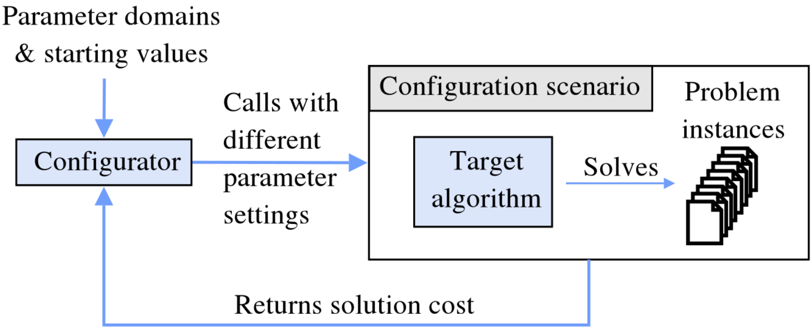
Practical Machine Learning in R

Tuning

Lars Kotthoff[1,2]
larsko@uwyo.edu

[1] with slides from Bernd Bischl and Michel Lang
[2] slides available at http://www.cs.uwyo.edu/~larsko/ml-fac

Tuning

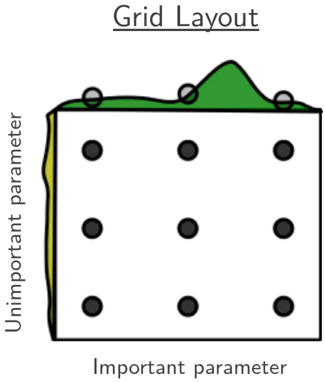


Frank Hutter and Marius Lindauer, “Algorithm Configuration: A Hands on Tutorial”, AAAI 2016

Hyperparameter Tuning

- used to find “best” hyperparameters for a method in a data-dependent way
- important to achieve good performance in practice
- essential for some methods, e.g. SVMs

Grid and Random Search



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

Population-Based Methods

- e.g. Racing and Genetic Algorithms
- start with population of random configurations
- eliminate “weak” individuals
- generate new population from “strong” individuals
- iterate

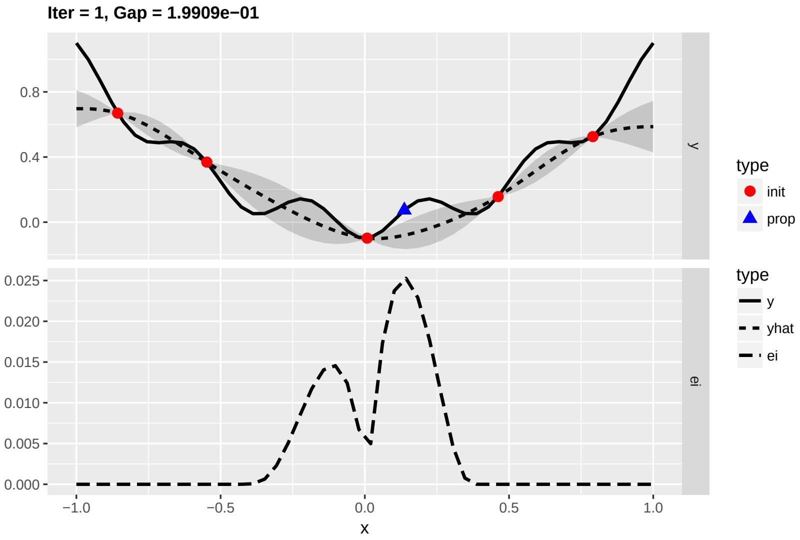
Model-Based Search

- currently considered state-of-the-art
- build surrogate model of parameter-response surface
- evaluate cheap model instead of expensive target function
- use model to propose next point to evaluate target function at
- iterate

