Stochastic thermodynamic integration: efficient Bayesian model selection via stochastic gradient MCMC

Umut Şimşekli, Roland Badeau, Gael Richard, Ali Taylan Cemgil

STOCHASTIC THERMODYNAMIC INTEGRATION: EFFICIENT BAYESIAN MODEL SELECTION VIA STOCHASTIC GRADIENT MCMC

Umut Şimşekli[1]*, Roland Badeau*[1]*, Gaël Richard*[1]*, Ali Taylan Cemgil*[2]

1: LTCI, CNRS, Télécom ParisTech, Université Paris-Saclay, 75013, Paris, France
2: Dept. of Computer Engineering, Boğaziçi University, 34342, Bebek, İstanbul, Turkey

ABSTRACT

Model selection is a central topic in Bayesian machine learning, which requires the estimation of the marginal likelihood of the data under the models to be compared. During the last decade, conventional model selection methods have lost their charm as they have high computational requirements. In this study, we propose a computationally efficient model selection method by integrating ideas from Stochastic Gradient Markov Chain Monte Carlo (SG-MCMC) literature and statistical physics. As opposed to conventional methods, the proposed method has very low computational needs and can be implemented almost without modifying existing SG-MCMC code. We provide an upper-bound for the bias of the proposed method. Our experiments show that, our method is 40 times as fast as the baseline method on finding the optimal model order in a matrix factorization problem.

Index Terms— Bayesian model selection, Markov Chain Monte Carlo, Non-negative matrix factorization

1. INTRODUCTION

Model selection is an important topic in various fields. The aim in this problem is to choose the best model that describes the data from a collection of models. In Bayesian statistics, model selection is formulated as computing the Bayes factor which requires to compute the marginal likelihood of the data under the models to be compared, that is given as follows:

$$p(x|m) = \int p(x|\theta, m)p(\theta|m)d\theta \tag{1}$$

where $x \equiv \{x_n\}_{n=1}^N$ is the observed data whose elements are assumed to be independent and identically distributed (i.i.d.), $m \in \{1, \dots, M\}$ denotes different models, and θ is a latent variable. Here, $p(x|\theta, m)$ is the likelihood function of model m and $p(\theta|m)$ is the prior distribution of θ. In Bayesian model selection, we aim to find the model with the highest marginal likelihood:

$$m^\star = \arg\max_m \int p(x|\theta, m)p(\theta|m)d\theta \tag{2}$$

where we need to evaluate the marginal likelihood for each model. A canonical example for model selection can be given as finding the optimal model order in a polynomial regression problem while avoiding over-fitting, where m would correspond to different degrees of polynomials (e.g., linear, quadratic, cubic, etc.).

This work is partly supported by the French National Research Agency (ANR) as a part of the EDISON 3D project (ANR-13-CORD-0008-02). A.T.C. is supported by TÜBİTAK 113M492 (Pavera) and Boğaziçi University BAP 10360-15A01P1.

For notational simplicity, we drop the model variable m and consider the following equation: $p(x) = \int p(x|\theta)p(\theta)d\theta$. Computing the marginal likelihood requires to integrate the joint distribution over all model parameters, which turns out to be intractable except for very few special cases. Therefore, in practice approximate methods are utilized for estimating the marginal likelihood.

Markov Chain Monte Carlo (MCMC) techniques are one of the most popular approaches that are used in marginal likelihood estimation [1–3]. However, despite their well known advantages, these methods have lost their charm in various machine learning applications especially during the last decade, as they are perceived to be computationally very demanding. Indeed, the conventional approaches require passing over the whole data set at each iteration, which makes the methods impractical even for mediocre N. Recently, alternative approaches, under the name of *stochastic gradient MCMC* (SG-MCMC), have been proposed, aiming to develop computationally efficient MCMC methods that can scale up to large-scale regime [4–11]. Unlike conventional MCMC methods, these methods require to 'see' only a small subset of the data per iteration, which enables the methods to handle large datasets.

Even though SG-MCMC techniques are easily applicable to a wide variety of probabilistic models, it is not straightforward to develop model selection algorithms that are based on these methods. Therefore, the majority of the current literature focuses on improving the prediction accuracy of these methods in various large-scale applications, whereas efficient model selection algorithms based on SG-MCMC are yet to be explored.

