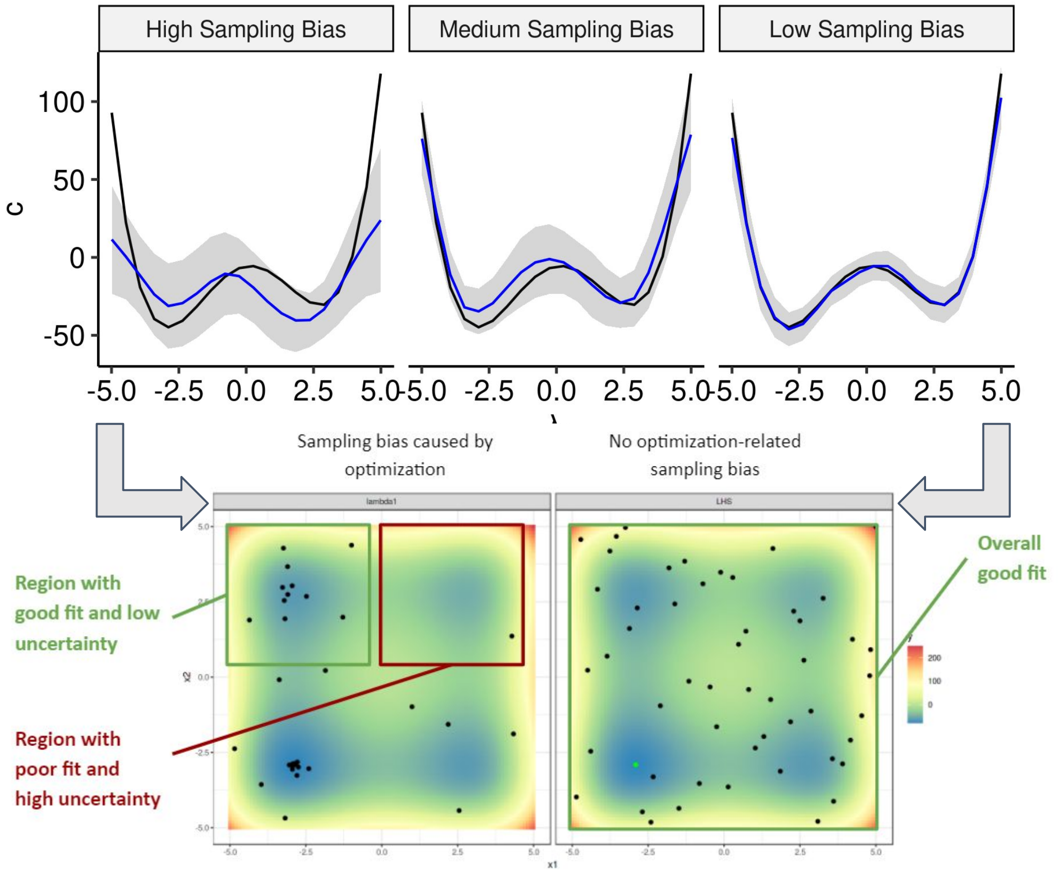




Ground truth
PDP
Uncertainty

Impact of Sampling Bias in Explaining AutoML

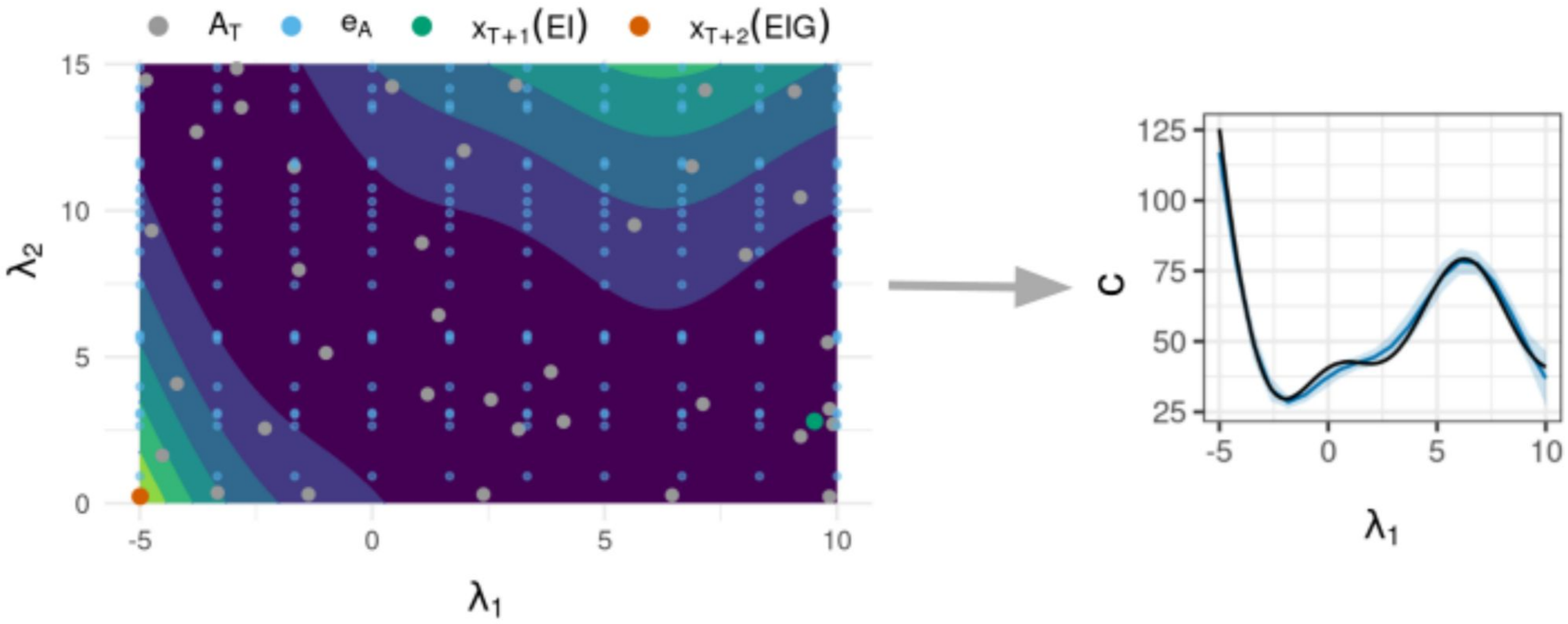
[Moosbauer et al. NeurIPS'22]