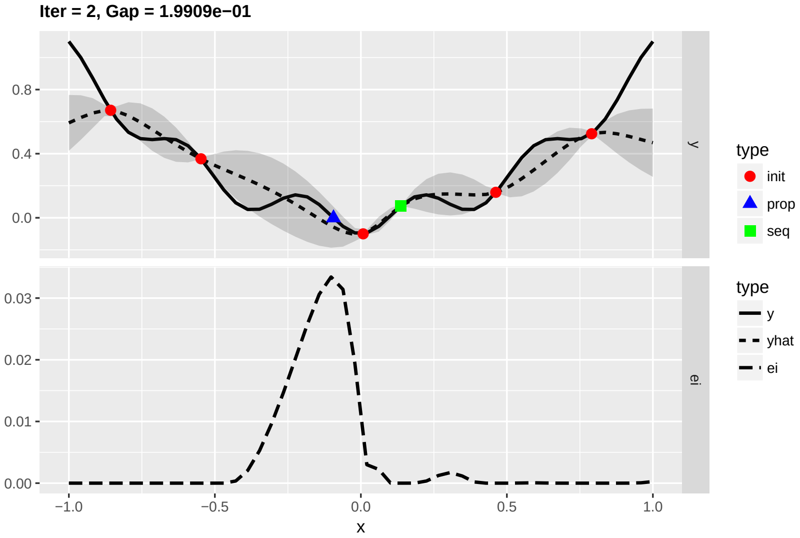
Model-Based Search – Components

- learner for surrogate model
- method for generating set of initial observations
- infill criterion – how to get next evaluation point
- termination criterion

