Fast Inference for Intractable Likelihood Problems using Variational Bayes

Article · September 2016

CITATIONS
2

READS
87

3 authors:

David Gunawan
UNSW Sydney
28 PUBLICATIONS **27** CITATIONS
SEE PROFILE

Minh-Ngoc Tran
The University of Sydney
65 PUBLICATIONS **511** CITATIONS
SEE PROFILE

Robert Kohn
UNSW Sydney
225 PUBLICATIONS **7,043** CITATIONS
SEE PROFILE

Fast Inference for Intractable Likelihood Problems using Variational Bayes

David Gunawan* Minh-Ngoc Tran† Robert Kohn‡

Abstract

Variational Bayes (VB) is a popular statistical method for Bayesian inference. The existing VB algorithms are restricted to cases where the likelihood is tractable, which precludes their use in many interesting models. Tran et al. (2015) extend the scope of application of VB to cases where the *likelihood* is intractable but can be estimated unbiasedly, and name the method "Variational Bayes with Intractable Likelihood (VBIL)". This paper presents a version of VBIL, named Variational Bayes with Intractable Log-Likelihood (VBILL), that is useful for cases, such as big data and big panel data models, where only unbiased estimators of the *log-likelihood* are available. In particular, we develop an estimation approach, based on subsampling and the MapReduce programming technique, for analysing massive datasets which cannot fit into a single desktop's memory. The proposed method is theoretically justified in the sense that, apart from an extra Monte Carlo error which can be controlled, it is able to produce estimators as if the true log-likelihood or full data were used. The proposed methodology is robust in the sense that it works well when only highly variable estimates of the log-likelihood are available. The method is illustrated empirically using several simulated datasets and a big real dataset based on the arrival time status of U. S. airlines.

Keywords. Pseudo Marginal Metropolis-Hastings, Debiasing Approach, Big Data, Panel Data, Difference Estimator.

1 Introduction

Given an observed dataset y and a statistical model with a vector of unknown parameters θ, a major aim of statistics is to carry out inference about θ, i.e., estimate the underlying θ that generated y and assess the associated uncertainty. The likelihood function $p(y|\theta)$, which is the density of the data y conditional on the postulated model and the parameter vector θ, is the cornerstone of many statistical procedures. Most of the popular likelihood-based methodologies, such as Maximum Likelihood Estimation, Markov chain Monte Carlo (MCMC), Importance Sampling and Variational Bayes, require exact evaluations of the likelihood $p(y|\theta)$ at each value of θ.

*School of Economics, UNSW Business School, david.gunawan@unsw.edu.au
†Business Analytics, University of Sydney Business School, minh-ngoc.tran@sydney.edu.au
‡School of Economics, UNSW Business School, r.kohn@unsw.edu.au

In many modern statistical applications, however, the likelihood function is either analytically intractable or computationally intractable, which makes it difficult to use likelihood-based methodologies.

An important situation in which the log-likelihood is computationally intractable is Big Data (Wang et al., 2015), where the log-likelihood function, under the independence assumption, is a sum of a very large number of terms that is too expensive to compute. Large panel data models (Fitzmaurice et al., 2011) are another example where the log-likelihood is both analytically and computationally intractable as it is a sum of many terms, each of which is the log of an integral over the random effects and cannot be computed analytically.

There are several methods in the literature that work with an intractable likelihood. A remarkable approach is the pseudo-marginal Metropolis-Hastings (PMMH) algorithm (Andrieu and Roberts, 2009), which replaces the likelihood in the Metropolis-Hastings ratio by its non-negative unbiased estimator and is able to generate samples from the posterior. Like standard Metropolis-Hastings algorithms, PMMH is extremely flexible. However, this method is highly sensitive to the variance of the likelihood estimator, the chain might get stuck and mix poorly if the likelihood estimates are highly variable (Flury and Shephard, 2011). This is because the asymptotic variance of PMMH estimators increases exponentially with the variance of likelihood estimator (Pitt et al., 2012). The PMMH method can be computationally expensive and is not parallelizable, which makes it unsuitable for Big Data applications.

This paper develops fast and efficient methodologies for statistical inference based on VB, with a special focus on computational efficiency and challenging situations such as Big Data and in particular Big Panel Data - a mainstream area of research in statistics and its related fields in the next decade. The existing VB algorithms are restricted to cases where the likelihood is tractable, which precludes the use of VB in many interesting models. Tran et al. (2015) extend the scope of application of VB to cases where the likelihood is intractable but can be estimated unbiasedly, and name the method "Variational Bayes with Intractable Likelihood" (VBIL). Their method works with non-negative unbiased estimators of the likelihood, and is useful in cases such as state space models and small panel data models, where it is easier and more efficient to obtain unbiased estimates of the likelihood than the log-likelihood. This paper presents a version of VBIL, called the Variational Bayes with Intractable Log-Likelihood (VBILL), that is useful for cases, such as big data and big panel data models, where only an unbiased estimator of the log-likelihood is available. Working with an unbiased estimator of the log-likelihood, which is a sum under the independence assumption of observations, has the advantage of being able to use subsampling techniques from the survey sampling literature to obtain efficient estimates of the log-likelihood (Quiroz et al., 2015). It is important to note that both PMMH and VBIL require the likelihood estimator to be non-negative almost surely, which rules out many interesting applications where an unbiased likelihood estimator exists but can take on negative values (Jacob and Thiery, 2015). VBILL does not impose any constraints on the sign of the log-likelihood estimator.

Our paper also makes use of the recent MapReduce programming technique and develops an approach for analysing massive datasets which do not fit into a single desktop's memory. The implementation of MapReduce uses the divide and combine idea where the data is divided into small chunks, each chunk is processed separately

and the chunk-based results are then combined to construct the final estimates. Under some regularity conditions, Battey et al. (2015) show that the information loss due to the divide and combine procedure is asymptotically negligible when the full sample size grows, as long as the number of chunks is not too large. In finite-sample settings, however, the resulting estimators are sensitive to how the data are divided. It is important to note that our final estimator is mathematically justified and independent of the data chunking, as we use the divide and combine procedure mainly to obtain an unbiased estimator of the log-likelihood.

The link between the precision of the log-likelihood estimator to the variance of the VBILL estimator is also studied. This helps us to understand the properties of our estimator when working with an estimated log-likelihood compared to the case where the log-likelihood is available. It is shown that the asymptotic variance of VBILL estimators increases linearly with the variance of the unbiased log-likelihood estimator. Unlike PMMH, our proposed methodology still works well when only highly variable estimators of the log-likelihood are available as shown in some simulated and real data examples.

The paper is organised as follows. Section 2 introduces the VBILL algorithm that works with an unbiased estimator of the log-likelihood. In particular, we describe an efficient scheme based on subsampling to approximate accurately the posterior distribution in Big Data. Section 3 discusses some theoretical properties of the proposed method. Section 4 applies the proposed method to analysing the US Airlines big dataset. Section 5 outlines applications of VBILL to big panel data models and presents some simulation studies. Section 6 concludes. An appendix discusses variance reduction methods, the natural gradient and exponential families, and provides proofs.

2 Variational Bayes with Intractable Log-Likelihood (VBILL)

Let $p(\theta)$ be a prior, and $\pi(\theta) \propto p(\theta)p(y|\theta)$ the posterior distribution defined on the space $\Theta \subset \mathbb{R}^d$. In almost all cases the posterior $\pi(\theta)$ does not have a standard form which makes it difficult to perform inference about θ. Variational Bayes (VB) is increasingly used as a computationally effective method for approximating the posterior distribution $\pi(\theta)$ (Bishop, 2006; Ormerod and Wand, 2010; Nott et al., 2012). VB approximates the posterior by a distribution $q_\lambda(\theta)$ within some easily accessible class, such as an exponential family, with parameter λ chosen to minimise the Kullback-Leibler divergence $KL(q_\lambda \| \pi)$ between $q_\lambda(\theta)$ and $\pi(\theta)$,

$$KL(\lambda) = KL(q_\lambda \| \pi) := \int q_\lambda(\theta) \log \frac{q_\lambda(\theta)}{\pi(\theta)} d\theta.$$

