

# Bayesian Active Learning for Posterior Estimation

Kirthevasan Kandasamy, Jeff Schneider, Barnabás Póczos

Carnegie Mellon University, Pittsburgh, PA, USA

{kandasamy, schneide, bapoczos}@cs.cmu.edu

## Abstract

This paper studies active posterior estimation in a Bayesian setting when the likelihood is expensive to evaluate. Existing techniques for posterior estimation are based on generating samples representative of the posterior. Such methods do not consider efficiency in terms of likelihood evaluations. In order to be query efficient we treat posterior estimation in an active regression framework. We propose two myopic query strategies to choose where to evaluate the likelihood and implement them using Gaussian processes. Via experiments on a series of synthetic and real examples we demonstrate that our approach is significantly more query efficient than existing techniques and other heuristics for posterior estimation.

## 1 Introduction

Computing the posterior distribution of parameters given observations is a central problem in statistics. We use the posterior distribution to make inferences about likely parameter values and estimate functionals of interest. For simple parametric models we may obtain the posterior in analytic form. In more complex models where the posterior is analytically intractable, we have to resort to approximation techniques. In some cases, we only have access to a black box which computes the likelihood for a given value of the parameters.

Our goal is an efficient way to estimate posterior densities when calls to this black box are expensive. This work is motivated by applications in computational physics and cosmology. Several cosmological phenomena are characterized by the cosmological parameters (e.g. Hubble constant, dark energy fraction). Given observations, we wish to make inferences about the parameters. Physicists have developed simulation-based probability models of the Universe which can be used to compute the likelihood of cosmological parameters for a given observation. Figure 1 shows different scenarios to estimate / compute the likelihood. Many problems in scientific computing have a similar flavour. Expensive simulators in molecular mechanics, computational biology and neuroscience are used to model many scientific processes.

**Our contribution** is to propose a query efficient method for estimating posterior densities when the likelihood function is expensive to evaluate. We adopt a Bayesian active regression approach on the log likelihood using the samples it has already computed. We refer to this approach as Bayesian Active Posterior Estimation (BAPE). We propose two myopic query strategies on the uncertainty regression model for sample selection. Our implementation uses Gaussian processes (GP) [Rasmussen and Williams, 2006] and we demonstrate the efficacy of the methods on multiple synthetic and real experiments.

## 2 Related Work

Practitioners have conventionally used sampling schemes [MacKay, 2003] to approximate the posterior distributions. Rejection sampling and various MCMC methods are common choices. The advantage of MCMC approaches is their theoretical guarantees with large sample sets [Robert and Casella, 2005] and thus they are a good choice when likelihood evaluations are cheap. However, none of them is intended to be query efficient when evaluations are expensive. Some methods spend most of their computation evaluating point likelihoods and then discard the likelihood values after doing an acceptance test. This gives insight into the potential gains possible by retaining those likelihoods for use in regression. Despite such deficiencies, MCMC remains one of the most popular techniques for posterior estimation in experimental science [Foreman-Mackey *et al.*, 2013; Parkinson *et al.*, 2006; Landau and Binder, 2005; Liu, 2001].

Approximate Bayesian computation (ABC) [Marin *et al.*, 2012; Marjoram *et al.*, 2003] is a method of last resort for estimating posteriors when a likelihood can not be computed. Unfortunately, it still requires the same generation of simulated data, which is expensive in our setup, and it does not address efficient selection of parameter values to be tested at all. Nested Sampling [Skilling, 2006] is a technique commonly used is Astrostatistics. Kernel Bayes' Rule [Fukumizu *et al.*, 2014] is a non-parametric method of computing a posterior based on the embedding of probabilities in an RKHS. All these methods require sampling from a distribution and do not address the question of which samples to choose if generating them is expensive. The work in Bryan *et al.* [2006]

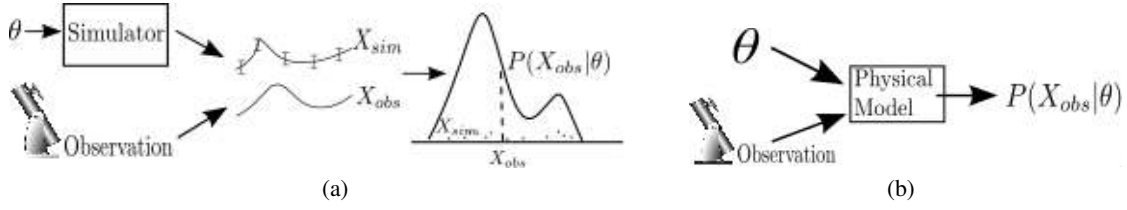


Figure 1: **Illustrations of Cosmological Experiments.** (a): Given a parameter value  $\theta$  the oracle produces several simulations  $\mathbf{X}_{\text{sim}}$ . The likelihood  $P(\mathbf{X}_{\text{obs}}|\theta)$  can then be estimated via a density estimate using  $\mathbf{X}_{\text{sim}}$  at the given  $\theta$ . (b): The oracle directly computes the likelihood using a physical model of the universe.

actively learns level sets of an expensive function and derives confidence sets from the results. Gotovos *et al.* [2013] also actively learn level sets via a classification approach. Our work is more general since we estimate the entire posterior.