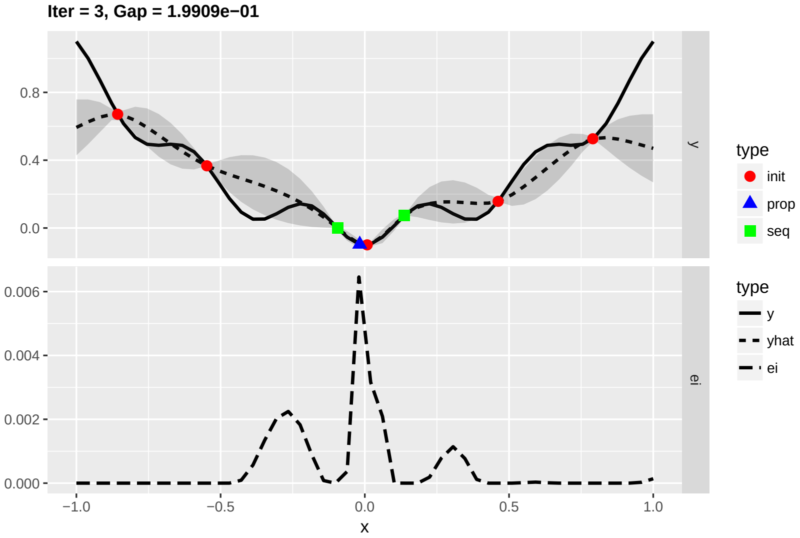
Model-Based Search Example – 1D



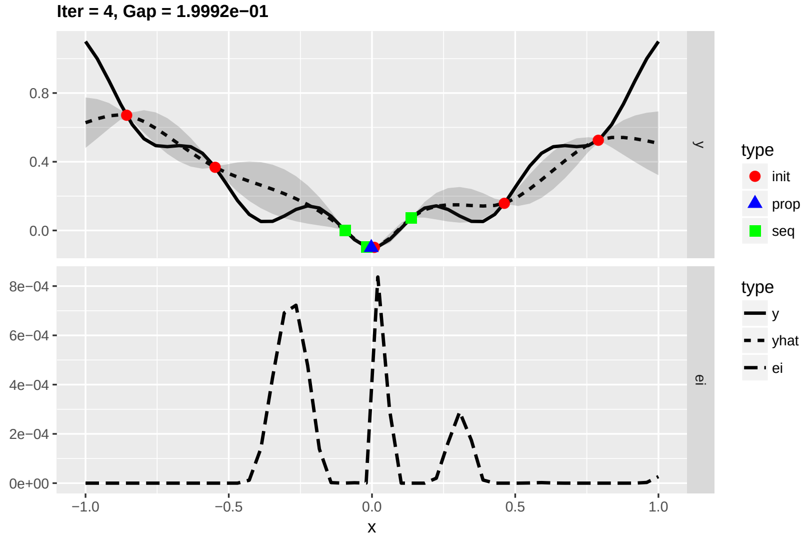
Model-Based Search Example – 1D



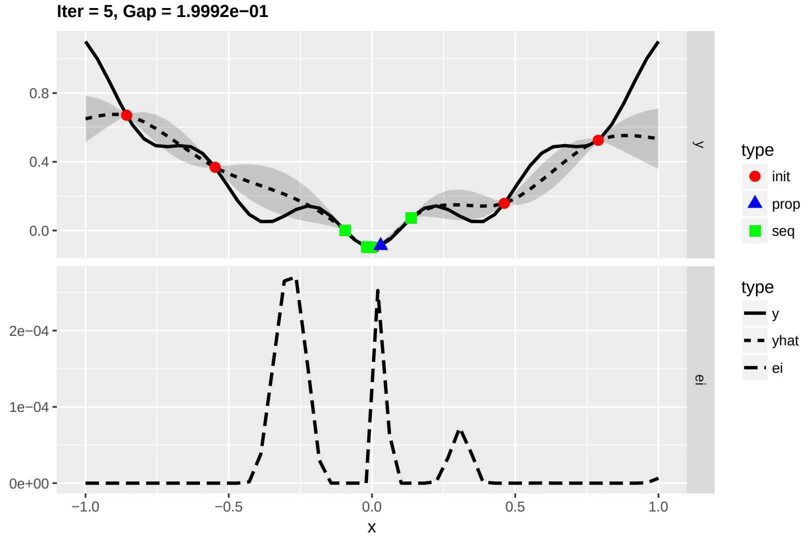
Model-Based Search Example – 1D



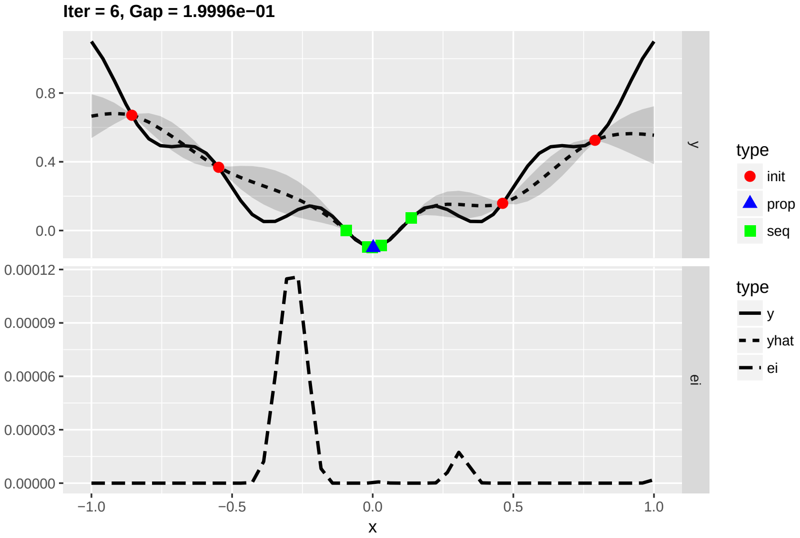
Model-Based Search Example – 1D



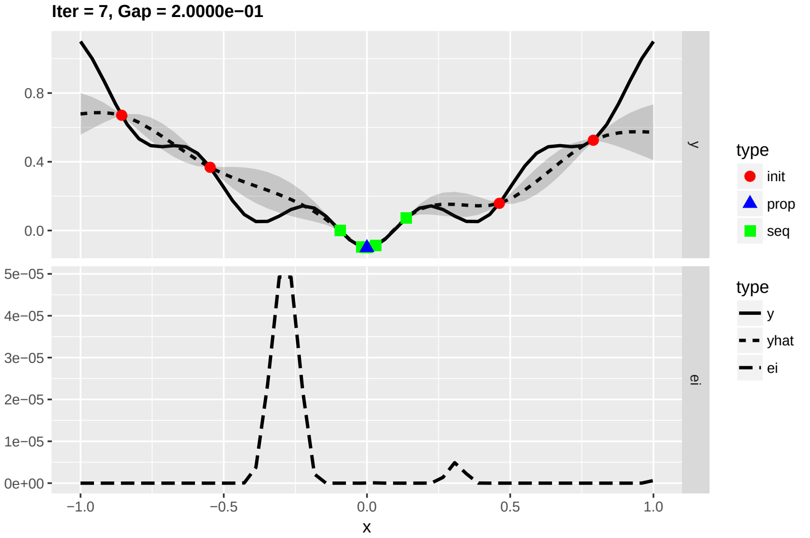
Model-Based Search Example – 1D



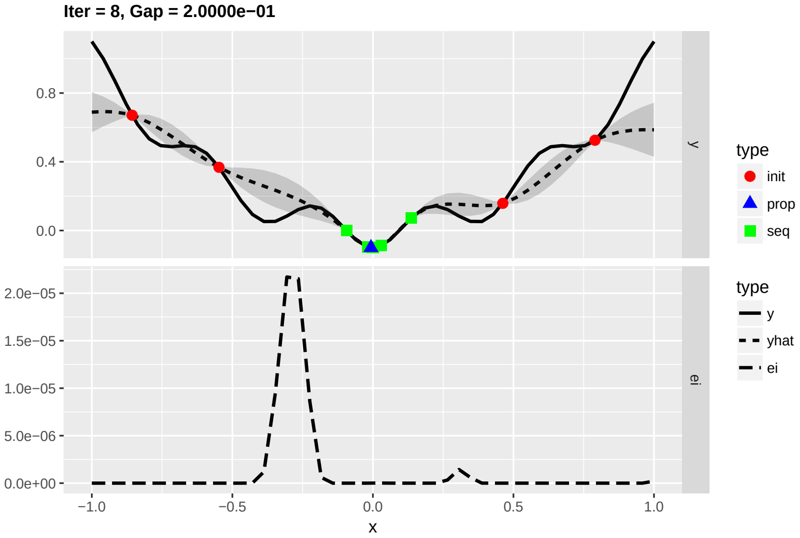