Let $\hat{l}(\theta) = \hat{l}(\theta, \gamma)$ be an unbiased estimator of the log-likelihood $l(\theta) = \log\ p(y|\theta)$, where γ denotes all the random variables used to compute $\hat{l}(\theta)$. Denote by $g(\gamma|\theta)$ the density of γ. The gradient of the Kullback-Leibler divergence between the variational

distribution $q_\lambda(\theta)$ and the posterior $\pi(\theta) = p(\theta)p(y|\theta)/p(y)$ is

$$\begin{aligned}
\nabla_\lambda KL\left(q_\lambda \| \pi\right) &= \nabla_\lambda \int q_\lambda\left(\theta\right) \log \frac{q_\lambda\left(\theta\right)}{\pi\left(\theta\right)} d\theta \\
&= \int q_\lambda\left(\theta\right) \nabla_\lambda \left[\log q_\lambda\left(\theta\right)\right] \left(\log q_\lambda\left(\theta\right) - \log\left(p\left(\theta\right) p\left(y|\theta\right)\right)\right) d\theta \\
&= E_{\theta\sim q_\lambda}\left\{\nabla_\lambda \left[\log q_\lambda\left(\theta\right)\right] \left(\log q_\lambda\left(\theta\right) - \log\left(p\left(\theta\right)\right) - \log\left(p\left(y|\theta\right)\right)\right)\right\} \\
&= E_{\theta\sim q_\lambda(\theta), \gamma\sim g(\gamma|\theta)}\left\{\nabla_\lambda \left[\log q_\lambda\left(\theta\right)\right] \left(\log q_\lambda\left(\theta\right) - \log\left(p\left(\theta\right)\right) - \hat{l}\left(\theta, \gamma\right)\right)\right\}.
\end{aligned}$$

By generating $\theta \sim q_\lambda(\theta)$ and $\gamma \sim g(\gamma|\theta)$, i.e. computing the estimate $\hat{l}\left(\theta\right)$, we are able to obtain an unbiased estimator $\widehat{\nabla_\lambda KL}\left(q_\lambda \| \pi\right)$ of the gradient $\nabla_\lambda KL\left(q_\lambda \| \pi\right)$. Therefore, we can use stochastic optimisation to optimise $\mathrm{KL}(\lambda)$. The following is the basic algorithm.

Algorithm 1.
- *Initialize* $\lambda^{(0)}$ *and stop the following iteration if the stopping criterion is met.*
- *For* $t = 0, 1, ...,$ *compute* $\lambda^{(t+1)} = \lambda^{(t)} - a_t \widehat{\nabla_\lambda KL}\left(\lambda^{(t)}\right)$.

The sequence $\{a_t, t \geq 0\}$ is the learning rate and should satisfy $a_t > 0$, $\sum_t a_t = \infty$ and $\sum_t a_t^2 < \infty$. We choose $a_t = 1/(1+t)$ in this paper. It is also possible to train a_t adaptively. Algorithm 1 is parallelisable within each iteration as the gradient is estimated by importance sampling. The performance of Algorithm 1 mainly depends on the variance of the noisy gradient $\mathbb{V}\left(\widehat{\nabla_\lambda KL}(\lambda)\right)$. As in Tran et al. (2015), we employ a range of methods, such as control variates and factorisation to reduce the variance of the gradient estimator. We also employ the natural gradient that takes into account the geometry of the variational density, which makes the convergence faster (Amari, 1998; Hoffman et al., 2013). The details can be found in Tran et al. (2015) and in the Appendix 7.1.

2.1 Stopping Criterion and Marginal Likelihood Estimation

The log of the marginal likelihood can be expressed as

$$\log p\left(y\right) = LB\left(\lambda\right) + KL\left(\lambda\right), \tag{2.1}$$

where the lower bound $LB\left(\lambda\right)$ is

$$LB\left(\lambda\right) := \mathbb{E}_{\theta\sim q_\lambda(\theta)}\left[\log p\left(\theta\right) - \log q_\lambda\left(\theta\right)\right] + \mathbb{E}_{\theta\sim q_\lambda(\theta), \gamma\sim g(\gamma|\theta)}\left(\hat{l}\left(\theta, \gamma\right)\right). \tag{2.2}$$

The first term in equation (2.2) is often computed in closed form, while the second term can be easily estimated unbiasedly by samples generated $\theta \sim q_\lambda(\theta)$ and $\gamma \sim g(\gamma|\theta)$. It is clear from equation (2.1) that minimising $KL\left(\lambda\right)$ is equivalent to maximising the lower bound $LB\left(\lambda\right)$. As in Tran et al. (2015), the updating algorithm is stopped if the change in an average value of the lower bounds over a window of K iterations $\overline{LB}\left(\lambda^{(t)}\right) = (1/K)\sum_{k=1}^{K} \widehat{LB}\left(\lambda^{(t-k+1)}\right)$, is less than some threshold ϵ, where $\widehat{LB}\left(\lambda\right)$ is an unbiased estimate of $LB\left(\lambda\right)$. At convergence, the

lower bound $LB(\lambda)$ is often used as a good approximation of the log of marginal likelihood $\log p(y)$, which is an important quantity for model selection (Sato, 2001; Nott et al., 2012). Note in general it is not clear if $KL \approx 0$ at convergence, but LB is still useful for model selection because in general $LB(\mathcal{M}_1)$ is relatively larger than $LB(\mathcal{M}_2)$ if model $\mathcal{M}_1$ is closer to the true model than $\mathcal{M}_2$.

2.2 VBILL with Data Subsampling and the Difference Estimator

Let $y = \{y_i, i = 1, ..., n\}$ be the data set. We assume that the likelihood is $p(y|\theta) = \prod_{i=1}^{n} p(y_i|\theta)$. Then the log-likelihood is given by

$$l(\theta) := \sum_{i=1}^{n} l_i(\theta), \quad \text{where} \quad l_i(\theta) = \log p(y_i|\theta). \tag{2.3}$$

We are concerned with the case where the log-likelihood is computationally intractable. This is the case of Big Data, where n is so big that computing the full sum over n terms is not practical. Another situation is a panel data model, where n may not be too big but computing each $l_i(\theta)$ is very expensive. It will be cheaper to obtain an unbiased estimator $\hat{l}(\theta)$ of the log-likelihood $l(\theta)$ based on a small random subset of the full data y. Here, we propose using VBILL with data subsampling and the difference estimator. Quiroz et al. (2015) use simple random sampling from the survey sampling literature combined with the difference estimator to obtain an unbiased estimator of the log-likelihood. This estimator subtracts an approximation $w_i(\theta)$ of the log-likelihood contribution $l_i(\theta)$ from each log-likelihood contribution to obtain a new population with elements that are roughly of the same size.

Write the log-likelihood as

$$\begin{aligned} l(\theta) &= \sum_{i \in F} w_i(\theta) + \sum_{i \in F} [l_i(\theta) - w_i(\theta)] \\ &= w + d. \end{aligned}$$

with

$$w := \sum_{i \in F} w_i(\theta), d := \sum_{i \in F} d_i(\theta), d_i(\theta) = l_i(\theta) - w_i(\theta),$$

where the set $F = \{1, 2, ..., n\}$ is the index set of all observations in the full dataset. Here, $w = \sum_{i \in F} w_i(\theta)$ is known for a given θ and the difference estimator is obtained by estimating d. In our case, the $w_i(\theta)$ are evaluated once at $\theta = \overline{\theta}$, where $\overline{\theta}$ is obtained using Maximum Likelihood, simulated Maximum Likelihood, etc; this could be based on the full data set or, for speed, a representative subset. Since $w_i(\theta)$ is an approximation of $l_i(\theta)$, the quantity $l_i(\theta) - w_i(\theta)$ should have roughly the same size for all i. We can therefore use simple random sampling with replacement (SIR) to estimate d

$$\widehat{d}_m = \frac{1}{m} \sum_{i=1}^{m} n d_{u_i},$$

where $\mathbf{u} = (u_1, ..., u_m)$, $u_i \in F$, is the $m \times 1$ vector of indices obtained by doing simple random sampling with replacement from the full dataset F. It is then easy

to show that

$$\mathbb{E}\left(\widehat{d}_m\right) = d.$$

Therefore, the difference estimator

$$\widehat{l}_m(\theta) := w + \widehat{d}_m \tag{2.4}$$

is an unbiased estimator of log-likelihood $l(\theta)$. It is much cheaper to compute $\widehat{l}_m(\theta)$ than the full data log-likelihood $l(\theta)$. This therefore provides a fast and highly efficient computational method for Big Data. The Appendix 7.2 presents an approach to estimate the variance $\mathbb{V}\left(\widehat{l}_m(\theta)\right)$.

3 Convergence Properties

Suppose that the equation $\nabla_\lambda \mathrm{KL}(\lambda) = 0$ has the unique solution λ^*. Let $\widehat{\lambda}_M$ be the estimator of λ^* obtained by Algorithm 1 after M iterations, and $\widetilde{\lambda}_M$ be the corresponding estimator obtained when the exact log-likelihood is available. Denote $\zeta_*(\theta) = \nabla_\lambda[\log q_\lambda(\theta)]\big|_{\lambda=\lambda^*}$ and denote by $\mathbb{E}_*(\cdot)$ and $\mathbb{V}_*(\cdot)$ the expectation and variance operators with respect to $q_{\lambda^*}(\theta)$. For simplicity, we consider the case that λ is scalar; the case with a multivariate λ can be obtained using Theorem 5 of Sacks (1958). We obtain the following results whose proof is in the Appendix.

