Using Meta Learning to Initialize Bayesian Optimization

Albert-Ludwigs-Universität Freiburg

Matthias Feurer[1] Jost Tobias Springenberg[2] Frank Hutter[1]

[1]Research Group on Learning, Optimization, and Automated Algorithm Design
[2]Machine Learning Lab
Department of Computer Science, University of Freiburg, Germany

ECAI-2014 Workshop on Meta-learning & Algorithm Selection, 19 August 2014

Your task: Build an Iris classification system

Your task: Build an Iris classification system

- Choose an algorithm based on dataset characteristics, e.g. for the Iris dataset this could be an SVM

The iris pictures on this slide are from wikimedia commons and used under the following licenses: Top left: Iris Versicolor is public domain; Bottom left: Iris setosa is licensed by Radomil under CC BY-SA 3.0; Top right: Iris Virginica is licensed by C T Johansson under CC BY 3.0.

Your task: Build an Iris classification system

- Choose an algorithm based on dataset characteristics, e.g. for the Iris dataset this could be an SVM
- Manual tuning -> fiddling with hyperparameters.

Your task: Build an Iris classification system

- Choose an algorithm based on dataset characteristics, e.g. for the Iris dataset this could be an SVM
- Manual tuning -> fiddling with hyperparameters.
- Better: Use automated methods like PSO, GA or SMBO

Your task: Build an Iris classification system

- Choose an algorithm based on dataset characteristics, e.g. for the Iris dataset this could be an SVM
- Manual tuning -> fiddling with hyperparameters.
- Better: Use automated methods like PSO, GA or SMBO
- Best: AutoWeka

Adding the Iris Japonica to the dataset

Adding the Iris Japonica to the dataset

- Manual tuning:
  Use experience and start from the parameters found on the Iris dataset

Adding the Iris Japonica to the dataset

- Manual tuning:
  Use experience and start from the parameters found on the Iris dataset
- Automated methods
  -> start from scratch

Adding the Iris Japonica to the dataset

- Manual tuning:
  Use experience and start from the parameters found on the Iris dataset
- Automated methods
  -> start from scratch
- $\rightarrow$ Cast *use experience* into an algorithm.

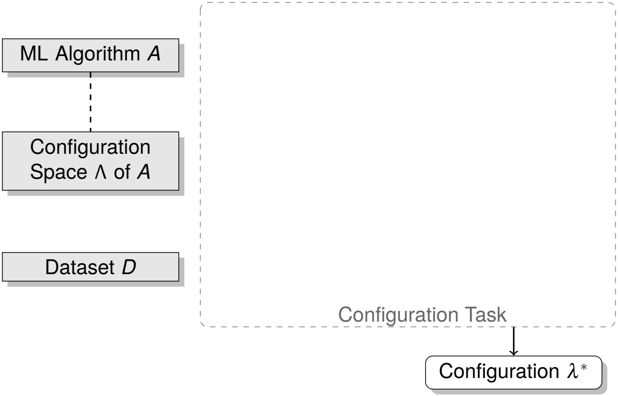
Sequential Model-based Bayesian Optimization (SMBO)

