BAYESIAN QUADRATURE: LESSONS LEARNED AND LOOKING FORWARD

Roman Garnett
Washington University in St. Louis

December 11, 2015

1. (BRIEF) INTRODUCTION

Bayesian quadrature

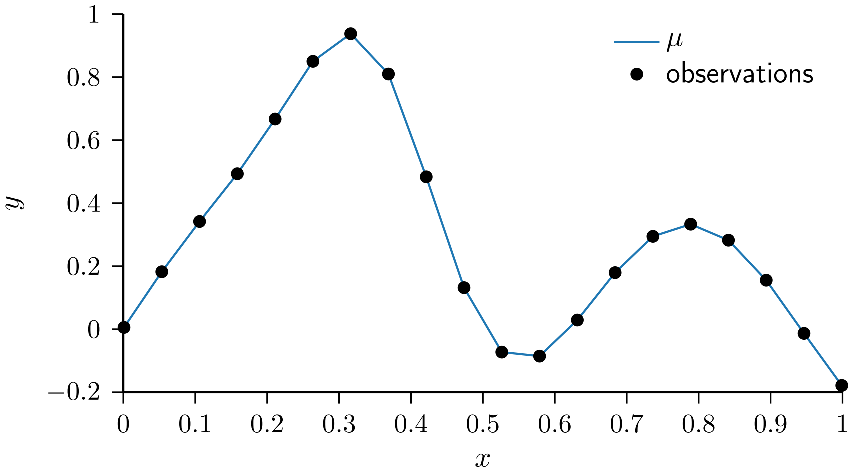
Bayesian quadrature: Introduction

(Thanks to Persi Diaconis!) Imagine trying to find the value of the following *definite integral*

$$\int_0^1 \exp\left(-\frac{(x-0.35)^2}{2(0.1)^2}\right) + \frac{\sin(10x)}{3}\,\mathrm{d}x.$$

...and you forgot most of calculus!

Trapezoid rule



Questions

Here the trapezoid rule gives

$$\int_0^1 f(x)\,\mathrm{d}x \approx 0.3104$$

(the true answer is ≈ 0.3119). Questions:

- *When* should I stop?
- *Where* should I measure the function?

A Bayesian approach

Let's try a *Bayesian* approach. Here we will treat the value of the integral

$$Z = \int_0^1 f(x)\,\mathrm{d}x$$

as a *random variable.* We will choose a prior for Z and use *Bayes' rule* to find the posterior distribution given our observations.

A Bayesian approach: Prior

It turns out that placing a prior on f rather than on Z directly is sometimes easier. A *Gaussian process* (GP) is a convenient choice:

$$p(f) = \mathcal{GP}(f; \mu, K).$$

Why? Because GPs are closed under affine transformations $L\colon f \mapsto L[f]$:

$$p\big(L[f]\big) = \mathcal{GP}\big(L[f]; L[\mu], L^2[K]\big).$$

A Bayesian approach: Quadrature

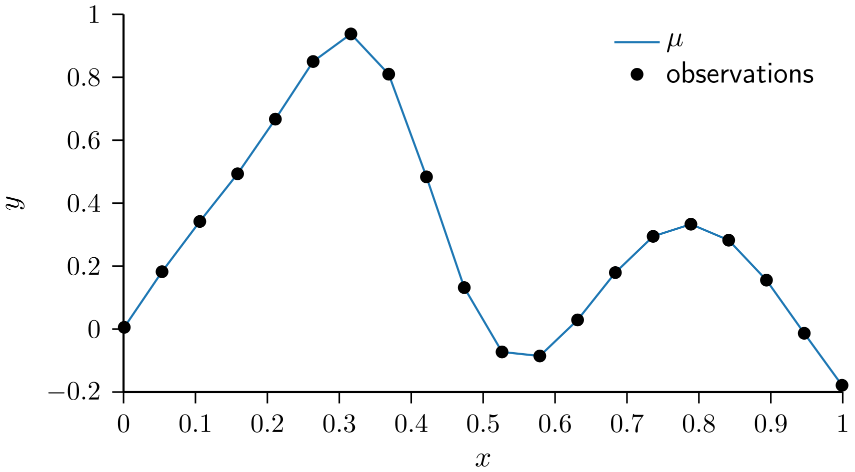
Why? Because *integration is a linear operator!*

$$p\left(\int_0^1 f(x)\,\mathrm{d}x\right) = \mathcal{N}\left(Z; \int_0^1 \mu(x)\,\mathrm{d}x, \int_0^1 \int_0^1 K(x, x')\,\mathrm{d}x\,\mathrm{d}x'\right).$$

A Bayesian approach: Example

Let's *revisit our example.* We choose *Brownian motion* as our GP prior for f, and estimate our integral by *integrating the posterior mean...*

Trapezoid rule



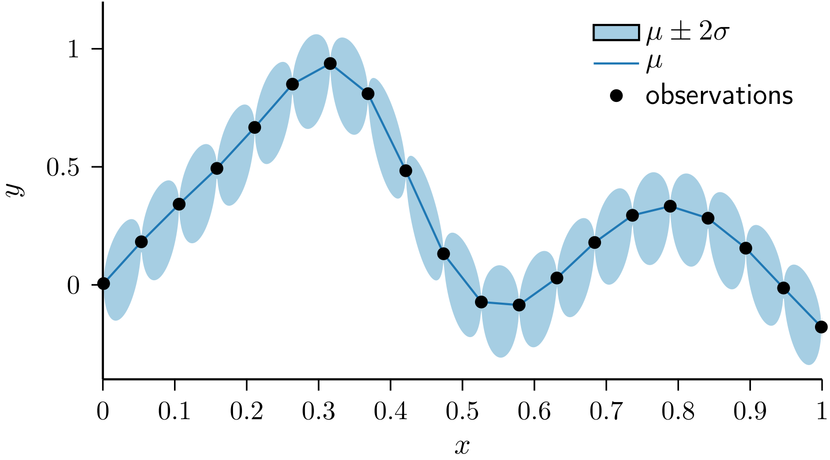
A Bayesian approach: Why??

