GP-Ouroboros: Using Gaussian Processes to model the log marginal likelihood in Bayesian hyperparameter Inference for Gaussian Processes

Zachary Lau

February 22, 2026

Abstract

Markov Chain Monte Carlo (MCMC) methods are particularly useful for estimating expectations in higher-dimensional parameter spaces. However, these algorithms may not be practical when the likelihood is very expensive to evaluate. This problem can occur in Bayesian inference on the hyperparameters of large Gaussian Processes (GP). One alternative which has been proposed for expensive likelihoods is to use the posterior mean of a Gaussian Process to model the log-likelihood or log posterior. However, just using the mean ignores the variance term also provided by GPs. We explore two alternative ways of incorporating this variance term; firstly by incorporating it in the mean of a log normal distribution (we call this the adjusted mean), and secondly by sampling gradients instead of evaluating them deterministically. Our experimental and theoretical results show that neither method is entirely satisfactory. The adjusted mean target displays unhelpful boundary seeking behaviour, while directly sampling gradients turns out to be theoretically unsound. We hope that in the future the failure of these algorithms can be learned from so that we can properly account for uncertainty in stochastic surrogate modelling of the log-likelihood for Bayesian inference.

1 Introduction and Background

1.1 Likelihood free and surrogate likelihood methods

The challenge of performing Bayesian inference with intractable or expensive likelihood functions has been approached from multiple directions in the past. Rasmussen [5] suggests making proposals using HMC applied to a surrogate model of the log posterior and then performing a Metropolis-Hastings accept-reject with evaluations of the true likelihood. However, it is not clear that this scheme has the correct stationary distribution. Christen & Fox [2] introduced delayed acceptance MCMC which similarly makes proposals using a surrogate, but uses a pre-screening step to ensure that the chain targets the correct distribution. On the other hand, likelihood free approaches attempt to perform bayesian inference when the likelihood function is not available or tractable, but simulations can be made from the data generating process. One such framework is Approximate Bayesian Computation (ABC) [1]. More recently Gutmann & Corander [3] have considered applying Bayesian optimization to improve simulation efficiency. Finally Stuart et al. [8] discuss error bounds on the distance between the true posterior and a Gaussian Process approximation of the posterior in terms of Hellinger distance. Our work differs from that of Rasmussen and Christen & Fox in that we focus on sampling directly from the surrogate model without a correction step. Furthermore, it differs from ABC and likelihood free methods in that the surrogate model is based on true evaluations of the likelihood.

1.2 Gaussian Processes

Gaussian processes (GP) modelling (see [6]) is a type of non-parametric Bayesian modelling. It can be viewed as placing a Gaussian process prior on a function, i.e. observations $y_1, y_2, \ldots, y_n$ at any finite set of input points $x_1, x_2, \ldots x_n$ are distributed according to a multivariate normal distribution. Such a prior is fully specified by a mean function $\mu(x)$ and the covariance function $k(x, x')$. Given a set of training points $x_1, \ldots, x_n$, inference at a set of test points $x'_1, \ldots, x'_m$ is found by computing the conditional distribution of $y'_1, \ldots, y'_m$ given $y_1, \ldots, y_n$. The resulting posterior distribution sees $[y'_1, \ldots, y'_m]^\top \sim \mathcal{N}(\mu', K')$ where μ' and K' are given by the inference equations

$$\mu' = \mu + k^*(K + \sigma^2 I)^{-1}\vec{y}$$
$$K' = k^{**} - k^*(K + \sigma^2 I)^{-1}k^{*\top}$$

Where μ represents the prior mean, $k^{**}_{ij} = k(x'_i, x'_j)$ gives the prior covariance, $k^*_{ij} = k(x'_i, x_j)$ the cross-covariance, $\vec{y}$ the vector of observations, and σ^2 represents i.i.d. observational noise. Even when there is no noise, a small amount of noise may be included for numerical stability of the inverse. An important feature of Gaussian process modelling is that in addition to giving estimates at new test points through the posterior mean, it also quantifies uncertainty through the posterior covariance.

We will consider GPs from two perspectives. Firstly, Bayesian hyperparameter inference in GPs provides a good test case where the likelihood evaluation can be prohibitive. Secondly GPs can themselves be used as a surrogate model for the likelihood that quantifies uncertainty at unobserved locations.

