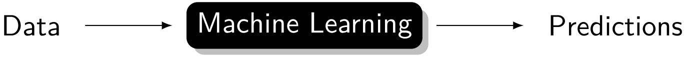
A Hands-On Introduction to Automatic Machine Learning

Lars Kotthoff

University of Wyoming
larsko@uwyo.edu

AutoML Workshop, 28 August 2018, Nanjing

Machine Learning

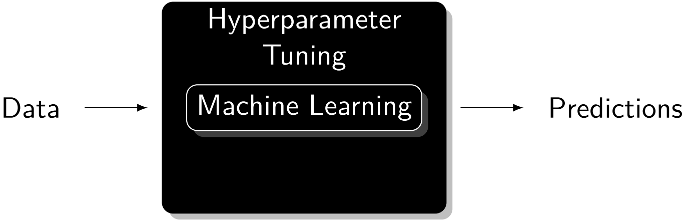


Data

Machine Learning

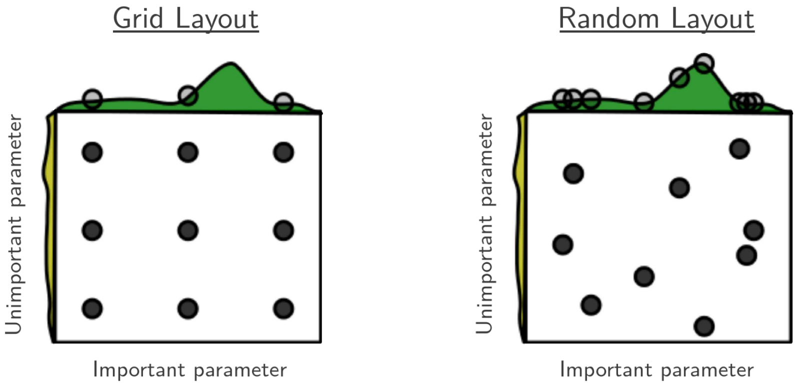
Predictions

Automatic Machine Learning



Grid and Random Search

- ▷ evaluate certain points in parameter space

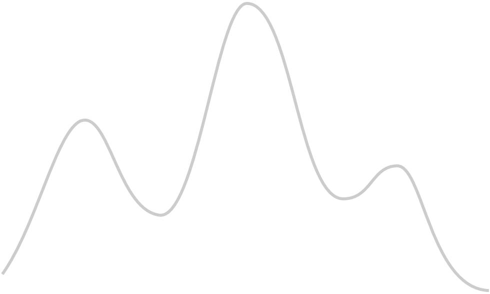


Bergstra, James, and Yoshua Bengio. "Random Search for Hyper-Parameter Optimization." J. Mach. Learn. Res. 13, no. 1 (February 2012): 281–305.

Local Search

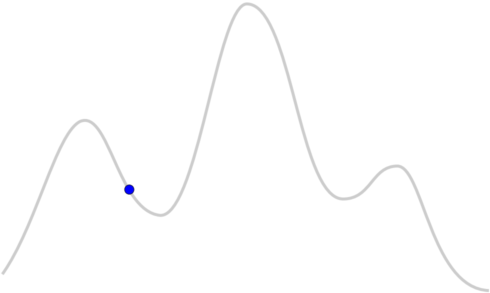
- start with random configuration
- change a single parameter (local search step)
- if better, keep the change, else revert
- repeat, stop when resources exhausted or desired solution quality achieved
- restart occasionally with new random configurations