- What's the point?
- We can also quantify our *uncertainty* in the integral!

$$\mathrm{var}[Z \mid \mathcal{D}] = \iint K_{f|\mathcal{D}} \, \mathrm{d}x \, \mathrm{d}x'$$

- This can help us answer the previous questions:
  - *When* should I stop?
  - *Where* should I measure the function?

Uncertainty!



When should I stop?

- The magnitude of the uncertainty in Z can help us decide *when* to stop.
- For this example, we have

$$\sqrt{\operatorname{var}[Z \mid \mathcal{D}]} \approx 0.015.$$

Where should I measure?

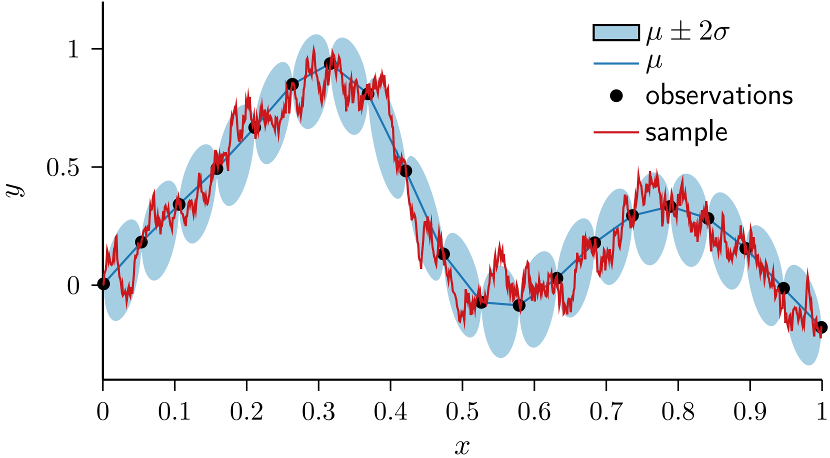
- The potential *reduction* in uncertainty in Z can help us decide *where* to measure.
- That is can compute the *value of information* by considering

$$V(x^*) = \sqrt{\mathrm{var}[Z \mid \mathcal{D}]} - \mathbb{E}_{y*}\left[\sqrt{\mathrm{var}\big[Z \mid \mathcal{D} \cup (x^*, y^*)\big]} \mid \mathcal{D}, x^*\right].$$

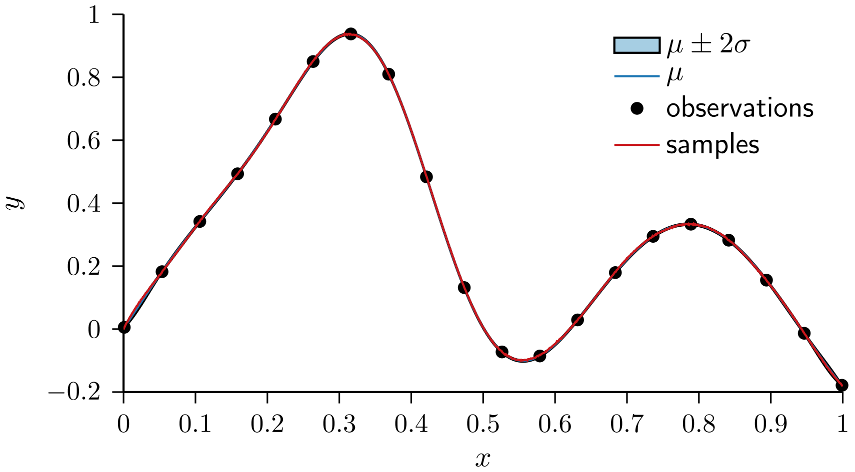
Remarkable fact

- The reduction in uncertainty *does not* depend on the observed value y^*!
- This allows us to compute *optimal quadrature rules* offline!

Another useful insight... (prior knowledge)



Another useful insight. . . (prior knowledge)



2. APPLICATIONS TO MACHINE LEARNING

Model Evidence, Robust Predictions

ML Integrals

What integrals do we need to estimate in *machine learning?*

- *Expectations:*

$$\mathbb{E}_p[f] = \int f(x)p(x)\,\mathrm{d}x$$

- (Special case) *Model evidence:*

$$Z = p(y \mid X, \mathcal{M}) = \int p(y \mid X, \theta, \mathcal{M})p(\theta \mid \mathcal{M})\,\mathrm{d}\theta$$

- *Predictions:*

$$p(y^* \mid x^*, \mathcal{D}) = \int p(y^* \mid x^*, \mathcal{D}, \theta)p(\theta \mid \mathcal{D})\,\mathrm{d}\theta$$

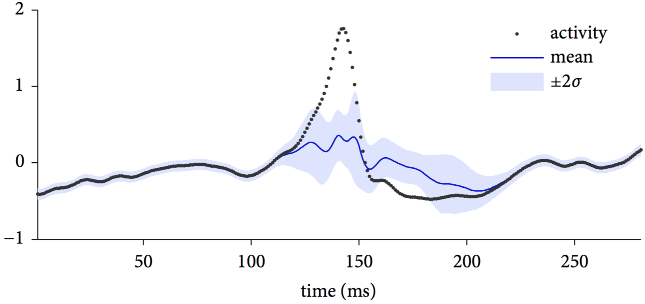
ML Integrals

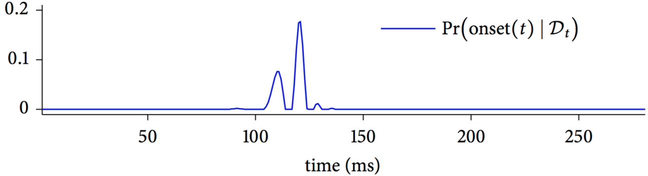
These problems are usually solved via *MCMC* or *MLE/MAP*.