Our methods draw inspiration from Gaussian Process based active learning methods such as Bayesian optimisation (BO) [Mockus and Mockus, 1991], Bayesian quadrature (BQ) [Osborne *et al.*, 2012], active GP Regression (AGPR) [Seo *et al.*, 2000] and several others [Srinivas *et al.*, 2010; Gunter *et al.*, 2014; Ma *et al.*, 2014; Krause *et al.*, 2008; Kandasamy *et al.*, 2015]. These methods have a common modus operandi to determining the experiment  $\theta_t$  at time step  $t$ : Construct a utility function  $u_t$  based on the posterior GP conditioned on the queries so far. Then maximize  $u_t$  to determine  $\theta_t$ .  $u_t(\theta)$  captures the value of performing an experiment at point  $\theta$ . Existing theoretical results [Golovin and Krause, 2011] justify such myopic strategies for homogeneous and stateless utility functions. Maximizing the typically multimodal  $u_t$  is itself a hard problem. However, it is generally assumed that querying the function is more costly than this maximization [Brochu *et al.*, 2010; Srinivas *et al.*, 2010]. The key difference in such methods is essentially in the specification of  $u_t$  to determine the next experiment. In our work, we adopt this strategy. We present two utility functions for active posterior estimation.

### 3 Bayesian Posterior Estimation

**Problem Setting:** We formally define our posterior distribution estimation problem in a Bayesian framework. We have a bounded continuous parameter space  $\Theta$  for the unknown parameters (e.g. cosmological constants). Let  $\mathbf{X}_{\text{obs}}$  denote our observations (e.g. signals from telescopes). For each  $\theta \in \Theta$  we have the ability to query an oracle for the value of the likelihood  $\mathcal{L}(\theta) = P(\mathbf{X}_{\text{obs}}|\theta)$ , but these queries are expensive. Assuming a prior  $P_\theta(\theta)$  on  $\Theta$ , we have the posterior  $P_{\theta|\mathbf{X}_{\text{obs}}}$ .

$$P_{\theta|\mathbf{X}_{\text{obs}}}(\theta|\mathbf{X}_{\text{obs}}) = \frac{\mathcal{L}(\theta)P_\theta(\theta)}{\int_{\Theta} \mathcal{L}(\theta)P_\theta(\theta)} = \frac{\mathcal{L}(\theta)P_\theta(\theta)}{P(\mathbf{X}_{\text{obs}})} \quad (1)$$

We wish to obtain an estimate  $\hat{P}_{\theta|\mathbf{X}_{\text{obs}}}$  of  $P_{\theta|\mathbf{X}_{\text{obs}}}$  while minimizing our queries to the oracle.

Some smoothness assumptions on the problem are warranted to make the problem tractable. In the Bayesian framework it is standard to assume that the function of interest is a sample from a Gaussian Process. In what follows we shall model the log joint probability of the cosmological parameters and the

observations via a GP<sup>1</sup>. This is keeping in line with Adams *et al.* [2008] who use a similar prior for GP density sampling and similar smoothness assumptions in Srinivas *et al.* [2010]. Assume that we have already queried the likelihood oracle at  $t - 1$  points, and for each query point  $\theta_i$  the oracle provided us with  $\mathcal{L}_i \approx P(\mathbf{X}_{\text{obs}}|\theta_i)$  answers. Let  $A_{t-1} = \{\theta_i, \mathcal{L}_i\}_{i=1}^{t-1}$  denote the set of these input output pairs. We build our GP on  $B_{t-1} = \{\theta_i, \log(\mathcal{L}_i P_\theta(\theta_i))\}_{i=1}^{t-1}$  input output pairs. If  $g$  is a sample from this GP, then  $f = \exp g / \int \exp g$  denotes a sample from the induced uncertainty model  $F_{\theta|\mathbf{X}_{\text{obs}}}$  for the posterior  $P_{\theta|\mathbf{X}_{\text{obs}}}$ . Finally, given any estimate  $\hat{P}^{A_t}(\mathbf{X}_{\text{obs}}, \theta)$  of the log joint probability, the estimate of the posterior distribution is,

$$\hat{P}^{A_t}(\theta|\mathbf{X}_{\text{obs}}) = \frac{\exp \hat{P}^{A_t}(\mathbf{X}_{\text{obs}}, \theta)}{\int_{\Theta} \exp \hat{P}^{A_t}(\mathbf{X}_{\text{obs}}, \theta)} \quad (2)$$

At time  $t$ , we wish to select the point  $\theta_t$  for the next experiment to evaluate the likelihood. We adopt a myopic strategy here by picking the point that maximizes a utility function. Our utility function needs to capture a measure of divergence  $D(\cdot||\cdot)$  between the densities. To construct this utility function, note that ideally we would like to select  $\theta_t$  to satisfy

$$\theta_t = \underset{\theta_+ \in \Theta}{\operatorname{argmin}} D(P_{\theta|\mathbf{X}_{\text{obs}}} || \hat{P}^{A_{t-1} \cup \{(\theta_+, \mathcal{L}(\theta_+))\}}) \quad (3)$$

where  $\hat{P}^{A_{t-1} \cup \{(\theta_+, \mathcal{L}(\theta_+))\}}$  is our estimate of the posterior using  $A_{t-1} \cup \{(\theta_+, \mathcal{L}(\theta_+))\}$ . Obviously, this objective is not accessible in practice, since we know neither  $P_{\theta|\mathbf{X}_{\text{obs}}}$  nor  $\mathcal{L}(\theta_+)$ . As surrogates to this ideal objective in Equation (3), in the following subsections we propose two utility functions for determining the next point: Negative Expected Divergence (NED) and Exponentiated Variance (EV). The first, NED adopts a Bayesian decision theoretic approach akin to Settles [2010]. Here, we choose the point in  $\Theta$  that yields the minimum expected divergence for the next estimate over the uncertainty model. Unfortunately, in our setting, the NED utility is computationally demanding. Therefore, we propose a cheaper alternative EV. In our experiments we found that both strategies performed equally well – so EV is computationally attractive. That said, some cosmological simulations are very expensive (taking several hours to a day) so NED is justified in such situations. We present our framework for BAPE using an appropriate utility function  $u_t$  in Algorithm 1.