Local Search Example



graphics by Holger Hoos

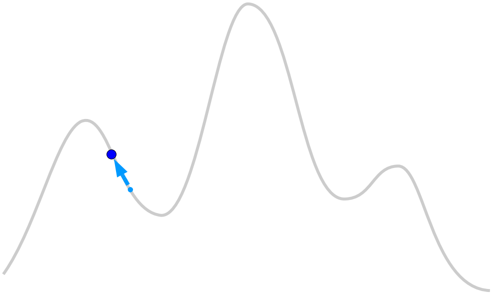
Local Search Example



Initialisation

graphics by Holger Hoos

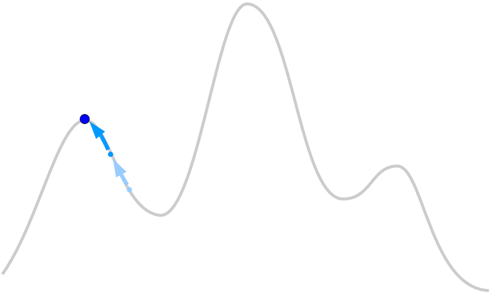
Local Search Example



Local Search

graphics by Holger Hoos

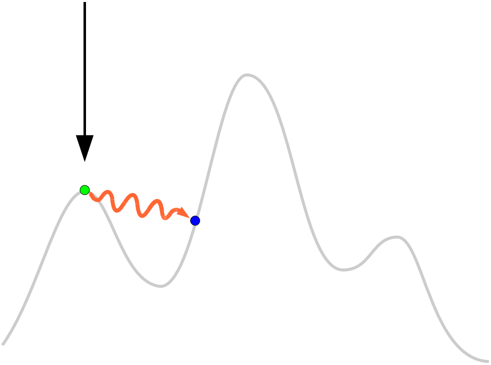
Local Search Example



Local Search

graphics by Holger Hoos

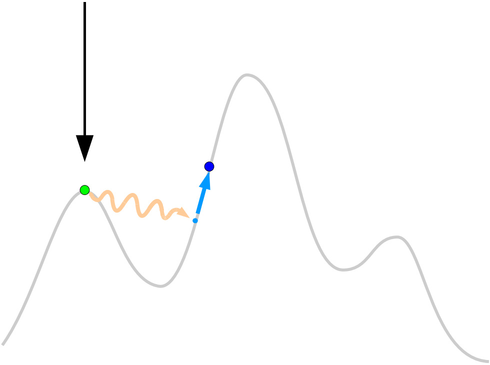
Local Search Example



Perturbation

graphics by Holger Hoos

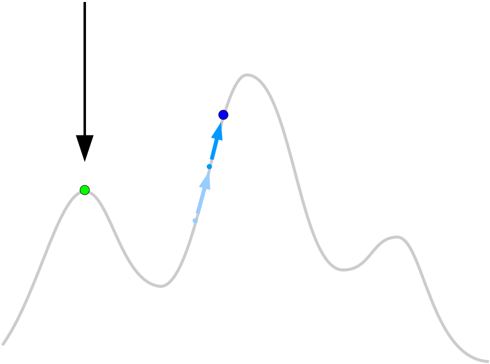
Local Search Example



Local Search

graphics by Holger Hoos

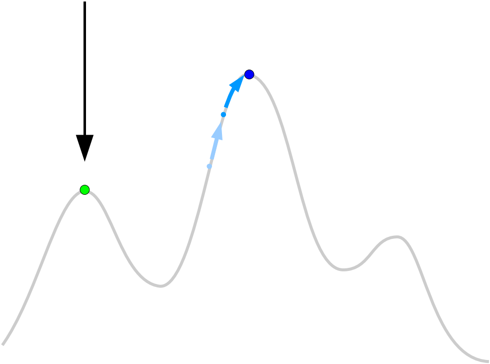
Local Search Example



Local Search

graphics by Holger Hoos

