AutoML 101

Marius Lindauer

@LindauerMarius
@AutoML_org

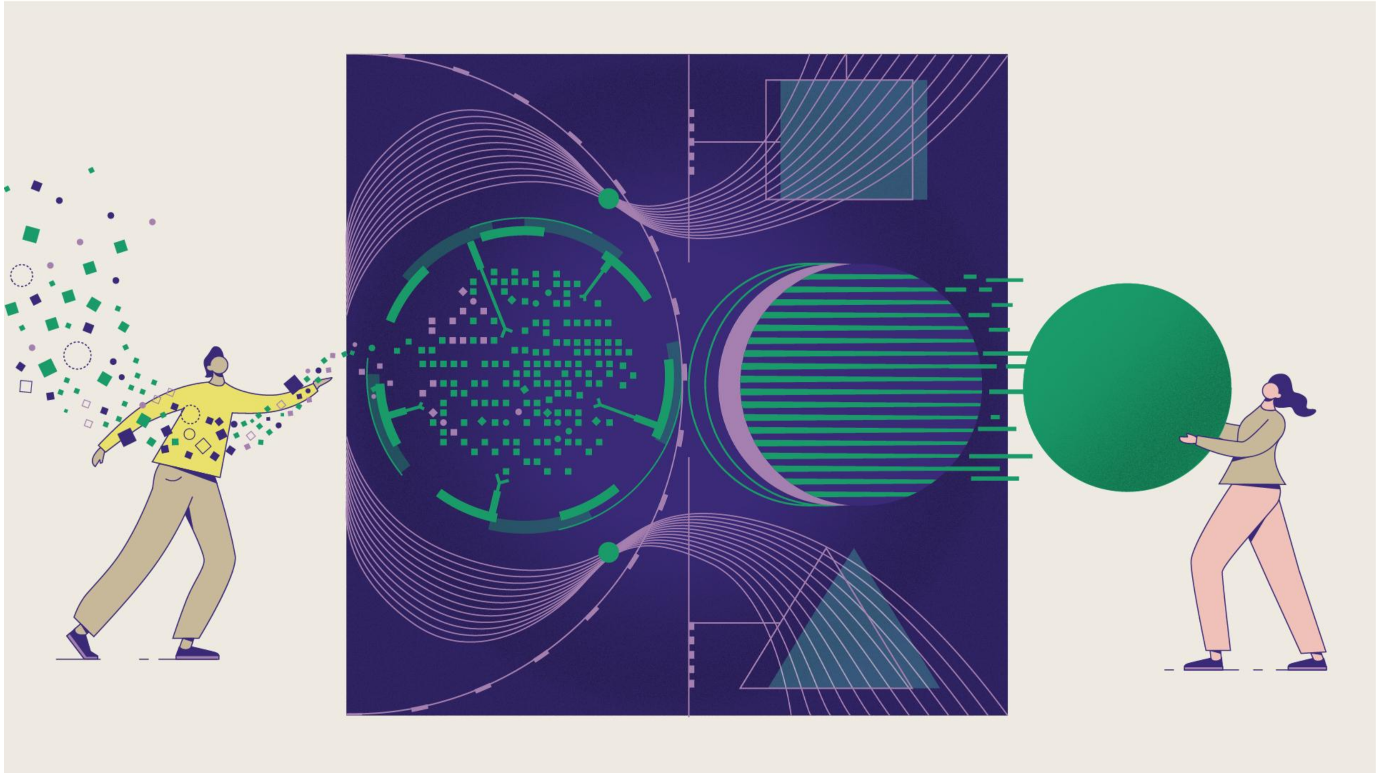
Leibniz Universität Hannover

Automated Machine Learning Hannover

AutoML.org

* Slides available at automl.org/talks

Why AutoML?