<sup>1</sup>We work on the log joint probability since the log smoothes out a function and is more conducive for GP modeling. We also avoid issues such as non-negativity of  $\hat{P}_m^A(\theta|\mathbf{X}_{\text{obs}})$ . Osborne *et al.* [2012] also use a similar log-transform before applying a GP.

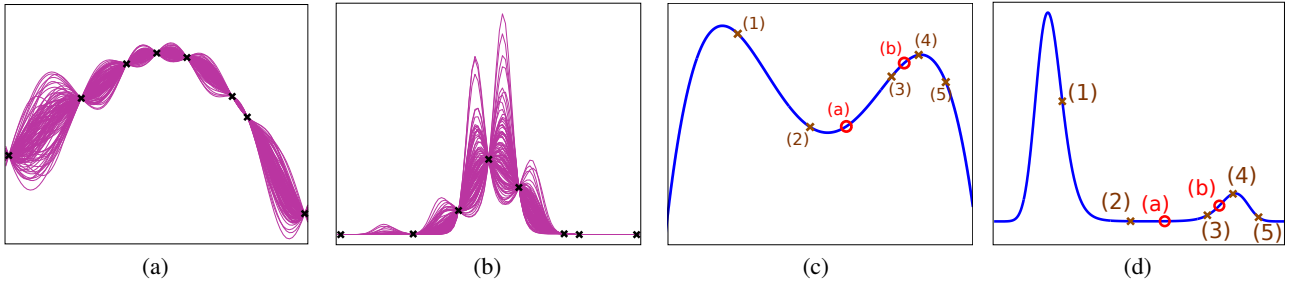


Figure 2: (a): Samples drawn from the GP uncertainty models in the log joint probability space. (b): The same samples after exponentiation. High variance in the low likelihood regions are squashed and low variances in the high likelihood regions are blown up. This is the key insight that inspires our methods. (c) and (d) are the true log joint probability and joint probability in blue. Assume that we have already queried at the brown crosses and let the red circles (a) and (b) be candidates. In BAPE we would be interested in querying (b) but not (a) . In AGPR we would be interested in both (a) and (b) whereas in BO we would be keen in neither.

### Algorithm 1 Bayesian Active Posterior Estimation

**Given:** Input space  $\Theta$ , GP prior  $\mu_0, k_0$ .

**For**  $t = 1, 2, \dots$  **do**

1.  $\theta_t = \operatorname{argmax}_{\theta_t \in \Theta} u_t(\theta)$
2.  $\mathcal{L}_t \leftarrow$  Query oracle at  $\theta_t$ .
3. Obtain posterior conditioned on  $(\theta_i, \mathcal{L}_i P_{\theta}(\theta_i))_{i=1}^t$

### 3.1 Negative Expected Divergence (NED)

Equation 3 says that we should choose the point that results in the highest reduction in divergence *if we knew the likelihood and the true posterior at that point*. In NED, we choose the point with the highest expected reduction in divergence. For this we first build uncertainty models for the value of the likelihood at  $\theta_+$  ( $L(\theta_+)$ ) and the posterior ( $F_{\theta|\mathbf{X}_{\text{obs}}}$ ). For the next evaluation we choose the point that minimizes the expected divergence between these models and the next estimate. Precisely,

$$u_t^{\text{NED}}(\theta_+) = -\mathbb{E}_{p_+} \mathbb{E}_h D(h \| \hat{P}_{m+1}^{A \cup \{\theta_+, p_+\}}). \quad (4)$$

Here  $p_+$  is sampled from  $L^A(\theta_+)$ , the uncertainty of the likelihood at  $\theta_+$ . The *density*  $h$  is sampled from  $F_{\theta|\mathbf{X}_{\text{obs}}}^{A \cup \{\theta_+, p_+\}}$ , the uncertainty model of the posterior obtained by adding  $(\theta_+, p_+)$ .  $\hat{P}_{m+1}^{A \cup \{\theta_+, p_+\}}$  denotes the estimate of the posterior obtained by re-training the GP with  $(\theta_+, p_+)$  as the  $(m+1)^{\text{th}}$  point along with the  $m$  points already available. The first expectation above captures our uncertainty over  $\mathcal{L}(\theta_+)$  while the second captures our remaining uncertainty over  $P_{\theta|\mathbf{X}_{\text{obs}}}$  after observing  $\mathcal{L}(\theta_+)$ . Equation (4) says that you should minimize the expected divergence by looking one step ahead.

The expectations in the NED utility above are computationally intractable. They can only be approximated empirically by drawing samples and require numerical integration. For these reasons we propose an alternate utility function below. In our experiments we found that both EV and NED performed equally well.

### 3.2 Exponentiated Variance (EV)

A common active learning heuristic is to choose the point that you are most uncertain about for the next experiment.

