

# AutoML: Interpretability

Overview: Automated Empirical Analysis

Bernd Bischl   Frank Hutter   Lars Kotthoff  
Marius Lindauer   Joaquin Vanschoren

- Big challenge of ML: Interpretability
  - ▶ In some applications, it is required to "understand" a prediction
  - ▶ Users have less trust in systems, they can't understand

- Big challenge of ML: Interpretability
  - ▶ In some applications, it is required to "understand" a prediction
  - ▶ Users have less trust in systems, they can't understand
- AutoML is even worse?
  - ▶ AutoML is a black-box that automates the design of another blackbox (ML)
  - ▶ Also ML-developers have an basic understanding of the design of their ML pipelines
- Automated empirical interpretability helps to
  - ▶ understand the finally returned ML system
  - ▶ understand the AutoML process

# Approach

- Insights:

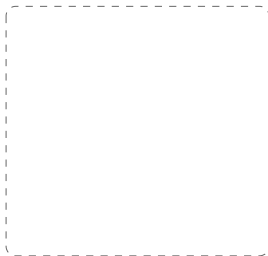
- ▶ AutoML is yet another optimization problem
- ▶ (Most) AutoML approach are iterative in nature

~> AutoML generates a lot of empirical data

Cost  $c$

Budgets

Design Space  $\Lambda$

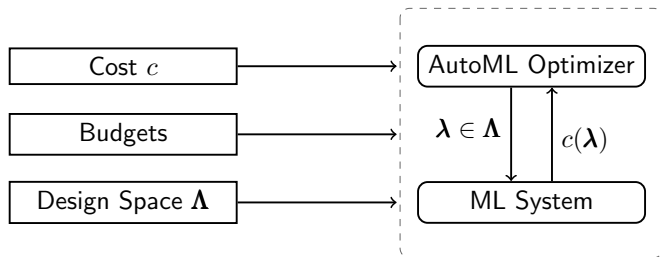


# Approach

- Insights:

- ▶ AutoML is yet another optimization problem
- ▶ (Most) AutoML approach are iterative in nature

~> AutoML generates a lot of empirical data

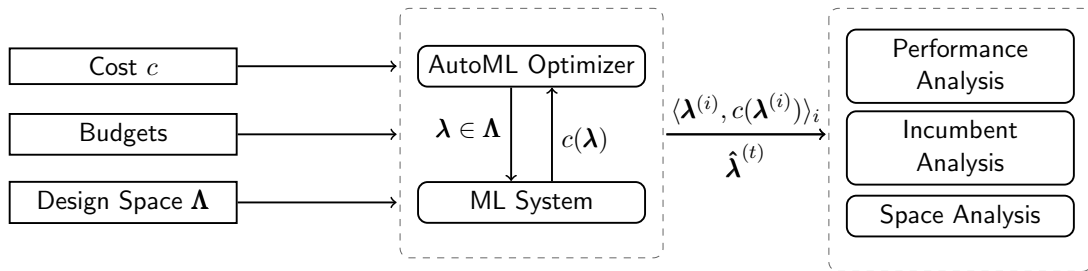


# Approach

- Insights:

- ▶ AutoML is yet another optimization problem
- ▶ (Most) AutoML approach are iterative in nature

~> AutoML generates a lot of empirical data

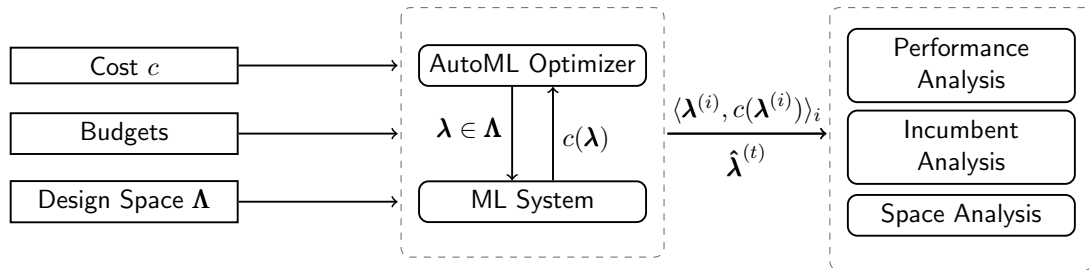


# Approach

- Insights:

- ▶ AutoML is yet another optimization problem
- ▶ (Most) AutoML approach are iterative in nature