In this study, we propose a novel marginal likelihood estimation method, namely Stochastic Thermodynamic Integration (STI), by integrating ideas from SG-MCMC literature and thermodynamic integration; a family of marginal likelihood estimation methods commonly used in statistical physics. As opposed to conventional model selection methods, STI has very low computational requirements thanks to data subsampling and it can be implemented almost without modifying existing SG-MCMC code as we will describe in detail in the following sections, where we also provide an upper-bound for the bias induced by STI. Our experiments on a speech enhancement application show that STI is able to find the optimal model order in a matrix factorization model in 9 minutes on a standard laptop computer, whereas the baseline method requires 6 hours for the same problem.

2. TECHNICAL BACKGROUND

2.1. Stochastic Gradient Langevin Dynamics

An important attempt for scaling up MCMC techniques was made by Welling and Teh [4], where the authors combined the ideas from statistical physics and stochastic optimization, and developed a scalable

MCMC framework called the stochastic gradient Langevin dynamics (SGLD). SGLD exploits the assumption that the data samples x_n are i.i.d. and it asymptotically generates a sample $\theta^{(k)}$ from the posterior distribution $p(\theta|x) \propto p(\theta)p(x|\theta)$ by iteratively applying the following update equation [4]:

$$\theta^{(k)} = \theta^{(k-1)} + \epsilon^{(k)}\Big(\frac{N}{N_s}\sum_{n\in S^{(k)}} \nabla \log p\big(x_n|\theta^{(k-1)}\big) + \nabla \log p\big(\theta^{(k-1)}\big)\Big) + \eta^{(k)} \quad (3)$$

where $S^{(k)} \subset \{1,\dots,N\}$ is a random data subsample that is drawn with or without replacement, $N_s = |S^{(k)}|$ is the number of data points in $S^{(k)}$, $\epsilon^{(k)}$ is the step-size, and $\eta^{(k)}$ is Gaussian noise: $\eta^{(k)} \sim \mathcal{N}(\eta^{(k)}; 0, 2\epsilon^{(k)}\mathbb{I})$ where $\mathbb{I}$ stands for the identity matrix. The step-size can be fixed or decreasing. A typical choice for decreasing step-size is $\epsilon^{(k)} = (a_\epsilon/k)^{b_\epsilon}$, where $a_\epsilon > 0$ and $b_\epsilon \in (0.5, 1]$. Several extensions of SGLD have been proposed [5–11].

2.2. Thermodynamic Integration

In this study, we consider a particular family of methods for estimating the marginal likelihood, called *path sampling* or *thermodynamic integration* (TI) [1]. TI forms a continuous *path* between two unnormalized densities, say $q_0(\theta)$ and $q_1(\theta)$ by introducing a *temperature* parameter $t \in [0, 1]$. A typical choice is forming a *geometric path* [1, 3], that is given as follows:

$$q(\theta|t) = q_0(\theta)^{1-t} q_1(\theta)^t \quad (4)$$

where $q(\theta|t = 0) = q_0(\theta)$ and $q(\theta|t = 1) = q_1(\theta)$. The main approach in TI is to choose $q_0(\theta)$ in such a way that its normalizing constant $z_0 = \int q_0(\theta)d\theta$ is known and to choose $q_1(\theta)$ as the distribution whose normalizing constant $z_1 = \int q_1(\theta)d\theta$ is to be estimated.

In this study, we consider *power posteriors* [3], where $q_0(\theta)$ is selected as the prior distribution $p(\theta)$ and $q_1(\theta)$ is selected as the unnormalized posterior $p(x|\theta)p(\theta)$. This choice imposes a specific form on $q(\theta|t)$ that is called the power posterior:

$$q(\theta|t) = p(\theta)p(x|\theta)^t. \quad (5)$$

Since we choose $q_0(\theta)$ as the prior distribution, we know that $z_0 = \int p(\theta)d\theta = 1$, and $z_1 = \int p(x|\theta)p(\theta)d\theta = p(x)$ is the marginal likelihood that we would like to compute. It is easy to verify that the following identity holds [3]:

$$\log p(x) = \log \frac{z_1}{z_0} = \int_0^1 \big\langle \log p(x|\theta)\big\rangle_{p(\theta|t)}\, dt \quad (6)$$

where $\big\langle f(\theta)\big\rangle_{\pi(\theta)} = \int f(\theta)\pi(\theta)d\theta$ denotes the expectation of $f(\theta)$ under $\pi(\theta)$, $p(\theta|t) = [1/z(t)]p(\theta)p(x|\theta)^t$ with $z(t) = \int p(\theta)p(x|\theta)^t d\theta$. Several approaches can be devised for approximately computing Eq. 6 [1]. One possible approach would be using numerical techniques for approximating the integration over t and MCMC simulations for approximating the expectations. In this study, we consider the approach given in [3], which approximates Eq. 6 by first discretizing t as $0 = t_0 < t_1 < \cdots < t_T = 1$ and using a trapezoidal rule for numeric integration, yielding the following equation: ($\Delta t_i = t_{i+1} - t_i$)

$$\log p(x) \approx \sum_{i=0}^{T-1} \Delta t_i \frac{\big\langle \log p(x|\theta)\big\rangle_{p(\theta|t_{i+1})} + \big\langle \log p(x|\theta)\big\rangle_{p(\theta|t_i)}}{2} \quad (7)$$

where the expectations are computed by using MCMC:

$$\big\langle \log p(x|\theta)\big\rangle_{p(\theta|t)} \approx \frac{1}{K}\sum_{k=1}^{K}\sum_{n=1}^{N} \log p(x_n|\theta^{(t,k)}) \quad (8)$$

Here, $\theta^{(t,k)}$ denotes samples drawn from $p(\theta|t)$.

3. STOCHASTIC THERMODYNAMIC INTEGRATION

Even though MCMC inference has been made much more efficient with the incorporation of stochastic gradients, marginal likelihood estimation methods that are based on MCMC still suffer from high computational complexity since they typically require the likelihood to be computed on the whole dataset for each sample (see Eq.8).

Inspired by the ideas from stochastic gradient MCMC and path sampling methods, in this study, we propose a novel method for marginal likelihood estimation that is based on data subsampling, called Stochastic Thermodynamic Integration (STI). STI follows almost the same derivations as we described in Section 2.2; however, instead of evaluating the log-likelihood on the whole dataset, it uses an *unbiased* estimator of the log-likelihood that is computed on a subsample of the data $S^{(t,k)}$, given as follows:

$$\big\langle \log p(x|\theta)\big\rangle_{p(\theta|t)} \approx \frac{1}{K}\frac{N}{N_s}\sum_{k=1}^{K}\sum_{n\in S^{(t,k)}} \log p(x_n|\theta^{(t,k)}). \quad (9)$$

Since STI is based on random subsamples, it can be easily integrated with any subsample-based MCMC method for generating the samples $\theta^{(t,k)}$. In this study, for simplicity we choose SGLD for generating the samples $\theta^{(t,k)}$, whereas SGLD can be replaced with any proper SG-MCMC method [9]. The SGLD update rule for generating samples from the power posteriors is almost identical to Eq. 3 and given as follows:

$$\theta^{(t,k)} = \theta^{(t,k-1)} + \epsilon^{(t,k)}\Big(\frac{N}{N_s}t\sum_{n\in S^{(t,k)}} \nabla \log p\big(x_n|\theta^{(t,k-1)}\big) + \nabla \log p\big(\theta^{(t,k-1)}\big)\Big) + \eta^{(t,k)}. \quad (10)$$

Having SGLD in its core, STI yields a very simple yet powerful algorithm, where a sample is generated by using Eq. 10 and the log-likelihood is immediately evaluated by using Eq. 9. These estimates are then used in Eq. 7 for computing the marginal likelihood. A useful property of STI is that the same subsample $S^{(t,k)}$ can be used both for generating the random sample $\theta^{(t,k)}$ and evaluating the log-likelihood, which increases the efficiency of the method and makes the method suitable for large-scale distributed problems.

