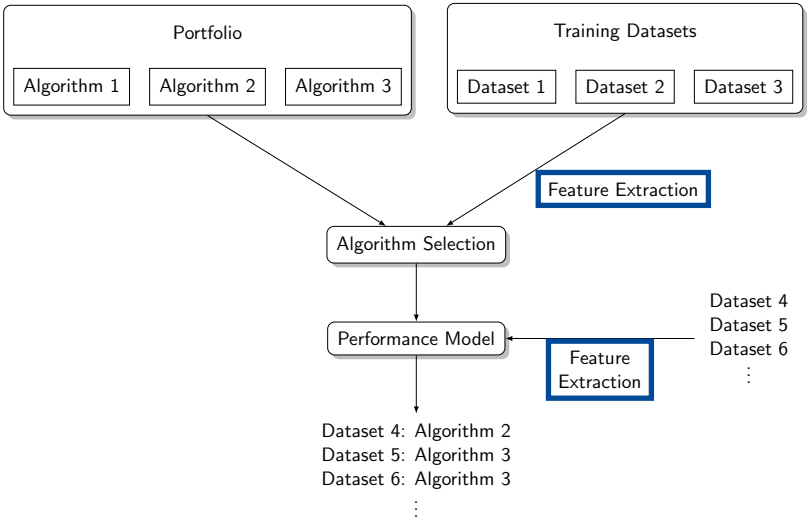
AutoML: Algorithm Selection

Features

Bernd Bischl Frank Hutter Lars Kotthoff
Marius Lindauer Joaquin Vanschoren

Algorithm Selection



Features

- relate properties of datasets to algorithm performance
- relatively cheap to compute – must be cheaper than running the algorithm to see what its performance is
- often specified by domain expert
- syntactic and information-theoretic – analyze dataset
- probing – run an algorithm for short time or on subset of data

Syntactic and Information-Theoretic Features

- number of binary/numeric/categorical features
- number of classes
- class entropy
- skewness of classes
- fraction of missing values
- correlation between features and target
- ...

Probing Features (Landmarkers)

- performance of majority class/mean value predictor
- decision stump performance
- simple rule model performance
- performance of algorithm of interest on 1% of data
- ...

$\rightarrow$ usually leads to much better results that using just syntactic and information-theoretic features

No Features

- use deep learning to process dataset or problem instance as-is
- no need for expert-designed features
- only preliminary applications so far, performance not good, no widespread adoption yet

Aside: Algorithm Features

- can characterize algorithm in addition to datasets
- allows to relate performance to specific aspects of an algorithm rather than black boxes
- for example size of code base, properties of abstract syntax tree...
- ongoing work

What Features Do We Need in Practice?

- trade-off between complex features and complex models
- in practice, very simple features can perform well
- often only few features of a set are needed (e.g. 5 out of >100)
- in the end, whatever works best