Bayesian Optimization

Bayesian optimization is a methodology for optimizing expensive objective functions that has proven success in the sciences, engineering, and beyond. This timely text provides a self-contained and comprehensive introduction to the subject, starting from scratch and carefully developing all the key ideas along the way. This bottom-up approach illuminates unifying themes in the design of Bayesian optimization algorithms and builds a solid theoretical foundation for approaching novel situations.

The core of the book is divided into three main parts, covering theoretical and practical aspects of Gaussian process modeling, the Bayesian approach to sequential decision making, and the realization and computation of practical and effective optimization policies.

Following this foundational material, the book provides an overview of theoretical convergence results, a survey of notable extensions, a comprehensive history of Bayesian optimization, and an extensive annotated bibliography of applications.

Roman Garnett is Associate Professor in Computer Science and Engineering at Washington University in St. Louis. He has been a leader in the Bayesian optimization community since 2011, when he cofounded a long-running workshop on the subject at the NeurIPS conference. His research focus is developing Bayesian methods – including Bayesian optimization – for automating scientific discovery.

ROMAN GARNETT
Washington University in St Louis

BAYESIAN OPTIMIZATION

CAMBRIDGE
UNIVERSITY PRESS

Shaftesbury Road, Cambridge CB2 8EA, United Kingdom

One Liberty Plaza, 20th Floor, New York, NY 10006, USA

477 Williamstown Road, Port Melbourne, VIC 3207, Australia

314–321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi – 110025, India

103 Penang Road, #05–06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of Cambridge University Press & Assessment, a department of the University of Cambridge.

We share the University's mission to contribute to society through the pursuit of education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781108425780

DOI: 10.1017/9781108348973

First published 2023

Printed in the United Kingdom by TJ Books Limited, Padstow Cornwall

A catalogue record for this publication is available from the British Library.

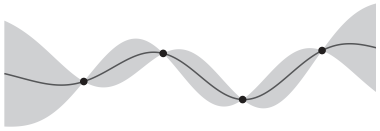
ISBN 978-1-108-42578-0 Hardback

CONTENTS

PREFACE

My interest in Bayesian optimization began in 2007 at the start of my doctoral studies. I was frustrated that there seemed to be a Bayesian approach to every task I cared about, *except* optimization. Of course, as was often the case at that time (not to mention now!), I was mistaken in this belief, but one should never let ignorance impede inspiration.

Meanwhile, my labmate and soon-to-be frequent collaborator Mike Osborne had a fresh copy of RASMUSSEN and WILLIAMS's *Gaussian Processes for Machine Learning* and just would *not* stop talking about GPs at our lab meetings. Through sheer brute force of repetition, I slowly built a hand-wavy intuition for Gaussian processes – my mental model was the "sausage plot" – without even being sure about their precise definition. However, I was pretty sure that marginals were Gaussian (what else?), and one day it occurred to me that one could achieve Bayesian optimization by maximizing the probability of improvement. This was the algorithm I was looking for! In my excitement I shot off an email to Mike that kicked off years of fruitful collaboration:



The first of many "sausage plots" to come.

> Can I ask a dumb question about GPs? Let's say that I'm doing function approximation on an interval with a GP. So I've got this mean function $m(x)$ and a variance function $v(x)$. Is it true that if I pick a particular point x, then $p\big(f(x)\big) \sim \mathcal{N}\big(m(x), v(x)\big)$? Please say yes.
>
> If this is true, then I think the idea of doing Bayesian optimization using GPs is, dare I say, trivial.

The hubris of youth!

Well, it turned out I was 45 years too late in proposing this algorithm,[1] and that it only seemed "trivial" because I had no appreciation for its theoretical foundation. However, truly great ideas are rediscovered many times, and my excitement did not fade. Once I developed a deeper understanding of Gaussian processes and Bayesian decision theory, I came to see them as a "Bayesian crank" I could turn to realize adaptive algorithms for *any* task. I have been repeatedly astonished to find that the resulting algorithms – seemingly by magic – *automatically* display intuitive emergent behavior as a result of their careful design. My goal with this book is to paint this grand picture. In effect, it is a gift to my former self: the book I wish I had in the early years of my career.

1 H. J. KUSHNER (1962). A Versatile Stochastic Model of a Function of Unknown and Time Varying Form. *Journal of Mathematical Analysis and Applications* 5(1):150–167.