Let $\sigma^2(\theta) := \mathbb{V}(\widehat{l}(\theta,\gamma)|\theta)$.

Theorem 1. *Suppose that the regularization conditions in Theorem 1 of Sacks (1958) hold.*
(i) Then,

$$\sqrt{M}(\widehat{\lambda}_M - \lambda^*) \xrightarrow{d} \mathcal{N}\Big(0, c_{\lambda^*}\mathbb{V}\big(\widehat{\nabla_\lambda \mathrm{KL}}(\lambda^*)\big)\Big), \text{ as } M \to \infty, \tag{3.1}$$

where c_{λ^*} is a positive constant that is independent of the random variables involved in estimating $\nabla_\lambda \mathrm{KL}(\lambda^*)$.
(ii) Let $\sigma^2_{asym}(\widehat{\lambda}_M) = c_{\lambda^*}\mathbb{V}\big(\widehat{\nabla_\lambda \mathrm{KL}}(\lambda^*)\big)$ be the asymptotic variance of $\widehat{\lambda}_M$ as $M \to \infty$. Similarly, let $\sigma^2_{asym}(\widetilde{\lambda}_M)$ be the asymptotic variance of $\widetilde{\lambda}_M$. Then,

$$\sigma^2_{\mathrm{asym}}(\widehat{\lambda}_M) = \sigma^2_{\mathrm{asym}}(\widetilde{\lambda}_M) + \frac{c_{\lambda^*}}{S}\mathbb{E}_*\Big\{\zeta_*^2(\theta)\sigma^2(\theta)\Big\}. \tag{3.2}$$

where S is the number of samples (θ_i,γ_i) used to compute the noisy gradient $\widehat{\nabla_\lambda KL}(\lambda)$.

Tran et al. (2015) obtain similar results under the assumption that the variance of log of the estimated likelihood $\mathbb{V}(\log\widehat{p}(y|\theta))$ is constant. We do not require such an assumption for Theorem 1 to hold. However, it is easier to understand the meaning of the results in Theorem 1 if we assume, for pedagogical purpose, that the number of quasi-random number in γ is tuned such that the variance of the log-likelihood estimator $\sigma^2(\theta) = \mathbb{V}\Big(\widehat{\log p(y|\theta)}\Big)$ is a constant σ^2. Then, it follows from equation (3.2) that the variance of VBILL estimators increases only linearly with σ^2. This suggests that VBILL still works well when only highly variable estimates of the log-likelihood are available.

4 Application: the US airlines data

The airline on-time performance data from the 2009 ASA Data Expo is used as an example to demonstrate our proposed methodology with a massive dataset that exceeds the memory (RAM) of a single computer. This dataset was used by Wang et al. (2015) and Kane et al. (2013). It consists of the flight arrival and departure details for all commercial flights within the USA, from October 1987 to April 2008. The full sample ignoring the missing values is 22,347,358 observations.

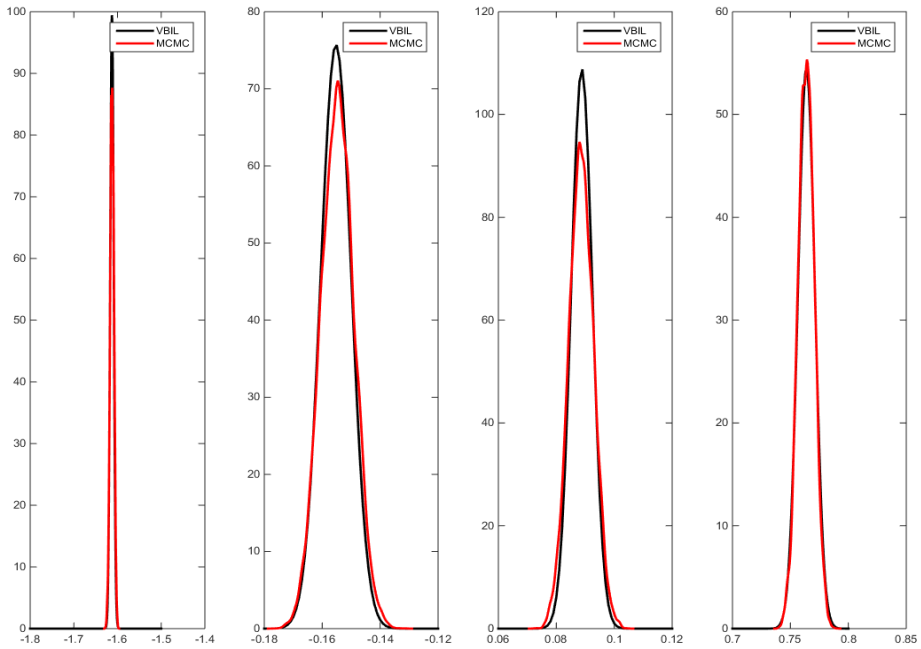
The response variable of the logistic regression model is late arrival, which was set to 1 if a flight was late by more than 15 minutes and 0 otherwise. There are three covariates. The two binary covariates are: `night` (1 if departure occurred at nights and 0 otherwise) and `weekend` (1 if departure occurred on weekends and 0 otherwise). One continuous covariate `distance` is also included, which is the distance from origin to destination (in 1000's of miles).

We first compare the performance of VBILL with data subsampling and the difference estimator to MCMC for a subset of one million observations from the full dataset. The MCMC chain consists of 30000 iterates with 10000 burn-in iterates. For the VBILL algorithm, the variance of the log-likelihood estimator $\mathbb{V}\left(\hat{l}_m(\theta)\right)$ can be set to a large value, as long as it is not too large for the stochastic search procedure to fail to converge. Given a prespecified maximum variance V_{max}, the subsample size m is adapted so that $\mathbb{V}\left(\hat{l}_m(\theta)\right)$ is never larger than V_{max}. The strategy is to increase m whenever $\mathbb{V}\left(\hat{l}_m(\theta)\right) > V_{max}$ until $\mathbb{V}\left(\hat{l}_m(\theta)\right) < V_{max}$. The formula to calculate the variance $\mathbb{V}\left(\hat{l}_m(\theta)\right)$ is given in Appendix 7.2. We set $V_{max} = 1000$ in this example after some experiment. VBILL uses around 1% of the full one million observations on average at each iteration and converges within a few iterations. This example is run on a single desktop with 4 local processors. Table 1 shows the estimates of the posterior mean and posterior variance (shown in brackets) as well as the running time for both VBILL and MCMC methodologies. As shown, the VBILL estimates are very close to the "gold standard" MCMC estimates, but VBILL is 27 times faster than MCMC in this small data example. Figure 4.1 also shows that the posterior density estimates from VBILL and MCMC are very close to each other.

Table 1: Logistic Model Estimation Results

Parameter	VBILL	CPU time (in mins)	MCMC	CPU time (in mins)
$\beta_{intercept}$	-1.6127 (0.0040)	0.5728	-1.6125 (0.0044)	15.4617
$\beta_{distance}$	-0.1553 (0.0053)		-0.1548 (0.0057)	
β_{night}	0.0888 (0.0037)		0.0886 (0.0043)	
$\beta_{weekend}$	0.7638 (0.0073)		0.7637 (0.0070)	

Figure 4.1: Comparisons of estimates of VBILL and MCMC for the Logistic model



We now run the VBILL for full dataset that exceeds the memory of a single desktop computer. We use the MapReduce programming technique in Matlab to process this big dataset. MapReduce is available in the R2014b release of Matlab. The MapReduce function requires three input arguments:

- A `datastore` function for reading the dataset into the "map" function in a chunk-wise fashion.
- A `map` function calculates the quantities of interest for each individual chunk of data. The MapReduce calls the map function one time for each chunk of the dataset stored in datastore.
- A `reduce` function aggregates outputs from the map function and produces final results.

