Journal of Machine Learning Research 21 (2020) 1-54 Submitted 12/19; Published 5/20

Target–Aware Bayesian Inference: How to Beat Optimal Conventional Estimators

Tom Rainforth* RAINFORTH@STATS.OX.AC.UK
Department of Statistics
University of Oxford
29 St Giles', Oxford, OX1 3LB, United Kingdom

Adam Goliński* ADAMG@ROBOTS.OX.AC.UK
Department of Statistics and Department of Engineering Science
University of Oxford
29 St Giles', Oxford, OX1 3LB, United Kingdom

Frank Wood FWOOD@CS.UBC.CA
Department of Computer Science
University of British Columbia
2366 Main Mall 201, Vancouver, BC V6T 1Z4, Canada

Sheheryar Zaidi SHEHERYAR.ZAIDI@SOME.OX.AC.UK
Department of Statistics
University of Oxford
29 St Giles', Oxford, OX1 3LB, United Kingdom

* Equal contribution

Editor: Kilian Weinberger

Abstract

Standard approaches for Bayesian inference focus solely on approximating the posterior distribution. Typically, this approximation is, in turn, used to calculate expectations for one or more target functions—a computational pipeline that is inefficient when the target function(s) are known upfront. We address this inefficiency by introducing a framework for *target-aware Bayesian inference* (TABI) that estimates these expectations directly. While conventional Monte Carlo estimators have a fundamental limit on the error they can achieve for a given sample size, our TABI framework is able to breach this limit; it can theoretically produce arbitrarily accurate estimators using only three samples, while we show empirically that it can also breach this limit in practice. We utilize our TABI framework by combining it with adaptive importance sampling approaches and show both theoretically and empirically that the resulting estimators are capable of converging faster than the standard $\mathcal{O}(1/N)$ Monte Carlo rate, potentially producing rates as fast as $\mathcal{O}(1/N^2)$. We further combine our TABI framework with amortized inference methods, to produce a method for amortizing the cost of calculating expectations. Finally, we show how TABI can be used to convert any marginal likelihood estimator into a target aware inference scheme and demonstrate the substantial benefits this can yield.

Keywords: Bayesian inference, Monte Carlo methods, importance sampling, adaptive sampling, amortized inference

1. Introduction

At its core, Bayesian modeling is rooted in the calculation of expectations: the eventual aim of modeling is typically to make a decision or to construct predictions for unseen data, both of which take the form of an expectation under the posterior (Robert, 2007). This aim can thus be sum-

marized in the form of one or more expectations $\mathbb{E}_{p(x|y)}\big[f(x)\big]$, where $f(x)$ is a target function and $p(x|y)$ is the posterior distribution on x for some data y, which we typically only know up to a normalizing constant $p(y)$. More generally, expectations with respect to distributions with unknown normalization constants are ubiquitous throughout the sciences (Robert and Casella, 2013).

Sometimes $f(x)$ is not known up front. Here it is typically convenient to first approximate $p(x|y)$, e.g. in the form of Monte Carlo (MC) samples, and then later use this approximation to calculate estimates, rather than addressing the final expectation directly. However, it is often the case in practice that a particular target function, or class of target functions, is known a priori. For example, in decision-based settings $f(x)$ takes the form of a loss function, while calculating any (parametric) posterior predictive density involves taking the expectation over the parameters of a known predictive distribution.

Though often overlooked, in such *target-aware* settings the aforementioned pipeline of first approximating $p(x|y)$ and then using this as a basis for calculating $\mathbb{E}_{p(x|y)}\big[f(x)\big]$ is suboptimal as it ignores relevant information in $f(x)$ (Torrie and Valleau, 1977a; Hesterberg, 1988; Wolpert, 1991; Oh and Berger, 1992; Evans and Swartz, 1995; Meng and Wong, 1996; Chen and Shao, 1997; Gelman and Meng, 1998; Lacoste-Julien et al., 2011; Owen, 2013; Rainforth et al., 2018b). In this paper, we look to address this inefficiency.

To do this, the key question we must answer is: how can we effectively incorporate information about $f(x)$ into our inference process? This transpires to be a somewhat more challenging problem than it might at first seem. For example, if we try to incorporate this information by running a Markov chain MC (MCMC) sampler that targets some distribution $q(x)$ encapsulating both $p(x|y)$ and $f(x)$ (e.g. Torrie and Valleau, 1977a), we find that we cannot use the resulting samples to construct a single direct MC estimate for $\mathbb{E}_{p(x|y)}\big[f(x)\big]$, due to the presence of other unknown terms (such as the evidence $p(y)$).

One approach that can be used is importance sampling (Hesterberg, 1988; Gelman and Meng, 1998; Owen, 2013). Specifically, we can set up a proposal $q(x)$ that incorporates information about $f(x)$ and then use this to produce a set of weighted samples whose locations are influenced by both $p(x|y)$ and $f(x)$. By *self-normalizing* these weights, we can then construct a self-normalized importance sampling (SNIS) estimate for $\mathbb{E}_{p(x|y)}\big[f(x)\big]$ that exploits information from $f(x)$.

However, this approach has a fundamental limitation: there is a theoretical lower bound on the error SNIS estimators can achieve for a given problem and sample size (see Eq 4). Though this bound is significantly better than what can be achieved by any single MCMC sampler or any approach that does not use information from $f(x)$, it can still represent a prohibitively large error if our sample budget is restricted. Moreover, it is typically insurmountably difficult to construct an estimator that achieves performance anywhere near to this bound, particularly if $p(x|y)$ and $p(x|y)f(x)$ are badly mismatched.

In this work, we show that this limitation can be overcome by avoiding self-normalization and instead deconstructing the target expectation into three separate parts, setting up separate estimators for each, and then recombining them to form an estimate for the overall problem. We refer to this framework as TABI, which stands for *target-aware Bayesian inference*. Critically, the breakdown TABI applies leads to component expectations which can each be individually estimated arbitrarily well using a tailored importance sampling estimator, even if this estimator is constructed with only a single sample. This, in turn, means that *TABI estimators are theoretically capable of estimating any expectation arbitrarily accurately using only three samples*. In other words, while using the optimal proposal for SNIS and MCMC schemes leads to estimators with finite errors for a given sample size, TABI estimators constructed with optimal proposals produce exact estimators regardless of the number of samples used.

To utilize our TABI framework, we show that it can be combined with adaptive importance sampling (AIS) methods (Oh and Berger, 1992; Cappé et al., 2004; Cornuet et al., 2012; Martino et al., 2017; Bugallo et al., 2017) to produce effective target-aware adaptive inference algorithms. Specifically, given an existing AIS approach, we show how to make it target-aware by applying it

to three different sampling problems, each related to a corresponding component expectation in the TABI framework. We refer to the resulting family of approaches as *target-aware adaptive importance sampling* (TAAIS) methods. We demonstrate theoretically that, given sufficiently expressive proposals families, TAAIS methods are capable of achieving faster mean squared error (MSE) convergence rates than the $\mathcal{O}(1/N)$ rate of conventional SNIS and MCMC methods (where N corresponds to the number of samples). We further confirm this empirically, achieving an MSE rate of $\mathcal{O}(\log(N)/N^2)$ when using a moment-matching-based TAAIS method on a problem where the proposal families include the target distributions. These gains stem from the fact that TAAIS is able to exploit the favorable convergence properties of AIS methods in settings where self-normalization is not required (Portier and Delyon, 2018).

We further extend our TABI framework to *amortized* inference settings (Stuhlmüller et al., 2013; Gershman and Goodman, 2014; Kingma and Welling, 2014; Ritchie et al., 2016; Paige and Wood, 2016; Le et al., 2017, 2018; Webb et al., 2018), wherein one looks to amortize the cost of inference across different possible data sets by first learning an artifact that assists with the inference process at runtime for a given data set. Existing inference amortization approaches do not operate in a target-aware fashion, such that even if the inference network learns proposals that perfectly match the true posterior for every possible data set, the resulting estimator is often still far from optimal. To address this, we introduce AMCI, a framework for performing *Amortized Monte Carlo Integration*.[1] Though still based on learning amortized proposals distributions, AMCI varies from standard approaches by learning three distinct amortized proposals, each targeting one of the component expectations in the TABI framework. Again, this breakdown allows for arbitrary performance improvements compared to conventional methods. To account for cases in which multiple possible target functions may be of interest, we show how AMCI can also be used to amortize over function parameters, rather than just over data sets. We further show that AMCI is able to empirically produce test-time errors lower than those of the respective theoretically optimal SNIS estimator and thus, by proxy, the best possible conventional amortized inference scheme.

We finish by exploring how our TABI framework might be exploited in settings where the base estimators are not constructed using conventional importance samplers. Namely, we show how TABI can be used with any method for approximating the marginal likelihood of a given unnormalized target density. We then exploit this to show how nested sampling (Skilling, 2004) and annealed importance sampling (Neal, 2001) can be converted into target-aware inference methods, confirming empirically the potential advantages this can bring.

To summarize, the rest of the paper is organized as follows. In Section 2, we formalize our problem setting and provide key background on importance sampling and self-normalization. In Section 3, we introduce our core TABI framework in an importance sampling context, highlight its key theoretical properties, and confirm that these are realizable in practice. We further provide insights for when the TABI framework will be relatively more and less useful, and discuss related work. In Section 4, we build on the TABI framework by combining it with adaptive sampling schemes to produce our TAAIS approach, providing both theoretical and empirical evidence of its utility. In Section 5, we consider the amortized inference setting, introducing our AMCI approach and empirically confirming the benefits it can yield. Finally, in Section 6, we show how the TABI framework can be applied to any marginal likelihood estimator to produce a target–aware inference scheme, demonstrating the advantages of doing this through the specific examples of nested sampling and annealed importance sampling.

1. Note that this paper extends the earlier conference publication

   Adam Goliński, Frank Wood, and Tom Rainforth. Amortized Monte Carlo Integration. In Proceedings of the 36th International Conference on Machine Learning, volume 97, pages 2309-2318, 2019.

   in which we focused on using the TABI framework in the amortized inference setting. Here we extend this to consider the TABI framework more generally, introduce the TAAIS framework given in Section 4.3, and use the TABI framework with estimators not based on direct importance sampling as per Section 6.

2. Background

Through most of this paper, we will consider using our TABI framework in an importance sampling context, before returning to this assumption in Section 6 to show how it can be applied more generally. As such, after introducing our problem setting, we now cover the essential basics of importance sampling, along with the concept of self–normalization and its limitations.

2.1. Problem Setting

In this paper, we are concerned with estimating one or more expectations of the form $\mu := \mathbb{E}_{\pi(x)}[f(x)]$, where $\pi(x)$ is a reference distribution and $f(x)$ is target function that we can evaluate pointwise. The reference distribution is assumed to be known up to a normalization constant, i.e. $\pi(x) = \gamma(x)/Z$, where $\gamma(x)$ can be evaluated pointwise, but Z is unknown. Some aspects of the paper will still be relevant in situations where Z is known, but here one should directly use its known value, rather than estimating it.

A particularly common class of problems in our problem setting originate from Bayesian modeling. Here one specifies a prior $p(x)$ over latent variables x along with a likelihood model $p(y|x)$ for data y, and then looks to make use of the posterior

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

where $p(y) = \mathbb{E}_{p(x)}[p(y|x)]$ is an unknown normalizing constant typically called the marginal likelihood or model evidence. In particular, one often wishes to calculate expectations with respect to this posterior, $\mathbb{E}_{p(x|y)}[f(x)]$, for which we thus have $\pi(x) = p(x|y)$, $\gamma(x) = p(x, y)$, and $Z = p(y)$ in our more general notation.

Due to its decision theoretic origins (Robert, 2007), Bayesian modeling is intricately tied to the calculation of such expectations. This is perhaps most easily seen by considering two of the most prevalent uses of Bayesian modeling: making decisions and predictions. In a Bayesian decision theory setting, then, for loss function $L(\cdot, \cdot)$, the optimal decision rule $d^*(y)$ is the one that minimizes the posterior expected loss $\mathbb{E}_{p(x|y)}[L(d(y), x)]$. Posterior predictive distributions, on the other hand, take the form $p(y'|y) = \mathbb{E}_{p(x|y)}[p(y'|x, y)]$ for some $p(y'|x, y)$. We thus see that both cases require the calculation of an expectation with respect to posterior. More generally, calculating expectations with respect to the posterior is one of the most fundamental uses of posteriors, such that calculating such expectations is an important and wide–reaching problem.

2.2. Importance Sampling

Importance Sampling (IS), in its most basic form, is a method for approximating an expectation $\mathbb{E}_{\pi(x)}\left[f(x)\right]$ when it is either not possible to sample from $\pi(x)$ directly, or when the simple MC estimate, $\hat{\mu}_{\text{MC}} := \frac{1}{N}\sum_{n=1}^{N} f(x_n)$ where $x_n \sim \pi(x)$, has problematically high variance (Kahn and Marshall, 1953; Hesterberg, 1988; Wolpert, 1991). Given a proposal $q(x)$—that we can sample from, evaluate the density of, and which has heavier tails than $\pi(x)$ (see, e.g., Owen, 2013)—it forms the following estimate (assuming for now that we can evaluate the density $\pi(x)$ directly)

$$\mu := \mathbb{E}_{\pi(x)}\left[f(x)\right] = \mathbb{E}_{q(x)}\left[\frac{\pi(x)}{q(x)}f(x)\right] \approx \hat{\mu}_{\text{IS}} := \frac{1}{N}\sum\nolimits_{n=1}^{N} f(x_n)w_n, \tag{1}$$

where $x_n \sim q(x)$ and $w_n := \pi(x_n)/q(x_n)$ is known as the importance weight of sample x_n. These importance weights act like correction factors to account for the fact that we sampled from $q(x)$ rather than our target. We note the important feature that this importance sampling estimate is unbiased.

For a general unknown target, the optimal proposal, i.e. the proposal which results in the estimator having the lowest possible variance, is generally taken to be the target distribution

$q^*_{\text{IS}}(x) = \pi(x)$ (see, e.g., Rainforth, 2017, Section 5.3.2.2). However, this no longer holds if we have some information about $f(x)$. Here the optimal proposal can be shown to be (Owen, 2013)

$$q^*_{\text{IS}}(x) \propto \pi(x)|f(x)|.$$

Interestingly, in the case where $f(x) \geq 0\ \forall x$ (or $f(x) \leq 0\ \forall x$), this leads to an exact estimator, i.e. $\hat{\mu}_{\text{IS}} = \mu$, for any number of samples N, even $N = 1$. To see this, notice that the normalizing constant for $q^*_{\text{IS}}(x)$ is $\int \pi(x)f(x)\,\mathrm{d}x = \mu$ and hence $q^*_{\text{IS}}(x) = \pi(x)f(x)/\mu$. Therefore, we have

$$\hat{\mu} = \frac{1}{N}\sum_{n=1}^{N} \frac{f(x_n)\pi(x_n)}{q^*_{\text{IS}}(x_n)} = \frac{1}{N}\sum_{n=1}^{N} \frac{f(x_n)\pi(x_n)\mu}{f(x_n)\pi(x_n)} = \mu$$

regardless of the values of N and $x_1, \ldots, x_N$.

Another point of note is that importance sampling also allows one to convert any integration problem into that of calculating an expectation. Namely,

$$\int_{x\in\mathcal{X}} f(x)dx = \mathbb{E}_{q(x)}\left[\frac{f(x)}{q(x)}\right]$$

for any $q(x)$ where $q(x) \neq 0 \forall \{x \in \mathcal{X} | f(x) \neq 0\}$. As such, all the techniques introduced in the paper can also be straightforwardly applied to the estimation of integrals (by taking $\pi(x) = 1\,\forall\{x \in \mathcal{X}|f(x) \neq 0\}$ and ignoring the fact that this is not a proper distribution), noting though that there is no need to separately estimate a normalization constant in such cases.

2.3. Self–Normalized Importance Sampling

In practice, we typically do not have access to the normalized form of $\pi(x)$ as explained earlier, such that we cannot directly evaluate the importance weights in (1). For example, in Bayesian settings we can only evaluate these weights up to our unknown marginal likelihood $p(y)$. To aid with clarity, we will now assume this Bayesian setting for most of the rest of the paper, but emphasize that the approaches we introduce apply more generally to expectations taken with respect to distributions whose normalization constants are unknown.

To get around this unknown normalization constant problem, we can *self-normalize* our weights as follows

$$\mathbb{E}_{p(x|y)}[f(x)] \approx \hat{\mu}_{\text{SNIS}} := \sum_{n=1}^{N} \frac{w_n}{\sum_{m=1}^{N} w_m} f(x_n) = \sum_{n=1}^{N} \bar{w}_n f(x_n) \tag{2}$$

where $x_n \sim q(x)$, $w_n := p(x_n, y)/q(x_n)$, and $\bar{w}_n = w_n / \sum_{m=1}^{N} w_m$. This approach is known as self-normalized importance sampling (SNIS).

Conveniently, we can also construct the SNIS estimate lazily (i.e. storing weighted samples) by calculating the empirical measure

$$p(x|y) \approx \sum_{n=1}^{N} \bar{w}_n \delta_{x_n}(x)$$

and then using this to construct the estimate in (2) when $f(x)$ becomes available. As such, we can also think of SNIS as a method for Bayesian inference as, informally speaking, the empirical measure produced can be thought of as an approximation of the posterior.

It is important to note that, unlike the standard importance sampling estimate in (1), the SNIS estimate is biased for finite sample sizes (see, e.g., Rainforth, 2017, Section 5.3.2.1). Both are consistent as $N \to \infty$, provided that $q(x)$ has heavier tails than $p(x|y)$ to ensure finite variance.

For SNIS, the optimal proposal transpires to be different to that for standard importance sampling. Specifically, we have (Hesterberg, 1988)

$$q^*_{\text{SNIS}}(x) \propto p(x,y)|f(x) - \mu|. \tag{3}$$

Unlike in the standard importance sampling case, here one can no longer achieve a zero variance estimator for finite N and nonconstant $f(x)$, even when using this optimal proposal. Instead, the achievable mean squared error (MSE) can be approximately lower bounded as follows.

First, we note that the MSE is always larger than or equal to the variance

$$\begin{aligned}\mathbb{E}[(\hat{\mu}_{\text{SNIS}} - \mu)^2] &= \text{Var}[\hat{\mu}_{\text{SNIS}}] + \left(\mathbb{E}[\hat{\mu}_{\text{SNIS}}] - \mu\right)^2 \\ &\geq \text{Var}[\hat{\mu}_{\text{SNIS}}].\end{aligned}$$

Next, we apply the delta method to this as per (Owen, 2013, Eq. 9.8) to yield

$$\approx \frac{1}{N}\,\mathbb{E}_{q(x)}\left[\left(\frac{p(x|y)}{q(x)}\left(f(x) - \mu\right)\right)^2\right].$$

Finally, we apply Jensen's inequality to get the final form of the bound

$$\begin{aligned}&\geq \frac{1}{N}\left(\mathbb{E}_{q(x)}\left[\frac{p(x|y)}{q(x)}|f(x) - \mu|\right]\right)^2 \\ &= \frac{1}{N}\left(\mathbb{E}_{p(x|y)}[|f(x) - \mu|]\right)^2.\end{aligned} \tag{4}$$

Here the first inequality and the approximation from using the delta method both become exact in the limit of large N, while the bound from using Jensen's inequality becomes exact if and only if $q(x) \propto p(x,y)|f(x) - \mu|$ (presuming a non-constant function), such that this derivation also serves to demonstrate (3). We thus see that this creates a fundamental limit on the performance of SNIS, even when information about $f(x)$ is incorporated.

This bound is problem–dependent: the larger the expected distance of $f(x)$ from its mean, the larger the bound; the bound collapses in the case of a constant function. It is interesting to note though, that in the case $f(x) \geq 0\,\forall x$ (or $f(x) \leq 0\,\forall x$), this bound is itself upper bounded by (see Appendix A.1 for a derivation)

$$\frac{1}{N}\left(\mathbb{E}_{p(x|y)}[|f(x) - \mu|]\right)^2 \leq \frac{4\mu^2}{N}, \tag{5}$$

where the bound is tight when $f(x)$ is a Dirac delta function. As such, though there is a limit on how well any SNIS sampler can do, there is also a limit to how poor the *optimal* SNIS sampler will be if the function is non-negative (or non-positive).

However, this point is generally redundant in practice: given that $q^*_{\text{IS}}(x)$ and $q^*_{\text{SNIS}}(x)$ make use of the true expectation μ, we will clearly not have access to them for real problems and cannot usually achieve performance that is anywhere near to the theoretical bounds they produce. Nonetheless, they provide a guide for the desirable properties of a proposal and can, at least in principle, be used as the basis for constructing adaptive or amortized IS methods, as we discuss later.

3. Target–Aware Bayesian Inference

The SNIS estimator we introduced in the last section has two key shortfalls:

1. It has a fundamental limit on the level of accuracy it can achieve for a given problem and sample size as per (4);

2. The optimal proposal $q^*_{\text{SNIS}}(x)$ cannot be evaluated, even in an unnormalized form, due to the μ term. This means it is difficult to construct or learn proposals that are close to it.

In this section, we will give insights into why these problems occur and introduce our *target-aware Bayesian inference* (TABI) framework to address them.

3.1. The Problem with Self–Normalization

The key insight into the limitation of SNIS is that it implicitly uses the same proposal to estimate two expectations, and this proposal, in general, cannot be simultaneously tailored to both. To see this, it is useful to note that the SNIS estimator can be derived as follows

$$\mu := \mathbb{E}_{p(x|y)}[f(x)] = \frac{\mathbb{E}_{p(x)}[p(y|x)f(x)]}{\mathbb{E}_{p(x)}[p(y|x)]} \tag{6}$$

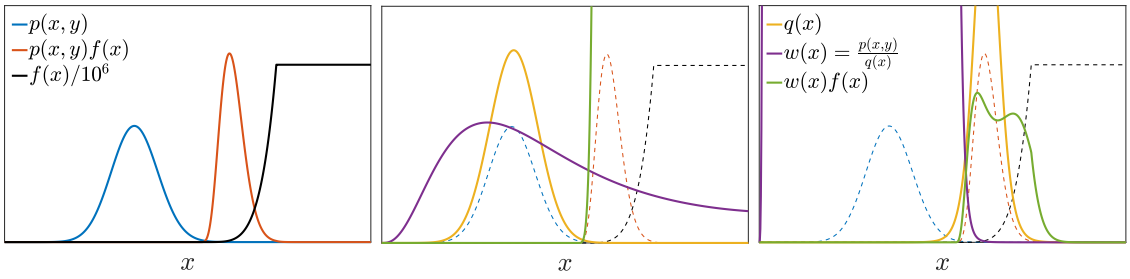
$$= \frac{\mathbb{E}_{q(x)}\left[\frac{p(x,y)}{q(x)}f(x)\right]}{\mathbb{E}_{q(x)}\left[\frac{p(x,y)}{q(x)}\right]} \approx \frac{\frac{1}{N}\sum_{n=1}^{N} w_n f(x_n)}{\frac{1}{N}\sum_{n=1}^{N} w_n} = \hat{\mu}_{\text{SNIS}} \tag{7}$$

where $x_n \sim q(x)$ and $w_n := p(x_n, y)/q(x_n)$ as before. SNIS is thus using the same proposal (and set of samples) to estimate both $\mathbb{E}_{p(x)}[p(y|x)f(x)]$ and $\mathbb{E}_{p(x)}[p(y|x)]$. If $p(x, y)$ and $p(x, y)|f(x)|$ are not well–matched (i.e. they do not result in similar distributions when normalized), then it is difficult to choose a $q(x)$ that is simultaneously effective for estimating both these expectations, while no such proposal can ever be perfect for both unless $f(x)$ is constant.

