Transport Monte Carlo

Leo L. Duan*

Abstract: In Bayesian inference, transport map is a promising alternative to the canonical Markov chain Monte Carlo for posterior estimation: it uses optimization to find a deterministic map from an easy-to-sample reference distribution to the posterior. However, the invertible map often does not exist between the two distributions; and it can be challenging to parameterize with sufficient flexibility. Motivated to address these issues and substantially simplify its use, we propose Transport Monte Carlo. Instead of relying on a single deterministic map, we consider a coupling joint distribution modeled by a non-parametric mixture of several maps. Such a coupling is guaranteed to exist between the reference and posterior distributions. To automate the map parameterization and estimation, we use the invertible neural networks to replace the manual design procedure. Once the coupling is estimated, one can rapidly generate a large number of samples that are completely independent. With a carefully chosen reference distribution, the difference between the generated samples and the exact posterior is negligibly small. Both theoretic and empirical results demonstrate its advantages for solving common but challenging sampling problems.

1 Introduction

Bayesian paradigms have been routinely used for uncertainty quantification. Markov chain Monte Carlo, particularly the Gibbs sampling, has been the mainstream tool for the posterior estimation: a primary challenge is that the samples tend to be auto-correlated when updating via the conditional given the current state, leading low effective sample size. To be able to have a high-accuracy estimate, one has to run a long simulation of the Markov chain. For a recent discussion on this issue, see Robert et al. (2018). A large literature of new Markov chain methods has been proposed, aiming to make each Markov state less dependent on the previous state. Methods include Metropolis-adjusted Langevin algorithm (Roberts et al., 1996), Hamiltonian Monte Carlo (Neal et al., 2011), piecewise deterministic (Bierkens et al., 2019), or continuous-time Markov chain Monte Carlo (Fearnhead et al., 2018). At the same time, there is some literature focusing on alternatives to Markov chain methods. For example, among others, Approximate Bayesian Computation (Beaumont et al., 2009) avoids the evaluation of the likelihood and instead relies on a divergence between simulated and observed data in rejection sampling; variational Bayes (Blei et al., 2017) replaces the complicated posterior with the mean-field approximation that is easier to sample. Despite the popularity of those approximations, there are critical issues, such as ignoring the correlation among the parameters, and difficulty to quantify the approximation error for diagnostics. For the discussion and remedy on those issues, see Giordano et al. (2018).

The transport map is an interesting new solution (El Moselhy and Marzouk, 2012). To briefly review, let \( \theta \in \Theta \) be a continuous parameter of interest, \( \Pi_0(\theta) \) the prior probability density, \( Y \) the data and \( L(Y; \theta) \) the likelihood. Slightly abusing notation, denote the random variable from the posterior as

\[
\theta \sim \Pi(\theta \mid Y) = \frac{L(Y; \theta)\Pi_0(\theta)}{m(Y)}
\]

where \( m(Y) = \int L(Y; \theta)\Pi_0(\theta)d\theta \) is normalizing constant. Consider another random variable \( \beta \in \beta \) from another proper measure (with \( \beta \) having the same number of elements as \( \theta \)), referred to as the reference
distribution
\[ \beta \sim \Pi_r(\beta). \]

If there exists an invertible map that pushes forward the reference measure to the posterior \( f : \beta \rightarrow \Theta \).

Using standard variable transformation \( \theta = f(\beta) \), the pullback density from the posterior should equal to the reference one:
\[ \frac{L\{Y; f(\beta)\} \Pi_0(f(\beta))}{m(Y)}|\text{det}\nabla f(\beta)| = \Pi_r(\beta), \tag{1} \]
with \( \nabla f \) the Jacobian matrix. With \( f \) flexibly parameterized and satisfying invertibility, one minimizes the difference between the left (often without the constant \( m(Y) \)) and right sides, using samples \( \beta_i \sim \Pi_r(\beta) \).

When the optimal is reached, \( f(\beta_i) \)'s are the exact posterior samples, and they are completely independent — a major advantage compared to the Markov chain methods.