As before we use a GP on the log joint probability. At any given point in this GP we have an associated posterior variance of the GP. However, this variance corresponds to the uncertainty of the *log* joint probability whereas our objective is in learning the joint probability – which is a multiplicative factor away from the posterior. Therefore, unlike in usual GP active learning methods Seo *et al.* [2000], the variance of interest here is in the exponentiated GP. The posterior mean and variance at  $\theta_+$  of our log-joint GP are given by,

$$\mu(\theta_+) \triangleq \mathbb{E}_{F_{\theta|\mathbf{X}_{\text{obs}}}} \log P(\mathbf{X}_{\text{obs}}, \theta_+) = k(A, \theta_+)^{\top} k(A, A)^{-1} \mathbf{j} \quad (5)$$

$$\sigma^2(\theta_+) \triangleq \mathbb{V}_{F_{\theta|\mathbf{X}_{\text{obs}}}} \log P(\mathbf{X}_{\text{obs}}, \theta_+) = k(\theta_+, \theta_+) - k(A, \theta_+)^{\top} k(A, A)^{-1} k(A, \theta_+)$$

where  $k(A, A) \in \mathbb{R}^{m \times m}$  is the kernel matrix of  $A$ ,  $k(A, \theta_+) \in \mathbb{R}^m$  is the kernel vector from  $\theta_+$  to  $A$  and  $\mathbf{j} = (\log \mathcal{L}_i P_{\theta}(\theta_i))_{i=1}^m \in \mathbb{R}^m$ . By observing that an exponentiated Gaussian is a log Normal distribution, the EV utility function is given by

$$u_t^{\text{EV}}(\theta_+) = \mathbb{V}_{F_{\theta|\mathbf{X}_{\text{obs}}}} P(\mathbf{X}_{\text{obs}}, \theta_+) = \exp(2\mu(\theta_+) + \sigma^2(\theta_+))(\exp(\sigma^2(\theta_+)) - 1) \quad (6)$$

We choose the point *maximizing* the above variance. The  $\exp(2\mu(\theta_+))$  will squash high variances in the low likelihood regions and amplify low variances in the high likelihood regions (Fig 2(a), 2(b)).

It's important to distinguish our objective in this work from similar active learning literature in the GP framework. In BO, the objective is to find the maximum of a function. This means that once the active learner realises that it has found the mode of a function it has less incentive to explore around as it would not improve the current maximum values. For instance, consider the log joint probability in Figure 2(c) and the joint probability in Figure 2(d). We have shown the points where we have already queried at as brown crosses and the red circles (a) and (b) show possible candidates for the next query. In BO, the active learner would not be interested in (b) as, by virtue of points (3), (4), (5) it knows that (b) is not likely to be higher than (4). On the other hand, in BAPE

we are keen on (b) as knowing it with precision will significantly affect our estimate of the posterior (Fig 2(d)). In particular to know the posterior well we will need to query at the neighborhood of modes and the heavy tails of a distribution. A BO utility is not interested in such queries. On the other extreme, in AGPR the objective is to learn the function uniformly well. This means in the same figures, AGPR will query point (a). However, given sufficient smoothness, the joint probability will be very low there due to exponentiation. Therefore, the BAPE active learner will not be interested in (a). As figures 2(a) and 2(b) indicate, while we model the log joint probability as a GP we are more interested in the uncertainty model of the posterior/ joint probability. Finally, as a special case for BQ, Osborne *et al.* [2012] consider evaluating the model evidence—i.e. the integral under the conditional. Their utility function uses approximations tailored to estimating the integral well. Note that our goal of estimating the posterior well is more difficult than estimating an integral under the conditional as the former implies the latter but not vice versa.

## 4 Other Algorithms for Comparison

We list some potential alternatives for posterior estimation and describe them here.

**1. MCMC - Density Estimation (MCMC-DE):** We implement MCMC with a Metropolis Hastings (MH) chain and use kernel density estimation (KDE) on the accepted points to estimate the posterior. When comparing MCMC against NED/EV we consider *the total number of queries* and not just those accepted. There are several variants of the MH proposal scheme and several tuning parameters. Comparing to all of them is nontrivial. We use MH in its basic form using a fixed Gaussian proposal distribution. Practitioners usually adjust the proposal based on the acceptance rate. Here, we chose the proposal manually by trying different values and picking the one that performed best within the queries used. Note that this comparison is advantageous to MCMC. In one experiment we test with Emcee [Foreman-Mackey *et al.*, 2013], a popular package for Affine Invariant MCMC which automatically fine tunes the proposal bandwidth based on acceptance rate [Foreman-Mackey *et al.*, 2013].

**2. MCMC - Regression (MCMC-R):** Here, as in MCMC-DE we use a MH Chain to generate the samples. However, this time we regress on the queries (not samples) to estimate the posterior. We include this procedure since MCMC can be viewed as a heuristic to explore the parameter space in high likelihood regions. We show that a principled query strategy outperforms this heuristic.