We apply the MapReduce programming technique to estimate the log-likelihood unbiasedly. The `datastore` function splits the full dataset into K chunks, each fits into the memory of a single desktop computer. The log-likelihood in equation (2.3) is decomposed correspondingly as

$$l(\theta) = \sum_{k=1}^{K} l^{(k)}(\theta),$$

where $l^{(k)}$ is the log-likelihood contribution based on data chunk k. Recall that $\widehat{l}_m(\theta)$ is the unbiased log-likelihood estimator based on a random subset of size m from the full dataset. In the same vein, we denote by $\widehat{l}_{m_k}(\theta)$ the unbiased estimator of

$l^{(k)}(\theta)$, based on a random subset of size m_k from data chunk k. The m_k's satisfy $m_1+...+m_K=m$, typically $m_k=m/K$. The `map` function is used to calculate the chunk based estimate $\widehat{l}_{m_k}(\theta)$ for each chunk k in the same manner as described in Section 2.2. The `reduce` function aggregates all the chunk-based unbiased log-likelihood estimates into the full data based unbiased log-likelihood estimate

$$\widehat{l}_m(\theta)=\sum_{k=1}^{K}\widehat{l}_{m_k}(\theta). \tag{4.1}$$

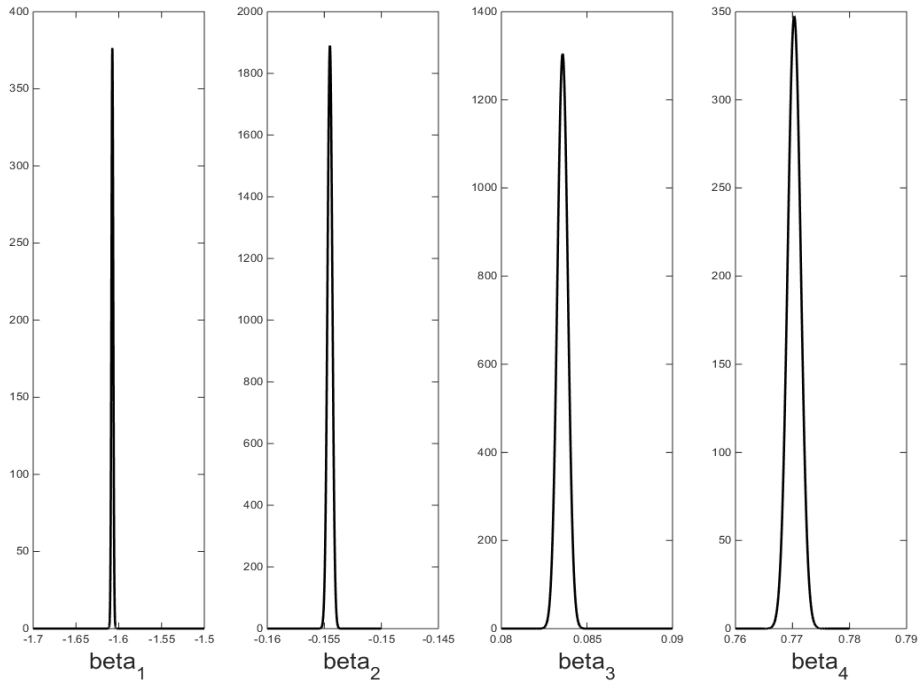
It is obvious that $\mathbb{E}\Big(\widehat{l}_m(\theta)\Big)=\sum_{k=1}^{K}\mathbb{E}\Big(\widehat{l}_{m_k}(\theta)\Big)=\sum_{k=1}^{K}l^{(k)}(\theta)=l(\theta)$. We note this method is computer-memory efficient in the sense that the full dataset does not need to remain on-hold, and provides a highly efficient computational method for Big Data. It is important to note that our VBILL estimator is mathematically justified and independent of data chunking, as the estimator $\widehat{l}_m(\theta)$ in equation (4.1) is guaranteed to be unbiased.

This example is run on an Intel Core i7 3.6 GHz desktop supported by the Matlab Parallel Toolbox with 4 local processors and the MapReduce built-in function. Given the maximum variance of the log-likelihood estimator $V_{max}=1000$, we use approximately 5% of the data in each subset. The VBILL algorithm converges within a few iterations. The CPU times taken to run the VBILL is 178.10 minutes. Although we use MapReduce to estimate the difference estimator and hence to obtain unbiased estimator of the log-likelihood so that the statistical properties of our estimator are still mathematically justified, we note that the time taken is a lot larger than the 1 million observations case. This is due to the communication cost between each data subset every time we estimate the difference estimator $\hat{d}_m$. The communication cost can be reduced by having a smaller number of subsets K. Figure 4.2 shows the marginal posterior density estimates of the parameters, which are bell shaped with very small variance as expected with a very large dataset. Table 2 shows the parameter estimates from the logistic model for the full sample. This example confirms that the VBILL methodology with data subsampling and the difference estimator is useful for Bayesian inference in Big Data.

Table 2: Logistic Model Estimation Results for full sample

Parameter	VBILL	CPU time (in mins)
$\beta_{intercept}$	-1.6075 (0.0011)	178.0959
$\beta_{distance}$	-0.1545 (0.0002)	
β_{night}	0.0836 (0.0003)	
$\beta_{weekend}$	0.7703 (0.0011)	

Figure 4.2: Marginal Posterior Estimates for the Logistic model for the full sample



5 Application: Big Panel Data Models

In random effects panel data models with n panels $\{y_1,...,y_n\}$, the likelihood is

$$p(y|\theta) = \prod_{i=1}^{n} p(y_i|\theta) = \prod_{i=1}^{n} \int p(y_i|\alpha_i,\theta) p(\alpha_i|\theta) d\alpha_i. \tag{5.1}$$

It is clear that with a very large number of panels, it is very expensive to compute the likelihood $p(y|\theta)$ at each value of θ. Here, the likelihood is intractable, but can be estimated unbiasedly using importance sampling (IS). Suppose that $p(y_i|\theta)$ is estimated unbiasedly by IS as

$$\widehat{p}_{n_k}(y_i|\theta) = \frac{1}{n_k} \sum_{j=1}^{n_k} w\left(\alpha_i^{(j)}\right), \; w\left(\alpha_i^{(j)}\right) = \frac{p\left(y_i|\alpha_i^{(j)},\theta\right) p\left(\alpha_i^{(j)}|\theta\right)}{g\left(\alpha_i^{(j)}|\theta,y_i\right)}, \tag{5.2}$$

where $\alpha_i^{(j)} \sim g\left(\alpha_i^{(j)}|\theta,y_i\right)$ for $j = 1,..,n_k$ and for some proposal density $g(\alpha_i|\theta,y_i)$ such that $\mathbb{V}(w) < \infty$ and n_k is the number of important samples. However, in order to use the VBILL algorithm, we need an unbiased estimator of the log-likelihood contribution $l_i(\theta)$. It is clear that $\log \widehat{p}_{n_k}(y_i|\theta)$ is a biased estimator of $l_i(\theta)$. This section describes two approaches to obtain unbiased or nearly unbiased estimators of the log-likelihood $l_i(\theta)$.

5.1 Exact Debiasing Approach

General methods to obtain unbiased estimators from a sequence of biased estimators, referred to as "exact debiasing approach", have been developed by McLeish (2012) and Rhee and Glynn (2015).

Let λ be an unknown constant that we want to estimate and let ζ_k, $k=0,1,...$ be a sequence of biased estimators of λ, such that it is possible to generate ζ_k for each k. We are interested in constructing an unbiased estimator $\widehat{\lambda}$ of λ, i.e. $\mathbb{E}(\widehat{\lambda})=\lambda$, based on the ζ_k's, so that $\widehat{\lambda}$ has a finite variance. We now present the debiasing approach, proposed independently by McLeish (2012) and Rhee and Glynn (2015), for constructing such a $\widehat{\lambda}$. The basic idea is to introduce randomization into the sequence $\{\zeta_k, k=0,1,2,...\}$ to eliminate the bias.

Proposition 1. *[Theorem 1 of Rhee and Glynn (2015)]: Suppose that T is a non-negative integer-valued random variable such that $P(T\geq k)>0$ for any $k=0,1,2,...$, and that T is independent of the ζ_k's. Let $\varpi_k := 1/P(T\geq k)$. If*

$$\sum_{k=1}^{\infty}\varpi_k\mathbb{E}\big((\zeta_{k-1}-\lambda)^2\big)<\infty, \tag{5.3}$$

then

$$\widehat{\lambda}=\zeta_0+\sum_{k=1}^{T}\varpi_k(\zeta_k-\zeta_{k-1}) \tag{5.4}$$

is an unbiased estimator of λ and has the finite variance

$$\mathbb{V}\Big(\widehat{\lambda}\Big)=\sum_{k=1}^{\infty}\varpi_k\big(\mathbb{E}\big((\zeta_{k-1}-\lambda)^2\big)-\mathbb{E}\big((\zeta_k-\lambda)^2\big)\big)-\mathbb{E}\big((\zeta_0-\lambda)^2\big)<\infty. \tag{5.5}$$

Unbiased estimators obtained using current debiasing approach can take negative values with a positive probability, even if their expectations are known to be non-negative. See Jacob and Thiery (2015) for a detailed discussion. This debiasing estimator may not be suitable for PMMH and VBIL since they require that the likelihood estimator must be non-negative almost surely.