Since we have multiple sources of stochasticity, it is not immediately clear how much bias is induced by STI. In fact, it is not even clear whether the estimates provided in Eq. 9 would converge to true expectations. However; fortunately, by the law of total expectation, we can still show that the estimates obtained via Eq. 9 converge to the true expected values (see [10, 12, 13]), since STI makes use of an unbiased estimator of the log-likelihood . Based on this observation, we provide the following theorem which forms a bound for the overall bias induced by STI with fixed step-size.

Theorem 1. *Let* $\mathcal{L} \triangleq \int_0^1 f(t)dt$ *be the log-marginal likelihood (Eq. 6) with* $f(t) \triangleq \big\langle \log p(x|\theta)\big\rangle_{p(\theta|t)}$ *and* $\hat{\mathcal{L}}$ *be the estimator obtained via STI (Eqs. 7 and 9). Assume that* $\{x_n\}_{n=1}^N$ *is i.i.d.,*

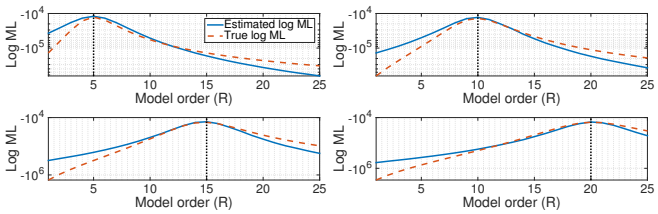

Fig. 1. Results of the synthetic data experiments conducted on the simple Gaussian model. The vertical lines show the true value of R.

$\log p(x,\theta)$ is differentiable, $f(t)$ is twice differentiable and its second derivative is uniformly bounded, i.e., $|f''(t)| < U$ for $t \in [0,1]$ and for some $U > 0$. The domain of the temperature variable t is uniformly discretized, i.e., $\Delta t_i = (1/T)$ for all $i = 0, 1, \ldots, T-1$, and $\theta^{(t,k)}$ is generated by an SG-MCMC method [9] with constant step-size ϵ (Eq.10). We further assume that $\log p(x|\theta)$ satisfies the conditions given in Assumption 1 described in [10]. Then, the bias of STI can be bounded as:

$$\left|\langle\hat{\mathcal{L}}\rangle - \mathcal{L}\right| = \mathcal{O}\Big(\frac{1}{K\epsilon} + \epsilon + \frac{1}{T^2}\Big). \tag{11}$$

The proof is given in the supplementary document [14]. Note that the theorem applies to the general case of STI, i.e., it covers any proper SG-MCMC method that can be used within STI (see [9, 10]), whereas SGLD appears as a special case.

4. EXPERIMENTS

4.1. Experiments on Synthetic Data

4.1.1. Gaussian Additive Model

In this section, we evaluate STI on a simple model whose marginal likelihood is analytically available. The model is given as follows:

$$\theta_r \sim \mathcal{N}(\theta_r; \mu_\theta, \sigma_\theta^2), \quad x_n|\theta \sim \mathcal{N}(x_n; \sum\nolimits_{r=1}^{R} \theta_r, \sigma_x^2) \tag{12}$$

where $\theta = \{\theta_r\}_{r=1}^R$ is the collection of the latent variables and $x = \{x_n\}_{n=1}^N$ denotes the observations. Here, each observation x_n is generated from a Gaussian distribution whose mean is the sum of R i.i.d. Gaussian latent variables. We consider the case where R is not known a-priori. Therefore, in order to determine the best R, we estimate the marginal likelihood of the data for M different values of R: $p(x|R_m) = \int p(x|\theta)p(\theta|R_m)d\theta$ for all $m \in \{1, \ldots, M\}$.

In these experiments, for several true R values, we generate θ and x by using the generative model. Then, we estimate the marginal likelihood for different R values by using STI and compare these estimates with the true marginal likelihood. Here, we set $\mu_\theta = 5$, $\sigma_\theta^2 = 3$, $\sigma_x^2 = 5$, and $N = 5000$. For t, we discretize the interval $[0, 1]$ into $T = 10$ points in a regular fashion: $t_i - t_{i-1} = t_{i+1} - t_i$ for all admissible i. At each epoch, we use only $N_s = 250$ observations for drawing samples and evaluating the log-likelihood. We generate $K = 3000$ samples at each SGLD run where we discard the first 1000 samples as burn-in. For the step-size of SGLD, we set $a_\epsilon = 10^{-8}$ and $b_\epsilon = 0.51$, and keep the step-size fixed after burn-in.

