LUH|AI

Leibniz
Universität
Hannover

AutoML: Replacing Data Scientists?

Marius Lindauer

Introducing myself

> 2005	2005-2014	2014-2019	Seit 2019
Berlin	**Potsdam**	**Freiburg**	**Hannover**
School	B.Sc. + M.Sc. + Ph.D in Computer Science	PostDoc in Frank Hutter's group	Professor of Machine Learning & AutoML

Photo by Stefan Widua on Unsplash

Photo by Chris Reyem on Unsplash

Photo by Kanan Khasmammadov on Unsplash

Photo by op23 on Unsplash

The need of AutoML!?

Rise of Literacy

- Only priests were able to read and write
- People believed that they don't need to read and write
- They went to the holy buildings

Photo by Anna Hunko on Unsplash

- Today, everyone can read and write
- No one doubts the benefits of it
- **⇒ Democratization of literacy**

Inspired by Andrew Ng

AI Literacy?

Photo by Max Duzij on Unsplash

- Only highly educated people can program new AI applications
- Power only with the large IT companies

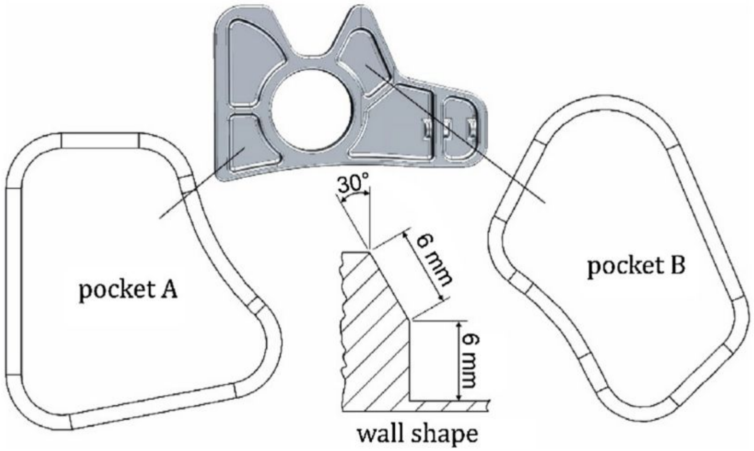
- In an age of limited resources, the need for efficient use gets more important

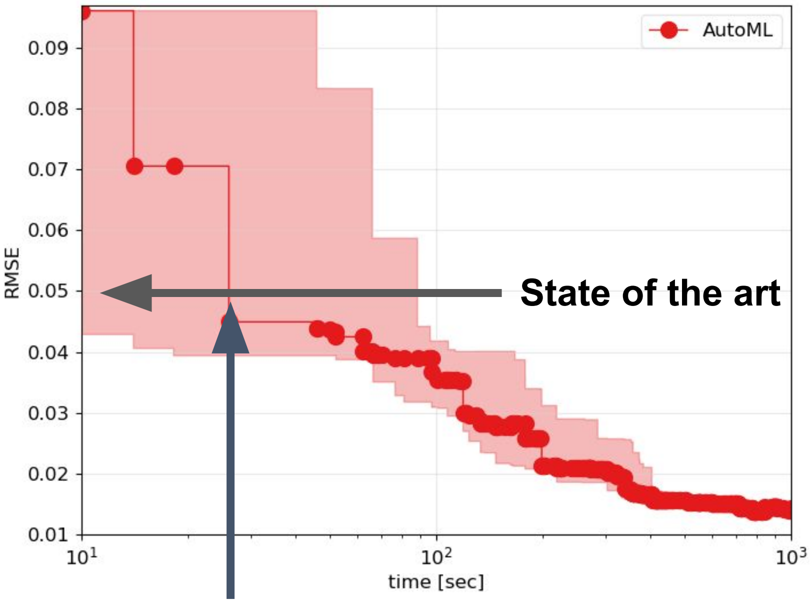
- **AutoML contributes to AI literacy!**

[See also my TEDx Talk]

A case study with engineers [Denkena et al. 2020]

Shape Error Prediction in Milling Processes

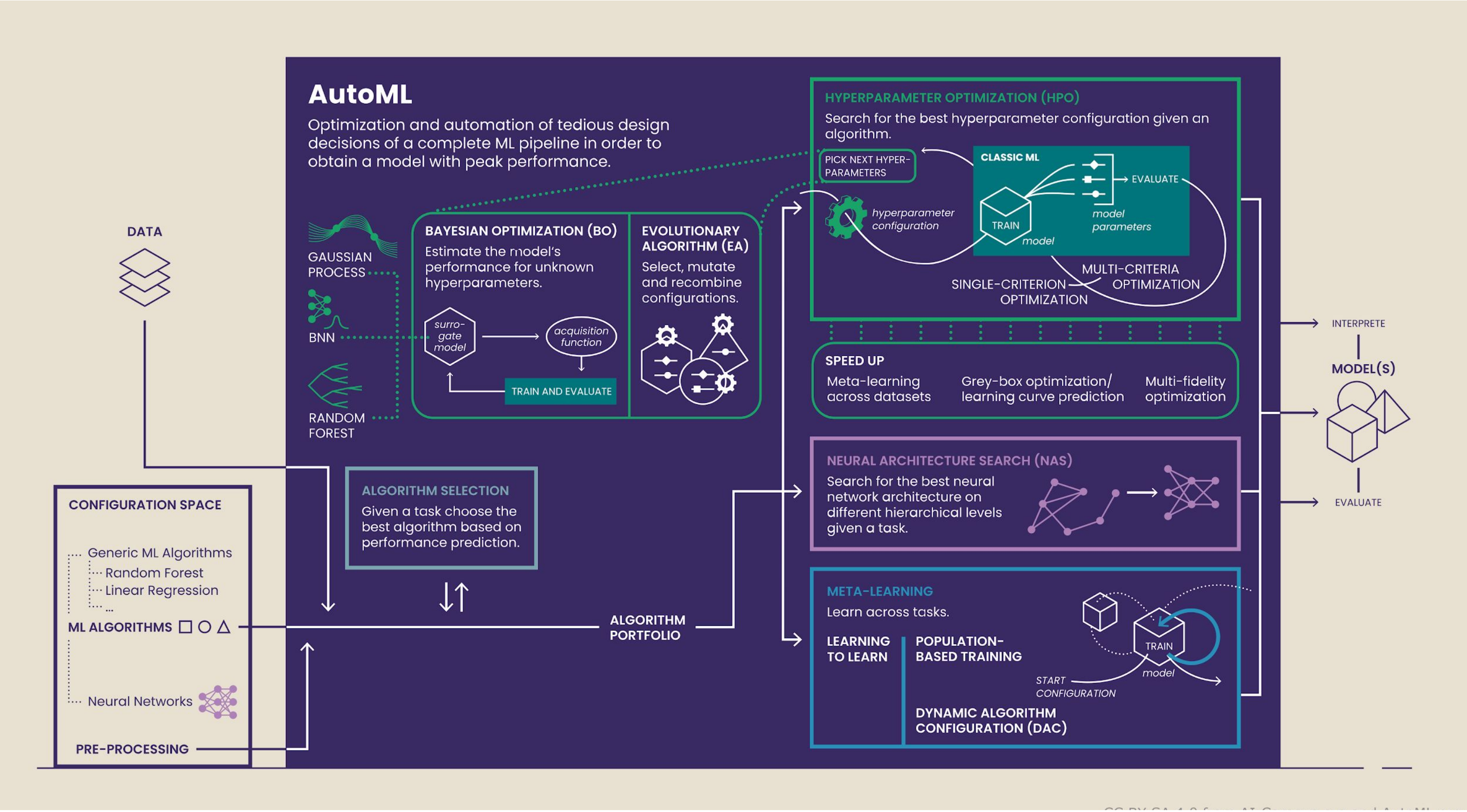




Better than state of the art after 27 sec!