Predictions

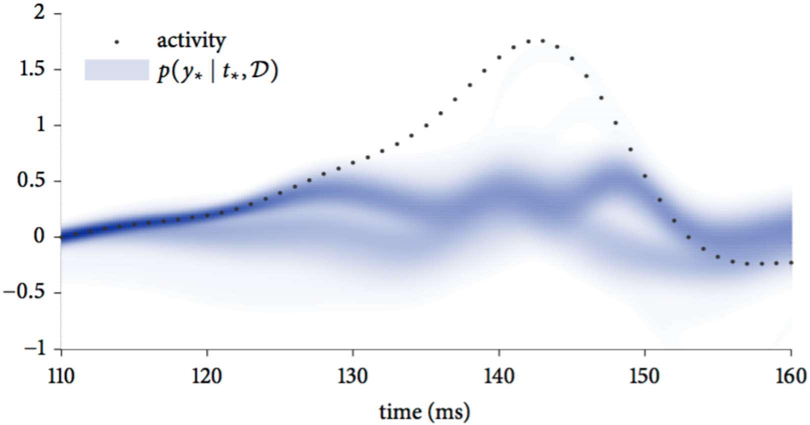
- In *"small data"* or *ambiguous* situations, it can be useful to *marginalize* model (hyper)parameters.
- It can also be useful to visualize *(hyper)parameter posteriors*
- Bayesian quadrature can be used to approximate both.

Example: EEG (Garnett, et al. 2010)





Example: EEG (Garnett, et al. 2010)



More: Osborne, et al. 2012

Bayesian Quadrature for Ratios

Michael A. Osborne
Department of Engineering Science
University of Oxford
Oxford OX1 3PJ, UK
mosb@robots.ox.ac.uk

Roman Garnett
Robotics Institute
Carnegie Mellon University
Pittsburgh PA 15213, USA
rgarnett@andrew.cmu.edu

Stephen J. Roberts
Department of Engineering Science
University of Oxford
Oxford OX1 3PJ, UK
sjrob@robots.ox.ac.uk

Christopher Hart
Department of Physics
University of Oxford
Oxford OX1 3RH, UK
{Christopher.Hart,

Suzanne Aigrain
Department of Physics
University of Oxford
Oxford OX1 3RH, UK
Suzanne.Aigrain,

Neale P. Gibson
Department of Physics
University of Oxford
Oxford OX1 3RH, UK
Neale.Gibson}@astro.ox.ac.uk

Abstract

We describe a novel approach to quadrature for ratios of probabilistic integrals, such as are used to compute posterior probabilities. This approach offers performance superior to Monte Carlo methods by exploiting a Bayesian quadrature framework. We improve upon previous Bayesian quadrature techniques by explicitly modelling the non-negativity of our integrands, and the correlations that exist between them. It offers most where the integrand is multi-modal and expensive to evaluate. We demonstrate the efficacy of our method on data from the Kepler space telescope.

1 Introduction

Bayesian inference often requires the evaluation of nonanalytic definite integrals. In the main, techniques for numerical integration estimate the integral given the value of the integrand on a set of sample points, a set that is limited in size by the computational expense of evaluating the integrand. As discussed in (O'Hagan, 1987), traditional Monte Carlo integration techniques do not make the best possible use of this valuable information. An alternative is found in Bayesian quadrature (O'Hagan, 1991), which uses these samples within a Gaussian process model to perform inference about the integrand. The analytic niceties of the Gaussian then permit inference to be performed about the integral itself, the ultimate object of our interest. However, this use of a Gaussian process comes at a cost: as the Gaussian has unbounded support, it cannot reflect the knowledge that the integrand is a non-negative probability. This means that this model will assign non-zero probability mass to negative probabilities, giving rise to misleading results. A second problem is encountered when we wish to estimate the ratio of two integrals with common terms, as is the case when we marginalise hyperparameters by evaluating the ratio of two integrals over the likelihood, as in

$$p(y|z) = \frac{\int p(y|z,\phi)p(z|\phi)p(\phi)\,\mathrm{d}\phi}{\int p(z|\phi)p(\phi)\,\mathrm{d}\phi}.$$

Here we are required to model the correlation that exists between the common terms in order to not overestimate the importance of samples in those terms.

We address the first of these problems by modeling the non-negative terms in our integrand with a Gaussian process on their logarithm. This, and the second of our problems, destroy the analytic results relied upon by previous formulations of Bayesian quadrature. We propose to linearise our ratio of integrals as a function of the terms in the integrand, around suitable 'best-fit' values. This gives us an algorithm, Bayesian Quadrature for Ratios, that on synthetic examples outperforms traditional Monte Carlo approaches. Our algorithm is also applied to real data drawn from the Kepler mission, where sophisticated inference is needed to model light curves given very noisy observations.

Appearing in Proceedings of the 15^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2012, La Palma, Canary Islands. Volume 22 of JMLR: W&CP 22. Copyright 2012 by the authors.

- Exploits *correlations* between predictions
- Extremely *complex*
- Could be improved (better kernel between predictive distributions?)

Model evidence

A compelling application of BQ is approximating *model evidence*:

$$Z = p(y \mid X, \mathcal{M}) = \int p(y \mid X, \theta, \mathcal{M}) p(\theta \mid \mathcal{M}) \, \mathrm{d}\theta$$

Bayesian quadrature could give a clean, *decision-theoretic* approach to *actively learning* model evidence for model selection, etc. . .