Although $\widehat{\lambda}$ is an unbiased estimator of λ, its variance can be large. We can reduce this variance by averaging over replications of $\widehat{\lambda}$, $\overline{\widehat{\lambda}}=\Big(\widehat{\lambda}_1+...+\widehat{\lambda}_{nrep}\Big)/nrep$, with the $\widehat{\lambda}_i$ independent replications of $\widehat{\lambda}$. Doing this also gives us an estimate of $\mathbb{V}\Big(\widehat{\lambda}\Big)$,

$$\widehat{\mathbb{V}}\Big(\hat{\lambda}\Big)=\frac{1}{nrep-1}\sum_{i=1}^{nrep}\Big(\hat{\lambda}_i-\overline{\hat{\lambda}}\Big)^2.$$

Then $\widehat{\mathbb{V}}\Big(\overline{\widehat{\lambda}}\Big)=\widehat{\mathbb{V}}\Big(\widehat{\lambda}\Big)/nrep$ is an estimator of the variance of $\overline{\widehat{\lambda}}$.

We now apply this exact debiasing approach to obtain unbiased estimators $\hat{l}_i(\theta)$ of $l_i(\theta)$. Assume that the proposal density g in equation (5.2) is sufficiently heavy tailed so that

$$\sigma_i^2(\theta)=\frac{\mathbb{V}(w)}{p(y_i|\theta)^2}<\infty.$$

Then

$$\epsilon_{i,n_k} := \sqrt{n_k}\left(\frac{\widehat{p}_{n_k}(y_i|\theta)}{p(y_i|\theta)} - 1\right)/\sigma_i(\theta)$$

has zero mean and unit variance, and is approximately normal $\mathcal{N}(0,1)$ as n_k grows. Let $\zeta_k = \log\widehat{p}_{n_k}(y_i|\theta)$,

$$\begin{aligned}\zeta_k - l_i(\theta) &= \log\left(1+\frac{\sigma_i(\theta)\epsilon_{i,n_k}}{\sqrt{n_k}}\right) \\ &= \frac{\sigma_i(\theta)\epsilon_{i,n_k}}{\sqrt{n_k}} - \frac{\sigma_i^2(\theta)\epsilon_{i,n_k}^2}{2n_k} + \frac{\sigma_i^3(\theta)\epsilon_{i,n_k}^3}{3n_k\sqrt{n_k}} + \ldots \end{aligned} \tag{5.6}$$

So

$$(\zeta_k - l_i(\theta))^2 = \frac{\sigma_i^2(\theta)\epsilon_{i,n_k}^2}{n_k} - \frac{\sigma_i^3(\theta)\epsilon_{i,n_k}^3}{n_k\sqrt{n_k}} + \ldots \tag{5.7}$$

Therefore, when n_k is large enough

$$\mathbb{E}\big((\zeta_k - l_i(\theta))^2\big) = \frac{\sigma_i^2(\theta)}{n_k} + o\big(n_k^{-1}\big).$$

Let τ be a number such that $0 < \tau < 1$. Define

$$n_k = \lceil \frac{1}{\tau^k \Pr(T \geq k+1)} \rceil \tag{5.8}$$

with $\lceil x \rceil$ being the smallest integer that is larger than or equal to x. Then, condition (5.3) is satisfied. Therefore,

$$\hat{l}_i(\theta) := \zeta_0 + \sum_{k=1}^{T} \varpi_k(\zeta_k - \zeta_{k-1})$$

is an unbiased estimator of $l_i(\theta)$.

Choosing Distribution for T. A possible choice of the distribution of T is a negative binomial distribution $\Pr(T=k) = \rho(1-\rho)^k$, $k = 0,1,\ldots$ for $0 < \rho < 1$. Then $\varpi_k = 1/\Pr(T \geq k) = 1/(1-\rho)^k$. The number of importance samples n_k for each of the ζ_k from equation (5.8) is

$$n_k = \lceil \frac{1}{\tau^k (1-\rho)^{k+1}} \rceil.$$

If k is large, then n_k can be so large that it can freeze the computer. Table 3 shows the number of importance samples n_k for different k and the probability $\Pr(T \geq k+1)$ when $\tau = 0.9$ and $\rho = 0.2$. When $k = 40$, n_k is too large to handle and it happens once every 10000 iterations. It is important to make n_k increase slowly while the condition in Proposition 1 is still guaranteed. One way to solve this problem is by constructing the distribution T carefully.

Table 3: n_k when $\tau=0.9$ and T is negative binomial with parameters 1 and $\rho=0.2$

k	n_k	$\Pr(T\geq k+1)$
10	34	0.0859
20	892	0.0092
30	23820	0.0009
40	636225	0.0001
50	16993593	1.1418×10^{-5}
60	453900047	1.2260×10^{-6}
70	1.2124×10^{10}	1.3164×10^{-7}

In Proposition 1, T can be any non-negative integer-valued random variable such that $\Pr(T\geq k)>0$ for any $k=0,1,2,....$ The idea here is to construct distribution T such that the number of important samples n_k increases slowly and is not too large to handle computationally. We propose one possible approach to construct such distribution. We define the distribution of T as a mixture:

$$\Pr(T=k)=w\Pr(T_1=k)+(1-w)\Pr(T_2=k).$$

Let K be a reasonably large number, but not too large for computational cost consideration. The idea here is to use the K biased estimators $\zeta_1,...,\zeta_K$ to build the unbiased estimators $\hat{l}_i(\theta)$. The first distribution $\Pr(T_1=k)$ is truncated so that its values lie within the interval $[0,K]$. The distribution of T_1 is given by:

$$\Pr(T_1=k)=\begin{cases}\frac{1}{k^{1+\beta}}/\sum_{l=1}^{K}\frac{1}{l^{1+\beta}}, & k=1,...,K\\ 0 & k>K\end{cases}$$

and

$$\Pr(T_1\geq k)=\begin{cases}1-\sum_{h=1}^{k-1}\frac{\frac{1}{h^{1+\beta}}}{\sum_{l=1}^{K}\frac{1}{l^{1+\beta}}}, & k=1,...,K\\ 0 & k>K\end{cases}$$

where β is small, for example 0.01. The distribution of T_2 is negative binomial distribution $NB(1,\rho)$, where ρ is chosen such that the probability of getting large k is very small.

So,

$$\begin{aligned}\Pr(T\geq k) &= w\Pr(T_1\geq k)+(1-w)\Pr(T_2\geq k)\\ &= w\Pr(T_1\geq k)+(1-w)(1-\rho)^k.\end{aligned}$$

The n_k can be computed using the following formula

$$n_k=\lceil\frac{1}{\tau^k\Big(w\Pr(T_1\geq k+1)+(1-w)(1-\rho)^{k+1}\Big)}\rceil$$

for some values of $0<\tau<1$. The first component in the denominator controls the rate of n_k. Without this component, n_k increases exponentially. Table 4 shows the number of importance samples n_k for different k and the probability $\Pr(T\geq k)$ when

$\tau=0.95$, $\rho=0.99$, $w=0.9$, $K=20$, and $\beta=0.01$. Using this setup, T still has a chance of being K, then

$$n_K=\lceil\frac{1}{\tau^K(1-w)(1-\rho)^{K+1}}\rceil$$

can be large. We can define $n_K=n_{K-1}$ and condition (5.3) still holds.

Table 4: n_k when $K=20$, $w=0.9$, $\tau=0.95$, $\rho=0.99$, $\beta=0.01$

k	n_k	$\Pr(T_1\geq k+1)$	$\Pr(T_2\geq k+1)$	$\Pr(T\geq k+1)$
5	4	0.3613	10^{-12}	0.3251
10	11	0.1832	10^{-22}	0.1649
19	216	0.0137	10^{-40}	0.0123
20	2.7895e+43	0	10^{-42}	10^{-43}

5.2 Taylor Correction Approach

Although the exact debiasing approach provides us with an exactly unbiased estimator of the log-likelihood, the estimator might have a high variance and be computationally expensive to compute. A fast alternative is to use the Taylor correction approach. From equation (5.6), $\epsilon_{i,n}\sim\mathcal{N}(0,1)$ as n is large, so $\mathbb{E}\big(\epsilon_{i,n}^3\big)\approx 0$ and thus,

$$\mathbb{E}(\log\hat{p}_n(y_i|\theta)-l_i(\theta))=-\frac{\sigma_i^2(\theta)}{2n}+O\big(n^{-2}\big),$$

where $\sigma_i^2(\theta)=n\mathbb{V}(\hat{p}_n(y_i|\theta))/p(y_i|\theta)^2$. Therefore, $\widetilde{l}_i(\theta):=\log\hat{p}_n(y_i|\theta)+\frac{\widehat{\sigma}_i^2(\theta)}{2n}$ is an approximately unbiased estimator of $l_i(\theta)$ with a bias of order n^{-2}. Although this method only provides us with a nearly unbiased estimator of $l_i(\theta)$, the bias term decreases to zero very fast with the number of samples n and the variance of $\widetilde{l}_i(\theta)$ is in general much smaller than the variance of the exact unbiased estimator $\widehat{l}_i(\theta)$.