**3. Approximate Bayesian Computing (ABC):** There are several variants of ABC [Marjoram *et al.*, 2003; Peters *et al.*, 2012]. We compare with a basic form given in Marin *et al.* [2012]. At each iteration, we randomly sample  $\theta$  from the prior and then sample an observation  $\mathbf{X}_{\text{sim}}$  from the likelihood. If  $d(\mathbf{X}_{\text{sim}}, \mathbf{X}_{\text{obs}}) < \epsilon$  we add  $\theta$  to our collection. Here  $d(\cdot, \cdot)$  is some metric on a sufficient statistic of the observation and  $\epsilon > 0$  is a prespecified threshold. We perform a

KDE on the collected samples to estimate the posterior. The performance of ABC depends on  $\epsilon$ : As for MCMC-DE we choose  $\epsilon$  by experimenting with different values and choosing the value which gives the best performance within the queries used. We compare with total number of parameter values proposed and not just those retained.

**4. Uniform Random Samples (RAND):** Here, we evaluate the likelihood at points chosen uniformly on  $\Theta$  and then regress on these points.

## 5 Experiments

We first look at a series of low and high dimensional synthetic and real astrophysical experiments. NED is only tested on low dimensional problems since empirical approximation and numerical integration is computationally expensive in high dimensions. Further, since the inner expectation in Equation (4) is expensive we approximate it using a one sample estimate. We use a squared exponential kernel in all our experiments. The bandwidth for the kernel was set to be  $5n^{-1/d}$  where  $n$  is the total number of queries and  $d$  is the dimension. This was following several kernel methods (such as kernel regression) which use a bandwidth on the order  $O(n^{\frac{-c_1}{c_2+d}})$  [Györfi *et al.*, 2002]. Other kernel hyperparameters was set via cross validation every 20 iterations. In our experiments, EV slightly outperforms NED probably since the EV utility can be evaluated exactly while NED can only approximated. We omit most technical details of the experiments due to space constraints.

### Low Dimensional Synthetic Experiments:

To illustrate our methods we have two simple yet instructive experiments. In the first, the parameters space is  $\Theta = (0, 1)$  equipped with a Beta(1.2, 1) prior. We draw  $\theta$  from the prior, and then draw 500 samples from a Bernoulli( $\theta^2 + (1 - \theta)^2$ ) distribution: i.e.  $\mathbf{X}_{\text{obs}} \in \{0, 1\}^{500}$ . The ambiguity on the true value of  $\theta$  creates a bimodal posterior. Figure 3(a) compares NED/EV against the other methods as a function of the number of queries. The second is a 2D problem with  $\Theta = (0, 1)^2$ . Here we artificially created a 3-modal log-joint posterior shown by green contours in Figure 3(c). Figure 3(b) compares all methods. As we artificially constructed the log likelihood ABC does not apply here. Figure 3(c) shows the points chosen by the NED query strategy in order. It shows that we have learned the high log joint probability regions well at the expense of being uncertain at low log joint probability areas. However, this does not affect the posterior significantly as they are very small after exponentiation. Our proposed methods outperform existing methods and other heuristics by orders of magnitude on these simple experiments.

### Higher Dimensional Synthetic Datasets:

We test in  $d = 5$  and 15 dimensions. We construct an artificial log likelihood so that the resulting posterior is mixture of 2 Gaussians centred at  $\mathbf{0}$  and  $\mathbf{1}$ . We evaluate performance by the ability to estimate certain linear functionals. The exact value of these functionals can be evaluated analytically since we know the true posterior.

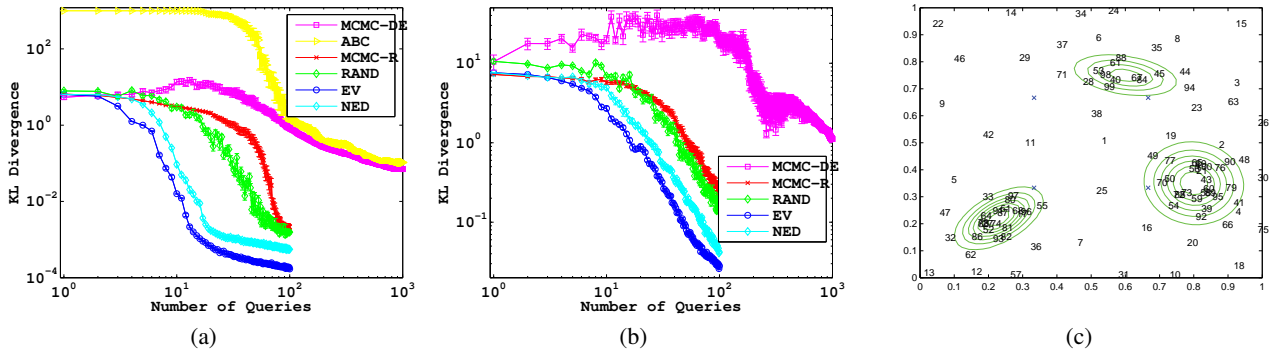


Figure 3: (a), (b): A comparison of NED/EV against MCMC-DE, ABC, MCMC-R, RAND procedures for the 1D and 2D synthetic experiments respectively. The x-axis is the number of queries and the y-axis is the KL divergence between the truth and the estimate. All figures were obtained by averaging over 60 trials. (c): The 100 points chosen by NED for the 2D experiment in the order they were queried. The green contours are the true posterior. Initially the algorithm explores the space before focusing on the high likelihood regions.

