General Bayesian Marginal Likelihood Estimation Using Iterative Density Estimation

Taylor McKenzie

Abstract

Bayesian statistics provides a very general, well-founded, and intuitive framework for model selection. Any exclusive models that permit a proper posterior distribution can be compared via Bayes' factors, and the probability that any given model from a set of potential models is correct can be calculated. However, it can be hard to estimate Bayes' factors due to difficulties in computing a model's marginal likelihood. Methods have been developed to make this problem computationally feasible for models that can be fit with Gibbs or Metropolis-Hastings samplers (Chib and Jeliazkov, 2001). Unfortunately, many models cannot be estimated with Gibbs sampling, and both Gibbs and Metropolis-Hastings sampling may be much slower to converge and less efficient than newer algorithms such as No U-Turn Sampling (Hoffman and Gelman, 2014). This research develops a general algorithm to estimate marginal likelihood and, by extension, Bayes' factors using iterative kernel density estimation. Using this algorithm with No U-Turn Sampling can produce unbiased, lower variance estimates of marginal likelihood for a broader class of models than those from other methods for similar numbers of sampling iterations.

1 Introduction

2 Literature Review

2.1 Model Selection

Density estimation could theoretically be used to estimate marginal likelihood. However, this method has not been feasible historically due to practical issues with many kernel density estimators, described in greater detail in the following subsection.

2.2 Kernel Density Estimation

2.3 Markov Chain Monte Carlo Samplers

3 Theory and Method

As mentioned previously, models estimated in the Bayesian framework can be compared via their marginal likelihoods. For two models M_j and M_k, the relative goodness-of-fit of M_k over M_j, called the Bayes' factor, is the ratio of the marginal likelihoods of each model, expressed as

$$\frac{m(y|M_k)}{m(y|M_j)}. \tag{1}$$

The marginal likelihood of M_k can be written as

$$m(y|M_k) = \frac{f(y|\theta, M_k)p(\theta|M_k)}{p(\theta|y, M_k)}, \tag{2}$$

where θ are parameters of the model, $f(y|\theta, M_j)$ is the likelihood of the data, $p(\theta|M_j)$ is the value of the prior density, and $p(\theta|y, M_j)$ is the posterior density. As noted by Chib (1995), this identity holds for each θ, and while the values of the likelihood and prior density are typically known (because they are specified to estimate the model), the value of the posterior density is usually unknown, motivating the development of many Markov Chain Monte Carlo (MCMC) techniques to sample from the posterior distribution. In practice, the marginal likelihood must be estimated at a point θ^* via estimation of the posterior density. As noted by Chib (1995), using θ^* from a high-density region of the posterior can reduce the variance of marginal likelihood estimates.

While a number of methods have been developed to estimate the value of the posterior density in certain cases, a simple and general approach is to use kernel density estimation (KDE), which can be used to construct and estimate values of a density function from samples of a random variable. Since all MCMC methods produce samples of $\theta|y, M_k$, this method can be used for any MCMC algorithm. Howevef, there

are two fundamental issues that complicate this approach. First, many standard KDE procedures produce biased estimates of the density function, systematically underestimating values in high-density regions and overestimating values in low-density regions (Silverman, 1986). Fortunately, this is easily remedied via use of more sophisticated KDE methods, such as adaptive KDE.

The other issue complicating the use of KDE to estimate posterior densities is based in the curse of dimensionality. In practice, the posterior density is often a function of several parameters, and KDE becomes less reliable as the number of dimensions and is often completely infeasible for more than five dimensions. To illustrate a solution, first denote the parameter vector as $\theta = (\theta_1, \theta_2, ..., \theta_P)'$, where P is the total number of parameters. Using laws of conditional probability (and now omitting the conditional on model M_k), we can write the marginal likelihood as