Model-Based Search Example – 1D



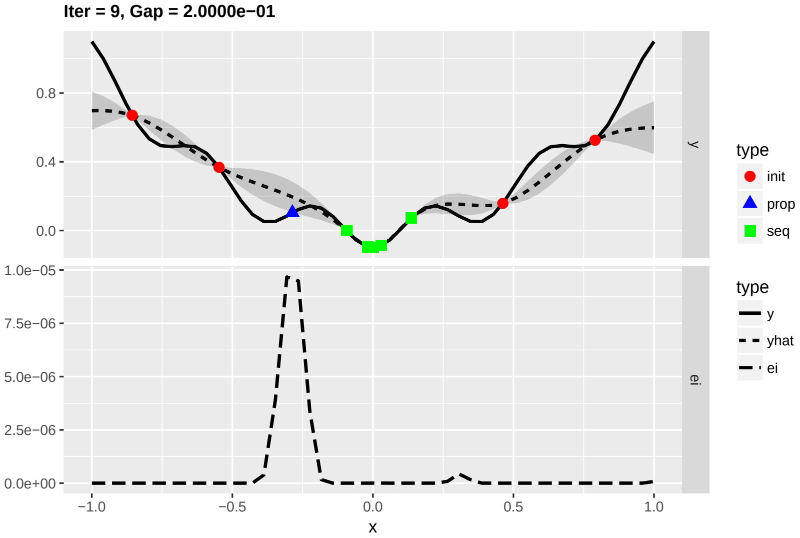
Model-Based Search Example – 1D



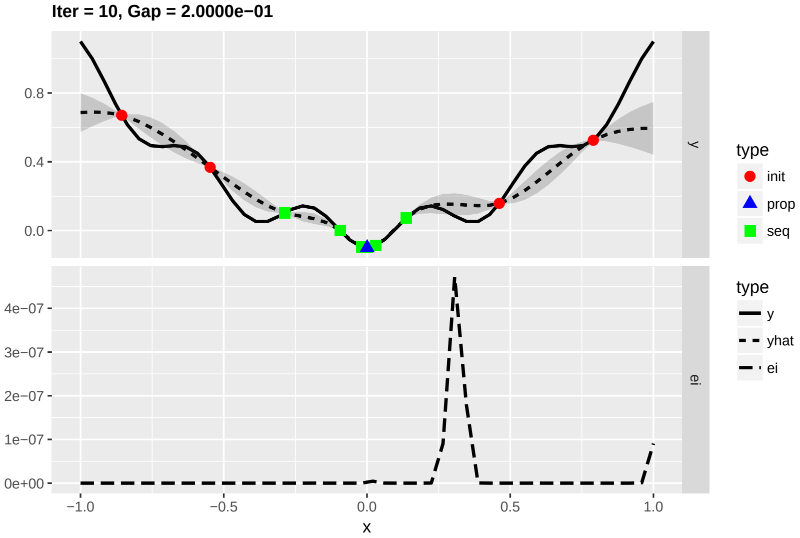
Model-Based Search Example – 1D



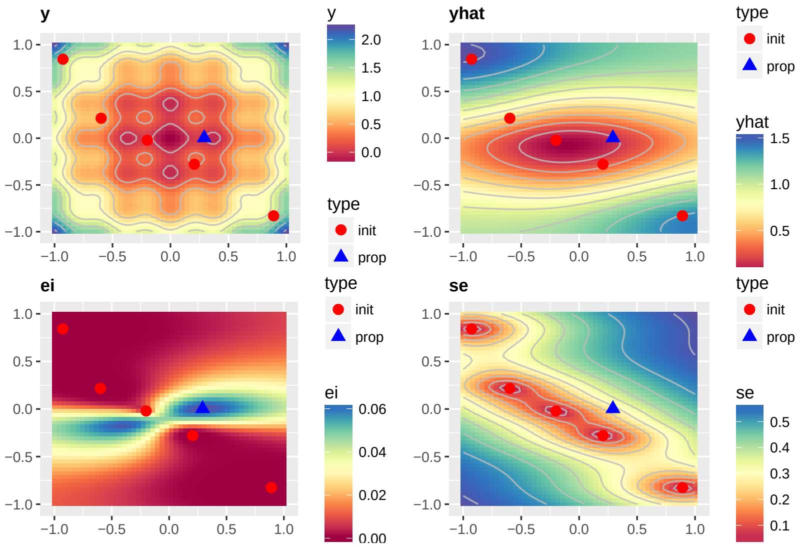
Model-Based Search Example – 1D



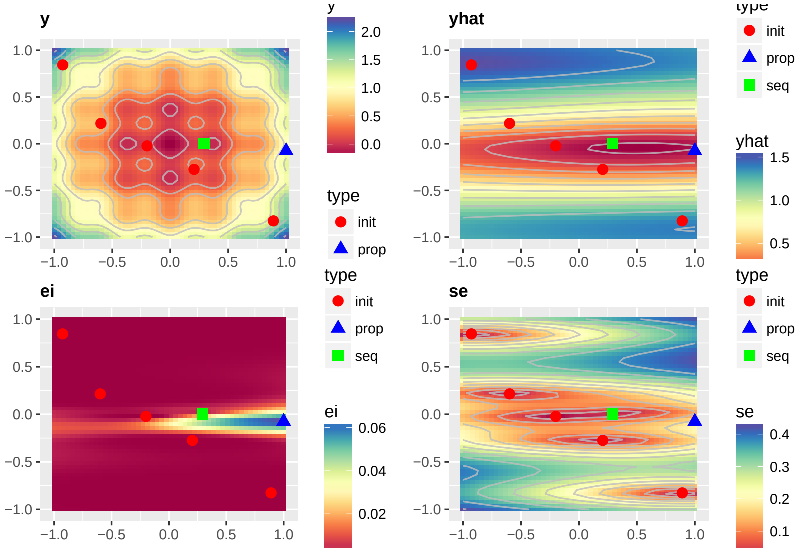
Model-Based Search Example – 1D



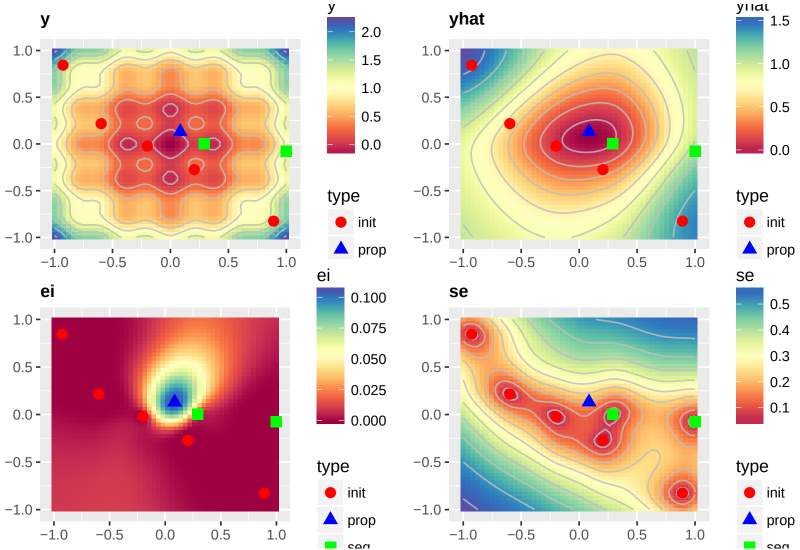
Model-Based Search Example – 2D