ML Algorithm A

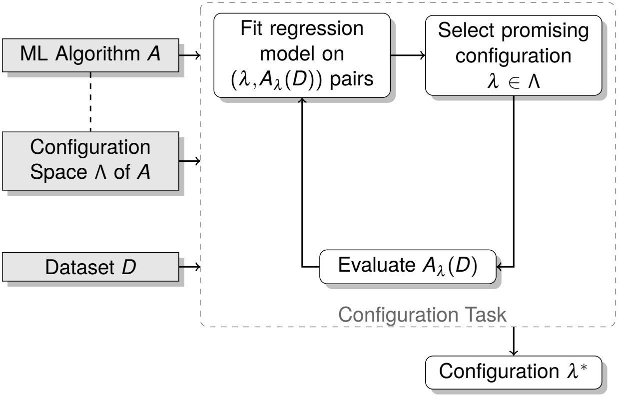
Configuration Space Λ of A

Dataset D

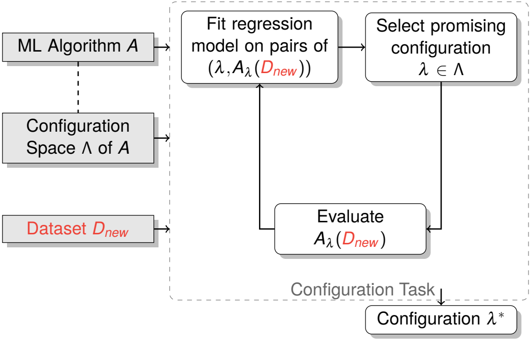
Sequential Model-based Bayesian Optimization (SMBO)



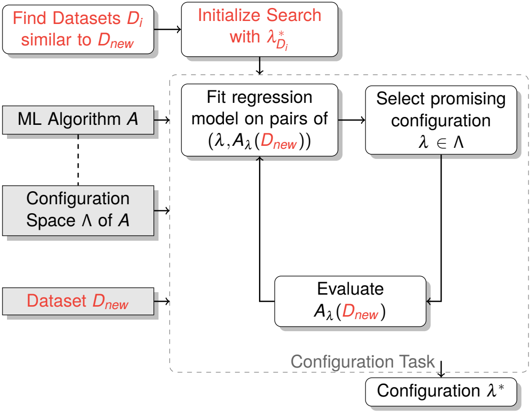
Sequential Model-based Bayesian Optimization (SMBO)



Metalearning-Initialized SMBO (MI-SMBO)



Metalearning-Initialized SMBO (MI-SMBO)



Metafeatures

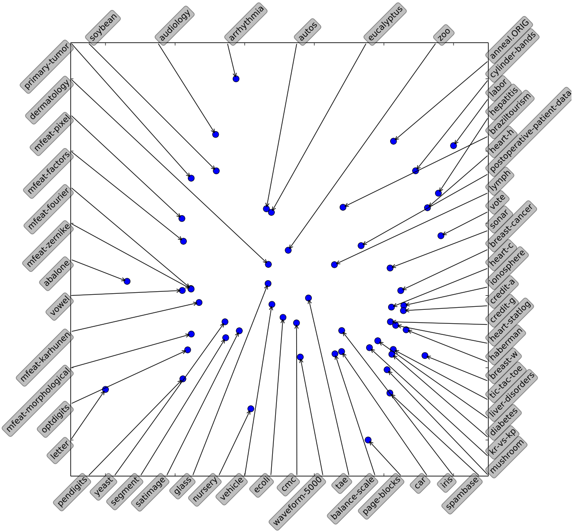
- # training examples: 150
- # classes: 3
- # features: 4
- # numerical features: 4
- # categorical features: 0
- missing values? No

Metalearning-Initialized Bayesian Optimization

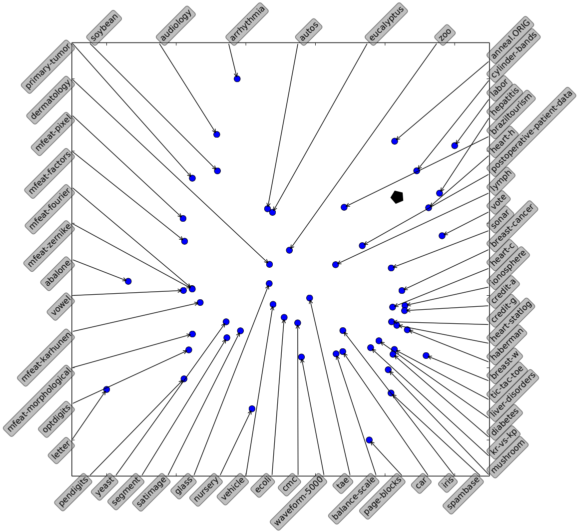
For a new dataset D_{new}:

- Sort known datasets $D_{1:N}$ by distance to D_{new}.
- For each of these datasets, extract the best known hyperparameter configuration $\lambda^*_{D_i}$.
- Initialize SMBO with the first k hyperparameter configurations from the sorted list.