We use a uniform prior. Our log-likelihood is,  $\ell(\theta) = \log(0.5\mathcal{N}(\theta; \mathbf{0}, 0.5\sqrt{d}I) + 0.5\mathcal{N}(\theta; \mathbf{1}, 0.5\sqrt{d}I))$ . Our functionals are  $T_1 = \mathbb{E} \sum_{i=1}^d X_i$ ,  $T_2 = \mathbb{E} \sum_{i=1}^d X_i^2$ ,  $T_3 = \mathbb{E} \sum_{i=1}^{d-1} X_i^2 X_{i+1}$  and  $T_4 = \mathbb{E} \sum_{i=1}^{d-2} X_i X_{i+1} X_{i+2}$ . For MCMC-DE, we draw samples  $Z_1, Z_2, \dots$  from the true likelihood. To estimate  $T_i = \mathbb{E} \phi_i(X)$  we use the empirical estimator  $\hat{T}_i = 1/N \sum_k \phi_i(Z_k)$ . For EV, MCMC-R and RAND we first use the queried points to obtain an estimate of the log-likelihood by regressing on the likelihood values as explained before. Then we run an MCMC chain on this *estimate* to collect samples and use the empirical estimator for the functionals. Note that evaluating the estimate, unlike the likelihood, is cheap. ABC does not apply in this experiment. The results are shown in Figure 4. They demonstrate the superiority of our query strategy over the alternatives.

### Type Ia Supernovae:

We use supernovae data for inference on 3 cosmological parameters: Hubble Constant ( $H_0 \in (60, 80)$ ), Dark Matter Fraction  $\Omega_M \in (0, 1)$  and Dark Energy Fraction  $\Omega_\Lambda \in (0, 1)$ . The likelihood for the experiment is given by the Robertson–Walker metric which models the distance to a supernova given the parameters and the observed red-shift. The dataset is taken from Davis et al [2007]. The parameter space is taken to be  $\Theta = (0, 1)^3$  (For  $H_0$  we map it to  $(60, 80)$  using an appropriate linear transform). We test NED/EV against MCMC-DE, ABC, MCMC-R, RAND and Emcee. For ABC, sampling from the likelihood is as expensive as computing the likelihood. Figure 5(a) compares all methods. Figure 5(b) shows the points queried by EV and the marginals of the true posterior. The KL for RAND decreases slowly since it accumulates points at the high likelihood region very slowly. MCMC-R performs poorly since it has only explored part of the high likelihood region. For NED/EV after an initial exploration phase after which the error shoots down.

### Luminous Red Galaxies:

Here we used data on Luminous Red Galaxies (LRGs) for inference on 8 cosmological parameters. We use software and data from Tegmark et al [2006]. Our parameter space is taken to be  $(0, 1)^8$  by appropriately linear transforming the

range of the variables. Each query takes about 4-5 seconds. In EV determining the next point takes about 0.5-1 seconds with  $\approx 2000$  points and about 10-15 seconds with  $\approx 10000$  points. In this regime, EV is wall clock time competitive with other methods. ABC does not apply in this experiment.

Fig. 6a shows points queried by MCMC, RAND and EV projected on the first 2 dimensions. MCMC has several high likelihood points but its queries are focused on a small region of the space. RAND does not have many points at high likelihood regions. EV has explored the space fairly well and at the same time has several queries at high likelihood regions. Fig 6c shows the evaluated log likelihood at each query. It shows that as predicted EV first explores the space (high likelihood queries are sparse) and then exploits the high likelihood regions. Since ground truth is difficult to obtain for this experiment, we perform the following simple test. We queried 250,000 points uniformly at random from the parameter space to form a test set. We then run EV, MCMC-R and RAND for up to 12,000 queries to collect points and estimate the posterior. Performance is evaluated by the mean squared reconstruction error of the *exponentiated* log joint probabilities (joint probabilities). Figure 6b shows the results. The error for RAND and MCMC-R stay the same throughout since the problem is difficult and they did not have sufficient number of high likelihood points throughout the space.

## 6 Conclusions

We proposed a framework for query efficient posterior estimation for expensive blackbox likelihood evaluations. Our methods use GPs and are based on popular ideas in Bayesian active learning. We demonstrate that our methods outperform natural alternatives in practice.

Note that in Machine Learning it is uncommon to treat posterior estimation in a regression setting. This is probably since the estimate will depend on the intricacies of the regression algorithm. So if likelihood evaluations are inexpensive, MCMC seems a natural choice due to its theoretical guarantees in the large sample regime. However, our work demonstrates that when likelihood evaluations are expensive, such