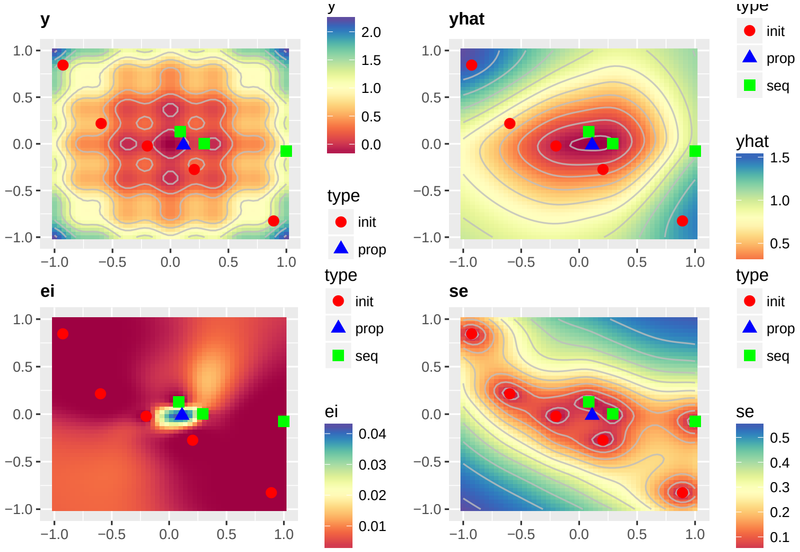
Model-Based Search Example – 2D



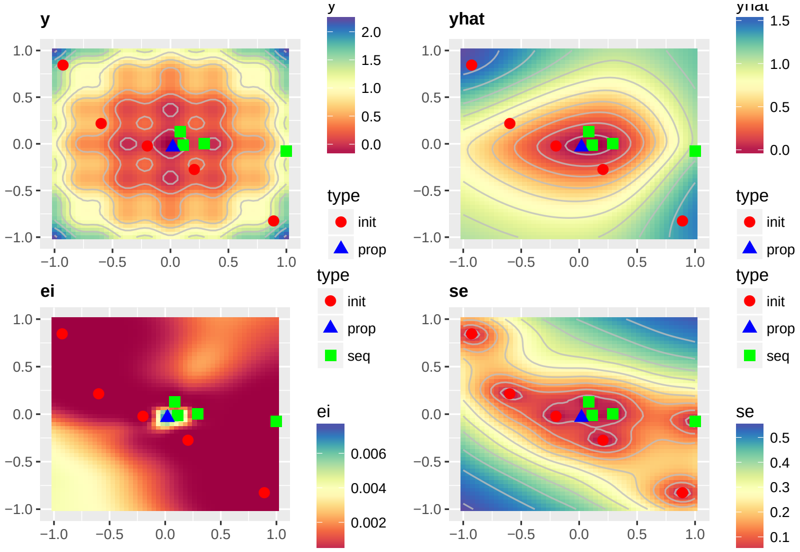
Model-Based Search Example – 2D



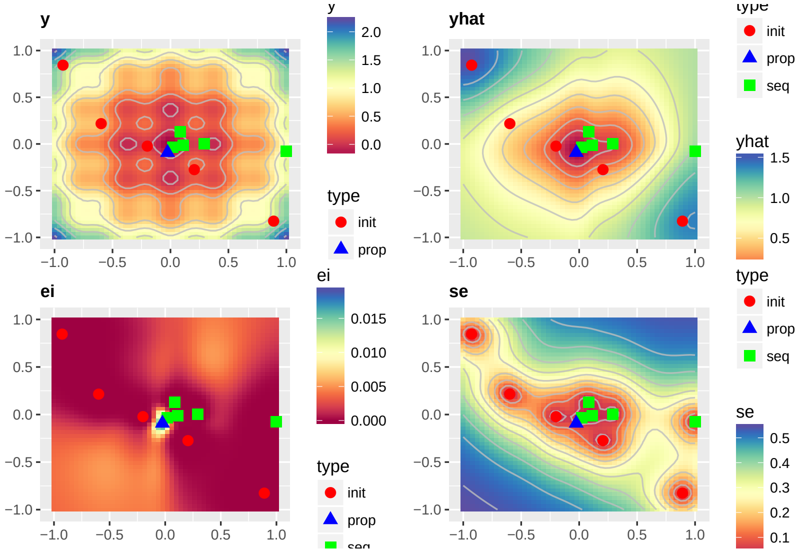
Model-Based Search Example – 2D



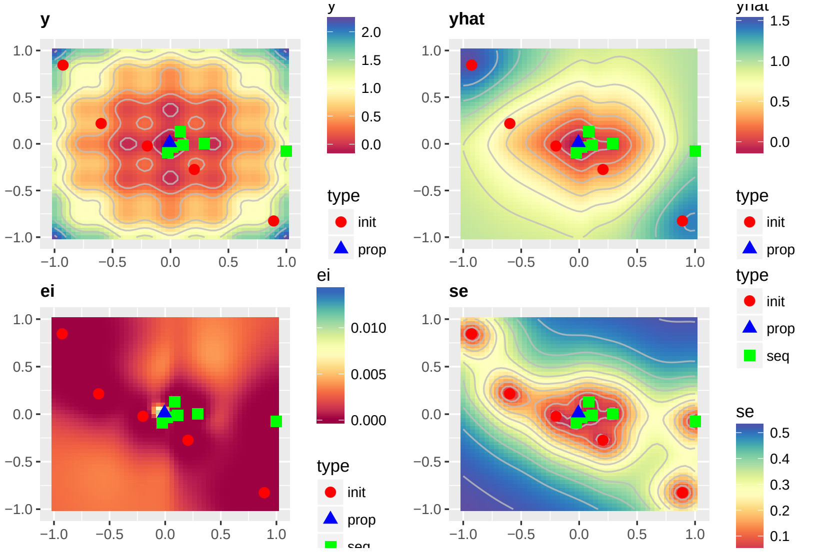
Model-Based Search Example – 2D



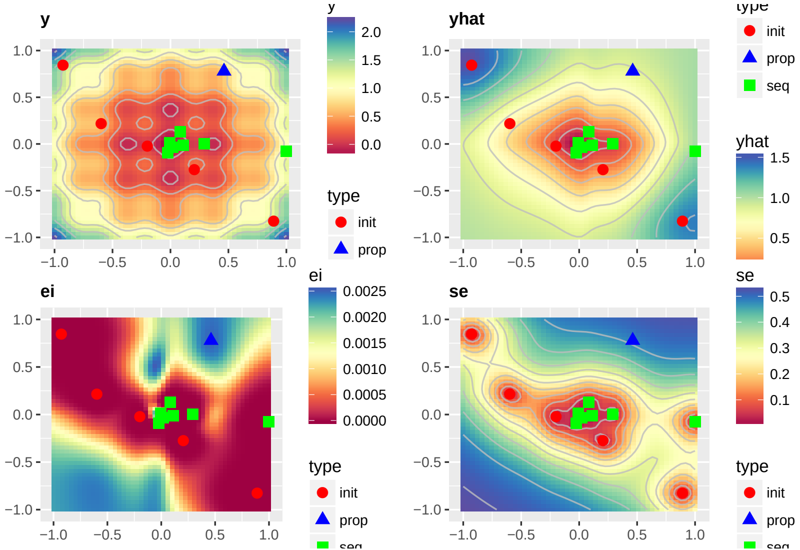
Model-Based Search Example – 2D



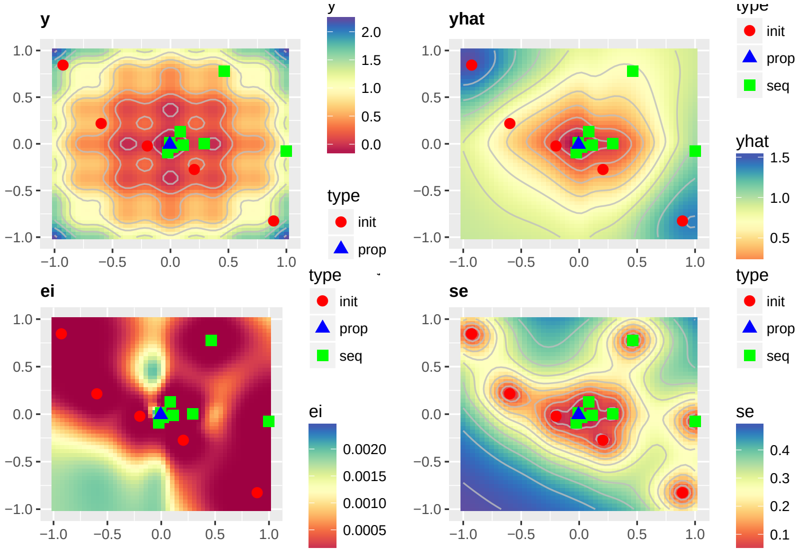