Despite its potentials, there are two limitations. First, most importantly, there is no guarantee that an invertible map exists between \( \Pi_r(\beta) \) and \( \Pi(\theta \mid Y) \) in general. This is also known as the ill-posed Monge problem [Santambrogio, 2015]. As an example, if \( \Pi_r(\beta) \) is a bivariate Gaussian but \( \Pi(\theta \mid Y) \) has one element fixed to a constant (Dirac measure), there is no map between the two due to the different dimensions. Even without the dimensionality issue, often \( f \) is parameterized with limited flexibility, which can lead to thinner distribution tails of \( f(\beta) \), compared to \( \Pi(\theta \mid Y) \). As shown in our later example, when \( \Pi(\theta \mid Y) \) is multimodal, and the estimated \( f(\beta) \) tends to converge to only one of the modes. As a remedy, existing methods instead use the transport map to optimize Markov transition kernel from one state to the next [Parno and Marzouk, 2018]. Despite some improvement, this arguably weakens the strength of having independence in the samples. We address this issue via a much more general coupling in the form of a mixture of multiple transport maps, borrowing strengths from the non-parametric Bayes for both flexibility and computing tractability.

Second, the typical implementation relies on the manual design of \( f \), as one has to choose a composite of several invertible functions. These procedures are cumbersome for the users. We are not the first to recognize this issue, as some automated solutions have been recently proposed (such as Levy et al. [2018]). Our uniqueness is that we explore the clear alternative to the unscalable optimization needed for every sample of \( (\beta_i, \theta_i) \): instead, first estimate \( f \) on a smaller training set of \( \beta_i \)'s, then generate a large number of samples with \( f \) fixed. We provide a theoretic examination on why the associated error is negligible, producing a motivated choice for the reference distribution.

## 2 Method

### 2.1 Random Transport via Non-parametric Coupling

For a concise exposition, we focus on \( \theta \) and \( \beta \) as continuous random variables. Extensions to other cases will be discussed at the end of the article. Consider a coupling, defined by the joint density \( P \), between \( \theta \) and \( \beta \):

\[ (\theta, \beta) \sim P(\theta, \beta), \]
\[ \int P(\theta, \beta) d\beta = \Pi(\theta \mid Y), \tag{2} \]
\[ \int P(\theta, \beta) d\theta = \Pi_r(\beta). \tag{3} \]

The coupling always exists — one trivial case would be the independent coupling \( P(\theta, \beta) = \Pi_r(\beta) \Pi(\theta \mid Y) \).

In fact, the transport map [1] is another case of deterministic coupling \( P(\theta, \beta) = \Pi(\theta \mid Y) \delta(\beta - f^{-1}(\theta)) \) (with \( \delta \) the Dirac delta), which relies on the strong assumption that [3] can be met for any \( \Pi_r(\beta) \).

To relax this assumption, we instead consider a random transport, by modeling the conditional distribution of \( \beta \) given \( \theta \) as a random output from one of several invertible maps, parameterized by a mixture...
distribution

\[ P(\theta, \beta) = P(\beta)P(\theta \mid \beta), \quad P(\theta) = \Pi(\theta \mid Y), \quad P(\beta \mid \theta) = \sum_{k=1}^{K} w_k(\theta)\delta\{\beta - f_k^{-1}(\theta)\}, \tag{4} \]

with the mixture weight \( w_k(\theta) \geq 0 \), \( \sum_{k=1}^{K} w_k(\theta) = 1 \) for any \( \theta \). We allow \( w \) to potentially change with \( \theta \) for more flexibility (this includes the simpler case with \( w \) invariant to \( \theta \)).

This mixture is inspired by the Bayesian non-parametric method to model an unknown conditional density (Dunson et al., 2007). The randomness in \( P(\beta \mid \theta) \) has two benefits: (i) we relax the stringent equal dimensionality requirement, as \( \Pi(\theta \mid Y) \) can have a lower measure dimension than \( \Pi_r(\beta) \); (ii) we can now use simpler \( f \) with less worry on its flexibility, as it can be compensated with multiple copies of \( f_k \)'s. The following theorem formalizes the high flexibility of random transport as \( K \to \infty \).

**Theorem 1.** Under (4), assuming \( \Pi_r(\beta) \) has an equal or higher dimension than \( \Pi(\theta \mid Y) \), as \( K \to \infty \) there exist \( \{(w_k, f_k)\}_k \) such that (2) and (3) are satisfied almost everywhere.

With this generalization, the random transport still enjoys an equality connecting \( \Pi_r(\beta) \) and \( \Pi(\theta \mid Y) \), providing a path for using optimization to estimate \( f \) and \( w \).