5.3 Simulation Study: Random Effects Panel Data Model

The proposed VBILL estimator with data subsampling and the difference estimator is written in Matlab. The examples with moderate data are run on an Intel Core i7 3.6 GHz desktop supported by the Matlab Parallel Toolbox with 4 local processors. The bigger data example is run on a high performance cluster with 12 local processors. The performance of VBILL with data subsampling and the difference estimator is compared to pseudo-marginal MCMC (PMMH) simulation (Andrieu and Roberts (2009)), which still generates samples from the posterior when the likelihood in the Metropolis-Hastings algorithm is replaced by its unbiased estimator. The likelihood in the panel data context is a product of n integrals over random effects. Each integral is estimated unbiasedly using importance sampling (IS), with the number of importance samples chosen such that the variance of unbiased likelihood estimator is approximately 1, for the optimal PMMH as suggested by Pitt et al. (2012). Each MCMC chain consists of 30000 iterates with another 10000 iterates used as burn in iterates.

Panel data are generated from the following logistic model with random effects:

$$p\left(y_{it}|\beta,\alpha_i\right) = Binomial\left(1,p_{it}\right),$$

and

$$Logit(p_{it}) = \beta_0 + \beta_1 x_{it} + \alpha_i \tag{5.9}$$

for $i=1,...,n$ and $t=1,...,5$, $\alpha_i \sim N(0,\tau^2)$. For the moderate simulation study, we generate two datasets of $n=400$ and $n=1000$, and $\beta=(-1.5,1.5)'$, $\tau^2=1.5$, and $x_{it}\sim U(0,1)$.

We use the variational distribution $q_\lambda(\theta)=q(\beta)q(\tau^2)$, where $q(\beta)$ is a $d=2$-variate normal $N(\mu,\Sigma)$ and $q(\tau^2)$ is an inverse gamma distribution $IG(a,b)$. We then run VBILL with data subsampling and difference estimator for both exact debiasing and the Taylor correction approach. For the difference estimator, $w_i(\theta)\equiv\log\widehat{p}\left(y_i|\overline{\theta}\right)$, with $\overline{\theta}$ a simulated maximum likelihood (SML) estimate of θ.

Table 5: Simulation Results for $n=400$

Parameter	True	VBILL (exact)	VBILL (Taylor)	MCMC
β_0	-1.5	−1.3611 (0.1300)	−1.4686 (0.1529)	−1.3980 (0.1713)
β_1	1.5	1.1564 (0.1948)	1.4062 (0.2093)	1.1887 (0.2393)
τ^2	1.5	1.6653 (0.2345)	1.6686 (0.3020)	1.8961 (0.3628)
	CPU time (mins)	16.78	0.0786	184.00

Table 6: Simulation Results for $n=1000$

Parameter	True	VBILL (exact)	VBILL (Taylor)	MCMC
β_0	-1.5	−1.5205 (0.1104)	−1.5514 (0.0939)	−1.4694 (0.1021)
β_1	1.5	1.6619 (0.1565)	1.6722 (0.0.1445)	1.6228 (0.1553)
τ^2	1.5	1.3853 (0.1375)	1.2724 (0.1670)	1.4580 (0.1824)
	CPU time (mins)	22.8204	0.0647	1330

Tables 5 and 6 show the estimates of the posterior mean and posterior variance (shown in brackets) of β_0, β_1, and τ^2 for the three methodologies MCMC, VBILL with exact debiasing, and VBILL with the Taylor series correction. All the estimates are close to their true values. In Table 5 where $n=400$, VBILL seems to underestimate the posterior variances compared to the "gold standard" MCMC estimates, while in Table 6 where $n=1000$, VBILL estimates the variance more accurately. Figure 5.1 and 5.2 plot the VBILL estimates and MCMC estimates of the marginal posteriors $\pi(\beta_0)$, $\pi(\beta_1)$, and $\pi(\tau^2)$. These two figures show that the VBILL marginal posterior estimates using both the bias correction approaches are very close to the MCMC estimates especially when the number of panels is 1000. This confirms that with a large number of panels or observations and a small number of parameters, we know that the posterior $\pi(\theta)$ is approximately normal, so the VBILL variational distribution $q(\beta)=N(\mu,\Sigma)$ should be a very accurate approximation of $\pi(\theta)$ as can be seen from Figure 5.2.

We note that the VBILL with the exact debiasing approach takes a much longer time to run compared to VBILL with Taylor correction, with a not much difference in the resulting marginal posterior estimates. On average at each iteration, VBILL with exact debiasing and with the Taylor series correction use around 8% and 1.5% of the full dataset, respectively. The variance of the exactly unbiased estimator is large in this example so it is computationally more expensive to reduce the variance to $V_{\max}$.

Figure 5.1: Simulation Results for $n = 400$ panels

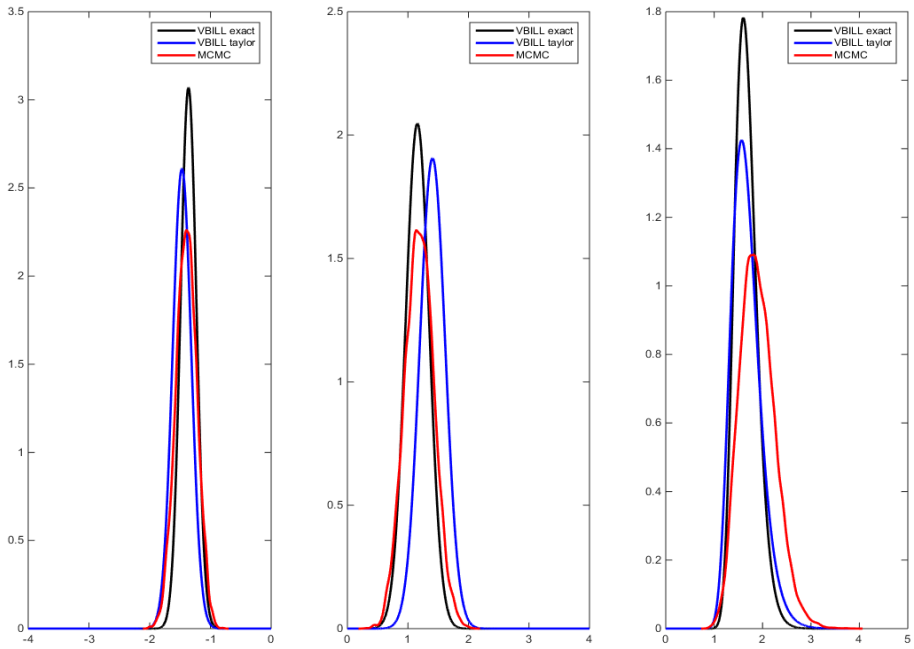
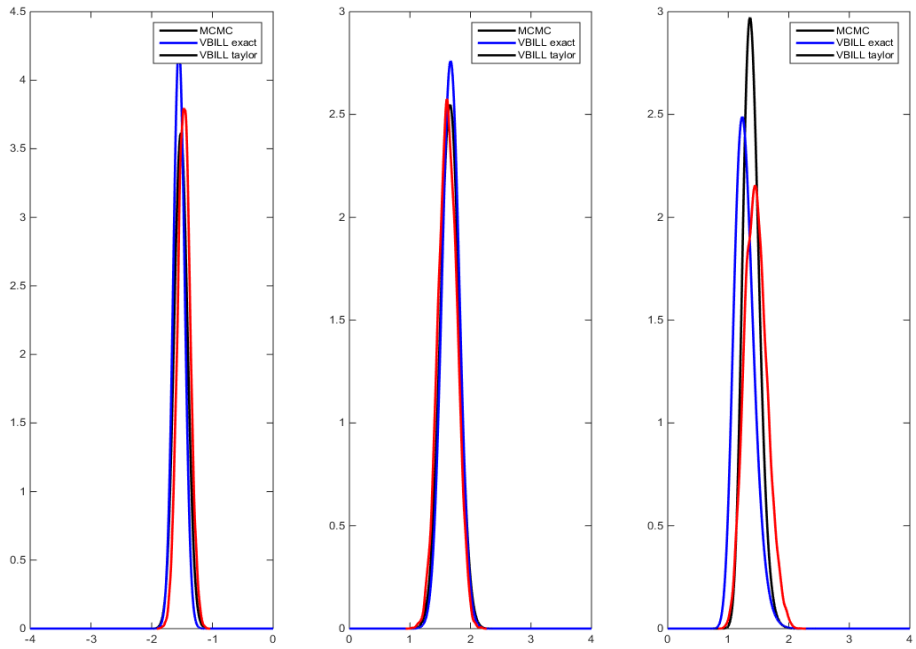