Model-Based Search Example – 2D



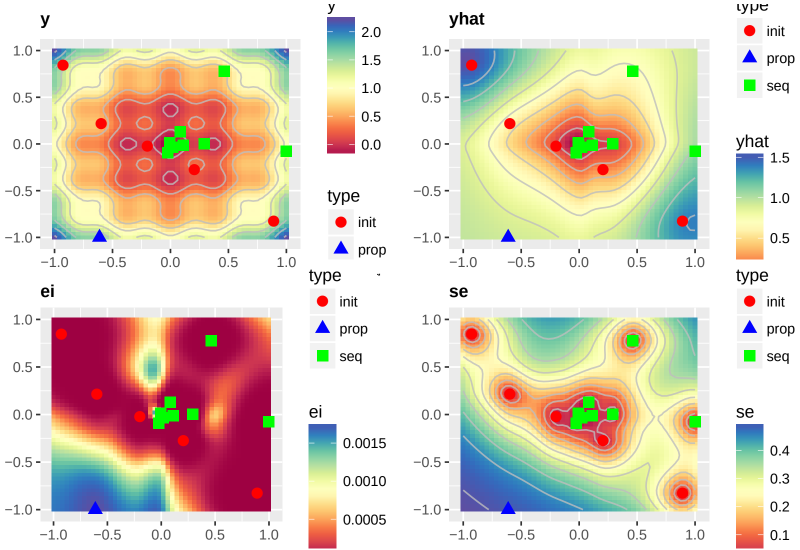
Model-Based Search Example – 2D



Model-Based Search Example – 2D



Model-Based Search Example – 2D



When are we done?

- most approaches incomplete
- cannot prove optimality, not guaranteed to find optimal solution (in finite time)
- performance highly dependent on configuration space

$\rightarrow$ How do we know when to stop?

