Julian Rodemann[1], Thomas Augustin[1]

Towards Prior-Mean Robust Bayesian Optimization

Young Statisticians Session (YSS)

DAGStat 2022

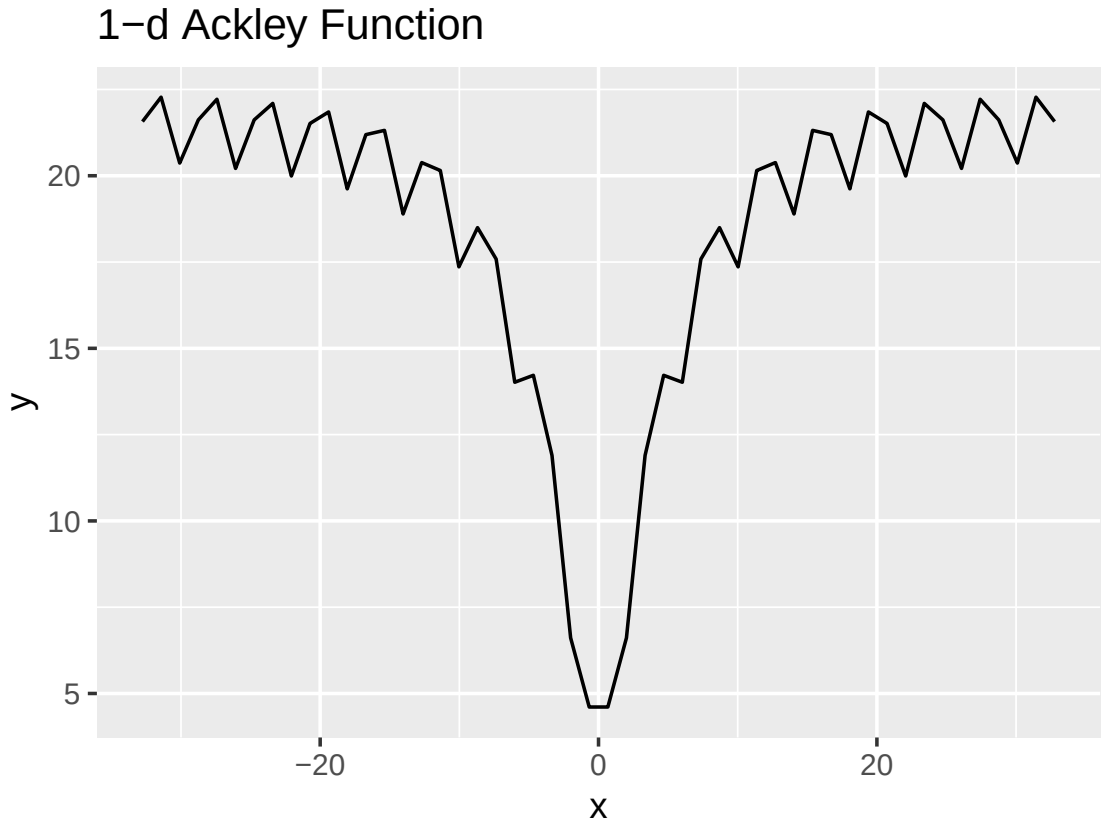
March 30, 2022, Hamburg