We can also view this argument from the point of view of the variances of the numerator and denominator components of the estimator. A pictorial demonstration of this is shown in Figure 1. Here we see that if we construct a proposal tailored to match the posterior [middle plot], this produces low variance weights and thus an accurate estimate for $\mathbb{E}_{p(x)}[p(y|x)]$. However, in doing so we may induce high variances on our $w_n f(x_n)$ terms, such that $\mathbb{E}_{p(x)}[p(y|x)f(x)]$ is estimated poorly. If we instead choose a $q(x)$ that is a good approximation to $p(x, y)|f(x)|$ [right plot], this produces low variance $w_n f(x_n)$ terms and thus a good estimate for $\mathbb{E}_{p(x)}[p(y|x)f(x)]$, but now the variance of the weights themselves explodes, yielding poor estimates for $\mathbb{E}_{p(x)}[p(y|x)]$. Thus in both cases we end up with an overall poor estimate.

Though these might at first seem like pathologically poor choices for our proposal, they actually represent the main two approaches used in practice, particularly when using methods for constructing proposals automatically such as the adaptive and amortized approaches we will consider in later sections. In particular, it is common to only attempt to match the posterior when choosing a proposal, even when $f(x)$ is known.

In practice, we could, of course, do better by constructing a proposal that puts mass in both regions of high posterior density and also regions where $p(x, y)|f(x)|$ is large. However, actually constructing such a proposal is generally difficult and usually simply avoided altogether. The reason for this is rooted in the dependency of the form of $q^*_{\text{SNIS}}(x)$ on the true value of μ, as per the second shortfall above. More generally, μ typically heavily influences the relative scaling of $p(x, y)$ and $p(x, y)|f(x)|$, such that constructing a single proposal that combines the needs of both generally requires some implicit knowledge of μ. That is not to say constructing or learning such a proposal is impossible, indeed there are various heuristics one can envisage, but as we now show, the problem can actually be circumvented entirely by avoiding self-normalization altogether.



(a) Joint and target function (b) $q(x)$ targeting $p(x,y)$ (c) $q(x)$ targeting $p(x,y)|f(x)|$

Figure 1: Demonstration of the difficulty of choosing an effective proposal for SNIS. Note that while terms have, in general, been scaled for visualization purposes, the scalings of $w(x)$ and $w(x)f(x)$ are matched within the same plot. [Left] A simple example model $p(x,y)$ (blue line), function $f(x)$ (black line), and the resulting $p(x,y)f(x)$ (red line, all three are also shown as dashed lines on the other plots for reference). [Middle] Using a proposal (yellow line) that targets the posterior leads to stable weights $w(x)$ (purple line, plotted as a function of sampled value) but unstable function-scaled weights $w(x)f(x)$ (green line, note this extends far beyond the y-axis limit of the plot). [Right] Choosing a proposal to instead target the function-scaled posterior stabilizes the function-scaled weights, but also leads to the weights themselves becoming unstable.

3.2. The TABI Framework

The high-level idea behind our TABI framework is to use separate estimators for the numerator and denominator in (6). Doing this allows us to construct separate proposals that are tailored to each expectation, rather than relying on a single proposal to estimate both, as is implicitly the case for SNIS. Namely, if we define $E_1 = \mathbb{E}_{p(x)}[p(y|x)f(x)]$ and $E_2 = \mathbb{E}_{p(x)}[p(y|x)]$, we can separately estimate each as follows

$$\mu = \frac{E_1}{E_2} = \frac{\mathbb{E}_{p(x)}[p(y|x)f(x)]}{\mathbb{E}_{p(x)}[p(y|x)]} = \frac{\mathbb{E}_{q_1(x)}\left[\frac{p(x,y)}{q_1(x)}f(x)\right]}{\mathbb{E}_{q_2(x)}\left[\frac{p(x,y)}{q_2(x)}\right]} \approx \frac{\frac{1}{N}\sum_{n=1}^{N}\frac{p(x'_n,y)}{q_1(x'_n)}f(x_n)}{\frac{1}{M}\sum_{m=1}^{M}\frac{p(x_m,y)}{q_2(x_m)}} =: \frac{\hat{E}_1}{\hat{E}_2} \tag{8}$$

where $x'_n \sim q_1(x)$ and $x_m \sim q_2(x)$, and $q_1(x)$ and $q_2(x)$ are separate proposals tailored to respectively approximate $p(x,y)|f(x)|$ (e.g. Figure 1c) and $p(x,y)$ (e.g. Figure 1b). By contrast, we can think of SNIS as choosing $q_1(x)$ and $q_2(x)$ to be the same distribution (along with fixing $M = N$ and sharing samples between the two estimators).

Breaking this restriction will allow the aforementioned theoretical limitations of SNIS to be overcome. Consider, for example, the case where $f(x) \geq 0\,\forall x$. If $q_1(x) \propto p(x,y)f(x)$ and $q_2(x) \propto p(x,y)$ then both $\hat{E}_1$ and $\hat{E}_2$ will form exact estimators (as per Section 2.2), even if $N = M = 1$. Consequently, we achieve an exact estimator for μ, allowing for arbitrarily large improvements over any SNIS estimator. Moreover, this approach will also make it far easier to construct effective mechanisms for learning good proposals in practice. Namely, even though the theoretically optimal proposals will not typically be achievable, we will still often be able to construct a highly effectively pair of proposals by using these optimal proposals as objectives, for example using the adaptive TAAIS scheme introduced in Section 4.3. Critical to achieving this is the fact that these optimal proposals can be evaluated up to a normalizing constant without knowing μ, unlike $q^*_{\text{SNIS}}(x)$.

We can actually refine this idea of using separate estimators to still produce an exact overall estimator in the case where $f(x) \geq 0 \, \forall x$ does not hold. Specifically, if we let[2]

$$f^+(x) = \max(f(x), 0) \tag{9a}$$
$$f^-(x) = -\min(f(x), 0) \tag{9b}$$

denote truncations of the target function into its positive and negative components, as per the concept of positivisation (Owen and Zhou 2000, Owen 2013, Section 9.13), then we can break down the overall expectation as

$$\mu := \frac{E_1^+ - E_1^-}{E_2} \quad \text{where} \tag{10a}$$
$$E_1^+ := \mathbb{E}_{p(x)}[p(y|x) f^+(x)], \tag{10b}$$
$$E_1^- := \mathbb{E}_{p(x)}[p(y|x) f^-(x)], \tag{10c}$$
$$E_2 := \mathbb{E}_{p(x)}[p(y|x)]. \tag{10d}$$

Analogously to (8), we can introduce a separate proposal for each of these component expectations and use this to construct a separate standard (i.e. not self-normalized) importance sampling estimator for each, before recombining these to form an estimate of the overall expectation

$$\hat{\mu} := \frac{\hat{E}_1^+ - \hat{E}_1^-}{\hat{E}_2} \quad \text{where} \tag{11a}$$
$$\hat{E}_1^+ := \frac{1}{N} \sum_{n=1}^{N} \frac{f^+(x_n^+) p(x_n^+, y)}{q_1^+(x_n^+)}, \quad x_n^+ \sim q_1^+(x), \tag{11b}$$
$$\hat{E}_1^- := \frac{1}{K} \sum_{k=1}^{K} \frac{f^-(x_k^-) p(x_k^-, y)}{q_1^-(x_k^-)}, \quad x_k^- \sim q_1^-(x), \tag{11c}$$
$$\hat{E}_2 := \frac{1}{M} \sum_{m=1}^{M} \frac{p(x_m, y)}{q_2(x_m)}, \quad x_m \sim q_2(x), \tag{11d}$$

which forms our (importance-sampling-based) TABI estimator.

Because each component estimator (i.e. $\hat{E}_1^+$, $\hat{E}_1^-$, and $\hat{E}_2$) is a standard importance sampling estimator whose target function is strictly non-negative, each can be arbitrarily accurate for a finite sample budget (as per Section 2.2), even when $M = N = K = 1$. This means that for any expectation, the TABI estimator using the corresponding set of optimal proposals will produce exact estimates with only three samples! This critical feature of the estimator is formalized in the following theoretical result.

Theorem 1 *If $E_1^+, E_1^-, E_2 < \infty$ and we use the corresponding set of optimal proposals $q_1^+(x) \propto f^+(x)p(x,y)$, $q_1^-(x) \propto f^-(x)p(x,y)$, and $q_2(x) \propto p(x,y)$, then the importance sampling TABI estimator defined in* (11) *satisfies*

$$\mathbb{E}[\hat{\mu}] = \mu, \;\; \mathrm{Var}[\hat{\mu}] = 0$$

for any $N \geq 1$, $K \geq 1$, and $M \geq 1$, such that it forms an exact estimator.

2. Practically, it may sometimes be beneficial to truncate the proposal about another point, c, by instead using $f^+(x) = \max(f(x) - c, 0)$ and $f^-(x) = -\min(f(x) - c, 0)$, then adding c onto our final estimate. One can even use a $c(x)$ that varies with x provided that $\mathbb{E}_{p(x|y)}[c(x)]$ is known, as per (Owen and Zhou, 2000, Section 7.1).

Proof The result follows straightforwardly from considering each estimator in isolation and noting that the normalization constants for our chosen q_1^+, q_1^-, q_2 are E_1^+, E_1^-, E_2,respectively. Therefore, starting with $\hat{E}_2$, we have

$$\hat{E}_2 = \frac{1}{M}\sum_{m=1}^{M}\frac{p(x_m, y)}{q_2(x_m)} = \frac{1}{M}\sum_{m=1}^{M}\frac{p(x_m, y)}{p(x_m, y)/E_2} = E_2$$

for all possible values of $x_1, \ldots, x_M$. Similarly, for $\hat{E}_1^+$

$$\hat{E}_1^+ = \frac{1}{N}\sum_{n=1}^{N}\frac{p(x_n^+, y)f^+(x_n^+)}{q_1(x_n^+)} = \frac{1}{N}\sum_{n=1}^{N}\frac{p(x_n^+, y)f^+(x_n^+)}{p(x_n^+, y)f^+(x_n^+)/E_1^+} = E_1^+$$

for all possible values of $x_1^+, \ldots, x_N^+$. Analogously, we have $\hat{E}_1^- = E_1^-$ for all possible values of $x_1^-, \ldots, x_K^-$. The result now follows from the fact that each sub-estimator is exact. ■

The significance of this result is that there is no limitation on how efficient TABI estimators can be: the better we make the proposal, the lower the error, with perfect proposals giving zero error regardless of the number of samples used. By contrast, the error for SNIS will saturate: there is a lower bound that we can never breach, no matter how good our proposal is.

Moreover, the achievable performances of other conventional estimation approaches are generally also limited by the SNIS bound. As such, these powerful theoretical properties of TABI are highly unusual; we are not aware of any previous general–purpose estimation strategy in the literature that shares them.[3]

For example, one might consider instead trying to use samples from an MCMC chain. However, by noting that MCMC is simply a mechanism for drawing samples from a target distribution, rather than direct estimator for an expectation, we see that the optimal MCMC sampler coincides with the optimal importance sampler for the same target: both produce equally weighted i.i.d. samples according to this target. Consequently, MCMC does not, in general, provide a mechanism for breaching the SNIS performance limit. In fact, what is achievable using MCMC will generally be much worse than SNIS: because MCMC samplers do not provide natural normalization constant estimates, we do not have the same flexibility to incorporate the target function information by aiming to produce samples from a different distribution than the posterior.

In summary, despite its simplicity, the TABI framework allows us to overcome a relatively fundamental theoretical bound in the achievable performance of conventional MC estimators in Bayesian inference settings. The key to achieving this is in breaking down the target expectation into three sub-expectations that can each be estimated arbitrarily well by existing methods. Even when we are unable to construct sufficiently good proposals to produce a TABI estimator that outperforms the theoretically optimal SNIS estimator, this breakdown will still often prove extremely useful in practice. Namely, the ability to tailor each proposal to their respective expectation will typically lead to a TABI estimator that is much more effective than the equivalent practically achievable SNIS estimator.

3.3. An Empirical Demonstration

As we will show in later sections, the demonstrated theoretical properties of the TABI estimator will be particularly beneficial in situations where the proposals are automatically learned as we are then often able to achieve highly effective proposals that successfully utilize the benefits of the TABI estimator. These settings will thus be the focus of our empirical evaluations. Nonetheless, there will still be many scenarios where the TABI framework is useful with manually constructed proposals.

3. Note, however, that other estimators with this property are possible, such as the one we introduce in Appendix C.

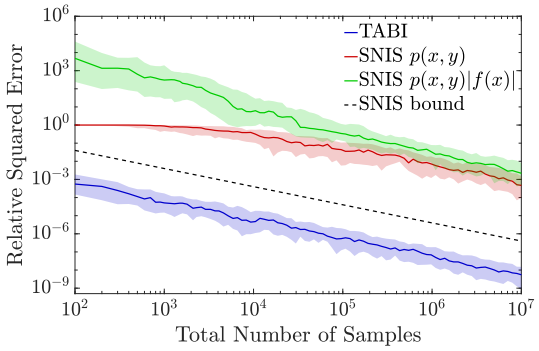


Figure 2: Convergence of simple example shown in Figure 1 using SNIS with the proposal targeting $p(x, y)$ from Figure 1b (red), SNIS with the proposal targeting $p(x, y)|f(x)|$ from Figure 1c (green), the TABI estimator that uses these proposals for E_2 and E_1^+ respectively with $M = N$ (blue, note $E_1^- = 0$ and so does not need estimating), and the theoretically optimal SNIS sampler (dotted black), i.e. the bound in (4). Solid lines represent the median across 100 runs, shading the 25% and 75% quantiles. Note that the x–axis corresponds to $M + N = 2M$ for TABI and M for SNIS, such that the cost of generating the latter is strictly larger for a given x–axis value (see Section 3.4).

As a simple demonstration of this, we consider comparing TABI with SNIS for the example introduced in Figure 1. For reproducibility, the precise definition of this model is given by

$$\begin{aligned} p(x) &= \textsc{Gamma}(x\,;\, \text{shape} = 5,\, \text{scale} = 4), \\ p(y|x) &= \mathcal{N}(y\,;\, x, 1), \\ f(x) &= \min\Big(15000,\, \max\big(0,\, 50\,(x-8)^5\big)\Big), \end{aligned}$$

where we take $y = 5$. The two proposals used for Figure 1b and Figure 1c are respectively given by a Gaussian with mean 5.4 and standard deviation 0.98, and a student-t distribution with 10 degrees of freedom centered at 9.3 and scaled by a factor of 0.5. The results of using TABI and different SNIS estimators are shown in Figure 2, where our metric of performance is the relative squared error

$$\hat{\delta} := \frac{(\hat{\mu} - \mu)^2}{\mu^2}. \tag{12}$$

We see that not only does TABI substantially outperform both of these SNIS estimators, it produces errors around two orders of magnitude less than the optimal SNIS sampler, highlighting the significant gains achievable from using the TABI framework.

3.4. Computational Cost

Constructing the TABI estimator requires $N+K$ evaluations of $f(x)$, $M+N+K$ evaluations of the joint density $p(x, y)$, $M+N+K$ proposal draws, and $M+N+K$ evaluations of a proposal density. By comparison, an N–sample SNIS estimator must make N evaluations of each. In the absence of additional information, a natural budget allocation for TABI is to use the same number of samples for each component estimator (i.e. $M = N = K$). This thus leads to an estimator whose cost is between double (when the cost of evaluating $f(x)$ is dominant) and triple (when either sampling or evaluation of the densities is the dominant cost) that of the equivalent N–sample SNIS estimator. However, there are scenarios where using different sample sizes will be beneficial and allow us to

reduce the relative cost of TABI. Namely, if one of the estimators is more accurate than the others, we can reduce the relative number of samples it uses; we typically want the *effective* sample size of each estimator to be roughly the same (see Section 3.6). In particular, if $f(x) \geq 0 \, \forall x$ or $f(x) \leq 0 \, \forall x$, we can set $K = 0$ and $N = 0$ respectively, such that here the cost of the TABI estimator now varies between being of comparable cost to, and double the cost of, the respective SNIS estimator.

In practice, these comparisons can be potentially misleading: the TABI approach will rarely be computationally wasteful relative to SNIS. The is because the samples generated by TABI are more tailored to the particular term they are estimating, such that their expected effective sample sizes for a given budget will typically be larger. As shown in Appendix B, it is actually possible to recycle all of the generated samples in TABI such that each sub–estimator uses all $M + N + K$ samples, producing an overall estimator that is similar to a $(M + N + K)$–sample SNIS estimator and with effectively equivalent cost. It is thus perhaps more appropriate to compare TABI to this estimator, relative to which the TABI estimator is never slower and provides the potential for speed–ups (of up to a factor of two depending on context) by omitting to recycle wasteful (or potentially even harmful) samples. Indeed, all empirical comparisons in the paper will benchmark against this SNIS estimator, such that they represent a conservative comparison for TABI in terms of real–time performance.

3.5. Discussions and Theoretical Insights

We now consider the question of when we expect TABI to work particularly well compared to SNIS, and the scenarios where it may be less beneficial, or potentially even harmful. Specifically, we gain insights into the relative performance of the two approaches in different settings using an asymptotic analysis in the limit of a large number of samples. We will assume $f(x) \geq 0 \, \forall x$ for simplicity,[4] such that $E_1^- = 0$ and does not need estimating, $f^+(x) = f(x)$, and $E_1^+ = E_1$; we thus drop the $+$ notation. As previously noted, we can think of SNIS as a special case of the TABI estimator where we set $q_1(x) = q_2(x)$, $N = M$, and share samples between the estimators.[5]

We start by defining the random variables

$$\xi_1 := \frac{\hat{E}_1 - E_1}{\sigma_1}, \quad \xi_2 := \frac{\hat{E}_2 - E_2}{\sigma_2}$$

which can be used to characterize the errors of the estimators, where

$$\sigma_1^2 := \mathrm{Var}\left[\hat{E}_1\right] = \frac{1}{N}\mathrm{Var}_{q_1(x)}\left[\frac{f(x)p(x,y)}{q_1(x)}\right], \quad \sigma_2^2 := \mathrm{Var}\left[\hat{E}_2\right] = \frac{1}{M}\mathrm{Var}_{q_2(x)}\left[\frac{p(x,y)}{q_2(x)}\right]$$

are the variances of $\hat{E}_1$ and $\hat{E}_2$ respectively. Now by noting that

$$\hat{\mu} = \frac{E_1 + \sigma_1\xi_1}{E_2 + \sigma_2\xi_2}, \tag{13}$$

and that the central limit theorem shows us that in limits $N \to \infty$ and $M \to \infty$ we respectively get $\xi_1 \sim \mathcal{N}(0,1)$ and $\xi_2 \sim \mathcal{N}(0,1)$, we can derive the following result for the mean squared error (MSE).

Theorem 2 *The asymptotic MSE of the TABI estimator $\hat{\mu}$ is given by*

$$\mathbb{E}\left[(\hat{\mu} - \mu)^2\right] = \frac{\sigma_2^2\mu^2}{E_2^2}\left((\kappa - \mathrm{Corr}[\xi_1, \xi_2])^2 + 1 - \mathrm{Corr}[\xi_1, \xi_2]^2\right) + \mathcal{O}(\epsilon) \tag{14}$$

where $\kappa := \sigma_1/(\mu\sigma_2)$ is a measure of the relative accuracy of the two estimators and $\mathcal{O}(\epsilon)$ represents asymptotically dominated terms that disappear in the limit $M, N \to \infty$.

4. The results trivially generalize to general $f(x)$ with suitable adjustment of the definition of σ_1.
5. Though we omit it from our considerations here, there are some interesting edge cases where $\hat{\mu}_{\mathrm{SNIS}}$ can converge even when the individual estimates $\hat{E}_1$ and $\hat{E}_2$ do not. Most notably, this can occur if $q(x) = 0$ in a finite-measure region where $f(x) = \mu$, which results in the asymptotic biases from the two estimators canceling.

Proof See Appendix A.2 ■

Remark 3 *In the standard TABI case—where $\hat{E}_1$ and $\hat{E}_2$ are independent estimators such that* $\mathrm{Corr}[\xi_1, \xi_2] = 0$*—this result straightforwardly simplifies to*

$$\mathbb{E}\left[(\hat{\mu} - \mu)^2\right] = \frac{\sigma_2^2 \mu^2}{E_2^2}\left(\kappa^2 + 1\right) + \mathcal{O}(\epsilon). \tag{15}$$

We can now examine the relative performance of TABI and SNIS in different settings by considering the effect of σ_2 and κ on the MSE in (14). While σ_2 is obviously an indicator for how effective an estimator $\hat{E}_2$ is for E_2 (with smaller values indicating the estimator is more effective), we can think of κ as representing the relative effectiveness of the two estimators (with smaller values indicating that $\hat{E}_1$ is the relatively more effective estimator). We see from (14) that smaller values of σ_2 are always preferable, while the optimal value of κ for a given σ_2 varies between 0 and 1 depending on $\mathrm{Corr}[\xi_1, \xi_2]$.

For SNIS, it is very difficult to independently control σ_2 and κ for a given problem: because $\hat{E}_1$ and $\hat{E}_2$ share the same proposal, we typically cannot force σ_2 to be small without causing κ $(= \sigma_1/(\mu\sigma_2))$ to explode; if we drive $\sigma_2 \to 0$, this results in $\kappa \to \infty$. Moreover, the larger the mismatch between $p(x, y)$ and $p(x, y)f(x)$, the harder it is to manage this trade-off effectively because the more difficult it becomes to have a proposal that keeps both σ_1 and σ_2 small. This yields the expected result that the errors for SNIS typically become large when the mismatch is large. For TABI, we can control κ for a given σ_2 through separately ensuring a good proposal for both $\hat{E}_1$ and $\hat{E}_2$, and, if desired, by adjusting M and N (relative to a fixed budget $M + N$). Consequently, we can achieve better errors than SNIS through this extra control.

On the other hand, as $p(x, y)$ and $p(x, y)f(x)$ become increasingly well–matched, then $\kappa \to 1$ and we find that TABI has less to gain over SNIS. In fact, we see that TABI (with non–optimal proposals) can potentially be worse than SNIS in this setting because here SNIS typically produces $\mathrm{Corr}[\xi_1, \xi_2]^2 \approx 1$ as using the same set of samples when $f(x)$ is near constant means $\hat{E}_1$ and $\hat{E}_2$ will become almost direct scalings of each other, thereby typically leading to their errors becoming highly correlated. This then causes a canceling effect, potentially leading to very low errors. By contrast the standard TABI approach has $\mathrm{Corr}[\xi_1, \xi_2] = 0$ because it uses independent estimators. We note though that, in some scenarios, it may be possible to mitigate this by correlating the estimates, e.g. through using common random numbers.

Thus, in summary, we see that the gains from using TABI will typically be largest when there is a significant mismatch between $p(x, y)$ and $p(x, y)f(x)$, whereas when these are well–matched it may be less helpful and, in extreme cases, potentially even harmful. We emphasize though that the optimal TABI estimator is always better than the optimal SNIS estimator as per Theorem 1; these results are more an insight into typical behavior when optimal proposals cannot be achieved.