**Theorem 2.** Under (3) and (4),

\[ \Pi_r(\beta) = \sum_{k=1}^{K} w_k\{f_k(\beta)\}\Pi\{f_k(\beta) \mid Y\}|\det\nabla f_k(\beta)|. \tag{5} \]

### 2.2 Transport Monte Carlo

With the above results, we can now develop a Transport Monte Carlo algorithm, which consists of two stages: (i) optimizing \( f \) and \( w \) using a set of training samples \( \beta_1, \ldots, \beta_n \sim \Pi_r \); (ii) generating new samples \( \theta \) using the estimated \( f \) and \( w \). In the first stage, we minimize

\[ \text{Loss} = \frac{1}{n} \sum_{i=1}^{n} \log g(\beta_i), \quad g(\beta_i) = \frac{\Pi_r(\beta_i)}{\sum_{k=1}^{K} w_k\{f_k(\beta_i)\}L\{Y \mid f_k(\beta_i)\}\Pi_0\{f_k(\beta_i)\}|\det\nabla f_k(\beta_i)|}, \tag{6} \]

which is related to the empirical Kullback-Leibler divergence as \( \text{Loss} + \log m(Y) \).

After the optimization converges, in the second stage, we sample new \( \beta \) and \( \theta \) via

\[ \beta \sim \Pi_r, \quad \theta = f_k(\beta) \text{ with probability } v_k, \quad v_k \propto w_k\{f_k(\beta)\}\Pi\{f_k(\beta) \mid Y\}|\det\nabla f_k(\beta)|. \tag{7} \]

We separate these two stages because the optimization is the time-consuming step, using a smaller set of samples leads to much better scalability; while the sampling is embarrassingly parallel and we can rapidly obtain a large number of samples. In the later section, we will show its associated error is very small with a well-chosen \( \Pi_r \) and sufficiently large \( n \).

### 2.3 Automated Computation for Invertible Maps

A major complexity of the transport map methods is in the parameterization and estimation of the invertible maps. To simplify and automate this process, we take advantage of the popularity of neural networks and their computational toolbox.
Following Dinh et al. (2017), we parameterize each $f_k$ as an M-layer invertible neural network. Without loss of generality, assuming $\beta$ is a p-element vector. We set each layer as a p-element to p-element transform:

$$f_k(\beta) = \sigma_k^k \circ \sigma_k^{k-1} \circ \cdots \circ \sigma_k^1(\beta)$$

$$\sigma_m^k(x) = [x_A \ast s_A^k(x_B) + t_A^k(x_B), x_B]'$$ if $m$ is even

$$\sigma_m^r(x) = [x_A, x_B \ast s_B^k(x_A) + t_B^k(x_A)]'$$ if $m$ is odd

where $\ast$ denotes the Hadamard elementwise product; $\{A, B\}$ are the partition of the indices; $s_A, s_B, t_A$ and $t_B$ are nested neural networks with a flexible forms (not necessarily invertiable), with the only requirement that all $s_i(x) > 0$.

Each $\sigma_m^k$ resembles a location-scale transform, except that the magnitudes of scaling and shifting also depend on the input, making the change non-linear. Although the nested $s_i$ and $t_i$ have flexible forms, the enclosing $\sigma_m$ is invertible $\sigma_m^{-1}(x) = [(x_A - t_A(x_B))/s_A(x_B), x_B]'$ for even $m$, $\sigma_m^{-1}(x) = [x_A, (x_B - t_B^M(x_A))/s_B^M(x_A)]'$ for odd $m$. Therefore, the entire $f_k$ is invertible. In addition, $\det \nabla f_k(\beta) = \prod_{a^k} s^k_{a^k}$, with $s^k_{a^k}$ the output elements of $s_A^k$ and $s_B^k$ in the neural network $f_k(\beta)$. See Dinh et al. (2017) for the detailed derivation. For $w(\theta)$, we parameterize it similarly to $f_k$ except using an extra layer of Softmax activation, so that the output is in $\Delta^{K-1}$. We parametrize $s_i$ and $t_i$ with a simple neural network containing one hidden layer, connected by the leaky ReLU activation function: the last layer of $s_i$ is exponentiated so that each element is positive; we use $M = 5$ in all the examples. Compared to a manually designed $f$, the neural network is much easier to use for approximating non-linear transform. In addition, it allows us to exploit popular computing toolbox such as PyTorch (Paszke et al. 2017). The optimization utilizes stochastic gradient descent, by randomly generating a new batch of $\beta$’s in each iteration for gradient calculation. For tractable computation, we approximate the infinite $K$ with a truncated $K = 20$ in most examples, and found empirically no difference compared to using larger $K$.