[1]Department of Statistics, LMU Munich

Agenda

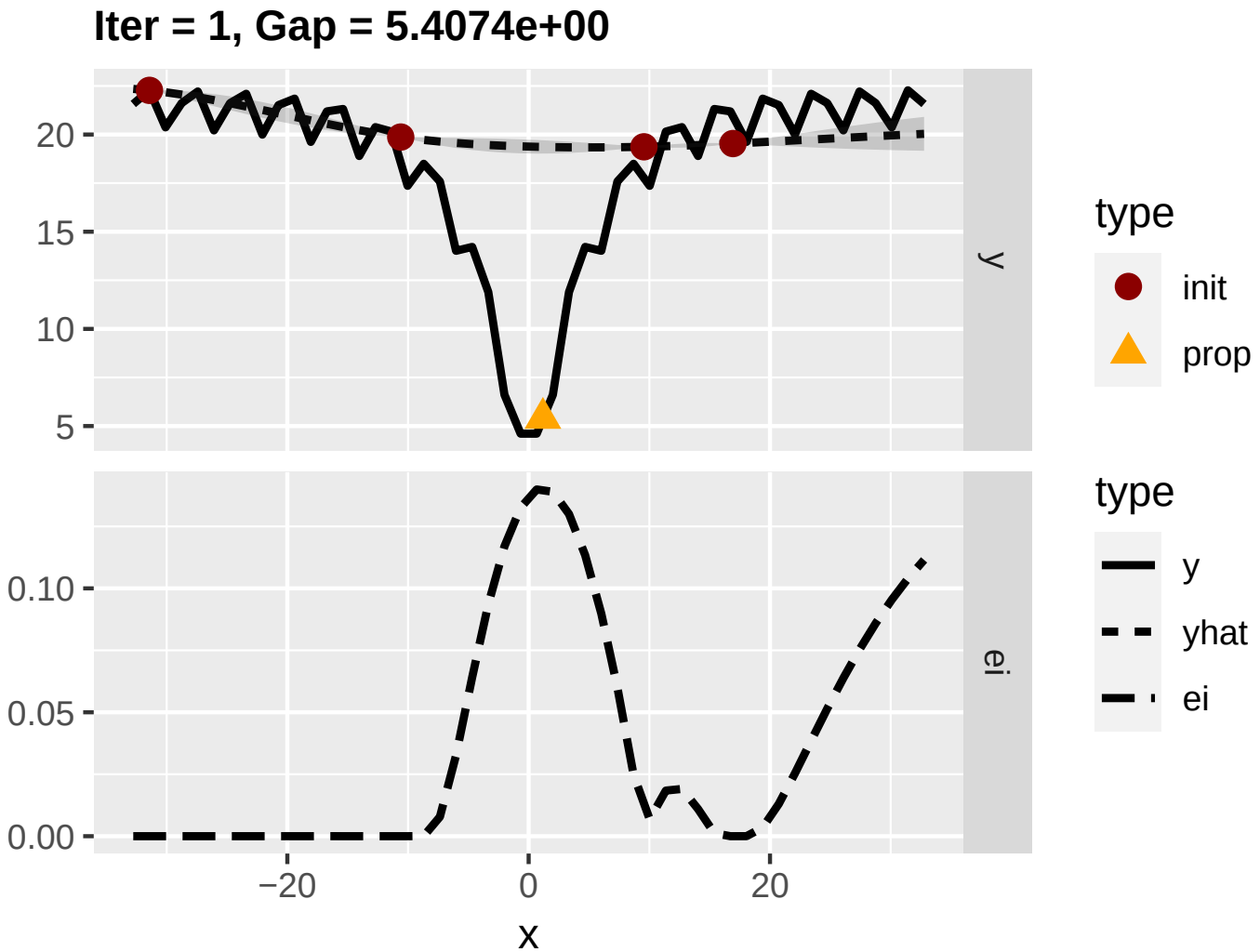
Agenda

Bayesian Optimization



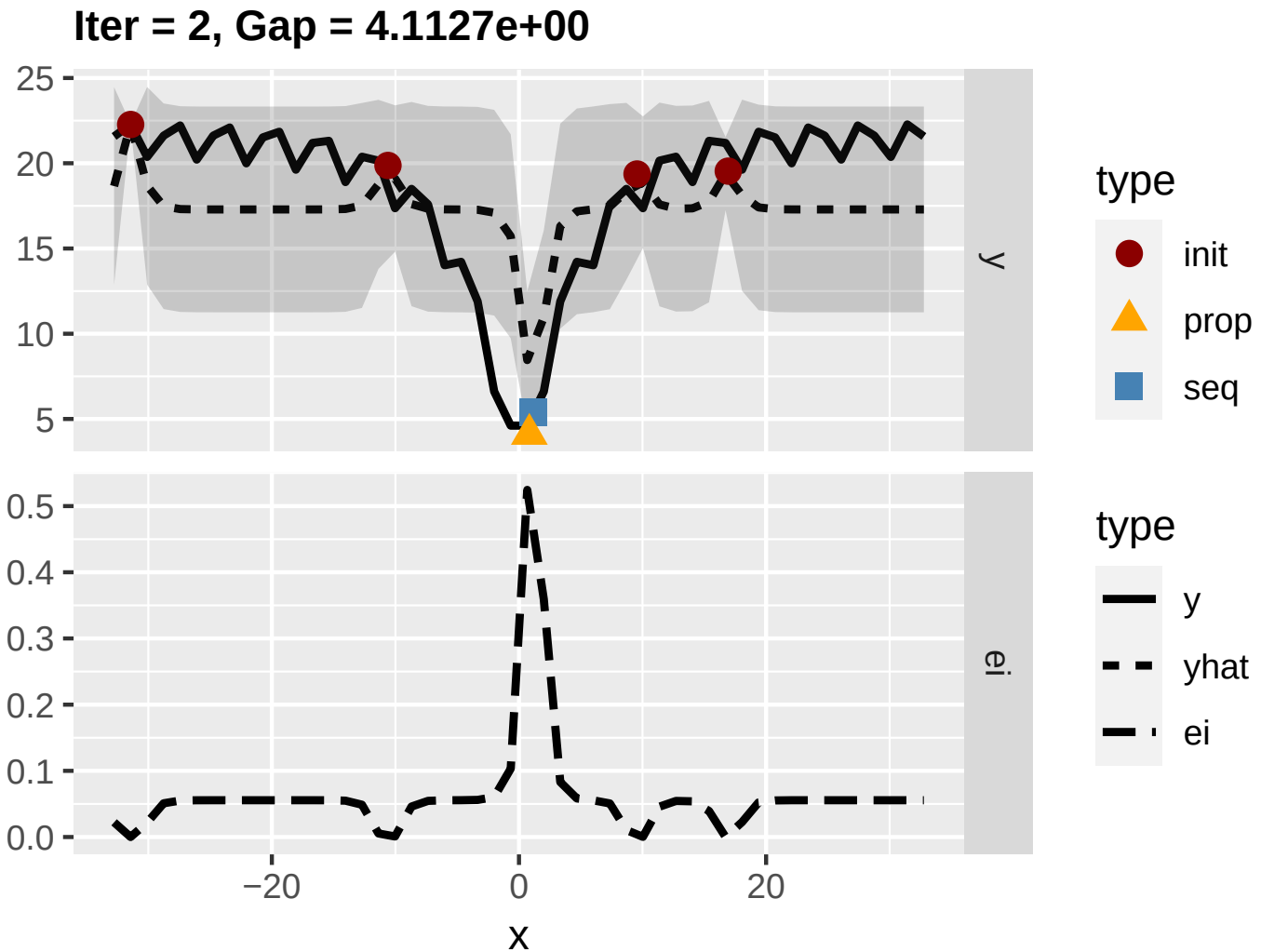
Note: If not otherwise stated, all figures are based on own computations using `ggplot2` [Wickham, 2016], `smoof` [Bossek, 2017] and `mlr(3)MBO` [Bischl et al., 2017]

Bayesian Optimization



Iteration 1: Surrogate Model (top) and Acquisition Function (bottom)