Exploration Strategy for Interpretability

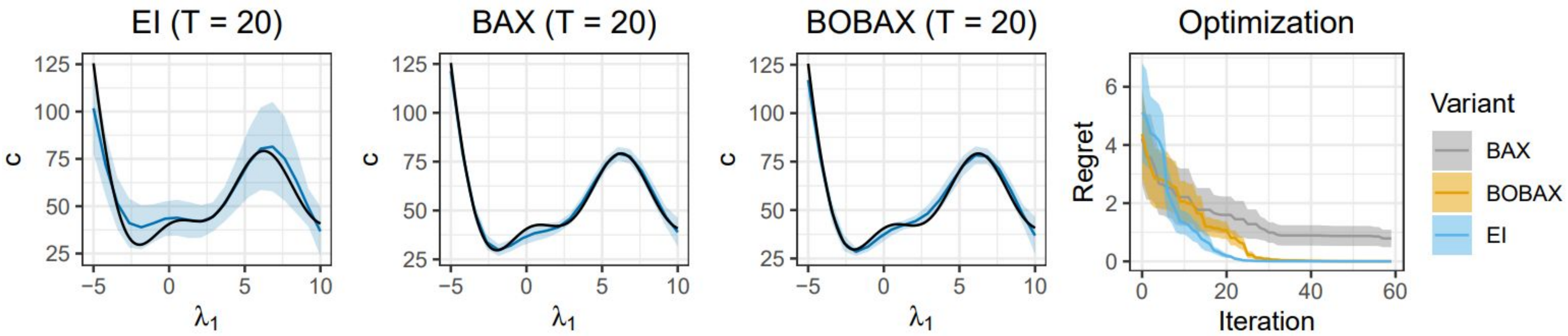
[Moosbauer et al. 2023]

- Main idea: Optimize for getting better interpretability of the HPO problem



Exploration Strategy for Interpretability

[Moosbauer et al. 2023]



1. Bayesian Optimization (BO) with EI leads to bad PDPs
2. Bayesian Algorithm Executation (BAX) leads to good PDPs, but poor optimization performance
3. Interleaving BO and BAX leads to good PDPs and strong optimization performance

Explaining II: Symbolic Regression

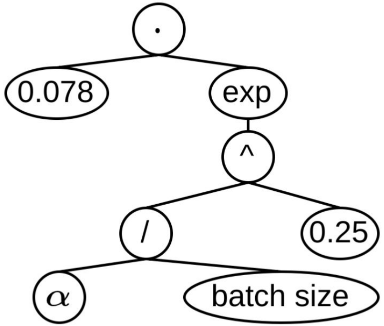
Symbolic Explanations for AutoML

[Segel et al. AutoML'23]

- Hyperparameter optimization (HPO) methods can find well-performing configurations efficiently
- Their **lack of transparency** can lead to missing trust of the users [Hasebrock et al. 2023]

Symbolic Explanations to the Rescue!

$$s(\alpha, \text{batch size}))$$
$$= 0.078 \cdot \exp\left((\alpha/\text{batch size})^{\frac{1}{4}}\right)$$

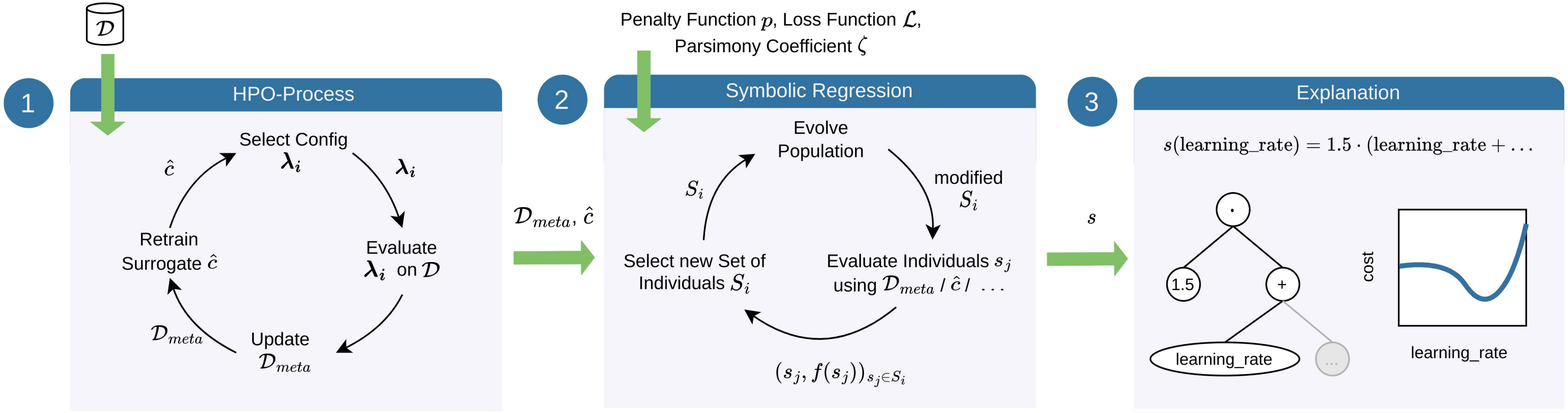


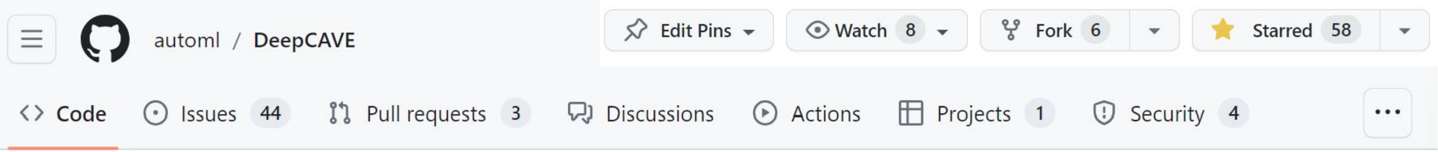
Symbolic Explanations for AutoML

[Segel et al. AutoML'23]

How to get more insights into hyperparameter effects?