In the context of machine learning, Bayesian optimization is an ancient idea – KUSHNER's paper appeared only three years after the term "machine learning" was coined! Despite its advanced age, Bayesian optimization has been enjoying a period of revitalization and rapid progress over the past ten years. The primary driver of this renaissance has been advances in computation, which have enabled increasingly sophisticated tools for Bayesian modeling and inference.

Ironically, however, perhaps the most critical development was not Bayesian at all, but the rise of deep neural networks, another old idea

granted new life by modern computation. The extreme cost of training these models demands efficient routines for hyperparameter tuning, and in a timely and influential paper, SNOEK et al. demonstrated (dramatically!) that Bayesian optimization was up to the task.[2] Hyperparameter tuning proved to be a "killer app" for Bayesian optimization, and the ensuing surge of interest has yielded a mountain of publications developing new algorithms and improving old ones, exploring countless variations on the basic setup, establishing theoretical guarantees on performance, and applying the framework to a huge range of domains.

2 J. SNOEK et al. (2012). Practical Bayesian Optimization of Machine Learning Algorithms. *NEURIPS 2012*.

Due to the nature of the computer science publication model, these recent developments are scattered across dozens of brief papers, and the pressure to establish novelty in a limited space can obscure the big picture in favor of minute details. This book aims to provide a self-contained and comprehensive introduction to Bayesian optimization, starting "from scratch" and carefully developing all the key ideas along the way. This bottom-up approach allows us to identify unifying themes in Bayesian optimization algorithms that may be lost when surveying the literature.

intended audience

The intended audience is graduate students and researchers in machine learning, statistics, and related fields. However, it is also my sincere hope that practitioners from more distant fields wishing to harness the power of Bayesian optimization will also find some utility here.

prerequisites

For the bulk of the text, I assume the reader is comfortable with differential and integral calculus, probability, and linear algebra. On occasion the discussion will meander to more esoteric areas of mathematics, and these passages can be safely ignored and returned to later if desired. A good working knowledge of the Gaussian distribution is also essential, and I provide an abbreviated but sufficient introduction in Appendix A.

Chapters 2–4: modeling the objective function with Gaussian processes

The book is divided into three main parts. Chapters 2–4 cover theoretical and practical aspects of modeling with Gaussian processes. This class of models is the overwhelming favorite in the Bayesian optimization literature, and the material contained within is critical for several following chapters. It was daunting to write this material in light of the many excellent references already available, in particular the aforementioned *Gaussian Processes for Machine Learning*. However, I heavily biased the presentation in light of the needs of optimization, and even experts may find something new.

Chapters 5–7: sequential decision making and policy building

Chapters 5–7 develop the theory of sequential decision making and its application to optimization. Although this theory requires a model of the objective function and our observations of it, the presentation is agnostic to the choice of model and may be read independently from the preceding chapters on Gaussian processes.

Chapters 8–10: Bayesian optimization with Gaussian processes

These threads are unified in Chapters 8–10, which discuss the particulars of Bayesian optimization with Gaussian process models. Chapters 8–9 cover details of computation and implementation, and Chapter 10 discusses theoretical performance bounds on Bayesian optimization algorithms, where most results depend intimately on a Gaussian process model of the objective function or the associated reproducing kernel Hilbert space.

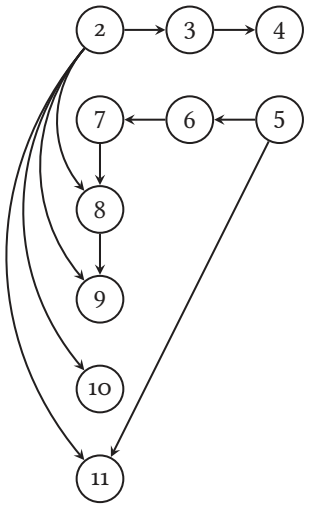
The nuances of some applications require modifications to the basic sequential optimization scheme that is the focus of the bulk of the book, and Chapter 11 introduces several notable extensions to this basic setup. Each is systematically presented through the unifying lens of Bayesian decision theory to illustrate how one might proceed when facing a novel situation.

Chapter 11: extensions

Finally, Chapter 12 provides a brief and standalone history of Bayesian optimization. This was perhaps the most fun chapter for me to write, if only because it forced me to plod through old Soviet literature (in an actual library! What a novelty these days!). To my surprise I was able to antedate many Bayesian optimization policies beyond their commonly attested origin, including expected improvement, knowledge gradient, probability of improvement, and upper confidence bound. (A reader familiar with the literature may be surprised to learn the last of these was actually the first policy discussed by KUSHNER in his 1962 paper.) Despite my best efforts, there may still be stones left to be overturned before the complete history is revealed.