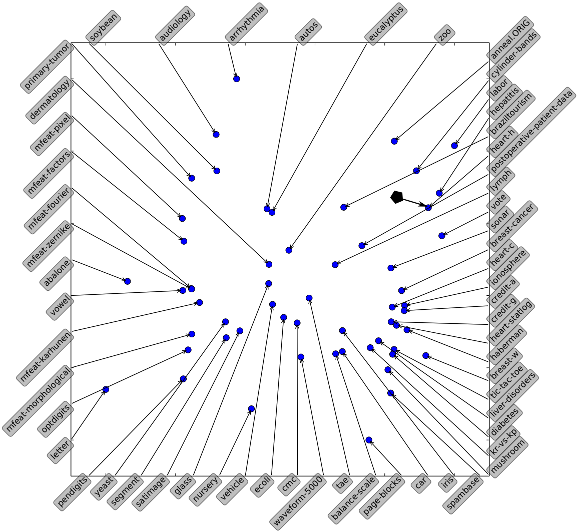
Similarity of Datasets



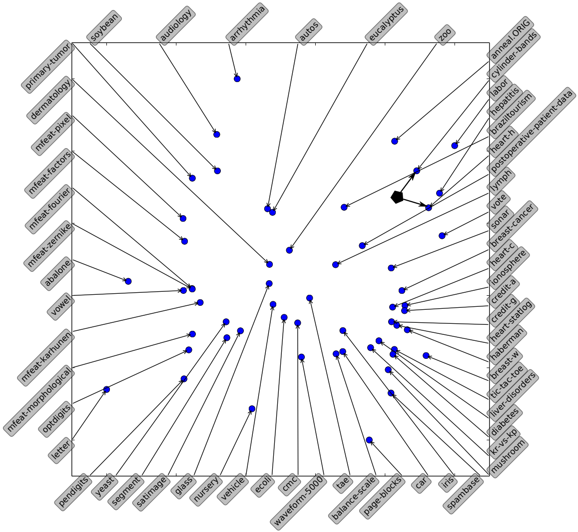
Finding the nearest datasets (1)



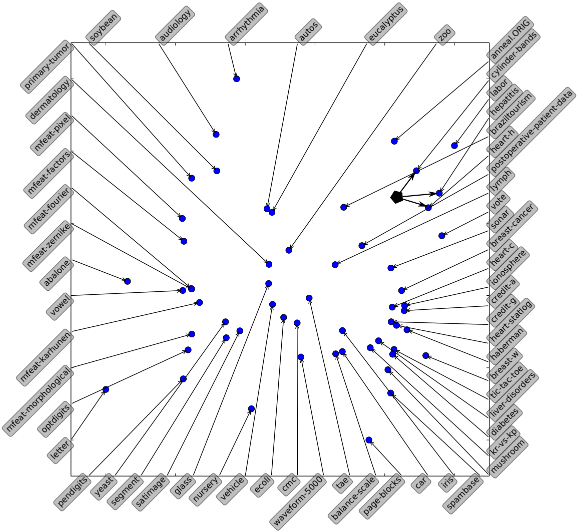
Finding the nearest datasets (2)



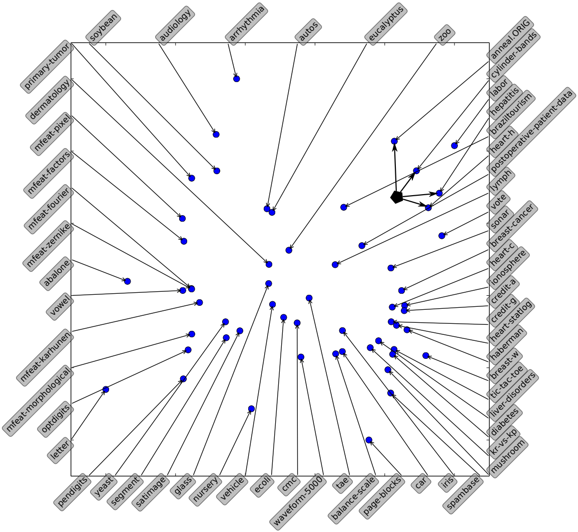
Finding the nearest datasets (3)