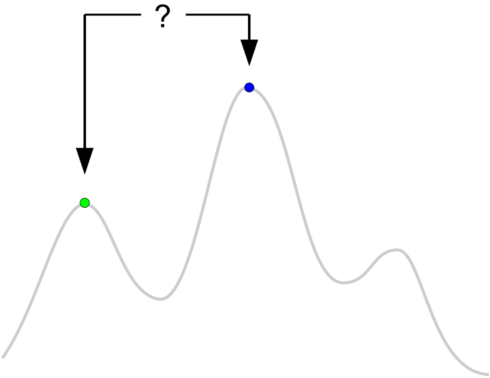
Local Search Example



Local Search

graphics by Holger Hoos

Local Search Example



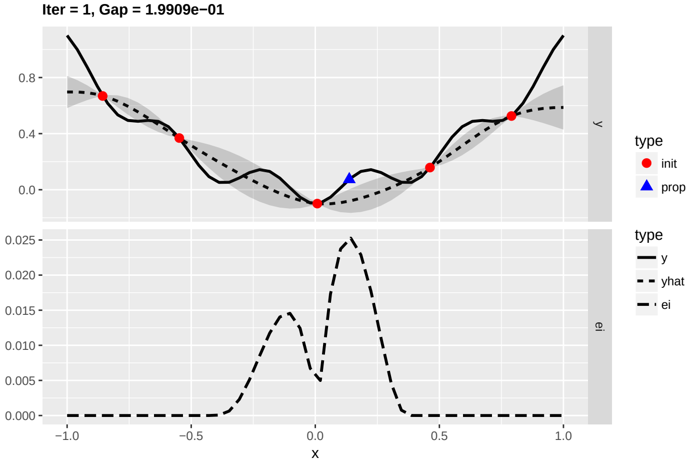
Selection (using Acceptance Criterion)

graphics by Holger Hoos

Model-Based Search

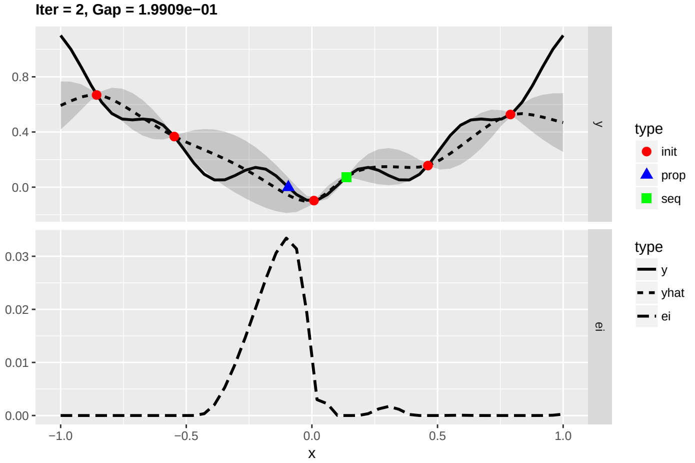
- evaluate small number of configurations
- build model of parameter-performance surface based on the results
- use model to predict where to evaluate next
- repeat, stop when resources exhausted or desired solution quality achieved
- allows targeted exploration of promising configurations

Model-Based Search Example



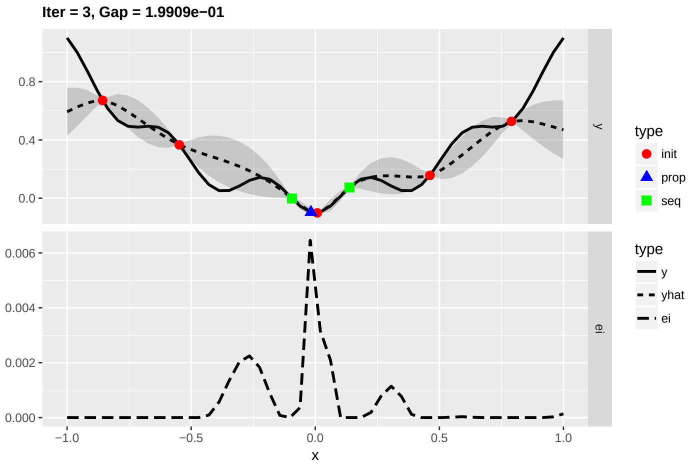
Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