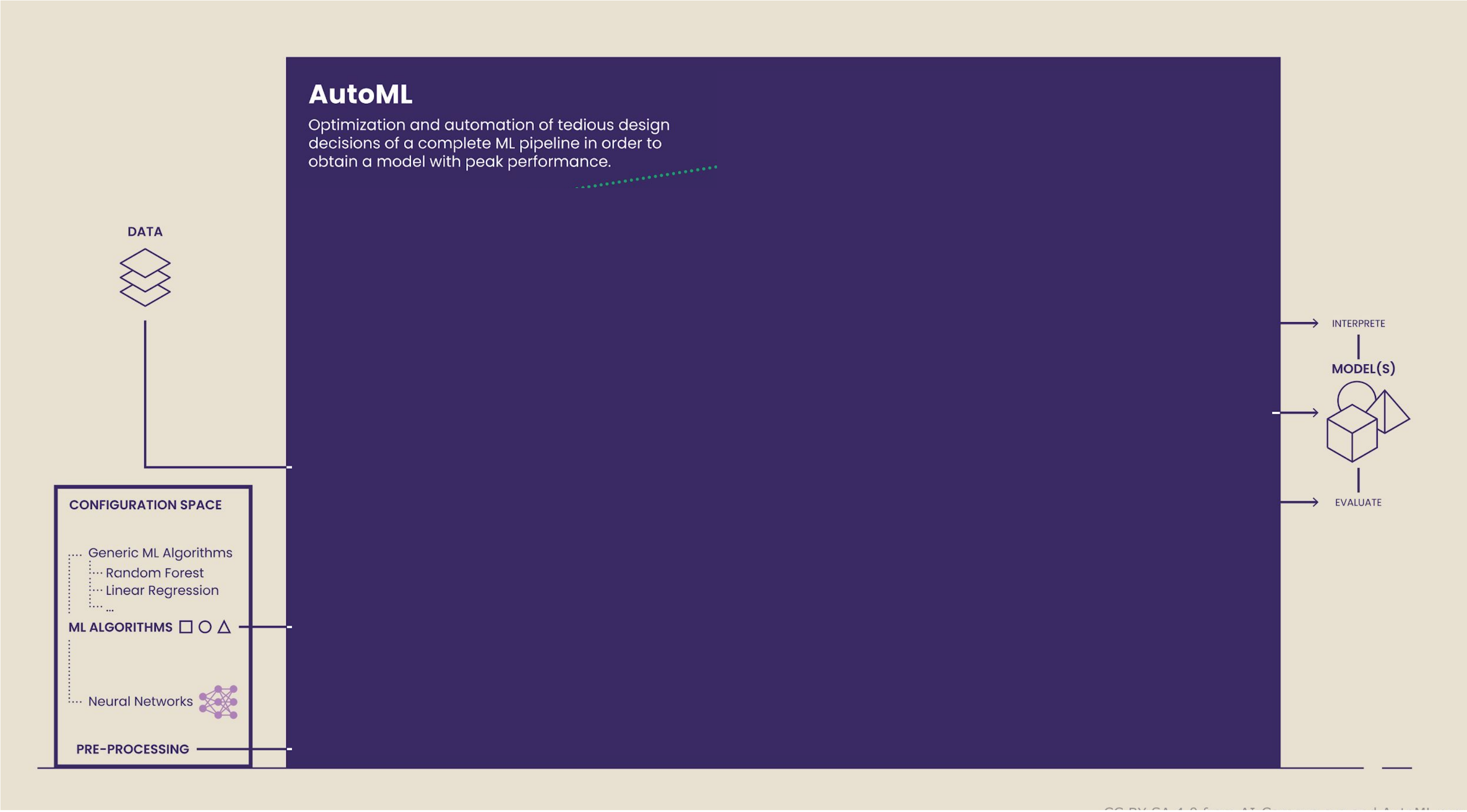
AutoML Landscape

AutoML A-Z

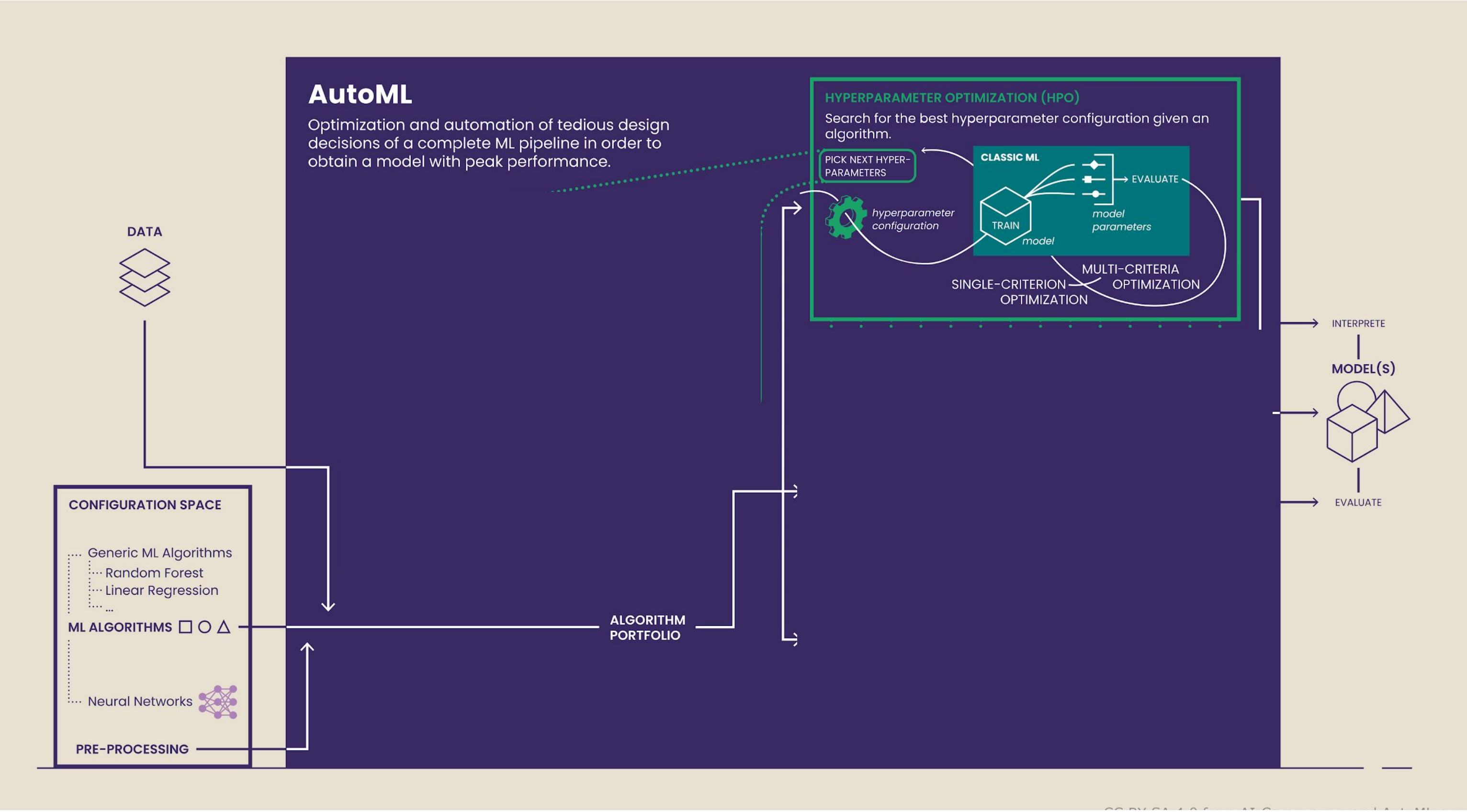


AutoML
Optimization and automation of tedious design decisions of a complete ML pipeline in order to obtain a model with peak performance.
DATA
GAUSSIAN PROCESS
BNN
RANDOM FOREST
BAYESIAN OPTIMIZATION (BO)
Estimate the model's performance for unknown hyperparameters.
surro-gate model
acquisition function
TRAIN AND EVALUATE
EVOLUTIONARY ALGORITHM (EA)
Select, mutate and recombine configurations.
HYPERPARAMETER OPTIMIZATION (HPO)
Search for the best hyperparameter configuration given an algorithm.
PICK NEXT HYPER-PARAMETERS
CLASSIC ML
EVALUATE
hyperparameter configuration
TRAIN
model
model parameters
MULTI-CRITERIA OPTIMIZATION
SINGLE-CRITERION OPTIMIZATION
SPEED UP
Meta-learning across datasets
Grey-box optimization/ learning curve prediction
Multi-fidelity optimization
NEURAL ARCHITECTURE SEARCH (NAS)
Search for the best neural network architecture on different hierarchical levels given a task.
META-LEARNING
Learn across tasks.
LEARNING TO LEARN
POPULATION-BASED TRAINING
TRAIN
model
START CONFIGURATION
DYNAMIC ALGORITHM CONFIGURATION (DAC)
ALGORITHM SELECTION
Given a task choose the best algorithm based on performance prediction.
ALGORITHM PORTFOLIO
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS □ ○ △
Neural Networks
PRE-PROCESSING
INTERPRETE
MODEL(S)
EVALUATE

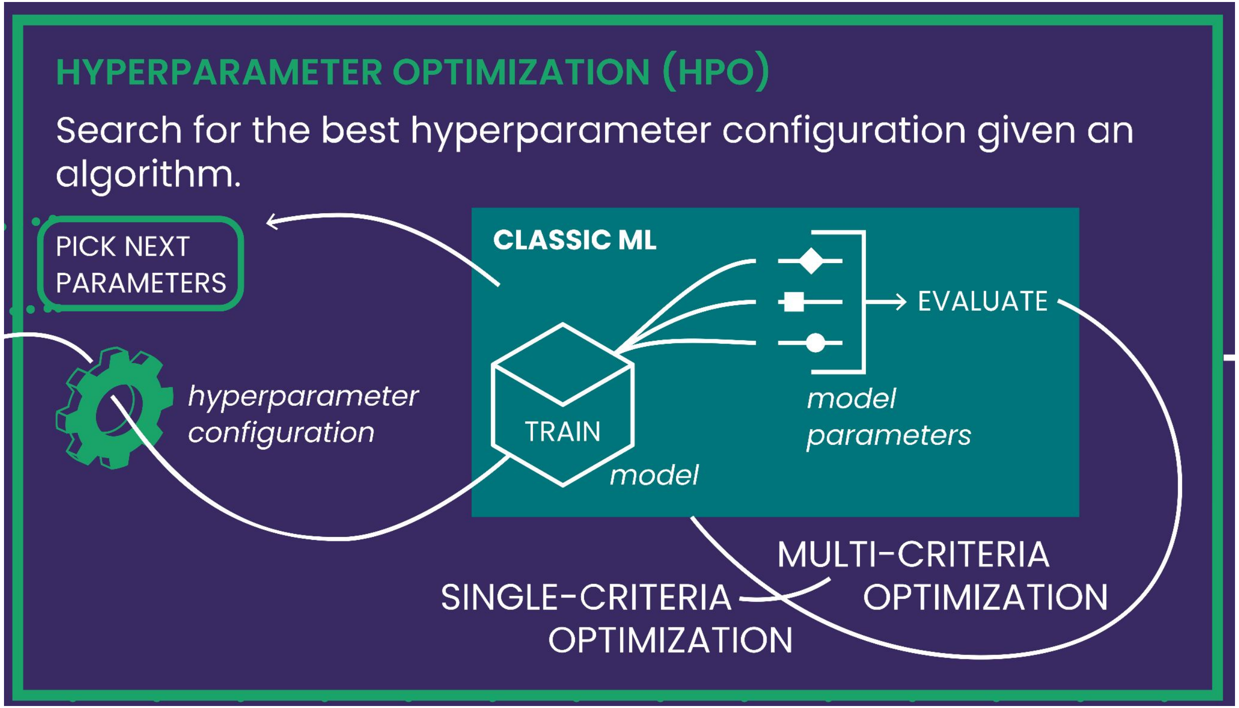
AutoML A-Z


AutoML
Optimization and automation of tedious design decisions of a complete ML pipeline in order to obtain a model with peak performance.
DATA
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS
Neural Networks
PRE-PROCESSING
INTERPRETE
MODEL(S)
EVALUATE

Hyperparameter Optimization



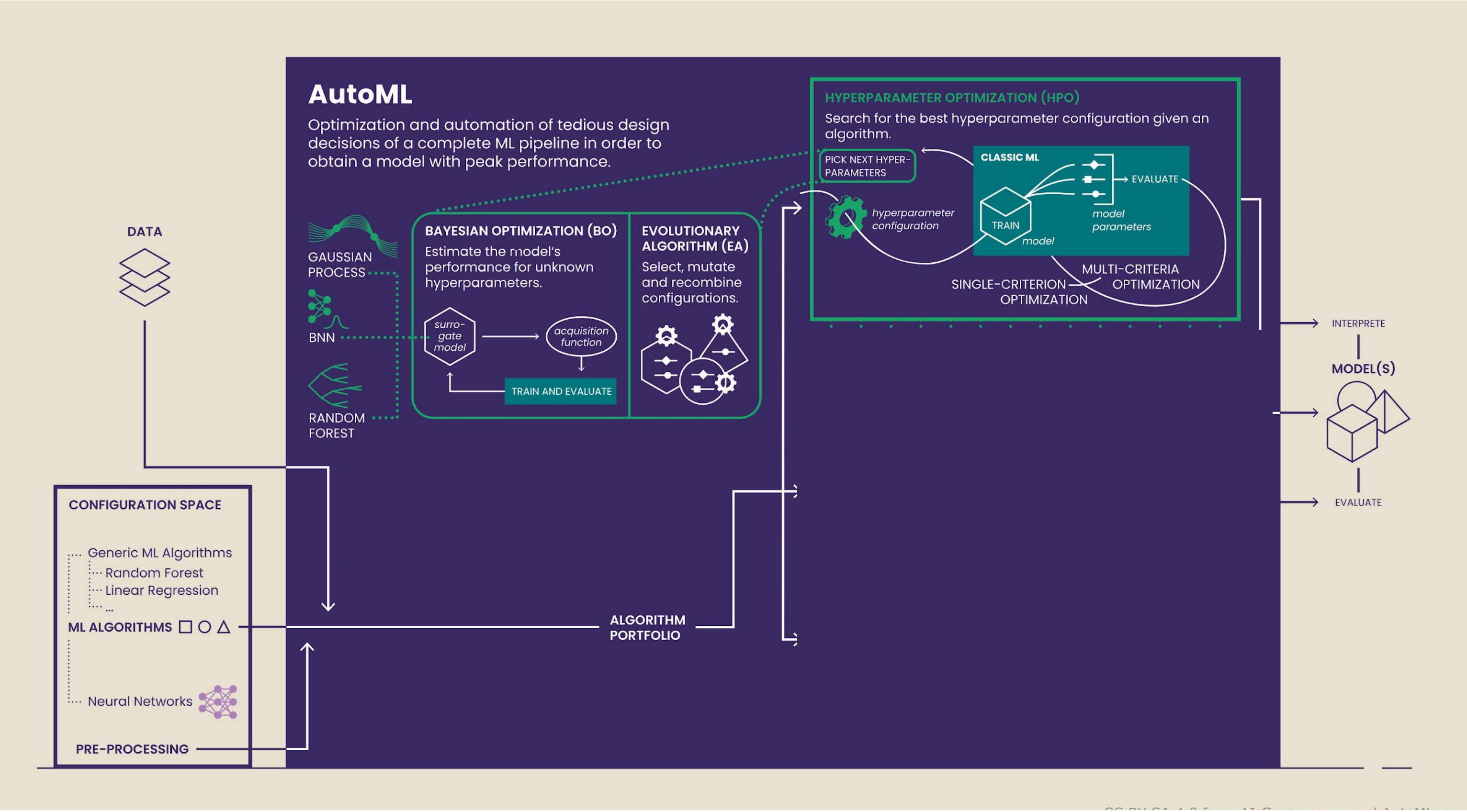
Hyperparameter Optimization



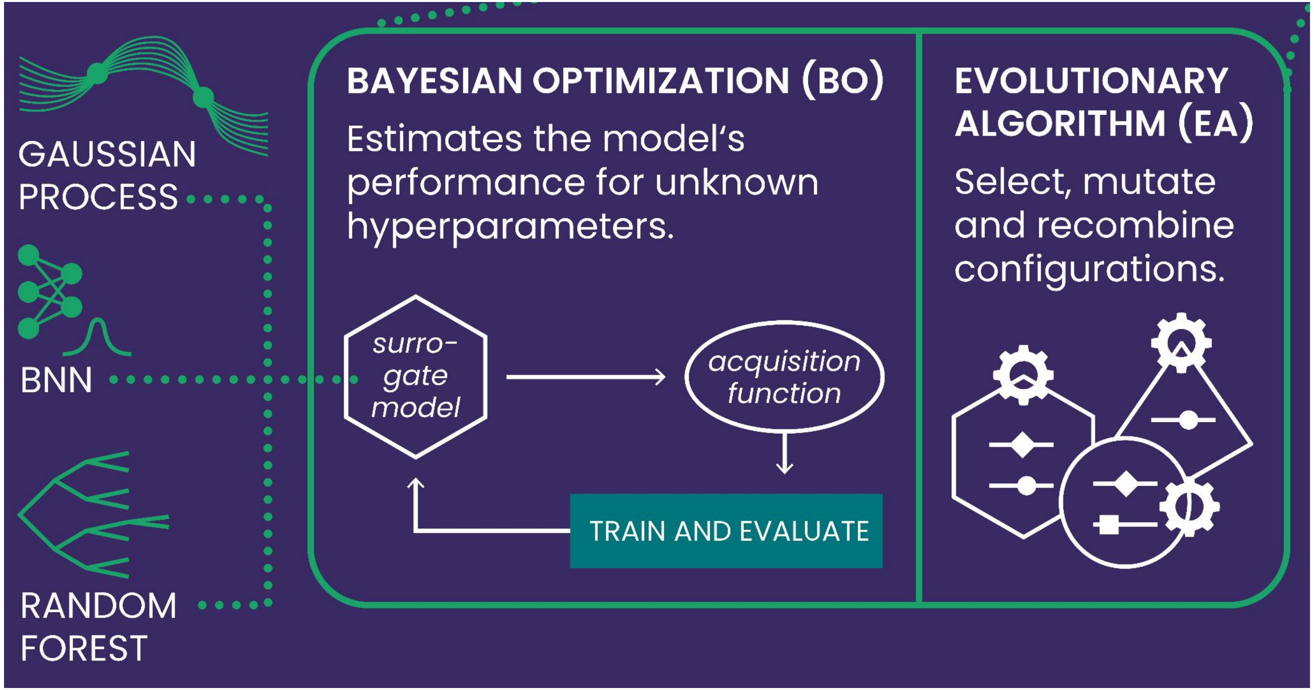
Optimize for

- Accuracy (& co)
- Memory consumption
- Energy consumption
- Inference time
- Training time
- Fairness
- Robustness
- Uncertainty quantification
- …