Bayesian Optimization



Iteration 2

Bayesian Optimization

Iteration 3

Bayesian Optimization

Iteration 4

Bayesian Optimization

Iteration 5

Bayesian Optimization

Iteration 6

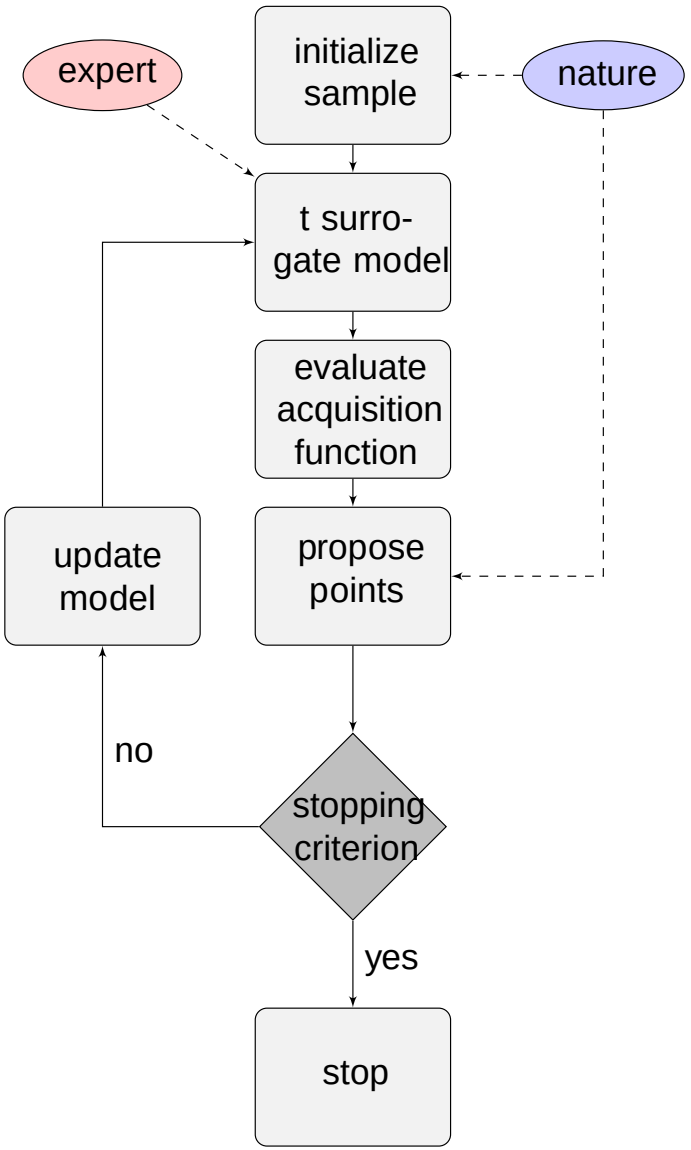
Bayesian Optimization

Iteration 7

Bayesian Optimization

Iteration 10

Bayesian Optimization

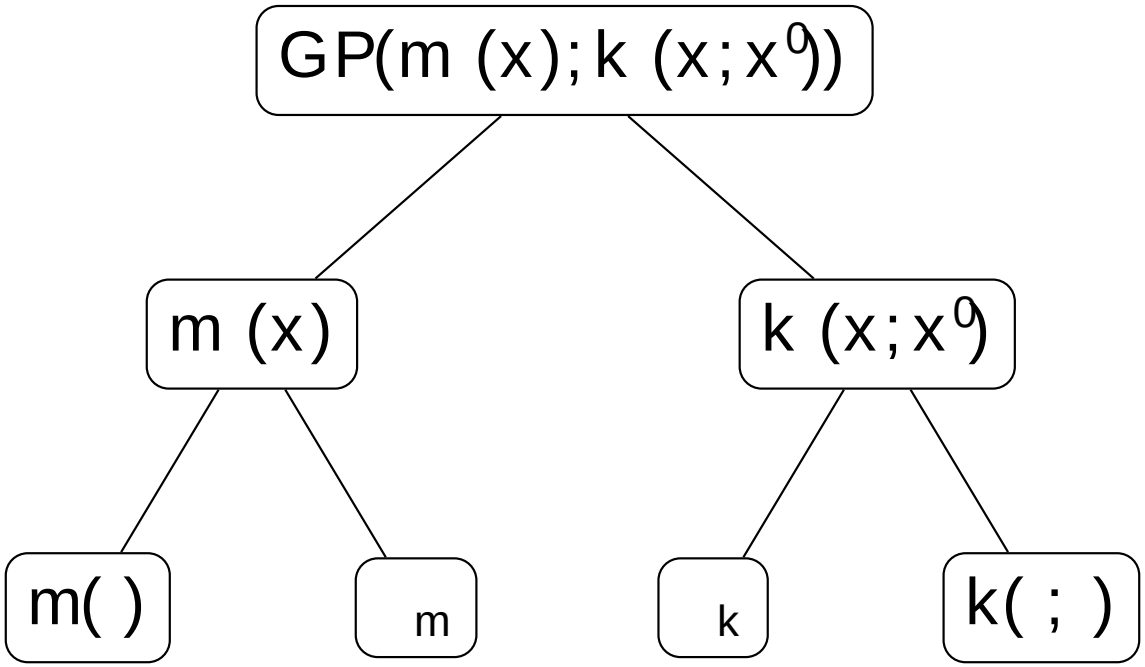


Agenda

Gaussian Processes - Intuition

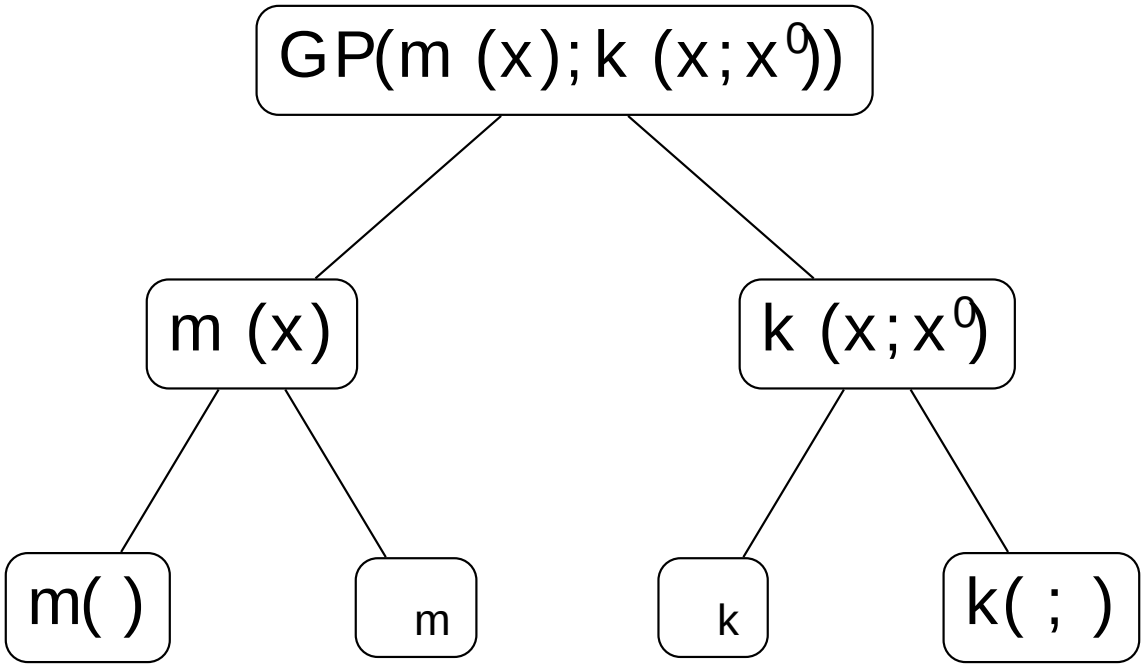
Functional GP regression: Three functions drawn from prior (a) and posterior (b) GP. Image credits: [Rasmussen, 2003].

Gaussian Processes { Prior Components



How to specify $m(\,)$, $_m$, $_k$ and $k(\,;\,)$
in absence of prior knowledge?

Gaussian Processes { Prior Components



And: Do they even matter?

Agenda

Agenda

Setup

We randomly select 50 synthetic test functions from the R package smoof [Bossek, 2017], stratified across the covariate space dimensions 2, 3, 4 and 7.

For each of them, a sensitivity analysis is conducted with regard to each of the four prior components.

- 5 functional forms
- 5 mean and kernel parameter specifications (relative deviation from global mean)
- we control for interaction effects

The initial design of size $n_{init} = 10$ is randomly sampled anew for each of the $R = 40$ BO repetitions with $T = 20$ iterations each.

Agenda

Results

Mean parameters in uence convergence the most, followed by the kernel's functional form .

Mean functional form and Kernel parameters play a (relatively) negligible role.

Agenda

Agenda

Prior near-ignorance models

Idea: Use set of$_m$ instead of precise$_m$. Fully specify the other components.

[Mangili, 2015] proposes imprecise Gaussian processes

$$GP\ \ Mh; k\ (x; x^0) + \frac{1 + M}{c}\ \ : h = \ \ 1; M\ \ \ 0\ \ ;$$

given a base kernel $k(x; x^0)$ and a degree of imprecision $c > 0$.

- ! results in a set of posteriors whose upper and lower mean estimates $\underline{\hat{}}(x)_c$, $\overline{\hat{}}(x)_c$ can be derived

Note: See [Benavoli and Za alon, 2015] for more on imprecise Gaussian processes and prior near-ignorance.

Agenda

Generalized Lower Confidence Bound (GLCB)