Chapter 12: brief history of Bayesian optimization

Dependencies between the main chapters are illustrated in the margin. There are two natural linearizations of the material. The first is the one I adopted and personally prefer, which covers modeling prior to decision making. However, one could also proceed in the other order, reading Chapters 5–7 first, then looping back to Chapter 2. After covering the material in these chapters (in either order), the remainder of the book can be perused at will. Logical partial paths through the book include:



A dependency graph for Chapters 2–11. Chapter 1 is a universal dependency.

- a minimal but self-contained introduction: Chapters 1–2, 5–7
- a shorter introduction requiring leaps of faith: Chapters 1 and 7
- a crash course on the underlying theory: Chapters 1–2, 5–7, 10
- a head start on implementing a software package: Chapters 1–9

A reader already quite comfortable with Gaussian processes might wish to skip over Chapters 2–4 entirely.

I struggled for some time over whether to include a chapter on applications. On the one hand, Bayesian optimization ultimately owes its popularity to its success in optimizing a growing and diverse set of difficult objectives. However, these applications often require extensive technical background to appreciate, and an adequate coverage would be tedious to write and tedious to read. As a compromise, I provide an annotated bibliography outlining the optimization challenges involved in notable domains of interest and pointing to studies where these challenges were successfully overcome with the aid of Bayesian optimization.

Annotated Bibliography of Applications: Appendix D, p. 313

The sheer size of the Bayesian optimization literature – especially the output of the previous decade – makes it impossible to provide a complete survey of every recent development. This is especially true for the extensions discussed in Chapter 11 and even more so for the bibliography on applications, where work has proliferated in myriad branching directions. Instead I settled for presenting what I considered

to be the most important ideas and providing pointers to entry points for the relevant literature. The reader should not read anything into any omissions; there is simply too much high-quality work to go around.

Additional information about the book, including a list of errata as they are discovered, may be found at the companion webpage:

`bayesoptbook.com`

I encourage the reader to report any errata or other issues to the companion GitHub repository for discussion and resolution:

`github.com/bayesoptbook/bayesoptbook.github.io`

Thank you!

Preparation of this manuscript was facilitated tremendously by numerous free and open source projects, and the creators, developers, and maintainers of these projects have my sincere gratitude. The manuscript was typeset in LaTeX using the excellent and extremely flexible memoir class. The typeface is Linux Libertine. Figures were laid out in MATLAB and converted to Ti*k*Z/PGF/PGFPLOTS for further tweaking and typesetting via the `matlab2tikz` script. The colors used in figures were based on `www.colorbrewer.org` by Cynthia A. Brewer, and I endeavored to the best of my ability to ensure that the figures are colorblind friendly. The colormap used in heat maps is a slight modification of the Matplotlib viridis colormap where the "bright" end is pure white.

I would like to thank Eric Brochu, Nando de Freitas, Matt Hoffman, Frank Hutter, Mike Osborne, Bobak Shahriari, Jasper Snoek, Kevin Swersky, and Ziyu Wang, who jointly provided the activation energy for this undertaking. I would also like to thank Eytan Bakshy, Ivan Barrientos, George De Ath, Neil Dhir, Peter Frazier, Lukas Fröhlich, Ashok Gautam, Jake Gardner, Javier González, Ryan-Rhys Griffiths, Philipp Hennig, Eugen Hotaj, Jungtaek Kim, Simon Kruse, Jack Liu, Bryan Low, Ruben Martinez-Cantin, Keita Mori, Kevin Murphy, Matthias Poloczeck, Jon Scarlett, Sebastian Tay, Sattar Vakili, Jiangyan Zhao, Qiuyi Zhang, Xiaowei Zhang, and GitHub users `cgoble001` and `chaos-and-patterns` for their suggestions, corrections, and valuable discussions along the way, as well as everyone at Cambridge University Press for their support and patience as I continually missed deadlines. Finally, special thanks are due to the students of two seminars run at Washington University reading, discussing, and ultimately improving the book.

Funding support was provided by the United States National Science Foundation (NSF) under award number 1845434. Any opinions, findings, and conclusions or recommendations expressed in this book are those of the author and do not necessarily reflect the views of the NSF.

This book took far more time than I initially anticipated, and I would especially like to thank my wife Marion, my son Max (arg Max?), and my daughter Matilda (who escaped being named Minnie!) for their understanding and support during this long journey.