$$\begin{aligned} p(\theta|y) &= p(\theta_1, ..., \theta_P|y) && \text{(3a)} \\ &= p(\theta_1|\theta_2, ..., \theta_P, y) \times p(\theta_2, ..., \theta_P|y) && \text{(3b)} \\ &= p(\theta_1|\theta_2, ..., \theta_P, y) \times p(\theta_2|\theta_3, ..., \theta_P, y) && \text{(3c)} \\ &\quad \times p(\theta_3, ..., \theta_P|y) && \text{(3d)} \\ &= ... && \text{(3e)} \\ &= p(\theta_1|\theta_2, ..., \theta_P, y) \times p(\theta_2|\theta_3, ..., \theta_P, y) && \text{(3f)} \\ &\quad \times ... \times p(\theta_P|y). && \text{(3g)} \end{aligned}$$

So, the value of the posterior density can be estimated using the following procedure:

1. Draw samples of $\theta|y$ using an MCMC algorithm.
2. Choose θ^* from a high-density region of $\theta|y$, such as the sample mean or maximum a posteriori.
3. Estimate the log-density of $\theta_P|y$ at θ_P^* using adaptive KDE, denoting that value $\ln \hat{p}(\theta_P^*|y)$.
4. For each i from $P-1, ..., 1$:
   (a) Re-estimate the model, setting $(\theta_{i+1}, ..., \theta_P) = (\theta_{i+1}^*, ..., \theta_P^*)$, to obtain draws of $(\theta_1, ..., \theta_i)|(\theta_{i+1}^*, ..., \theta_P^*), y$.
   (b) Estimate the log-density of $\theta_i|\theta_{i+1}^*, ..., \theta_P^*, y$ at θ_i^* using adaptive KDE, denoting that value $\ln \hat{p}(\theta_i^*|\theta_{i+1}^*, ..., \theta_P^*, y)$.
5. Find the sum of each of the estimated partial log-densities to arrive at an estimate for the overall log-posterior density, denoted $\ln \hat{p}(\theta^*|y)$.

This iterative formulation is by no means novel (a similar formulation was used in Chib (1995) and Chib and Jeliazkov (2001) for Gibbs and Metropolis-Hastings (M-H) samplers, respectively), nor is the method to estimate densities. However, when combined with new MCMC methods, such as No U-Turn Sampling (NUTS), which offer better mixing than traditional samplers, the described methodology can offer lower-variance unbiased estimates of marginal likelihood compared with Gibbs and M-H samplers for the same number of sampling iterations (and even for similar computational run-times in some cases). Further, since MCMC algorithms like NUTS can be practically used to estimate a more general class of models than Gibbs or M-H samplers, the described methodology can be used to compare models that would otherwise be incomparable with traditional MCMC samplers. The following section presents simulation results that compare the described methodology with methods presented by Chib (1995) for models that can be estimated with Gibbs sampling to provide evidence that the proposed method can produce unbiased estimates of marginal likelihood. The research then continues to compare models that are difficult or impossible to fit using Gibbs or M-H sampling.

4 Simulation Results

This section presents simulation results to first illustrate the unbiased, lower-variance estimates of marginal likelihood produced by the proposed methodology compared with the method proposed by Chib (1995) to estimate marginal likelihood using Gibbs sampling. The proposed methodology relies on use of a MCMC algorithm that provides better mixing than traditional samplers in order to reduce variance of marginal likelihood estimates. This research utilizes the No U-Turn Sampler (NUTS) implemented in the Stan Modeling Language (Stan Development Team, 2016). Second, this section presents simulations comparing models that are difficult or impossible to estimate and compare using traditional samplers to illustrate the generality of this methodology. Specific applications include testing between logit and probit specifications and comparing parametric stochastic frontier models with a Bayesian analogue of a non-parametric stochastic frontier model.