Success of Machine Learning

Astronomy

Robotic

Teaching

Creative Arts

Material Design

Energy

Game Play

Search

Chemistry

Health Care

Physics

Image Recognition

Weather Prediction

Product Recommendation

Drug Discovery

Service

Manufacturing

Financial Services

Traffic Prediction

Retail

Credit Assignment

Media

Maintenance Prediction

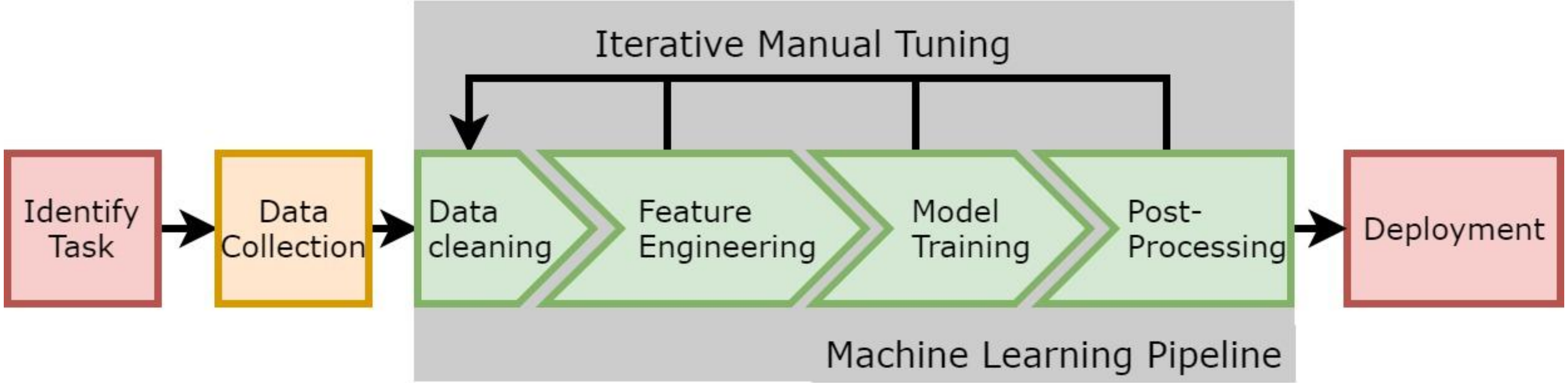
Social Media

Summary Generation

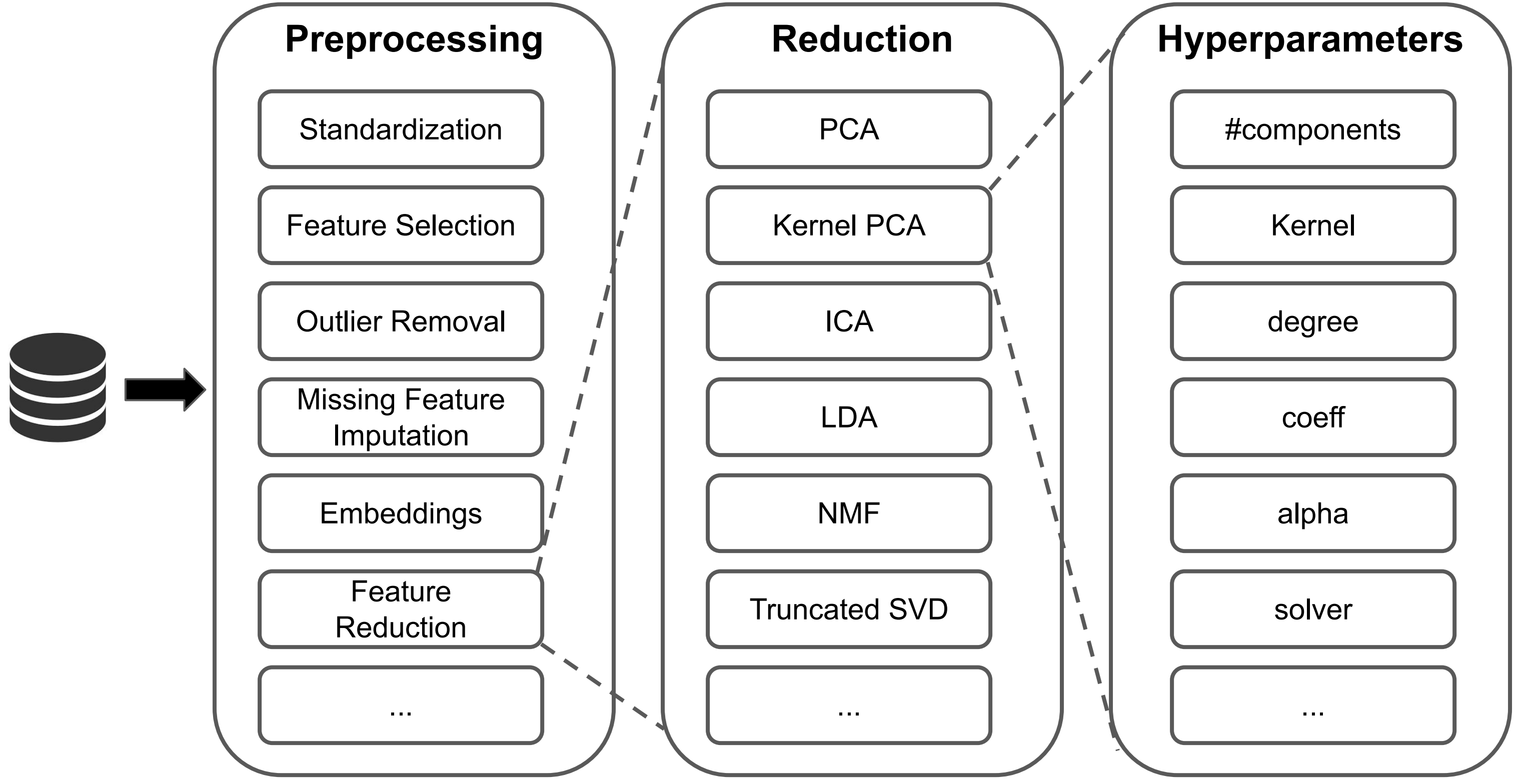
Leibniz Universität Hannover

AutoML.org

Machine Learning Pipeline

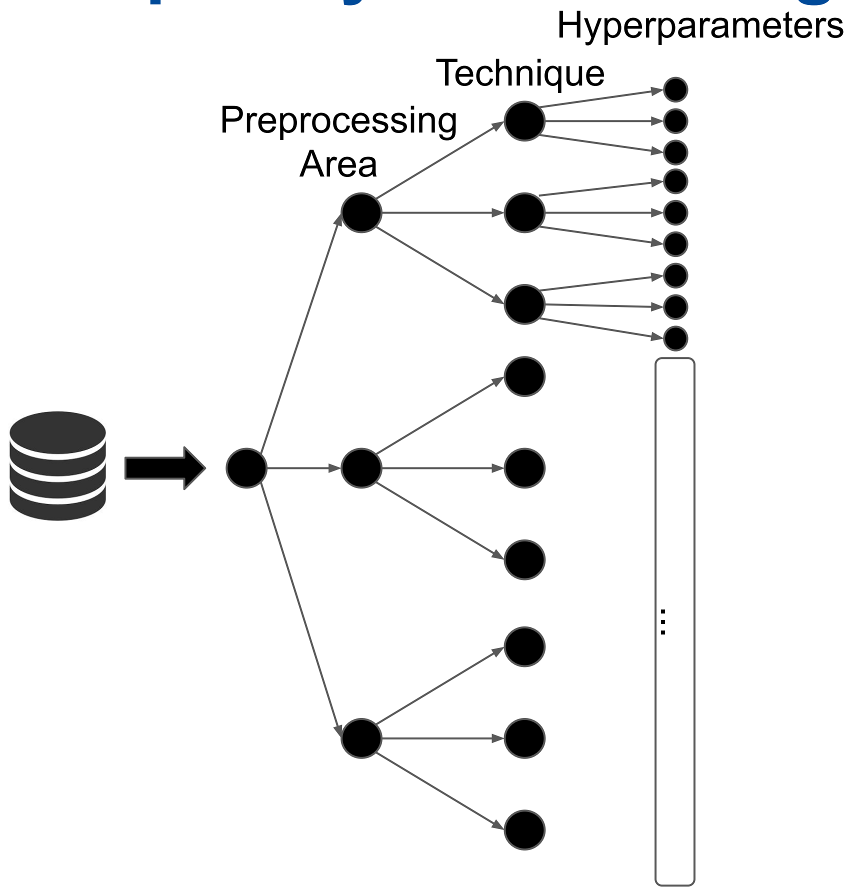


Preprocessing?



→ **We might want more than 1 data preprocessor!**

Complexity of Choosing the Preprocessing



- Naive Assumptions:
  only 3 decisions at each level
- **Possible options**: 3 x 3 x 3 = **27**

- More realistic assumption:
  at least 10 decisions at leach level
- **Possible options:** 10 x 10 x 10 = **1000**

- Choose 3 preprocessors instead of 1
  → 1000 x 1000 x 1000 =
  1 000 000 000

- Still naive!
  → Hyperparameters are often continuous and not discrete
  → **infinite amount of settings!**

Classification Algorithms

LARS

Least Squares

Elastic-net

Linear SVM

Kernel SVM

Recurrent Network

Least-Angle

Lasso

ResNet

Poly. SVM

Feedforward Network

Ridge

Bayesian Regression

Naive Bayes

Logistic Regression

SGD

LGB

Gaussian Process

Nearest Neighbor

Boosting Trees

XGB

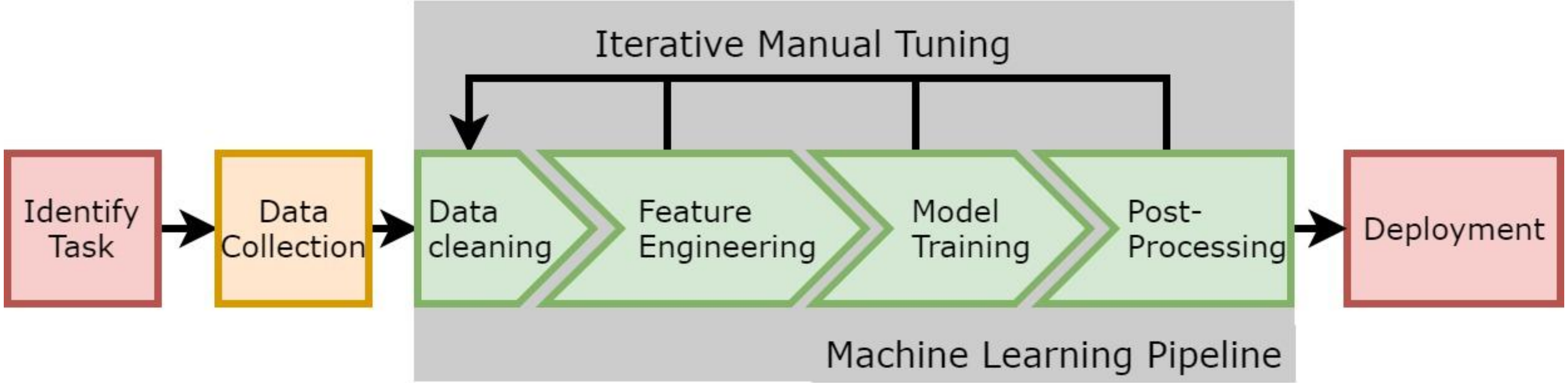
Decision Trees

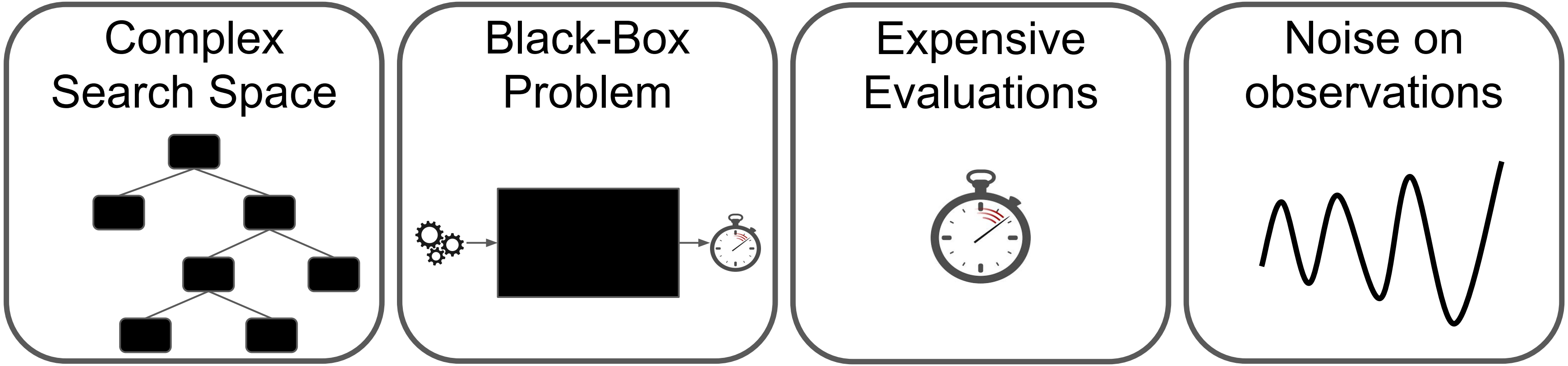
Random Forests

→ There are more than 100 classification algorithms!

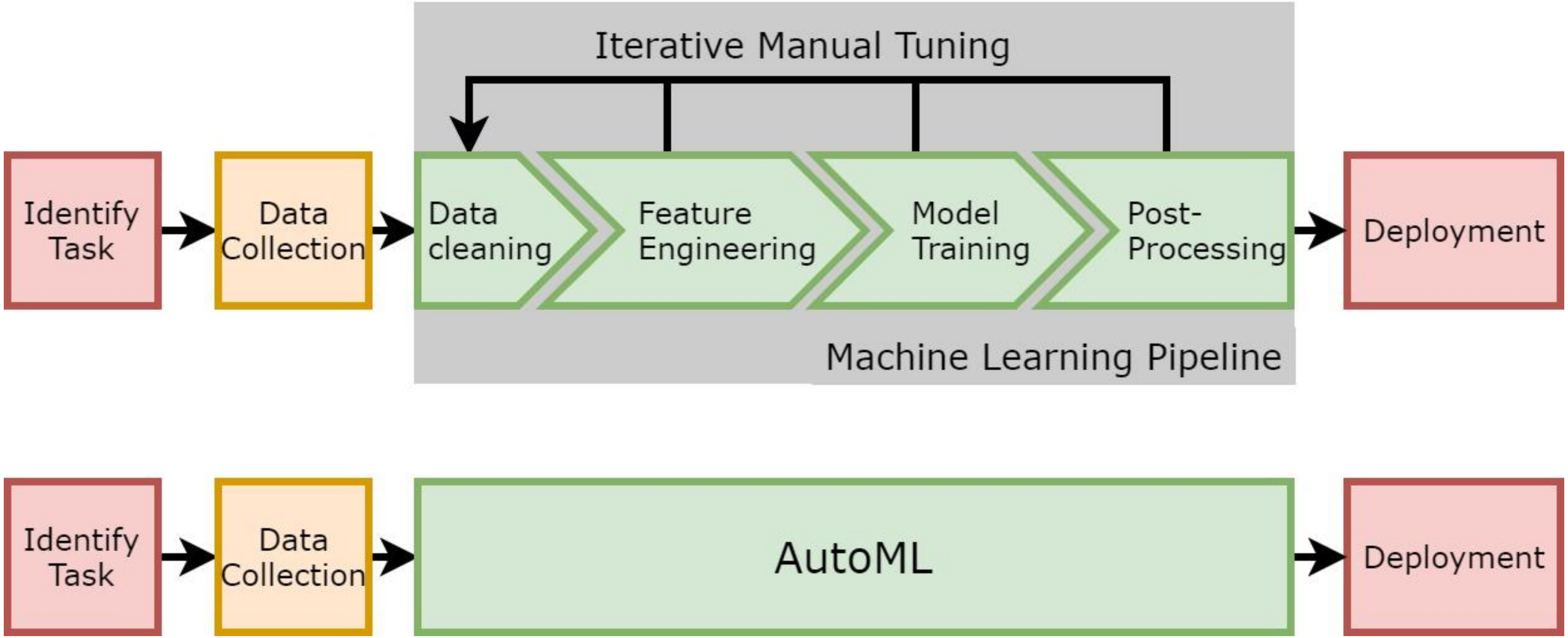
→ Each of these has 2-50 hyperparameters

Challenges in Designing ML Pipelines





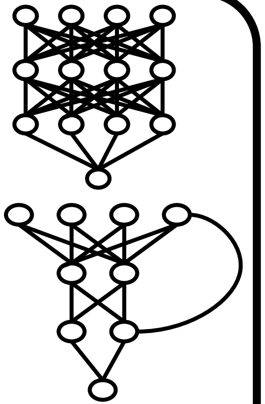
From Manual ML to Automated ML



Design Decisions taken care by AutoML

classifier	#λ
AdaBoost (AB)	4
Bernoulli naïve Bayes	2
decision tree (DT)	4
extreml. rand. trees	5
Gaussian naïve Bayes	-
gradient boosting (GB)	6
kNN	3
LDA	4
linear SVM	4
kernel SVM	7
multinomial naïve Bayes	2
passive aggressive	3
QDA	2
random forest (RF)	5
Linear Class. (SGD)	10