1.3 Hyper-parameter inference for Gaussian Processes

Typically the covariance function used for GP inference belongs to a parametric family. For example, the squared-exponential family in one-dimension has the form $k(x, x') = \sigma_f^2 \exp\left(-\frac{1}{2}\left(\frac{x-x'}{\ell}\right)^2\right)$ with hyperparameters σ_f^2 and ℓ. It is most common to choose these parameters given the data by maximizing the so-called log-marginal likelihood, i.e. the likelihood of the observations after marginalizing over the unobserved latent function. With observation noise, this function becomes

$$l(\theta) = -\frac{1}{2}y^\top(K + \sigma^2 I)^{-1}y - \frac{1}{2}\log|K + \sigma^2 I| - \frac{n}{2}\log(2\pi)$$

An alternative to using MLE is to place a prior over the hyperparameters, and then perform inference by averaging over the hyperparameter posterior. Our target then becomes

$$\log p(\theta|y) = \log p(\theta) - \frac{1}{2}y^\top(K + \sigma^2 I)^{-1}y - \frac{1}{2}\log|K + \sigma^2 I| - \frac{n}{2}\log(2\pi)$$

Some of the major challenges with sampling from such a distribution include multi-modality and the computational cost of computing matrix solves and determinants. The latter can be a particular challenge in the case of GPs with large numbers of observations, particularly since their cost scales as $O(n^3)$.

2 Algorithms

2.1 Set up and motivation

Consider a Bayesian inference problem with an expensive likelihood such as the GP problem described above. The first premise of this work, whose validity may be explored in future work, is

that in the expensive likelihood regime, Bayesian inference should be separated into two parts

- *Likelihood model building*: judicious evaluations of the likelihood function to provide the most useful understanding of the likelihood function
- *Sampling*: Generation of samples from the posterior implied *by our model*

2.2 Mean estimation

The most obvious way to sample from the GP surrogate model is to use the posterior mean as the log likelihood. That is, if the posterior mean is given by $\mu(\theta)$ we take $\hat{l}(\theta) = \mu(\theta)$ so that the likelihood is given by $\mathcal{L}(\theta) = \exp(\mu(\theta))$. Another method is to consider the posterior mean of f itself. Note that $f(\theta)$ is marginally log-normal so it has expectation $\mathbb{E}f(\theta) = \exp\left(\mu(\theta) + \frac{1}{2}\sigma^2(\theta)\right)$, giving log-likelihood $\hat{l}(\theta) = \mu(\theta) + \frac{1}{2}\sigma^2(\theta)$. This second estimate takes into account uncertainty, but only insofar as it affects the mean. The posteriors arising from such likelihoods can be targeted using off the shelf samplers like NUTS [4].

2.3 Uncorrected Langevin Dynamics

The third method we propose to analyze involves taking *samples* from the gradient instead of using a fixed surrogate model in order to reflect uncertainty in the surrogate model. Importantly, we note that the gradient of a GP and the function itself are jointly still a GP, so sampling amounts to sampling from a GP posterior. We restrict ourselves to uncorrected Langevin dynamics because this requires only a single sample of the gradient. Sampling a longer HMC trajectory would be significantly more expensive because each gradient sample would need to be taken conditional on the previous samples. With sample i requiring a roughly $O(i^2)$ block matrix update to the inverse, the total trajectory cost would be roughly $O(L^3)$. On the other hand, given a precomputed inverse that is shared amongst all points, a single Langevin step with a model built on m true evaluations requires only a single $O(m^2)$ matrix multiplication.

2.4 Theoretical accuracy

As discussed in [8], the theoretical optimum model, in the sense of Hellinger distance, to sample from would be the *joint* distribution over f and θ, i.e. the model defined by the following generative process.

$$\begin{aligned} \log f &\sim \mathcal{GP} \\ \theta &\sim \frac{p_0(\theta) f(\theta)}{\mathcal{Z}(f)} \end{aligned}$$

Unfortunately, sampling from this joint distribution is a difficult problem that our proposed methods only approximate in the well informed case.

Firstly, we consider our Langevin dynamics inspired sampler. This can be seen as an approximate Metropolis-within-Gibbs sampler where the marginal $f|\theta$ is approximated by the GP posterior but *not conditioned on* θ. A true conditional update of f would need to preserve (see derivation in Appendix)

$$\begin{aligned} p(f|\theta) &= \frac{p(f,\theta)}{p(\theta)} \\ &\propto \frac{p(f) f(\theta)}{\mathcal{Z}(f)} \end{aligned}$$

On the other hand, if we approximate $\mathcal{Z}(f)$ as a constant, we recover the log mean sampler $\hat{l}(\theta) = \mu(\theta) + \frac{1}{2}\sigma^2(\theta)$. To see this, note that integrating our joint distribution over f we find

$$\begin{aligned} p(\theta) &= \int p(f, \theta) df \\ &\propto \int p_0(\theta) f(\theta) p(f) df \\ &= p_0(\theta) \mathbb{E} f(\theta) \\ &= p_0(\theta) \exp\left(\mu(\theta) + \frac{1}{2}\sigma^2(\theta)\right) \end{aligned}$$

Thus none of our samplers targets the "correct" stationary distribution, with the former not accounting for the $f(\theta)$ or $\mathcal{Z}(f)$ term, and the latter failing to account for the $\mathcal{Z}(f)$ term.