Model evidence: Problems

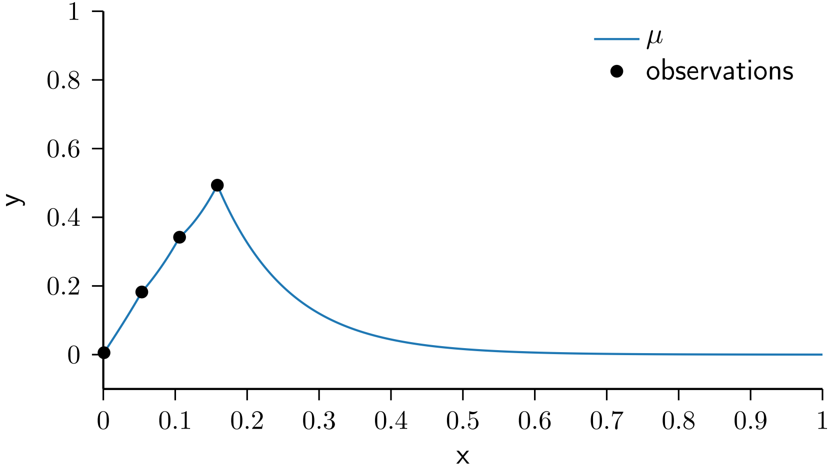
. . . but, there many problems:

- Likelihood functions look *absolutely nothing* like draws from typical GP priors (nonnegative, large dynamic range, nonstationary)
- In a naïve approach, *adaptive* sampling is impossible!

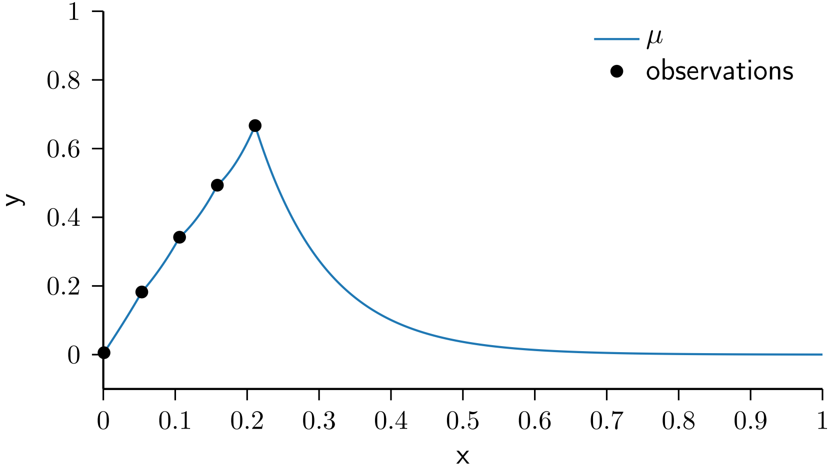
~~Remarkable~~ Unfortunate fact

- The reduction in uncertainty *does not* depend on the observed value y^*!
- This allows us to compute *optimal quadrature rules* offline!