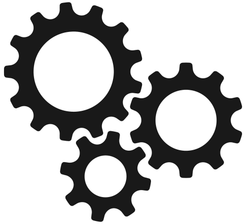
Algorithms



Architecture Design

preprocessor	#λ
extreml. rand. trees prepr.	5
fast ICA	4
feature agglomeration	4
kernel PCA	5
rand. kitchen sinks	2
linear SVM prepr.	3
no preprocessing	-
nystroem sampler	5
PCA	2
polynomial	3
random trees embed.	4
select percentile	2
select rates	3
one-hot encoding	2
imputation	1
balancing	1
rescaling	1

Pre-processing



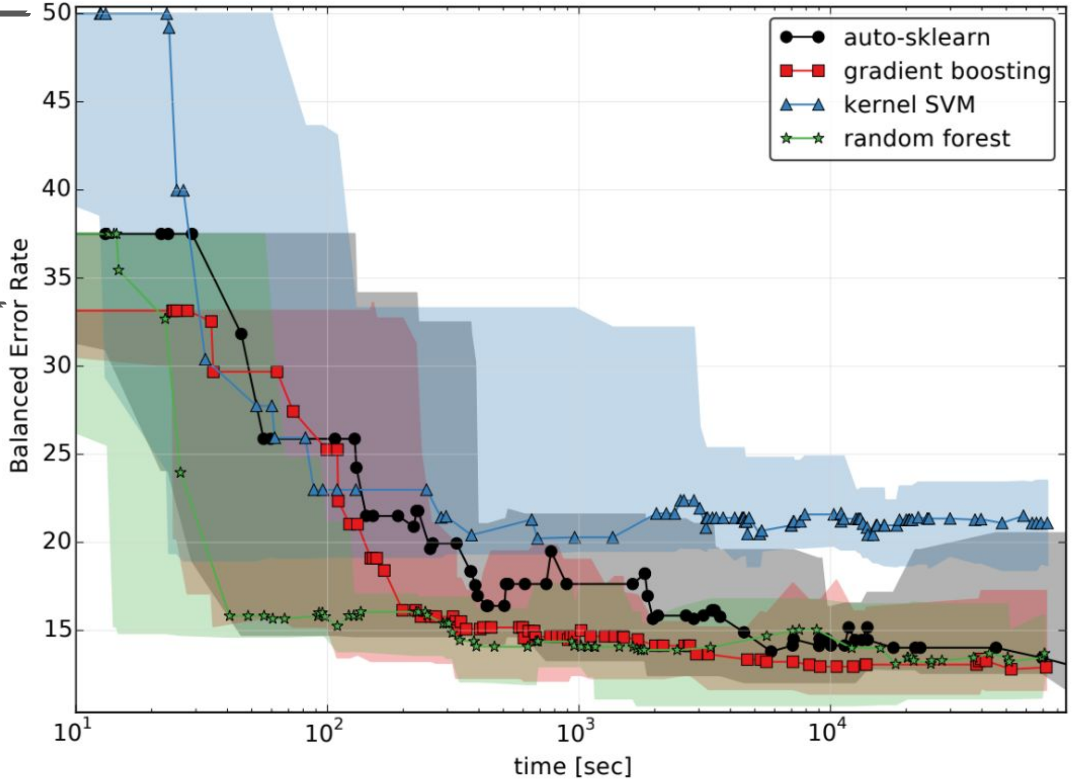
Hyper-parameters

...

Importance of Design Decisions in ML

(example on one specific dataset)

Choosing the correct algorithm → 17% improv.



Optimized hyperparameters → 20% - 29% improvement

Benefits of AutoX Methods

Domain	Default vs. Optimized
Answer Set Solving [Gebser et al. 2011]	up to $14\times$ speedup
AI Planning [Vallati et al. 2013]	up to $40\times$ speedup
Mixed Integer Programming [Hutter et al. 2010]	up to $52\times$ speedup
Satisfiability Solving [Hutter et al. 2017]	up to $3000\times$ speedup
Minimum Vertex Cover [Wagner et al. 2017]	up to 9% absolute impr.
Machine Learning [Feuer et al. 2015]	up to 35% absolute impr.
Deep Learning [Zimmer et al. 2020]	up to 49% absolute impr.

Topics

Bayesian Optimization

Multi-Objective Optimization

Gradient-based NAS

Hyperparameter Optimization

Neural Architecture Search

Reinforcement Learning

AutoML

Performance Predictions

Dynamic Algorithm Configuration

Explainability

Evolutionary Strategies

Meta Learning

Portfolio Construction

Transfer Learning

Bandit Algorithms

Success Stories in AutoML

International Challenge on AutoML I + II

Problem Setting

Constraints

- Time budget (only minutes)
- Memory constraint (few GB)
- Compute power (few CPUs)

Remarks

- More than 100 teams in first challenge
- 44 teams in second challenge
- Both AutoML and human teams

Results

#	User	Entries	Date of Last Entry	<Rank>
1	aad_freiburg	5	03/28/18	2.8000
2	narnars0	7	03/14/18	3.8000
3	Malik	2	03/15/18	5.4000
4	wlWangl	30	03/14/18	5.4000
5	thanhdng	4	03/19/18	5.4000
6	KaseyM	4	03/26/18	6.0000
7	vn.sju	17	03/26/18	6.2000
8	Joy	9	03/26/18	7.2000
9	hehe	4	03/21/18	8.0000
10	ObserverL	10	03/14/18	8.8000

Auto-Sklearn [Feurer et al. 2015, 2018, 2020]

Shape Error Prediction in Milling Processes

[Denkena et al. 2020]

Problem Setting

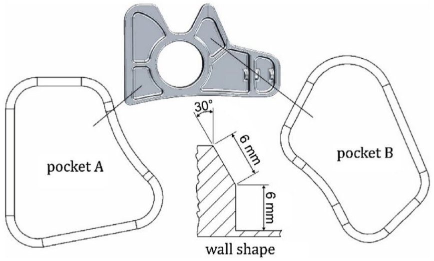
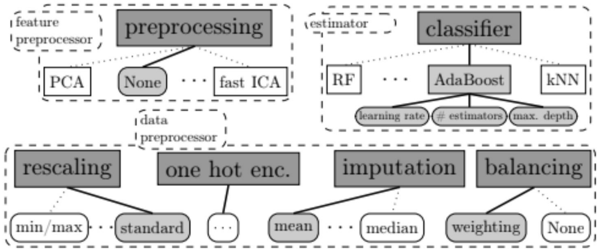