- Employ **symbolic regression** to learn an **interpretable formula** that captures the relationship between hyperparameter configurations and model performance

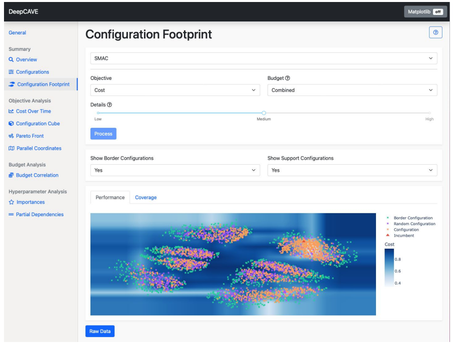


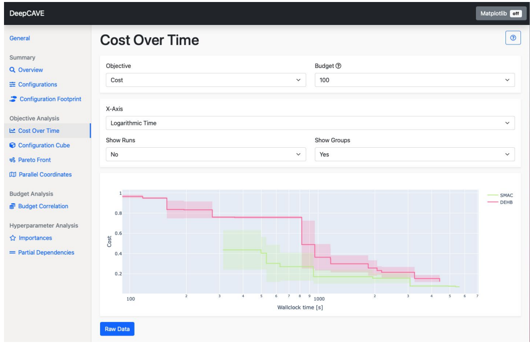


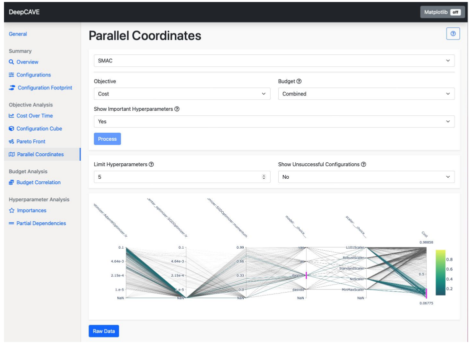
AutoML.org Leibniz Universität Hannover

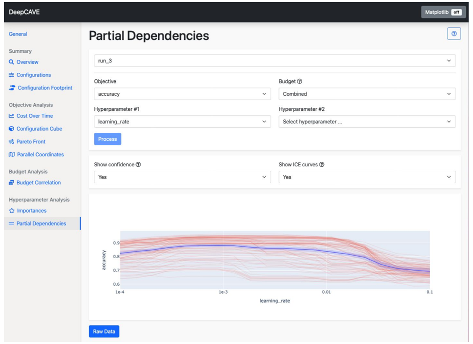
DeepCAVE [Sass et al. 2022]

- **Interactive Dashboard** to self-analyze optimization runs/processes.
- **Analyzing while optimizing**
- **Exploration of multiple areas** like performance, hyperparameter and budget analysis.
- **Modularized plugin** structure with access to selected runs/groups to provide maximal flexibility.
- **Asynchronous execution** of expensive plugins and caching of their results.
- **API mode** gives you full access to the code