2.4 Generalization Error and Choosing the Reference Distribution

As described above, the samples of $(\beta, \theta)$ are generated using $f$ and $w$, estimated from a finite training set of $\beta$’s. There can be a small difference between the sampled distribution and the exact posterior, which we will refer to as the generalization error (based on some resemblance to the common training-and-prediction task). Such an error can be quantified theoretically, providing a guide on choosing the reference $\Pi_r$. To distinguish from the training samples, in this section, we will use $\beta^*$ and $\theta^*$ to denote the newly generated samples.

When the equality \ref{eq:kl} holds, each training sample $\beta_i$ has a common minimized value $g(\beta_i) = -\log m(Y)$, a constant. In practice, this also holds approximately, as exhibited by the low sample variance of $g(\beta_i)$’s. Intuitively, if $\beta_i$’s are dense enough to cover most of the high probability region in $\Pi_r(\beta)$, the new sample $\beta^* \sim \Pi_r(\beta)$ will be near a certain training $\beta_i$’s with high probability, hence the associated loss $g(\beta^*)$ should be near the minimized loss as well. This concept can be formalized as a high probability $\epsilon$-net, defined as

$$\beta_{\epsilon, \delta} = \{ (\beta_1, \ldots, \beta_n) : \int_{\mathcal{A}} \Pi_r(\beta) d\beta \geq 1 - \delta, \mathcal{A} = \{ \beta : \inf_{i \in \{1 \ldots n\}} \| \beta - \beta_i \| \leq \epsilon \} \}.$$  

Using this concept, we now formally state the accuracy guarantee.

**Theorem 3.** For given $\epsilon > 0$ and $\delta \geq 0$, suppose the training samples satisfies $(\beta_1, \ldots, \beta_n) \in \beta_{\epsilon, \delta}$, and further if

1. $\Pi(\theta \mid Y)$ and $\Pi_r(\beta)$ are absolutely continuous with respect to Lebesgue measure,
2. each $s$ in \ref{eq:transform} is Lipschitz continuous and bounded,

then

$$\inf_{i \in \{1 \ldots n\}} \| \log \{ g(\beta^*) \} - \log \{ g(\beta_i) \} \| = O(\epsilon),$$

with probability $1 - \delta$.  


For this theorem to be practically useful, we need a high probability $\epsilon$-net that achieves small $\epsilon$ under moderate size $n$. Therefore, an important task is to choose a good $\Pi_r$, with the spacing $\epsilon$ decreases rapidly in $n$, associated with the classical problem of bounding maximal spacing (Janson 1987). The multivariate normal, commonly used in transport-related methods, is in fact sub-optimal for this purpose as $\epsilon \approx O(p/\sqrt{\log n})$ (Deheuvels et al. 1986). Instead, in this article we choose uniform distribution $\Pi_r(\beta) = \text{Uniform}(0, 1)^p$, which has a much faster rate $\epsilon \approx O\left\{ (p \log n)/n \right\}$ (Devroye et al., 1982). For a tighter but more involved bound, see Janson (1987).

For diagnostics, we have a tractable error estimate using the empirical Kullback-Leibler divergence

$$KL\{\Pi(\theta^*)||\Pi(\theta \mid Y)\} = \frac{1}{n^*} \sum_{i=1}^{n^*} \log g(\beta^*_i) + \log m(Y).$$

For intractable $m(Y)$ is, we replace it with an estimator $\hat{\log m} = -(1/n) \sum_i \log g(\beta_i)$.