Image Source: Dittrich et al. 2018

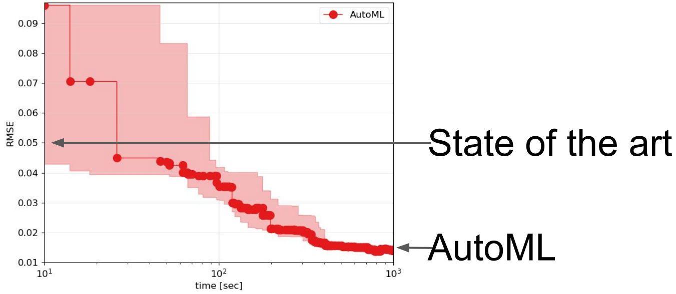
Search Space



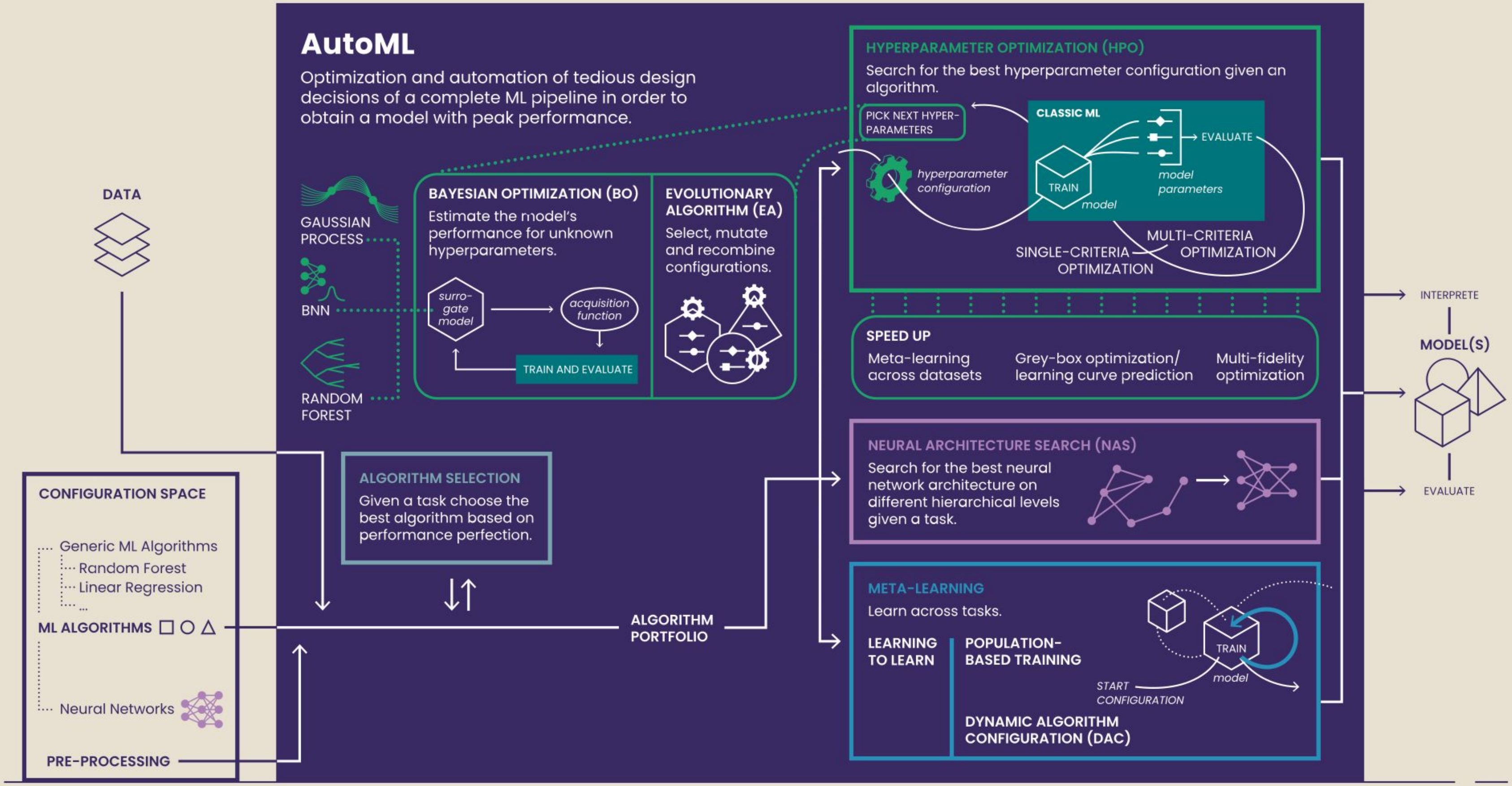
Remarks

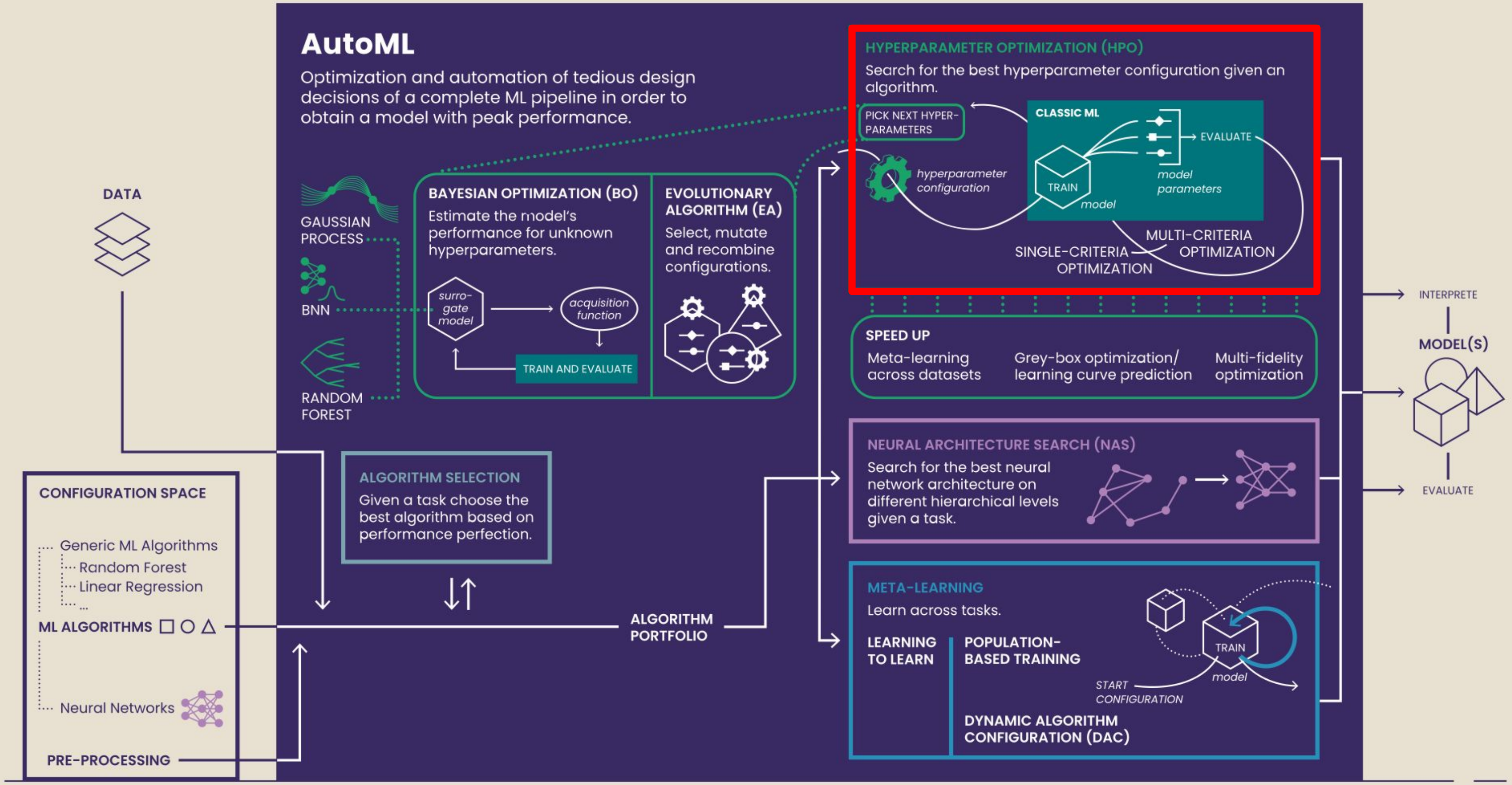
- Application of AutoML out-of-box
- Better results than Phd student of machining after spending substantial time
- Reading in the data format cost the most dev. time to let AutoML run

Results



Main Ideas


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-CRITERIA 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)
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS
Neural Networks
PRE-PROCESSING
ALGORITHM SELECTION
Given a task choose the best algorithm based on performance perfection.
ALGORITHM PORTFOLIO
INTERPRETE
MODEL(S)
EVALUATE

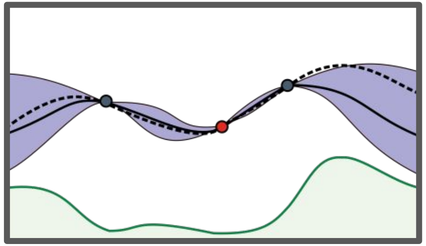


Leibniz Universität Hannover

AutoML.org

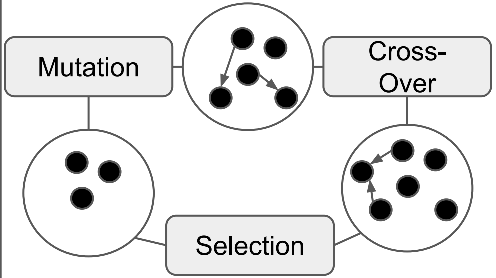
AutoML Optimizer

Bayesian Optimization



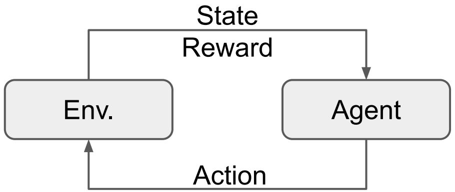
- Global optimization strategy
- Very sample efficient
- Very efficient for small/med. config. spaces

Evolutionary Algorithms

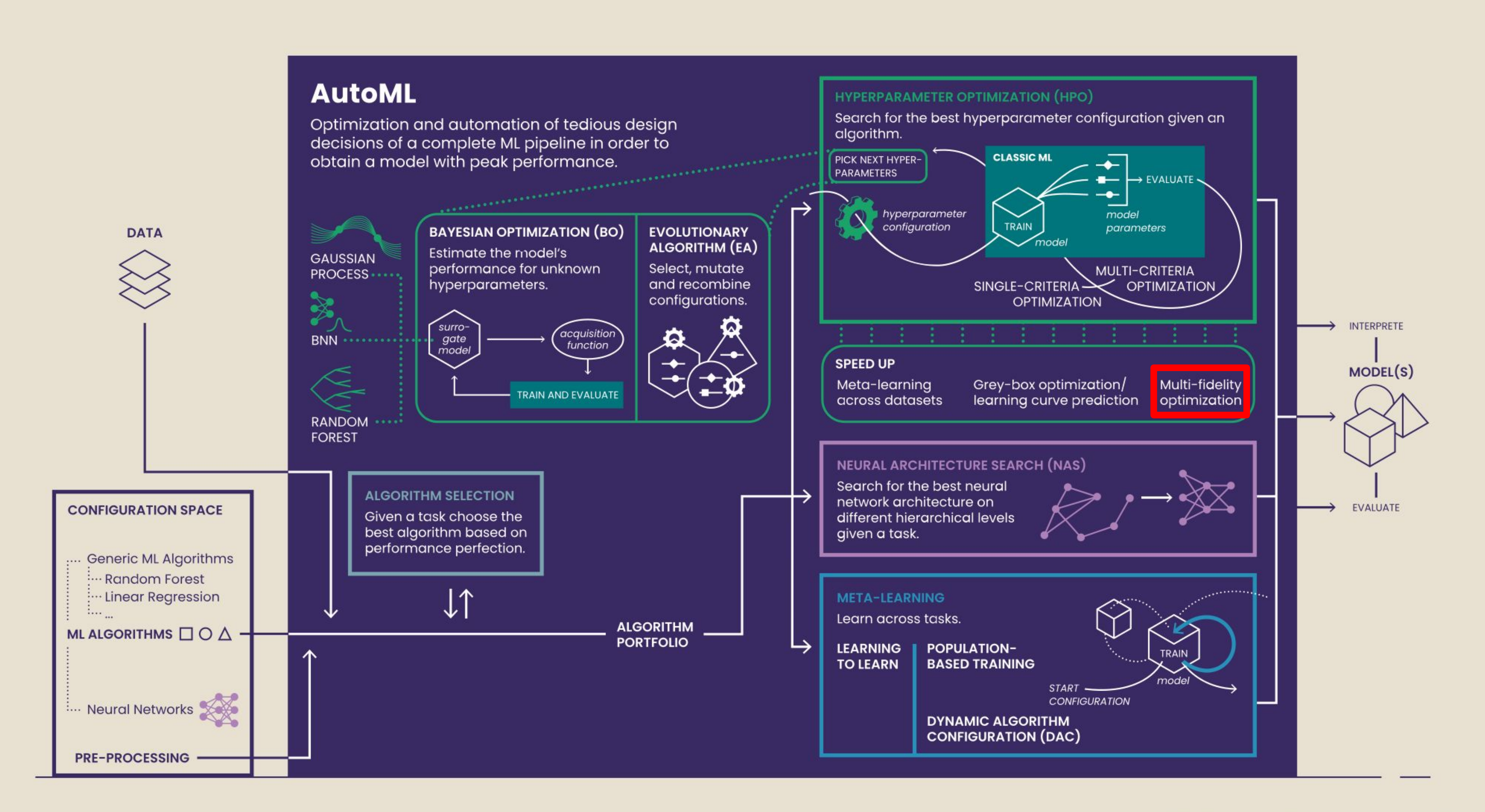


- Population based-approach
- Strong performance for longer budget
- Easy to parallelize

Reinforcement Learning



- Learning of a policy
- Can learn a generalizable policy
- Human-like approach


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-CRITERIA OPTIMIZATION
SPEED UP
Meta-learning across datasets
Grey-box optimization/ learning curve prediction
Multi-fidelity optimization
INTERPRETE
MODEL(S)
EVALUATE
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS □ ○ △
Neural Networks
PRE-PROCESSING
ALGORITHM SELECTION
Given a task choose the best algorithm based on performance perfection.
ALGORITHM PORTFOLIO
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)