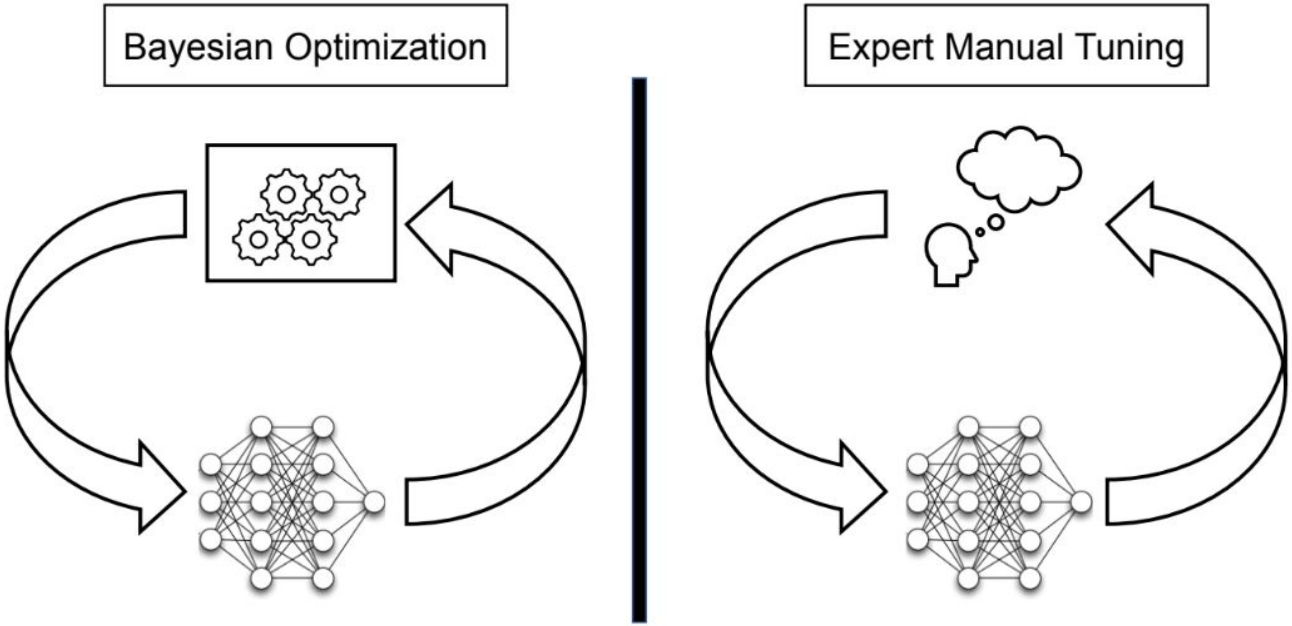




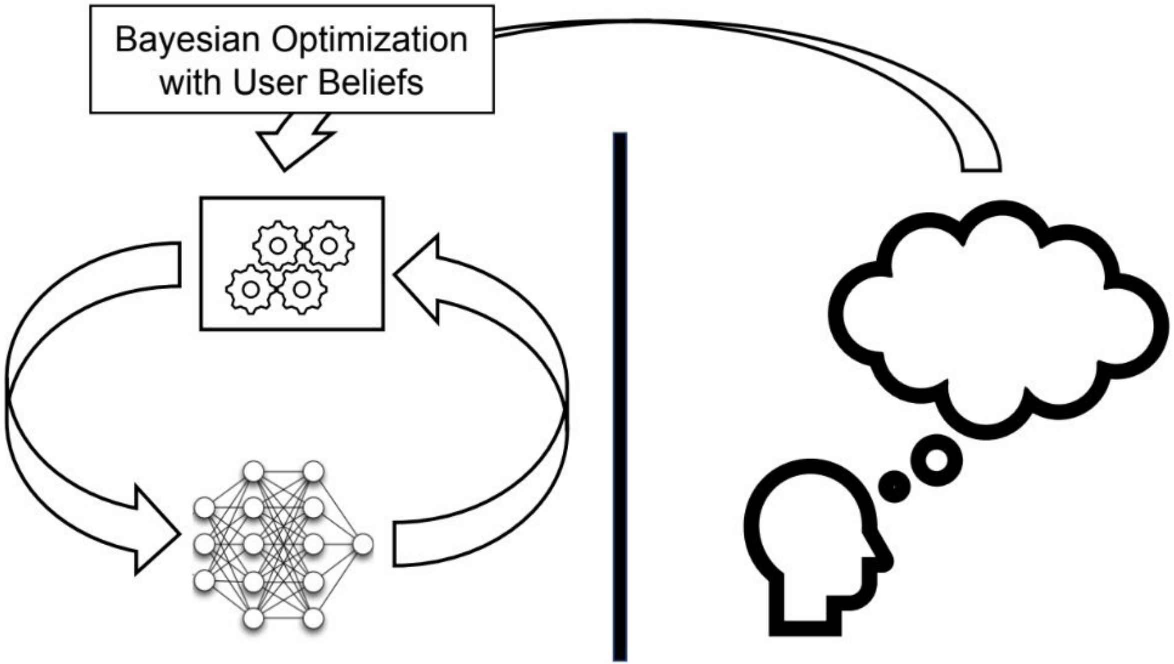


Interaction I: Expert-Priors

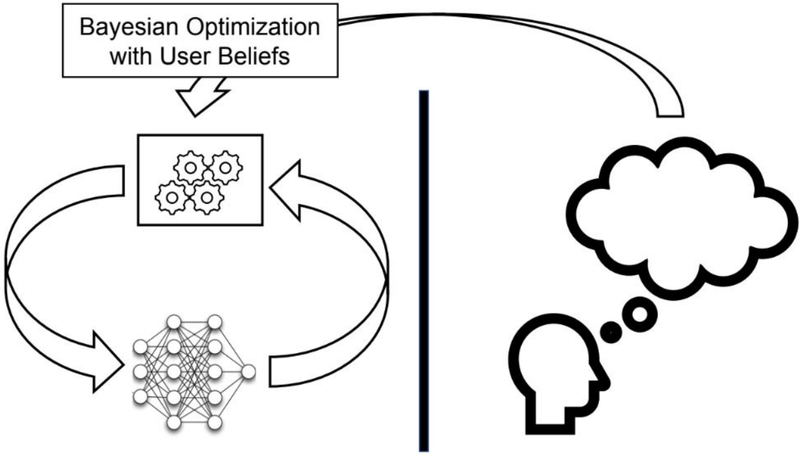
Bayesian Optimization vs Manual Tuning for HPO



Bayesian Optimization with Expert Knowledge



piBO [Hvarfner et al. ICLR'22]



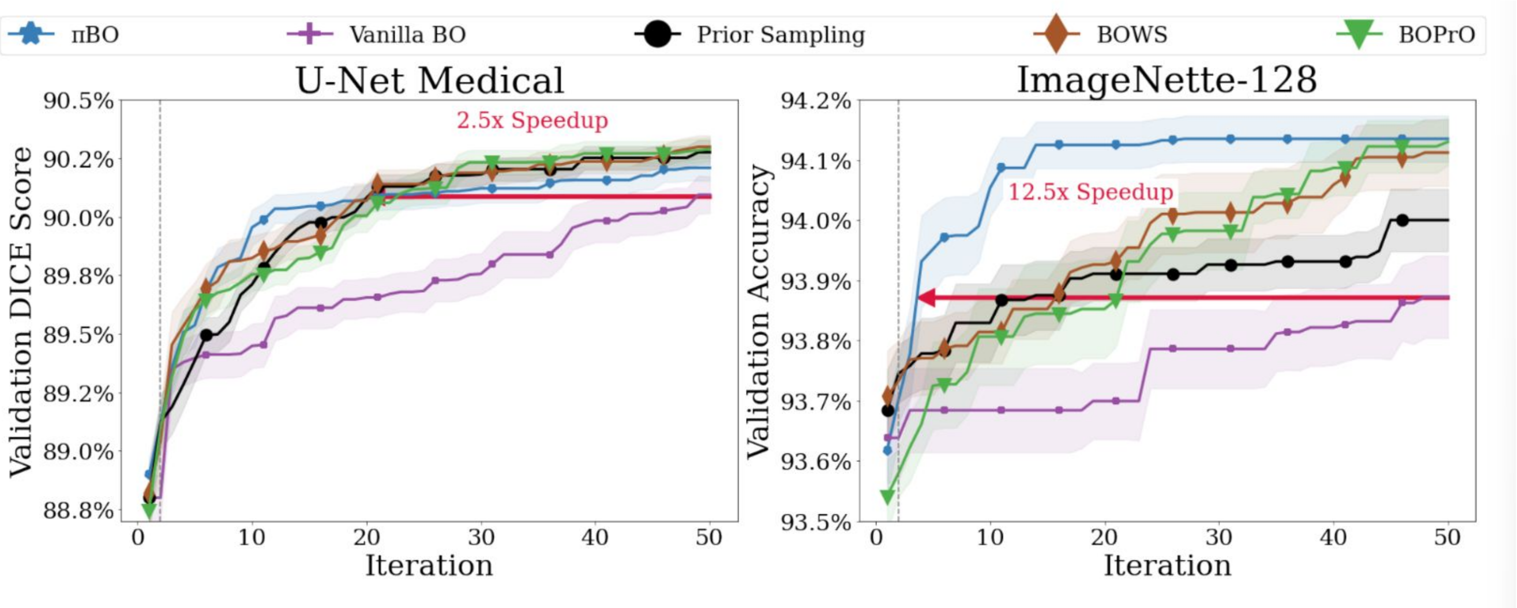
$$\boldsymbol{x}_n \in \underset{\boldsymbol{x}\in\mathcal{X}}{\arg\max}\, \alpha(\boldsymbol{x}, \mathcal{D}_n)\pi(\boldsymbol{x})^{\beta/n}$$