Finding the nearest datasets (3)



Finding the nearest datasets (4)



Distance metric (1)

Commonly used in literature, the L_1 norm:

$$d(D_{\text{new}}, D_j) = \sum_i |m_i^{\text{new}} - m_i^j| \tag{1}$$

Experimental Setup

- 57 datasets from the OpenML repository

Experimental Setup

- 57 datasets from the OpenML repository
- 46 metafeatures from the literature:
  - Split into five different subsets, including landmarking [Pfahringer et al. 2000]

Experimental Setup

- 57 datasets from the OpenML repository
- 46 metafeatures from the literature:
  - Split into five different subsets, including landmarking [Pfahringer et al. 2000]
- Two case studies
  - Support Vector Machine with MI-Spearmint [Snoek et al. 2012]
  - AutoSklearn with MI-SMAC [Hutter et al. 2011]

Experimental Setup

- 57 datasets from the OpenML repository
- 46 metafeatures from the literature:
  - Split into five different subsets, including landmarking [Pfahringer et al. 2000]
- Two case studies
  - Support Vector Machine with MI-Spearmint [Snoek et al. 2012]
  - AutoSklearn with MI-SMAC [Hutter et al. 2011]
- Tried 5, 10, 20 and 25 initial configurations

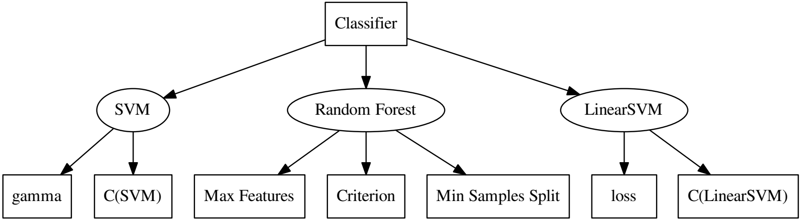
Experimental Setup

- 57 datasets from the OpenML repository
- 46 metafeatures from the literature:
  - Split into five different subsets, including landmarking [Pfahringer et al. 2000]
- Two case studies
  - Support Vector Machine with MI-Spearmint [Snoek et al. 2012]
  - AutoSklearn with MI-SMAC [Hutter et al. 2011]
- Tried 5, 10, 20 and 25 initial configurations
- ran each instantiation 10 times on each dataset
  $\rightarrow$ 26220 optimization runs

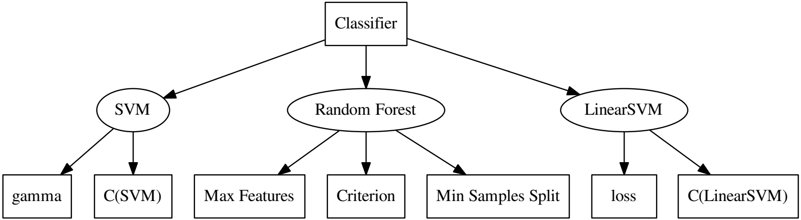
Experimental Setup

- 57 datasets from the OpenML repository
- 46 metafeatures from the literature:
  - Split into five different subsets, including landmarking [Pfahringer et al. 2000]
- Two case studies
  - Support Vector Machine with MI-Spearmint [Snoek et al. 2012]
  - AutoSklearn with MI-SMAC [Hutter et al. 2011]
- Tried 5, 10, 20 and 25 initial configurations
- ran each instantiation 10 times on each dataset
  $\rightarrow$ 26220 optimization runs
- therefore, precomputed a dense grid for every dataset