3.6. Optimal Sample Allocation

An interesting corollary from Theorem 2 is that we can use it to derive the asymptotically optimal allocation of samples for TABI given a budget $T = N + M$. Starting with (15), we can find the optimal N simply by setting the derivative for the MSE with respect to N to zero which yields

$$N^* = \frac{\varsigma_1}{\varsigma_1 + \mu\varsigma_2} T = \frac{\varsigma_1/E_1}{\varsigma_1/E_1 + \varsigma_2/E_2} T,$$

where $\varsigma_1 = \sqrt{\mathrm{Var}_{q_1(x)}\left[f(x)p(x, y)/q_1(x)\right]}$ and $\varsigma_2 = \sqrt{\mathrm{Var}_{q_2(x)}\left[p(x, y)/q_2(x)\right]}$ are the standard deviations of the one sample estimators for E_1 and E_2 respectively. We thus see that it is optimal to use a number of samples proportional to the relative standard deviation of the one–sample estimator (i.e. its standard deviation divided by its true value). This is intuitively what one would expect, as

it corresponds to estimating each term to the same relative degree of accuracy. Note, that the result can also straightforwardly be extended to the general TABI setting where $f(x)$ is not bounded, for which we get (defining ς_1^+ and ς_1^- in an analogous manner)

$$N^* = \frac{\varsigma_1^+/E_1^+}{\varsigma_1^+/E_1^+ + \varsigma_1^-/E_1^- + \varsigma_2/E_2}\,T, \qquad K^* = \frac{\varsigma_1^-/E_1^-}{\varsigma_1^+/E_1^+ + \varsigma_1^-/E_1^- + \varsigma_2/E_2}\,T.$$

3.7. Related Work

We believe that the complete form of the TABI estimator in (11) has not previously been suggested in the literature (other than in the earlier version of this work, Goliński et al. 2019), nor the applications and extensions presented later in the paper considered. However, individual elements of the estimator have previously been noted.

Firstly, the general idea of using multiple proposals is well established through the concept of multiple importance sampling (MIS) (Veach and Guibas, 1995; Owen and Zhou, 2000; Cornuet et al., 2012; Elvira et al., 2019). The high–level idea of MIS is to draw samples from a set of different proposals before properly weighting them according to the target distribution. There transpires to be a number of different valid approaches to both the sampling and the weighting (see, e.g., Elvira et al. 2019 and the references therein), with many based around implied proposal distributions and Rao-Blackwellized estimators.

MIS is closely related to the idea of positivisation that we used in breaking the numerator of the target expectation, E_1, into E_1^+ and E_1^- in (10). Indeed, in Appendix C we show how one can use ideas from MIS to formulate a distinct approach for estimating E_1 that shares TABI's compelling theoretical properties. Critically though, existing MIS approaches still rely on self-normalization and so are still subject to the previously demonstrated bounds for the performance of SNIS. They also do not naturally allow for the applications and extensions of TABI we cover in subsequent sections, such as target–aware adaptive sampling, target–aware amortized inference, and using base estimators other than importance sampling.

The use of two separate proposals for E_1 and E_2 (i.e. Eq 8), on the other hand, was recently independently suggested by Lamberti et al. (2018) in work published concurrently to an early version of our own.[6] Though they do not consider adaptive or amortized sampling as we will later, Lamberti et al. (2018) do instead consider an interesting alternative static estimation approach. Namely, they first draw a single set of samples from a fixed proposal, as per SNIS, but then apply an optimization procedure to learn two linear mappings for these samples, thereby implicitly defining two new proposals. This produces an estimator similar to (8) where $N = M$ and $x'_{1:N}$ is a linear mapping of $x_{1:M}$. They show that this offers small improvements compared to the optimal SNIS sampler (reducing the error by around a factor of 1.5 to 2) for some simple one-dimensional problems.

However, it remains to be seen whether this can still be beneficial in more complex or multidimensional settings, while the approach has some significant drawbacks compared to TABI. For example, it cannot match TABI estimator's theoretical capabilities or small sample size performance because it relies on samples from the proposal to learn the linear mappings themselves, such that these maps will be inaccurate for small sample sizes. Their approach also has some potential issues with cost, as the optimization procedure applied is liable to be substantially more costly than the original sampling procedure itself.

3.7.1. Alternative Approaches

We have discussed at length how target information can be incorporated into inference when using importance sampling techniques. We now take a short interlude to discuss alternatives approaches.

6. A preliminary version of our TABI framework was first presented in a short-form paper at the Workshop on Uncertainty in Deep Learning as part of the 2018 Conference on Uncertainty in Artificial Intelligence.

We highlight that none of the discussed approaches have the theoretical advantages of TABI, while none have been used in the amortized inference context we discuss in Section 5.

Bridge sampling (Meng and Wong, 1996; Gelman and Meng, 1998; Meng and Schilling, 2002; Wang and Meng, 2016; Gronau et al., 2017) is an approach for estimating the ratio of the normalizing constants for two unnormalized densities given a set of samples from each, typically generated using MCMC methods. It relies on exploiting the overlapping region of the two densities. In the case where $f(x) > 0 \, \forall x$, it can be used to incorporate target information into the inference process by taking these unnormalized densities to be $p(x, y)f(x)$ and $p(x, y)$ respectively. It also shares a, predominantly superficial, similarity to our TABI framework, in that it uses two independent sub-estimators as a mechanism to estimate the target ratio. However, these estimators target different expectations than those in our framework and are based on leveraging the overlap between the two distributions, rather than separately estimating the two terms. Moreover, its underlying motivation and characteristics are highly distinct from our own. For example, its efficiency is heavily limited by the level of overlap between the distributions (Meng and Wong, 1996; Meng and Schilling, 2002; Frühwirth-Schnatter, 2004). This, along with the restriction that $f(x)$ must be strictly positive, means it is typically poorly suited to our setting.

Umbrella sampling (Torrie and Valleau, 1977a; Mezei, 1987; Kästner, 2011; Thiede et al., 2016; Matthews et al., 2018) is an MCMC approach, most commonly used for free–energy estimation, that allows one to force additional sampling in regions of interest using biasing functions, also known as window functions or umbrellas, before applying corrective factors to remove the resulting biases. It is often used either to make it easier to sample from a multi-modal distribution, or to force additional sampling in the tails of the distribution. In principle, it can also be used to construct posterior estimates whose accuracy is focused on regions of interest, such as where $|f(x)|$ is large. Though a potentially useful mechanism for incorporating target information, umbrella sampling requires the additional complex estimation of normalizing constants for each of the constructed biased distributions (see, e.g., Matthews et al. 2018, Section 2.1). Carrying this out reliably can be very difficult, particularly when there is significant discrepancy between the umbrella distributions, something which is typically difficult to avoid. Even if this can be overcome, the theoretically achievable performance of umbrella sampling is still limited, unlike TABI estimators.

Another issue with umbrella sampling is that the umbrellas used must be manually chosen by the user in a manner that balances both the need for overlap between umbrellas and successful targeting of important regions of the space. This is further compounded by the fact that increasing the number of umbrellas naturally leads to the cost of the algorithm increasing. Perhaps because of these issues, umbrella sampling has seen little use as a general–purpose sampling strategy, despite its successful application to a wide variety of specific sampling problems in physics and chemistry (Torrie and Valleau, 1977b; Virnau and Müller, 2004).

Lacoste-Julien et al. (2011) and Cobb et al. (2018) consider a problem setting that is related to our own: calibrating the output of a Bayesian inference to some loss function defined with respect to a decision task. Their focus is on constructing variational posterior approximations that lead to decisions with low posterior risk. They highlight that different metrics for the quality of the posterior approximation can lead to vastly different levels of calibration between the asserted approximation error and the error in the final decision. To account for this, they introduce a loss–calibrated expectation maximization approach that uses information from the loss function to construct well–calibrated variational approximations to the posterior, such that if a good approximation is achieved in their framework, this implies the decision taken will have low posterior risk.

4. Target–Aware Adaptive Importance Sampling

In the previous section, we showed how our TABI framework can be used to produce highly efficient estimators for expectations, given access to effective proposals. However, we did not consider how such effective proposals might be achieved other than to note their optimal forms. We now show

how adaptive importance sampling (AIS) methods can be used to learn such proposals in an online manner and how by combining them with our TABI framework we can produce effective adaptive methods for performing target–aware inference. Notably, we will find that, in some settings, these methods are able to both theoretically and empirically achieve convergence rates superior to standard Monte Carlo estimators such as SNIS.

4.1. Adaptive Importance Sampling (AIS)

The performance of importance sampling approaches, and indeed almost all MC methods, is critically dependent on the proposal used. However, hand-crafting proposals is typically extremely difficult; knowing a good proposal is tantamount to already having a good approximation of the target distribution. To address this issue, AIS methods exploit information gathered from previously drawn samples to adaptively update the proposal and try to improve it for future iterations (Oh and Berger, 1992; Gelman and Meng, 1998; Cappé et al., 2004; Cornebise et al., 2008; Cornuet et al., 2012; Martino et al., 2017; Bugallo et al., 2017; Rainforth et al., 2018b; Lu et al., 2018; Portier and Delyon, 2018).

In general, AIS methods try to adapt the proposal distribution $q(x)$ to match some target distribution $\pi(x)$. As before, $\pi(x)$ can typically only be evaluated up to a normalizing constant, i.e. $\pi(x) = \gamma(x)/Z$ where $\gamma(x)$ can be evaluated pointwise and Z is unknown, such that AIS methods generally rely on self-normalization when evaluating expectations. Though a wide range of approaches have been suggested for adapting $q(x)$ (see, e.g., Bugallo et al. 2017 for a recent review), these generally share a common framework wherein they alternate between constructing a batch of weighted samples using the current proposal $q_t(x)$ and updating the proposal $q_t(x) \to q_{t+1}(x)$ using these samples.

For the first step, the only complication is in choosing how to set the weights. The simplest common approach is to just weight according to the proposal the sample was drawn from, such that

$$x_{r,t} \overset{\text{i.i.d.}}{\sim} q_t(x), \quad w_{r,t} = \frac{\gamma(x_{r,t})}{q_t(x_{r,t})}, \quad \forall t \in \{1, \dots, T\},\, r \in \{1, \dots, R\} \tag{16}$$

where we have $N = RT$ total samples. However, there are a number of schemes that try and improve on this using ideas from MIS, specifically by using alternative weights based on the implied mixture proposal of different t, see e.g. Tables 3 and 4 in Bugallo et al. (2017). One common thing between the simple weighting scheme given in (16) and these more advanced approaches is that they produce consistent and unbiased estimates of the marginal likelihood (presuming the proposal adaptation is set up to ensure the proposals used always remain valid):

$$\hat{Z} := \frac{1}{TR} \sum_{t=1}^{T} \sum_{r=1}^{R} w_{r,t}, \quad \mathbb{E}[\hat{Z}] = Z, \quad \lim_{R\to\infty} \hat{Z} = Z, \quad \lim_{T\to\infty} \hat{Z} = Z. \tag{17}$$

Here the unbiasedness can be shown straightforwardly by noting that $\mathbb{E}[\hat{Z}] := \frac{1}{TR} \sum_{t=1}^{T} \sum_{r=1}^{R} \mathbb{E}[w_{r,t}]$ and each $\mathbb{E}[w_{r,t}] = Z$. The convergences in the limit of either large T or large R can be shown by a combination of a) the above unbiasedness result; b) noting that, even though the $x_{r,t}$ are correlated across t due to the adaptation, $\mathbb{E}[(w_{r,t} - Z)(w_{r',t'} - Z)] = 0$ unless $r = r'$ and $t = t'$; and c) applying the weak law of large numbers.

For the proposal update step, there is a multitude of different approaches that can be taken. For example, one common approach is to update the proposal by minimizing the Kullback-Leibler (KL) divergence from the current target distribution estimate,

$$\hat{\pi}(x) = \sum_{i=1}^{t} \sum_{r=1}^{R} \bar{w}_{r,i} \delta_{x_{r,i}}(x), \quad \text{where} \quad \bar{w}_{r,i} = \frac{w_{r,i}}{\sum_{j=1}^{t} \sum_{n=1}^{R} w_{j,n}}, \tag{18}$$

to $q_{t+1}(x)$ (Douc et al., 2007; Cappé et al., 2008; Chatterjee et al., 2018; Lu et al., 2018):

$$q_{t+1} := \operatorname*{argmin}_{q \in Q} \int \hat{\pi}(x) \log\left(\frac{\hat{\pi}(x)}{q(x)}\right) dx = \operatorname*{argmin}_{q \in Q} -\sum_{i=1}^{t}\sum_{r=1}^{R} \bar{w}_{r,i} \log\left(q(x_{r,i})\right). \tag{19}$$

Here Q represents the set of valid proposals, usually corresponding to a parameterized proposal definition where the optimization is carried out over these parameters. Actually evaluating (19) (or more typically it gradients) is generally difficult; naively trying to solve it from scratch at each iteration would lead to a $\mathcal{O}(N^2)$ cost. To avoid this, methods typically either a) chose proposal families where the optimization can be done analytically (or at least simply), such as exponential family proposals where q_{t+1} can be found by simply keeping online estimates of sufficient statistics and then moment matching (Cornuet et al., 2012); or b) take a stochastic gradient approach, where q_{t+1} is found by applying a gradient update to the parameters of q_t using only the new samples (Elvira et al., 2015; Müller et al., 2019).

Other common approaches for adapting the proposal include using systems of interacting particles to produce implicit nonparametric proposals (Cappé et al., 2004); constructing MCMC chains targeting $\pi(x)$ and then using proposals centered around these chains (Martino et al., 2017); and recursively partitioning the sample space to construct tree-based proposals (Rainforth et al., 2018b; Lu et al., 2018).

4.2. AIS with a Known Target Function

As explained in the previous section, AIS methods take as input some unnormalized target density $\gamma(x)$ and return a self-normalized set of weighted samples approximating the normalized target $\pi(x)$ as per (18), along with an estimate $\hat{Z}$ for the normalizing constant as per (17). Critically, unlike for MCMC methods, when using AIS to estimate an expectation, $\pi(x)$ does not need to correspond to the distribution that expectation is taken with respect to. Specifically, if we wish to estimate $\mu = \mathbb{E}_{\varrho(x)}[f(x)]$ for some arbitrary distribution $\varrho(x)$ using AIS targeting the unnormalized target density $\gamma(x)$, we simply need to factor our weights in the final IS estimator as follows

$$\mu := \mathbb{E}_{\varrho(x)}[f(x)] = \mathbb{E}_{q(x)}\left[\frac{\varrho(x)f(x)}{q(x)}\right] = \mathbb{E}_{q(x)}\left[\frac{\gamma(x)}{q(x)}\frac{\varrho(x)}{\gamma(x)}f(x)\right] \approx \hat{\mu}_{\text{IS}} = \frac{1}{N}\sum_{n=1}^{N} w_n v_n f(x_n)$$

where $n = (t-1)R + r$ is a flattening of sample indices, $w_n = \gamma(x_n)/q_{t(n)}(x_n)$ are the weights produced by the AIS method, and $v_n = \varrho(x_n)/\gamma(x_n)$ are corrective weights to account for the fact we targeted $\gamma(x)$ rather than $\varrho(x)$.

Analogously, if the normalized density of the reference distribution is unknown, e.g. $\varrho(x) = p(x|y)$, we can use AIS to instead construct an SNIS estimate as follows

$$\mu := \mathbb{E}_{p(x|y)}[f(x)] = \frac{\mathbb{E}_{q(x)}\left[\frac{p(x,y)f(x)}{q(x)}\right]}{\mathbb{E}_{q(x)}\left[\frac{p(x,y)}{q(x)}\right]} = \frac{\mathbb{E}_{q(x)}\left[\frac{\gamma(x)}{q(x)}\frac{p(x,y)}{\gamma(x)}f(x)\right]}{\mathbb{E}_{q(x)}\left[\frac{\gamma(x)}{q(x)}\frac{p(x,y)}{\gamma(x)}\right]} \approx \hat{\mu}_{\text{SNIS}} = \frac{\sum_{n=1}^{N} w_n v_n f(x_n)}{\sum_{n=1}^{N} w_n v_n} \tag{20}$$

where we now have $v_n = p(x_n, y)/\gamma(x_n)$.

The ability of AIS methods to target a different distribution to that which the expectation is taken with respect to means that they can, at least in principle, incorporate information about $f(x)$ by choosing an appropriate $\gamma(x)$ that captures this information. In the case where the reference distribution is normalized, this is indeed straightforward: from Section 2.2 we know that $q^*_{\text{IS}}(x) \propto \varrho(x)|f(x)|$ and so we simply take $\gamma(x) = \varrho(x)|f(x)|$, such that our adaptation tries to take $q(x)$ towards $q^*_{\text{IS}}(x)$. In the setting where $f(x) \geq 0\,\forall x$ and the family of $q(x)$ contains the true optimal proposal q^*_{IS}, some methods based around this approach have been shown to achieve faster convergence rates than those of standard MC methods (Zhang, 1996; Portier and Delyon, 2018).

However, in the SNIS case, incorporating information about $f(x)$ transpires to be far more difficult. We know from Section 2.3 that the optimal target would be $\gamma(x) = p(x,y)|f(x) - \mu|$ as this will try to produce the optimal proposal $q^*_{\text{SNIS}}(x) \propto p(x,y)|f(x) - \mu|$. Unfortunately, this is not generally a viable choice because μ is, by construction, unknown. One method that is sometimes used as a substitute is to take $\gamma(x) = p(x,y)|f(x)|$, i.e. treating the problem as if we were not performing self-normalization. However, as we showed in Figure 1, this can lead to very poor estimates for $p(y)$, i.e. the denominator in (20), and thus in turn μ. One could instead try to use $\gamma(x) = p(x,y)|f(x) - c|$ for some constant c, but again this is far from satisfactory due to the fact that choosing an appropriate value of c is equivalent to already having information about μ, while choosing an inappropriate c can lead to learning a highly inappropriate proposal.

A perhaps more principled, but rarely taken, approach would be to use $\gamma_{t+1}(x) = p(x,y)|f(x) - \hat{\mu}_t|$ where $\hat{\mu}_t$ represents the running estimate of μ, such that $\gamma(x)$ itself also adapts as the algorithm is run. This, however, has its own shortfalls. Firstly, if the initial proposal is poor, we can get a chicken and egg situation where we need a good estimate of μ to form a good target for our proposal, but without a good target for our proposal we will struggle to achieve a good estimate for μ. Further, such an approach is susceptible to computational issues such that it may be difficult to avoid a $\mathcal{O}(N^2)$ cost: many of the adaptation approaches discussed in the last section rely on online updates, but if $\gamma(x)$ is itself changing, it may be necessary to re-evaluate previously sampled points. Even if this can be avoided, the fact that $\gamma(x)$ is not static can still be a serious complication factor for implementation; some methods may not even be able to cope with this at all.

Due to this multitude of issues, it is often common practice when using AIS methods for SNIS estimators to simply ignore information about $f(x)$ and instead target the posterior, that is take $\gamma(x) = p(x,y)$. However, this is clearly far from a satisfactory solution and will perform poorly whenever $p(x,y)$ and $p(x,y)|f(x)|$ are not well–matched.

4.3. TAAIS

In the last section, we explained how incorporating information about $f(x)$ is straightforward for AIS methods when performing standard importance sampling, but can be extremely challenging when relying on self-normalization. We now show how our TABI framework provides a mechanism to get around this problem, while also opening the door to achieving better estimates, and even in some cases better convergence rates, compared with the optimal SNIS sampler.

We refer to our approach as TAAIS, which stands for *target-aware adaptive importance sampling*. In short, TAAIS splits up our target expectation into $\mu = (E_1^+ - E_1^-)/E_2$ as per the general TABI framework, and then runs AIS separately for each, using a tailored $\gamma(x)$ in each case. Because each of these component expectations do not require self-normalization, they fit neatly into the category of problems where AIS can straightforwardly incorporate information about $f(x)$. Specifically, we can use the targets

$$\gamma_1^+(x) = p(x,y)f^+(x) \tag{21a}$$

$$\gamma_1^-(x) = p(x,y)f^-(x) \tag{21b}$$

$$\gamma_2(x) = p(x,y). \tag{21c}$$

Therefore, not only does TAAIS maintain the benefits over SNIS of the general TABI framework discussed in Section 3.2, it also solves the difficulties AIS has in choosing an appropriate target distribution for the adaptation. The choice of the targets in (21) also offers a further convenience: the component expectation estimates are given simply by the marginal likelihood estimates produced by the AIS algorithm (as the expected weight is the normalizing constant of the target and these normalizing constants are E_1^+, E_1^-, and E_2 respectively), such that

$$\hat{\mu}_{\text{TAAIS}} = \frac{\hat{Z}_1^+ - \hat{Z}_1^-}{\hat{Z}_2}. \tag{22}$$

We can thus summarize the TAAIS approach as the following simple algorithm:

1. Run AIS separately for each of $\gamma_1^+(x)$, $\gamma_1^-(x)$, and $\gamma_2(x)$ defined as per (21);
2. Combined the returned marginal likelihood estimates as per (22) to estimate μ.

4.4. Theoretical Advantages

We now demonstrate a theoretical result that shows TAAIS is capable of achieving substantially improved convergence rates over the standard approach of using AIS methods with the SNIS estimator (which we will refer to as SNIS-AIS) when the distribution family for our proposal contains the target distribution and our proposal adaptation scheme asymptotically converges to the optimal proposal. At a high–level, this result stems from the fact that when self-normalization is not required, AIS methods are able to produce faster convergence rates than static MC estimators under certain conditions (Portier and Delyon, 2018). As TAAIS is comprised of three independent such estimators, it is able to retain this property. More precisely we have the following result.

Theorem 4 *Let the three AIS marginal likelihood estimators used by TAAIS be given by*

$$\hat{Z}_1^+ := \frac{1}{N}\sum_{n=1}^{N} w_{1,n}^+, \quad \hat{Z}_1^- := \frac{1}{K}\sum_{k=1}^{K} w_{1,k}^-, \quad \hat{Z}_2 := \frac{1}{M}\sum_{m=1}^{M} w_{2,m},$$

where $w_{1,n}^+$, $w_{1,k}^-$, and $w_{2,m}$ are all valid importance sampling weights for $\gamma_1^+(x)$, $\gamma_1^-(x)$, and $\gamma_2(x)$ respectively as defined in (21)*. Assume that each of these estimators is generated independently of each other. If the proposal adaption for each AIS estimator converges such that there are some constants $s_1^+, s_1^-, s_2, a, b, c > 0$ for which the following bounds hold for all $n, k, m > 0$*

$$\mathrm{Var}[w_{1,n}^+] \leq \frac{s_1^+}{n^a}, \quad \mathrm{Var}[w_{1,k}^-] \leq \frac{s_1^-}{k^b}, \quad \mathrm{Var}[w_{2,m}] \leq \frac{s_2}{m^c},$$

then the MSE of $\hat{\mu}_{\mathrm{TAAIS}} := (\hat{Z}_1^+ - \hat{Z}_2^-)/\hat{Z}_2$ converges in expectation as follows

$$\mathbb{E}\left[(\hat{\mu}_{\mathrm{TAAIS}} - \mu)^2\right] \leq \frac{1}{E_2^2}\left(\frac{s_1^+}{N^2}h(a,N) + \frac{s_1^-}{K^2}h(b,K) + \frac{\mu^2 s_2}{M^2}h(c,M)\right) + \mathcal{O}(\epsilon) \tag{23}$$

where $\mathcal{O}(\epsilon)$ represents asymptotically dominated terms, and

$$h(\alpha, L) = \begin{cases} \frac{L^{1-\alpha}}{1-\alpha}, & \text{if} \quad 0 < \alpha < 1 \\ \log(L) + \eta, & \text{if} \quad \alpha = 1 \\ \zeta(\alpha), & \text{if} \quad \alpha > 1 \end{cases}$$

where $\eta \approx 0.577$ is the Euler-Mascheroni constant and ζ is the Riemann-zeta function.