Acquisition Function

User Prior

Speed of forgetting user prior

piBO [Hvarfner et al. ICLR’22]



- ➔ Uses expert knowledge to speed up Bayesian Optimization
- ➔ Robust also against wrong believes
- ➔ Substantially speeds up AutoML
- ➔ Follow up with PriorBand [Mallik et al. NeurIPS’23]

Interaction II: Preferences for Multi-Objective AutoML

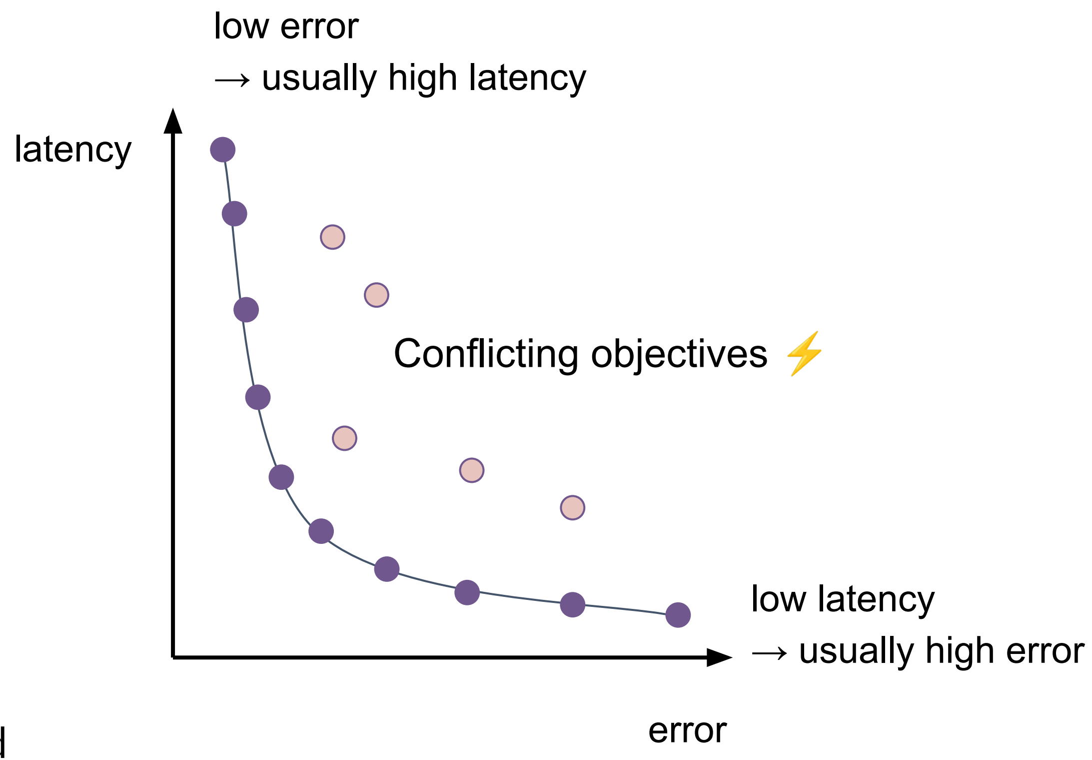
Multi-Objective AutoML

In practice, we often care about more than a single objective, e.g.

- error,
- inference time,
- unfairness,
- energy consumption,
- model complexity,
- and many more

~~Goal: Find a Neural Network with high accuracy and low latency~~

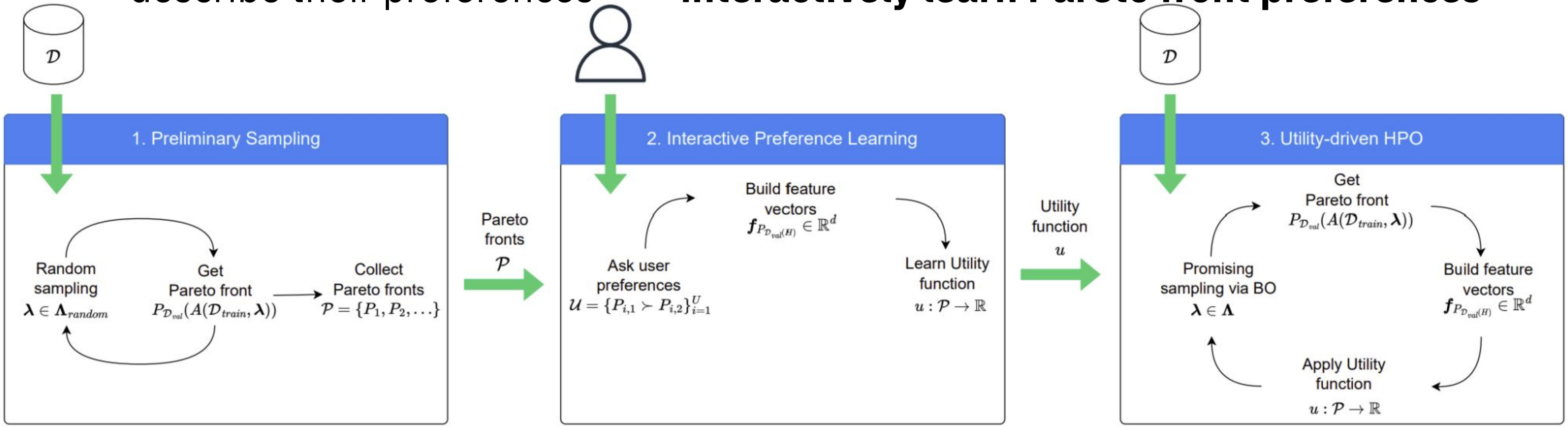
Goal: Find the Pareto Set of Neural Networks that balance accuracy and latency.