Optimizers for HPO



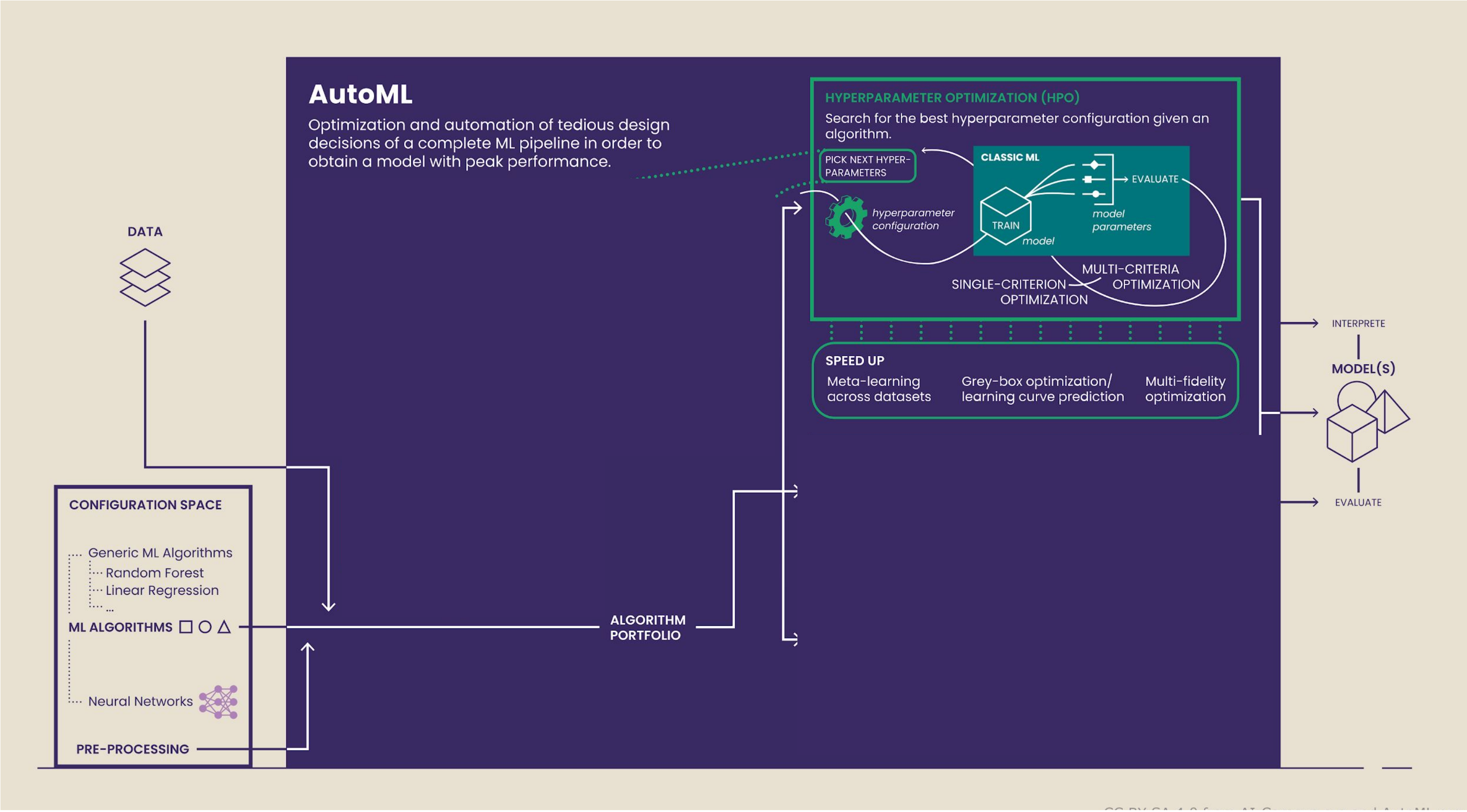
Optimizers for HPO



Further alternatives:

- Grid search
- Random search
- Reinforcement Learning
- Planning

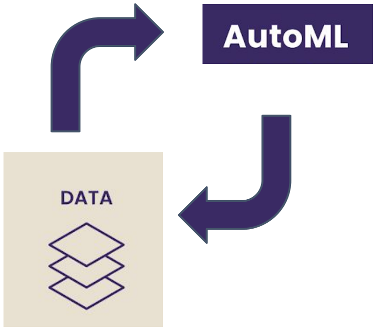
Speeding Up

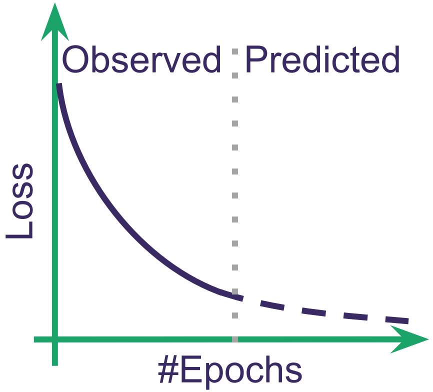

AutoML
Optimization and automation of tedious design decisions of a complete ML pipeline in order to obtain a model with peak performance.
HYPERPARAMETER OPTIMIZATION (HPO)
Search for the best hyperparameter configuration given an algorithm.
PICK NEXT HYPER-PARAMETERS
CLASSIC ML
EVALUATE
TRAIN
model
model parameters
hyperparameter configuration
MULTI-CRITERIA OPTIMIZATION
SINGLE-CRITERION OPTIMIZATION
SPEED UP
Meta-learning across datasets
Grey-box optimization/ learning curve prediction
Multi-fidelity optimization
DATA
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS
Neural Networks
PRE-PROCESSING
ALGORITHM PORTFOLIO
INTERPRETE
MODEL(S)
EVALUATE

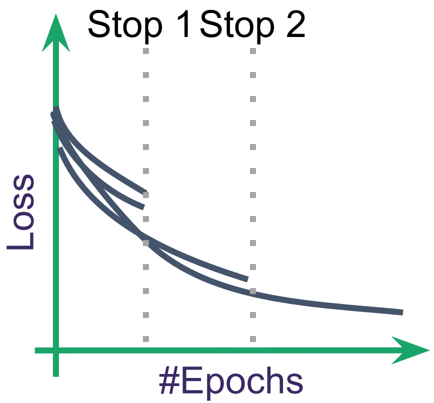
Speeding Up

SPEED UP

Meta-learning across datasets | Grey-box optimisation/ learning curve prediction | Multi-fidelity optimisation







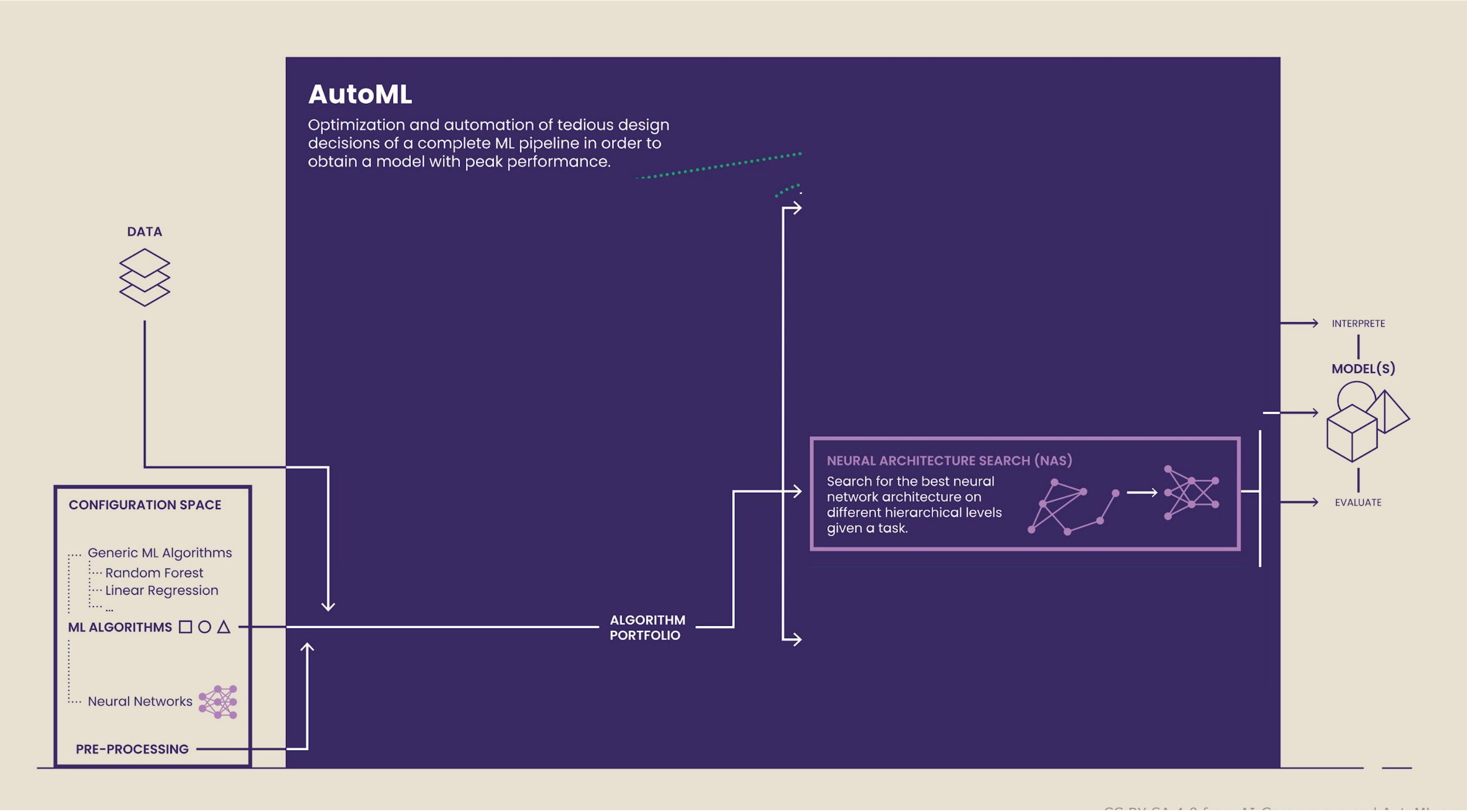
HPO Packages

Package	Complex Hyperparameter Spaces	Multi-Objective	Multi-Fidelity	Instances	CLI	Parallelism
HyperMapper	✅	✅	❌	❌	❌	❌
Optuna	✅	✅	✅	❌	✅	✅
Hyperopt	✅	❌	❌	❌	✅	✅
BoTorch	❌	✅	✅	❌	❌	✅
OpenBox	✅	✅	❌	❌	❌	✅
HpBandSter	✅	❌	✅	❌	❌	✅
SMAC	✅	✅	✅	✅	✅	✅

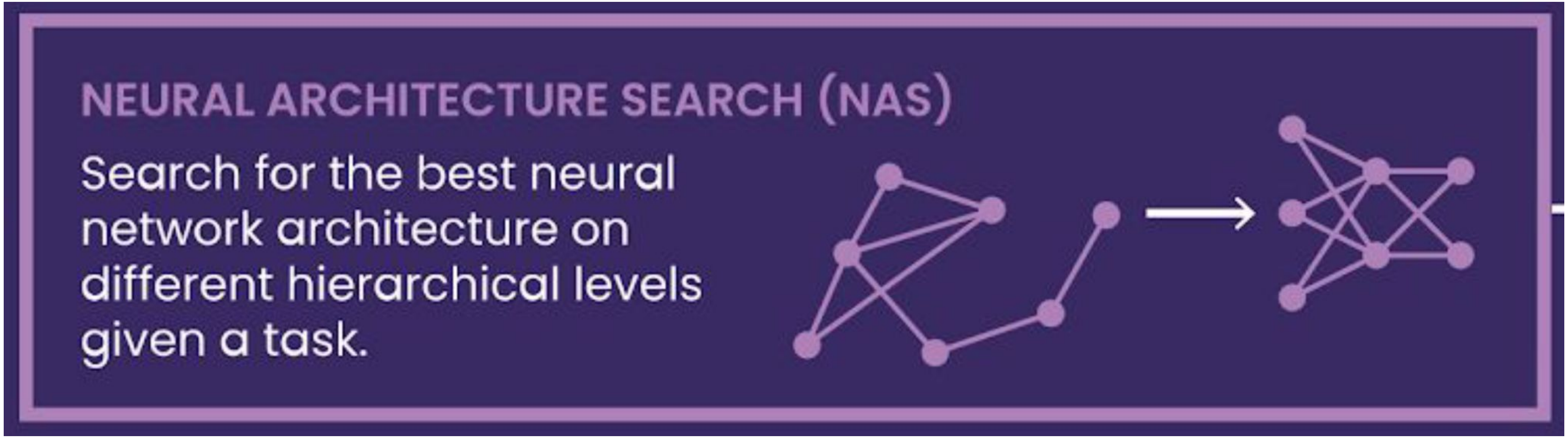
😉

last update of table in 2021

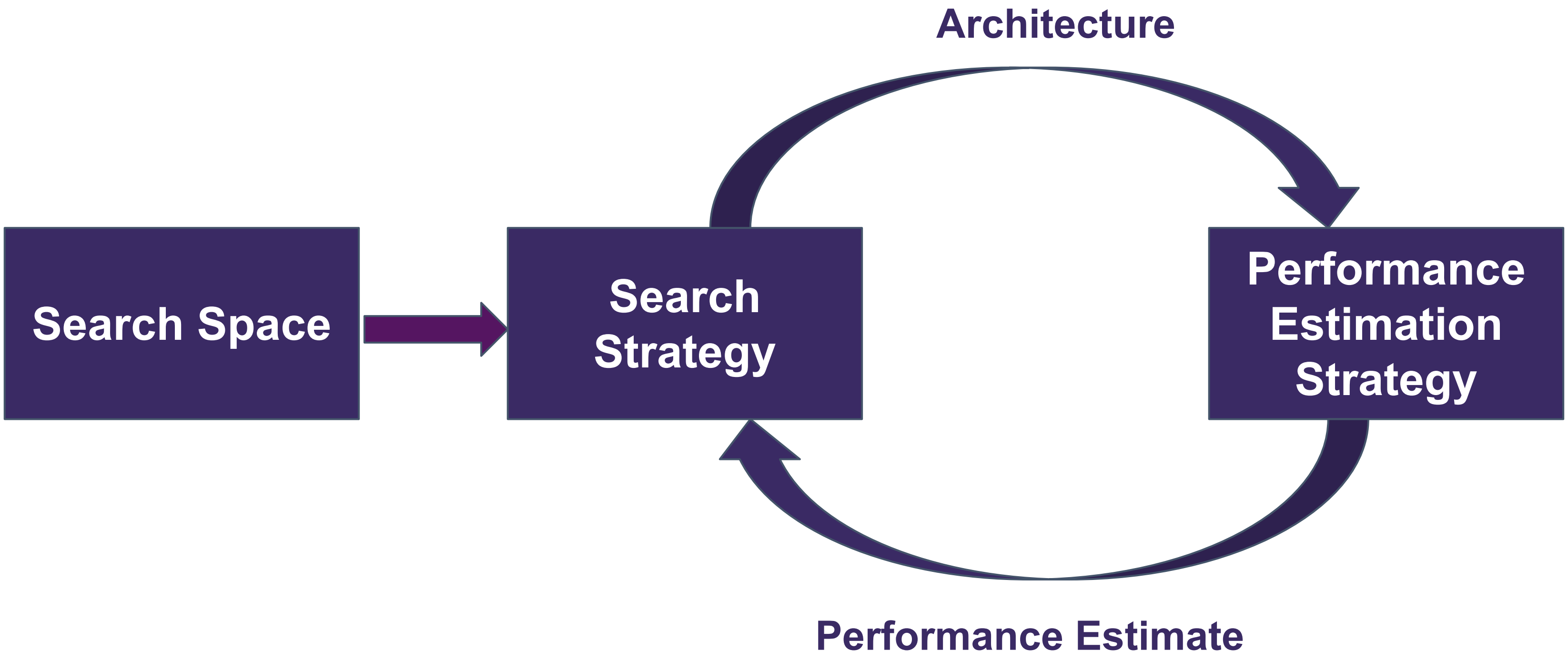
Neural Architecture Search (NAS)