Roman Garnett
St. Louis, Missouri, November 2022

NOTATION

vectors and matrices

All vectors are column vectors and are denoted in lowercase bold: $\mathbf{x} \in \mathbb{R}^d$. Matrices are denoted in uppercase bold: $\mathbf{A}$.

matrix calculus convention

We adopt the "numerator layout" convention for matrix calculus: the derivative of a vector by a scalar is a (column) vector, whereas the derivative of a scalar by a vector is a row vector. This results in the chain rule proceeding from left-to-right; for example, if a vector $\mathbf{x}(\theta)$ depends on a scalar parameter θ, then for a function $f(\mathbf{x})$, we have:

chain rule

$$\frac{\partial f}{\partial \theta} = \frac{\partial f}{\partial \mathbf{x}} \frac{\partial \mathbf{x}}{\partial \theta}.$$

indicator functions

When an indicator function is required, we use the Iverson bracket notation. For a statement s, we have:

$$[s] = \begin{cases} 1 & \text{if } s \text{ is true;} \\ 0 & \text{otherwise.} \end{cases}$$

The statement may depend on a parameter: $[x \in A]$, $[x \geq 0]$, etc.

logarithms
nats

Logarithms are taken with respect to their natural base, e. Quantities in log units such as log likelihoods or entropy thus have units of *nats*, the base-e analogue of the more familiar base-2 bits.

SYMBOLS WITH IMPLICIT DEPENDENCE ON LOCATION

There is one notational innovation in this book compared with the Gaussian process and Bayesian optimization literature at large: we make heavy use of symbols for quantities that depend *implicitly* on a putative (and arbitrary) input location x. Most importantly, to refer to the value of an objective function f at a given location x, we introduce the symbol $\phi = f(x)$. This avoids clash with the name of the function itself, f, while avoiding an extra layer of brackets. We use this scheme throughout the book, including variations such as:

$$\phi' = f(x'); \qquad \boldsymbol{\phi} = f(\mathbf{x}); \qquad \gamma = g(x); \qquad \text{etc.}$$

To refer to the outcome of a (possibly inexact) measurement at x, we use the symbol y; the distribution of y presumably depends on ϕ.

We also allocate symbols to describe properties of the marginal predictive distributions for the objective function value ϕ and observed value y, all of which also have implicit dependence on x. These appear in the following table.

COMPREHENSIVE LIST OF SYMBOLS

A list of important symbols appears on the following pages, arranged roughly in alphabetical order.