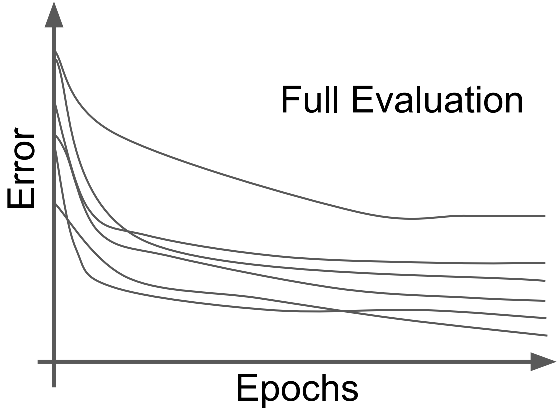
AutoML: Model Selection

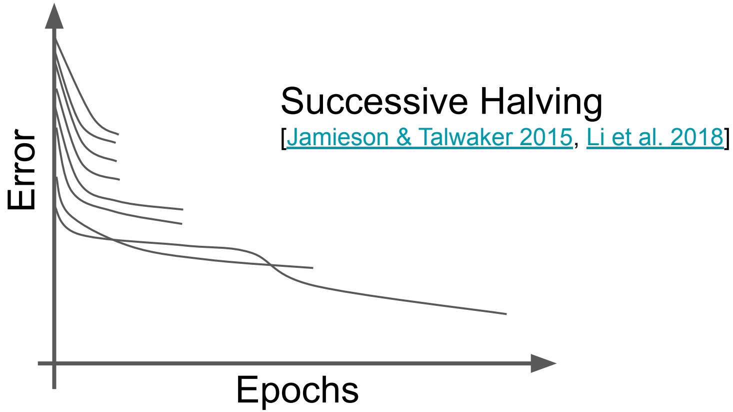
Hold-Out

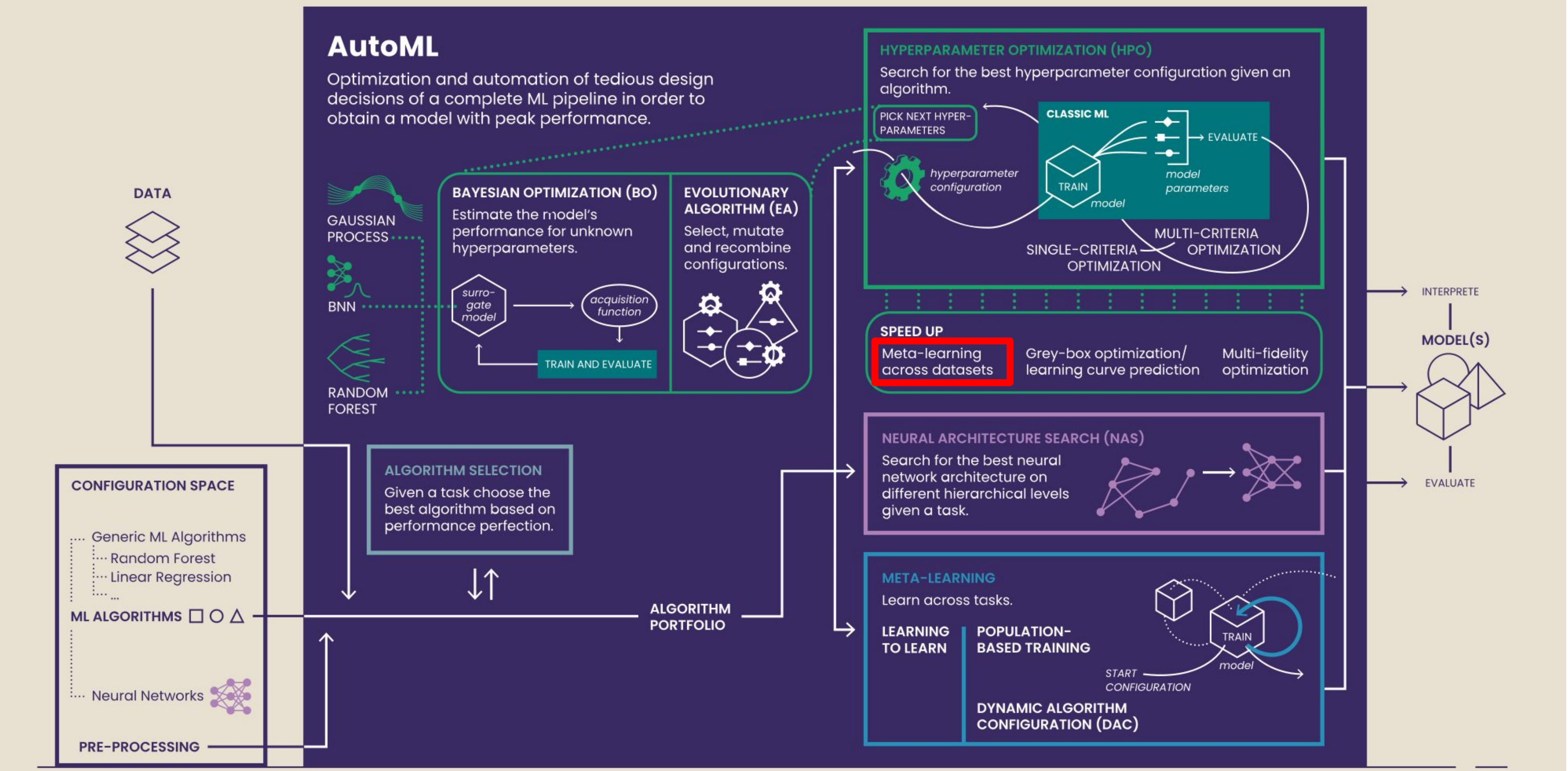
Training	Validation

Cross Validation

Fold 1	Fold 2	Fold 3	Fold 4



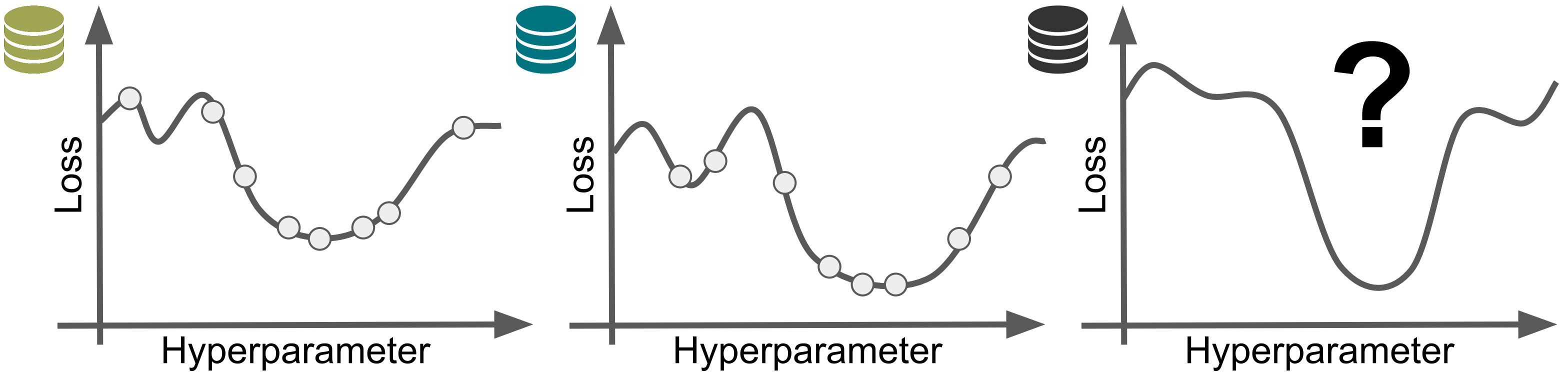




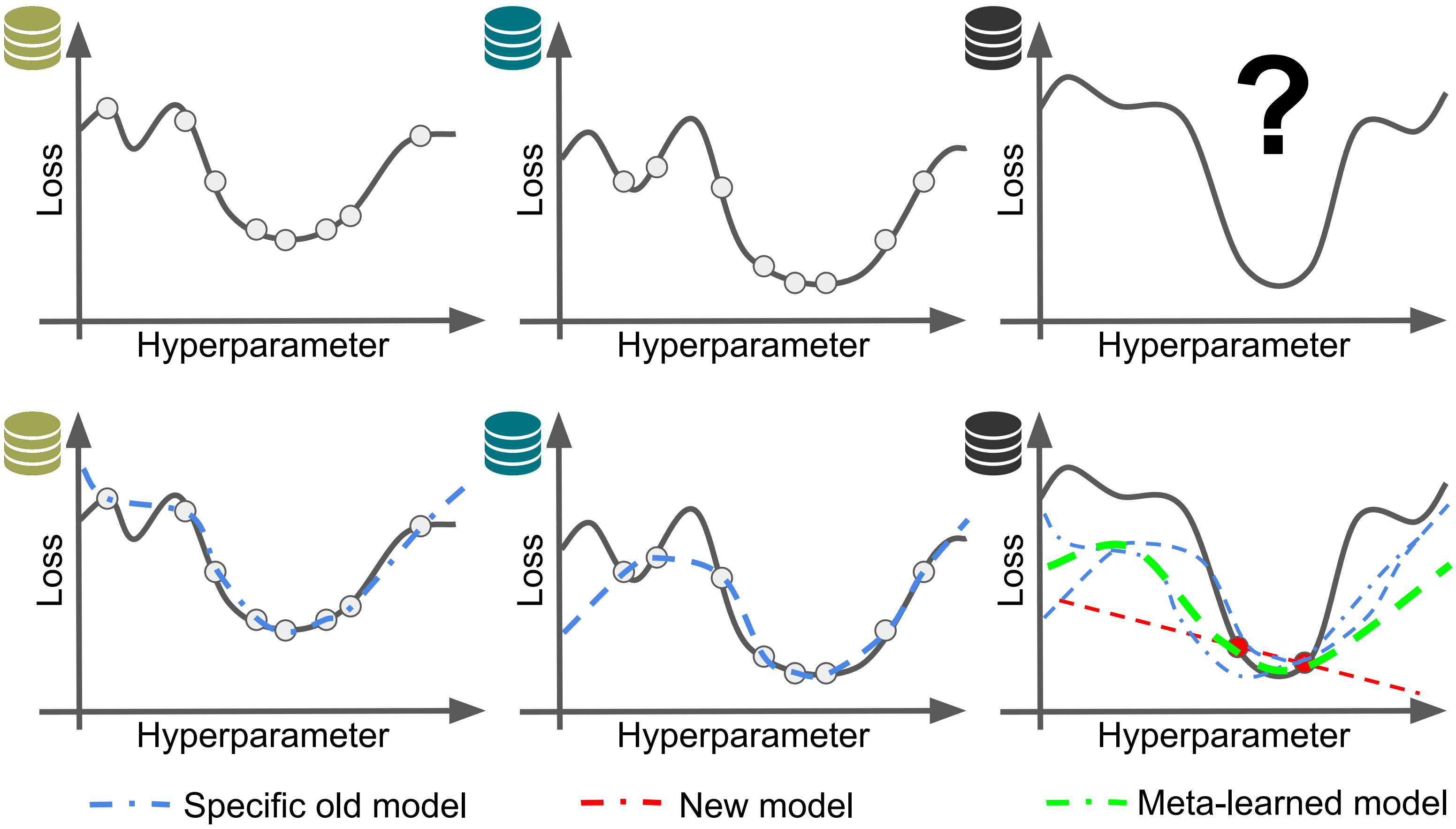
AutoML
Optimization and automation of tedious design decisions of a complete ML pipeline in order to obtain a model with peak performance.
DATA
GAUSSIAN PROCESS
BNN
RANDOM FOREST
BAYESIAN OPTIMIZATION (BO)
Estimate the model's performance for unknown hyperparameters.
surro-gate model
acquisition function
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-CRITERIA 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)
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS □ ○ △
Neural Networks
PRE-PROCESSING
ALGORITHM SELECTION
Given a task choose the best algorithm based on performance perfection.
ALGORITHM PORTFOLIO
INTERPRETE
MODEL(S)
EVALUATE