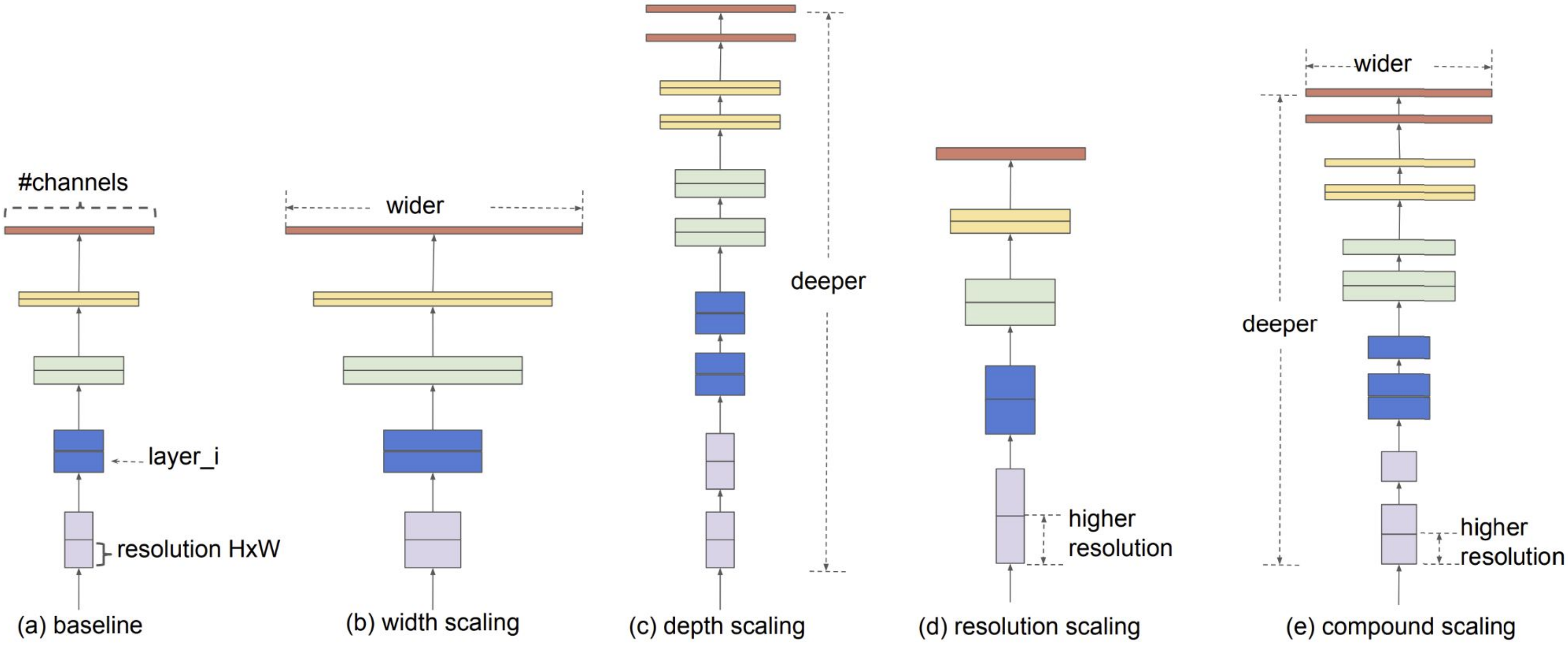
Neural Architecture Search (NAS)



The Components of NAS



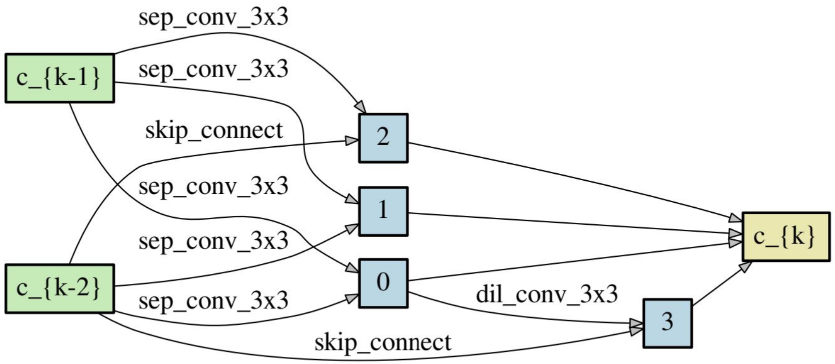
Search Space 1: Macro NAS

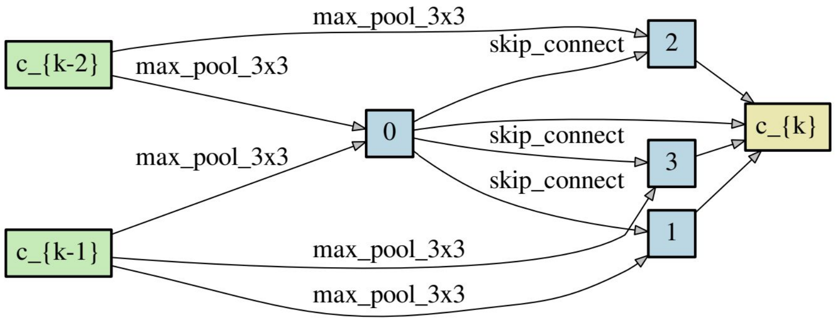


Source: [Tan & Le. 2019]

→ direct relationship to HPO: NAS as HPO

Search Space 2: Cell-based NAS





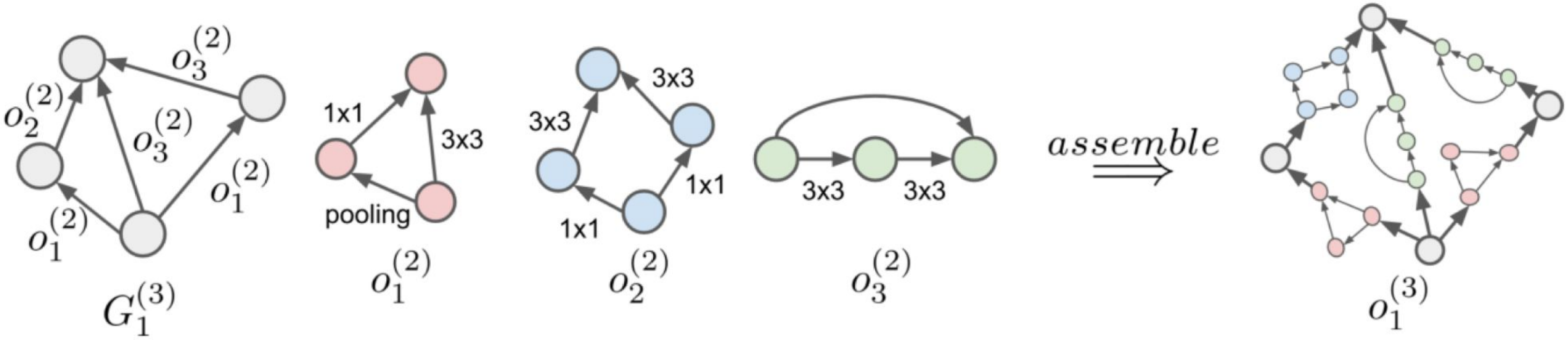
Source: [Liu et al. 2019]

Search Space 3: Hierarchical NAS

Search on multiple levels of the hierarchy

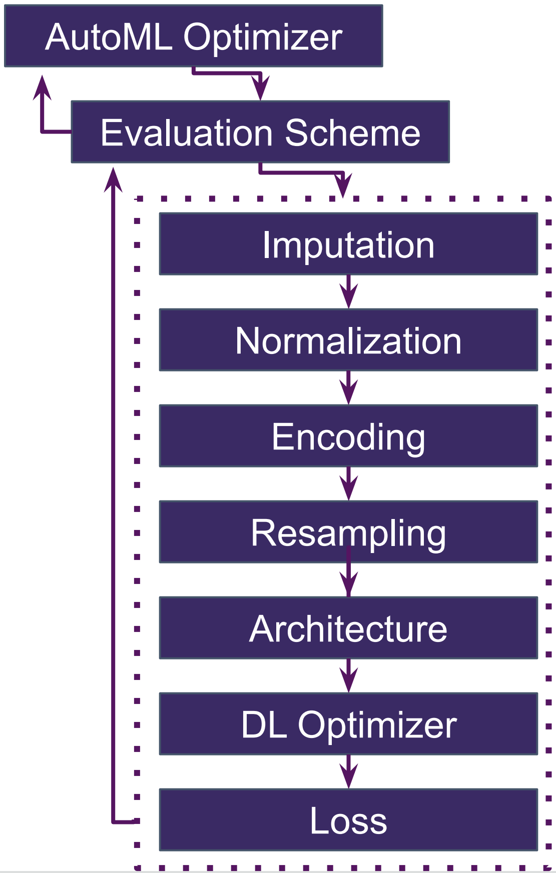
- **Lower levels:** create reusable building blocks
- **Higher levels:** combine building blocks

Like transformers are composed of lower-level building blocks (e.g., attention)



Source: [Liu et al, 2018]

AutoDL: Joint NAS & HPO



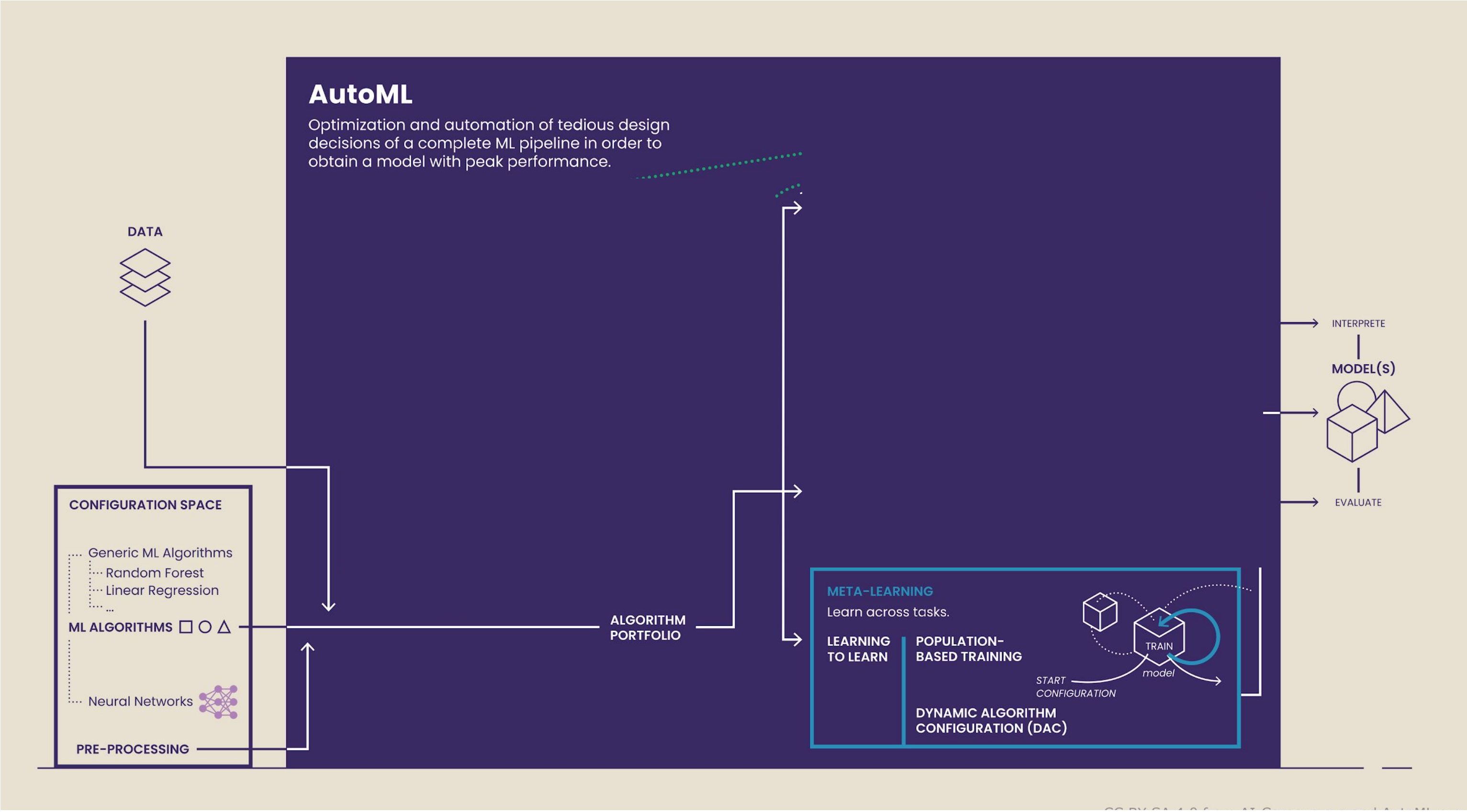
Inspired by [Zimmer et al. 2021]