~> AutoML generates a lot of empirical data



~> Let's use this data to learn something about our AutoML problem

# Basic Examples

- Visualize final incumbent  $\hat{\lambda}$ 
  - ▶ ML pipeline with its components
  - ▶ Neural architecture



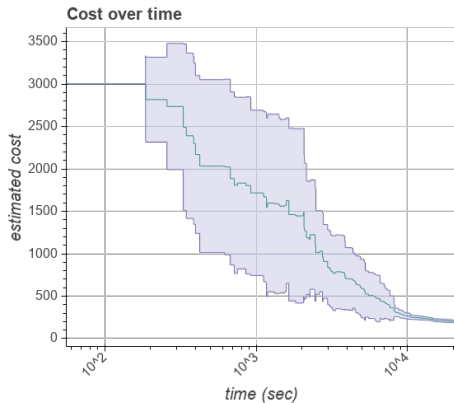
# Basic Examples

- Visualize final incumbent  $\hat{\lambda}$ 
  - ▶ ML pipeline with its components
  - ▶ Neural architecture
- Compare what changed between  $\lambda_{\text{def}}$  and  $\hat{\lambda}$

# Basic Examples

- Visualize final incumbent  $\hat{\lambda}$ 
  - ▶ ML pipeline with its components
  - ▶ Neural architecture
- Compare what changed between  $\lambda_{\text{def}}$  and  $\hat{\lambda}$
- Show  $\hat{\lambda}$  on different budgets (if you used a multi-fidelity approach)

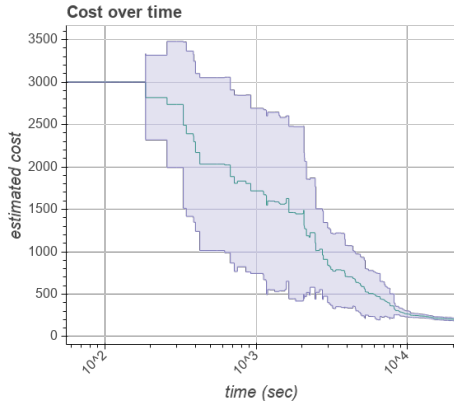
# Cost Over Time



- Study how your AutoML tool improves cost (or loss) over time

Source: [Lindauer et al. 2019]

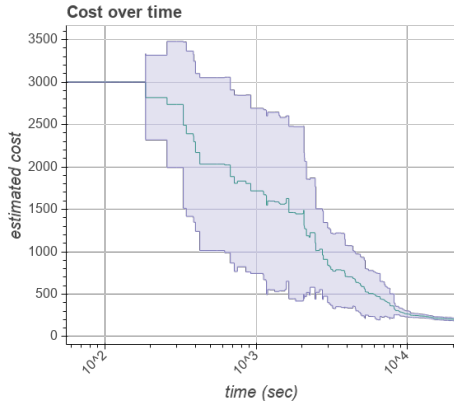
# Cost Over Time



Source: [Lindauer et al. 2019]

- Study how your AutoML tool improves cost (or loss) over time
- Allows to identify whether
  - ▶ you need less time next time or

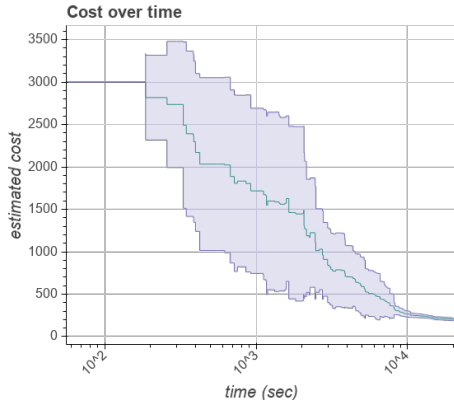
# Cost Over Time



Source: [Lindauer et al. 2019]

- Study how your AutoML tool improves cost (or loss) over time
- Allows to identify whether
  - ▶ you need less time next time or
  - ▶ the AutoML system is still improving; so you should give it more time

# Cost Over Time



Source: [Lindauer et al. 2019]

- Study how your AutoML tool improves cost (or loss) over time
- Allows to identify whether
  - ▶ you need less time next time or
  - ▶ the AutoML system is still improving; so you should give it more time
- Notes:
  - ▶ Plot on log-scale to see details in the beginning
  - ▶ If you done several runs, plot distribution (e.g., median and 25/75%-quartiles)