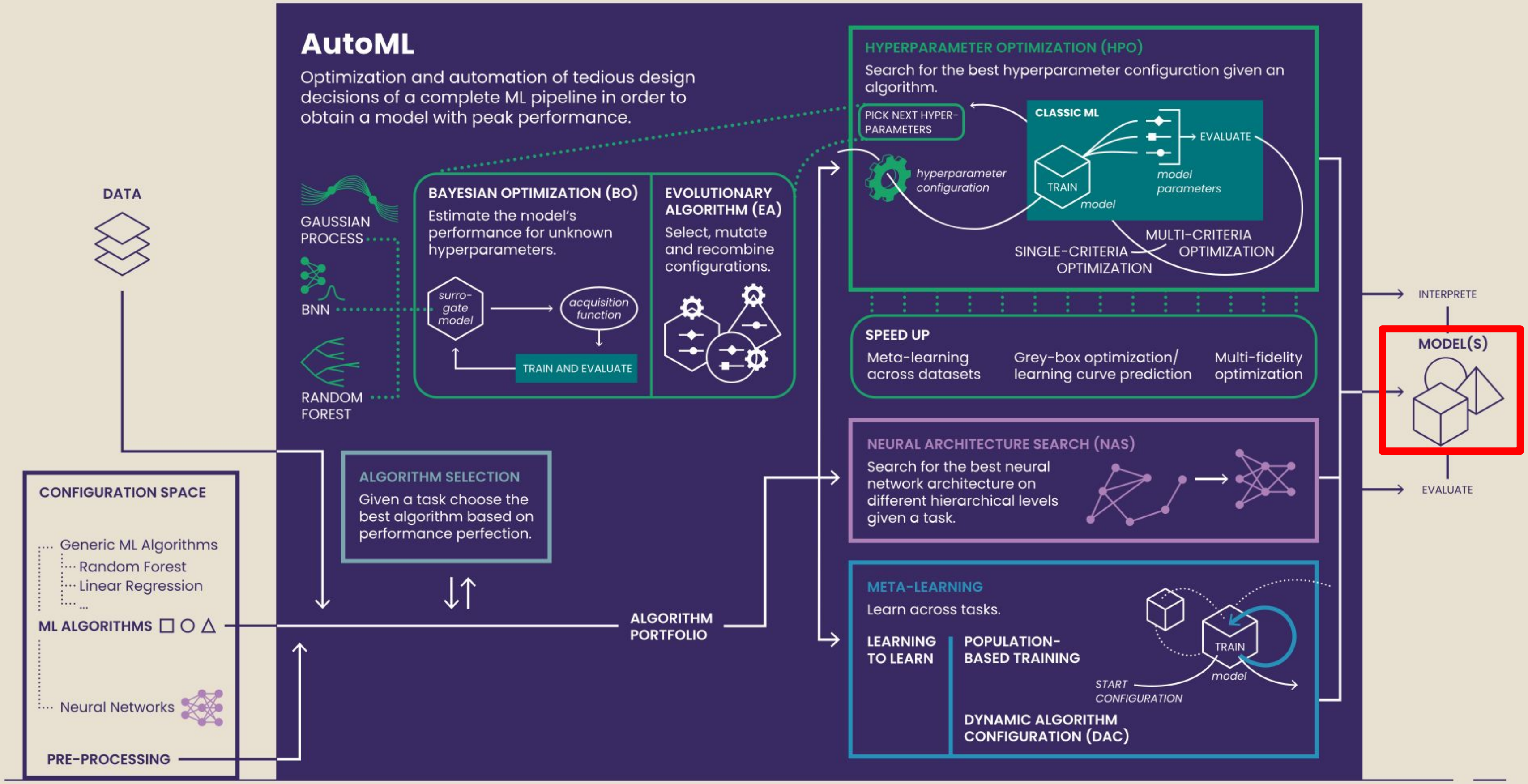
Warmstarting via Meta-Learning

[Feurer et al. 2015, Lindauer & Hutter. 2017]



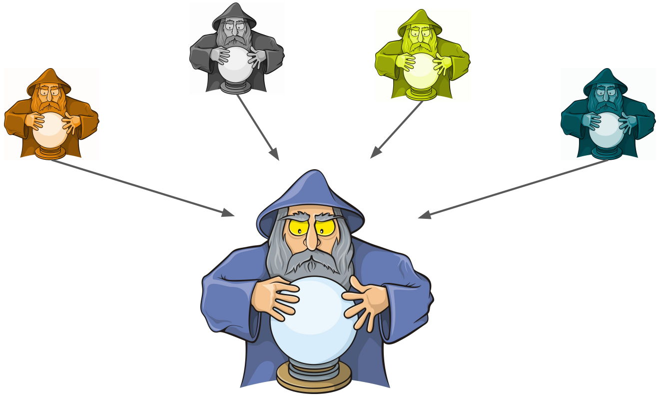
Warmstarting via Meta-Learning [Lindauer & Hutter. 2017]





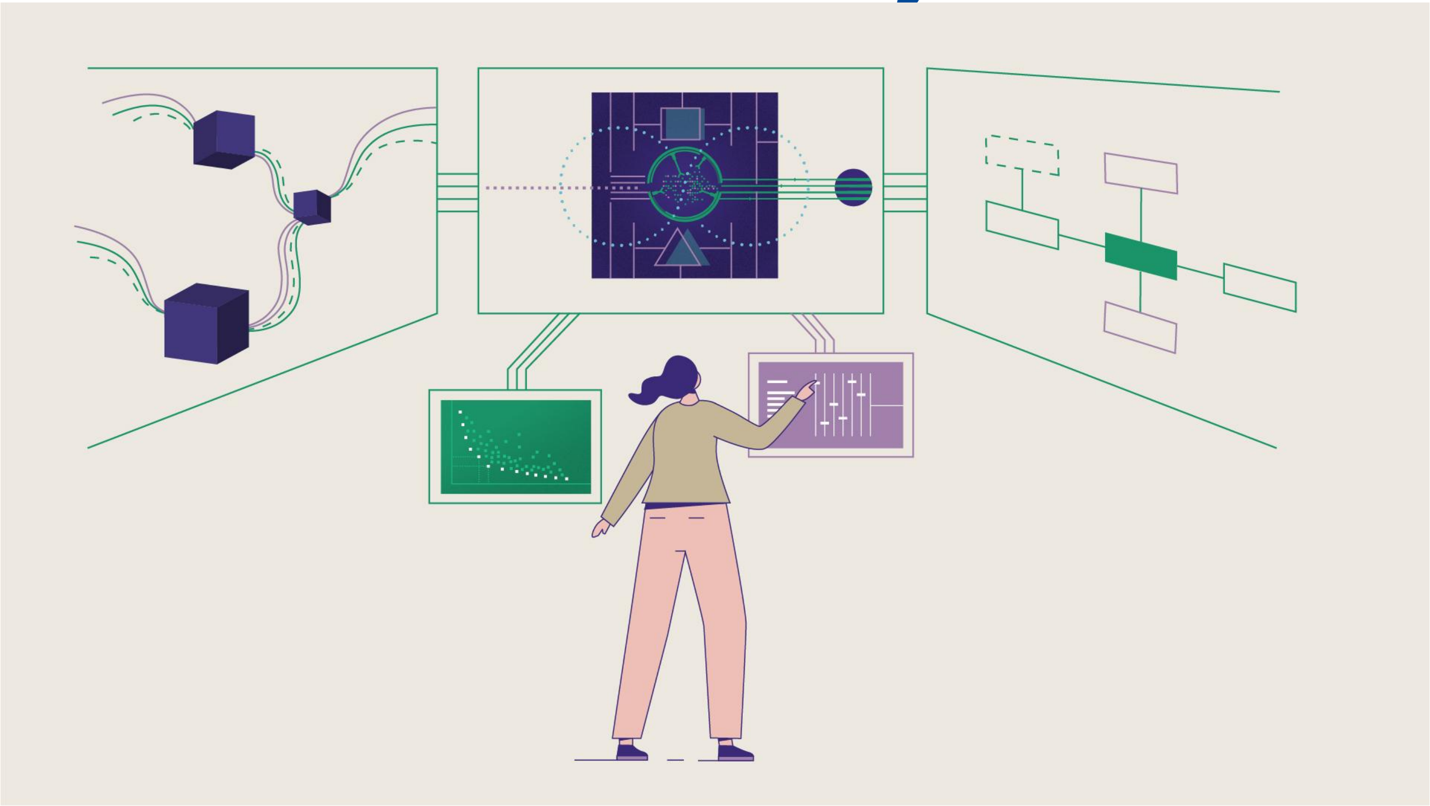
AutoML
Optimization and automation of tedious design decisions of a complete ML pipeline in order to obtain a model with peak performance.
DATA
GAUSSIAN PROCESS
BNN
RANDOM FOREST
BAYESIAN OPTIMIZATION (BO)
Estimate the model's performance for unknown hyperparameters.
surro-gate model
acquisition function
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-CRITERIA 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
START CONFIGURATION
model
DYNAMIC ALGORITHM CONFIGURATION (DAC)
ALGORITHM SELECTION
Given a task choose the best algorithm based on performance perfection.
ALGORITHM PORTFOLIO
CONFIGURATION SPACE
Generic ML Algorithms
Random Forest
Linear Regression
...
ML ALGORITHMS
Neural Networks
PRE-PROCESSING
INTERPRETE
MODEL(S)
EVALUATE