1. DL also includes complex pipelines
2. NAS & HPO need to go hand in hand

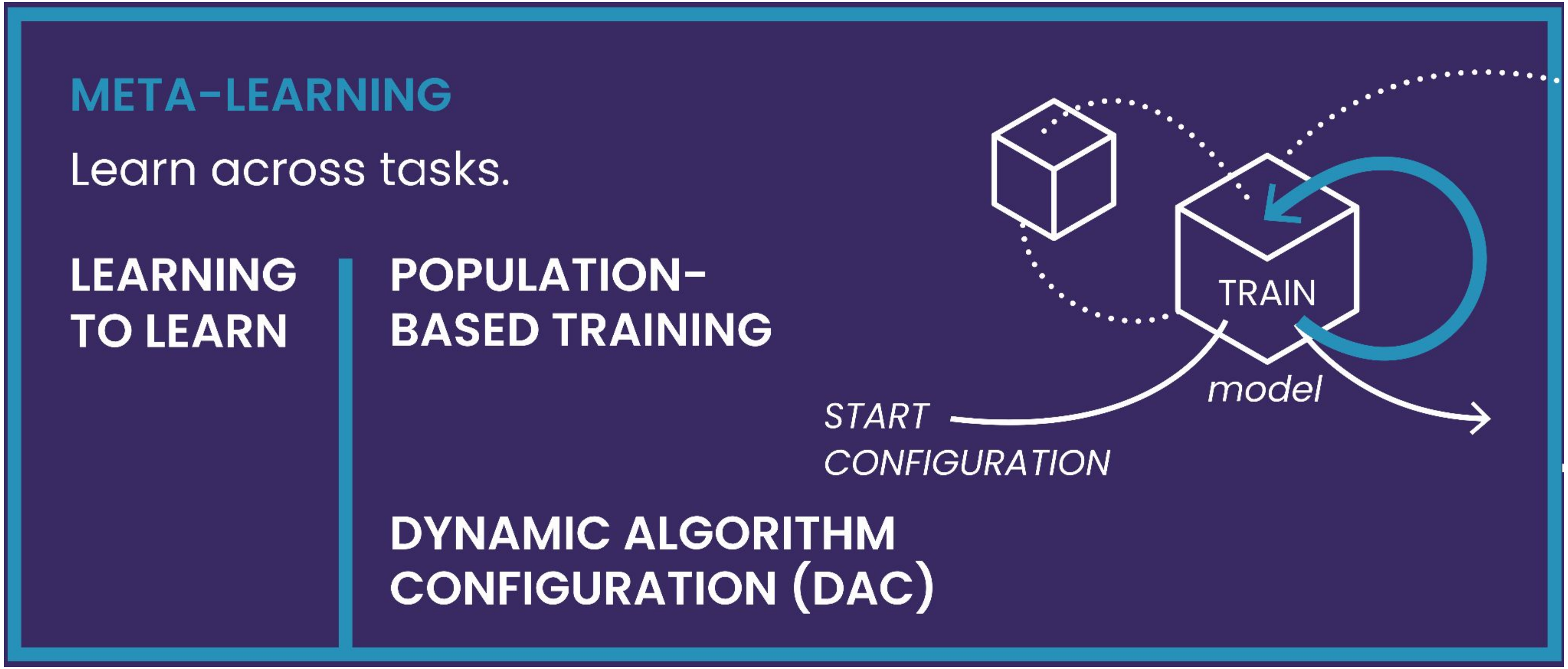
⇒ Auto-PyTorch [Zimmer et al. 2021] and Auto-PyTorch for Time Series Forecasting [Deng et al. 2022]

```
# initialise Auto-PyTorch api
api = TabularClassificationTask()

# Search for an ensemble of machine learning algorithms
api.search(
    X_train=X_train,
    y_train=y_train,
    X_test=X_test,
    y_test=y_test,
    optimize_metric='accuracy',
    total_walltime_limit=300,
    func_eval_time_limit_secs=50
)

# Calculate test accuracy
y_pred = api.predict(X_test)
```

Meta-Learning



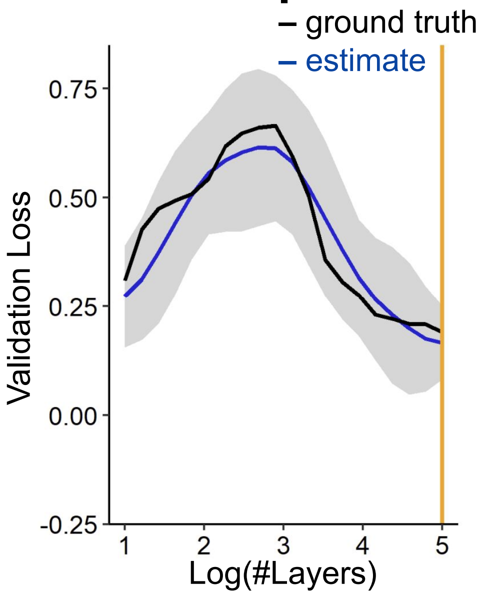
Meta-Learning



Learning about Learning Algorithms

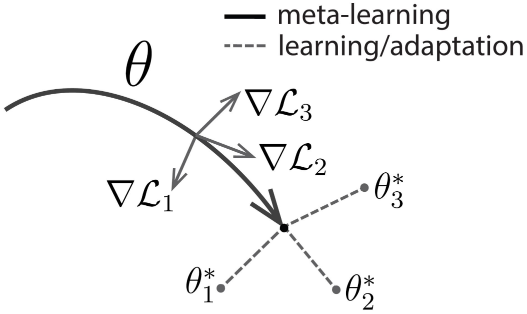
Performance prediction

Hyperparameter Effects & Importance



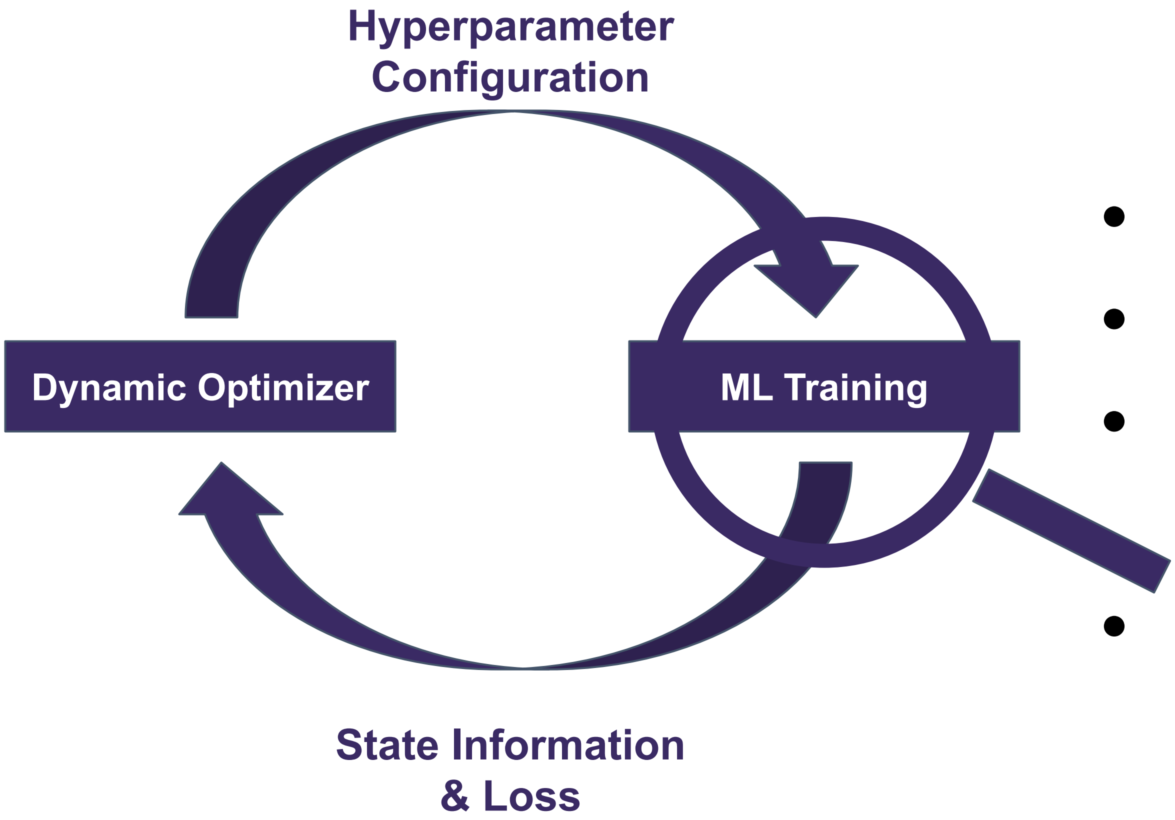
Source: [Moosbauer et al. 2021]

Learning NN weight initializations



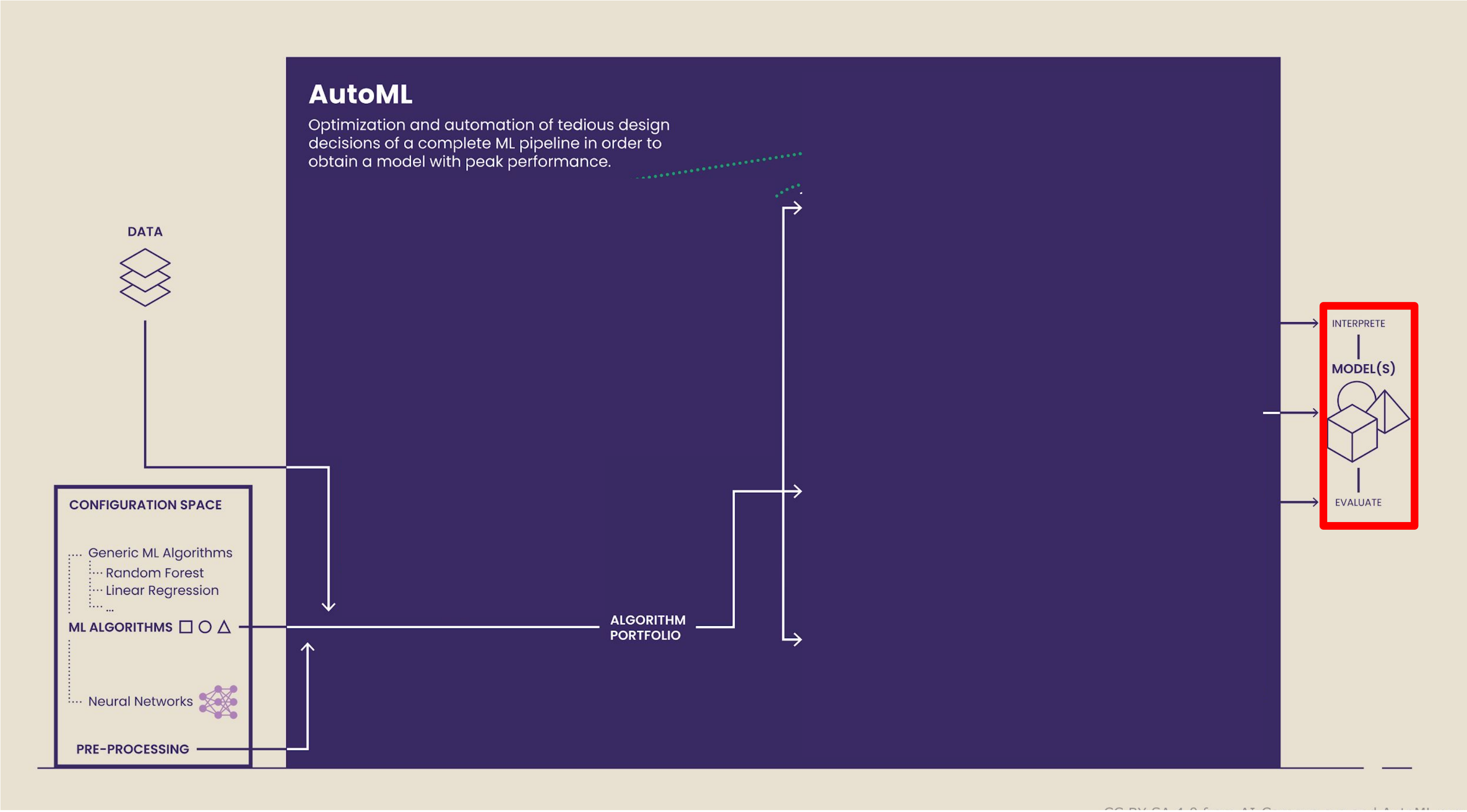
Source: [Finn et al, 2017]

Dynamic AutoML

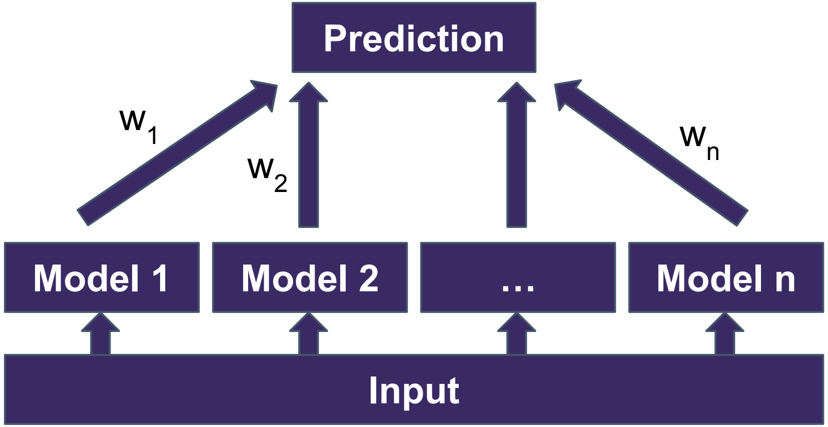


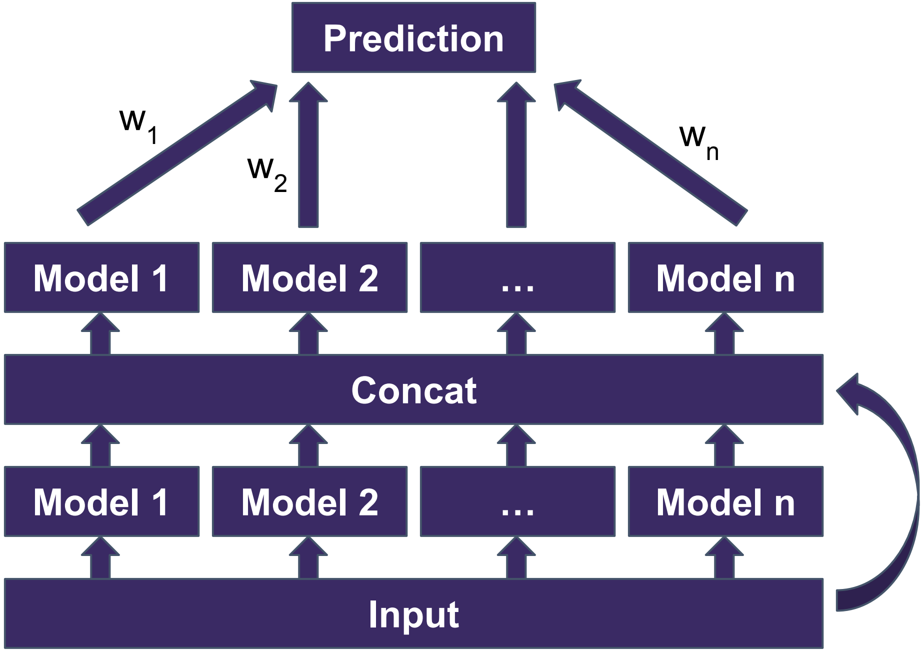
- Population-based Training [Jaderberg et al. 2017]
- Population-based Bandits [Parker-Holder et al. 2020]
- Dynamic Algorithm Configuration via RL [Biedenkapp et al. 2020, Adriaensen et al. 2022]
- Adapting Bayesian Optimization [Benjamins et al. 2022]