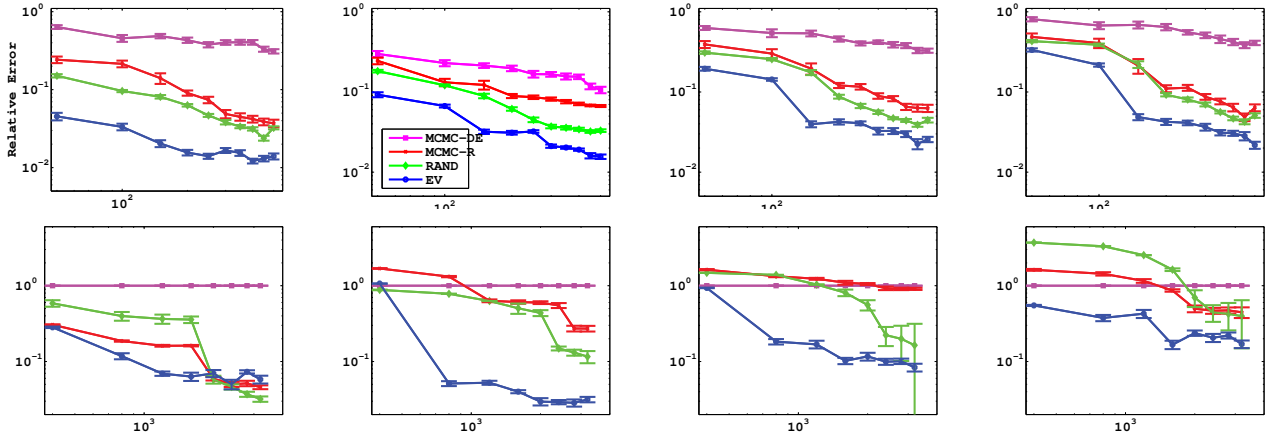


Figure 4: The first and second rows are for  $d = 5$  and  $d = 15$  and the 4 columns are for the functionals  $T_1, T_2, T_3, T_4$  respectively. The x-axis is the number of queries and the y-axis is  $|\hat{T}_i - T_i|/|T_i|$ . We go up to 500 queries for  $d = 5$  and 3200 queries for  $d = 15$ . All figures were obtained by averaging over 30 trials.

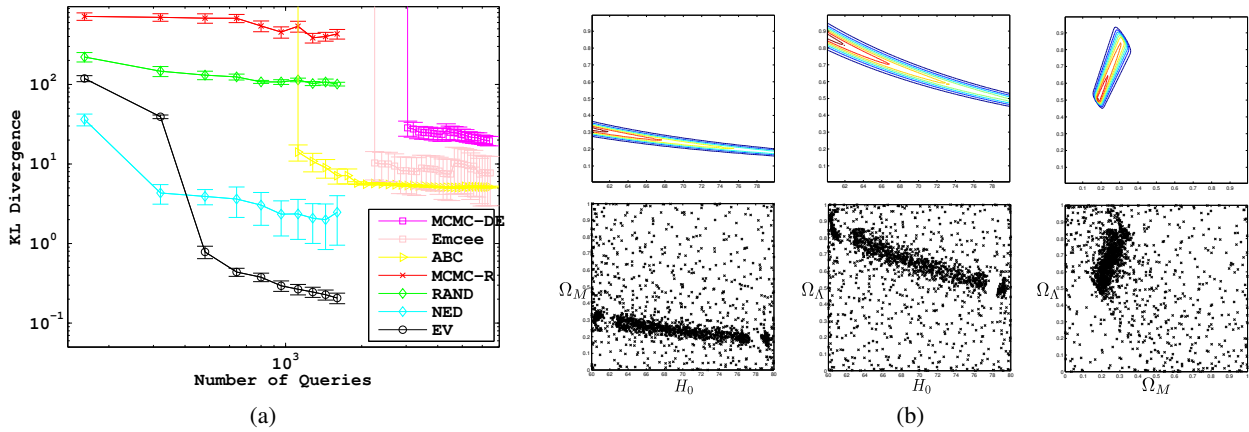


Figure 5: (a): Comparison of NED/EV against MCMC-DE, ABC, Emcee, MCMC-R and RAND. For all regression methods we show results for up to 1600 queries and up to 4 times as many for MCMC and ABC. For evaluation, KL was approximated via numeric integration on a  $(100)^3$  grid. Note that MCMC and ABC require several queries before a nontrivial KL with the truth is obtained. All curves were obtained by averaging over 30 runs. (b): Projections of the points selected by EV (bottom row) and the marginal distributions (top row).

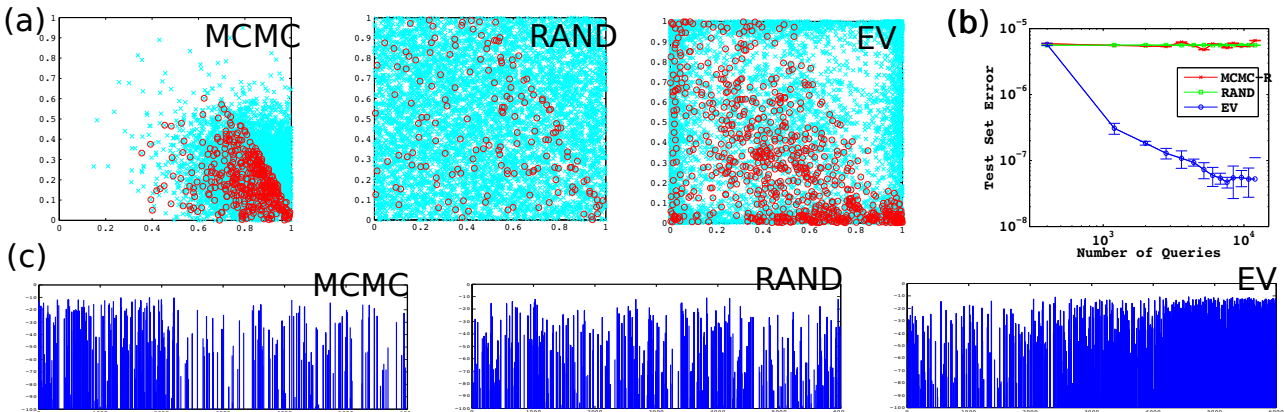


Figure 6: (a): The projections of the first 6000 points selected by MCMC, RAND and EV on to the first 2 dimensions in cyan. The points shown in red are queries at high likelihood ( $\log P > -50$ ) points. (b): Comparison of EV against MCMC-R and RAND. We use up to 12000 queries for all methods. The y-axis is the mean squared reconstruction error. The curves were obtained by averaging over 16 runs. (c): The value of the log likelihood (y-axis) obtained at each query (x-axis) for the 3 methods. Observe that in EV, initially the high likelihood evaluations are sparse—indicating exploration, and then there are several high likelihood evaluations—indicating exploitation.

as in scientific simulations, treating posterior estimation in an active regression framework enables us to be significantly query efficient.