Model	# Trials	Gibbs/Chib	Iterative KDE	Mean Test p-value
Multivariate Linear	500	-481.353 (0.154) Iter = 5,000	-481.348 (0.078) Iter = 5,000	0.493
Probit	500	-23.991 (0.04) Iter = 50,000	-23.989 (0.057) Iter = 5,000	0.446

Table 1: Comparison of Gibbs and Iterative KDE

4.1 Multivariate Normal Linear Model, Comparison With Gibbs Sampling

These simulations begin with a standard multivariate linear model with iid normal errors. This model takes the form

$$y = X\beta + \varepsilon \tag{4a}$$
$$\varepsilon \sim iid\ N(0, \sigma^2). \tag{4b}$$

The matrix of independent variable data, X, contained 100 rows (observations) three columns: one constant columns of ones and two independent columns of uniformly random data in the interval $[-10, 10]$. The parameters of the model were arbitrarily chosen as $\beta = (-2, 5, 3)'$ and $\sigma = 25$. The data was generated once and used repeatedly with a Gibbs sampler and NUTS to produce an empirical distribution of marginal likelihoods for this data and model.

Priors over parameters were chosen so that conditional distributions of parameters could be derived, thereby allowing estimation via Gibbs sampling. Specifically, the priors chosen were

$$\beta \sim N(0_3, 100 \times I_3) \tag{5a}$$
$$\sigma^2 \sim \Gamma^{-1}(1, 1). \tag{5b}$$

Using these priors, Gibbs sampling of the posterior distribution $\beta, \sigma^2 | y, X$ can be achieved via alternative sampling of the conditional distributions

$$\beta | \sigma^2, X, y \sim N(\mu_\beta, \Sigma_\beta) \tag{6a}$$
$$\sigma^2 | \beta, X, y \sim \Gamma^{-1}\left(\frac{N}{2}, \frac{e'e}{2} + 1\right), \tag{6b}$$

where

$$\Sigma_\beta = \left(\frac{X'X}{\sigma^2} + \frac{1}{100} \times I_3\right)^{-1} \tag{7a}$$
$$\mu_\beta = \Sigma_\beta \left(\frac{X'y}{\sigma^2}\right) \tag{7b}$$
$$e = y - X\beta. \tag{7c}$$

Estimation of marginal likelihood from this Gibbs sampler followed the three vector block example from Chib (1995). The same model and assumptions were also coded in Stan and marginal likelihood was estimated using the previously described methodology. Each method used 500 warm-up and 5,000 sampling iterations and each was run 500 times to sample the distribution of marginal likelihoods.

The results of this simulation can be found in the first row of Table 1. The Gibbs and Iterative KDE columns show the sample mean of marginal likelihood and standard deviation in parentheses. The sample means from each method are approximately equal, and a mean equality test with the alternative hypothesis that the mean marginal likelihoods are not equal yielded a p-value of 0.493, indicating that the data do not suggest the true means are different at the 10% level of significance. Since the method used in Chib (1995) yields unbiased estimates of marginal likelihood, this finding provides evidence that the proposed method also produces an unbiased estimator of marginal likelihood.

Further, the standard deviation of the Gibbs sampling based method was 0.154 while that of the iterative KDE method was 0.078. A variance equality test was run, with the alternative hypothesis that the variance of the iterative KDE method was less than that of the Gibbs-based method, and yielded a very small p-value (below machine precision), implying the data provides evidence that the proposed method has lower variance than the Gibbs-based method. As mentioned before, this is likely due to the fact that

NUTS provides better mixing and therefore a "better" sample of the posterior distribution, thereby reducing the variance of marginal likelihood estimates. However, the iterative KDE method took around three times as long to run as the Gibbs sampling method on average. Another simulation was run, using 500 warm-up and 2,500 sampling iterations for NUTS and 1,000 warm-up and 15,000 sampling iterations for Gibbs sampling to make computational runtimes approximately equivalent,[1] and similar results were found. Both methods still had equivalent means at the 10% level, and while the variance of the iterative KDE method was higher (0.100) and that of the Gibbs-based method was lower (0.146) than the previously presented results, the iterative KDE method still had significantly lower variance. It is important to note that this final result may not generalizable; as the number of parameters increases (especially parameters that can be evaluated in blocks by the Gibbs sampler, like β), the iterative KDE method will take relatively more time to run compared to the Gibbs-based method because the model must be rerun conditioning on each individual parameter when using iterative KDE.