Combined Algorithm Selection and Hyperparameter Optimization problem (CASH)



Combined Algorithm Selection and Hyperparameter Optimization problem (CASH)



[Auto-WEKA, Thornton et al. 2013]

AutoSklearn: Hyperparameters

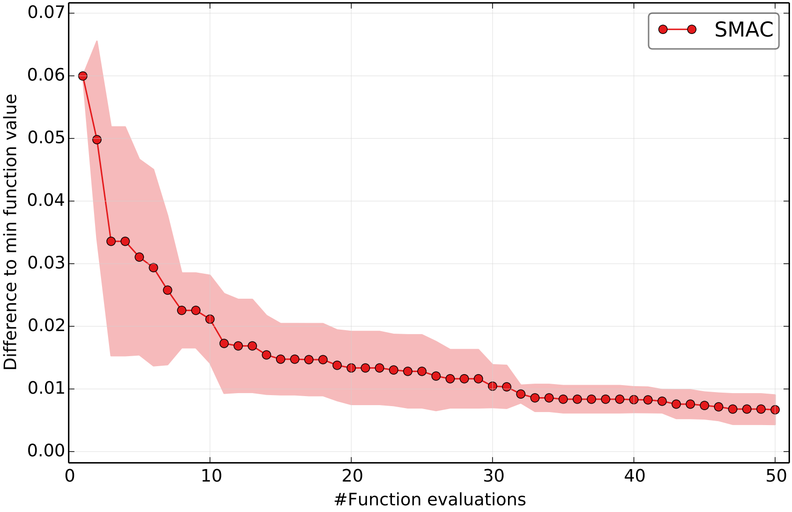
Component	Hyperparameter	# Values
Main	$\lambda_{\text{classifier}}$	3
Main	preprocessing	2
SVM	$\log_2(C)$	21
SVM	$\log_2(\gamma)$	19
LinearSVM	$\log_2(C)$	21
LinearSVM	penalty	2
RF	min splits	5
RF	max features	10
RF	criterion	2
PCA	variance to keep	2

AutoSklearn: Hyperparameters

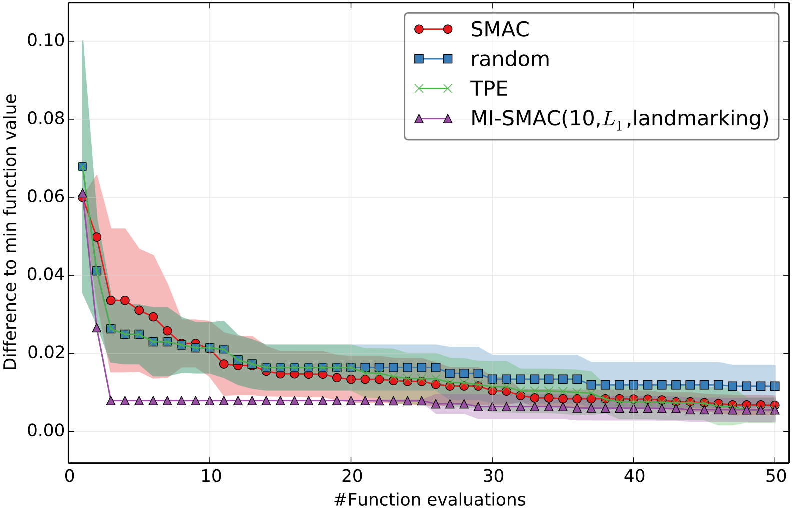
Component	Hyperparameter	# Values
Main	$\lambda_{\text{classifier}}$	3
Main	preprocessing	2
SVM	$\log_2(C)$	21
SVM	$\log_2(\gamma)$	19
LinearSVM	$\log_2(C)$	21
LinearSVM	penalty	2
RF	min splits	5
RF	max features	10
RF	criterion	2
PCA	variance to keep	2