3 Experiments

Each method (GP mean, adjusted mean, sampled gradients) was tested on the Cartesian product of $n \in \{10, 100, 1000\}$, $d \in \{1, 5, 50\}$ and $m \in \{10, 100\}$ where n is the number of observations in the base model, d the dimensionality of the base model and m the number of likelihood evaluations used to build the surrogate model. The surrogate model was built on random evaluations uniformly chosen in our parameter space. For $n \in \{10, 100\}$ the results were compared to a posterior sample computed using the true log-likelihood, but this was prohibitive for $n = 1000$. Implementation was in Python and used Stan [7] for NUTS. Full code, results and details of the experiments can be found on Github [1]. The Langevin sampler behaves overall poorly. This is not surprising in high dimensions given its local behaviour, however even in moderate dimension ($d = 5$) it displays pathological behaviour (see Figure 1). This may be explained by the fact that it does not target the correct joint distribution. The most interesting result from the adjusted mean sampler is its occasional boundary seeking behaviour (see Figure 3). The long right-tail of the log-normal distribution encourages it to sample from areas of high-variance even when this is of detriment to its qualitative results. We hypothesize that this is related to the target's failure to account for the normalizing factor $\mathcal{Z}(f)$ which tempers the strength of the upper tail. Overall the mean-model performs decently, displaying few pathological behaviours. It could likely be improved by more judicious choices of likelihood evaluations. This is a possible future area of research.

4 Conclusion

In this project we provide a renewed exploration of the idea of using stochastic surrogate models of the likelihood function in Bayesian inference, particularly with a focus on Bayesian inference for Gaussian process hyperparameter learning. We propose two basic algorithms to incorporate uncertainty: incorporating it through the mean of a log-normal or through sampling. While both methods have experimental and theoretical shortfalls, we hope this opens the door for future research on understanding how surrogate models which quantify uncertainty can be used in Bayesian inference. In particular, another promising area of research we have not yet explored is how to acquire the log-likelihood evaluations used to build the surrogate model in the first place.

[1] https://github.com/zach-lau/STAT547D-Project

References

[1] Mark A. Beaumont. Approximate bayesian computation. *Annual Review of Statistics and Its Application*, 6:379–403, 2019.

[2] J. Andrés Christen and Colin Fox. Markov chain monte carlo using an approximation. *Journal of Computational and Graphical Statistics*, 14(4):795–810, 2005.

[3] Michael U. Gutmann and Jukka Corander. Bayesian optimization for likelihood-free inference of simulator-based statistical models. *Journal of Machine Learning Research*, 17(125):1–47, 2016.

[4] Matthew D. Hoffman and Andrew Gelman. The no-u-turn sampler: Adaptively setting path lengths in hamiltonian monte carlo, 2011.

[5] Carl Edward Rasmussen. Gaussian processes to speed up hybrid monte carlo for expensive bayesian integrals. In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. F. M. Smith, and M. West, editors, *Bayesian Statistics 7*, pages 651–659, Oxford, UK, 2003. Oxford University Press.

[6] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA, USA, 2006.

[7] Stan Development Team. The Stan Core Library, 2026. Version 2.38.0.

[8] Andrew M. Stuart and Aretha L. Teckentrup. Posterior consistency for gaussian process approximations of bayesian posterior distributions. *Mathematics of Computation*, 87(310):721–753, 2018.

A Target conditional

Given our generative model, we get joint distribution

$$p(f, \theta) = p(f)\frac{p_0(\theta)f(\theta)}{\mathcal{Z}(f)}$$

Where the normalizing constant $\mathcal{Z}(f) = \int p_0(\theta)f(\theta)d\theta$. This gives the θ marginal as

$$\begin{aligned} p(\theta) &= \int p(f)\frac{p_0(\theta)f(\theta)}{\mathcal{Z}(f)}df \\ &= p_0(\theta)\int p(f)\frac{f(\theta)}{\mathcal{Z}(f)}df \end{aligned}$$

The conditional distribution of $f|\theta$ is given by

$$p(f|\theta) \propto \frac{p(f)f(\theta)}{\mathcal{Z}(f)}$$

Intuitively this skews f higher at the current value of θ, but this scaling is tempered by the subsequent increase in the normalizing constant.