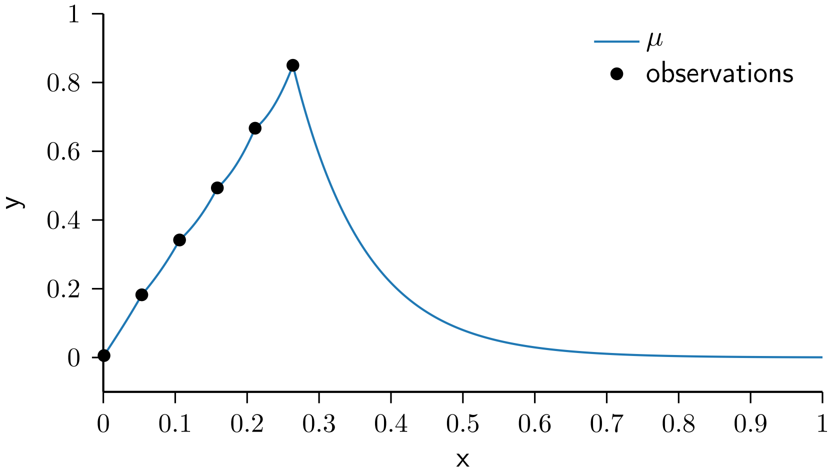
Unfortunate behavior



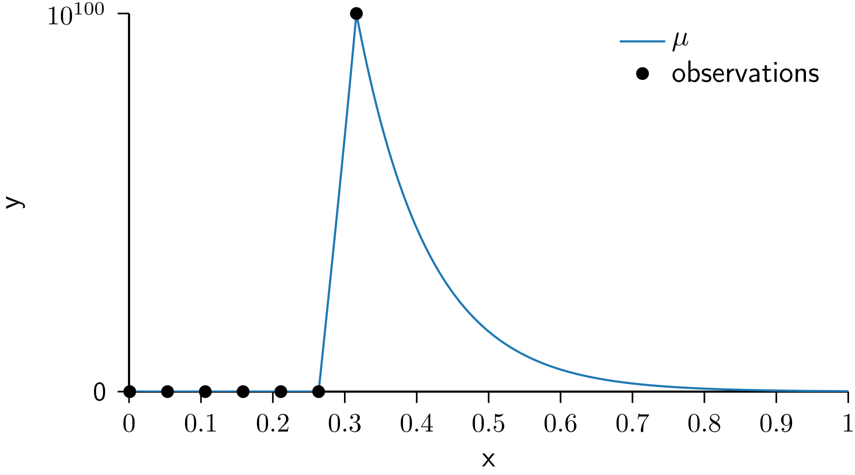
Unfortunate behavior



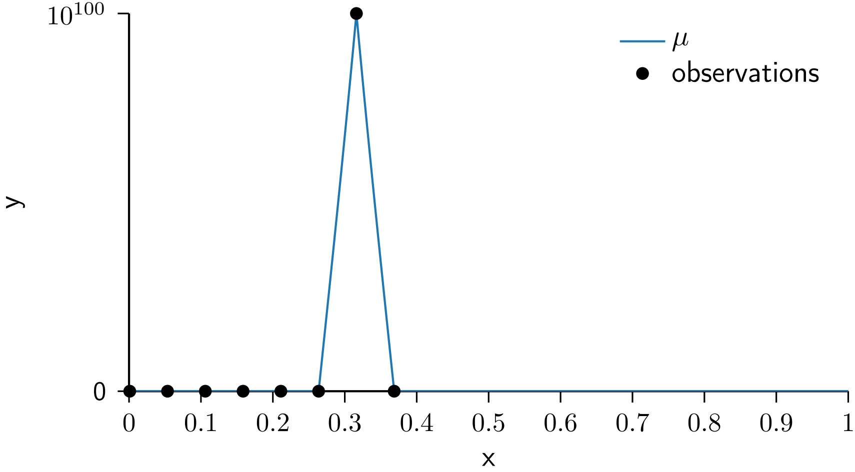
Unfortunate behavior



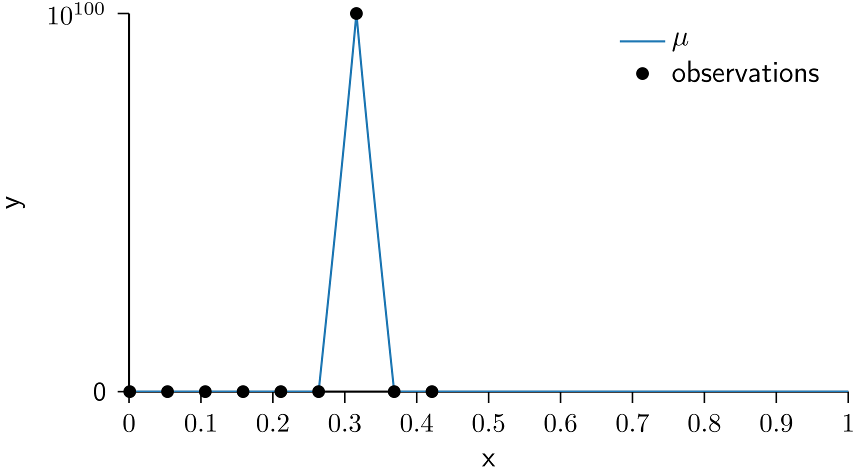
Unfortunate behavior



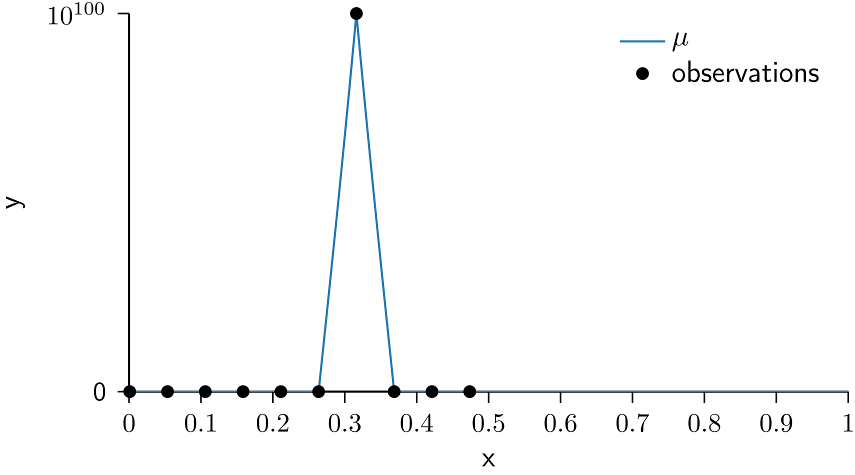
Unfortunate behavior



Unfortunate behavior



Unfortunate behavior



A useful idea

A useful *trick* is to fit a GP to a *transformation* of the likelihood:

$$p\big(g(\mathcal{L})\big) = \mathcal{GP}\big(g(\mathcal{L}); \mu, K\big),$$

do inference, then fit another GP to the *inverse-transformed* function:

$$p(\mathcal{L}) \approx \mathcal{GP}\big(\mathcal{L}; \mu', K'\big)$$

This typically gives *nonstationarity* and posterior covariances that *depend* on observed values.

Active learning: Osborne, et al. 2012

Active Learning of Model Evidence Using Bayesian Quadrature

Michael A. Osborne
University of Oxford
mosb@robots.ox.ac.uk

David Duvenaud
University of Cambridge
dkd23@cam.ac.uk

Roman Garnett
Carnegie Mellon University
rgarnett@cs.cmu.edu

Carl E. Rasmussen
University of Cambridge
cer54@cam.ac.uk

Stephen J. Roberts
University of Oxford
sjrob@robots.ox.ac.uk

Zoubin Ghahramani
University of Cambridge
zoubin@eng.cam.ac.uk

Abstract

Numerical integration is a key component of many problems in scientific computing, statistical modelling, and machine learning. Bayesian Quadrature is a model-based method for numerical integration which, relative to standard Monte Carlo methods, offers increased sample efficiency and a more robust estimate of the uncertainty in the estimated integral. We propose a novel Bayesian Quadrature approach for numerical integration when the integrand is non-negative, such as the case of computing the marginal likelihood, predictive distribution, or normalising constant of a probabilistic model. Our approach approximately marginalises the quadrature model's hyperparameters in closed form, and introduces an active learning scheme to optimally select function evaluations, as opposed to using Monte Carlo samples. We demonstrate our method on both a number of synthetic benchmarks and a real scientific problem from astronomy.