Final Step of AutoML



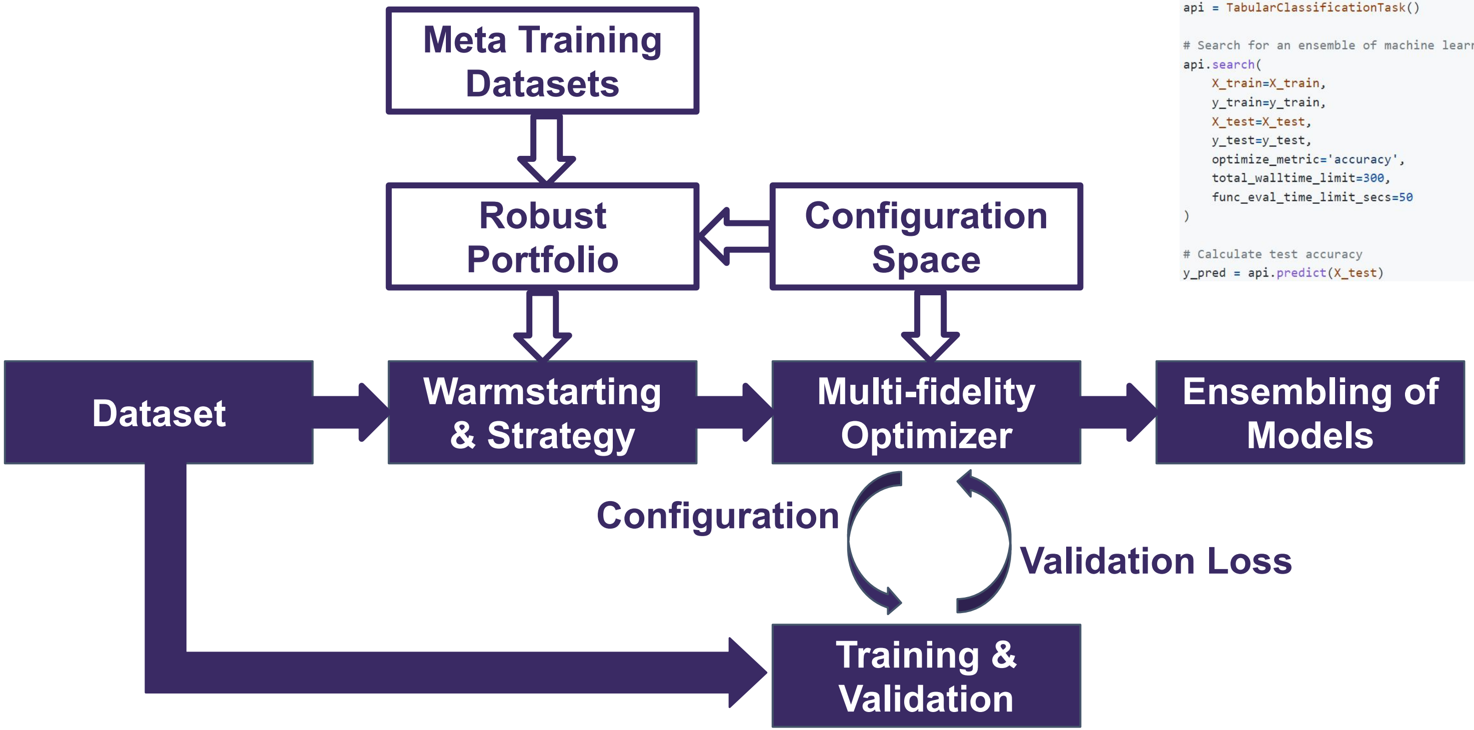
Ensembling vs Stacking





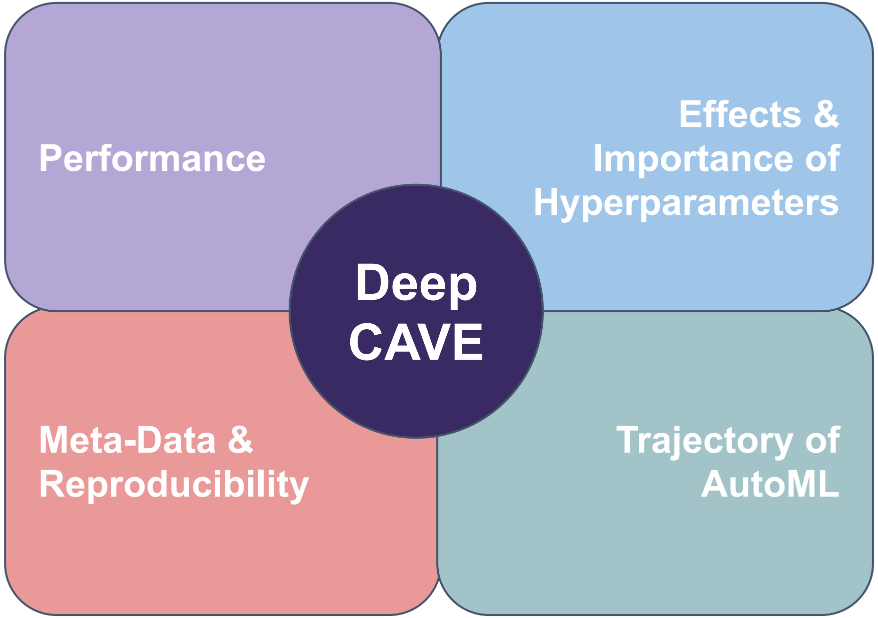
Source [Erickson et al. 2020]

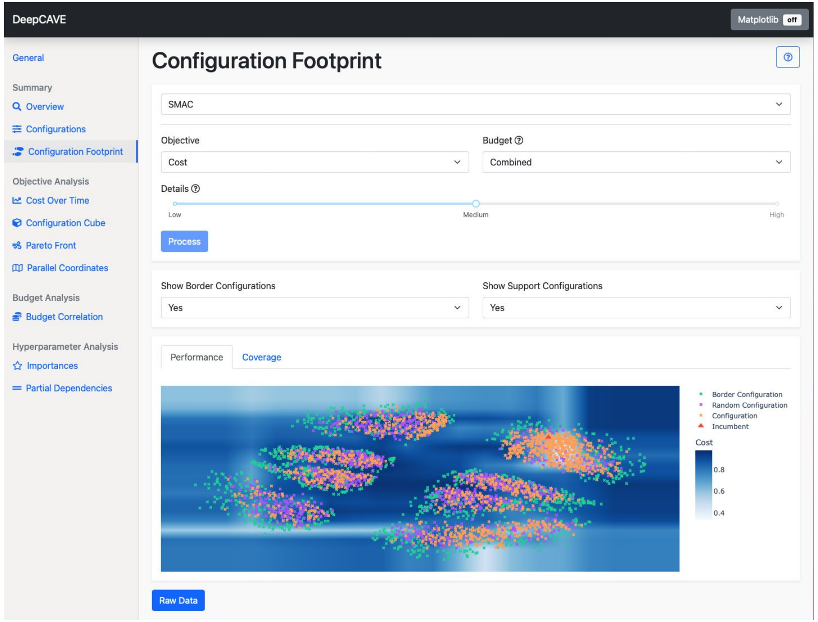
Auto-Sklearn [Feurer et al. 2015, Feurer et al. 2022] & Auto-PyTorch [Zimmer et al. 2021]



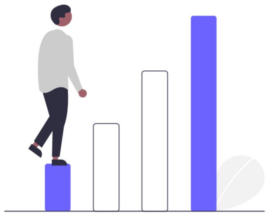

```python
# initialise Auto-PyTorch api
api = TabularClassificationTask()

# Search for an ensemble of machine learning algorithms
api.search(
    X_train=X_train,
    y_train=y_train,
    X_test=X_test,
    y_test=y_test,
    optimize_metric='accuracy',
    total_walltime_limit=300,
    func_eval_time_limit_secs=50
)

# Calculate test accuracy
y_pred = api.predict(X_test)
```

Monitoring AutoML [Sass et al. 2022]

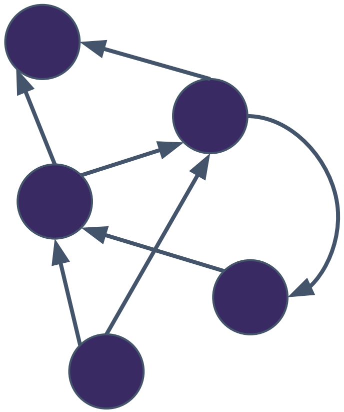




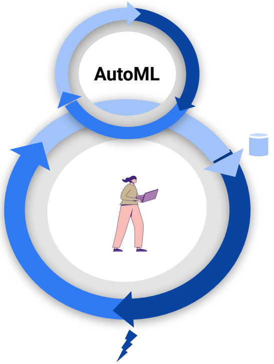
Selection of Open Challenges



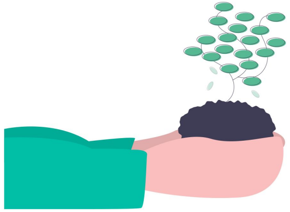
Scaling up AutoML
for very large models



Finding substantially
novel architectures

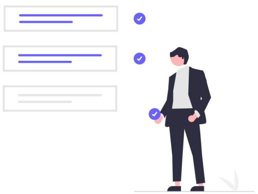


Human-centered AutoML

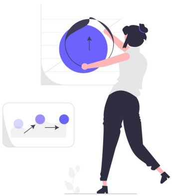


Green AutoML

Are Data Scientists still needed? Yes



Determine your objectives, metrics and constraints



Design the configuration space

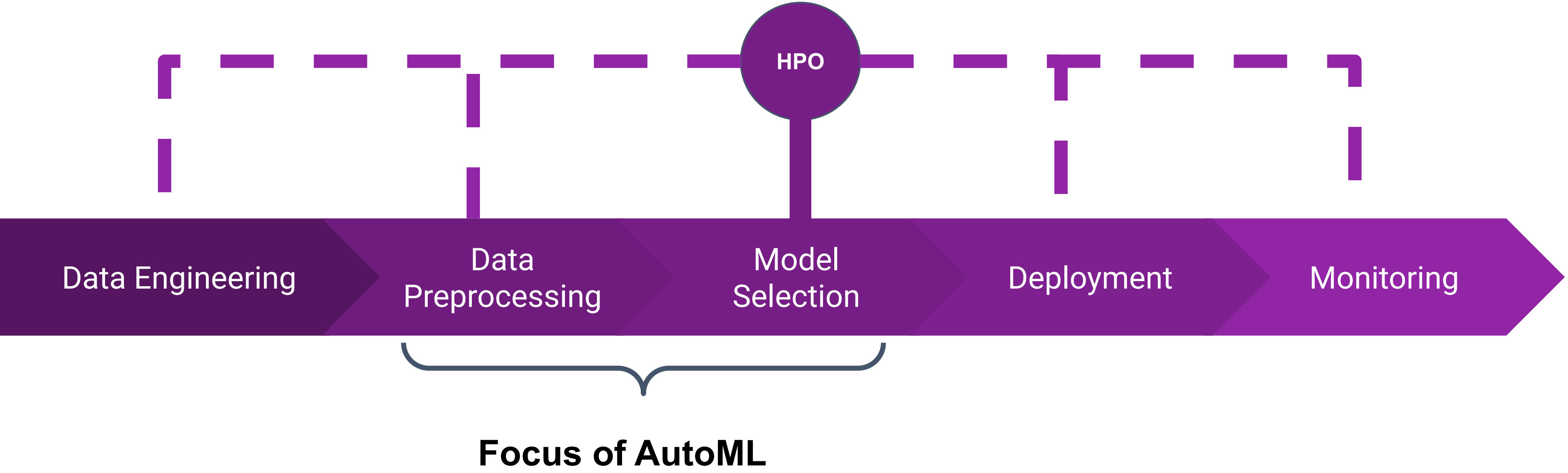
Bring in the domain knowledge

Determine Budgets

Running AutoML

Monitor AutoML

HPO → AutoML → AutoDS



Can we explain what AutoML figured out?