AutoML Ensembles



- If ensemble members make uncorrelated errors
- Already a diverse set of ensemble members can perform well
- AutoML can help to find even better ensembles

Open Source Software Projects



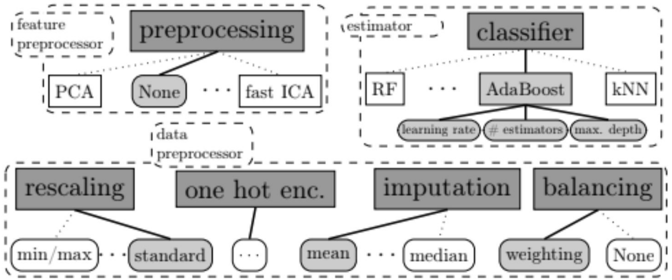
Leibniz Universität Hannover

AutoML.org

Auto-Sklearn [Feuerer et al. 2015, 2018, 2020]

Star 4.8k Fork 912

Takes care of well-performing ML-pipeline



Easy-to-use

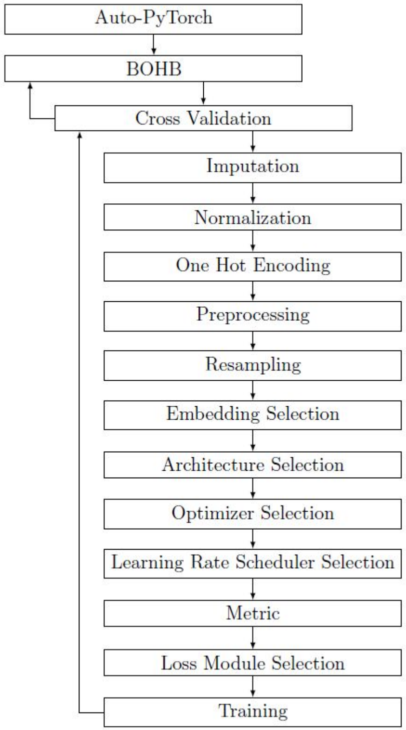
```
import autosklearn.classification as cls
automl = cls.AutoSklearnClassifier()
automl.fit(X_train, y_train)
y_hat = automl.predict(X_test)
```

- Winner of 1st and 2nd AutoML Challenge
- Improved efficiency in Version 2.0 by
  - Meta-learning, multi-fidelity optimization, automating AutoML

Leibniz Universität Hannover
AutoML.org

Auto-PyTorch [Mendoza et al. 2019, Zimmer et al. 2021]

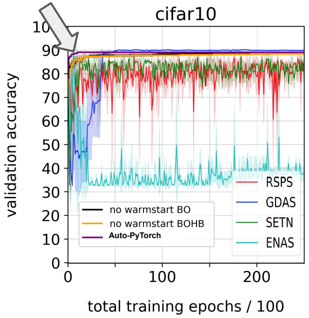
Star 1,164 Fork 144




```
autoPyTorch = AutoNetClassification("tiny_cs",  # config preset
                                    log_level='info',
                                    max_runtime=300,
                                    min_budget=30,
                                    max_budget=90)

autoPyTorch.fit(X_train, y_train, validation_split=0.3)
y_pred = autoPyTorch.predict(X_test)
```

- Strong performance against other state-of-the AutoML tools on tabular data
- Even competitive on image data against gradient-based methods
- Efficiency due to meta-learning, multi-fidelity optimization and ensembling



SMAC [Hutter et al. 2011, Lindauer et al. 2017]

Star 509 Fork 146

```
x, cost, _ = fmin_smac(func=branin,  # function
                       x0=[0, 0],    # default configuration
                       bounds=[(-5, 10), (0, 15)],  # limits
                       maxfun=10,   # maximum number of evaluations
                       rng=3)        # random seed
print("Optimum at {} with cost of {}".format(x, cost))
```

- Working horse for Auto-Sklearn
  - Soon also for Auto-PyTorch

- Implements state-of-the-art approaches for
  - Bayesian optimization
  - Multi-fidelity optimization
    - E.g., successive halving, hyperband, BOHB
  - Algorithm configuration
    - Robust configurations across many tasks

Leibniz Universität Hannover

AutoML.org

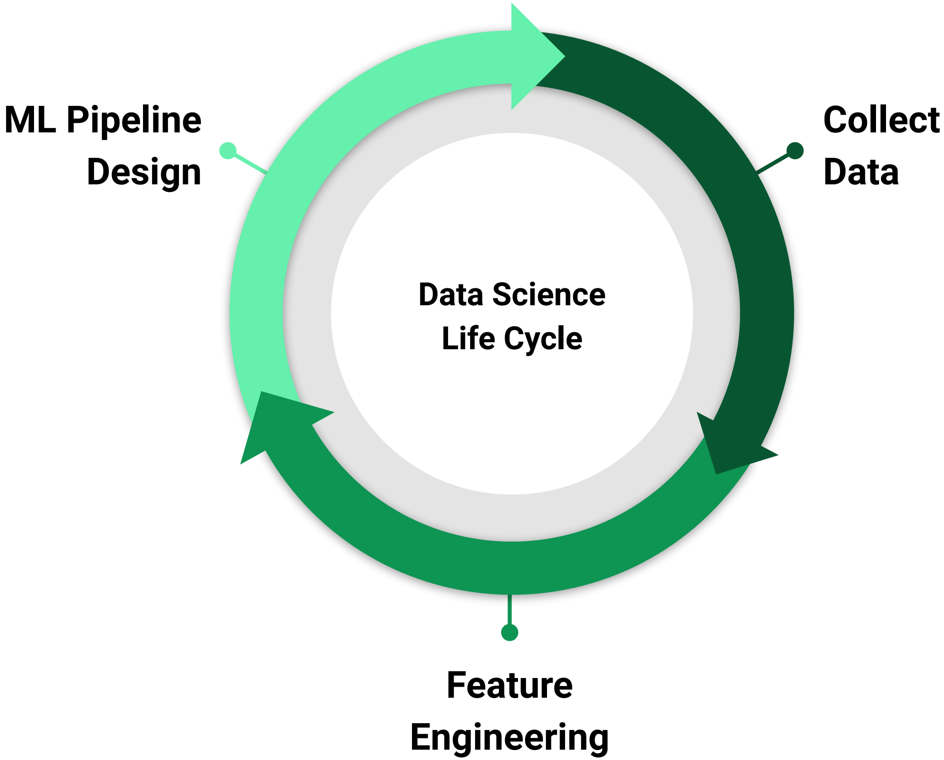
Pros and Cons of AutoML

Pros and Cons

- Saves human developer time
- Costs compute time

AutoML vs. Expert Knowledge

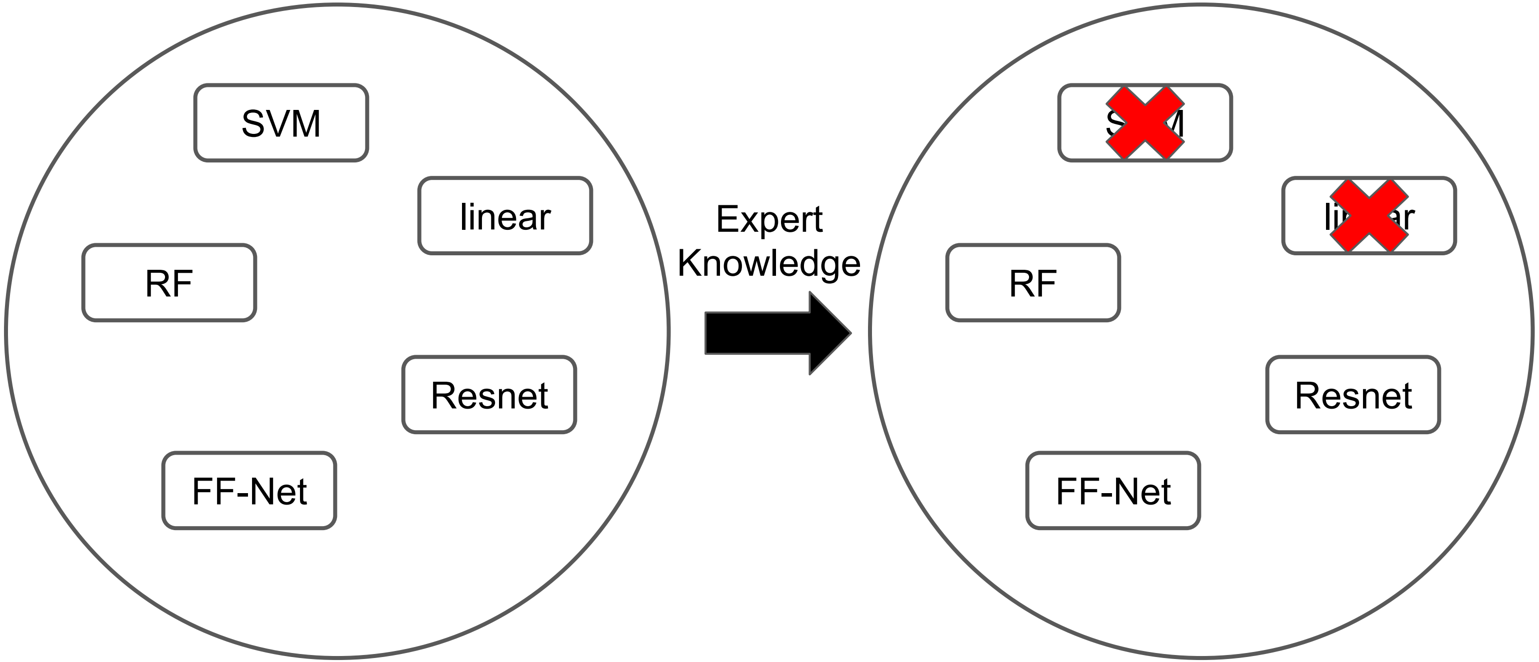
Use the Gained Time for Feature Engineering