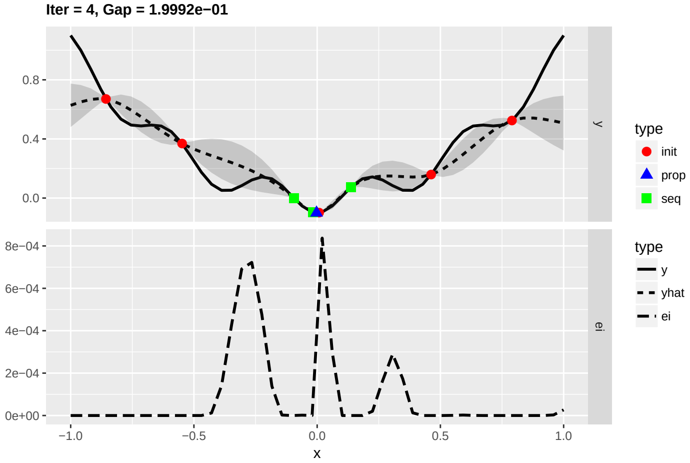
Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



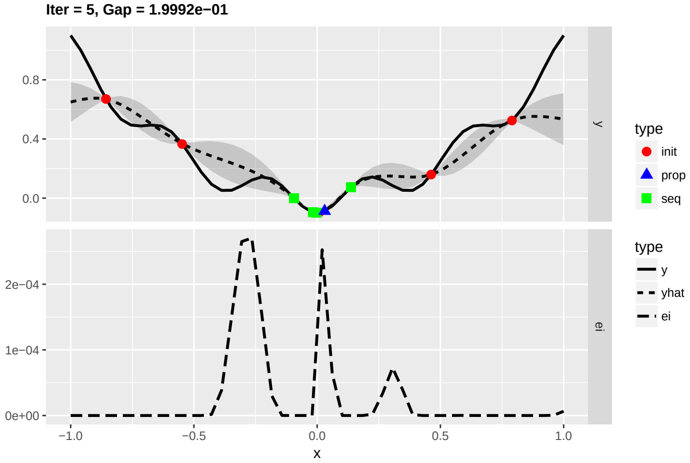
[1] Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



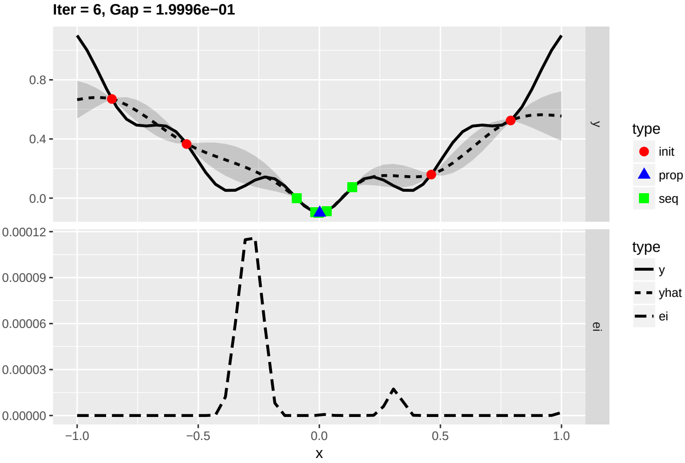
Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



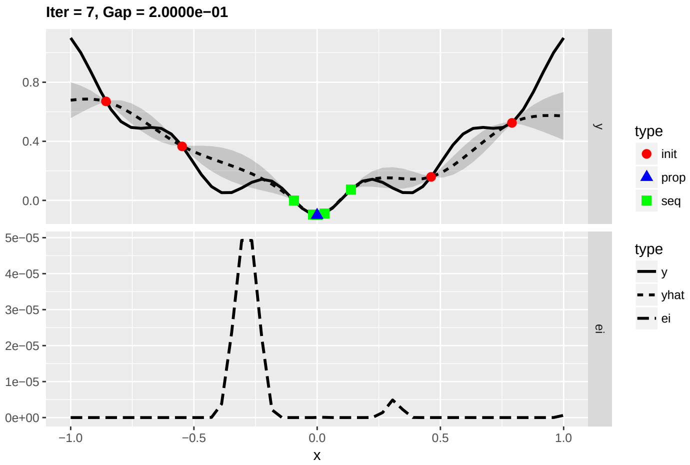
Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. “MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions,” March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