Figure 5.2: Simulation Results for $n = 1000$ panels



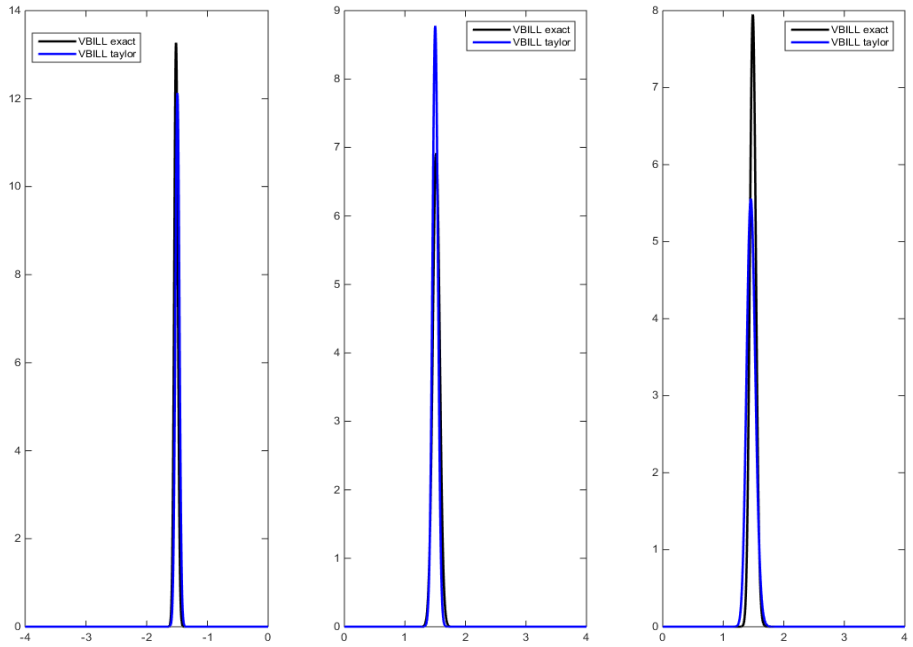
Large Panel Data Example

This section describes a scenario where it is difficult to use the PMMH method. We consider a large data case with the number of panels $n = 10000$. The PMMH method will not work since the variance of unbiased estimator of the likelihood is very large and it requires a huge number of importance samples in order to target the optimal variance of 1. So, if an optimal PMMH procedure is run on our computer to generate 40000 iterations, it would take 124,530 minutes. We run the VBILL based on the exact debiasing and Taylor correction, with the maximum variance V_{max} set to 5000 and 1000, respectively, which require approximately 20% and 5% of the full dataset on average in each iteration of the VBILL procedure. This is because the exact debiasing approach produces an unbiased estimator of the log-likelihood with a larger variance, thus it requires a bigger subsample size to keep the variance below the maximum variance. Both VBILL methods converge after a few iterations. Table 7 summarizes the results. Figure 5.3 plots the variational approximations of the marginal posteriors, which are bell shaped as expected with a very large dataset. The two debiasing correction approaches produce very similar results, however, the CPU time for VBILL with the Taylor correction is much smaller than the VBILL with the exact debiasing approach. This makes VBILL with the Taylor correction suitable for the big panel data models.

Table 7: Simulation results: large dataset application to the logistic model with random effects)

Parameter	True	VBILL (exact)	VBILL (Taylor)
β_0	-1.5	-1.5166 (0.0301)	-1.4940 (0.0329)
β_1	1.5	1.5174 (0.0577)	1.5002 (0.0455)
τ^2	1.5	1.4917 (0.0025)	1.4618 (0.0052)
	CPU time (in mins)	452.43	1.18

Figure 5.3: Simulation results: large dataset application to the logistic model with random effects



6 Conclusions

We have proposed the VBILL algorithm with data subsampling and the difference estimator, which is useful for Bayesian inference in big data and big panel data models. For panel data examples, our proposed algorithms, especially the VBILL algorithm with the Taylor series correction, are much faster than PMMH and produce estimates that are very close to PMMH. Furthermore, the proposed methodology works well when the log-likelihood estimates are highly variable. We also make use of the advanced MapReduce programming technique to develop an approach to analyse massive datasets which cannot fit into a single desktop's memory. Our estimator is mathematically justified and independent of data chunking.

7 Appendix

7.1 Variance Reduction Methods

The performance of Algorithm 1 depends greatly on the variance of the noisy gradient. This section describes control variate methods to reduce this variance.

Control Variate

Denote $\widehat{h}(\theta,\gamma) := \log p(\theta) + \widehat{l}(\theta,\gamma)$ and let $\theta_s \sim q_\lambda(\theta)$ and $\gamma_s \sim g(\gamma|\theta_s)$ for $s = 1,...,S$, be S samples from the variational distribution $q_\lambda(\theta)g(\gamma|\theta)$. For any number c_i, consider

$$\widehat{\nabla_{\lambda_i} KL}(\lambda) = \frac{1}{S}\sum_{s=1}^{S} \nabla_{\lambda_i}(\log q_\lambda(\theta_s))\Big(\log q_\lambda(\theta_s) - \widehat{h}(\theta_s,\gamma_s) - c_i\Big)$$

which is still an unbiased estimator of $\nabla_{\lambda_i} KL(\lambda)$, whose variance can be reduced by an appropriate choice of c_i. Similar ideas are considered in the literature; see Tran et al. (2015), Paisley et al. (2012); Ranganath et al. (2014). The optimal c_i that minimizes the variance of $\widehat{\nabla_{\lambda_i} KL}(\lambda)$ is given by,

$$c_i = \frac{\operatorname{cov}\Big(\nabla_{\lambda_i}(\log q_\lambda(\theta))\Big(\log q_\lambda(\theta) - \widehat{h}(\theta,\gamma)\Big), \nabla_{\lambda_i}(\log q_\lambda(\theta))\Big)}{\mathbb{V}(\nabla_{\lambda_i}(\log q_\lambda(\theta)))}, \tag{7.1}$$

which can be estimated by samples $(\theta_s,\gamma_s) \sim q_\lambda(\theta,\gamma)$. The samples used to estimate c_i must be independent of the samples used to estimate the gradient to ensure the unbiasedness of the gradient estimator. In practice, the c_i can be updated sequentially as follows. At iteration t, we use the c_i computed in the previous iteration $t-1$ to estimate the gradient $\widehat{\nabla_{\lambda_i} KL}\big(\lambda^{(t)}\big)$, which is estimated using new samples from $q_{\lambda^{(t)}}(\theta,\gamma)$. We then update the c_i using this new set of samples. Doing this reduces computational cost since no extra samples are needed to be generated in updating c_i and the unbiasedness of the gradient estimator is achieved.

Exponential Family and Natural Gradient

Suppose that the variational distribution $q_\lambda(\theta)$ belongs to an exponential family of the form,

$$q_\lambda(\theta) = \exp\Big(T(\theta)'\lambda - Z(\lambda)\Big),$$

where $T(\theta)$ is the vector of sufficient statistics and λ is vector of natural parameters. Then, as shown in Tran et al. (2015),

$$\nabla_\lambda KL(\lambda) = I_F(\lambda)\lambda - H(\lambda).$$

Here $H(\lambda) = E_{\theta,\gamma}\Big(\widehat{h}(\theta,\gamma)\nabla_\lambda(\log q_\lambda(\theta))\Big)$ and $I_F(\lambda) = \operatorname{cov}_{q_\lambda}(T(\theta),T(\theta))$ is the Fisher information matrix and can be computed in closed form in most cases. The vector $H(\lambda)$ can be estimated unbiasedly by samples from $q_\lambda(\theta)g(\gamma|\theta)$. The control variate

method can be used to reduce the variation in estimating $H(\lambda)$. Given S samples (θ_s,γ_s), the ith element is estimated unbiasedly by:

$$\widehat{H}_i(\lambda)=\frac{1}{S}\sum_{s=1}^{S}\Big(\widehat{h}(\theta_s,\gamma_s)-c_i\Big)\nabla_{\lambda_i}(\log q_\lambda(\theta)),$$

where

$$c_i=\frac{\mathrm{cov}\Big(\widehat{h}(\theta,\gamma)\nabla_{\lambda_i}(\log q_\lambda(\theta)),\nabla_{\lambda_i}(\log q_\lambda(\theta))\Big)}{\mathbb{V}(\nabla_{\lambda_i}(\log q_\lambda(\theta)))}.$$

Using the natural gradient in minimising the Kullback-Leibler divergence is in general more efficient and reliable than the traditional gradient (Amari, 1998; Tran et al., 2015). If the variational distribution $q_\lambda(\theta)$ has the exponential family form, the natural gradient is given by

$$\nabla_\lambda KL(\lambda)^{natural}=\lambda-I_F(\lambda)^{-1}H(\lambda).$$

Using the natural gradient, and assuming that the variational distribution $q_\lambda(\theta)$ has the exponential family form, the algorithm 1 becomes,

Algorithm 2.