Interactive HPO in Multi-Objective Problems via Preference Learning [Giovanelli et al. AAAI'24]



- Multi-objective (Auto)ML gets more and more important
  - e.g., hardware-aware NAS, fairness-aware AutoML or energy-efficient AutoML
- **Practical challenge**: Different multi-objective indicators lead to different approximated Pareto fronts and users cannot always mathematically describe their preferences ⇒ **interactively learn Pareto front preferences**



Evaluation of Preference-Learned Indicators

[Giovanelli et al. AAAI'24]



- Benchmark:
  - LCBench
  - Accuracy vs. Energy-Consumption
- Let's assume : User randomly chose a multi-objective (MO) indicator, but was actually hoping for the behavior of another MO indicator
- ⇒ learned preferences are better than randomly choosing a MO indicator

PB\IB	$HV\uparrow$	$SP\downarrow$	$MS\uparrow$	$R2\downarrow$
$HV\uparrow$	0.76 (±0.17) \ **0.77 (±0.17)**	**0.76 (±0.17)** \ 0.52 (±0.24)	**0.76 (±0.17)** \ 0.52 (±0.21)	0.76 (±0.17) \ **0.77 (±0.16)**
$SP\downarrow$	**0.01 (±0.01)** \ 0.03 (±0.02)	**0.01 (±0.01)** \ **0.01 (±0.0)**	**0.01 (±0.01)** \ 0.04 (±0.03)	**0.01 (±0.01)** \ 0.04 (±0.02)
$MS\uparrow$	**0.61 (±0.09)** \ 0.19 (±0.08)	**0.61 (±0.09)** \ 0.19 (±0.12)	0.61 (±0.09) \ **0.65 (±0.06)**	**0.61 (±0.09)** \ 0.23 (±0.11)
$R2\downarrow$	0.23 (±0.16) \ **0.22 (±0.16)**	**0.23 (±0.16)** \ 0.47 (±0.23)	**0.23 (±0.16)** \ 0.45 (±0.21)	0.23 (±0.16) \ **0.21 (±0.16)**

AutoML in Constrained Applications

AutoML in Heavily Constrained Applications

[Neutatz et al. VLDBJ’23]


Default AutoML Configuration

Validation Strategy:	Holdout 66/33
Ensembling:	yes
Incremental Training:	yes
Validation split reshuffle:	no
ML Hyperparameter space:	
SVM:	Yes
SVM_tol:	Yes
SVM_C:	Yes
Extra Trees:	Yes
KNN:	Yes
Multilayer Perceptron:	Yes
Any Feature Preprocessor:	Yes
302 hyperparameters	Yes

Adapt AutoML parameters to ML task and deactivate undesired ML hyperparameters

Dynamic AutoML Configuration

Validation Strategy:	Holdout 46/54
Ensembling:	no
Incremental Training:	yes
Validation split reshuffle:	yes
ML Hyperparameter space:	
SVM:	Yes
SVM_tol:	Yes
SVM_C:	No
Extra Trees:	Yes
KNN:	No
Multilayer Perceptron:	No
Any Feature Preprocessor:	No
302 hyperparameters	Yes/No

Searches for the optimal ML pipeline in the defined search space. A pipeline is defined by the selected ML hyperparameters.

ML Pipeline

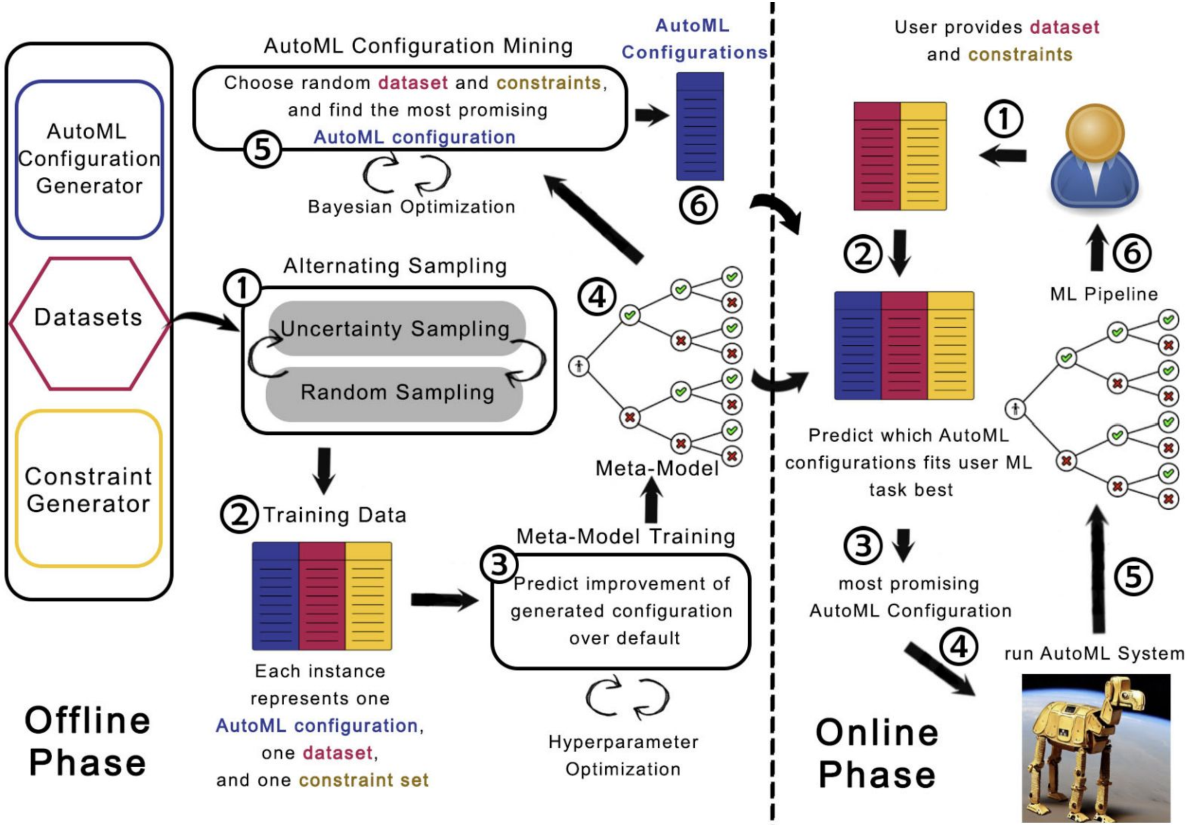
For SVM, the **model parameters** are the weights w:

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max\left(0, 1 - y_i(\mathbf{w}^{\mathsf{T}}\mathbf{x}_i - b)\right)\right] + \lambda\|\mathbf{w}\|^2.$$

ML Hyperparameters:	
SVM:	Yes
SVM_tol:	1e-5
SVM_C:	1.0 (default)
Extra Trees:	No
KNN:	No
Multilayer Perceptron:	No
Any Feature Preprocessor:	No
302 hyperparameters	...