Time Budget

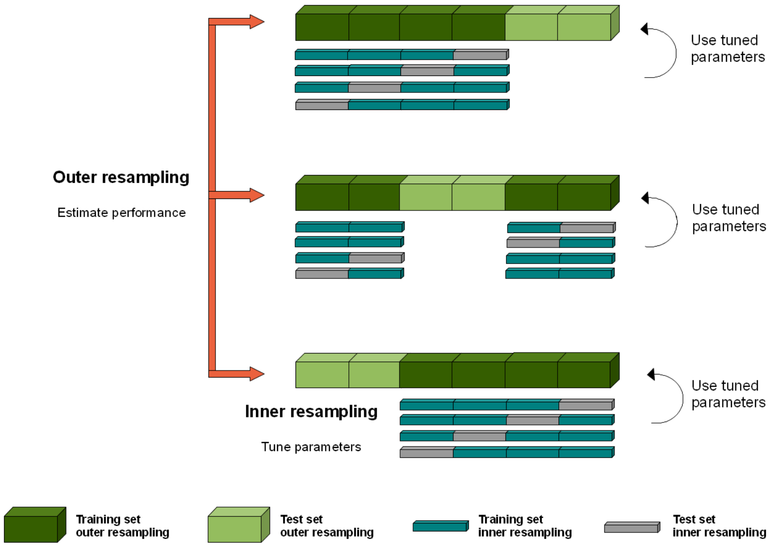
How much time/how many function evaluations?

- too much $\rightarrow$ wasted resources
- too little $\rightarrow$ suboptimal result
- experiment with different settings
- run several times with different random initializations

Evaluation

- repeated evaluation with same train/test split statistically unsound $\rightarrow$ violates independence assumption
- example: parameters have no real effect, only random variation $\rightarrow$ still one parameter setting will "win"
- solution: different train/test splits

Nested Resampling



In mlr

- tuning with different methods available as wrapper
- model-based optimization available in `mlrMBO` package
- nested resampling available as resampling method

Exercises

`http://www.cs.uwyo.edu/~larsko/ml-fac/07-tuning-exercises.Rmd`