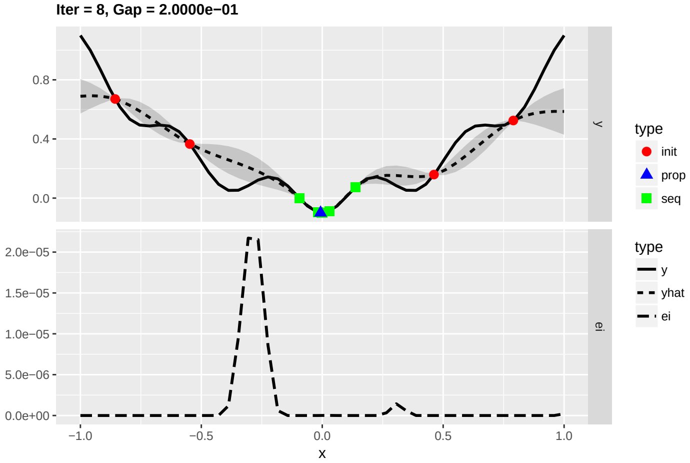
Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



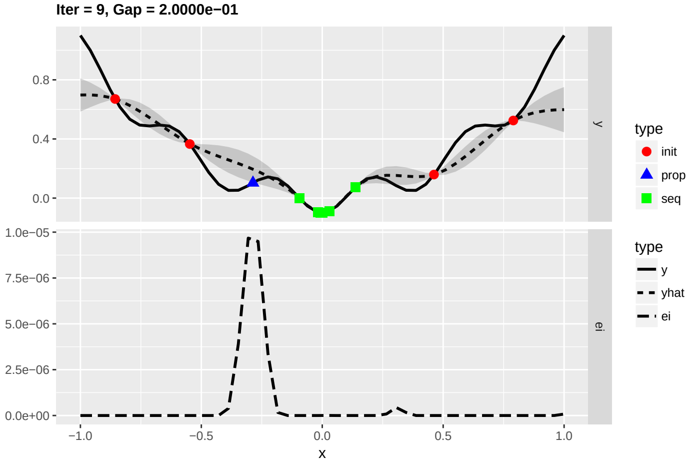
Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



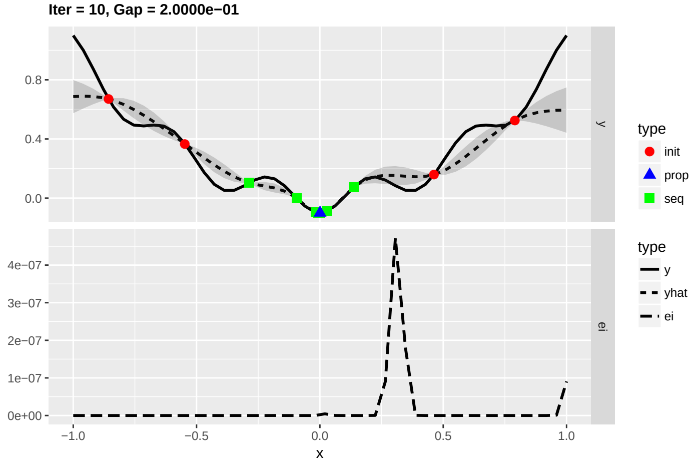
Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. “MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions,” March 9, 2017. http://arxiv.org/abs/1703.03373.

Model-Based Search Example



Bischl, Bernd, Jakob Richter, Jakob Bossek, Daniel Horn, Janek Thomas, and Michel Lang. "MlrMBO: A Modular Framework for Model-Based Optimization of Expensive Black-Box Functions," March 9, 2017. http://arxiv.org/abs/1703.03373.