symbol	description
$\equiv$	identical equality of functions; for a constant c, $f \equiv c$ is a constant function
∇	gradient operator
$\varnothing$	termination option: the action of immediately terminating optimization
$\prec$	either Pareto dominance or the Löwner order: for symmetric $\mathbf{A}, \mathbf{B}$, $\mathbf{A} \prec \mathbf{B}$ if and only if $\mathbf{B} - \mathbf{A}$ is positive definite
$\omega \sim p(\omega)$	is sampled according to: ω is a realization of a random variable with probability density $p(\omega)$
$\bigsqcup_i \mathcal{X}_i$	disjoint union of $\{\mathcal{X}_i\}$: $\bigsqcup_i \mathcal{X}_i = \bigcup_i \left\{(x, i) \mid x \in \mathcal{X}_i\right\}$
$\lvert\mathbf{A}\rvert$	determinant of square matrix $\mathbf{A}$
$\lvert\mathbf{x}\rvert$	Euclidean norm of vector $\mathbf{x}$; $\lvert\mathbf{x} - \mathbf{y}\rvert$ is thus the Euclidean distance between vectors $\mathbf{x}$ and $\mathbf{y}$
$\lVert f\rVert_{\mathcal{H}_K}$	norm of function f in reproducing kernel Hilbert space $\mathcal{H}_K$
$\mathbf{A}^{-1}$	inverse of square matrix $\mathbf{A}$
$\mathbf{x}^\top$	transpose of vector $\mathbf{x}$
$\mathbf{0}$	vector or matrix of zeros
$\mathcal{A}$	action space for a decision
$\alpha(x; \mathcal{D})$	acquisition function evaluating x given data $\mathcal{D}$
$\alpha_\tau(x; \mathcal{D})$	expected marginal gain in $u(\mathcal{D})$ after observing at x then making $\tau - 1$ additional optimal observations given the outcome
$\alpha^*_\tau(\mathcal{D})$	value of $\mathcal{D}$ with horizon τ: expected marginal gain in $u(\mathcal{D})$ from τ additional optimal observations
α_{EI}	expected improvement
α_{f^*}	mutual information between y and f^*
α_{KG}	knowledge gradient
α_{PI}	probability of improvement
α_{x^*}	mutual information between y and x^*
α_{UCB}	upper confidence bound
α_{TS}	Thompson sampling "acquisition function:" a draw $f \sim p(f \mid \mathcal{D})$
β	confidence parameter in Gaussian process upper confidence bound policy
$\beta(\mathbf{x}; \mathcal{D})$	batch acquisition function evaluating $\mathbf{x}$ given data $\mathcal{D}$; may have modifiers analogous to α
$\mathbf{C}$	prior covariance matrix of observed values $\mathbf{y}$: $\mathbf{C} = \operatorname{cov}[\mathbf{y}]$
$c(\mathcal{D})$	cost of acquiring data $\mathcal{D}$
$\operatorname{chol}\mathbf{A}$	Cholesky decomposition of positive definite matrix $\mathbf{A}$: if $\mathbf{\Lambda} = \operatorname{chol}\mathbf{A}$, then $\mathbf{A} = \mathbf{\Lambda}\mathbf{\Lambda}^\top$
$\operatorname{corr}[\omega, \psi]$	correlation of random variables ω and ψ; with a single argument, $\operatorname{corr}[\omega] = \operatorname{corr}[\omega, \omega]$
$\operatorname{cov}[\omega, \psi]$	covariance of random variables ω and ψ; with a single argument, $\operatorname{cov}[\omega] = \operatorname{cov}[\omega, \omega]$
$\mathcal{D}$	set of observed data, $\mathcal{D} = (\mathbf{x}, \mathbf{y})$
$\mathcal{D}', \mathcal{D}_1$	set of observed data after observing at x: $\mathcal{D}' = \mathcal{D} \cup \left\{(x, y)\right\} = (\mathbf{x}', \mathbf{y}')$
$\mathcal{D}_\tau$	set of observed data after τ observations
$D_{\text{KL}}[p \parallel q]$	Kullback–Leibler divergence between distributions with probability densities p and q
$\Delta(x, y)$	marginal gain in utility after acquiring observation (x, y): $\Delta(x, y) = u(\mathcal{D}') - u(\mathcal{D})$
$\delta(\omega - a)$	Dirac delta distribution on ω with point mass at a
$\operatorname{diag}\mathbf{x}$	diagonal matrix with diagonal $\mathbf{x}$
$\mathbb{E}, \mathbb{E}_\omega$	expectation, expectation with respect to ω
ε	measurement error associated with an observation at x: $\varepsilon = y - \phi$
f	objective function; $f \colon \mathcal{X} \to \mathbb{R}$
$f\lvert_{\mathcal{Y}}$	the restriction of f onto the subdomain $\mathcal{Y} \subset \mathcal{X}$
f^*	globally maximal value of the objective function: $f^* = \max f$
γ_τ	information capacity of an observation process given τ iterations