Proof See Appendix A.3. ■

Remark 5 *Presuming that we set $M \propto K \propto N$ and that each of the proposals converges at the same rate such that the variance on their n^{th} weight is $O(1/n^a)$ for some a, then this result implies three different convergence rates depending on the value of a:*

$$\mathbb{E}\left[(\hat{\mu}_{\mathrm{TAAIS}} - \mu)^2\right] = \begin{cases} \mathcal{O}\left(\frac{1}{N^{1+a}}\right), & \text{if} \quad 0 < a < 1 \\ \mathcal{O}\left(\frac{\log(N)}{N^2}\right), & \text{if} \quad a = 1 \\ \mathcal{O}\left(\frac{1}{N^2}\right), & \text{if} \quad a > 1 \end{cases}$$

This result shows that TAAIS is able to improve on the standard Monte Carlo convergence rate of $\mathcal{O}(1/N)$ if our proposal family contains the target distribution and our adaptation scheme is sufficiently powerful to ensure the proposal converges to the optimal proposal. When this is the case, we expect that $a = 1$ will often be typical, leading to a convergence rate of $\mathcal{O}(\log(N)/N^2)$, a substantial improvement on SNIS. The rationale for this is that $a = 1$ corresponds to the variance of the weights themselves converging to zero at the Monte Carlo error rate, as might be expected when using, for example, a moment-matching AIS method (for which our parameters are themselves taken from a Monte Carlo estimate). This assertion is also consistent with our empirical observations in the next section, along with the theoretical results of Zhang (1996); Lu et al. (2018).

4.5. Experiments

Having confirmed the theoretical capabilities of TAAIS in the last section, we now show that it is also able to provide substantial empirical benefits over SNIS-AIS methods in practice. Code for these experiments and others is available at `https://github.com/twgr/tabi`.

4.5.1. Gaussian Example

We first show that the theoretical $\mathcal{O}(\log(N)/N^2)$ convergence rate can be achieved in practice when the proposal families contain their respective target distributions. For this, we use a simple Gaussian model defined as

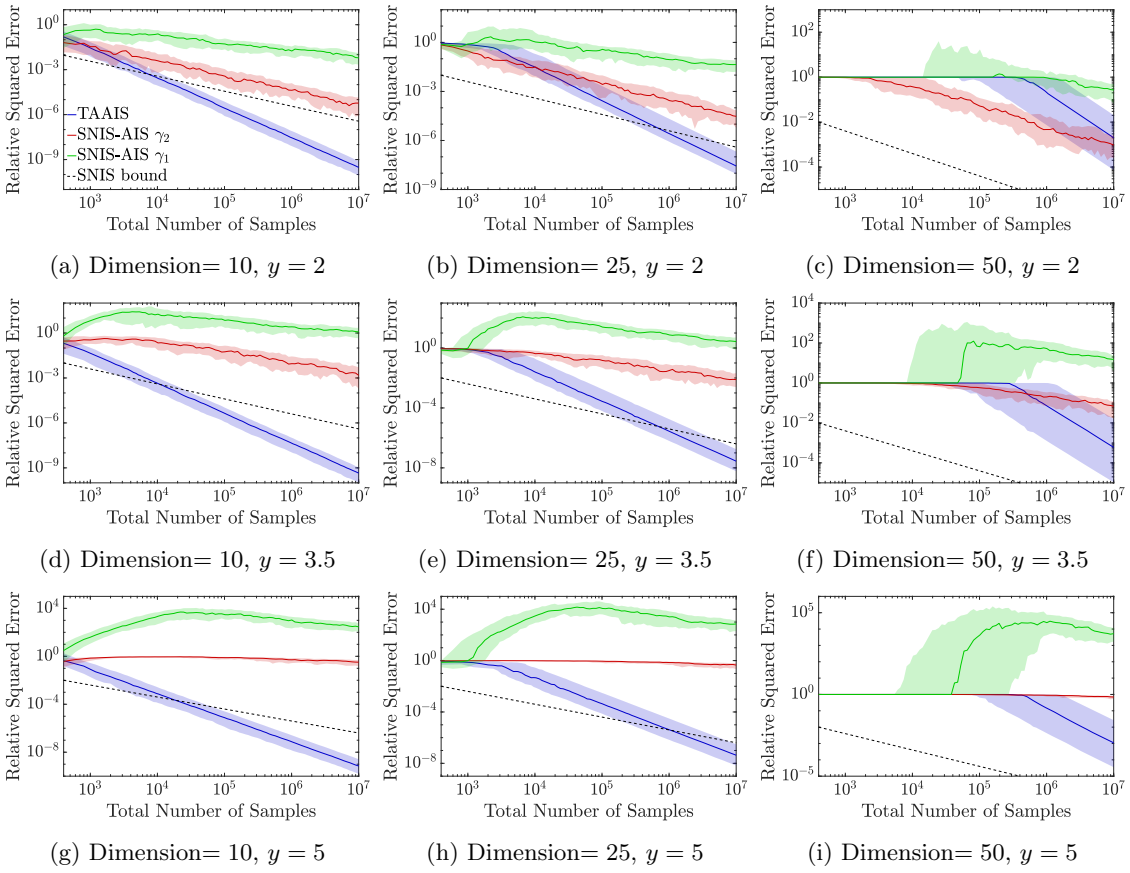
$$p(x) = \mathcal{N}(x; 0, I), \quad p(y|x) = \mathcal{N}\left(-\frac{y}{\sqrt{D}}\mathbf{1}; x, I\right), \quad f(x) = \mathcal{N}\left(x; \frac{y}{\sqrt{D}}\mathbf{1}, \frac{1}{2}I\right) \tag{24}$$

where D is the dimensionality, I is the identity matrix, $\mathbf{1}$ is a vector of ones with length D, and y represents the radial distance of the observation from the origin, such that it dictates the level of separation between the distributions. Note that we are implicitly parameterizing the function by y, where this is a fixed variable for any given experiment. This problem effectively equates to that of calculating the posterior predictive density of a point at $(y/\sqrt{D})\mathbf{1}$ under a Gaussian unknown mean model with prior centered at the origin and an observation at $(-y/\sqrt{D})\mathbf{1}$.

Though simple, this problem has a number of useful characteristics that motivate its use as a testbed. Firstly, we can easily calculate ground truth values for μ and the SNIS bound given in (4). Secondly, we can arbitrarily vary the difficulty of the problem through changes to y and D: the larger the value of y the larger the discrepancy between $p(x, y)$ and $p(x, y)f(x)$, while increasing the dimensionality inevitably makes the problem harder. Thirdly, because both the posterior and function-scaled posterior are Gaussian, we can easily construct a proposal family that satisfies the assumptions of Theorem 4. Namely, we use a moment matching approach where the proposal is a Gaussian whose mean and diagonal covariance are based on the samples taken thus far. Using the superscript d to denote different dimensions, each proposal thus takes the form

$$q_t(x) = \mathcal{N}(x; m_t, \Sigma_t) \quad \text{where}$$
$$m_t^d = \sum_{i=1}^{t-1}\sum_{r=1}^{R} \bar{w}_{r,i} x_{r,i}^d, \quad \Sigma_t^{d,d} = \max\left(\Sigma_{\min}, \sum_{i=1}^{t-1}\sum_{r=1}^{R} \bar{w}_{r,i}\left(x_{r,i}^d\right)^2 - \left(m_t^d\right)^2\right),$$

the off-diagonal terms in Σ_t are all zero, and $\Sigma_{\min}$ is a fixed minimum variance to ensure proposals are guaranteed to remain valid for the distribution we are targeting. We note that as $f(x) \geq 0\, \forall x$, we need not calculate E_1^- for this problem. We take $\Sigma_{\min} = 0.4^2$ when targeting $\gamma_2(x)$ and $\Sigma_{\min} = 0.2^2$ when targeting $\gamma_1(x)$. We draw $R = 200$ samples from each $q_t(x)$ between each proposal update, with the proposal updates themselves performed by making an online update to running estimates of the moments to avoid unnecessary recalculations and ensure the cost of the proposal update remains constant as the number of iterations increases. We further take $N = M$ for TAAIS.



(a) Dimension= 10, $y = 2$ (b) Dimension= 25, $y = 2$ (c) Dimension= 50, $y = 2$

(d) Dimension= 10, $y = 3.5$ (e) Dimension= 25, $y = 3.5$ (f) Dimension= 50, $y = 3.5$

(g) Dimension= 10, $y = 5$ (h) Dimension= 25, $y = 5$ (i) Dimension= 50, $y = 5$

Figure 3: Convergence plots of relative squared error (as per Eq 12) for TAAIS and SNIS-AIS on the Gaussian model defined in (24) for different dimensionalities and separations y. The solid line represents the median across 100 runs, shading the 25% and 75% quantiles. The AIS method used is the moment matching approach described in Section 4.5. Note here that TAAIS is actually slighter quicker than SNIS-AIS for the same number of total samples drawn because it only has to evaluate $f(x)$ for half of these samples. As such, the relative real-time performance of TAAIS is actually slightly better than these comparisons. We note the overhead cost of the adaptation is the same for all methods and is lower than the combined cost of sampling and evaluating the weights.

We now consider the different values of $D \in \{10, 25, 50\}$ and $y \in \{2, 3.5, 5\}$, giving nine variations of the problem of varying difficulty. We compare TAAIS to two SNIS-AIS variants, one using the target $\gamma(x) = \gamma_1(x) = p(x, y)f(x)$ and one using $\gamma(x) = \gamma_2(x) = p(x, y)$, with that the latter of these corresponding to a standard inference approach that does not use information about the target. We also compare to the theoretically optimal SNIS sampler, i.e. the bound given in (4). The results are given in Figure 3 and Table 1.

We see that TAAIS comfortably beats the SNIS-AIS samplers targeting γ_1 and γ_2 in all cases except $D = 50$ and $y = 2$, for which the final errors are comparable for TAAIS and SNIS-AIS using γ_2. In all cases, TAAIS can be seen to give an empirical convergence rate that fits perfectly with the $\mathcal{O}(\log(N)/N^2)$ rate predicted by Theorem 4; indeed the nature of this convergence is remarkably

y	Dimension	SNIS-AIS γ_2	SNIS-AIS γ_1	TAAIS
2	10	-12.85 ± 0.26	-5.58 ± 0.21	$\mathbf{-22.25 \pm 0.20}$
	25	-10.96 ± 0.22	-3.74 ± 0.19	$\mathbf{-17.14 \pm 0.21}$
	50	$\mathbf{-7.37 \pm 0.22}$	-2.00 ± 0.19	$\mathbf{-6.91 \pm 0.39}$
3.5	10	-7.01 ± 0.22	-0.21 ± 0.14	$\mathbf{-22.19 \pm 0.27}$
	25	-5.23 ± 0.25	0.36 ± 0.21	$\mathbf{-17.16 \pm 0.29}$
	50	-3.12 ± 0.19	2.32 ± 0.25	$\mathbf{-7.88 \pm 0.46}$
5	10	-1.58 ± 0.18	5.31 ± 0.16	$\mathbf{-21.21 \pm 0.22}$
	25	-0.95 ± 0.14	5.83 ± 0.25	$\mathbf{-16.96 \pm 0.30}$
	50	-0.66 ± 0.15	8.12 ± 0.30	$\mathbf{-7.18 \pm 0.39}$

Table 1: Comparison of final results for Gaussian example when allowing a budget of 10^7 total samples (such that $N = M = 5 \times 10^6$ for TAAIS and $N = 10^7$ for SNIS-AIS). Unlike Figure 3, the values shown here are the mean and standard error of the log relative squared error across 100 runs. Results shown in bold represent either the best achieved error for that problem or a result where we cannot reject the hypothesis that this result has the same mean as the best result at the 5% significance level of a t-test.

stable and consistent with the theory across the different runs. In all the 10 and 25 dimensional cases, TAAIS further outperformed the optimal SNIS sampler within the allocated budget.

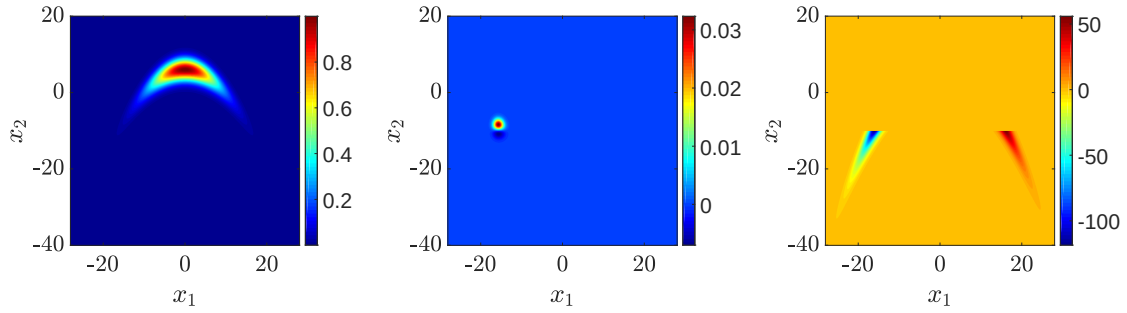
The behavior of TAAIS in 50 dimensions was particularly interesting: a large number of samples were required before the AIS methods were able to start effectively adapting, but once this occurred, the TAAIS sampler starts converging very quickly, despite the fact that this represents an unusually high dimensionality for AIS methods. Though the performance of the SNIS-AIS baselines quickly diminishes with increasing y (representing increasing mismatch between $p(x, y)$ and $p(x, y)f(x)$), the performance of TAAIS was almost completely unaffected. Another result of note was the particularly poor behavior of SNIS-AIS targeting $\gamma_1(x)$. This is most likely due to the fact that $q(x) \propto \gamma_1(x)$ would actually represent an invalid proposal for an estimator for $\hat{Z}_2$ and so this choice of adaptation scheme potentially leads to a non-convergent overall SNIS estimator.

4.5.2. Banana Example

In the previous sections, we showed that TAAIS is able to achieve improved convergence rates compared with the optimal SNIS sampler when the proposal family contains the target distributions. We now investigate whether it still offers practical benefits in a problem setup where this does not hold. For this, we consider the classic two-dimensional banana problem where[7]

$$p(x, y) \propto \exp\left(-\frac{1}{2}\left(0.03x_1^2 + \left(\frac{x_2}{2} + 0.03\left(x_1^2 - 100\right)\right)^2\right)\right), \tag{25}$$

7. Though there is actually no observed data y here, we maintain $p(x, y)$ as a notation for an unnormalized density to avoid confusion with the AIS targets.



(a) $p(x, y)$ (b) $p(x, y)f_a(x)$ (c) $p(x, y)f_b(x)$

Figure 4: Visualizations of banana distribution and its product with the functions $f_a(x)$ and $f_b(x)$.

along with two different target functions

$$f_a(x) := (x_2 + 10) \cdot \exp\left(-\frac{1}{4}(x_1 + x_2 + 25)^2\right) \tag{26a}$$

$$f_b(x) := (x_1 - 2)^3 \cdot \mathbb{I}(x_2 < -10). \tag{26b}$$

Visualizations of $p(x, y)$, $p(x, y)f_a(x)$, and $p(x, y)f_b(x)$ are shown in Figure 4. We note that both $f_a(x)$ and $f_b(x)$ have regions where they return negative values, such that the full TAAIS estimator is required in both cases.

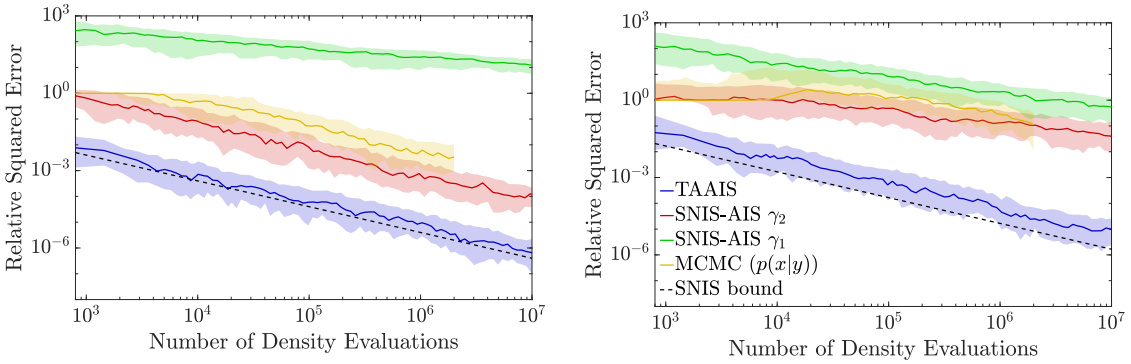
Because of the more complex target densities for this problem, we employ a more advanced AIS method, namely the parallel interacting Markov adaptive importance sampling (PI-MAIS) approach of Martino et al. (2017). PI-MAIS is a state-of-the-art AIS approach that, given a target distribution $\gamma(x)$, runs S independent MCMC samplers each targeting $\gamma(x)$. It then uses the locations of these chains to, at each iteration, construct a mixture of Gaussians proposal distribution, with each component centered on the location of one of the chains such that the proposal at iteration t is

$$q_t(x) = \frac{1}{S}\sum_{s=1}^{S} \mathcal{N}(x; \tilde{x}_{s,t}, \Sigma)$$

where $\tilde{x}_{s,t}$ is the location of the s^{th} MCMC chain at the t^{th} iteration and Σ is a fixed covariance matrix. We use $S = 40$ such chains and draw $R = 200$ samples from each $q_t(x)$.[8] We use a random walk Gaussian proposal for the MCMC samplers with covariance Σ_{MCMC}, and choose the following covariance setups for the different problem configurations: $[f_a, \gamma_2]$ $\Sigma = 36I$ and $\Sigma_{\text{MCMC}} = 2.25I$; $[f_a, \gamma_1^+$ and $\gamma_1^-]$ $\Sigma = 2.25I$ and $\Sigma_{\text{MCMC}} = 2.25I$; $[f_a, \gamma_1]$ $\Sigma = 9I$ and $\Sigma_{\text{MCMC}} = 2.25I$; $[f_b$, all $\gamma]$ $\Sigma = 16I$ and $\Sigma_{\text{MCMC}} = I$.

We compare TAAIS to the same baselines as our Gaussian example (taking $\gamma_1(x) = p(x, y)|f(x)|$ for the corresponding SNIS-AIS estimator), the results of which are given in Figure 5. We further include a comparison to conventional MCMC sampling by using the $\tilde{x}_{s,t}$ samples generated by the PI-MAIS sampler targeting $\gamma_2(x) \propto p(x|y)$. We see that MCMC sampling of posterior and both SNIS-AIS methods were relatively ineffective for both target functions compared with TAAIS. In particular, SNIS-AIS targeting $\gamma_1(x)$ was poor throughout, while all methods other than TAAIS and the optimal SNIS sampler struggled for $f_b(x)$. Perhaps unsurprisingly, given that we are using a relatively simple proposal class that is not able to completely encapsulate its targets, TAAIS

8. In practice, we Rao-Blackwellize the selection of the mixture component by drawing 5 samples from each of the 40 component Gaussians.



(a) $f_a(x) = (x_2 + 10) \cdot \exp\left(-\frac{1}{4}(x_1 + x_2 + 25)^2\right)$ (b) $f_b(x) = (x_1 - 2)^3 \cdot \mathbb{I}(x_2 < -10)$

Figure 5: Convergence plots of TAAIS and SNIS-AIS on banana example with two different target functions. Conventions as per Figure 3. Here inference is run using PI-MAIS as explained in the text. As in Figure 3, this is a slightly conservative evaluation of the performance of TAAIS due to not taking into account its slightly reduced cost for a given number of density evaluations. Note that the MCMC results stop short of the others as these are generated as a byproduct of the PI-MAIS sampler targeting $\gamma_2(x)$ and only represent $1/6^{\text{th}}$ of the budget spent by this.

was not able to outperform the optimal SNIS sampler and produced convergence rates in line with standard MC, rather than the faster rates observed earlier. Nonetheless, it was still able to produce estimates with errors that were very close to the optimal SNIS sampler, which represents highly effective performance given the non-standard form of the target distribution. In particular, TAAIS still produced estimators with many orders of magnitude lower error than all the baseline approaches.

5. Amortized Monte Carlo Integration

So far we have focused on solving individual inference problems where both the data and target function are presumed to be known and fixed. We now consider an *amortized inference* setting, wherein we wish to learn a predictive model, known as an *amortization artifact*, that can help run inference across a range of different problems (Stuhlmüller et al., 2013; Gershman and Goodman, 2014; Kingma and Welling, 2014; Ritchie et al., 2016; Paige and Wood, 2016; Le et al., 2017, 2018; Webb et al., 2018). Typically, the amortization artifact acts as a regressor from data sets to proposal parameters expected to be effective for the resulting inference problem. As we will later show, our TABI framework can be particularly useful in this setting, due to the ability of inference amortization methods to learn effective proposals.

More specifically, we will introduce *Amortized Monte Carlo integration* (AMCI, Goliński et al. 2019), a framework for amortizing the cost of calculating a family of expectations $\mu(y, \theta) := \mathbb{E}_{\pi(x;y)}[f(x;\theta)]$. Here y represents changeable aspects of the reference distribution $\pi(x;y)$ (e.g., the data set y for the joint density $p(x,y)$ in the Bayesian setting) and θ represents changeable parameters of the target function $f(x;\theta)$. As before, the reference distribution is typically known only up to a normalization constant, i.e. $\pi(x;y) = \gamma(x;y)/Z(y)$ where $\gamma(x;y)$ can be evaluated pointwise, but $Z(y)$ is unknown.

Amortization can be performed over y and/or θ. When amortizing over y, the function does not need to be explicitly parameterized by θ; we just need to be able to evaluate it pointwise. Similarly, when amortizing over θ, the reference distribution can be fixed.

For consistency of notation with the rest of the paper, we will presume a Bayesian inference setting in the rest of this section, i.e. $\pi(x;y)=p(x|y)$, $\gamma(x;y)=p(x,y)$, and $Z(y)=p(y)$.

5.1. Inference Amortization

Inference amortization involves learning an *amortization artifact* that takes a data set as input and produces a proposal tailored to the corresponding inference problem. This amortization artifact typically takes the form of a parametrized proposal, $q(x;\varphi(y;\eta))$, that takes in data y and produces proposal parameters using an *inference network* $\varphi(y;\eta)$, which itself has parameters η and typically corresponds to a neural network. When clear from the context, we will use the shorthand $q(x;y,\eta)$ for this proposal.