4.2 Probit Model, Comparison With Gibbs Sampling

Next, the probit model of binary outcomes will be considered. This model has the form

$$z = X\beta \tag{8a}$$

$$\Pr(y = 1|X) = \Phi(z) \tag{8b}$$

$$\Pr(y = 0|X) = 1 - \Phi(z), \tag{8c}$$

where Φ is the cumulative normal distribution. The matrix of independent variable data, X, had 100 observations and two columns: one constant column of ones and one column of uniformly distributed random numbers in the interval $[-1, 1]$. The parameter of the model was arbitrarily chosen to be $\beta = (-2, 5)'$. The data and parameters had to be chosen carefully because convergence of the Gibbs sampler can be difficult in the probit model when the latent variable z takes on extreme values (this presents much less of a problem in the Stan implementation of NUTS). The priors of the model were specified as

$$\beta \sim N(0_2, 100 \times I_2) \tag{9}$$

[1]Numbers of iterations were chosen to make Gibbs sampling runtimes slightly longer than NUTS to give the former method the benefit of the doubt.

The sampler and estimates of the marginal likelihood were obtained via the methodology directly described in Chib (1995). Each algorithm was again run 500 times to produce empirical distributions of marginal likelihoods.

Results of this simulation are shown in the second row of Table 1. Once again, the mean marginal likelihood estimates were not found to be significantly different at the 10% level. The standard deviation of the iterative KDE method was about twice as large as that of the Gibbs-based method due to differences in the numbers of sampling iterations used by each method. The Gibbs sampler generally takes longer to converge than NUTS. In this case, 5,000 warm-up iterations were needed to ensure convergence of the Gibbs sampler, and 50,000 sampling iterations were drawn. On the other hand, NUTS only needed 500 warm-up iterations at most to achieve convergence. Unfortunately, the iterative KDE methodology becomes computationally infeasible for large numbers of sampling iterations due to limitations in adaptive KDE. Thus, only 5,000 sampling iterations were used in the iterative KDE method in this illustration. As a result, the variance of the Gibbs-based marginal likelihood estimation was lower than that of the iterative KDE method.

4.3 Comparison of Probit and Logit Models: An Example from Chib (1995)

In his seminal work, Chib (1995) tested several specifications of a binary probit model using data describing prostatic nodal involvement among 53 prostate cancer patients. To test between those probit specifications, one can use a methodology identical to that in the previous subsection. However, one may also wish to test the choice of link function that transforms the latent variable z into a probability of incidence. Specifically, rather than using a probit model, which uses the cumulative normal as a link function, one could use a logit model, which uses the sigmoid function as a link. The choice of one of these specifications over the other is often at the whim of the researcher, and it can be difficult to test between the specifications both in classical and Bayesian frameworks. If using classical statistics, the two specifications are not nested, so methods like likelihood-ratio tests are not valid. On the other hand, logit models can not be estimated via Gibbs sampling (no conditional distributions exist) and can be difficult in

Specification	Logit	Probit	Chib Est. p-value
C	-38.021 (0.037)	-38.504 (0.038)	0.871
$C + x_1$	-42.303 (0.071)	-43.165 (0.065)	0.123
$C + \log(x_2)$	-36.847 (0.064)	-37.909 (0.062)	0.244
$C + x_3$	-34.323 (0.054)	-35.33 (0.06)	0.247
$C + x_4$	-36.243 (0.065)	-37.229 (0.06)	0.375
$C + x_5$	-38.111 (0.059)	-39.079 (0.058)	0.528
$C + \log(x_2) + x_4$	-34.625 (0.068)	-36.128 (0.076)	0.11
$C + \log(x_2) + x_3 + x_4$	-32.528 (0.077)	-34.559 (0.077)	0.419
$C + \log(x_2) + x_3 + x_4 + x_5$	-33.738 (0.092)	-36.24 (0.079)	0.391

Table 2: Comparison of Logit and Probit Models Using an Example from Chib (1995)

M-H sampling, making it hard or impossible to use methods like those presented in Chib (1995) and Chib and Jeliazkov (2001) to estimate marginal likelihoods. Fortunately, both logit and probit models can be estimated in NUTS, so iterative KDE can be used to compare and test those specifications.