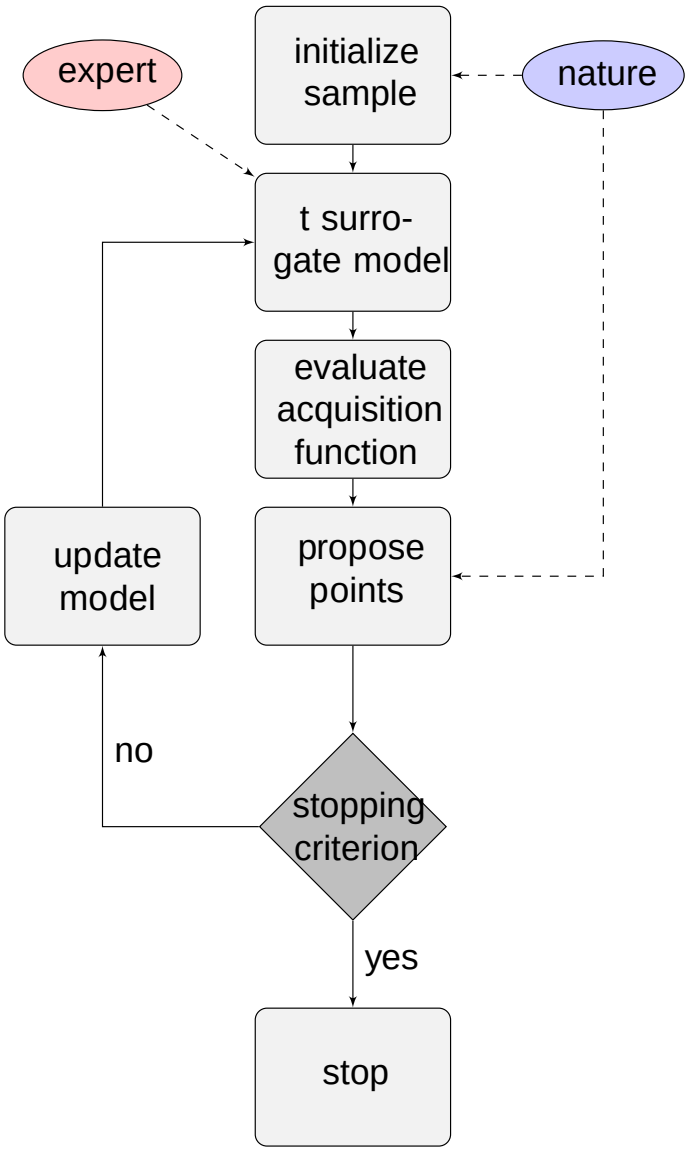
$$\mathrm{LCB}(x) = b(x) + \underbrace{\sqrt{\mathrm{Var}(b(x))}}_{\text{"classical" uncertainty}}$$

$$\mathrm{GLCB}(x) = b(x) + \underbrace{\sqrt{\mathrm{Var}(b(x))}}_{\text{"classical" uncertainty}} + \underbrace{(\bar{b}(x)_c - \underline{b}(x)_c)}_{\text{prior-induced imprecision}}$$