AutoML in Heavily Constrained Applications

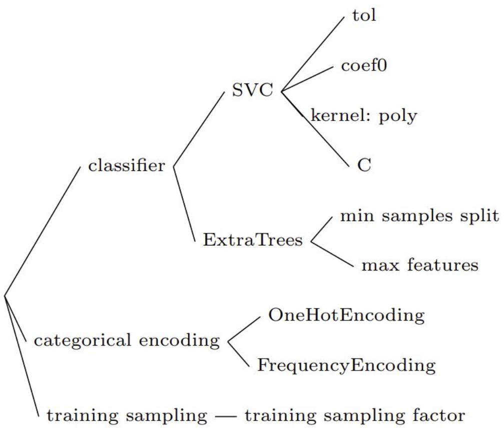
[Neutatz et al. VLDBJ'23]



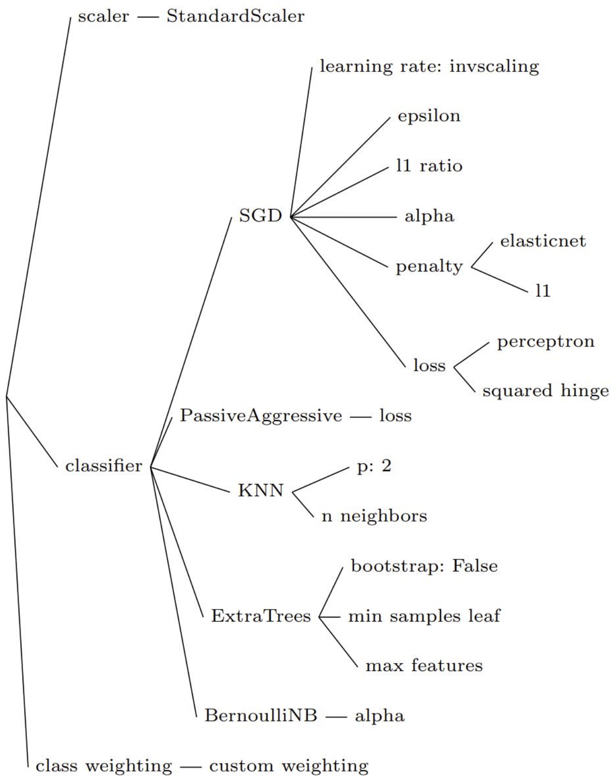
Possible application constraints:

- AutoML budget
- Inference time
- Memory consumption
- Energy consumption
- Fairness thresholds
- …

Can it learn to select different configuration spaces? [Neutatz et al. 2023]



(a) Christine (1min search time)



(b) Robert (5min search time)

Take-Aways for Meta-Learning AutoML Conf.

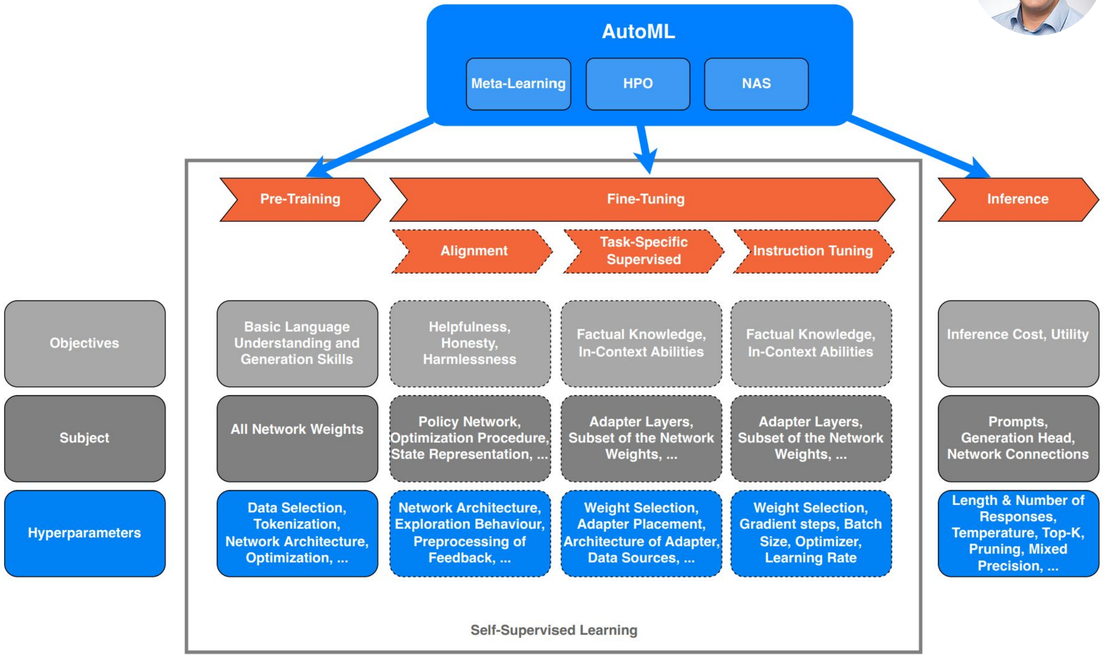

- **Assumption**: If we invest more time into the development of AutoML packages (incl. meta-learning), we save a lot of compute resources for using it
- **Positive** take-away:
  Yes, we can meta-learn how to configure AutoML systems and achieve new state-of-the-art performance
- **Negative** take-away:
  We cannot easily do it for large AutoML budgets (beyond 10min) without enormous compute resources
- **Future challenge**: How to configure AutoML on expensive tasks;
  "Expensive" can mean:
  - very expensive ML models (e.g., LLMs)
  - very complex configuration spaces with thousands of ML trainings