Problems

- How good are we really?
- How much of it is just random chance?
- Can we do better?

Underlying Issues

- true performance landscape unknown
- resources allow to explore only tiny part of hyperparameter space
- results inherently stochastic

Potential Solutions

- better-understood benchmarks
- more comparisons
- more runs with different random seed

Two-Slide MBO ML

```
# http://www.cs.uwyo.edu/~larsko/mbo.py
params = { 'C': np.logspace(-2, 10, 13),
               'gamma': np.logspace(-9, 3, 13) }
param_grid = [ { 'C': x, 'gamma': y } for x in params['C']
                                      for y in params['gamma'] ]
# [{'C': 0.01, 'gamma': 1e-09}, {'C': 0.01, 'gamma': 1e-08}...]

initial_samples = 3
evals = 10
random.seed(1)

def est_acc(pars):
    clf = svm.SVC(**pars)
    return np.median(cross_val_score(clf, iris.data, iris.target, cv = 10))

data = []
for pars in random.sample(param_grid, initial_samples):
    acc = est_acc(pars)
    data += [ list(pars.values()) + [ acc ] ]
# [[1.0, 0.1, 1.0],
#  [1000000000.0, 1e-07, 1.0],
#  [0. 1, 1e-06,0.9333333333333333]]
```

Two-Slide MBO ML

```
regr = RandomForestRegressor(random_state = 0)
for evals in range(0, evals):
    df = np.array(data)
    regr.fit(df[:,0:2], df[:,2])

    preds = regr.predict([ list(pars.values()) for pars in param_grid ])
    i = preds.argmax()

    acc = est_acc(param_grid[i])
    data += [ list(param_grid[i].values()) + [ acc ] ]
    print("{}: best predicted {} for {}, actual {}"
            .format(evals, round(preds[i], 2), param_grid[i], round(acc, 2)))

i = np.array(data)[:,2].argmax()
print("Best accuracy ({}) for parameters {}".format(data[i][2], data[i][0:2]))
```

Two-Slide MBO ML

```
0: best predicted 0.99 for {'C': 1.0, 'gamma': 1e-09}, actual 0.93
1: best predicted 0.99 for {'C': 1000000000.0, 'gamma': 1e-09}, actual 0.93
2: best predicted 0.99 for {'C': 1000000000.0, 'gamma': 0.1}, actual 0.93
3: best predicted 0.97 for {'C': 1.0, 'gamma': 0.1}, actual 1.0
4: best predicted 0.99 for {'C': 1.0, 'gamma': 0.1}, actual 1.0
5: best predicted 1.0 for {'C': 1.0, 'gamma': 0.1}, actual 1.0
6: best predicted 1.0 for {'C': 1.0, 'gamma': 0.1}, actual 1.0
7: best predicted 1.0 for {'C': 1.0, 'gamma': 0.1}, actual 1.0
8: best predicted 1.0 for {'C': 0.01, 'gamma': 0.1}, actual 0.93
9: best predicted 1.0 for {'C': 1.0, 'gamma': 0.1}, actual 1.0
Best accuracy (1.0) for parameters [1.0, 0.1]
```

Tools and Resources

iRace	`http://iridia.ulb.ac.be/irace/`
TPOT	`https://github.com/EpistasisLab/tpot`
mlrMBO	`https://github.com/mlr-org/mlrMBO`
SMAC	`http://www.cs.ubc.ca/labs/beta/Projects/SMAC/`
Spearmint	`https://github.com/HIPS/Spearmint`
TPE	`https://jaberg.github.io/hyperopt/`

Auto-WEKA	`http://www.cs.ubc.ca/labs/beta/Projects/autoweka/`
Auto-sklearn	`https://github.com/automl/auto-sklearn`

Available soon: edited book on automatic machine learning `https://www.automl.org/book/` (Frank Hutter, Lars Kotthoff, Joaquin Vanschoren)

I'm hiring!

Several funded graduate/postdoc positions available.

NSF