Though the exact process varies with context, the inference network is usually trained either by drawing latent-data sample pairs from the joint $p(x,y)$ (Paige and Wood, 2016; Le et al., 2017, 2019), or by drawing mini-batches from a large data set using stochastic variational inference approaches (Hoffman et al., 2013; Kingma and Welling, 2014; Rezende et al., 2014; Ritchie et al., 2016). Once trained, it provides an efficient means of approximately sampling from the posterior of a particular data set, typically using SNIS.

Out of several variants, we focus on the method introduced by Paige and Wood (2016), as this is the one AMCI will build upon. In their approach, η is trained to minimize the expectation of $D_{KL}\left[p(x|y)\;||\;q(x;y,\eta)\right]$ across possible data sets y, giving the objective

$$\begin{aligned}\mathcal{J}(\eta) &= \mathbb{E}_{p(y)}\left[D_{KL}\left[p(x|y)\;||\;q(x;y,\eta)\right]\right] \\ &= \mathbb{E}_{p(x,y)}\left[-\log q(x;y,\eta)\right] + \text{const wrt } \eta \end{aligned} \tag{27}$$

We note that the distribution $p(y)$ over which we are taking the expectation is actually chosen somewhat arbitrarily: it simply dictates how much the network prioritizes a good amortization for one data set over another, different choices are equally valid and imply different loss functions.

This objective requires us to be able to sample from the joint distribution $p(x,y)$ and it can be optimized using stochastic gradient methods, since the gradient can easily unbiasedly estimated using

$$\nabla_\eta \mathcal{J}(\eta) = \mathbb{E}_{p(x,y)}\left[-\nabla_\eta \log q(x;y,\eta)\right] \approx -\frac{1}{B}\sum_{i=1}^{B}\nabla_\eta \log q(x_i;y_i,\eta), \tag{28}$$

where $(x_i,y_i)\sim p(x,y)$ and B is the batch size.

5.2. AMCI

Existing amortized inference methods implicitly evaluate expectations using SNIS (or some other form of a self-normalized estimator (Paige and Wood, 2016; Le et al., 2018)), targeting the posterior as the optimal proposal. Not only is this proposal suboptimal in its failure to use information about the target function when this is available, but it also suffers from the various issues with SNIS samplers that have been outlined throughout the paper.

To overcome this, AMCI instead applies inference amortization in the TABI framework by learning an amortized proposal for each of E_1^+, E_1^-, and E_2, and then using the TABI estimator per (11). We refer to these three amortized proposals respectively as $q_1^+(x;y,\theta,\eta_1^+)$, $q_1^-(x;y,\theta,\eta_1^-)$, and $q_2(x;y,\eta_2)$, with their optimal forms as per Theorem 1, such that the unnormalized target densities for our training will respectively be $p(x,y)f^+(x;\theta)$, $p(x,y)f^-(x;\theta)$, and $p(x,y)$.

Learning $q_2(x;y,\eta_2)$ is equivalent to the standard inference amortization problem and so we will just use the objective given by (27). The approaches for learning $q_1^+(x;y,\theta,\eta_1^+)$ and $q_1^-(x;y,\theta,\eta_1^-)$ are equivalent, other than the function that is used in the estimators. Therefore, for simplicity, we

introduce our amortization procedure in the case where $f(x;\theta) \geq 0 \; \forall x, \theta$, such that we can need only learn a single proposal, $q_1(x; y, \theta, \eta_1)$, for the numerator as per (8). This trivially extends to the full TABI setup by separately repeating the same training procedure for $q_1^+(x; y, \theta, \eta_1^+)$ and $q_1^-(x; y, \theta, \eta_1^-)$. To avoid clutter, we will further drop the subscript on η_1.

5.2.1. Fixed Function $f(x)$

We first consider the scenario where $f(x)$ is fixed (i.e. we are not amortizing over function parameters θ) and hence temporarily drop the dependence of q_1 on θ.

To learn the parameters η for the first amortized proposal $q_1(x; y, \eta)$, we need to adjust the target in (27) to incorporate the effect of the target function. Let $E_1(y) = \mathbb{E}_{p(x)}\big[f(x)p(y|x)\big]$ be the numerator expectation with data set y and $g(x|y) := f(x)\, p(x, y)/E_1(y)$ be the corresponding normalized optimal proposal for q_1. Naively adjusting (27) leads to a doubly intractable objective:

$$\begin{aligned}\mathcal{J}_1'(\eta) &= \mathbb{E}_{p(y)}[D_{KL}\big(g(x|y)||q_1(x; y, \eta)\big)] \\ &= \mathbb{E}_{p(y)}\left[-\int \frac{f(x)\, p(x, y)}{E_1(y)} \log q_1(x; y, \eta)\text{x}\right] + \text{const wrt } \eta, \end{aligned} \tag{29}$$

where the double intractability comes from the fact that we do not know $E_1(y)$ and, at least at the beginning of the training process, we cannot estimate it efficiently either.

To address this, we use our previous observation that the expectation over $p(y)$ in the above objective is chosen somewhat arbitrarily. Namely, it dictates how much relative weighting the objective assigns to different possible data sets during training and not the optimal proposal for each individual data set; disregarding the finite capacity of the network, the global optimum is still always $q_1(x; y, \eta) = g(x|y) \; \forall x, y$. We thus maintain a well-defined objective if we choose a different reference distribution over data sets.

In particular, if we take the expectation with respect to $h(y) \propto p(y)\mu(y) = E_1(y)$, we get

$$\begin{aligned}\mathcal{J}_1(\eta) &= \mathbb{E}_{h(y)}\left[D_{KL}\big(g(x|y)\,||\,q_1(x; y, \eta)\big)\right] \\ &= c^{-1} \cdot \mathbb{E}_{p(x,y)}\left[-f(x) \log q_1(x; y, \eta)\right] + \text{const wrt } \eta \end{aligned} \tag{30}$$

where $c = \mathbb{E}_{p(y)}\big[\mu(y)\big] > 0$ is a positive constant that does not affect the optimization—it is the normalization constant for the distribution $h(y)$—and can thus be ignored. Each term in this expectation can now be evaluated directly, meaning we can again run stochastic gradient descent algorithms to optimize it. Note that this does not require evaluation of the density $p(x, y)$, only the ability to draw samples.

This choice of $h(y)$ can be interpreted as giving larger importance to the values of y which yield larger $p(y)\mu(y)$ (i.e. $E_1(y)$), rather than just larger $p(y)$. This behavior may often actually be beneficial. For example, it is often reasonable to assume that the expected magnitude of our the error in estimating $\mu(y)$ roughly scales as its true value, such that $\mathbb{E}\left[|\hat{\mu}(y) - \mu(y)|\big|y\right] \propto \mu(y)$ approximately holds. This then implies that $h(y) \propto p(y)\mu(y)$ is likely to be an effective proposal for estimating the L_1 risk $\mathbb{E}_{p(y)}\left[\mathbb{E}\left[|\hat{\mu}(y) - \mu(y)|\big|y\right]\right]$, or at least more effective than sampling directly from $p(y)$. As such, $h(y)$ should, in turn, be an appropriate distribution for generating our training data if we want to learn schemes that perform well under this metric (noting that we are not targeting it directly). More generally, this choice of $h(y)$ will typically be preferable to using $h(y) = p(y)$ (in the hypothetical scenario where the latter is viable) whenever we care about an expected absolute loss at test time, while the latter will typically be preferable when we care about an expected *relative* loss, e.g. $\mathbb{E}_{p(y)}[|\hat{\mu}(y) - \mu(y)|/\mu(y)]$.

To allow for greater flexibility between these different settings and arbitrary prioritizing of different y, we further note that we can, in general, choose $h(y) \propto p(y)\mu(y)\omega(y)$ for any positive evaluable

function $\omega : \mathcal{Y} \to \mathbb{R}^+$ to yield a tractable objective of the form

$$\mathcal{J}_1(\eta;\omega) = c(\omega)^{-1} \cdot \mathbb{E}_{p(x,y)}\left[-f(x)\omega(y)\log q_1(x;y,\eta)\right] + \text{const wrt } \eta, \tag{31}$$

where $c(\omega) = \mathbb{E}_{p(y)}\left[\mu(y)\omega(y)\right] > 0$ is again a positive constant which does not affect the optimization.

5.2.2. Parameterized Function $f(x;\theta)$

As previously mentioned, AMCI also allows for amortization over parametrized functions, to account for cases in which multiple possible target functions may be of interest. We can incorporate this by using a *pseudo prior* $p(\theta)$ to generate example parameters during training.

Analogously to $h(y)$, the choice of $p(\theta)$ determines how much importance we assign to different possible functions that we would like to amortize over. Since, in practice, perfect performance is unattainable over the entire space of θ, the choice of $p(\theta)$ is important and it will have an important effect on the performance of the system.

Incorporating $p(\theta)$ is straightforward: we take the expectation of the fixed target function training objective over θ. In this setting, our inference network φ needs to take θ as input when determining the parameters of q_1 and hence we let $q_1(x;y,\theta,\eta) := q_1(x;\varphi(y,\theta;\eta))$. Defining $g(x|y;\theta) := f(x;\theta)\,p(x,y)/\mathbb{E}_{p(x)}\left[f(x;\theta)p(y|x)\right]$, and taking $h(y,\theta) \propto p(y)p(\theta)\mu(y,\theta)$, we get an objective which is analogous to (30):

$$\begin{aligned}\mathcal{J}_1(\eta) &= \mathbb{E}_{h(y,\theta)}\Big[D_{KL}\big(g(x|y;\theta)\,||\,q_1(x;y,\theta,\eta)\big)\Big] \\ &= c^{-1} \cdot \mathbb{E}_{p(x,y)p(\theta)}\left[-f(x;\theta)\log q_1(x;y,\theta,\eta)\right] + \text{const wrt } \eta,\end{aligned} \tag{32}$$

where $c = \mathbb{E}_{p(y)p(\theta)}\left[\mu(y,\theta)\right] > 0$ is again a positive constant that does not affect the optimization.

5.2.3. Efficient Training

The most straightforward way to train the inference network is to form Monte Carlo estimators for the gradients of the loss functions (30) and (32) that directly sample from $p(\theta)p(x)p(y|x)$, namely

$$c\nabla_\eta \mathcal{J}_1(\eta) = \mathbb{E}_{p(x,y)p(\theta)}\left[-f(x;\theta)\nabla_\eta \log q_1(x;y,\theta,\eta)\right] \approx -\frac{1}{B}\sum_{b=1}^{B} f(x_i;\theta_i)\nabla_\eta \log q_1(x_i;y_i,\theta_i,\eta),$$

where $(x_i,y_i,\theta_i) \sim p(\theta)p(x)p(y|x)$. However, if $f(x;\theta)$ and $p(\theta)p(x)$ are poorly matched, i.e. $f(x;\theta)$ is relatively large in regions where the density $p(\theta)p(x)$ is small, this approach can be inefficient. Instead, it will generally be preferable to reformulate the estimator so that samples are generated from $g(\theta,x)p(y|x)$ where $g(\theta,x) \propto p(\theta)p(x)f(x;\theta)$. For example, defining $d = \mathbb{E}_{p(\theta)p(x)}[f(x;\theta)]$ as another constant that does not effect the optimization, we have

$$(c/d)\nabla_\eta \mathcal{J}_1(\eta) = \mathbb{E}_{g(\theta,x)p(y|x)}\left[-\nabla_\eta \log q_1(x;y,\theta,\eta)\right] \approx -\frac{1}{B}\sum_{b=1}^{B} \nabla_\eta \log q_1(\hat{x}_i;\hat{y}_i,\hat{\theta}_i,\eta), \tag{33}$$

where $(\hat{x}_i,\hat{y}_i,\hat{\theta}_i) \sim g(\theta,x)p(y|x)$.

Though $g(\theta,x)$ is itself an intractable distribution, drawing samples from it represents a standard, rather than an amortized, inference problem and so it is much more manageable than the overall training. Namely, as the samples do not depend on the proposal we are learning or the data sets, we can carry out this inference process as a pre-training step that is substantially less costly than the problem of training the inference network itself.

For example, one simple approach is to construct an MCMC sampler targeting $g(\theta,x)$ to generate the required samples for (33), noting that this can be done upfront before training.[9] Another is to

9. Also note here that it will generally be beneficial to randomly permute the order of these samples to reduce autocorrelation effects.

use an importance sampler as follows

$$c\nabla_\eta \mathcal{J}_1(\eta) = \mathbb{E}_{q'(\theta,x)p(y|x)}\left[-\frac{p(\theta)p(x)f(x;\theta)}{q'(\theta,x)}\nabla_\eta \log q_1(x;y,\theta,\eta)\right] \tag{34}$$

$$\approx -\frac{1}{B}\sum_{b=1}^{B}\frac{p(\tilde{\theta}_i)p(\tilde{x}_i)f(\tilde{x}_i;\tilde{\theta}_i)}{q'(\tilde{\theta}_i,\tilde{x}_i)}\nabla_\eta \log q_1(\tilde{x}_i;\tilde{y}_i,\tilde{\theta}_i,\eta) \tag{35}$$

where $(\tilde{x}_i,\tilde{y}_i,\tilde{\theta}_i) \sim q'(\theta,x)p(y|x)$ and $q'(\theta,x)$ is a proposal as close to $g(\theta,x)$ as possible. Again these samples can be generated upfront before training, potentially using an AIS method instead of simple importance sampling (noting no self-normalization is required).

Both these techniques also apply equally well in the case where we do not look to amortize over function parameters. In this case, there is no need to take an expectation over $p(\theta)$ and so we instead look to draw MCMC samples from, or construct an importance sampler for, $g(x) \propto p(x)f(x)$.

5.3. Experiments

AMCI is theoretically able to achieve exact estimators with a finite number of samples as per our general TABI estimator results. In particular, if our amortized proposal families are sufficiently expressive, and our training schemes sufficiently powerful, that we managed to learn artifacts which achieve $\mathcal{J}_1^+(\eta_1^+) = \mathcal{J}_1^-(\eta_1^-) = \mathcal{J}_2(\eta_2) = 0$, then our corresponding final TABI estimators will be exact regardless of the number of samples used.

However, this will rarely be realizable for practical problems, for which learning perfect proposals is not typically realistic, particularly in amortized contexts (Cremer et al., 2018). It is, therefore, necessary to test its empirical performance to assert that gains are possible with imperfect proposals. To this end, we investigate AMCI's performance on two illustrative examples.

Our primary baseline is the SNIS approach implicitly used by most existing inference amortization methods, namely the SNIS estimator with proposal $q_2(x;y)$. Though this effectively represents the previous state–of–the–art in amortized expectation calculation, it turns out to be a very weak baseline. We, therefore, introduce another simple approach one could hypothetically consider using: training separate proposals as per AMCI, but then using this to form a mixture distribution proposal for an SNIS estimator. Namely, we consider using

$$q_m(x;y,\theta) = \frac{1}{2}q_1(x;y,\theta) + \frac{1}{2}q_2(x;y), \quad \text{or}$$
$$q_m(x;y,\theta) = \frac{1}{3}q_1^+(x;y,\theta) + \frac{1}{3}q_1^-(x;y,\theta) + \frac{1}{3}q_2(x;y),$$

depending on whether $f(x)$ can be bounded or not (i.e. whether one of E_1^+ or E_1^- is zero). This is an SNIS proposal that takes into account the needs of both E_1 and E_2. We refer to this method as the amortized *mixture* SNIS estimator and emphasize that it represents a novel amortization approach in its own right. We also note that there is not a simple analogous version of this mixture SNIS for the adaptive importance sampling scenario we previously considered, because our ability to adapt the separate proposals for TAAIS relied on the use of separate estimators.

We also compare AMCI to the theoretically optimal SNIS estimator, i.e. the error bound given by (4). As we will show, AMCI is often able to empirically outperform this bound, thereby giving better performance than *any* approach based on SNIS, whether that approach is amortized or not.

Both our experiments correspond to a setting where $f(x) \geq 0 \, \forall x$, such that we can omit the calculation of E_1^- and learn a single amortized proposal $q_1(x;y,\theta)$ for E_1. We further considered using SNIS with this proposal, which is perhaps the most natural naïve way to adjust an amortized inference approach to use information about $f(x)$. However, this transpired to perform extremely poorly throughout (far worse than $q_2(x;y)$) and so we omit its results from the main paper, giving them in Appendix E. In all experiments, we use the same number of sample from each proposal to

form the estimate (i.e. $N = M = K$). An implementation for AMCI and our associated experiments is available at `http://github.com/talesa/amci`.

5.3.1. Tail Integral Calculation

We start with the conceptually simple problem of calculating tail integrals for Gaussian distributions, namely

$$p(x) = \mathcal{N}(x; 0, \Sigma_1), \quad p(y|x) = \mathcal{N}(y; x, \Sigma_2), \tag{36a}$$

$$f(x; \theta) = \prod_{i=1}^{D} \mathbb{1}_{x_i > \theta_i}, \quad p(\theta) = \textsc{Uniform}(\theta; [0, u_D]^D), \tag{36b}$$

where D is the dimensionality, we set $\Sigma_2 = I$, and Σ_1 is a fixed covariance matrix. This problem was chosen because it permits easy calculation of the ground truth expectations by exploiting analytic simplifications, while remaining numerically challenging for values of θ far away from the mean when we do not use these simplifications.

We performed one and five-dimensional variants of the experiment. For the one-dimensional case we used $u_1 = 5$ and $\Sigma_1 = \Sigma_2 = 1$, while for the five-dimensional case we used $u_5 = 3$, $\Sigma_2 = I$ and the randomly generated matrix

$$\Sigma_1 = \begin{bmatrix} 1.2449 & 0.2068 & 0.1635 & 0.1148 & 0.0604 \\ 0.2068 & 1.2087 & 0.1650 & 0.1158 & 0.0609 \\ 0.1635 & 0.1650 & 1.1665 & 0.1169 & 0.0615 \\ 0.1148 & 0.1158 & 0.1169 & 1.1179 & 0.0620 \\ 0.0604 & 0.0609 & 0.0615 & 0.0620 & 1.0625 \end{bmatrix}.$$

We used normalizing flows (Rezende and Mohamed, 2015) to construct our proposals, providing a flexible and powerful means of representing the target distributions. Different flow architectures were used for the two variants, with both taking an isotropic Gaussian base distribution. For the one–dimensional case, our flow comprised of 10 radial flow layers (Rezende and Mohamed, 2015). Amortization was provided by using a neural network taking in the values of y and θ as input, and returning the parameters defining the flow transformations. This network was comprised of 3 fully connected layers with 1000 hidden units in each layer and relu activation functions.

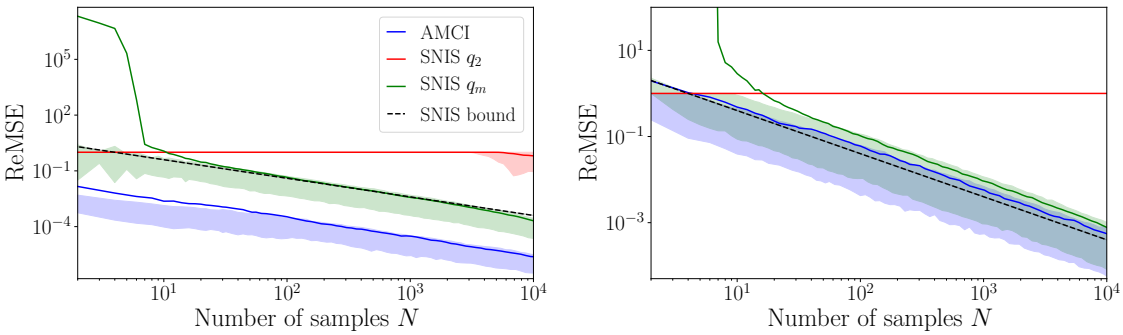
For the more challenging five–dimensional case, we instead used conditional masked autoregressive flows (CMAF) (Papamakarios et al., 2017) with 4 flow layers of 1024 hidden units each. We adapted the implementation from `http://github.com/ikostrikov/pytorch-flows` but we did not find batch normalization helpful and therefore omitted it. Note that CMAF naturally allows for amortization by conditioning on y and θ, such that we did not need an explicit inference network to learn the proposal's parameters in this case.

In both cases, training was done by using importance sampling to generate the values of θ and x as per (34), with

$$q'(\theta, x) = p(\theta) \cdot \textsc{HalfNormal}(x; \theta, \operatorname{diag}(\Sigma_2)).$$

The Adam optimizer (Kingma and Ba, 2015) was adopted for both, with learning rates of 10^{-2} and 10^{-4} respectively. We used a data set generation and mini–batching procedure for learning the inference network introduced by Paige and Wood (2016), these are discussed in Appendix D.

For the one–dimensional variant, the ground truth values of $\mu(y, \theta)$ were determined analytically using $\mu(y, \theta) = \mathbb{E}_{p(x|y)}\left[f(x; \theta)\right] = 1 - \Phi(\theta)$, where $\Phi(\cdot)$ is the standard normal cumulative distribution function. For the five–dimensional variant, they were numerically estimated to a very high degree of accuracy using an SNIS estimator with 10^{10} samples and the proposal $q(x; \theta) = \textsc{HalfNormal}(x; \theta, \operatorname{diag}(\Sigma_2))$.



(a) One-dimensional tail integral

(b) Five-dimensional tail integral

Figure 6: Relative mean squared errors (as per Eq 37) for [left] the one-dimensional and [right] the five-dimensional tail integral example. The solid lines for each estimator indicate the median of $\delta(y,\theta)$ across y and θ, estimated using a common set of 100 samples from $y,\theta \sim p(y)p(\theta)$, with each corresponding $\delta(y,\theta)$ separately estimated by taking the mean of 100 samples of the respective $\hat{\delta}(y,\theta)$. The shading instead shows the estimates from replacing this estimate for $\delta(y,\theta)$ with the 25% and 75% quantiles of these samples of $\hat{\delta}(y,\theta)$ for a given y and θ. Note that solid line (the median of the mean estimates) is at times outside of this shaded region (the median of the quantile estimates) because the former is often dominated by a few large outliers, i.e. the distribution of $\hat{\delta}(y,\theta)|y,\theta$ tends to have a large positive skew. The dashed line shows the median over y and θ of the ReMSE of the optimal SNIS estimator, namely $\textsc{median}_{p(y)p(\theta)}\left[(\mathbb{E}_{p(x|y)}[|f(x;\theta)-\mu(y,\theta)|])^2/N\right]$. We note that the error for SNIS with q_2 proposal is to a large extent flat because there is typically not a single sample in the estimator for which $f(x;\theta)>0$, such that they return $\hat{\mu}(y,\theta)=0$ and hence give $\delta(y,\theta)=1$. The true q_2 error here is probably actually much larger, i.e. $\textsc{median}_{p(y)p(\theta)}(\delta(y,\theta)) \gg 1$, but is difficult to estimate accurately due to the large skew of the estimator $\hat{\delta}(y,\theta)$. In Figure (b) the SNIS q_m line reaches a ReMSE value of 10^{18} at $N=2$; the y-axis limits have been adjusted to allow clear comparison at higher N. This effect is caused by the bias of SNIS: these extremely high errors for SNIS q_m arise when all N samples happen to be drawn from distribution q_1, for further explanation and the full picture see Figure 14 in Appendix E.