For both the logit and probit models, the priors used were the same as those used by Chib (1995): each β_k was assumed to be independent and normally distributed with mean 0.75 and standard deviation of 5. As in Chib (1995), the models were estimated using 500 warm-up and 5,000 sampling iterations. Each model was run 100 times in both specifications and marginal likelihoods were estimated using iterative KDE. Marginal likelihood estimates for each specification used in Chib (1995) under probit and logit link functions are shown in Table 2. A test of whether mean of the probit simulations was equal to the estimate presented by Chib, with the alternative hypothesis of inequality, was performed and p-values are shown in the final column of Table 2. We can first notice that none of the probit means were found to be significantly different than those presented by Chib at the 10% level. Next, the logit link function performed better than its probit counterpart in each specification. Finally, the best fitting specification was still $C + \log(x_2) + x_3 + x_4$ as in Chib (1995), though the logit form fit better than the probit by a sizable margin.

4.4 Comparison of Probit and Logit Models: Simulation Results

The final simulation offered in this paper investigates the ability of iterative KDE to discriminate between logit and probit models. Data was generated using the following binary models:

$$z = X\beta \tag{10a}$$

$$\Pr(y = 1|X) = L(z) \tag{10b}$$

$$\Pr(y = 0|X) = 1 - L(z), \tag{10c}$$

where L is the cumulative normal in the probit model and sigmoid function in the logit model. The parameter β was arbitrarily chosen to be $(-5, 13)$. The independent variable data X contained 100 observations of 2 columns: one constant column of ones and one column of uniform random data in the interval $[-1, 1]$.[2]

In both the probit and logit models, the prior assumption over the model parameter was

$$\beta \sim N(0_2, 100 \times I_2). \tag{11}$$

[2]These parameters and data were chosen in part because they present convergence difficulties for Gibbs sampling but presented no problems for the Stan implementation of NUTS.

Data	Probit Probability	Logit Probability	Pr(Probit > Logit)	Pr(Logit > Probit)
Probit	0.679	0.321	0.96	0.04
Logit	0.459	0.541	0.33	0.67

Table 3: Monte Carlo Comparison of Logit and Probit Models

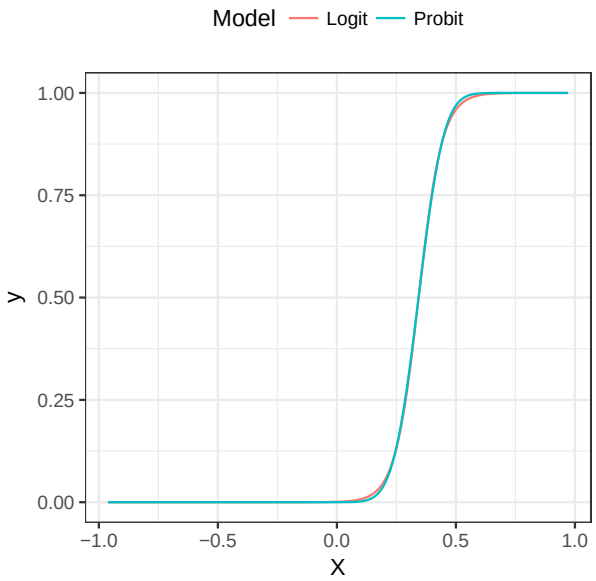

Figure 1: Comparison of Probit and Logit Curves

Both estimation models were used against both data generating processes. Each model was estimated using 500 warm-up iterations and 5,000 sampling iterations, and iterative KDE was used to estimate marginal likelihoods. Data was regenerated and models were fit 100 times to determine average ability of iterative KDE to discriminate between the probit and logit models. Results are presented in Table 3 with rows presenting results from both data generating processes. The probit and logit probability columns detail the average model probabilities for each estimation model, and the final two probability columns show the percentage of times the marginal likelihood for one estimation model exceeded that of the other model.