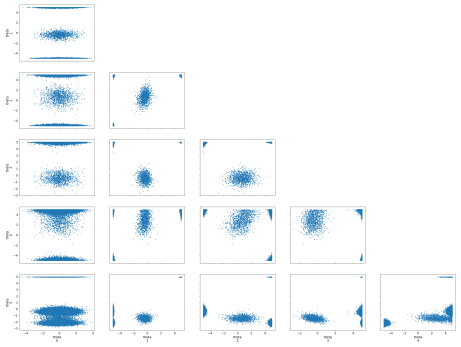
Algorithm 1 Unadjusted Langevin for GP Likelihood

Require: Stepsize τ, Iterations N
  Compute K the covariance amongst train points
  $Ki \leftarrow (K + \sigma^2 I)^{-1}$
  Initialize $\theta \sim p_0(\theta)$
  for $i = 0$ to N **do**
    Sample $\nabla \log f(\theta)$ from GP model
    Sample $\xi \sim \mathcal{N}(0, 1)$
    $\theta \leftarrow \theta + \tau \nabla \log f(\theta) + \sqrt{2\tau}\xi$
    Save θ
  end for

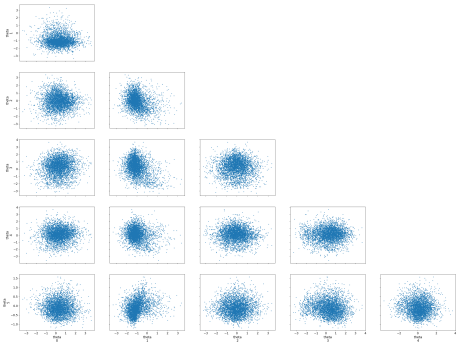
B Algorithms

Algorithm 1 is used for uncorrected Langevin dynamics using GP sampled gradients. In implementation step size is scaled down proportional to the standard deviation of observed log-likelihoods.

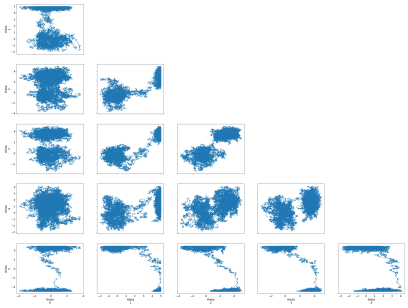
C Figures and Tables



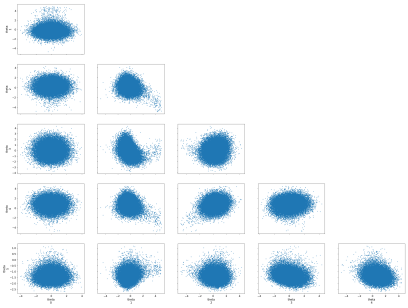
(a) Adjusted mean



(b) "True" likelihood

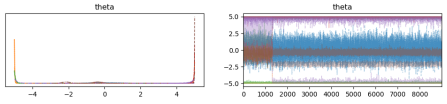


(c) Langevin

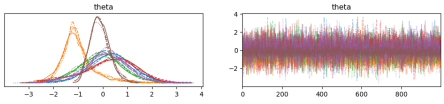


(d) GP Mean

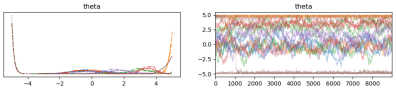
Figure 1: Pair plots for different methods of sampling from GP surrogate model where the base model has $n = 10$ observations in $d = 5$ dimensions and the surrogate log-likelihood model is built on $m = 100$ likelihood evaluations. We can see the boundary seeking behaviour inherent in using the adjusted mean sampler. Furthermore the Langevin dynamics based sampler displays a combination of poor mixing and bizarrely invents multimodality. Note that due to the $O(n^3)$ cost of evaluating the true likelihood and $O(m^2)$ of evaluating the surrogate it may sometimes make sense to have $m > n$, although this particular example is somewhat unrealistic.