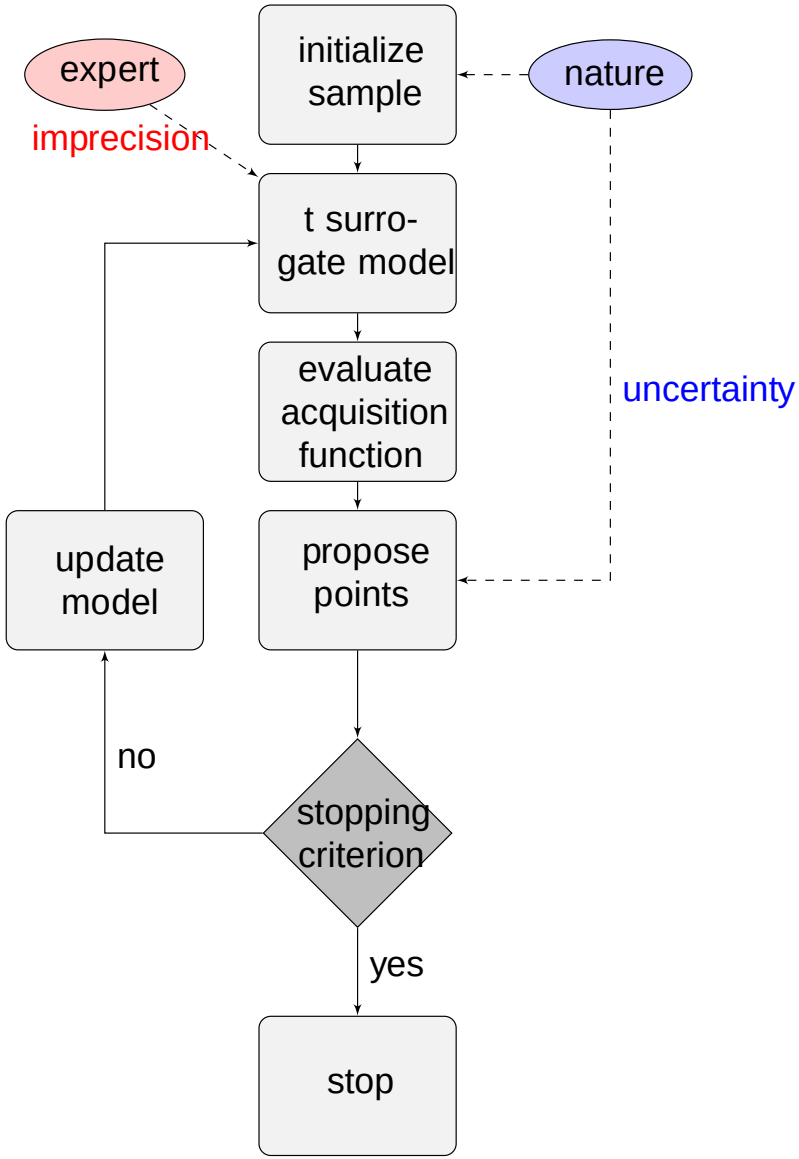
is the degree of risk-aversion

is the degree of ambiguity-aversion

Bayesian Optimization



Bayesian Optimization



Generalized Lower Con dence Bound (GLCB)

Notably, $\overline{\hat{\kappa}}(x) \quad \underline{\hat{\kappa}}(x)$ simpli es to an expression only dependent on predictive kernels $k_x = [k(x; x_1); :::; k(x; x_n)]^T$, the base kernel matrix K_n (from training) and the degree of imprecision c. For some values of c (depending on observations):[1]

$$\overline{\hat{\kappa}}(x) \quad \underline{\hat{\kappa}}(x) = (1 \quad k_x^T s_k) \frac{s_k^T}{S_k} y + \frac{c}{S_k} \quad \frac{s_k^T y}{c + S_k} \tag{1}$$

[1] For a thorough case distinction, please refer to the Appendix.

Generalized Lower Confidence Bound (GLCB)

For sufficiently high c, the model imprecision $\overline{\hat{\eta}}(x) \quad \underline{\hat{\eta}}(x)$ even simplifies further:

$$\overline{\hat{\eta}}(x) \quad \underline{\hat{\eta}}(x) = 2c\frac{|1 \quad k_x^T s_k|}{S_k} \tag{2}$$

In this case, GLCB's hyperparameters α and c collapse to one.

Agenda

Application in Material Science

Experimental set-up of graphene production: "The preparation of a samp
to be irradiated requires about one week." [Kottho , 2019]

Image credits: Lars Kottho , University of Wyoming

GLCB { Results

BO with GLCB on Graphene function. GLCB-1-50 means GLCB with = 1, c = 50. Data source: [Wahab et al., 2020].

Agenda

Discussion

Limitations

- robust only with regard to possible misspeci cation of the mean function parameter given a constant trend
- how to specify c?

Venues for future work

locally

- multivariate extensions
- Can we ensure $\left|\frac{s_k y}{S_k}\right| \leq 1 + \frac{c}{S_k}$ such that hyperparameters and collapse to one?

globally

- Imprecise probabilities o er vivid framework to represent ignorance in surrogate-assisted derivative-free optimization

Discussion

Thanks a lot for your attention!

Feel free to try out PROBO yourself:
https://github.com/rodemann/gp-imprecision-in-bo

We are looking forward to your feedback and comments of an
kind!

PROBO: Literature

Rodemann, J.: Robust Generalizations of Stochastic Derivative-Free Optimization. Master's thesis, LMU Munich (2021) [1]

Rodemann, J., Augustin, T.: Accounting for Gaussian Process Imprecision in Bayesian Optimization. In: Honda, K., Entani, T., Ubukata, S., Huynh, V.N., Inuiguchi, M. (eds.) IUKM. Springer Lecture Notes in Computer Science (LNCS). pp. 92{104. Springer, Cham (2022)

[1] https://epub.ub.uni-muenchen.de/77441/1/MA_Rodemann.pdf

Agenda

Literature I


Benavoli, A. and Zaffalon, M. (2015).
Prior near ignorance for inferences in the k-parameter exponential family.
Statistics, 49(5):1104–1140.


Bischl, B., Richter, J., Bossek, J., Horn, D., Thomas, J., and Lang, M. (2017).
mlrmbo: A modular framework for model-based optimization of expensive black-box functions.
arXiv preprint arXiv:1703.03373.


Bossek, J. (2017).
smoof: Single- and multi-objective optimization test functions.
The R Journal.

Literature II


Kotthoff, L. (2019).
Ai for materials science: Tuning laser-induced graphene production and beyond.