- All of these steps are important
- Often iterative cycle between these
- If one of these gets automated, more time for others available

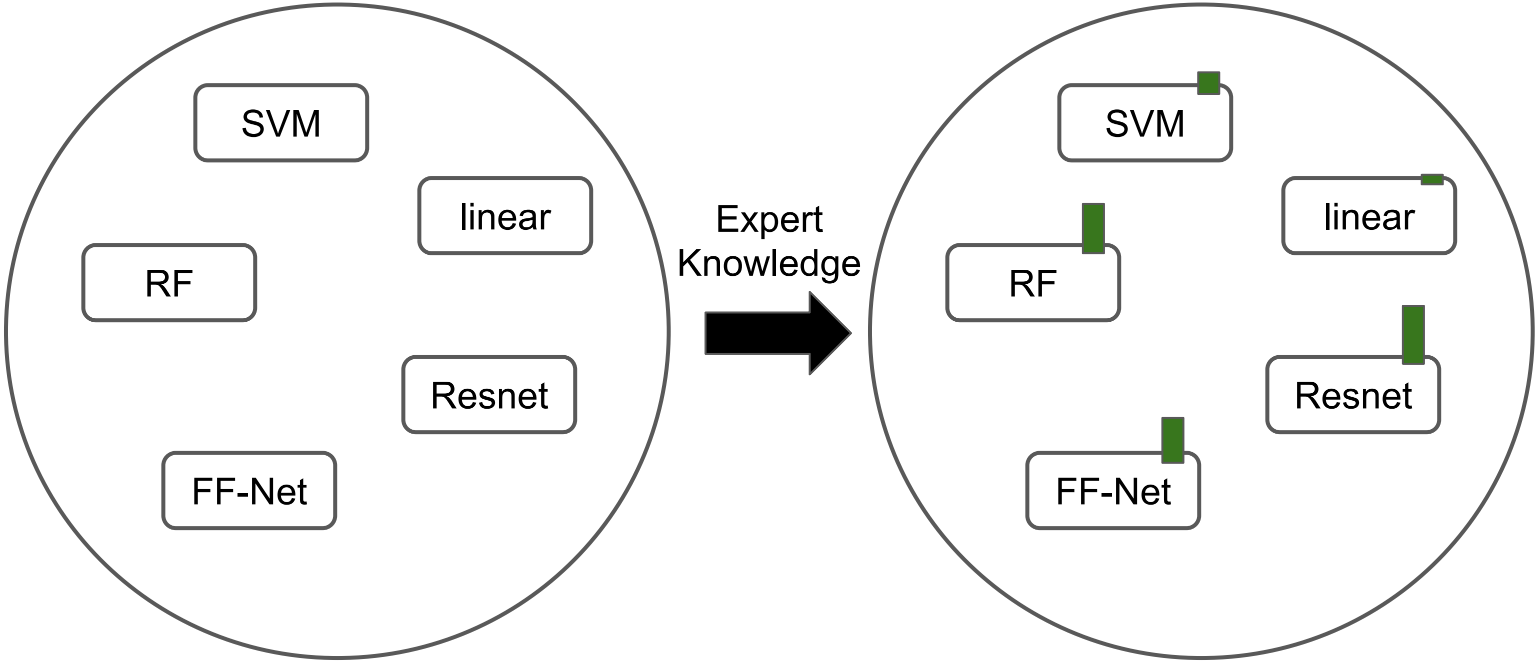
- Feature engineering is often one of the best places to consider expert knowledge
  - Nevertheless, it can also be automated

Cut Down the Search Space



Expert Knowledge as Probability Distributions

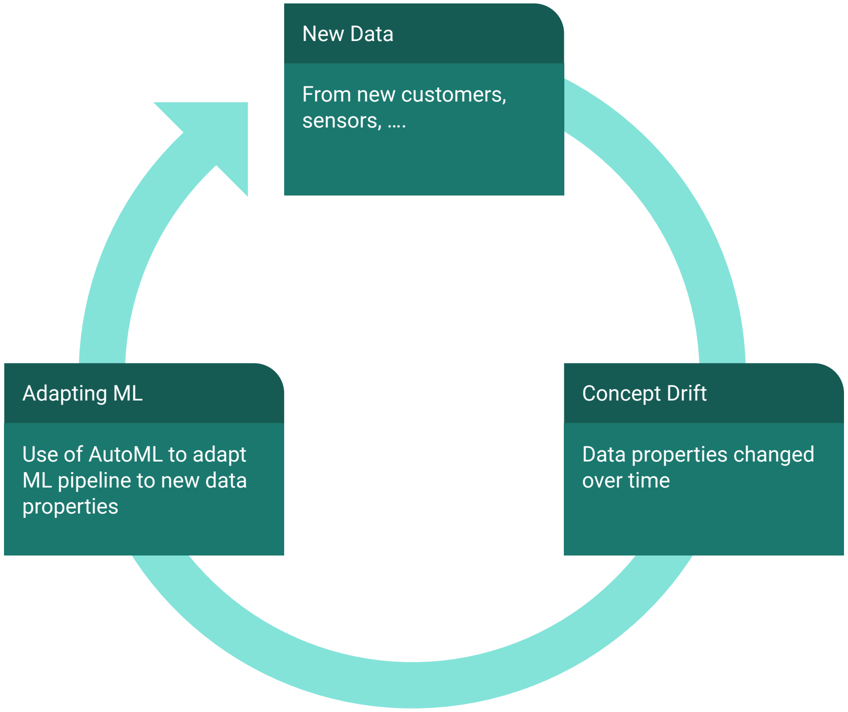
[Luis et al. 2020]



→ Increases efficiency of AutoML

What's next?

AutoML + Meta-Learning

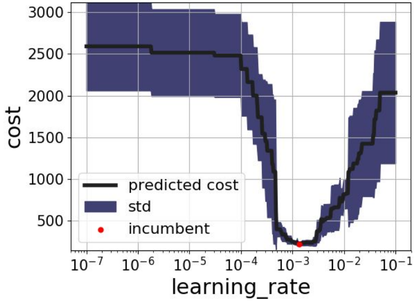


- Amount of data is continuously increasing
  → Models have to be updated or even trained from scratch if a concept drift occurs
  → ML pipeline needs adjustments

- AutoML could help to enable maintainability also in the long run [Celik & Vanschoren 2020]
- Similarly, AutoML can help even if the underlying ML algorithm changes [Stoll et al. 2020]

xAutoML: Explainable AutoML
[Biedenkapp et al. 2017, 2018, 2019, Moosbauer et al. 2021]

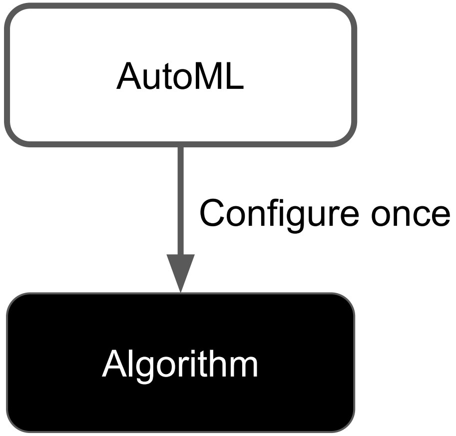
- Users want to know more than the result
  For example:
  - Which design decisions were important?
  - Why was the returned pipeline chosen?
  - Was the approach of the AutoML tool appropriate for the dataset at hand?
  - …
- **What would be interesting for you?**



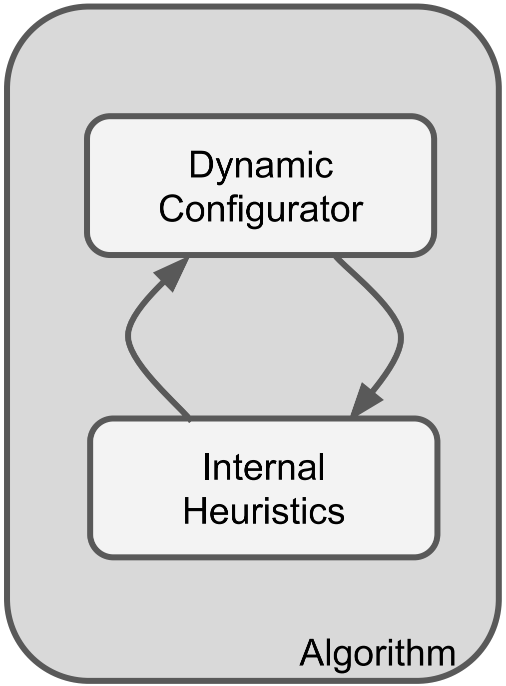
	fANOVA	LPI
discount	19.32	38.88
learning rate	3.70	35.4
batch size	15.77	21.5
# units 1	1.86	0.07
# units 2	0.39	0.01

Dynamic Algorithm Control via RL

[Biedenkapp et al. 2019 + 2020, Shala et al. 2020, Speck et al. 2021]



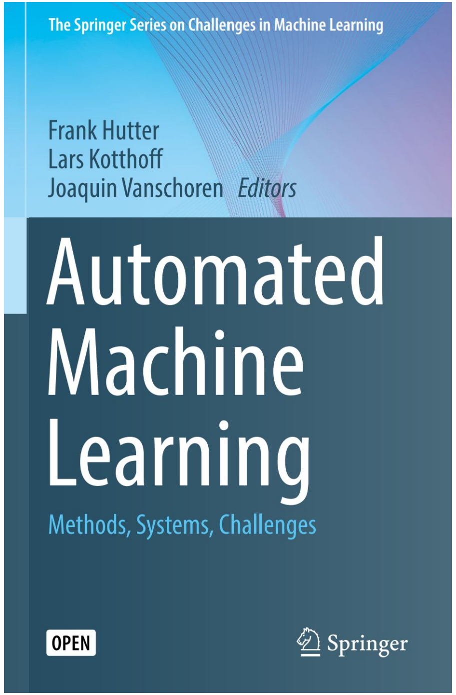
Traditional AutoML:
Blackbox-Optimization



Future: Reinforcement learning to learn to adjust hyperparameters over time

Learning more about AutoML?

Further Material



KI-Campus BETA

AutoML online course starting April 2021

Leibniz Universität Hannover

AutoML.org

Our Vision: Democratization of AI

1. We need tools s.t. AI is **easy-to-use**

2. **Efficient development** of new AI applications

3. AutoML will leverage **interdisciplinary applications**: ML + ?

4. **Improved understanding** of AI systems

Leibniz Universität Hannover

AutoML.org

@LindauerMarius
@AutoML_org

Thank you!