- *Initialize* $\lambda^{(0)}$ *and stop the following iteration if the stopping criterion is met.*
- *For* $t=0,1,...,$ *compute* $\lambda^{(t+1)}=(1-a_t)\lambda^{(t)}-a_tI_F\big(\lambda^{(t)}\big)^{-1}\widehat{H}\big(\lambda^{(t)}\big)$.

7.2 Estimating the variance of the unbiased log-likelihood estimator $\widehat{l_m}(\theta)$

The estimator of the log-likelihood from a data subsample of size m is of the following form,

$$\widehat{l_m}(\theta)=w+\widehat{d}_m,\ \ \widehat{d_m}=\frac{1}{m}\sum_{i=1}^{m}nd_{u_i}$$

with u_i independently and uniformly distributed on the index set $\{1,...,n\}$. So

$$\mathbb{V}\left(\widehat{l_m}(\theta)\right)=\mathbb{V}(\widehat{d_m})=\frac{n^2}{m}\mathbb{V}(d_{u_i})=\frac{n^2}{m}\sigma^2_{pop},$$

where

$$\sigma^2_{pop}=\frac{1}{n}\sum_{i=1}^{n}d_i^2-\left(\frac{1}{n}\sum_{i=1}^{n}d_i\right)^2$$

can be considered as the population variance of the entire population $\{d_1,...,d_n\}$. Given observations $\{d_{u_1},...,d_{u_m}\}$, this population variance can be estimated by the sample variance

$$\widehat{\sigma}^2_{pop}=\frac{1}{m-1}\sum_{j=1}^{m}\left(d_{u_j}-\frac{1}{m}\sum_{i=1}^{m}d_{u_i}\right)^2=\frac{1}{n^2(m-1)}\sum_{j=1}^{m}\left(nd_{u_j}-\widehat{d}_m\right)^2.$$

The variance of $\widehat{l_m}(\theta)$ is estimated by

$$\widehat{\mathbb{V}}\left(\widehat{l_m}(\theta)\right)=\frac{n^2}{m}\widehat{\sigma}^2_{pop}=\frac{1}{m(m-1)}\sum_{j=1}^{m}\left(nd_{u_j}-\widehat{d}_m\right)^2.$$

7.3 Proof of Theorem 1

Proof of Theorem 1. (i) Algorithm 1 is the Robbins-Monro procedure for finding the root λ^* of the equation $\nabla_\lambda \mathrm{KL}(\lambda) = 0$. So (3.1) follows from Theorem 1 of Sacks (1958).

(ii) We denote by $\widetilde{\nabla_\lambda \mathrm{KL}}(\lambda^*)$ the noisy gradient obtained when the log-likelihood is available. Then, noting that $\mathbb{E}_*(\zeta_*(\theta)) = 0$, the constant c in (7.1) is

$$c = \frac{\mathbb{E}_{\theta,\gamma}\{\zeta_*(\theta)^2(\log q_{\lambda^*}(\theta) - \log p(\theta) - \widehat{l}(\theta,\gamma))\}}{\mathbb{E}_*\{\zeta_*(\theta)^2\}} = \frac{\mathbb{E}_*\{\zeta_*(\theta)^2(\log q_{\lambda^*}(\theta) - \log p(\theta) - l(\theta))\}}{\mathbb{E}_*\{\zeta_*(\theta)^2\}} = \widetilde{c}.$$

We note that $\widetilde{c}$ is the control variate constant we would use to compute $\widetilde{\nabla_\lambda \mathrm{KL}}(\lambda^*)$ if the log-likelihood was known. By the law of total variance,

$$\begin{aligned}
\mathbb{V}\big(\widehat{\nabla_\lambda \mathrm{KL}}(\lambda^*)\big) &= \frac{1}{S}\mathbb{V}_{\theta,\gamma}\Big\{\zeta_*(\theta)(\log q_{\lambda^*}(\theta) - \log p(\theta) - \widehat{l}(\theta,\gamma) - c)\Big\} \\
&= \frac{1}{S}\mathbb{E}_*\Big\{\zeta_*^2(\theta)\sigma^2(\theta)\Big\} + \frac{1}{S}\mathbb{V}_*\Big\{\zeta_*(\theta)(\log q_{\lambda^*}(\theta) - \log p(\theta) - l(\theta) - c)\Big\} \\
&= \frac{1}{S}\mathbb{E}_*\Big\{\zeta_*^2(\theta)\sigma^2(\theta)\Big\} + \frac{1}{S}\mathbb{V}_*\Big\{\zeta_*(\theta)(\log q_{\lambda^*}(\theta) - \log p(\theta) - l(\theta) - \widetilde{c})\Big\} \\
&= \frac{1}{S}\mathbb{E}_*\Big\{\zeta_*^2(\theta)\sigma^2(\theta)\Big\} + \mathbb{V}\big(\widetilde{\nabla_\lambda \mathrm{KL}}(\lambda^*)\big).
\end{aligned}$$

Therefore,

$$\sigma^2_{\mathrm{asym}}(\widehat{\lambda}_M) = c_{\lambda^*}\mathbb{V}\big(\widehat{\nabla_\lambda \mathrm{KL}}(\lambda^*)\big) = \sigma^2_{\mathrm{asym}}(\widetilde{\lambda}_M) + \frac{c_{\lambda^*}}{S}\mathbb{E}_*\Big\{\zeta_*^2(\theta)\sigma^2(\theta)\Big\}.$$

□

References

Amari, S. (1998). Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276.

Andrieu, C. and Roberts, G. (2009). The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37:697–725.

Battey, H., Fan, J., Liu, H., Lu, J., and Zhu, Z. (2015). Distributed estimation and inference with statistical guarantees. Technical report. arXiv:1509.05457v1.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. New York: Springer.

Fitzmaurice, G. M., Laird, N. M., and Ware, J. H. (2011). *Applied Longitudinal Analysis*. John Wiley & Sons, Ltd, New Jersey, 2nd edition.

Flury, T. and Shephard, N. (2011). Bayesian inference based only on simulated likelihood: Particle filter analysis of dynamic economic models. *Econometric Theory*, 1:1–24.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. (2013). Stochastic variational inference. *Journal of Machine Learning Research*, 14:1303–1347.

Jacob, P. and Thiery, A. H. (2015). On non-negative unbiased estimators. *Annals of Statistics*, 43(2):769–784.

Kane, M. J., Emerson, J., and Weston, S. (2013). Scalable strategies for computing with massive data. *Journal of Statistical Software*, 55 (14):1–19.

McLeish, D. (2012). A general method for debiasing a Monte Carlo estimator. *Monte Carlo Methods and Applications*, 17:301–315.

Nott, D. J., Tran, M. N., and Leng, C. (2012). Variational approximation for heteroscedastic linear models and matching pursuit algorithm. *Statistics and Computing*, 22(2):497–512.

Ormerod, J. T. and Wand, M. P. (2010). Explaining variational approximations. *American Statistician*, 64:140–153.

Paisley, J., Blei, D., and Jordan, M. (2012). Variational Bayesian inference with stochastic search. In *International Conference on Machine Learning*, Edinburgh, Scotland, UK.

Pitt, M. K., Silva, R. S., Giordani, P., and Kohn, R. (2012). On some properties of Markov chain Monte Carlo simulation methods based on the particle filter. *Journal of Econometrics*, 171(2):134–151.

Quiroz, M., Villani, M., and Kohn, R. (2015). Scalable MCMC for large data problems using data subsampling and the difference estimator. Technical report, Stockholm University. http://arxiv.org/abs/1507.02971v2.

Ranganath, R., Gerrish, S., and Blei, D. M. (2014). Black box variational inference. In *International Conference on Artificial Intelligence and Statistics*, volume 33, Reykjavik, Iceland.

Rhee, C. H. and Glynn, P. W. (2015). Unbiased estimation with square root convergence for sde model. *Operation Research*, 63(5):1026–1043.

Sacks, J. (1958). Asymptotic distribution of stochastic approximation procedures. *The Annals of Mathematical Statistics*, 29(2):373–405.

Sato, M. (2001). Online model selection based on the variational Bayes. *Neural Computation*, 13(7):1649–1681.

Tran, M.-N., Nott, D., and Kohn, R. (2015). Variational Bayes with intractable likelihood. Technical report, University of Sydney. arXiv:1503.08621.

Wang, C., Chen, M. H., Schifano, E., Wu, J., and Yan, J. (2015). A survey of statistical methods and computing for big data. Technical report. arXiv:1502.07989v1.