3 Numeric Illustrations

3.1 Multi-modal Posterior

We first sample from the posterior from the mixture of multivariate Gaussian. Consider $\theta$ from the two-component mixture with unequal component means and unequal covariances

$$\Pi(\theta \mid Y) = 0.5\text{No}(\theta; [1], \begin{bmatrix} 1 & 0.5 \\ 0.5 & 1 \end{bmatrix}) + 0.5\text{No}(\theta; [6], \begin{bmatrix} 1 & -0.9 \\ -0.9 & 1 \end{bmatrix}).$$

We choose this example for two reasons: (i) to demonstrate that the deterministic transport map can fail even in such a simple case, and how multiple maps in Transport Monte Carlo can solve this issue; (ii) we intentionally leave out the normalizing constant $m(Y) = 2\pi$ when implementing the loss function (6), in order to test the accuracy of the estimator $\hat{\log m}$.

As shown in Figure 1(b), the deterministic transport map fails to discover the second component, even though the two modes are not too far apart. This is due to the different covariances in the two components, particularly the different correlation signs, disrupting the log-concavity and causing one transport map to collapse to only one component. On the other hand, as shown in Figure 1(c), Transport Monte Carlo correctly identifies both modes as it exploits multiple transport maps. The estimated $\hat{\log m} = 1.830$ is very close to the true $\log m(Y) \approx 1.837$ (based on $n = 10,000$). To assess the new sample error, we generate another 100,000 samples, and the empirical Kullback-Leibler divergence is only 0.011. As a comparison baseline, the kernel-based Kullback-Leibler estimates (Boltz et al. 2009) between the two exact samples from $\Pi(\theta \mid Y)$ is $0.009 \pm 0.004$. This suggests that the Transport Monte Carlo is highly accurate.
(a) Reference distribution $\beta \sim \text{Uniform}(0,1)^2$, which creates a dense net over $(0,1)^2$.

(b) Single transport map $\theta = f(\beta)$ collapses to only one mode of the mixture of Gaussian.

(c) Transport Monte Carlo $\theta = f_k(\beta)$ with probability $v_k$ correctly identifies the two modes.

Figure 1: Comparing the performance of transporting from a uniform distribution (panel a) to the two-component mixture of correlated Gaussian distributions. Transport map, with a single deterministic invertible map, fails to discover the second mode (panel b), while Transport Monte Carlo correctly estimates the posterior (panel c).

3.2 High-Dimensional Parameter

To illustrate the performance with the high dimensional parameter, we consider a sparse linear regression with the regularized horseshoe prior [Piironen and Vehtari 2017]. For data index $j = 1, \ldots, N$ and covariate index $k = 1, \ldots, p$,

$$y_j \sim \text{No}(x_j^T b, \sigma_0^2),$$

$$b_k \sim \text{No}(0, \lambda_k^2 \tau^2), \quad \lambda_k^2 = \frac{c^2 \tilde{\lambda}_k}{c^2 + \tau^2 \tilde{\lambda}_k^2}, \quad \lambda_k \sim C^+(0,1),$$

$$c^2 \sim \text{Ga}^{-1}(v/2, v s^2 / 2).$$

where $x_j \in \mathbb{R}^p$ is the covariate; $C^+$ denotes the half-Cauchy and $\text{Ga}^{-1}$ the inverse-gamma. Compared to the original horseshoe [Carvalho et al. 2010], the large signals approximately follow a normal prior $\text{No}(0,c^2)$, instead of Cauchy, hence has a finite prior mean. As the result of this modification, Gibbs sampling is no longer applicable, and the Hamiltonian Monte Carlo is suggested instead.

When simulating the data, we use $p = 2,000$ and $N = 100$; $\sigma^2 = 0.01$; $x_{j,k} \sim \text{No}(0,1)$; $b_k \sim \text{No}(5,1)$ for $k = 1 \ldots 5$, and $b_k = 0$ for $k = 6 \ldots 2000$. For both $\sigma^2$ and $\tau^2$, we use the informative prior $\text{Exp}(100)$ to induce low noise and shrinkage global scale; for $c^2$, we set $v = 5$, $s = 10$, as suggested by [Piironen and Vehtari 2017]. The results are compared with Hamiltonian Monte Carlo, using the No-U-TURN sampler provided by the PyMC3 package. As most of the parameters are close to the degenerate zero, Hamiltonian Monte Carlo can only use small leap-frog step, resulting in extremely slow mixing [2]. We run it for 100,000 iterations and thin the chain at every 100th step, and use them as the posterior sample. This takes approximately 8 hours. In contrast, the Transport Monte Carlo only takes a few minutes in optimization with $n = 10,000$ and can generate new samples almost instantaneously. Due to the independence, no thinning is needed. The results are almost identical to the one obtained in costly Hamiltonian Monte Carlo.
4 Discussion