As in previous sections, we use the relative mean squared error (ReMSE) as our main basis for comparing performance. However, here the impact of the varying y and θ mean more care is required in detailing how this estimated. Firstly, we formally define the ReMSE as

$$\delta(y,\theta) = \mathbb{E}\left[\hat{\delta}(y,\theta)\middle|\theta,y\right], \tag{37}$$

where $\hat{\delta}(y,\theta)$ is as per (12). We then consider summary statistics across different $\{y,\theta\}$, such as its median when $y,\theta \sim p(y)p(\theta)$.[10] In calculating this, $\delta(y,\theta)$ was separately estimated for each value of y and θ using 100 samples of $\hat{\delta}(y,\theta)$ (i.e. 100 realizations of the estimator).

As shown in Figure 6, AMCI outperformed SNIS in both the one- and five-dimensional cases. For the one-dimensional example, AMCI significantly outperformed all of SNIS q_2, SNIS q_m, and the theoretically optimal SNIS estimator. SNIS q_2, the approach implicitly taken by existing inference amortization methods, typically failed to place even a single sample in the tail of the distribution,

10. Variability in $\delta(y,\theta)$ between different instances of $\{y,\theta\}$ is considered in Figures 15 and 17 in Appendix E.

even for large N. Interestingly, SNIS q_m closely matched the theoretical SNIS bound, suggesting that this amortized proposal is very close to the theoretically optimal one. However, this still constituted significantly worse performance than AMCI—taking about 10^3 more samples to achieve the same relative error—demonstrating the ability of AMCI to outperform the best possible SNIS estimator.

For the five-dimensional example, AMCI again significantly outperformed our main baseline SNIS q_2. Though it still also outperformed SNIS q_m, its advantage was less than in the one-dimensional case, and it did not outperform the SNIS theoretical bound. SNIS q_m itself did not match the bound as closely as in the one-dimensional example either, suggesting that the proposals learned were worse than in the one-dimensional case.

Further comparisons based on using the mean squared error (instead of ReMSE) are given in Appendix E and show qualitatively similar behavior.

5.3.2. Planning Cancer Treatment

To demonstrate how AMCI might be used in a more real-world scenario, we now consider an illustrative example relating to cancer diagnostic decisions. Imagine that an oncologist is trying to decide whether to administer a treatment to a cancer patient. Because the treatment is highly invasive, they only want to administer it if there is a realistic chance of it being successful, i.e. that the tumor shrinks sufficiently to allow a future operation to be carried out. However, they are only able to make noisy observations about the current size of the tumor, and there are various unknown parameters pertaining to its growth, such as the patient's predisposition to the treatment. To aid in the oncologist's decision, the clinic provides a simulator of tumor evolution, a model of the latent factors required for this simulator, and a loss function for administering the treatment given the final tumor size. We wish to construct an amortization of this simulator so that we can quickly and directly predict the expected loss function for administering the treatment from a pair of noisy observations of the tumor size taken at separate points in time. We note that this is a problem for which the target function $f(x)$ does not have any changeable parameters (i.e. $\theta = \emptyset$).

To introduce this problem more precisely, we presume that the size of the tumor is measured at the time of admission $t = 0$ and five days later ($t = 5$), yielding observations c'_0 and c'_5. These are noisy measurements of the true sizes c_0 and c_5. The loss function $\ell(c_{100})$ is based only on the size of the tumor after $t = 100$ days of treatment. The simulator for the development of the tumor takes the form of an ordinary differential equation (ODE) and is based on Hahnfeldt et al. (1999) and Enderling and Chaplain (2014), while the specific experimental setup itself is adapted from an experiment in Rainforth et al. (2018a).

This ODE is defined on two variables, the size of the tumor at time t, c_t, and the corresponding carrying capacity, K_t, where we take $K_0 = 700$. In addition to the initial tumor size c_0, the key parameter of the ODE, and the only one we model as varying across patients, is $\epsilon \in [0, 1]$, a coefficient determining the patient's response to the anti-tumor treatment. The ODE now takes the form

$$\frac{\dot{\mathrm{c}}}{\dot{\mathrm{t}}} = -\lambda c \log\left(\frac{c}{K}\right) - \epsilon c \qquad \frac{\dot{\mathrm{K}}}{\dot{\mathrm{t}}} = \phi c - \psi K c^{2/3} \tag{38}$$

where the values of the parameters—taken as $\phi = 5.85$, $\psi = 0.00873$, and $\lambda = 0.1923$—are based on those recommended in Hahnfeldt et al. (1999). We further assume the statistical model

$$c_0 \sim \textsc{Gamma}(\text{shape} = 25, \text{scale} = 20) \tag{39a}$$

$$\epsilon \sim \textsc{Beta}(5.0, 10.0) \tag{39b}$$

$$c'_t \sim \textsc{Gamma}\left(\frac{c_t^2}{10000}, \frac{c_t}{10000}\right). \tag{39c}$$

To relate the model to our general notation, we have: $x = \{c_0, \epsilon\}, y = \{c'_0, c'_5\}$.

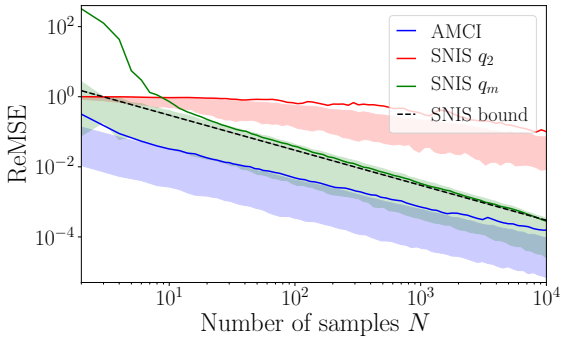


Figure 7: Relative mean squared errors for the cancer example. Conventions as per Figure 6.

Our target function is the loss function for administering the treatment given the final tumor size provided to us by the clinic, which is defined as

$$f(c_0, \epsilon) = \ell(c_{100}(c_0, \epsilon)) = \frac{1 - 2 \times 10^{-8}}{2} \left(\tanh\left(-\frac{c_{100}(c_0, \epsilon) - 300}{150} \right) + 1 \right) + 10^{-8}. \tag{40}$$

where $c_{100}(c_0, \epsilon)$ denotes the outcome of the deterministic process of running an ODE solver on (38) up to time t with a given ϵ and initializations c_0 and $K_0 = 700$.

For this problem, our amortized proposals are based on using parametric distributions: a Gamma distribution for c_0 and a Beta distribution for ϵ. We then train a single–layer perceptron with 500 hidden units to predict the parameters of these distributions as a function of (c_0', c_5'). Since we do not face an overwhelming mismatch between the target function and the prior, unlike in the tail integral example, the training was done by generating the values of c_0 and ϵ from the priors given in (39) as per (32). Training was performed using the Adam optimizer with a learning rate of 10^{-4}, see Appendix D for details on the mini–batching procedure. Ground truth values $\mu(y)$ were estimated numerically to a high degree of accuracy using an SNIS estimator with 10^{10} samples and the proposal set to the prior.

To evaluate the learned proposals we followed the same procedure as for the tail integral example. The results are presented in Figure 7. AMCI again significantly outperformed the literature baseline of SNIS q_2—it took about $N = 10^4$ samples for SNIS q_2 to achieve the level of relative error of AMCI for $N = 2$. AMCI further maintained an advantage over SNIS q_m, which itself again closely matched the optimal SNIS estimator. Further comparisons are given in Appendix E and show qualitatively similar behavior.

5.4. Discussion

In all experiments, AMCI performed better than SNIS with either q_2 or q_m for its proposal. Moreover, we found that AMCI was able to break the theoretical bound on the achievable performance of any SNIS estimator in practice. Interestingly, the mixture SNIS estimator we also introduced proved to be a strong baseline as it closely matched the theoretically optimal SNIS estimator in both experiments. However, such an effective mixture proposal is only possible thanks to learning the multiple inference artifacts we suggest as part of the AMCI framework, while its performance was still generally inferior to AMCI itself.

To assess if the theory discussed in Section 3.5 manifests in practice for AMCI, we revisit our tail integral example, comparing large and small mismatch scenarios. The results, shown in Figure 8, strongly agree with these theoretical findings. Namely, we see that the gains provided by AMCI are much larger in the case where there is large mismatch between $p(x, y)$ and $p(x, y)f(x; \theta)$.

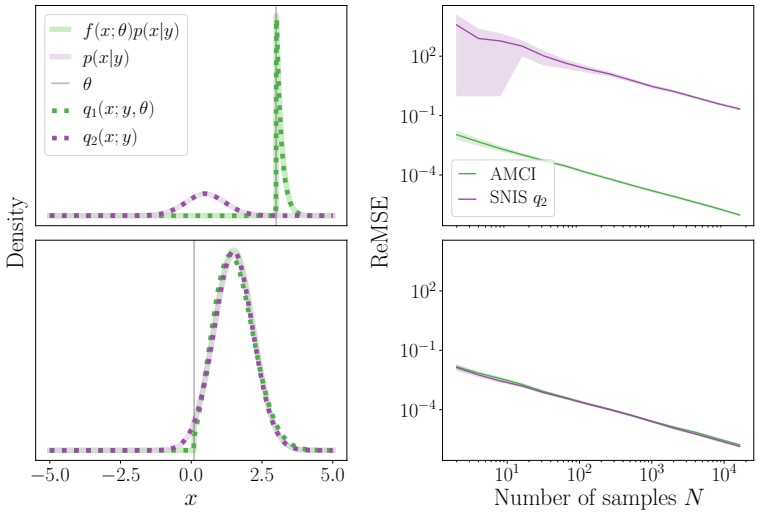


Figure 8: Results for the one-dimensional tail integral model in a setting with large mismatch [top] and low mismatch [bottom], with (y, θ) respectively equal to $(1, 3)$ and $(3, 0.1)$. The left column illustrates the shape of the proposal q_1 and the achievable quality of fit to $f(x;\theta)p(x|y)$, we see that AMCI is able to learn very accurate proposals in both cases. The right column compares the performance of the AMCI and the SNIS estimators where we see that the gain for AMCI is much larger when the mismatch is large. Uncertainty bands in column two are estimated over a 1000 runs.

More generally, as Theorem 1 tells us that the AMCI estimator can achieve an arbitrarily low error for any given target function, while SNIS cannot, we know that its potential gains are larger the more accurate we are able to make our proposals. As such, as advances elsewhere in the field allow us to produce increasingly effective amortized proposals, e.g. through advanced normalizing flow approaches (Durkan et al., 2019; Papamakarios et al., 2019), the larger the potential gains are from using AMCI.

6. A Generalized Form of the TABI Framework

Thus far, we have focused on using our TABI framework alongside some sort of importance sampling approach. This was based largely on the impressive theoretical properties we demonstrated this produces. However, the general concept of breaking up the target expectation and separately estimating each component is far more general than this.

One simple such approach, which can be highly effective in low dimensions, is to use a classical quadrature method to approximate each of the component expectations. Indeed doing this is the natural way to calculate ground truth estimates in low-dimensions and was, for example, the approach we used to generate the ground truth in the banana example in Section 4.5. More generally, we can use the TABI framework with *any* approach for estimating expectations, thereby opening up a wide array of interesting possible extensions.

One particularly interesting class of approaches we can use with the TABI framework are methods that produce marginal likelihood estimates when given an unnormalized target density. This includes conventional importance sampling approaches as already discussed, but also methods such as sequential Monte Carlo (Doucet et al., 2001), nested sampling (Skilling, 2004), and annealed importance sampling (Neal, 2001).

Combining these with the TABI framework is straightforward and follows the approach used by TAAIS in Section 4.3. Namely, we just need to carry out the following simple algorithm:

1. Call a marginal likelihood estimator separately for $\gamma_1^+(x) = p(x,y)f^+(x)$, $\gamma_1^-(x) = p(x,y)f^-(x)$, and $\gamma_2(x) = p(x,y)$ (as per Eq 21), returning $\hat{Z}_1^+$, $\hat{Z}_1^-$, and $\hat{Z}_2$ respectively;
2. Combine the returned marginal likelihood estimates to estimate μ as follows

$$\hat{\mu} := \frac{\hat{Z}_1^+ - \hat{Z}_1^-}{\hat{Z}_2}. \tag{41}$$

Presuming the individual marginal likelihood estimators are consistent, the consistency of (41) follows straightforwardly from Slutsky's Theorem (Slutsky, 1925). Note that there is no need for the marginal likelihood estimators to be called with the same settings and budget, or even for them to correspond to the same type of marginal likelihood estimator at all. We also do not require that these estimators are unbiased, only that they are consistent.

In the rest of this section, we show how this generalized TABI framework can be exploited using two marginal likelihood estimation approaches: nested sampling (Skilling, 2004), and annealed importance sampling (Neal, 2001).

6.1. Target–Aware Nested Sampling (TANS)

Nested sampling (NS) is an algorithm for estimating the marginal likelihood $Z = p(y)$ of a Bayesian model (Skilling, 2004; Evans, 2007; Skilling, 2009; Chopin and Robert, 2010; Feroz et al., 2019). It requires the ability to sample from the prior distribution $p(x)$ and evaluate the likelihood $L(x) = p(y|x)$ for any given x. Alongside returning a consistent (but biased) estimate $\hat{Z}$ of Z, it also returns approximate samples from the posterior $p(x|y)$, which can, in turn, be used to estimate expectations.

The idea behind NS is to approximate the one-dimensional integral

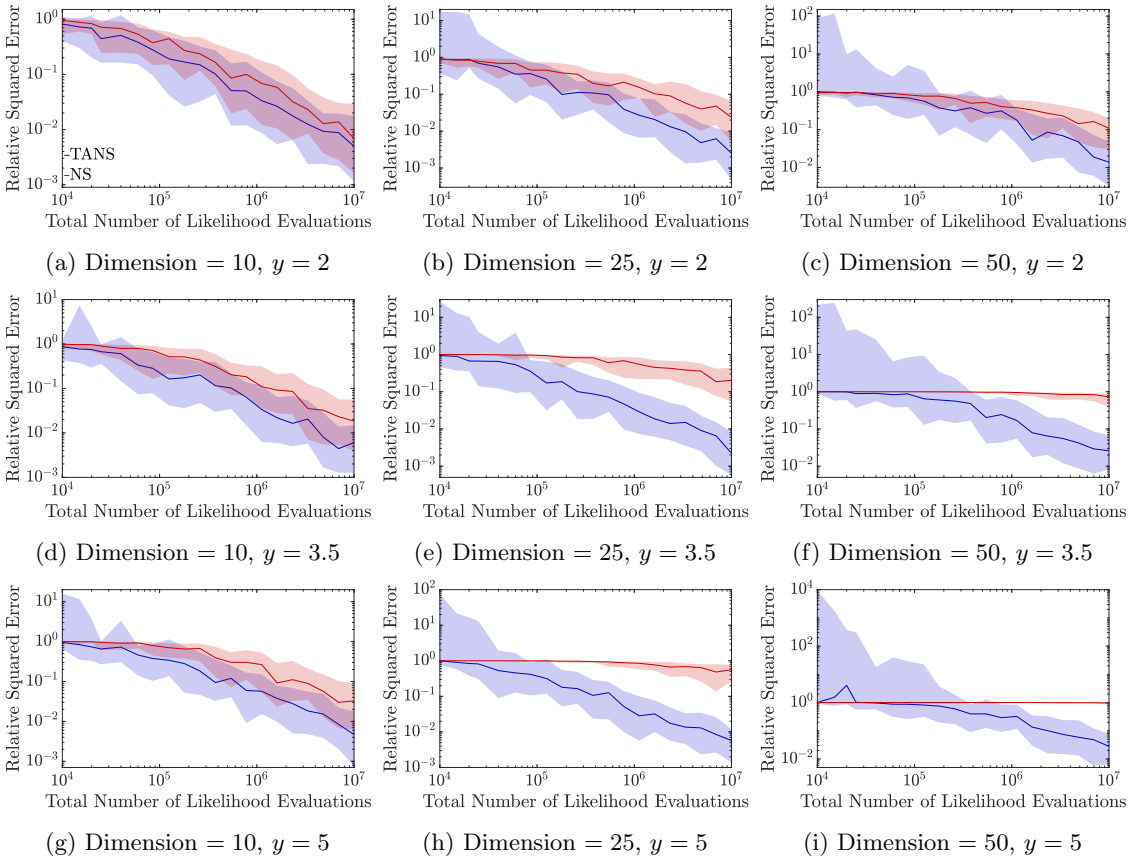
$$Z = \int_0^1 \psi(t)\,\mathrm{d}t \tag{42}$$

where the function ψ is defined via its inverse $\psi^{-1} : t \mapsto \mathbb{P}_{p(x)}(L(x) > t)$ and $\mathbb{P}_{p(x)}(L(x) > t)$ is the probability that a sample from the prior will have a likelihood value greater than t. This identity holds because: a) if $F(t)$ is the cumulative distribution function (cdf) of a random variable X, then $F^{-1}(U)$ has the same distribution as X when $U \sim \mathcal{U}[0,1]$; and b) if $G(t) = 1 - F(t)$ is the complementary cdf, then $G^{-1}(1-t) = F^{-1}(t)$.

Algorithmically, NS approximates the integral (42) using a Riemann sum as follows:

1. Initialize a set of particles $\mathcal{X} := \{x^{(1)}, \ldots, x^{(n)}\}$ by independently sampling each from the prior, i.e. $x^{(1)}, \ldots, x^{(n)} \overset{\text{i.i.d.}}{\sim} p(x)$, along with the marginal likelihood estimate $\hat{Z} \leftarrow 0$.
2. For $i = 1, \ldots, T$:
   (a) Find $L_i = \min_{x \in \mathcal{X}} L(x)$ and set $x_i = x^{(m)}$ where m is the corresponding index of the minimum such that $L(x_i) = L_i$.
   (b) Set $w_i = \exp(-(i-1)/n) - \exp(-i/n)$, and increment $\hat{Z} \leftarrow \hat{Z} + w_i L_i$.
   (c) Sample a new point $x' \sim p(x)$ conditional on $L(x') > L_i$ using an MCMC sampler, and replace $x^{(m)}$ by x' in $\mathcal{X}$.

The resulting quantity $\hat{Z}$ is a biased, but consistent, estimate of Z (Evans, 2007; Skilling, 2009; Chopin and Robert, 2010). Here the bias is due to approximation errors involved in using a Riemann sum approximation and Step 2(b) in the algorithm above; the reader is referred to Skilling (2004)



(a) Dimension $= 10$, $y = 2$ (b) Dimension $= 25$, $y = 2$ (c) Dimension $= 50$, $y = 2$

(d) Dimension $= 10$, $y = 3.5$ (e) Dimension $= 25$, $y = 3.5$ (f) Dimension $= 50$, $y = 3.5$

(g) Dimension $= 10$, $y = 5$ (h) Dimension $= 25$, $y = 5$ (i) Dimension $= 50$, $y = 5$

Figure 9: Convergence plots of TANS and NS on the Gaussian model defined in (24) for different dimensionalities and separations y. Solid lines represent the median across 100 runs, shading the 25% and 75% quantiles. The x-axis shows the computational budget in terms of the number of likelihood evaluations. Unlike in Figure 3, different x-axis values on this plot constitute independent evaluations: NS is not an online estimator and must be rerun when the budget is increased. Note that the apparently strong relative performance of NS for small sample sizes in higher dimensions and separations is because its estimator is heavily skewed and we are looking at its quantiles: here it typically just returns a trivial estimate of effectively zero (giving a relative squared error of 1), while very rarely it will produce a very large estimate with a huge error. As such, its displayed performance somewhat flatters to deceive. TANS on the other hand is over– and under–estimating with relatively similar frequency, such that its results are generally representative.

for a discussion on this. A weighted sample approximation of the posterior follows as a byproduct of the algorithm, allowing us to construct the following estimate

$$\mathbb{E}_{p(x|y)}[f(x)] \approx \hat{\mu}_{\text{NS}} := \frac{\sum_{i=1}^{T} w_i L_i f(x_i)}{\hat{Z}}. \tag{43}$$

As per the general case described earlier, to combine NS with our TABI framework we simply use it to produce marginal likelihood estimates for each of $\gamma_1^+(x)$, $\gamma_1^-(x)$, and $\gamma_2(x)$, then combine the resulting estimates as per (41). This effectively equates to calling NS three times with the

same prior $p(x)$, but respective likelihoods of $L_1^+(x) := p(y|x)f^+(x)$, $L_1^-(x) := p(y|x)f^-(x)$, and $L_2(x) := p(y|x)$. We refer to the resulting approach as *target-aware nested sampling* (TANS).

To investigate the empirical performance of TANS, we revisit the simple Gaussian model setup of Section 4.5.1, comparing to the standard NS approach that does not use information about $f(x)$. In our implementation, Step 2(c) above is conducted by running 20 steps of a Metropolis-Hastings chain initialized at a randomly chosen point in $\mathcal{X} \setminus \{x_i\}$ and targeting the unnormalized density $p(x)\,\mathbb{I}(L(x) > L_i)$. The proposal for this sampler is based on an isotropic Gaussian with fixed variance.[11] We vary the number of particles, n, depending on the allowed computational budget, instead keeping T/n fixed to 250.

As before, we run NS and TANS for $D \in \{10, 25, 50\}$ and $y \in \{2, 3.5, 5\}$. The variance of the proposal was set to I, $0.09\,I$, and $0.01\,I$ for $D = 10$, 25, and 50 respectively (where I is the identity matrix). The results are shown in Figure 9 and Table 2. We see that TANS outperforms NS throughout, with the advantage of TANS becoming larger both as the separation between the posterior distribution and target function is increased, and when the dimensionality increases. Namely, while the performance of TANS was remarkably robust to increasing the separation or dimensionality, these both caused the performance of NS to deteriorate.

6.2. Target–Aware Annealed Importance Sampling (TAAnIS)

Annealed importance sampling (AnIS) (Neal, 2001) is an inference algorithm based on sampling from a series of annealing distributions that act as stepping stones from a known initial distribution $\pi_0(x)$ that we can sample from, to a target distribution $\pi_n(x) = \pi(x)$ that, as usual, is only known up to a normalizing constant, i.e. $\pi_n(x) = \gamma(x)/Z$ where $\gamma(x)$ is known but Z is not. AnIS returns a weighted sample approximation of $\pi_n(x)$ along with an *unbiased* estimate of its normalizing constant Z, both of which are consistent in the limit of independently running the algorithm a large number of times and averaging the outputs. Though, strictly speaking, AnIS is an importance sampler on an extended space, its formulation, algorithmic procedure, and typical behavior are very different from conventional importance sampling. In particular, it leverages MCMC samples and often remains effective for high dimensional problems. It is often especially effective at estimating the normalization constant and has seen substantial use for this purpose in high-dimensional models (Wallach et al., 2009; Salakhutdinov and Larochelle, 2010; Wu et al., 2017).

