Efficient Benchmarking of Hyperparameter Optimizers via Surrogates

Katharina Eggensperger and Frank Hutter
University of Freiburg
{eggenspk, fh}@cs.uni-freiburg.de

Holger H. Hoos and Kevin Leyton-Brown
University of British Columbia
{hoos, kevinl}@cs.ubc.ca

...in 30 seconds

- Benchmarking hyperparameter optimization methods is costly, as it requires many evaluations of the used benchmark problem.
- We propose cheap, but realistic surrogate benchmarks based on predictive performance models.
- Our benchmarks are publicly available and allow extensive white-box tests and fast evaluation of optimization algorithms.

Our Approach

1. Collect <configuration, performance> pairs from different benchmarks
2. Train Regression model
3. Replace call to benchmark function with model prediction

Question: How well does this work?

8 regression models
- Tree-based models
- Gaussian Processes
- Support Vector Regression
- K-Nearest Neighbour
- Linear Regression

9 benchmark functions, e.g.
- Logistic Regression
- (Deep) Neural Networks
- onlineLDA

Collect data by conducting optimization runs and random search
Evaluate quality of regression models
Compare optimizer performance on true vs. surrogate benchmark

Empirical Results

- Almost perfect for low-dimensional benchmarks and still acceptable for higher dimensions
- Reduce benchmark overhead to <1 sec

Answer: Surrogates work well, especially based on Random Forests

Setup

Experiments

- Bayesian optimization: Surrogate benchmark
- Randomized search: Real benchmark
- Randomized search: Surrogate benchmark

Table: Properties of the benchmarks for which we provide surrogate benchmarks

Table: Empirical comparison of three optimizers on various real and surrogate-based benchmarks

Our surrogate benchmarks...

- provide a 60 to 3600 x speedup
- require negligible computational resources
- allow unit tests
- help to analyze how optimizers work on complex benchmark

... and are publicly available:

For more information visit:
www.automl.org/benchmarks.html

Implementation publicly available: https://github.com/KEggensperger/SurrogateBenchmarks

Configuration Space \( \Lambda \)

Select \( \lambda \in \Lambda \)

Assess \( \text{Performance } f(\lambda) \)

Best performing \( \lambda^* \)

Hyperparameter Optimization

Problems with existing benchmark functions

Realistic benchmark problems:
- Complex & interesting
- Expensive to evaluate
- Complicated to set up (libraries, dependencies, special hardware, etc.)
Examples: Deep Neural Networks, onlineLDA, AutoWEKA

Synthetic test functions:
- Easy to set up
- Cheap to evaluate
- Unrealistic shape & too smooth
Examples: BBOB, Table-Lookups, Brain

Benefit

Our surrogate benchmarks...

- and are publicly available:

Implementation publicly available: https://github.com/KEggensperger/SurrogateBenchmarks

For more information visit: www.automl.org/benchmarks.html