1623 hyperparameter configurations

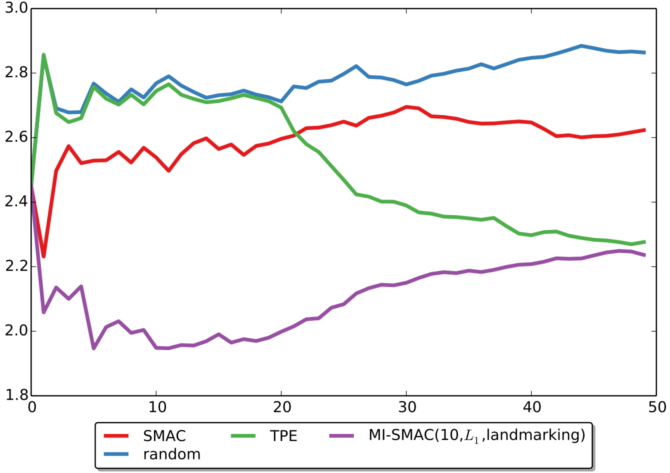
AutoSklearn: Results (1)



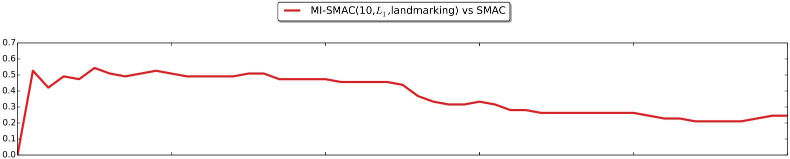
AutoSklearn: Results (1)



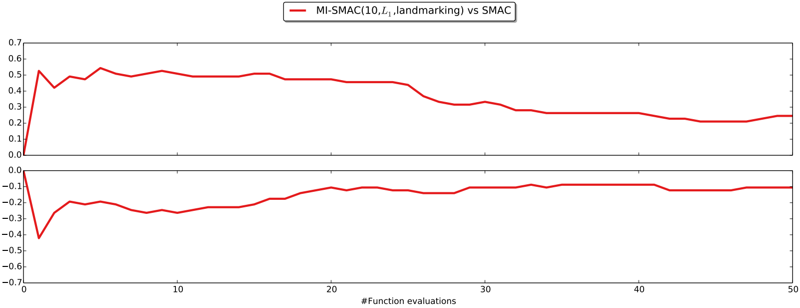
AutoSklearn: Results (2)

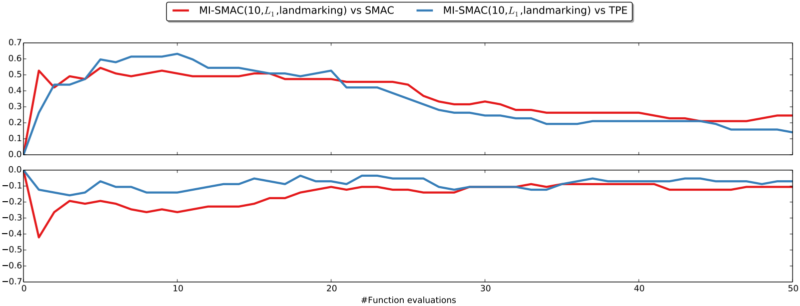


AutoSklearn: Results (3)

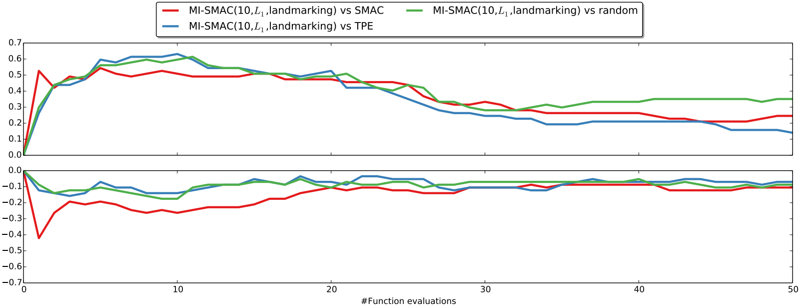


AutoSklearn: Results (3)




MI-SMAC(10,L_1,landmarking) vs SMAC
MI-SMAC(10,L_1,landmarking) vs TPE
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.7
0
10
20
30
40
50
#Function evaluations

AutoSklearn: Results (3)



Open questions

- Does MI-SMBO scale to larger configuration spaces?
- What if gridsearch is too expensive?
- Can the metalearning component be added directly into the SMBO procedure?