Each of the estimation models was able to successfully select its own data generation process the majority of the time. The average model selection probability of the probit model under the probit data generation process was 0.679, and the probit marginal likelihood exceeded the logit marginal likelihood 96% of the time in that case. Conversely, the average model selection probability of the logit model under the logit data generation process was 0.541, and the logit was more likely than the probit in 67% of the simulations. While this result may initially seem underwhelming, it is made more impressive the remarkable similarity in probit and logit curves, as shown in Figure 1.[3] Further, as mentioned in the previous subsection, model comparison between logit and probit models has presented a struggle for both classical and Bayesian methods. As shown in this example, iterative KDE opens the possibility of comparing these models in a statistically meaningful way.

5 Parametric and Non-Parametric Stochastic Frontiers

Stochastic frontier models, developed in the seminal paper Aigner et al. (1977), estimate production frontiers under an error specification with one- and two-sided components. Specifically, the model assumes output can be described by

$$y_i = f(x_i) + \varepsilon_i + \delta_i, \tag{12}$$

where y_i is log-output, x_i are inputs, f is some function that transforms inputs to outputs, ε_i is a two-sided error component (e.g., coming from a normal distribution), and δ_i is a negative one-sided error component (e.g., coming from a half-normal or exponential distribution). Estimation is simplified when the density of the sum of one- and two-sided error components is known. As detailed in Aigner et al. (1977), when $\varepsilon \sim N(0, \sigma_\varepsilon^2)$ and $\delta \sim N^-(0, \sigma_\delta^2)$, then the error term $v = \varepsilon + \delta$ has the density function

$$f(v) = \frac{2}{\sigma}\phi\left(\frac{v}{\sigma}\right)\left(1 - \Phi\left(\frac{v\lambda}{\sigma}\right)\right), \tag{13}$$

where $\sigma^2 = \sigma_\varepsilon^2 + \sigma_\delta^2$, $\lambda = \sigma_\delta/\sigma_\varepsilon$, and ϕ and Φ are standard normal density and distribution functions,

[3] These curves were produced using maximum likelihood estimates for each model against simulated data to illustrate similarities of the two curves in practical empirical modeling.

respectively. A density function can also be derived if $-\delta$ followed an exponential distribution.

Even with the exact form of the density function of the error composition, estimation of stochastic frontier models is notoriously difficult in a classical framework.

Parametric specifications of f have traditionally been used, typically in log-linear (each log-transformed input included) or translog (addition of all second-order log terms) form. Non-parametric forms of f have also been recently proposed, such as in Du et al. (2013), and typically involve a two-step procedure: First, the mean of the data is fit using some non-parametric method (such as kernel smoothing), then differences between the fitted curve and the observed data are assumed to be of the above form, from which parameters of the one- and two-sided distributions can be estimated. An integral problem with all classical kernel smoothing methods is the choice of the bandwidth matrix. A

6 Conclusion

References

Aigner, D., C. K. Lovell, and P. Schmidt (1977). Formulation and estimation of stochastic frontier production function models. *Journal of econometrics 6*(1), 21–37.

Chib, S. (1995). Marginal likelihood from the gibbs output. *Journal of the american statistical association 90*(432), 1313–1321.

Chib, S. and I. Jeliazkov (2001). Marginal likelihood from the metropolis–hastings output. *Journal of the American Statistical Association 96*(453), 270–281.

Du, P., C. F. Parmeter, and J. S. Racine (2013). Nonparametric kernel regression with multiple predictors and multiple shape constraints. *Statistica Sinica*, 1347–1371.

Hoffman, M. D. and A. Gelman (2014). The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *Journal of Machine Learning Research 15*(1), 1593–1623.

Silverman, B. (1986). Density estimation for statistical analysis.

Stan Development Team (2016). RStan: the R interface to Stan. R package version 2.14.1.