AutoML ↔ LLMs

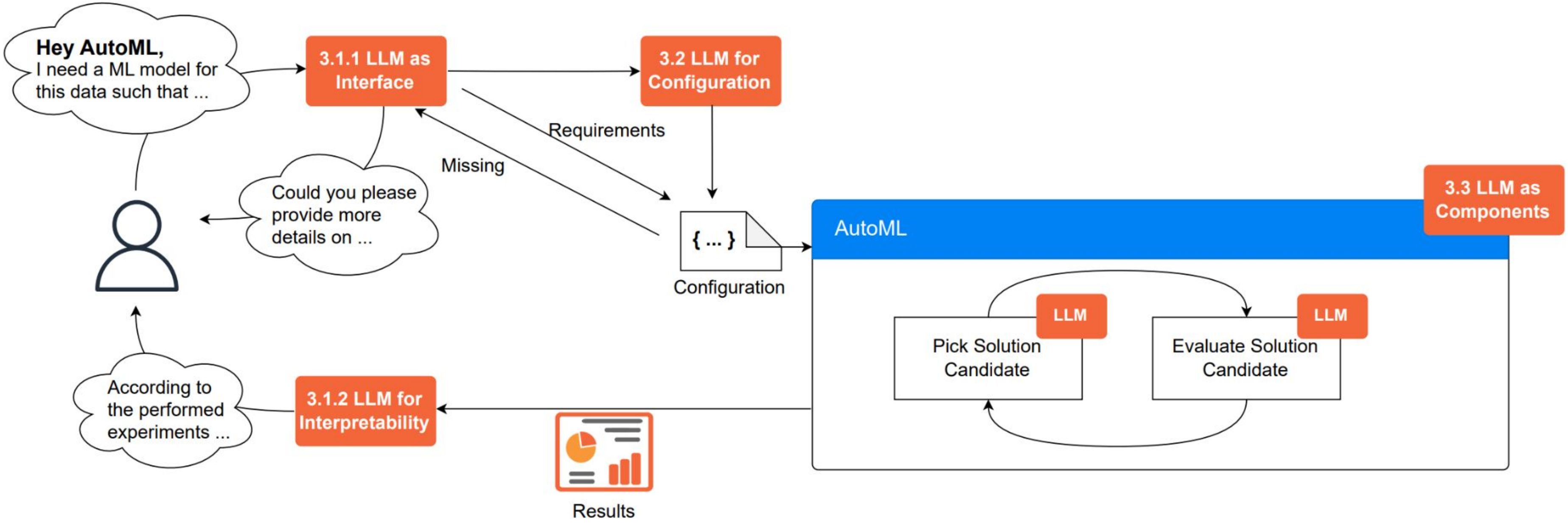
AutoML ➡ LLMs [Tornede et al. 2023]

Challenges

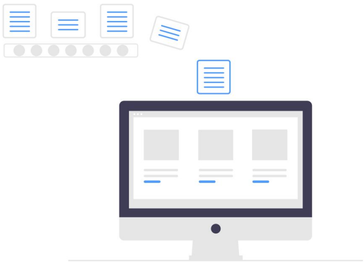
1. Cost of Pre-Training Base Models
2. Multitude of Different Stages
3. Multitude of Performance Indicators
4. Combination of Different Learning Paradigms
5. Determining Neural Architectures for LLMs



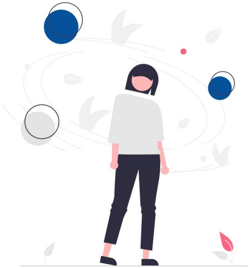
AutoML ⬅ LLMs [Tornede et al. 2023]



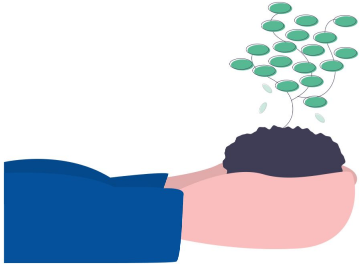
Our Research Foci



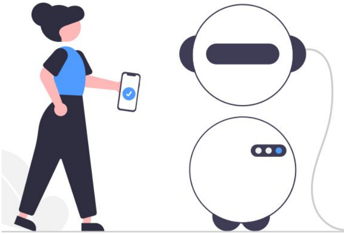
Core AutoML



Human-centered AutoML

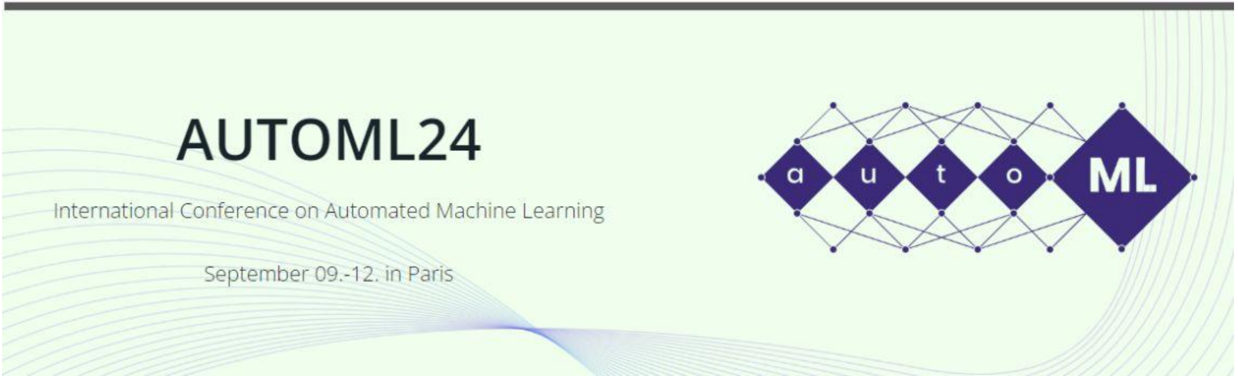


Green AutoML



AutoRL

AutoML Weeks 2024



Find Us

AutoML.org

automl-org

automl

@AutoML_org

LUH|AI

luh-ai

LUH-AI

@luh-ai

Funded by:

European Research Council
Established by the European Commission

DFG
Deutsche Forschungsgemeinschaft

L3S

DAAD
Deutscher Akademischer Austauschdienst
German Academic Exchange Service

Federal Ministry of Education and Research

Federal Ministry for Economic Affairs and Energy

Federal Ministry for the Environment, Nature Conservation, Nuclear Safety and Consumer Protection

Niedersachsen