Fig.1 shows the results. We can observe that, in all cases, the mode of the marginal likelihood coincides with the true value of R and the estimates provided by STI are very accurate especially when R_m is close to the mode. These results show that, as opposed to the conventional methods that need to use the whole data set for generating samples and evaluating the likelihoods, STI provides very accurate estimations with much less computational needs. The computational advantage will be illustrated more clearly in the sequel.

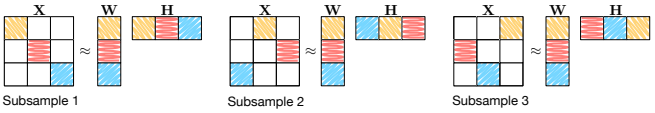

Fig. 2. Illustration of PSGLD. Given the blocks in a subsample, the corresponding blocks in W and H become conditionally independent, as illustrated in different textures.

4.1.2. Non-negative Matrix Factorization

Non-negative matrix factorization (NMF) [15] is an important modeling tool in data analysis and has been shown to be useful in various domains, such as recommender systems, audio processing, finance, computer vision, and bioinformatics [16–18]. The aim of the NMF model is to decompose an observed *non-negative* data matrix $X \in \mathbb{R}_+^{I\times J}$ into the form: $X \approx WH$, where $W \in \mathbb{R}_+^{I\times R}$ and $H \in \mathbb{R}_+^{R\times J}$ are the non-negative factor matrices, typically known as the *dictionary* and the *activation* matrices, respectively. In this study, we consider a particular NMF model that has the following probabilistic generative model [19]:

$$W_{ir} \sim \mathcal{E}(W_{ir}; \lambda_w), \quad H_{rj} \sim \mathcal{E}(H_{rj}; \lambda_h)$$
$$X_{ij}|W_{i:}, H_{:j} \sim \mathcal{PO}\Big(X_{ij}; \sum\nolimits_{r=1}^{R} W_{ir}H_{rj}\Big) \tag{13}$$

where $\mathcal{E}$ and $\mathcal{PO}$ denote the exponential and Poisson distributions, respectively. In this context, we have $x = \{X_{ij}\}_{i,j}$ with $N = IJ$ and $\theta = \{W_{:r}, H_{r:}\}_{r=1}^R$. Here R determines the *rank* of the factorization, which is typically unknown and determined manually.

In this section, we evaluate STI on the estimation of the rank variable R in Poisson NMF. For matrix factorization models, the computational complexity of STI can be reduced even more by modifying SGLD in such a way that the update rule given in Eq. 10 can be run in parallel [20–22]. In this study, we make use of Parallel SGLD (PSGLD) [20] that exploits the conditional independence structure of the matrix factorization models. The main idea in PSGLD is that it utilizes a biased subsampling schema where the data is carefully partitioned into mutually disjoint blocks and the latent factors are also partitioned accordingly. This approach is illustrated in Figure 2. At each iteration, PSGLD subsamples multiple blocks from X, in such a way that these blocks do not 'touch' each other in any dimension of X. This biased subsampling schema enables parallelism, since given a subsample, the SGLD updates can be applied to different blocks of the latent factors in parallel.

We use an experimental setting similar to that we described in Section 4.1.1: we generate W, H, and X by using the generative model for two different values of R. Then, we estimate the marginal likelihood for different R values by using STI. We set $I = 100$, $J = 75$, $\lambda_w = \lambda_h = 5$. For inference, we choose $T = 5$ and we generate $K = 10000$ samples at each PSGLD run where we use the last 2000 samples for computing the expectations. We partition X into 5×5 blocks (i.e., $N_s = IJ/5$) and set $a_\epsilon = 10^{-5}$ and $b_\epsilon = 0.51$, and keep the step-size fixed after burn-in. .