To introduce AnIS more formally, let the series of intermediate distributions be denoted as $\pi_0, \pi_1, \ldots, \pi_n$. Though in theory these intermediate distributions can take any form, the overwhelmingly most common choice is

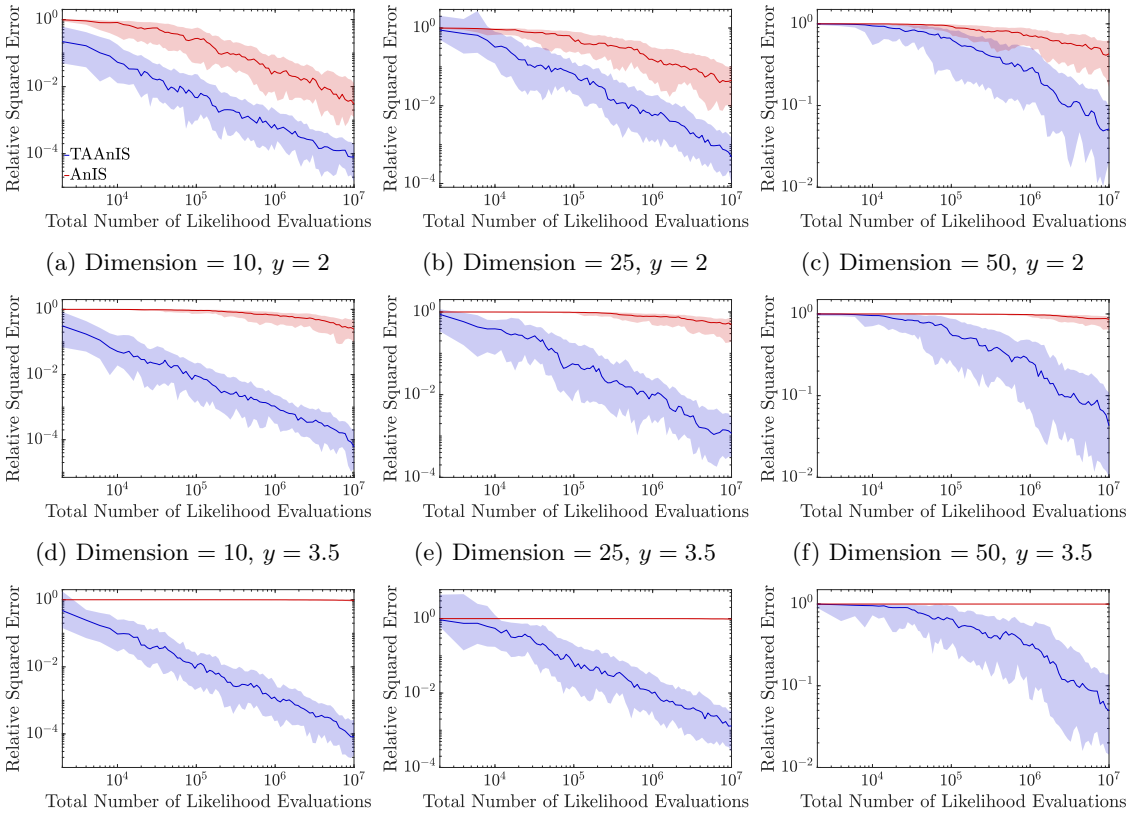
$$\pi_i(x) \propto \lambda_i(x) := \pi_0(x)^{1-\beta_i}\gamma(x)^{\beta_i}$$

where $0 = \beta_0 < \beta_1 < \cdots < \beta_n = 1$. The AnIS algorithm further requires the definition of a series of Markov chain transition kernels $\tau_1(x, x'), \tau_2(x, x'), \ldots, \tau_{n-1}(x, x')$ where $\tau_i(x, x')$ leaves π_i invariant. To generate *one* sample x from $\pi_n(x)$ with a corresponding weight w, AnIS proceeds as follows:

1. Sample $x^{(1)} \sim \pi_0(\cdot)$;
2. For $i = 1 : n-1$, sample $x^{(i+1)} \sim \tau_i(x^{(i)}, \cdot)$;
3. Return the sample $x = x^{(n)}$ with corresponding weight

$$w = \frac{\lambda_1(x^{(1)})\lambda_2(x^{(2)})\cdots\lambda_n(x^{(n)})}{\pi_0(x^{(1)})\lambda_1(x^{(2)})\cdots\lambda_{n-1}(x^{(n)})}. \tag{44}$$

11. We note that using adaptive proposals for the MCMC steps here can induce asymptotic biases in TANS because, even if it produces samples being generated from the target distribution, this adaptation can induce an asymptotic bias in the marginal likelihood estimate.



(a) Dimension = 10, $y = 2$ (b) Dimension = 25, $y = 2$ (c) Dimension = 50, $y = 2$

(d) Dimension = 10, $y = 3.5$ (e) Dimension = 25, $y = 3.5$ (f) Dimension = 50, $y = 3.5$

(g) Dimension = 10, $y = 5$ (h) Dimension = 25, $y = 5$ (i) Dimension = 50, $y = 5$

Figure 10: Convergence plots of TAAnIS and AnIS on the Gaussian model defined in (24) for different dimensionalities and separations y. Solid lines represent the median across 100 runs, shading the 25% and 75% quantiles. Conventions as per Figure 9.

By repeating this procedure N times, one obtains N independent samples $x_1, \ldots, x_N$ with corresponding weights $w_1, \ldots, w_N$, from which the following estimates can be formed:

$$Z \approx \hat{Z}_{\text{AnIS}} := \frac{1}{N} \sum_{i=1}^{N} w_i \tag{45}$$

$$\mathbb{E}_{\pi_n(x)}[f(x)] \approx \hat{\mu}_{\text{AnIS}} := \frac{\sum_{i=1}^{N} w_i f(x_i)}{\sum_{i=1}^{N} w_i}. \tag{46}$$

The consistency of these estimators, and unbiasedness of $\hat{Z}$, can be shown by demonstrating that AnIS implicitly corresponds to a properly–weighted importance sampling procedure on an extended space with marginal distribution $\pi_n(x)$ (Neal, 2001).

In the context of Bayesian inference, $\pi_0(x)$ is typically taken as the prior $p(x)$ and $\gamma(x)$ as the joint $p(x, y)$ (such that $\pi_n(x) = p(x|y)$). As with NS, combining AnIS with our TABI framework is straightforward: we simply call it three times with $\pi_0(x) = p(x)$ in each instance and $\gamma(x)$ respectively set to $\gamma_1^+(x)$, $\gamma_1^-(x)$, and $\gamma_2(x)$. The resulting marginal likelihood estimators can then be combined in the standard TABI way, namely (41). We refer to the resulting approach as *target-aware annealed importance sampling* (TAAnIS).

y	Dimension	NS	TANS	AnIS	TAAnIS
2	10	-5.36 ± 0.24	-5.76 ± 0.23	-5.90 ± 0.21	$\mathbf{-10.03 \pm 0.20}$
	25	-4.20 ± 0.22	-6.50 ± 0.25	-3.59 ± 0.22	$\mathbf{-8.07 \pm 0.24}$
	50	-2.65 ± 0.20	$\mathbf{-4.74 \pm 0.22}$	-1.23 ± 0.18	-3.26 ± 0.17
3.5	10	-4.69 ± 0.26	-5.68 ± 0.23	-1.57 ± 0.16	$\mathbf{-10.20 \pm 0.26}$
	25	-1.85 ± 0.20	-6.42 ± 0.23	-1.08 ± 0.18	$\mathbf{-7.28 \pm 0.24}$
	50	-0.75 ± 0.17	$\mathbf{-3.94 \pm 0.19}$	-0.62 ± 0.14	-3.43 ± 0.23
5	10	-4.11 ± 0.24	-5.82 ± 0.24	-0.17 ± 0.07	$\mathbf{-9.69 \pm 0.22}$
	25	-0.85 ± 0.18	-5.88 ± 0.25	-0.01 ± 0.09	$\mathbf{-7.21 \pm 0.21}$
	50	-0.19 ± 0.14	$\mathbf{-3.98 \pm 0.20}$	-0.03 ± 0.04	-3.29 ± 0.23

Table 2: Comparison of final results for Gaussian example when allowing a budget of 10^7 total number of likelihood evaluations as per Figure 9 and Figure 10. Values shown are the mean and standard error of the log relative squared error across 100 runs. As per Table 1, the best result(s) on each problem are shown in bold.

As before, our experimental set-up to test TAAnIS follows the Gaussian example described in Section 4.5.1, again taking $D \in \{10, 25, 50\}$ and $y \in \{2, 3.5, 5\}$. We use $n = 200$ intermediate distributions and each τ_i is a 5-step Metropolis-Hastings chain targeting the unnormalized density λ_i, the proposal for which is an isotropic Gaussian centered on the current point with covariances of $0.1225I$, $0.04\,I$, and $0.01\,I$ for $D = 10$, 25, and 50 respectively.[12]

The results are shown in Figure 10 and Table 2. We see that TAAnIS consistently outperforms AnIS across all values of D and y. Indeed, even for a small separation $y = 2$, TAAnIS converges faster with better final estimates for the allowed full budget of 10^7 likelihood evaluations. The advantage of TAAnIS is further apparent as the separation y increases. As shown in Table 2, TAAnIS was also substantially more effective than TANS in lower dimensions, but interestingly slightly less effective for $D = 50$. It should be noted, however, that some of these differences between TAAnIS and TANS may be down to changes in the MCMC proposals used, as these were not carefully optimized. Though neither TAAnIS nor TANS performed as well as our TAAIS approach in Section 4.5.1 (noting that the total budget allowed for each was the same), this is perhaps not surprising given that the proposal family for TAAIS was carefully setup to match the target (namely both were Gaussians with diagonal covariance structures). By contrast, TAAnIS and TANS were used as more generic estimation approaches and were still able to provide highly effective performance.

Given that AnIS is known to often still be effective in high dimensions, we further investigated whether TAAnIS also retains this property. Namely, we tested the performance of AnIS and TAAnIS in a high-dimensional variant ($D = 500$) of the Gaussian example with large separation ($y = 5$). Now allowing a budget of 10^9 likelihood evaluations, we ran AnIS and TAAnIS with $n = 10^4$ intermediate distributions and with each τ_i given by a 100-step Metropolis-Hastings chain, again with an isotropic Gaussian proposal, this time with covariance $0.0016\,I$. The results are shown in Figure 11. Remarkably, TAAnIS still accurately estimated the expectation. It is interesting to note that, while the expectation being estimated in this case is approximately 3.4×10^{-88}, the best final estimate returned by AnIS across the 10 runs was approximately 10^{-107}, representing an estimate that was still over 15 orders of magnitude away from the truth.

12. As with NS, care should be taken here to not use an adaptive MCMC proposal as this asymptotically biases the marginal likelihood estimator even if the samples produced are still consistent.

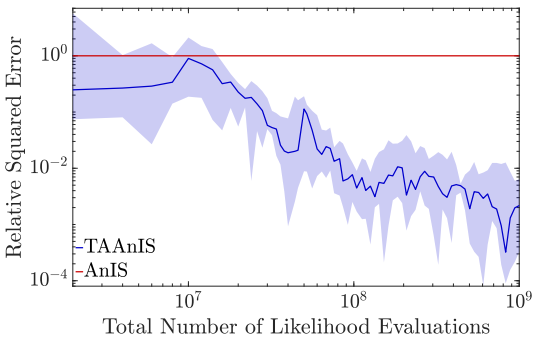


Figure 11: Convergence of AnIS and TAAnIS on the Gaussian model defined in (24) with dimension $D = 500$ and separation $y = 5$. Solid lines represents the median across 10 runs, shading the 25% and 75% quantiles (note this is imperceptibly small for AnIS).

7. Conclusions

We have presented TABI, a framework for performing *target–aware Bayesian inference*. TABI directly targets posterior expectations by breaking them up into three component expectations, separately estimating each using a tailored estimator, and then recombining them to estimate the original expectation. This can offer substantial benefits over conventional estimation approaches because each component expectation can be estimated more accurately when considered in isolation. Notably, it is capable of producing finite sample estimators with arbitrarily low error, thereby providing a mechanism to outperform the theoretically optimal conventional Monte Carlo estimator.

We have demonstrated that the TABI framework can be particularly effective when combined with adaptive or amortized inference procedures, leading to estimators which can not only theoretically, but also practically, outperform the theoretically optimal estimators of conventional approaches, even when those approaches are not themselves in practice capable of achieving performance anywhere near their theoretical optimum. In some cases, we were even able to empirically demonstrate faster–than–Monte–Carlo convergence rates for adaptive TABI estimators.

Another substantial advantage of our TABI framework is that it makes it far *simpler* to construct inference algorithms in a manner that makes them target–aware. For example, whereas constructing adaptive SNIS schemes that incorporate information about the target function is extremely challenging due to the double intractability of the corresponding optimal proposal, the unnormalized densities of the optimal proposals in the TABI framework can be directly evaluated and thus straightforwardly setup as the objectives for adaptive or amortized samplers. As such, TABI is often able to conduct effective target–aware inference in scenarios where conventional approaches might struggle to exploit target information at all. In particular, we have shown how TABI is able to convert *any* marginal likelihood estimator into a target-aware inference algorithm.

Acknowledgments

We would like to thank Yee Whye Teh, Stefan Webb, and Tuan Anh Le for providing helpful discussions. TR's research leading to these results has received funding from a Christ Church Oxford Junior Research Fellowship; the European Research Council under the European Unions Seventh Framework Programme (FP7/20072013) / ERC grant agreement no. 617071; from EPSRC under grant EP/P026753/1; and from Tencent AI Labs. AG is supported by the UK EPSRC CDT in Autonomous Intelligent Machines and Systems. FW is supported by DARPA D3M, under Cooperative Agreement FA8750-17-2-0093, Intel under its LBNL NERSC Big Data Center, and an NSERC Discovery grant. SZ is supported by Aker Scholarship.

Appendix A. Proofs

A.1. Derivation of Equation 5

Under the assumption that $f(x) \geq 0 \, \forall x$, we can derive (5) as follows

$$\begin{aligned}
\mathbb{E}_{p(x|y)}[|f(x) - \mu|] &= \mathbb{E}_{p(x|y)}[(f(x) - \mu)\mathbb{I}(f(x) \geq \mu) - (f(x) - \mu)\mathbb{I}(f(x) < \mu)] \\
&= \mathbb{E}_{p(x|y)}[(f(x) - \mu)\left(1 - \mathbb{I}(f(x) < \mu)\right) - (f(x) - \mu)\mathbb{I}(f(x) < \mu)] \\
&= \underbrace{\mathbb{E}_{p(x|y)}[f(x) - \mu]}_{=0} - 2\,\mathbb{E}_{p(x|y)}[(f(x) - \mu)\mathbb{I}(f(x) < \mu)] \\
&= 2\mu\,\mathbb{E}_{p(x|y)}[\mathbb{I}(f(x) < \mu)] - 2\,\mathbb{E}_{p(x|y)}[f(x)\mathbb{I}(f(x) < \mu)].
\end{aligned}$$

Now given that $f(x) \geq 0 \, \forall x$, this second term is non-negative so we have

$$\begin{aligned}
&\leq 2\mu\,\mathbb{E}_{p(x|y)}[\mathbb{I}(f(x) < \mu)] \\
&\leq 2\mu
\end{aligned}$$

because the expectation must be between 0 and 1. Substituting the above into the left hand side of (5) subsequently gives the desired result, remembering that $\mathbb{E}_{p(x|y)}[|f(x) - \mu|]$ must itself be positive. Note that in the case where $f(x)$ is a Dirac delta function, both of these inequalities become tight, such that the result becomes an equality.

In the case, $f(x) \leq 0 \, \forall x$, we can straightforwardly show the equivalent bound

$$\mathbb{E}_{p(x|y)}[|f(x) - \mu|] \leq -2\mu,$$

which again leads to (5) by simple substitution.

A.2. Proof of Theorem 2

Proof Starting with (13) and using Taylor's Theorem on $1/\left(E_2 + \sigma_2\xi_2\right)$ about $1/E_2$ gives

$$\begin{aligned}
\hat{\mu} &= \frac{E_1 + \sigma_1\xi_1}{E_2 + \sigma_2\xi_2} \\
&= (E_1 + \sigma_1\xi_1)\left(\frac{1}{E_2} - \frac{\sigma_2\xi_2}{E_2^2} + \mathcal{O}(\epsilon)\right) \\
&= \mu + \frac{\sigma_1\xi_1}{E_2} - \frac{E_1\sigma_2\xi_2}{E_2^2} - \frac{\sigma_1\sigma_2\xi_1\xi_2}{E_2^2} + \mathcal{O}(\epsilon)
\end{aligned}$$

where $\mathcal{O}(\epsilon)$ represents asymptotically dominated terms. We can further drop the $\sigma_1\sigma_2\xi_1\xi_2/E_2^2$ term as this will be of order $\mathcal{O}(1/\sqrt{MN})$ and will thus be asymptotically dominated, giving

$$\hat{\mu} = \mu + \frac{\sigma_1\xi_1}{E_2} - \frac{E_1\sigma_2\xi_2}{E_2^2} + \mathcal{O}(\epsilon). \tag{47}$$

To calculate the MSE of $\hat{\mu}$, we start with the standard bias–variance decomposition

$$\mathbb{E}\left[(\hat{\mu}-\mu)^2\right] = \mathrm{Var}\left[\hat{\mu}\right] + \left(\mathbb{E}\left[\hat{\mu}-\mu\right]\right)^2.$$

For the variance, we use the relationship $\mathrm{Var}[X-Y] = \mathrm{Var}[X] + \mathrm{Var}[Y] - 2\,\mathrm{Cov}[X,Y]$, yielding

$$\mathrm{Var}[\hat{\mu}] = \mathrm{Var}\left[\frac{\sigma_1\xi_1}{E_2}\right] + \mathrm{Var}\left[\frac{E_1\sigma_2\xi_2}{E_2^2}\right] - 2\,\mathrm{Cov}\left[\frac{\sigma_1\xi_1}{E_2}, \frac{E_1\sigma_2\xi_2}{E_2^2}\right] + \mathcal{O}(\epsilon). \tag{48}$$

Now noting from the central limit theorem $\mathrm{Var}[\xi_1] \to 1$, $\mathrm{Var}[\xi_2] \to 1$, and (using Slutsky's theorem) $\mathrm{Cov}[\xi_1,\xi_2] \to \mathrm{Corr}\left[\xi_1,\xi_2\right]$, we have

$$\begin{aligned}
&= \frac{\sigma_1^2}{E_2^2} + \frac{E_1^2\sigma_2^2}{E_2^4} - 2\frac{E_1\sigma_1\sigma_2}{E_2^3}\,\mathrm{Corr}[\xi_1,\xi_2] + \mathcal{O}(\epsilon) \\
&= \frac{1}{E_2^2}\left(\sigma_1^2 + \sigma_2^2\mu^2 - 2\mu\sigma_1\sigma_2\mathrm{Corr}\left[\xi_1,\xi_2\right]\right) + \mathcal{O}(\epsilon) \\
&= \frac{\sigma_2^2\mu^2}{E_2^2}\left((\kappa - \mathrm{Corr}[\xi_1,\xi_2])^2 + 1 - \mathrm{Corr}[\xi_1,\xi_2]^2\right) + \mathcal{O}(\epsilon)
\end{aligned} \tag{49}$$

where we have used $\kappa = \sigma_1/(\mu\sigma_2)$. The bias squared term, on the other hand, is asymptotically dominated as taking the expectation of (47) yields $\mathbb{E}\left[\hat{\mu}-\mu\right] = \mathcal{O}(\epsilon)$ because $\mathbb{E}[\xi_1] = \mathbb{E}[\xi_2] = 0$.[13] We thus have $\mathbb{E}[(\hat{\mu}-\mu)^2] \to \mathrm{Var}[\hat{\mu}]$, such that (49) gives us the desired asymptotic result. ■

A.3. Proof of Theorem 4

Proof Let $\delta_1^+ := \hat{Z}_1^+ - E_1^+$, $\delta_1^- := \hat{Z}_1^- - E_1^-$, and $\delta_2 = \hat{Z}_2 - E_2$. Applying Taylor's Theorem on $\hat{\mu}_{\mathrm{TAAIS}}$ in a manner analogous to in the proof of Theorem 2 yields

$$\hat{\mu}_{\mathrm{TAAIS}} = \left(E_1^+ - E_1^- + \delta_1^+ - \delta_1^-\right)\left(\frac{1}{E_2} - \frac{\delta_2}{E_2^2} + \frac{\delta_2^2}{E_2^3}\right) + \mathcal{O}(\epsilon).$$

As in the previous proof, we next use the standard bias–variance decomposition of the MSE:

$$\mathbb{E}\left[(\hat{\mu}_{\mathrm{TAAIS}}-\mu)^2\right] = \mathrm{Var}\left[\hat{\mu}_{\mathrm{TAAIS}}\right] + \left(\mathbb{E}\left[(\hat{\mu}_{\mathrm{TAAIS}}-\mu)\right]\right)^2$$

Starting with the bias component and noting that each of δ_1^+, δ_1^-, and δ_2 are independent and have an expectation of zero, we get

$$\begin{aligned}
\mathbb{E}\left[\hat{\mu}_{\mathrm{TAAIS}}-\mu\right] &= \left(E_1^+ - E_1^-\right)\left(\mathbb{E}\left[\frac{\delta_2^2}{E_2^3}\right] - \underbrace{\mathbb{E}\left[\frac{\delta_2}{E_2^2}\right]}_{=0}\right) + \underbrace{\mathbb{E}\left[\delta_1^+ - \delta_1^-\right]}_{=0}\mathbb{E}\left[\frac{1}{E_2} - \frac{\delta_2}{E_2^2} + \frac{\delta_2^2}{E_2^3}\right] + \mathcal{O}(\epsilon) \\
&= \frac{E_1^+ - E_1^-}{E_2^3}\mathbb{E}[\delta_2^2] + \mathcal{O}(\epsilon) \\
&= \frac{\mu}{E_2^2}\mathrm{Var}[\delta_2] + \mathcal{O}(\epsilon).
\end{aligned} \tag{50}$$

13. This is demonstrated more formally in the proof of Theorem 4.

For the variance, by noting that any terms containing products of δs will by dominated and again exploiting independences, we instead have

$$\begin{aligned}\mathrm{Var}\left[\hat{\mu}_{\mathrm{TAAIS}}\right] &= \mathrm{Var}\left[\frac{\delta_1^+ - \delta_1^-}{E_2}\right] + \mathrm{Var}\left[-\frac{\mu\delta_2}{E_2}\right] \\ &= \frac{1}{E_2^2}\mathrm{Var}\left[\delta_1^+\right] + \frac{1}{E_2^2}\mathrm{Var}\left[\delta_1^-\right] + \frac{\mu^2}{E_2^2}\mathrm{Var}\left[\delta_2\right] + \mathcal{O}(\epsilon). \end{aligned} \tag{51}$$

Each of these terms has an equivalent form, so taking the first by way of example we have

$$\mathrm{Var}\left[\delta_1^+\right] = \mathrm{Var}\left[\hat{Z}_1^+\right] = \frac{1}{N^2}\sum_{n=1}^{N}\mathrm{Var}\left[w_{1,n}^+\right] + \underbrace{\frac{1}{N^2}\sum_{n=1}^{N}\sum_{n'=1,n'\neq n}^{N}\mathbb{E}\left[\left(w_{1,n}^+ - E_1^+\right)\left(w_{1,n'}^+ - E_1^+\right)\right]}_{=0}$$

where the second term being zero comes from the fact that, even when we are doing adaptive sampling, $w_{1,n}^+$ and $w_{1,n'}^+$ are independent for $n \neq n'$. Now by invoking our bound on $\mathrm{Var}\left[w_{1,n}^+\right]$ from the theorem statement, we get

$$\mathrm{Var}\left[\delta_1^+\right] = \frac{1}{N^2}\sum_{n=1}^{N}\mathrm{Var}\left[w_{1,n}^+\right] \leq \frac{s_1^+}{N^2}\sum_{n=1}^{N}\frac{1}{n^a} = \frac{s_1^+}{N^2}H_a[N]$$

where $H_a[N]$ is the N^{th} generalized harmonic number of order a. For $a = 1$, $H_1[N] \to \log(N) + \eta$ where $\eta \approx 0.577$ is the Euler-Mascheroni constant (Choi and Srivastava, 2011). For $a > 1$, $H_a[N] \to \zeta(a)$ where ζ is the Riemann-zeta function (where, for $a > 1$, $1 < \zeta(a) < \infty$ and $\zeta(a)$ monotonically decreases with a). For $a < 1$, we can apply the Euler-Maclaurin formula to give

$$\begin{aligned} H_a[N] &= \int_{n=1}^{n=N} n^{-a}dn + \frac{N^{-a}+1}{2} + \mathcal{O}(\epsilon) \\ &= \frac{N^{1-a}-1}{1-a} + \frac{N^{-a}+1}{2} + \mathcal{O}(\epsilon) \\ &\to \frac{N^{1-a}-1}{1-a} + \frac{1}{2} \end{aligned}$$

for large N. Given that we are further multiplying this term by $1/N^2$, we can further ignore all terms that do not increase with N to give $H_a[N]/N^2 \to \frac{1}{N^2}\frac{N^{1-a}}{1-a}$. We thus see that h represents the required asymptotic expressions for $H_a[N]$.