Take home messages

- SMBO can be substantially improved by providing good initial configurations.
- Metalearning provides a sound framework to find these configurations.
- MI-SMAC improves on state-of-the-art methods on a large configuration space, namely AutoSklearn.

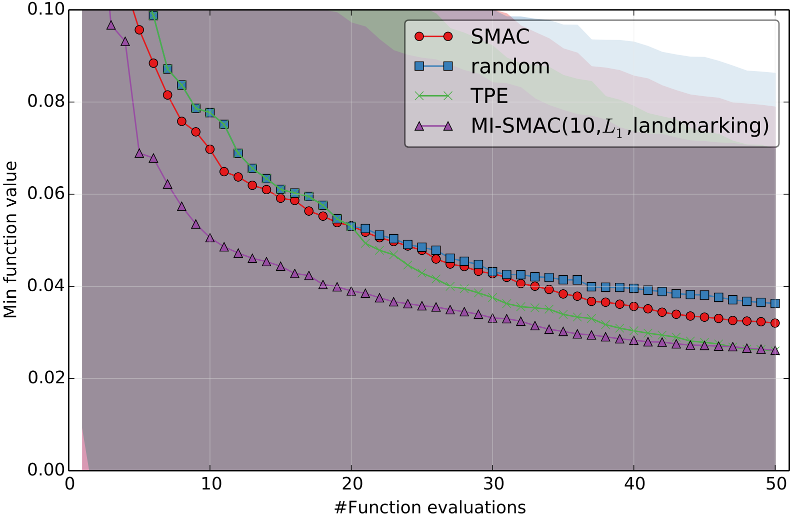
The end

Thank you for your attention.

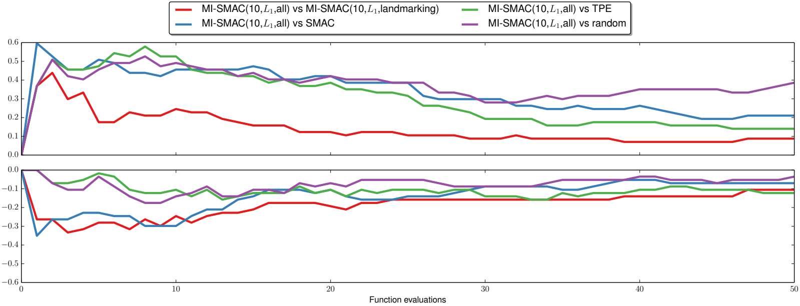
Further questions: `feurerm@cs.uni-freiburg.de`

This presentation was partially supported by an *ECCAI Travel Award* and the *ECCAI sponsors*.

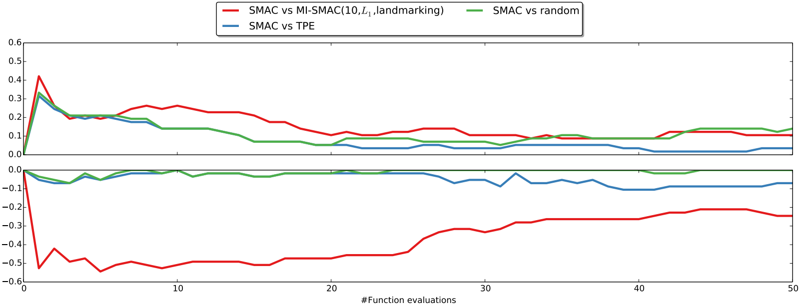
AutoSklearn: Results (5)



AutoSklearn: Results (7)



AutoSklearn: Results (8)



AutoSklearn: Results (9)