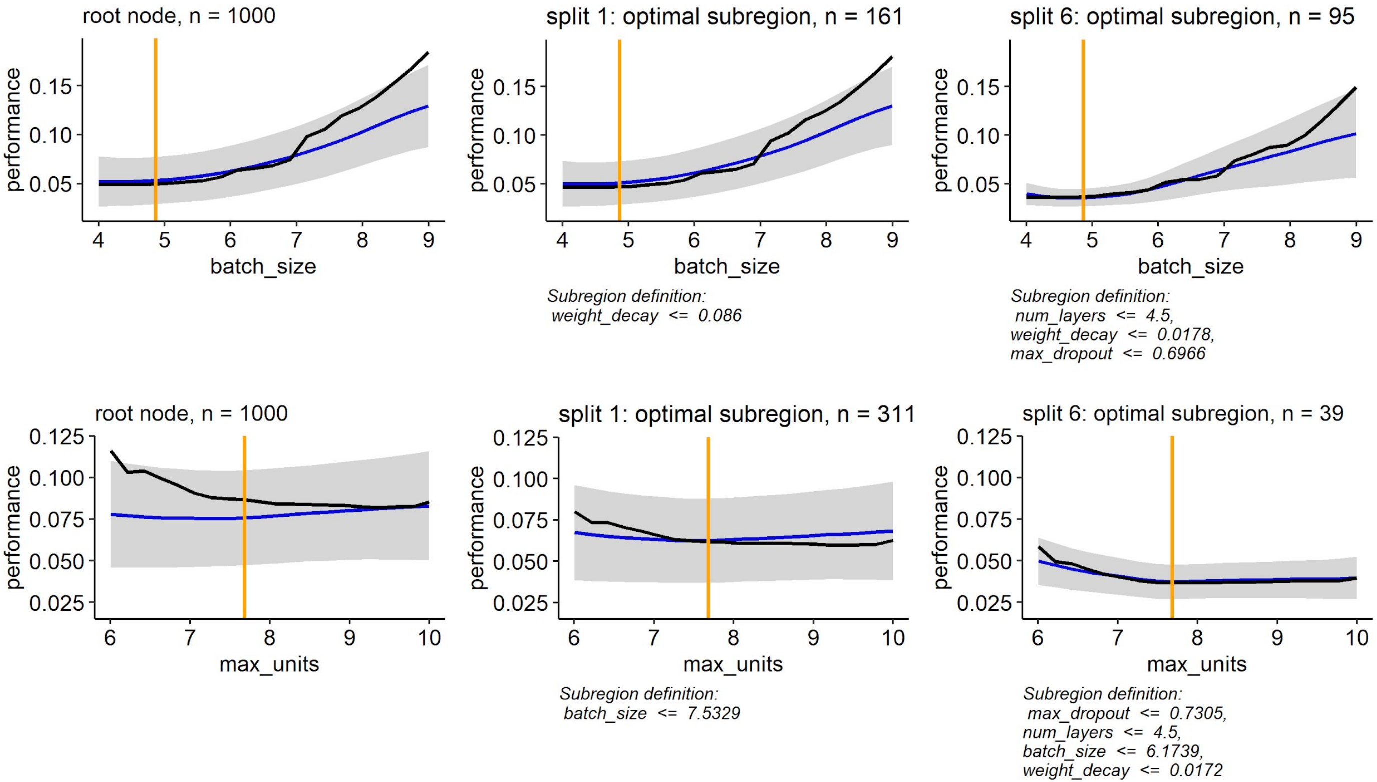
[Moosbauer et al. NeurIPS’21, Moosbauer et al. 2022]





Explaining Hyperparameter Effects via PDPs

Ground truth
PDP
incumbent



Partial Dependence Plots

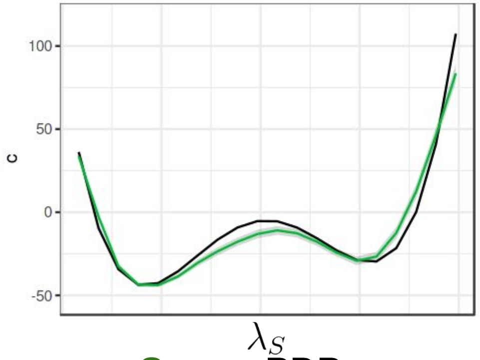
For, a subset S of the hyperparameters, the partial dependence function is:

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

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.

[Hutter et al. 2014] showed how to do this efficiently for RFs as surrogate models.

Partial Dependence Plots with Uncertainties

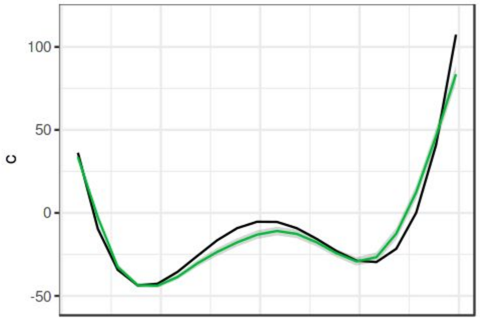
$$\begin{aligned} &\hat{s}^2_S(\lambda_S) \\ &= \mathbb{V}_{\hat{c}}\left[\hat{c}_S\left(\lambda_S\right)\right] \\ &= \mathbb{V}_{\hat{c}}\left[\frac{1}{n}\sum_{i=1}^{n}\hat{c}\left(\lambda_S, \lambda_C^{(i)}\right)\right] \\ &= \frac{1}{n^2}1^\top \hat{K}\left(\lambda_S\right)\ 1. \end{aligned}$$

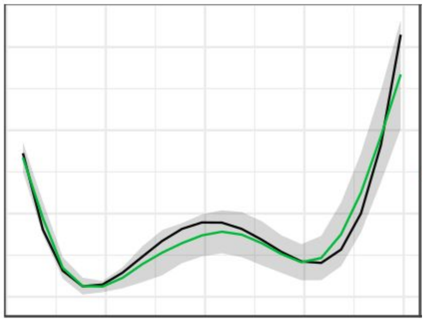
→ requires a kernel correctly specifying the covariance structure (e.g., GPs).

Approximation:

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

→ Model-agnostic (local) approximation

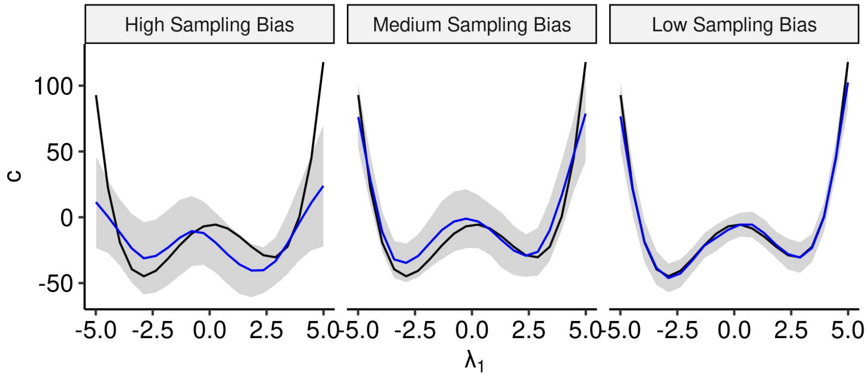




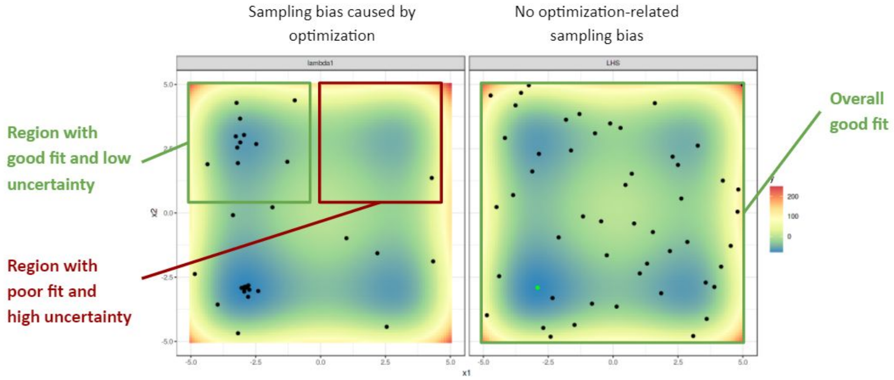
Ground truth
PDP
Uncertainty

Impact of Sampling Bias in Explaining AutoML

- Simply using all observations from AutoML tools might lead to misleading PDPs
- Uncertainty estimates help to quantify the poor fits



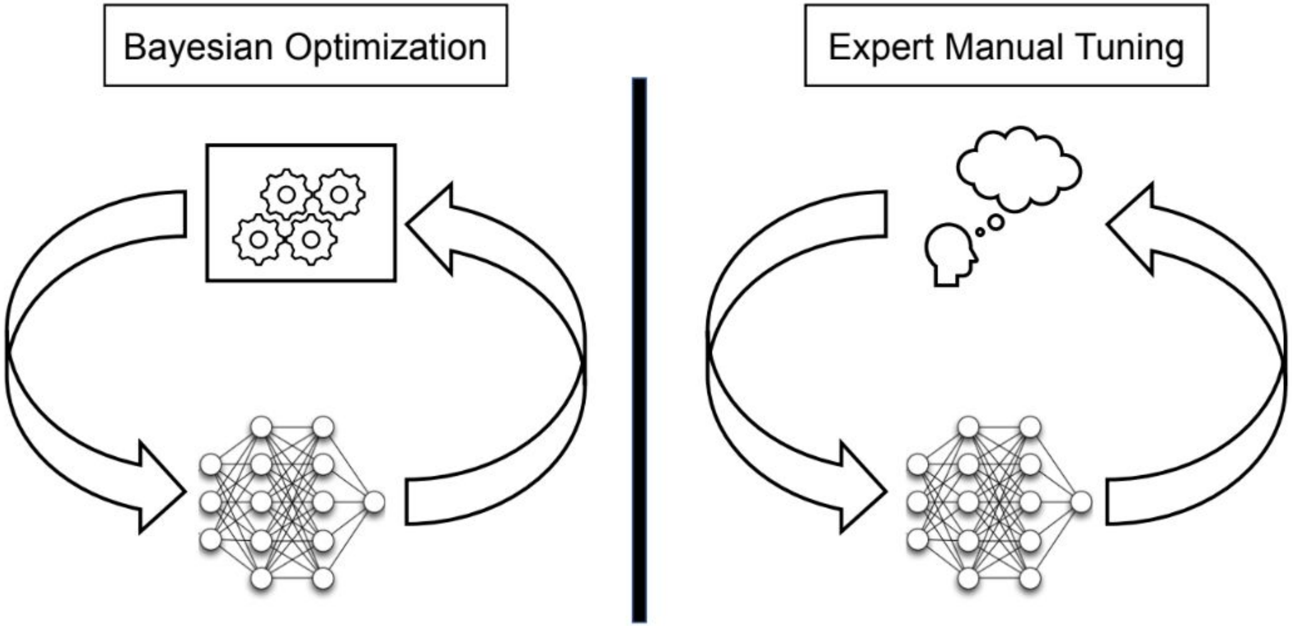
→ of course, sampling bias is wanted and the solution cannot be to change the sampling behavior



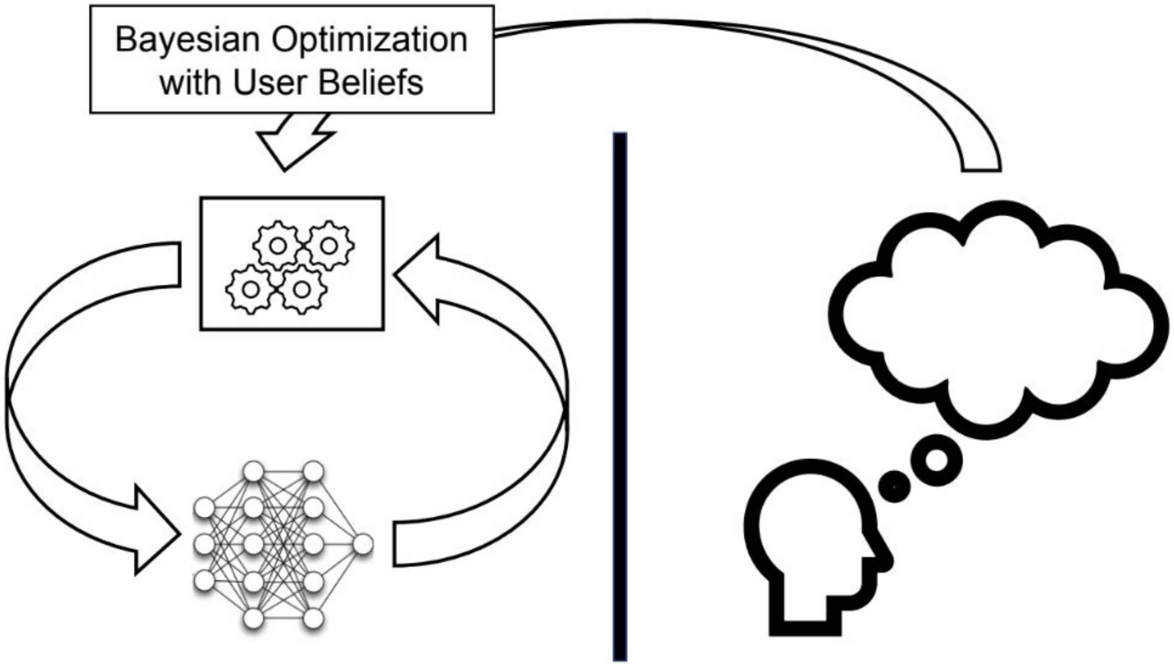
Can AutoML consider expert knowledge?