1 Introduction

The fitting of complex models to big data often requires computationally intractable integrals to be approximated. In particular, machine learning applications often require integrals over probabilities

$$Z = \langle \ell \rangle = \int \ell(\mathbf{x}) p(\mathbf{x}) \mathrm{d}\mathbf{x}, \tag{1}$$

where $\ell(\mathbf{x})$ is non-negative. Examples include computing marginal likelihoods, partition functions, predictive distributions at test points, and integrating over (latent) variables or parameters in a model. While the methods we will describe are applicable to all such problems, we will explicitly consider computing model evidences, where $\ell(\mathbf{x})$ is the unnormalised likelihood of some parameters $x_1, \dots, x_D$. This is a particular challenge in modelling big data, where evaluating the likelihood over the entire dataset is extremely computationally demanding.

There exist several standard randomised methods for computing model evidence, such as annealed importance sampling (AIS) [1], nested sampling [2] and bridge sampling. For a review, see [3]. These methods estimate Z given the value of the integrand on a set of sample points, whose size is limited by the expense of evaluating $\ell(\mathbf{x})$. It is well known that convergence diagnostics are often unreliable for Monte Carlo estimates of partition functions [4, 5, 6]. Most such algorithms also have parameters which must be set by hand, such as proposal distributions or annealing schedules.

An alternative, model-based, approach is Bayesian Quadrature (BQ) [7, 8, 9, 10], which specifies a distribution over likelihood functions, using observations of the likelihood to infer a distribution

- Fits a GP to the log probability; fits a GP to the exponentiated log GP, works with either as appropriate
- *Extremely complex*

Active learning: Osborne, et al. 2012

(we omit the laborious details).

3. WSABI

Working towards simpler, scalable BQ

Problems

- Likelihood functions are *nonnegative.*
- Computing the *expected utility/value of information* is expensive, due to nonlinear transformations (and other complications).

WSABI: Solutions

- Use a simpler transformation: $\mathcal{L} \mapsto \sqrt{\mathcal{L}}$
- Use a simpler acquisition function: *uncertainty sampling* on the pulled-back GP.

Dealing with a square root

We offer two approximations to the pulled-back GP from

$$\sqrt{\mathcal{L}} \sim \mathcal{GP}(\mu, K)$$

- *Linearization* (WSABI-L):

$$\mu'(x) = \mu(x)^2$$
$$K'(x, x') = \mu(x)K(x, x')\mu(x')$$

- *Moment-matching* (WSABI-M):

$$\mu'(x) = \mu(x)^2 + K(x, x)$$
$$K'(x, x') = \mu(x)K(x, x')\mu(x') + K(x, x')$$

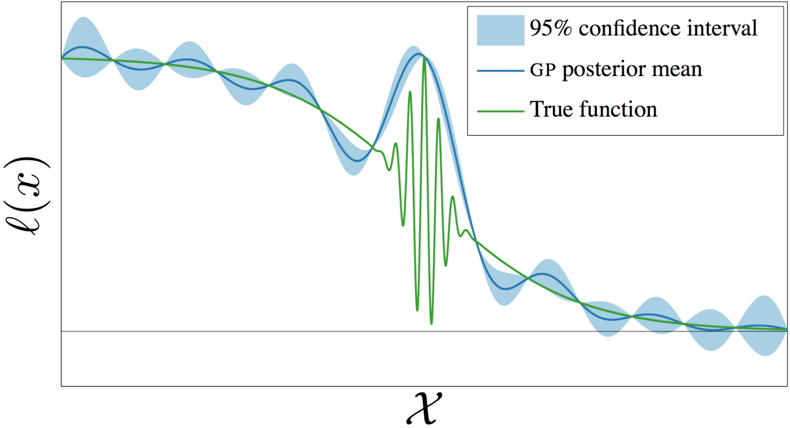
Acquisition function

We acquire observations using *uncertainty sampling*; for example, for WSABI-L, we sample at:

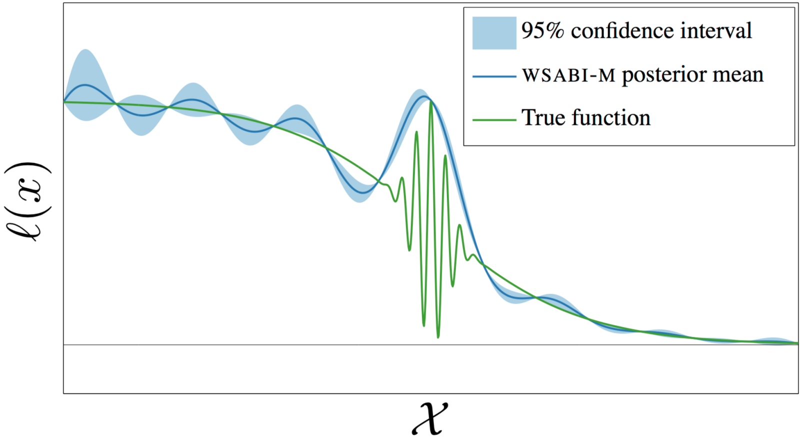
$$x^* = \arg\max_x \mu(x)^2 K(x, x).$$

- Naturally balances *exploitation and exploration!*
- *Cheap!*
- *Adapative!*

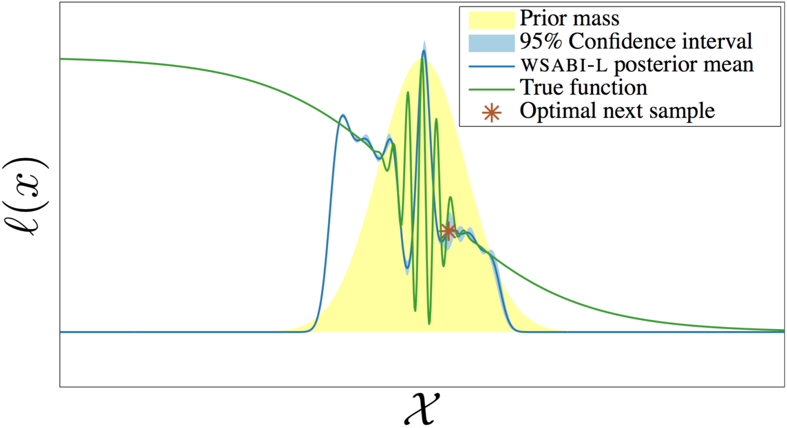
WSABI: Example (normal GP)



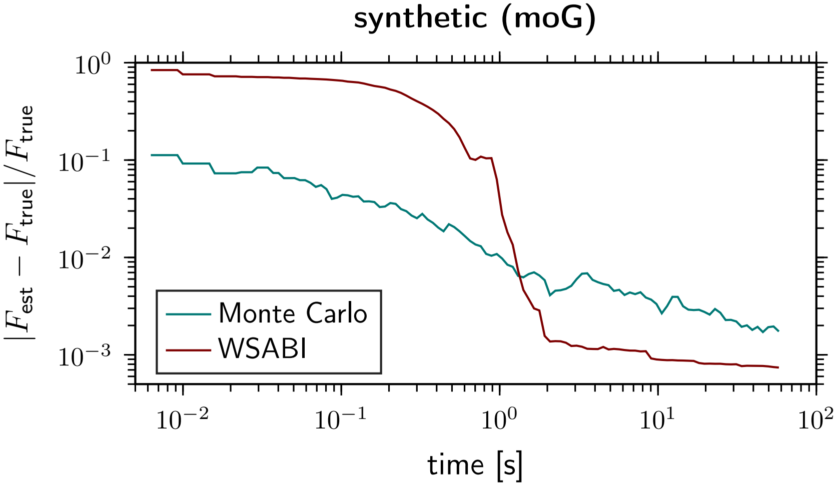
WSABI: Example (WSABI-L)