Mangili, F. (2015).
A prior near-ignorance Gaussian process model for nonparametric regression.
In *ISIPTA '15: Proceedings of the 9th International Symposium on Imprecise Probability: Theories and Applications*, pages 187–196.


Rasmussen, C. E. (2003).
Gaussian processes in machine learning.
In *Summer school on machine learning*, pages 63–71. Springer.

Literature III

Wahab, H., Jain, V., Tyrrell, A. S., Seas, M. A., Kotthoff, L., and Johnson, P. A. (2020).
Machine-learning-assisted fabrication: Bayesian optimization of laser-induced graphene patterning using in-situ raman analysis.
Carbon, 167:609–619.

Wickham, H. (2016).
ggplot2: Elegant Graphics for Data Analysis.
Springer-Verlag New York.

Agenda

Mean Optimization Path

Definition (Mean Optimization Path)

Given R repetitions of Bayesian optimization applied on a test function $\Psi(x)$ with T iterations each, let $\Psi(x^{\star})_{r;t}$ be the best incumbent target value at iteration $t \in \{1, \ldots, T\}$ from repetition $r \in \{1, \ldots, R\}$. The elements

$$MOP_t = \frac{1}{R} \sum_{r=1}^{R} \Psi(x^{\star})_{r;t}$$

shall then constitute the T-dimensional vector MOP, which we call *mean optimization path (MOP)* henceforth.

Definition (Accumulated Difference of MOPs)

Consider an experiment comparing S different prior specifications on a test function with R repetitions per specification and T iterations per repetition. Let the results be stored in a $T \quad S$-matrix of mean optimization paths for iterations $t \ 2 \ f1; :::; T g$ and prior specification $s \ 2 \ f1; :::; S g$ (e.g. constant, linear, quadratic etc. trend as mean functional form) with entries $MOP_{t;s} = \frac{1}{R} \quad \sum_{r=1}^{R} \Psi(x \ )_{r;t;s}$: The *accumulated di erence (AD)* for this experiment shall then be:

$$AD = \sum_{t=1}^{T} \max_{s} MOP_{t;s} \quad \min_{s} MOP_{t;s} \quad :$$

Results

Mean functional form	Kernel functional form	Mean parameters	Kernel parameters
42.49	68.20	77.91	11.40

Table: Sum of relative ADs of all 50 MOPs per prior specification. Comparisons between mean and kernel are more valid than between functional form and parameters.

Upper and lower mean estimates

In order to derive upper and lower bounds for the mean estimate, let $k\ (x; x^0)$ be a kernel function as defined in [Rasmussen, 2003]. The finitely positive semi-definite matrix K_n is then formed by applying $k\ (x; x^0)$ on the training data vector $x\ 2\ X$:

$$K_n = [k\ (x_i; x_j^0)]_{ij}: \tag{3}$$

Let x be a scalar input of test data, whose $f(x)$ is to be predicted. Then $k_x = [k\ (x; x_1); :::; k\ (x; x_n)]^T$ is the vector of covariances between x and the training data. Furthermore, name the training target vector y and define $s_k = K_n{}^1 1_n$ as well as $S_k = 1_n^T K_n{}^1 1_n$.

Upper and lower mean estimates

Then [Mangili, 2015] shows that if $j\frac{s_k y}{S_k} j \quad 1 + \frac{c}{S_k}$:

$$\bar{\hat{}}(x) = k_x^T K_n{}^1 y + (1 \quad k_x^T s_k)\frac{s_k^T}{S_k} y + c\frac{j1 \quad k_x^T s_k j}{S_k} \tag{4}$$

$$\underline{\hat{}}(x) = k_x^T K_n{}^1 y + (1 \quad k_x^T s_k)\frac{s_k^T}{S_k} y \quad c\frac{j1 \quad k_x^T s_k j}{S_k} \tag{5}$$

Upper and lower mean estimates

If $j\frac{s_k y}{S_k} j > 1 + \frac{c}{S_k}$:

$$\overline{\hat{}}(x) = k_x^T K_n{}^1 y + (1 \quad k_x^T s_k)\frac{s_k^T}{S_k} y + c\frac{1 \quad k_x^T s_k}{S_k} \tag{6}$$

$$\underline{\hat{}}(x) = k_x^T K_n{}^1 y + (1 \quad k_x^T s_k)\frac{s_k^T y}{c + S_k} \tag{7}$$