[Hvarfner et al. ICLR'22]

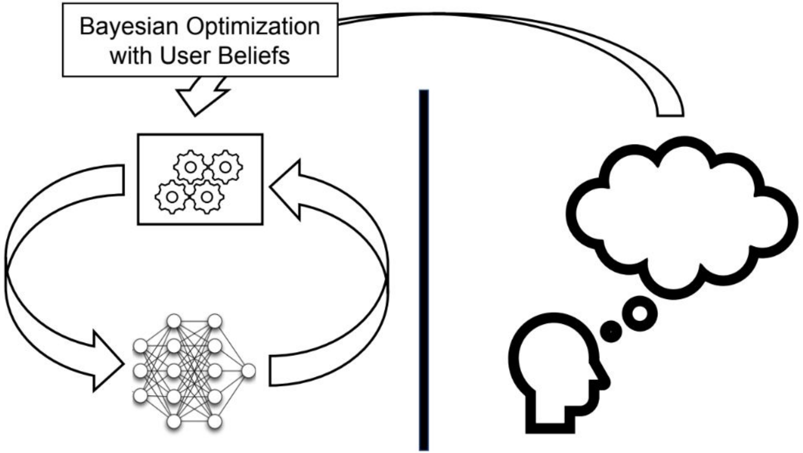
Bayesian Optimization vs Manual Tuning



Bayesian Optimization with Expert Knowledge



piBO



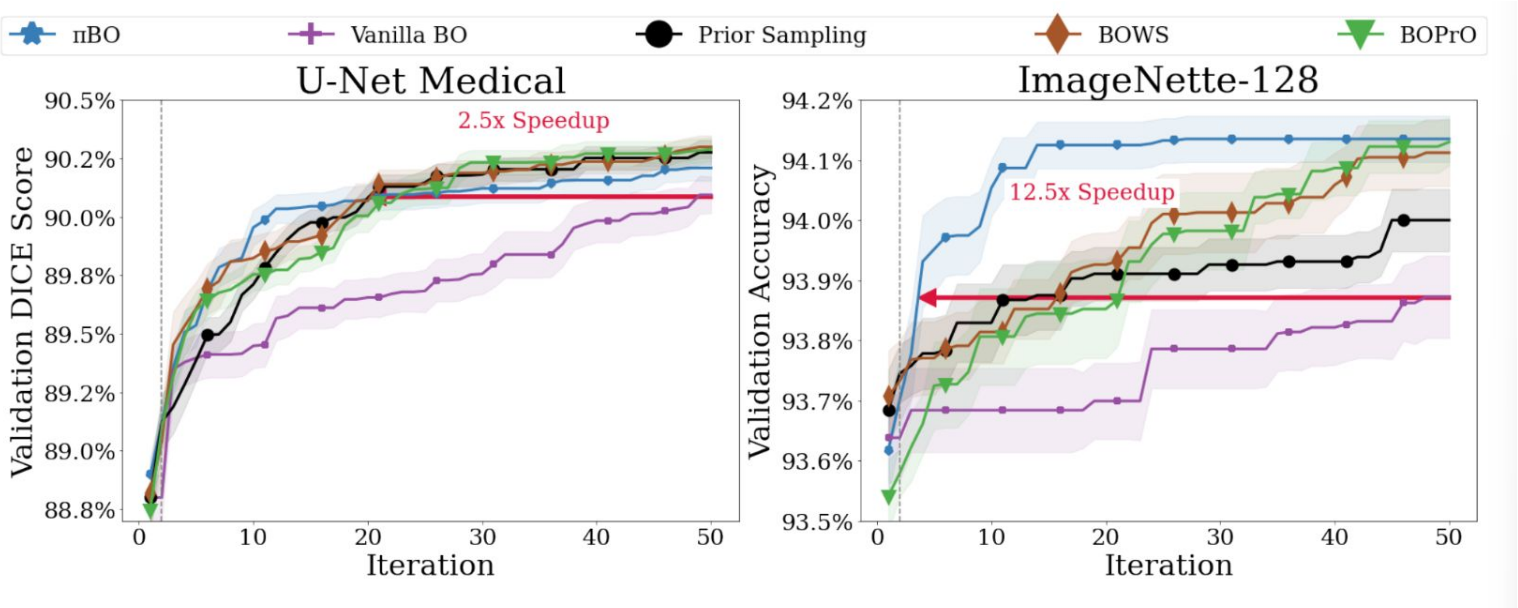
$$\boldsymbol{x}_n \in \underset{\boldsymbol{x}\in\mathcal{X}}{\arg\max}\, \alpha(\boldsymbol{x}, \mathcal{D}_n)\pi(\boldsymbol{x})^{\beta/n}$$

Acquisition Function

User Prior

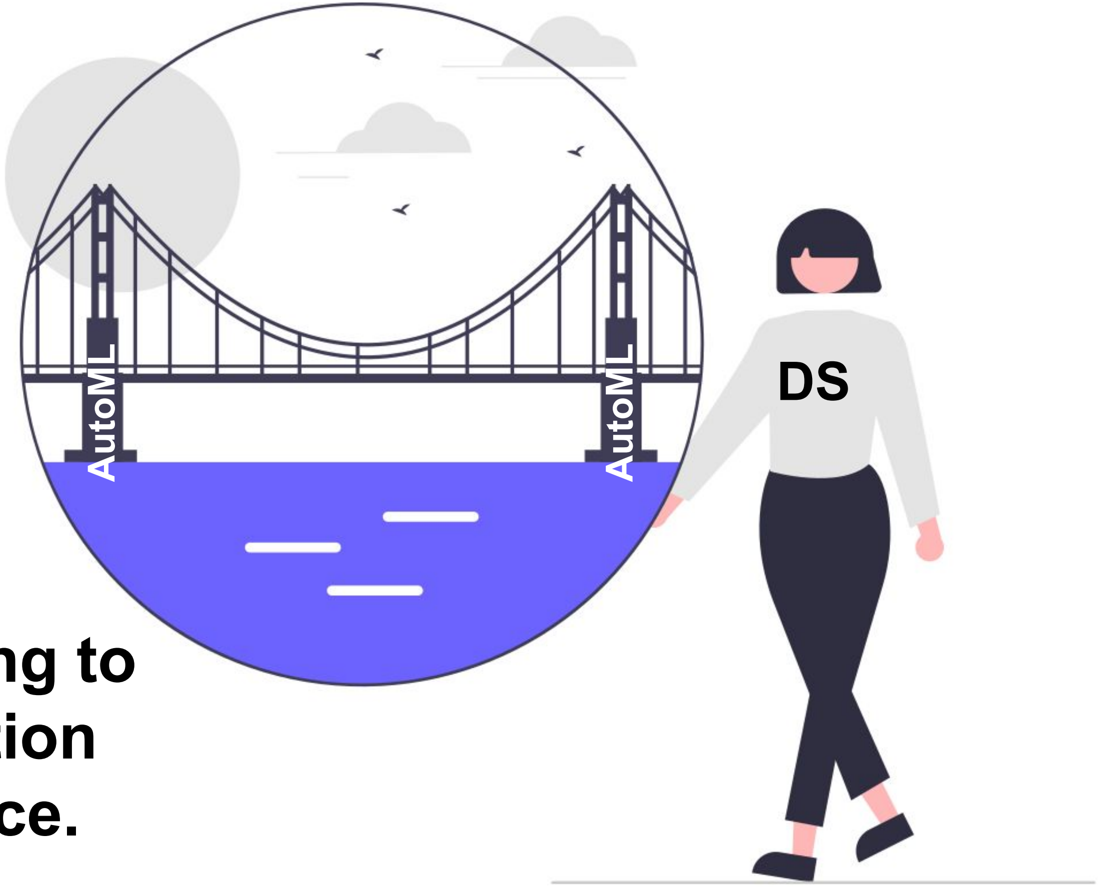
Speed of forgetting user prior

PiBO [Hvarfner et al. ICLR'22]



- ➔ Uses expert knowledge to speed up Bayesian Optimization
- ➔ Robust also against wrong believes
- ➔ Substantially speeds up AutoML

Will AutoML replace Data Scientists?



AutoML: Helping to bridge application and data science.

AutoML.org

/AutoML_org/

/automl/

https://tinyurl.com/automlyt

LUH|AI

Leibniz Universität Hannover

Funded by:

erc
European Research Council
Established by the European Commission

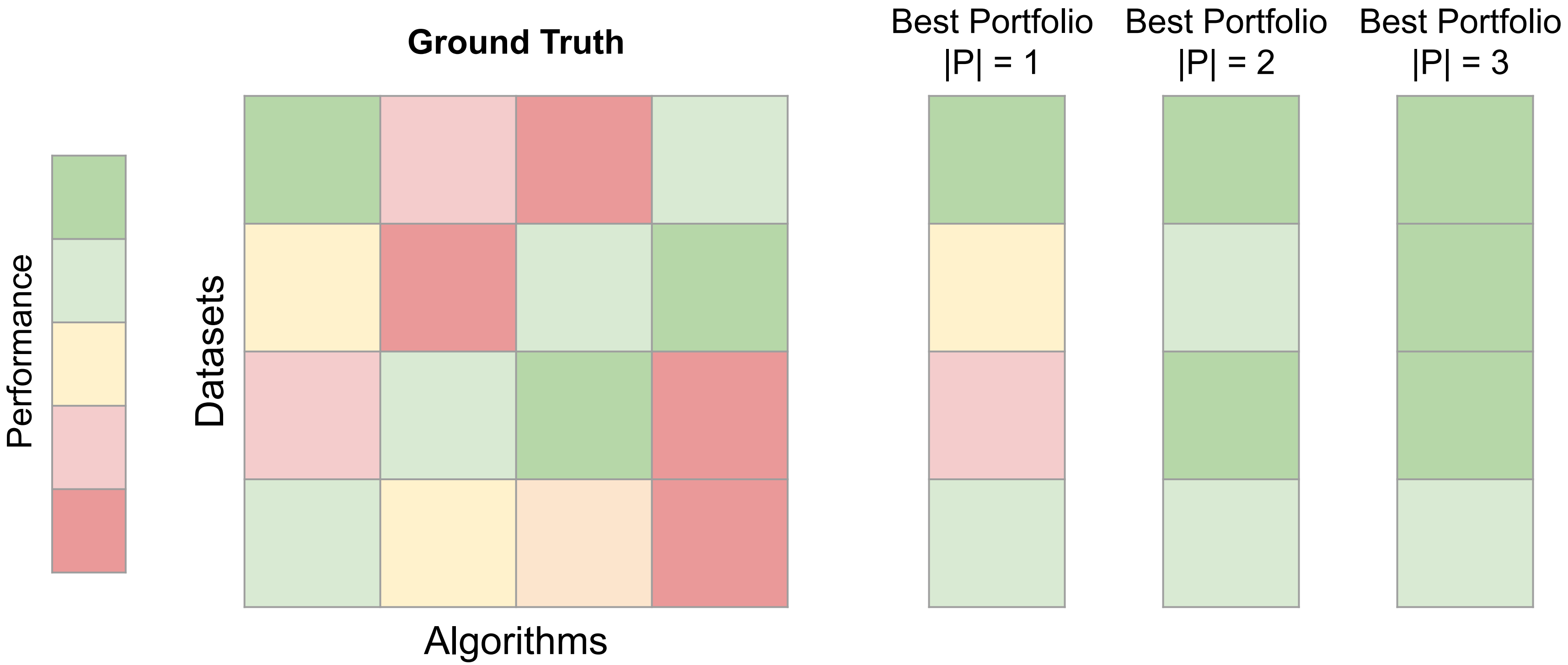
Federal Ministry of Education and Research

DFG
Deutsche Forschungsgemeinschaft

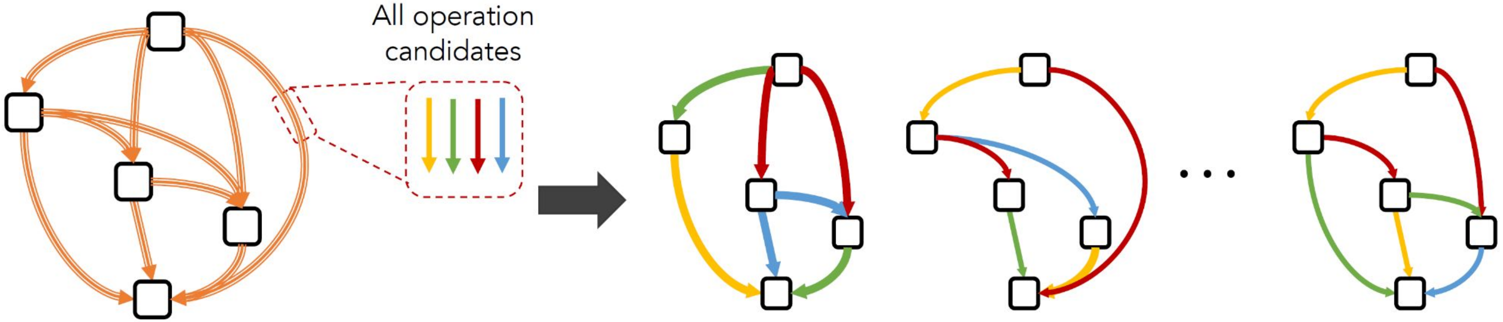
Federal Ministry for Economic Affairs and Energy

Backup slides

Portfolios for Warmstarting [Feurer et al. 2022]



Oneshot NAS: Weight Sharing Across Architectures



- For each choice between operations, the supernet includes all of them
- A linear number of weights shared by an exponential number of architectures
- Thus, updating the weights of one architecture simultaneously updates parts of the weights of exponentially many other architectures

Zero-Cost Proxies for NAS

Very hot topic in NAS, but no consistent improvements over trivial baselines, such as #parameters or FLOPs

Zero-Cost Proxies for NAS

ZC proxies are a particular type of performance predictor

- They aim to judge the performance of an architecture in a few seconds
- Often by a single forward pass on a mini-batch
- Thus, the term “zero-cost”

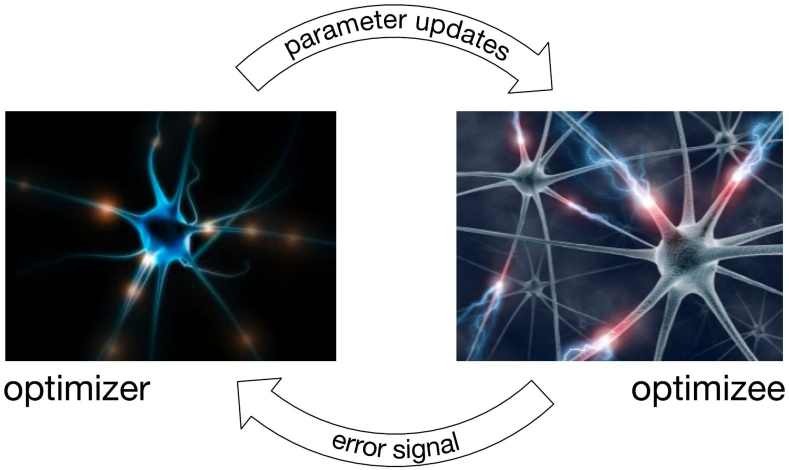
Examples

- Change of error when dropping network weights
- Dissimilarity of activation patterns for points in a batch

Very hot topic in NAS, but no consistent improvements over using number of parameters or FLOPS

Learning to learn

E.g., “Learning to learn by gradient descent by gradient descent“ [Chen et al. 2016]



Source: [Chen et al. 2016]

E.g., Alpha-Zero [Silver et al. 2017]

Maturity of AutoML