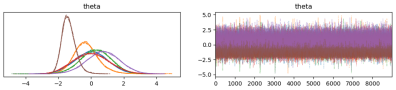
(a) Adjusted mean



(b) "True" likelihood



(c) Langevin



(d) GP Mean

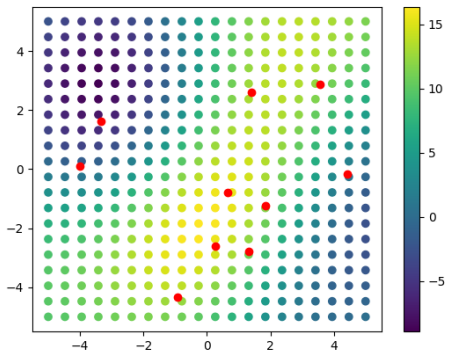
Figure 2: Traceplots and marginals of different algorithms for sampling from GP surrogate with $n = 10$, $d = 5$ and surrogate model based on $m = 100$.

Parameter	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
theta[0]	-0.01	1.00	-1.92	1.84	0.01	0.01	23649.0	20483.0	1.00
theta[1]	-2.33	4.30	-5.00	4.99	2.12	1.16	7.0	26.0	1.60
theta[2]	-0.17	4.84	-5.00	4.99	2.40	0.12	6.0	91.0	1.76
theta[3]	4.76	1.02	4.79	5.00	0.26	0.64	23.0	23.0	1.11
theta[4]	2.68	3.96	-4.96	5.00	1.88	1.24	7.0	23.0	1.53
theta[5]	0.56	2.66	-2.26	5.00	1.32	0.72	5.0	23.0	2.42

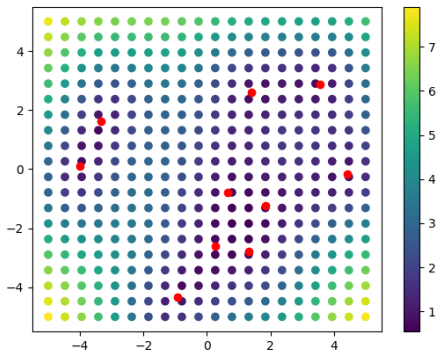
Table 1: Posterior summary statistics for adjusted mean target sampled with NUTS. Boundary effects induce multi-modality which makes this problem particularly hard to deal with for the sampler, as reflected in the low ESS and high Rhat for many variables.

Parameter	mean	sd	hdi_3%	hdi_97%	mcse_mean	mcse_sd	ess_bulk	ess_tail	r_hat
theta[0]	-0.10	1.03	-1.79	1.97	0.15	0.07	49.0	101.0	1.05
theta[1]	3.52	2.23	-0.85	5.00	1.07	0.61	7.0	33.0	1.53
theta[2]	2.03	1.77	-1.45	4.44	0.76	0.36	8.0	73.0	1.51
theta[3]	2.43	1.84	-1.21	4.47	0.84	0.46	7.0	32.0	1.52
theta[4]	1.16	1.22	-1.43	3.32	0.39	0.16	11.0	42.0	1.29
theta[5]	-2.27	4.22	-5.00	4.89	2.02	1.04	7.0	42.0	1.50

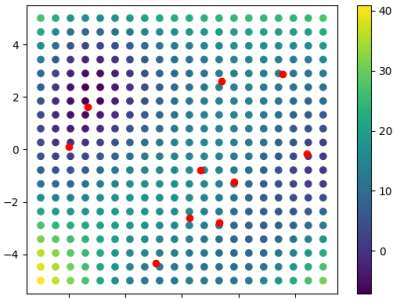
Table 2: Posterior summary statistics with GP sampled gradients. Mixing is poor due to random walk behaviour.



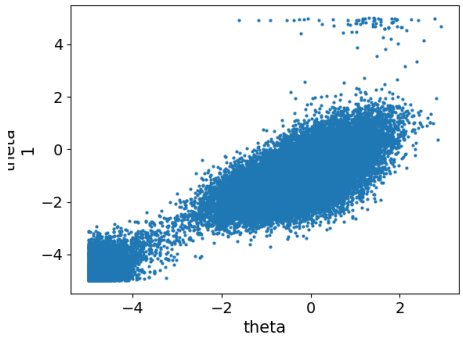
(a) GP mean of surrogate model



(b) Standard deviation of surrogate model



(c) Adjusted mean of surrogate model



(d) Samples taken with NUTS

Figure 3: The GP mean and standard deviation for a surrogate model of the likelihood for a base model with $n = 10$ observations in $d = 1$ dimension with $m = 10$ likelihood evaluations. The likelihood evaluations used to build the surrogate model are shown in red. The effective likelihood has higher density in both areas of higher mean and standard deviation. The sample on the right targets the posterior arising from using the adjusted mean likelihood with a standard normal prior.