We show that Transport Monte Carlo can serve as an appealing alternative to conventional Markov chain Monte Carlo methods, and enjoys theoretic justification via a coupling. On the other hand, several drawbacks remain and merit future work. Although we show the error is very small, unlike Markov chain Monte Carlo, there is no asymptotic exactness guarantee as the number of samples goes to infinity. A possible correction amenable to the high dimensional parameter is using the obtained sample as initial values for parallel Markov chains (Hoffman et al. arXiv preprint-1903.03704). Another extension is to consider \( \theta \) in a discrete or constrained space. Our random transport can be combined with the transport that changes one discrete variable to another (Tran et al., 2019); nevertheless, a careful measure-theoretic study is still an interesting open problem. Lastly, for conciseness, we leave out the cases when one has degenerate mass mixed with the continuous random variable, for example, in the spike-and-slab prior (Ishwaran and Rao, 2005). To accommodate this, the reference \( \Pi_r(\beta) \) needs to be adaptive as well and it is worth exploring as a case study.

References

Beaumont, M. A., J.-M. Cornuet, J.-M. Marin, and C. P. Robert (2009). Adaptive Approximate Bayesian Computation. *Biometrika* 96(4), 983–990.

Bierkens, J., P. Fearnhead, G. Roberts, et al. (2019). The Zig-Zag Process and Super-efficient Sampling for Bayesian Analysis of Big Data. *The Annals of Statistics* 47(3), 1288–1320.

Blei, D. M., A. Kucukelbir, and J. D. McAuliffe (2017). Variational Inference: a Review for Statisticians. *Journal of the American Statistical Association* 112(518), 859–877.

Boltz, S., E. Debreuve, and M. Barlaud (2009). High-dimensional Statistical Measure for Region-of-interest Tracking. *IEEE Transactions on Image Processing* 18(6), 1266–1283.

Carvalho, C. M., N. G. Polson, and J. G. Scott (2010). The Horseshoe Estimator for Sparse Signals. *Biometrika* 97(2), 465–480.

Deheuvels, P. et al. (1986). On the Influence of the Extremes of an IID Sequence on the Maximal Spacings. *The Annals of Probability* 14(1), 194–208.
Appendix

Proof of theorem 1

Proof. It is trivial to see that (2) holds. For (3), we show its existence via one (among many) construction. For any measurable \( A \), denote the conditional probability

\[
\int_A \frac{P(\theta, \beta)}{\Pi(\theta \mid Y)} d\beta = \mu(A) \in (0, 1].
\]
As any measurable function is the pointwise limit of simple function, for any \( h : \Theta \to \mathbb{R} \) measurable, there exists limit

\[
h(\beta) = \lim_{n \to \infty} \sum_{i=1}^{n} g_{[n]}i 1(\beta \in E_{[n]}i),
\]

where \( E_{[n]}1, \ldots, E_{[n]}n \) the disjoint partition of \( \Theta \) and \( g_{[n]}i \) a constant depending on \( E_{[n]}i \).

Integrate over \( \beta \),

\[
\int h(\beta)d\beta = \lim_{n \to \infty} \sum_{i=1}^{n} g_{[n]}i \mu(E_{[n]}i).
\]

As \( \mathcal{A} \neq \emptyset \), we can choose \( E_{[n]}i \) small enough such that \( \mathcal{A} = \bigcup_{i \in I} E_{[n]}i \) with \( I \) some set of \( i \). Therefore \( \mathcal{A} \cap E_{[n]}i = \emptyset \) unless \( E_{[n]}i \subseteq \mathcal{A} \). Take \( h(\beta) \) to be \( 1(\beta \in \mathcal{A}) \), the left hand side is \( \mu(\mathcal{A}) \), choose the right hand side \( g_{[n]}i = 1(E_{[n]}i \subseteq \mathcal{A}) \).

Given \( \beta \), the integral using (1) has

\[
\int_{\mathcal{A}} P(\beta | \theta)d\beta = \sum_{k=1}^{\infty} w_k(\theta) 1\{f_k^{-1}(\theta) \in \mathcal{A}\}.
\]

For each \( k \), let \( w_k(\theta) = \mu(E_{[n]}i) \) and \( f_k \) satisfy \( f_k^{-1}(\Theta) \subseteq E_{[n]}i \), so that \( 1\{f_k^{-1}(\theta) \in \mathcal{A}\} = g_{[n]}i \), this yields the result.