Unfortunately, the marginal likelihood of the Poisson NMF model is analytically not available. Therefore, we compare STI with a popular marginal likelihood estimation algorithm, namely Chib's method [2]. This method estimates the marginal likelihood by using the samples obtained from a Gibbs sampler and has been shown

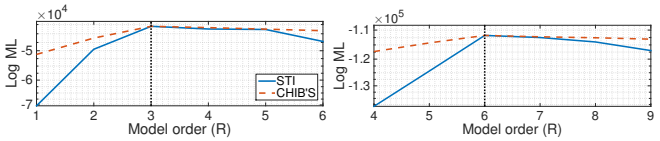

Fig. 3. Results of the synthetic data experiments conducted on the NMF model. The vertical lines show the true value of R.

to be useful for matrix and tensor factorization models [19, 23]. In order be able to obtain the full conditional distributions that are required by the Gibbs sampler, we need to introduce an auxiliary tensor and augment the model in Eq. 13 as follows [19, 24]:

$$C_{ijr}|W_{ir}, H_{rj} \sim \mathcal{PO}(C_{ijr}; W_{ir}H_{rj}), \quad X_{ij} = \sum_{r=1}^{R} C_{ijr},$$

where the prior distributions remain unchanged. For Chib's method, we first generate 9000 samples with the Gibbs sampler, where we discard the first 7000 of them as burn-in. Then, we generate 5000 more samples for certain computations required by the method. These experiments are conducted on a standard laptop computer with 2.5GHz Quad-core Intel Core i7 CPU. The methods are implemented in C, where we use GSL and BLAS for the matrix operations, and OpenMPI for parallel computing.

Fig. 3 shows the results. It can be seen that the estimates obtained by both methods are similar, especially near the mode. Similarly to the previous set of experiments, the discrepancy between these estimates becomes larger at the tails. We can assume that Chib's method is more accurate at these regions given that it uses the whole data set at each epoch and enjoys the conjugacy of the model. Nevertheless, the shapes of the estimations are quite similar; the modes of the marginal likelihood coincide with the true values of R, which is crucial for model selection applications.

The key advantage of STI over Chib's method appears in the computation time. Even though the number of samples generated by STI is 3 times the number of samples generated by Chib's method, thanks to the usage of the subsamples, STI is 6 times as fast as Chib's method: even for these rather simple problems, Chib's method takes 835 seconds to compute the marginal likelihood for 10 different values of R, whereas STI finishes all the computations in 137 seconds. Besides, since the Gibbs sampler requires generating N multinomial random variables of size R at each epoch, Chib's method becomes even more impractical for large R. On the other hand, STI is also more efficient than Chib's method in terms of space complexity: Chib's method requires most of the samples to be stored whereas STI only needs to store the latest sample.

4.2. Experiments on Audio

In this section, we evaluate STI on a speech enhancement application, where the aim is to recover the clean speech signal, given a noisy speech signal. Here, we consider a semi-supervised approach and model the *magnitude spectrogram* of the noisy mixture as:

$$X_{ij}|\cdot \sim \mathcal{PO}\Big(X_{ij}; \sum_{r=1}^{R^{\text{sp}}} W_{ir}^{\text{sp}} H_{rj}^{\text{sp}} + \sum_{r=1}^{R^{\text{no}}} W_{ir}^{\text{no}} H_{rj}^{\text{no}}\Big) \quad (14)$$

where i denotes frequencies, j denotes time-frames, 'sp' denotes speech, and 'no' denotes noise. In this setting, the usual approach is to train the dictionary matrix W^{sp} on a clean speech corpus by using the model given in Eq.13 and to fix it during denoising time, in which all the other variables are estimated. Since the noise signals usually do not have much variation, in practice it is sufficient to set

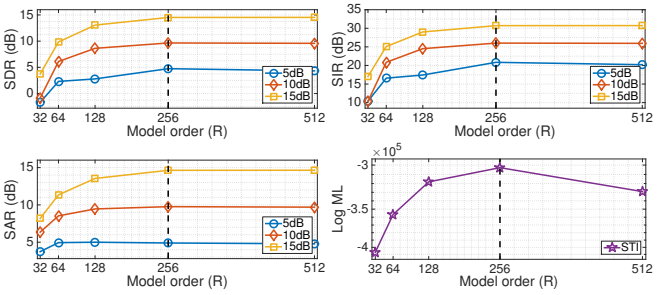

Fig. 4. Results of the speech enhancement experiments. In the first three plots, different colors represent different mixing SNRs.