**Acknowledgement:** This research was partly funded by DOE grant DESC0011114.

## References

- Ryan Prescott Adams, Iain Murray, and David J. C. MacKay. The Gaussian Process Density Sampler. In *NIPS*, 2008.
- Eric Brochu, Vlad M. Cora, and Nando de Freitas. A Tutorial on Bayesian Optimization of Expensive Cost Functions, with Application to Active User Modeling and Hierarchical Reinforcement Learning. *CoRR*, 2010.
- Brent Bryan, Jeff Schneider, Robert Nichol, Christopher Miller, Christopher Genovese, and Larry Wasserman. Active learning for identifying function threshold boundaries. In *Advances in Neural Information Processing Systems*. MIT Press, Cambridge, MA, 2006.
- T. M. Davis et al. Scrutinizing Exotic Cosmological Models Using ESSENCE Supernova Data Combined with Other Cosmological Probes. *The Astrophysical Journal*, pages 716–725, 2007.
- Daniel Foreman-Mackey, David W. Hogg, Dustin Lang, and Jonathan Goodman. emcee: The MCMC Hammer, January 2013.
- Kenji Fukumizu, Le Song, and Arthur Gretton. Kernel Bayes’ Rule: Bayesian Inference with Positive Definite Kernels. *Journal of Machine Learning Research*, 2014.
- Daniel Golovin and Andreas Krause. Adaptive Submodularity: Theory and Applications in Active Learning and Stochastic Optimization. *Journal of Artificial Intelligence Research (JAIR)*, 2011.
- Alkis Gotovos, Nathalie Casati, Gregory Hitz, and Andreas Krause. Active Learning for Level Set Estimation. In *IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013*, 2013.
- Tom Gunter, Michael A. Osborne, Roman Garnett, Philipp Hennig, and Stephen J. Roberts. Sampling for Inference in Probabilistic Models with Fast Bayesian Quadrature. In *Advances in Neural Information Processing Systems*, 2014.
- László Györfi, Micael Kohler, Adam Krzyzak, and Harro Walk. *A Distribution Free Theory of Nonparametric Regression*. Springer Series in Statistics, 2002.
- Kirthevasan Kandasamy, Jeff Schneider, and Barnabás Póczos. High Dimensional Bayesian Optimisation and Bandits via Additive Models. In *International Conference on Machine Learning*, 2015.
- Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies. *J. Mach. Learn. Res.*, 2008.
- David Landau and Kurt Binder. *A Guide to Monte Carlo Simulations in Statistical Physics*. Cambridge University Press, 2005.
- Jun S. Liu. *Monte Carlo strategies in Scientific computing*. Springer, 2001.
- Yifei Ma, Roman Garnett, and Jeff Schneider. Active Area Search via Bayesian Quadrature. In *International Conference on Artificial Intelligence and Statistics*, 2014.
- David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press, 2003.
- Jean-Michel Marin, Pierre Pudlo, Christian P. Robert, and Robin J. Ryder. Approximate Bayesian computational methods. *Statistics and Computing*, 2012.
- Paul Marjoram, John Molitor, Vincent Plagnol, and Simon Tavaré. Markov Chain Monte Carlo without Likelihoods. *Proceedings of the National Academy of Sciences of the United States of America*, 2003.
- J.B. Mockus and L.J. Mockus. Bayesian approach to global optimization and application to multiobjective and constrained problems. *Journal of Optimization Theory and Applications*, 1991.
- M. Osborne, D. Duvenaud, R. Garnett, C. Rasmussen, S. Roberts, and Z. Ghahramani. Active Learning of Model Evidence Using Bayesian Quadrature. In *Neural Information Processing Systems*, 2012.
- David Parkinson, Pia Mukherjee, and Andrew R Liddle. A Bayesian model selection analysis of WMAP3. *Physical Review*, D73:123523, 2006.
- Gareth W. Peters, Y. Fan, and Scott A. Sisson. On sequential Monte Carlo, partial rejection control and approximate Bayesian computation. *Statistics and Computing*, 2012.
- C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. Adaptive computation and machine learning series. University Press Group Limited, 2006.
- Christian P. Robert and George Casella. *Monte Carlo Statistical Methods (Springer Texts in Statistics)*. Springer-Verlag New York, Inc., 2005.
- Sambu Seo, Marko Wallat, Thore Graepel, and Klaus Obermayer. Gaussian Process Regression: Active Data Selection and Test Point Rejection. In *International Joint Conference on Neural Networks*, 2000.
- Burr Settles. Active Learning Literature Survey. Technical report, University of Wisconsin-Madison, 2010.
- John Skilling. Nested sampling for general Bayesian computation. *Bayesian Anal.*, 2006.
- Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. Gaussian Process Optimization in the Bandit Setting: No Regret and Experimental Design. In *International Conference on Machine Learning*, 2010.
- M. Tegmark et al. Cosmological Constraints from the SDSS Luminous Red Galaxies. *Physical Review*, December 2006.