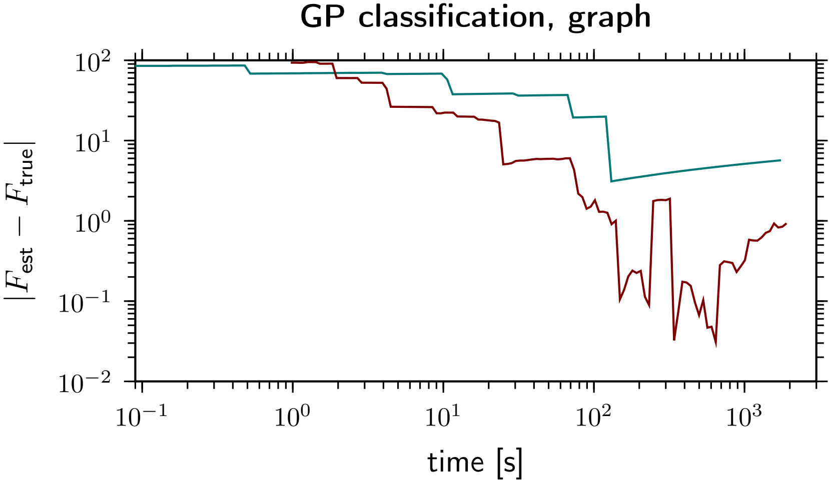
WSABI: Complex Example



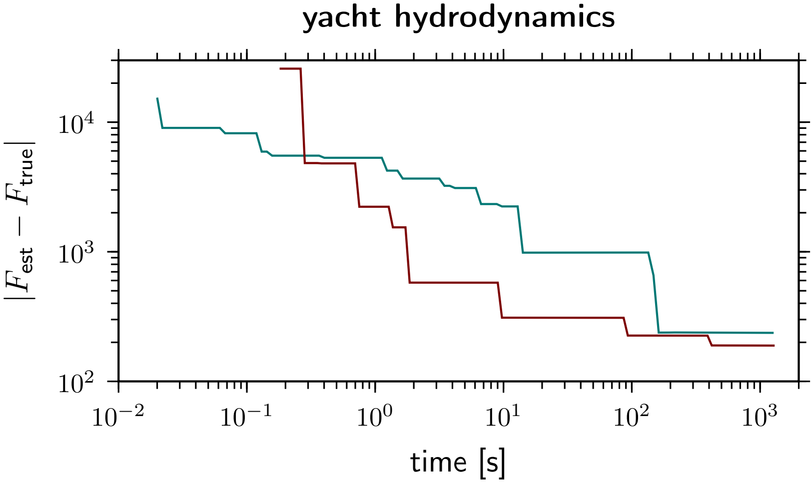
WSABI: Results



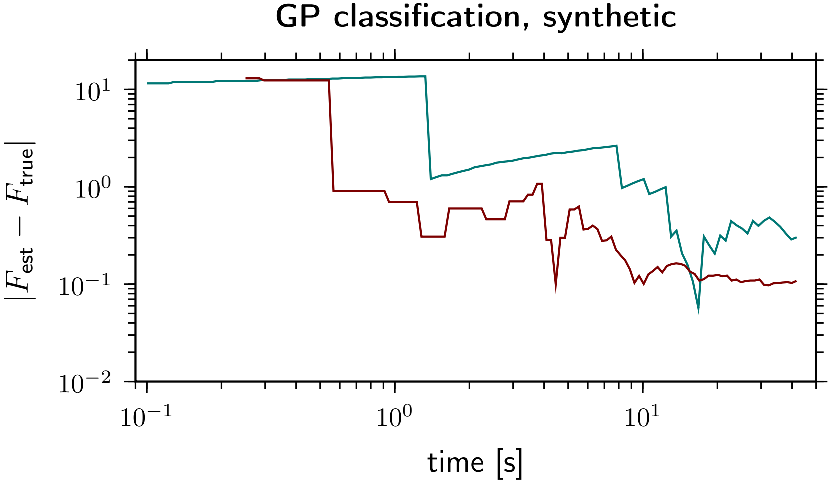
WSABI: Results



WSABI: Results



WSABI: Results



Aside: Why uncertainty sampling?

- One insight: connect to *determinantal point processes* (DPPs).
- Result: Greedy DPP MAP is equivalent to GP uncertainty sampling!
- Here the *“quality score”* is the posterior mean μ, and the *“diversity function”* is the posterior covariance K.
- Quality scores are adaptively updated!

Aside: Why use the square root?

. . . I don't know!

Isserlis' theorem

BULL. AUSTRAL. MATH. SOC. 62H05
VOL. 32 (1985), 103-107. (60E05)

THE MOMENTS OF THE MULTIVARIATE NORMAL

C.S. WITHERS

Explicit expressions are given for the noncentral moments of the multivariate normal. Finding the general moment is shown to be equivalent to finding the general derivative of the density of the multivariate normal, that is to finding an expression for the multivariate Hermite polynomial.

1. Introduction and summary

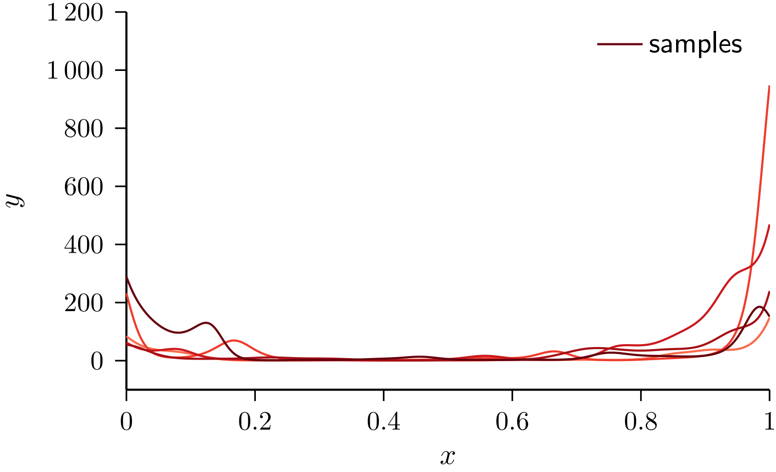
Expressions are given for the general moment of a p-dimensional normally distributed random variable $X = (X_1, \ldots, X_p)$ with mean μ and covariance $\Sigma = (\sigma_{ij})$.

Moment-matched exp-log-GP

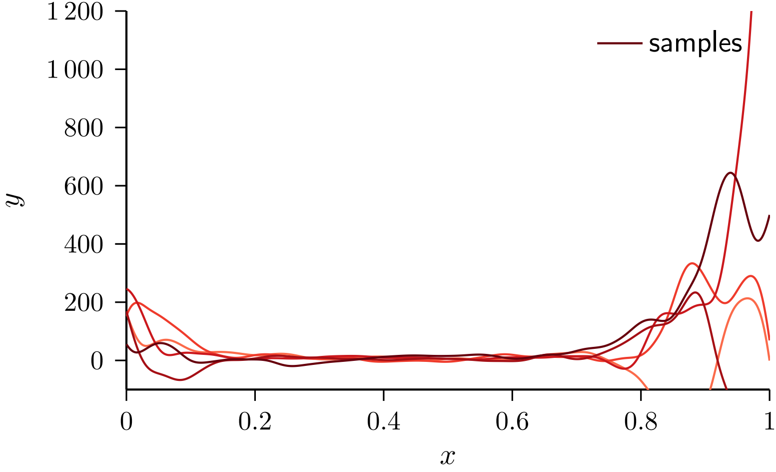
We can moment-match to *any transformation* in terms of the latent mean and covariance! For example, for the log-GP, we have:

$$\begin{aligned} \mu'(x) &= \exp\big(\mu(x) + \tfrac{1}{2}K(x,x)\big) \\ K'(x,x') &= \mu'(x)\Big(\exp\big(K(x,x')\big) - 1\Big)\mu'(x') \end{aligned}$$

Not bad



Not bad



4. LOOKING FORWARD

Challenges

Killer app?

Where are the *applications?*

Deep BQ?

$$\int\!\!\int\!\!\int\!\!\int\!\!\int\!\!\int\!\!\int\!\!\int f(x)\,\mathrm{d}x$$

Model building

We have *no hope* of integrating most functions modeled by typical GPs! Once you write

$$f \sim \mathcal{GP}(0, \text{isotropic Gaussian})$$

you have *already failed.*

Model building

- Should we be *identifying/solving lower-dimensional integrals?* (Informative *line integrals?*)
- What about the hyperparameters of the likelihood GP? Can we learn them *jointly?*
- What are *best practices* for modeling likelihoods? (There really should be a study on this alone.)

Expensive acquisition

Dirty secret: we still have to perform *nonconvex optimization every iteration!*

Computational cost

The even bigger elephant in the room...

Example: Nile depth (Garnett, et al. 2010)