**Proof of theorem 2**

**Proof.**

\[
\Pi_r(\beta) = \int \sum_{k=1}^{K} w_k(\theta) \Pi(\theta | Y) \delta\{\beta - f_k^{-1}(\theta)\}d\theta
\]

\[
= \sum_{k=1}^{K} \int w_k(\theta) \Pi(\theta | Y) \delta\{\theta - f_k(\beta)\} | \det \nabla f_k(\beta) | d\theta
\]

\[
= \sum_{k=1}^{K} w_k \{ f_k(\beta) \} \Pi \{ f_k(\beta) | Y \} | \det \nabla f_k(\beta) |,
\]

where the Jacobian emerges due to \( \delta\{g(x)\} = \delta(x - x_0) | \det \nabla g(x_0) |^{-1} = \delta(x - x_0) | \det \nabla g^{-1}(x_0) | \) with \( g(x_0) = 0 \) for invertible \( g \).

**Proof of theorem 3**

**Proof.** The task is equivalent to showing \( \log \{g(\beta)\} \) has a bounded derivative almost everywhere. Rewriting

\[
\log \{g(\beta)\} = \log \sum_{k=1}^{K} \exp h_k(\beta),
\]

\[
h_k(\beta) = -\log K + \log \Pi_r(\beta) - \log w_k \{ f_k(\beta) \} - \log L \{ Y | f_k(\beta) \} \Pi_0 \{ f_k(\beta) \} - \log | \det \nabla f_k(\beta) |
\]

and taking derivative with respect to the \( m \)th sub-coordinate of \( \beta \), denoted by \( \beta_{[m]} \), its magnitude satisfies

\[
\left| \frac{\partial \log \{g(\beta)\}}{\partial \beta_{[m]}} \right| = \left| \sum_{k=1}^{K} \frac{\exp h_k(\beta) \partial h_k(\beta) / \partial \beta_{[m]}}{\sum_{k=1}^{K} \exp h_k(\beta)} \right|
\]

\[
\leq \sum_{k=1}^{K} \frac{\exp h_k(\beta) \left| \frac{\partial h_k(\beta)}{\partial \beta_{[m]}} \right|}{\sum_{k=1}^{K} \exp h_k(\beta)}
\]

\[
\leq \max_{k \in \{1 \ldots K\}} \left| \frac{\partial h_k(\beta)}{\partial \beta_{[m]}} \right|.
\]
Examining the derivative yields

\[
\left| \frac{\partial h_k(\beta)}{\partial \beta_{[m]}} \right| \leq \left| \frac{\partial \log \Pi_r(\beta)}{\partial \beta_{[m]}} \right| + \left| \frac{\partial \log w_k(\theta) L \{ Y | \theta \} \Pi_0(\theta)}{\partial \theta} \bigg|_{\theta = f_k(\beta)} \right| \left| \frac{\partial f_k(\beta)}{\partial \beta_{[m]}} \right| + \sum_{\text{all } s_k} \left| \frac{\partial \log |s_k|}{\partial \beta_{[m]}} \right|.
\]

By absolute continuity, the first two absolute values are finite almost everywhere. For the third term, \( |\partial f_k(\beta)/\partial \beta_{[m]}| = \prod_{\text{all } s_k} |s_k| < \infty \). For the last term, each \( s_k \) is a finite composition of Lipschitz functions, therefore it is Lipschitz with respect to \( \beta \); as \( s_k > 0 \), each derivative is equal to \( \partial |s_k|/\partial \beta_{[m]}/|s_k| < \infty \).

Denote the index that achieves the minimum distance as \( i_0 = \arg \inf_{i \in \{1, \ldots, n\}} \| \beta^* - \beta_i \| \), then

\[
\inf_{i \in \{1, \ldots, n\}} \| \log \{g(\beta^*)\} - \log \{g(\beta_i)\} \| \leq \| \log \{g(\beta^*)\} - \log \{g(\beta_{i_0})\} \| = O(\| \beta^* - \beta_{i_0} \|).
\]

Combining the above yields the result. \( \square \)