R^{no} to a small value. However, it is known that the enhancement performance heavily relies on the rank of the speech dictionary [25]. In this section, we evaluate STI on automatic determination of R^{sp}.

We conduct our experiments on NOIZEUS noisy speech corpus [26]. This dataset contains 30 sentences that are uttered by 3 female and 3 male speakers (i.e. 5 sentences per speaker). These sentences are corrupted by using 8 different real noise signals at 4 different signal-to-noise ratio (SNR) levels. We analyze the signals by using the short-time Fourier transform with a Hamming window of length 512 samples and 50% overlap. We follow a speaker- and gender-independent approach, and use the first 20 clean speech signals (2 female, 2 male) as the training corpus, which yields a matrix X of size 257×1661. Then, we estimate the marginal likelihood by using STI for 5 different values of R^{sp}: $2^5, \ldots, 2^9$. We set $T = 5$ and generate $K = 1250$ samples at each PSGLD run where we use the last 500 samples for the computations. We partition X into 8×8 blocks and set $a_\epsilon = 5 \times 10^{-7}$, $b_\epsilon = 0.51$, and $\lambda_w = \lambda_h = 0.0004$.

We compare the performance of STI with *oracle* results: we first train W^{sp} using the Expectation-Maximization (EM) algorithm [15,19] for each R^{sp} value. Then, by fixing W^{sp} and setting $R^{\text{no}} = 5$, we evaluate the models on the noisy mixtures that are obtained by corrupting clean speech signals that are not used during training. The quality of the enhancement is measured by the signal-to-distortion ratio (SDR), signal-to-interference ratio (SIR), and signal-to-artifact ratio (SAR) that are computed with BSS$_{\text{EVAL}}$ version 3.0 [27].

Fig. 4 shows the results. We can observe that the quality of the enhancement increases as we increase R^{sp} up to 256, and after that point, increasing R^{sp} does not improve the enhancement performance. This outcome is correctly captured by STI; the mode of the marginal likelihood increases until $R^{\text{sp}} = 256$, then increasing R^{sp} results in lower marginal likelihood. As opposed to conventional cross-validation methods that require training and testing for each R^{sp}, STI is able to find the correct model order without needing a validation set. On the other hand, STI computes the marginal likelihood for 5 different R^{sp} values in only 9 minutes, whereas Chib's method becomes impractical for this problem since it requires approximately 6 hours for generating 1250 samples for 5 different values of R^{sp}.

5. CONCLUSION

In this study, we proposed STI, a novel method for marginal likelihood estimation by integrating ideas from SG-MCMC literature and statistical physics. STI has very low computational needs and can be implemented almost without modifying existing code. We provided a bound for the bias of STI. We showed that STI is 40 times as fast as the baseline method on finding the optimal model order in a matrix factorization problem.

6. REFERENCES

[1] A. Gelman and X. L. Meng, "Simulating normalizing constants: from importance sampling to bridge sampling to path sampling," *Statist. Sci.*, vol. 13, no. 2, pp. 163–185, 05 1998.

[2] S. Chib, "Marginal likelihood from the Gibbs output," *Journal of the American Statistical Association*, vol. 90, no. 432, pp. 1313–1321, 1995.

[3] N. Friel and A. N. Pettitt, "Marginal likelihood estimation via power posteriors," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 70, no. 3, pp. 589–607, July 2008.

[4] M. Welling and Y. W Teh, "Bayesian learning via stochastic gradient Langevin dynamics," in *Proceedings of the 28th International Conference on Machine Learning*, June 2011, pp. 681–688.

[5] S. Ahn, A. Korattikara, and M. Welling, "Bayesian posterior sampling via stochastic gradient Fisher scoring," in *International Conference on Machine Learning*, June 2012.

[6] S. Patterson and Y. W. Teh, "Stochastic gradient Riemannian Langevin dynamics on the probability simplex," in *Advances in Neural Information Processing Systems*, Dec. 2013.

[7] T. Chen, E. B. Fox, and C. Guestrin, "Stochastic gradient Hamiltonian Monte Carlo," in *Proc. International Conference on Machine Learning*, June 2014.