symbol	description
$\mathcal{GP}(f; \mu, K)$	Gaussian process on f with mean function μ and covariance function K
$\mathcal{H}_K$	reproducing kernel Hilbert space associated with kernel K
$\mathcal{H}_K[B]$	ball of radius B in $\mathcal{H}_K$: $\{f \mid \|f\|_{\mathcal{H}_K} \leq B\}$
$H[\omega]$	discrete or differential entropy of random variable ω
$H[\omega \mid \mathcal{D}]$	discrete or differential of random variable ω after conditioning on $\mathcal{D}$
$I(\omega; \psi)$	mutual information between random variables ω and ψ
$I(\omega; \psi \mid \mathcal{D})$	mutual information between random variables ω and ψ after conditioning on $\mathcal{D}$
$\mathbf{I}$	identity matrix
K	prior covariance function: $K = \operatorname{cov}[f]$
$K_{\mathcal{D}}$	posterior covariance function given data $\mathcal{D}$: $K_{\mathcal{D}} = \operatorname{cov}[f \mid \mathcal{D}]$
K_{M}	Matérn covariance function
K_{SE}	squared exponential covariance function
κ	cross-covariance between f and observed values $\mathbf{y}$: $\kappa(x) = \operatorname{cov}[\mathbf{y}, \phi \mid x]$
ℓ	either a length-scale parameter or the lookahead horizon
λ	output-scale parameter
$\mathcal{M}$	space of models indexed by the hyperparameter vector $\boldsymbol{\theta}$
$\mathbf{m}$	prior expected value of observed values $\mathbf{y}$, $\mathbf{m} = \mathbb{E}[\mathbf{y}]$
μ	either the prior mean function, $\mu = \mathbb{E}[f]$, or the predictive mean of ϕ: $\mu = \mathbb{E}[\phi \mid x, \mathcal{D}] = \mu_{\mathcal{D}}(x)$
$\mu_{\mathcal{D}}$	posterior mean function given data $\mathcal{D}$: $\mu_{\mathcal{D}} = \mathbb{E}[f \mid \mathcal{D}]$
$\mathcal{N}(\boldsymbol{\phi}; \boldsymbol{\mu}, \Sigma)$	multivariate normal distribution on $\boldsymbol{\phi}$ with mean vector $\boldsymbol{\mu}$ and covariance matrix Σ
$\mathbf{N}$	measurement error covariance corresponding to observed values $\mathbf{y}$
$\mathcal{O}$	is asymptotically bounded above by: for nonnegative functions f, g of τ, $f = \mathcal{O}(g)$ if f/g is asymptotically bounded by a constant as $\tau \to \infty$
$\mathcal{O}^*$	as above with logarithmic factors suppressed: $f = \mathcal{O}^*(g)$ if $f(\tau)(\log \tau)^k = \mathcal{O}(g)$ for some k
Ω	is asymptotically bounded below by: $f = \Omega(g)$ if $g = \mathcal{O}(f)$
p	probability density
q	either an approximation to probability density p or a quantile function
$\Phi(z)$	standard normal cumulative density function: $\Phi(z) = \int_{-\infty}^{z} \phi(z')\,\mathrm{d}z'$
ϕ	value of the objective function at x: $\phi = f(x)$
$\phi(z)$	standard normal probability density function: $\phi(z) = (\sqrt{2\pi})^{-1} \exp(-\frac{1}{2}z^2)$
$\Pr$	probability
$\mathbb{R}$	set of real numbers
R_τ	cumulative regret after τ iterations
$\bar{R}_\tau[B]$	worst-case cumulative regret after τ iterations on the RKHS ball $\mathcal{H}_K[B]$
r_τ	simple regret after τ iterations
$\bar{r}_\tau[B]$	worst-case simple regret after τ iterations on the RKHS ball $\mathcal{H}_K[B]$
$\mathbf{P}$	a correlation matrix
ρ	a scalar correlation
ρ_τ	instantaneous regret on iteration τ
s^2	predictive variance of y; for additive Gaussian noise, $s^2 = \operatorname{var}[y \mid x, \mathcal{D}] = \sigma^2 + \sigma_n^2$
Σ	a covariance matrix, usually the Gram matrix associated with $\mathbf{x}$: $\Sigma = K_{\mathcal{D}}(\mathbf{x}, \mathbf{x})$
σ^2	predictive variance of ϕ: $\sigma^2 = K_{\mathcal{D}}(x, x)$
σ_n^2	variance of measurement error at x: $\sigma_n^2 = \operatorname{var}[\varepsilon \mid x]$
$\operatorname{std}[\omega]$	standard deviation of random variable ω
$\mathcal{T}(\phi; \mu, \sigma^2, \nu)$	Student-t distribution on ϕ with ν degrees of freedom, mean μ, and variance σ^2
$\mathcal{TN}(\phi; \mu, \sigma^2, I)$	truncated normal distribution, $\mathcal{N}(\phi; \mu, \sigma^2)$ truncated to interval I

symbol	description
τ	either decision horizon (in the context of decision making) or number of optimization iterations passed (in the context of asymptotic analysis)
Θ	is asymptotically bounded above and below by: $f = \Theta(g)$ if $f = \mathcal{O}(g)$ and $f = \Omega(g)$
$\boldsymbol{\theta}$	vector of hyperparameters indexing a model space $\mathcal{M}$
$\operatorname{tr} \mathbf{A}$	trace of square matrix $\mathbf{A}$
$u(\mathcal{D})$	utility of data $\mathcal{D}$
$\operatorname{var}[\omega]$	variance of random variable ω
x	putative input location of the objective function
$\mathbf{x}$	either a sequence of observed locations $\mathbf{x} = \{x_i\}$ or (when the distinction is important) a vector-valued input location
x^*	a location attaining the globally maximal value of f: $x^* \in \arg\max f$; $f(x^*) = f^*$
$\mathcal{X}$	domain of objective function
y	value resulting from an observation at x
$\mathbf{y}$	observed values resulting from observations at locations $\mathbf{x}$
z	z-score of measurement y at x: $z = (y - \mu)/s$