[8] N. Ding, Y. Fang, R. Babbush, C. Chen, R. D. Skeel, and H. Neven, "Bayesian sampling using stochastic gradient thermostats," in *Advances in Neural Information Processing Systems*, Dec. 2014, pp. 3203–3211.

[9] Y. A. Ma, T. Chen, and E. Fox, "A complete recipe for stochastic gradient MCMC," in *Advances in Neural Information Processing Systems*, 2015, pp. 2899–2907.

[10] C. Chen, N. Ding, and L. Carin, "On the convergence of stochastic gradient MCMC algorithms with high-order integrators," in *Advances in Neural Information Processing Systems*, 2015, pp. 2269–2277.

[11] X. Shang, Z. Zhu, B. Leimkuhler, and A. J. Storkey, "Covariance-controlled adaptive langevin thermostat for large-scale bayesian sampling," in *Advances in Neural Information Processing Systems*, 2015, pp. 37–45.

[12] I. Sato and H. Nakagawa, "Approximation analysis of stochastic gradient Langevin dynamics by using Fokker-Planck equation and Ito process," in *Proceedings of the 31st International Conference on Machine Learning*. June 2014, pp. 982–990, JMLR Workshop and Conference Proceedings.

[13] Y. W. Teh, A. Thiéry, and S. Vollmer, "Consistency and fluctuations for stochastic gradient Langevin dynamics," *arXiv preprint arXiv:1409.0578*, 2014.

[14] U. Şimşekli, R. Badeau, G. Richard, and A. T. Cemgil, "Stochastic thermodynamic integration: Efficient Bayesian model selection via stochastic gradient MCMC: Supplementary Document," `https://hal.archives-ouvertes.fr/hal-01248011`.

[15] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization.," *Nature*, vol. 401, pp. 788–791, 1999.

[16] P. Smaragdis and J. C. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, Oct. 2003, pp. 177–180.

[17] K. Devarajan, "Nonnegative matrix factorization: An analytical and interpretive tool in computational biology," *PLoS Computational Biology*, vol. 4, 2008.

[18] A. Cichoki, R. Zdunek, A. H. Phan, and S. Amari, *Nonnegative Matrix and Tensor Factorization*, Wiley, 2009.

[19] A. T. Cemgil, "Bayesian inference in non-negative matrix factorisation models," *Computational Intelligence and Neuroscience*, 2009.

[20] U. Şimşekli, H. Koptagel, H. Güldaş, A. T. Cemgil, F. Öztoprak, and Ş. İ. Birbil, "Parallel stochastic gradient Markov Chain Monte Carlo for matrix factorisation models," *arXiv preprint arXiv:1506.01418*, 2015.

[21] S. Ahn, A. Korattikara, N. Liu, S. Rajan, and M. Welling, "Large-scale distributed Bayesian matrix factorization using stochastic gradient MCMC," in *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Aug. 2015.

[22] U. Şimşekli, *Tensor Fusion: Learning in Heterogeneous and Distributed Data*, Ph.D. thesis, Boğaziçi University, 2015.

[23] U. Şimşekli and A. T. Cemgil, "Markov chain Monte Carlo inference for probabilistic latent tensor factorization," in *2012 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, Sept. 2012, pp. 1–6.

[24] C. Févotte and A. T. Cemgil, "Nonnegative matrix factorisations as probabilistic inference in composite models," in *European Signal Processing Conference*, Aug. 2009, pp. 1913–1917.

[25] X. Jaureguiberry, E. Vincent, and G. Richard, "Multiple-order non-negative matrix factorization for speech enhancement," in *Interspeech*, May 2014, p. 4.

[26] Y. Hu and P. C. Loizou, "Subjective comparison and evaluation of speech enhancement algorithms," *Speech communication*, vol. 49, no. 7, pp. 588–601, 2007.

[27] E. Vincent, H. Sawada, P. Bofill, S. Makino, and J. P. Rosca, "First stereo audio source separation evaluation campaign: data, algorithms and results," in *Independent Component Analysis and Signal Separation*, pp. 552–559. Springer, Sept. 2007.