Now invoking analogous arguments for δ_1^- and δ_2 and substituting these bounds into (50) and (51) respectively yields

$$\begin{aligned}\left(\mathbb{E}\left[\hat{\mu}_{\mathrm{TAAIS}} - \mu\right]\right)^2 &\leq \left(\frac{\mu s_2}{M^2 E_2^2}\right)^2 \\ \mathrm{Var}\left[\hat{\mu}_{\mathrm{TAAIS}}\right] &\leq \frac{1}{E_2^2}\left(\frac{s_1^+}{N^2}h(a,N) + \frac{s_1^-}{K^2}h(b,K) + \frac{\mu^2 s_2}{M^2}h(c,M)\right) + \mathcal{O}(\epsilon).\end{aligned}$$

To complete the proof we now simply observe that the bias squared component of the MSE is dominated, such that it can be absorbed into $\mathcal{O}(\epsilon)$, while the variance bound above is equivalent to the MSE bound quoted in the theorem. ■

Appendix B. Reusing Samples

The TABI estimator in (11) requires taking $T = N + K + M$ samples, but only N, K, or M are used to evaluate each of the individual estimators. Given that, in practice, we do not have access to the perfectly optimal proposals, it can sometimes be more efficient to reuse samples in the calculation of multiple components of the expectation, particularly if the target function is cheap to evaluate relative to the proposal. Care is required though to ensure that this is only done when a proposal remains valid (i.e. has finite variance) for the different expectations.

We will focus on the case where $f(x) \geq 0 \; \forall x$ such that we can use a single proposal for the numerator (i.e. $K = 0$, $E_1^+ = E_1$, etc, giving a final estimator of the form in Eq 8). Under this assumption, we can use the following estimator

$$\mu \approx \hat{\mu}_{\text{RS}} := \frac{\alpha \hat{E}_1(q_1) + (1-\alpha)\hat{E}_1(q_2)}{\beta \hat{E}_2(q_1) + (1-\beta)\hat{E}_2(q_2)} \tag{52}$$

where $\hat{E}_i(q_j)$ indicates the estimate for E_i using the samples from q_j. The level of interpolation is set by parameters α, β which can vary between 0 and 1.

For cases where $f(x)$ can instead be both positive and negative, and we also want to recycle samples from q_1^+ for E_1^- and/or q_1^- for E_1^+, we should do this using the approach introduced in Appendix C as this is more efficient than a naive recycling. If desired, this can then also straightforwardly be combined with the estimator above to recycle samples between E_1 and E_2 by replacing $\hat{E}_1(q_1)$ with $\hat{E}_1^{\text{MIS}}$ and constructing $\hat{E}_2(q_1)$ in an analogous manner to $\hat{E}_1^{\text{MIS}}$, taking $f(x) = 1 \, \forall x$.

If we had direct access to the optimal proposals, it would naturally be preferable to set $\alpha = 1$ and $\beta = 0$ in (52), leading to a zero-variance estimator. However, for imperfect proposals, the optimal values vary slightly from this as we now show. Note that while it is possible to set $\beta > 0$ for negligible extra computational cost as $\hat{E}_2(q_1)$ depends only on weights needed for calculating $\hat{E}_1(q_1)$, setting $\alpha < 1$ requires additional evaluations of the target function and so will likely only be beneficial when this is cheap relative to sampling from or evaluating the proposal.

We first consider the empirical effect of reusing samples. For this, we extend the Gaussian tail integral AMCI experiment from Figure 8 which considers a case where $p(x|y)$ and $p(x|y)f(x;\theta)$ are closely matched and case where they are not. Here q_1 is not a valid proposal for E_2 so we set $\beta = 0$, but q_2 is a valid for E_1 so we consider varying α, the results for which are given in Figure 12. We see that in the well–matched case, the optimum α is less than 1 and provides some, relatively modest, gains.

To delve deeper into this, we now derive the optimal values of α and β in terms of minimizing the mean squared error (MSE) of the estimator in (52). Analogously to in Theorem 2, we introduce error terms

$$\xi_{ij} = \frac{\hat{E}_i(q_j) - E_i}{\sigma_{ij}}, \quad \text{where} \quad \sigma_{ij}^2 = \text{Var}[\hat{E}_i(q_j)],$$

such that

$$\hat{\mu}_{\text{RS}} = \frac{E_1 + \alpha\sigma_{11}\xi_{11} + (1-\alpha)\sigma_{12}\xi_{12}}{E_2 + \beta\sigma_{21}\xi_{21} + (1-\beta)\sigma_{22}\xi_{22}}, \tag{53}$$

and ξ_{ij} are again asymptotically distributed according to a standard normal (assuming the σ_{ij} are finite). Note that ξ_{ij} and $\xi_{\ell m}$ are independent if and only if $j \neq m$. Applying Taylor's Theorem in an analogous manner to the proof of Theorem 2 now gives

$$\hat{\mu}_{\text{RS}} = \mu + \frac{1}{E_2}\left(\alpha\sigma_{11}\xi_{11} + (1-\alpha)\sigma_{12}\xi_{12}\right) - \frac{\mu}{E_2}\left(\beta\sigma_{21}\xi_{21} + (1-\beta)\sigma_{22}\xi_{22}\right) + \mathcal{O}(\epsilon)$$

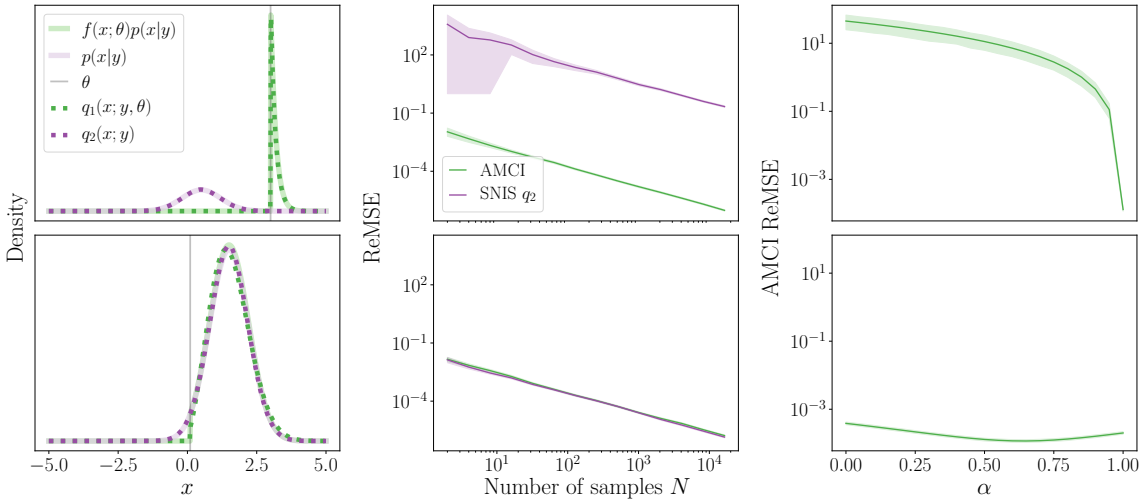


Figure 12: Extension of Figure 8 showing the effects of reusing samples by varying the parameter α in (52) ($\beta = 0$, number of samples is fixed to $N = M = 64$), where we see that this sample re-usage (i.e. choosing an $\alpha < 1$) provides small gains for the low mismatch case [bottom], but no gains for the considered values of α in the high mismatch case [top].

where, as usual, $\mathcal{O}(\epsilon)$ represents asymptotically dominated terms. As before, the MSE will straightforwardly be asymptotically dominated by the variance. Further exploiting the aforementioned independences between certain ξ_{ij} yields

$$\mathbb{E}\left[(\hat{\mu}_{\text{RS}} - \mu)^2\right] = \frac{1}{E_2^2}\left(\text{Var}\left[\alpha\sigma_{11}\xi_{11} - \beta\mu\sigma_{21}\xi_{21}\right] + \text{Var}\left[(1-\alpha)\sigma_{12}\xi_{12} - (1-\beta)\mu\sigma_{22}\xi_{22}\right]\right)$$

and making use of the relationships $\text{Var}[X - Y] = \text{Var}[X] + \text{Var}[Y] - 2\text{Cov}[X, Y]$, $\text{Var}[\xi_{ij}] \to 1$, and $\text{Cov}[\xi_{ij}, \xi_{\ell m}] \to \text{Corr}[\xi_{ij}, \xi_{\ell m}]$ as before, we get

$$\begin{aligned} &= \frac{1}{E_2^2}\Big(\alpha^2\sigma_{11}^2 + \beta^2\mu^2\sigma_{21}^2 + (1-\alpha)^2\sigma_{12}^2 + (1-\beta)^2\mu^2\sigma_{22}^2 \\ &\qquad - 2\alpha\beta\mu\sigma_{11}\sigma_{21}\text{Corr}[\xi_{11}, \xi_{21}] - 2(1-\alpha)(1-\beta)\mu\sigma_{12}\sigma_{22}\text{Corr}[\xi_{12}, \xi_{22}]\Big). \end{aligned}$$

The optimal values of α and β can now be found by solving the set of linear simultaneous equations that result from setting the derivatives of this with respect to both α and β to zero. Doing this with a symbolic solver leads to the following expressions where we adopt the shorthands $C_1 = \text{Corr}[\xi_{11}, \xi_{21}]$ and $C_2 = \text{Corr}[\xi_{12}, \xi_{22}]$

$$\alpha^* = \frac{\sigma_{12}^2\left(\sigma_{21}^2 + \sigma_{22}^2\right) + C_1\mu\sigma_{11}\sigma_{21}\sigma_{22}^2 - C_2\mu\sigma_{12}\sigma_{22}\sigma_{21}^2 - C_2^2\sigma_{12}^2\sigma_{22}^2 - C_1C_2\sigma_{11}\sigma_{12}\sigma_{21}\sigma_{22}}{\left(\sigma_{11}^2 + \sigma_{12}^2\right)\left(\sigma_{21}^2 + \sigma_{22}^2\right) - \left(C_1\sigma_{11}\sigma_{21} + C_2\sigma_{12}\sigma_{22}\right)^2},$$

$$\beta^* = \frac{\sigma_{22}^2\left(\sigma_{11}^2 + \sigma_{12}^2\right) + (C_1/\mu)\sigma_{11}\sigma_{21}\sigma_{12}^2 - (C_2/\mu)\sigma_{12}\sigma_{22}\sigma_{11}^2 - C_2^2\sigma_{12}^2\sigma_{22}^2 - C_1C_2\sigma_{11}\sigma_{12}\sigma_{21}\sigma_{22}}{\left(\sigma_{11}^2 + \sigma_{12}^2\right)\left(\sigma_{21}^2 + \sigma_{22}^2\right) - \left(C_1\sigma_{11}\sigma_{21} + C_2\sigma_{12}\sigma_{22}\right)^2}.$$

In principle, each of these terms can be estimated from existing samples, thereby providing a mechanism for automatically setting α and β, potentially even in an adaptive manner.

If we assume that $\text{Corr}[\xi_{11}, \xi_{21}] = \text{Corr}[\xi_{12}, \xi_{22}] = 0$ then these optimal forms simplify to

$$\alpha^* = \frac{\sigma_{12}^2}{\sigma_{11}^2 + \sigma_{12}^2}, \qquad \beta^* = \frac{\sigma_{22}^2}{\sigma_{21}^2 + \sigma_{22}^2}.$$

This is now a somewhat intuitive result as it corresponds to the α and β which optimize the estimation of E_1 and E_2 when considered in isolation. Note that in the case where we have the optimal TABI proposals, then $\hat{E}_1(q_1) = \hat{E}_2(q_2) = 0$ such that we find $\alpha^* = 1$ and $\beta^* = 0$. At the other extreme, if our proposals are identical, as per the SNIS setting, then we get $\alpha^* = \beta^* = 0.5$.

Appendix C. An Alternative Variant of the TABI Estimator

In the importance sampling TABI estimator presented in Section 3.2, we broke down the numerator as $E_1 = E_1^+ - E_1^-$, and then introduced separate estimators for each of these. Here we present an interesting alternative to estimating E_1 based on a Rao-Blackwellized multiple importance sampling (MIS) estimator (Veach and Guibas, 1995; Owen and Zhou, 2000) that, when combined with a separate estimator for E_2 in the same manner as TABI, produces an alternative overall estimator for which Theorem 1 also holds. We note that this approach is identical to TABI except in the breakdown of estimation of E_1, such that it is redundant in cases where $f(x) \geq 0 \, \forall x$ (or $f(x) \leq 0 \, \forall x$). In fact, the estimator will actually use exactly the same set of samples as TABI, but will effectively weight them in a different way. We also note that the overall alternative estimator does not fit into the conventional MIS framework due to the separate estimation of E_2. However, it does highlight some interesting links between TABI and MIS, while also providing the potential for some modest performance improvements in some scenarios.

As with TABI, the key to estimating E_1 for this approach is to use to proposals $q_1^+(x)$ and $q_1^-(x)$ whose optimal densities are proportional to $p(x,y)f^+(x)$ and $p(x,y)f^-(x)$ respectively. However, rather than using them to construct separate estimators, we instead consider a MIS estimator that uses the mixture proposal

$$q_1(x) = \nu q_1^+(x) + (1-\nu) q_1^-(x) \tag{54}$$

for some $0 < \nu < 1$, and then Rao-Blackwellizes the choice of the mixture component. Specifically, we derive our alternative estimator as

$$\begin{aligned} E_1 &= \mathbb{E}_{q_1(x)}\left[\frac{p(x,y)f(x)}{q_1(x)}\right] \\ &= \nu \mathbb{E}_{q_1^+(x)}\left[\frac{p(x,y)f(x)}{q_1(x)}\right] + (1-\nu)\mathbb{E}_{q_1^-(x)}\left[\frac{p(x,y)f(x)}{q_1(x)}\right] \\ &\approx \frac{\nu}{N}\sum_{n=1}^{N} \frac{f(x_n^+)p(x_n^+,y)}{q_1(x_n^+)} + \frac{1-\nu}{K}\sum_{k=1}^{K}\frac{f(x_k^-)p(x_k^-,y)}{q_1(x_k^-)} \quad \text{where} \quad x_n^+ \sim q_1^+(x),\ x_k^- \sim q_1^-(x), \\ &:= \hat{E}_1^{\text{MIS}}. \end{aligned} \tag{55}$$

In line with MIS, the natural choice of ν is $\nu = N/(N+K)$ (though others are possible), which leads to

$$\hat{E}_1^{\text{MIS}} = \sum_{n=1}^{N} \frac{f(x_n^+)p(x_n^+,y)}{Nq_1^+(x_n^+) + Kq_1^-(x_n^+)} + \sum_{k=1}^{K}\frac{f(x_k^-)p(x_k^-,y)}{Nq_1^+(x_k^-) + Kq_1^-(x_k^-)} \tag{56}$$

which is the form we will assume from here on. This can be combined with a separate estimator $\hat{E}_2$ for E_2 in the same manner as TABI (where $\hat{E}_2$ is also itself constructed in the same manner), giving

the overall estimator

$$\mathbb{E}_{p(x|y)}[f(x)] \approx \hat{\mu}^{\text{MIS}} := \frac{\hat{E}_1^{\text{MIS}}}{\hat{E}_2}. \tag{57}$$

Looking closely, we see that this estimator uses the same set of samples as TABI, but it combines them in a different way. The link is perhaps best seen by thinking of $\hat{E}_1^+ - \hat{E}_1^-$ in the TABI estimator as a single estimator of $N + K$ samples where each sample is weighted according to

$$\frac{\mathbb{I}(f(x) \geq 0)p(x,y)}{Nq_1^+(x)} \quad \text{or} \quad \frac{\mathbb{I}(f(x) \leq 0)p(x,y)}{Kq_1^-(x)}$$

depending on whether the sample came from $q_1^+(x)$ or $q_1^-(x)$. By comparison, $\hat{E}_1^{\text{MIS}}$, applies a weight

$$\frac{p(x,y)}{Nq_1^+(x) + Kq_1^-(x)}$$

regardless of which proposal the sample originated from.

It is now easy to see that if

$$q_1^+(x) = 0 \,\forall x : f(x) < 0 \quad \text{and} \quad q_1^-(x) = 0 \,\forall x : f(x) > 0 \tag{58}$$

then these two weighting schemes will be identical, such that $\hat{E}_1^{\text{MIS}} = \hat{E}_1^+ - \hat{E}_1^-$. As this is satisfied when using the optimal proposals for TABI, we see that $\hat{\mu}^{\text{MIS}}$ retains the key theoretical property of TABI that its error is not lower bounded even for finite sample sizes, i.e. Theorem 1 also holds for $\hat{\mu}^{\text{MIS}}$. However, if (58) does not hold, then the two estimators will behave slightly differently. Moreover, $\hat{E}_1^{\text{MIS}}$ will generally be the lower variance of the two (noting that both are unbiased). This is firstly because it removes the need to set the weights to zero if they fall outside the targeted region, thereby ensuring information from these samples is preserved and increasing the expected effective sample size. Secondly, weighting samples according to the implied mixture distribution is a well–established as a mechanism for reducing variance in the MIS literature (Elvira et al., 2019).

Typically though, the gains from this approach will be modest. Ignoring the pathological scenarios where $q_1^+(x)$ is a better proposal than $q_1^-(x)$ for E_1^- or $q_1^-(x)$ is a better proposal than $q_1^+(x)$ for E_1^+, then the gains will be larger the more overlap there is between $q_1^+(x)$ and $q_1^-(x)$. We have already seen that for the extreme where there is no overlap then the two estimators coincide and there are no gains. The other extreme occurs when $q_1^+(x) = q_1^-(x) \,\forall x$ and here if we take $N = K$ then we find $\hat{E}_1^{\text{MIS}}$ will be twice as efficient as $\hat{E}_1^+ - \hat{E}_1^-$ because (in expectation) half the samples of the latter will be wasted (i.e. half will have a weight of zero). As, by construction, we generally expect the overlap between $q_1^+(x)$ and $q_1^-(x)$ to be relatively small, the earlier scenario is anticipated to be much more typical, while even for the latter the gains are relatively modest. Empirically we found this to hold as well: conducting some simple tests of setup similar to that of Section 3.3 but where $f(x)$ is no longer non-negative, we found no distinguishable difference in the empirical performance of the two estimators, even when the proposals are not carefully constructed. Nonetheless, using $\hat{\mu}^{\text{MIS}}$ as an alternative to TABI might provide useful gains in some scenarios where constructing good proposals for both $q_1^+(x)$ and $q_1^-(x)$ is challenging.

Unfortunately, these gains come at some potentially significant costs. Firstly, the cost of the estimator is slightly higher as the density of all samples must now be evaluated under both $q_1^+(x)$ and $q_1^-(x)$ instead of just the one used to produce them. Unless the cost of evaluating $f(x)$ dominates, this extra cost is likely to be significant. Secondly, and perhaps more importantly, it does not naturally permit some of the demonstrated extensions of TABI such as TAAIS, the training phase of AMCI, or the generalized TABI form given in Section 6. Namely, these all relied on the separation of estimators for their success, e.g. using $f^+(x)p(x,y)$ as a target for learning $q^+(x)$ when using TAAIS, such that the entanglement of the two proposals for $\hat{E}_1^{\text{MIS}}$ causes complications. Consequently, the main immediate use cases we see for this alternative approach are static expectation settings and test–time construction of the estimator in AMCI.

Appendix D. Data Generation and Mini–Batching Procedure

AMCI operates in a slightly unusual setting for neural network training because instead of having a fixed data set, we are instead training on samples from our model $p(x, y)$. The typical way to perform batch stochastic gradient optimization involves many epochs over the training data set, stopping once the error increases on the validation set which is a symptom of overfitting. Each epoch is itself broken down into multiple iterations, wherein one takes a random mini-batch (subsample) from the data set (without replacement) and updates the parameters based on a stochastic gradient step using these samples, with the epoch finishing once the full data set has been used.

However, there are different ways the training can proceed when we have the ability to generate an infinite amount of data from our model $p(x, y)$ and we (at least in theory) no longer face the risk of overfitting. There are two extremes approaches one could take. The first one would be sampling two large but fixed-size data sets (training and validation) before the time of training and then following the standard training procedure for the finite data sets outlined above. The other extreme would be to completely surrender the idea of a data set or epoch, and sample each batch of data presented to the optimizer directly from $p(x, y)$. In this case, we would not need a validation data set as we would never be at risk of overfitting—we would finish the training once we are satisfied with the convergence of the loss value.

Paige and Wood (2016) found that the method which empirically performed best in a similar amortized inference setting was one in the middle between the two extremes outlined above. They suggest a method which decides when to sample new synthetic (training and validation) data sets, based on performance on the previous validation data set. They draw fixed–sized training and validation data sets and optimize the model using the standard finite data procedure on the training data set until the validation error increases. When that happens they sample new training and validation data sets and repeat the procedure. This continues until empirical convergence of the loss value. In practice, they allow a few missteps (steps of increasing value) for the validation loss before they sample new synthetic data sets, and limit the maximum number of optimization epochs performed on a single data set.

We use the above method throughout all of our experiments. We allowed a maximum of 2 missteps with respect to the validation data set and a maximum of 30 epochs on a single data set before sampling new data sets. We note that while training was robust to the number of missteps allowed, adopting the general scheme of Paige and Wood (2016) was very important in achieving effective training: we initially tried generating every batch directly from the model $p(x, y)$ and we found that the proposals often converged to the local minimum of just sampling from the prior.

The way training and validation data sets are generated is modified slightly when using the importance sampling approach for generating x and θ detailed in Section 5.2.3. Whenever we use the objective in (34), instead of sampling the training and validation data sets from the prior $p(x, y)$ we will sample them from the distribution $q'(\theta, x) \cdot p(y|x)$ where q' is a proposal chosen to be as close to $p(x)p(\theta)f(x;\theta)$ as possible.

For each of the experiments, we trained for 1000 generated data sets, the size of the training data set was 10 times the batch size, the size of the validation data set was equal to the batch size, and the batch sizes were 15000, 6000, and 2500 for the one– and five–dimensional tail integral, and cancer examples, respectively.

Appendix E. Additional Experimental Results for AMCI

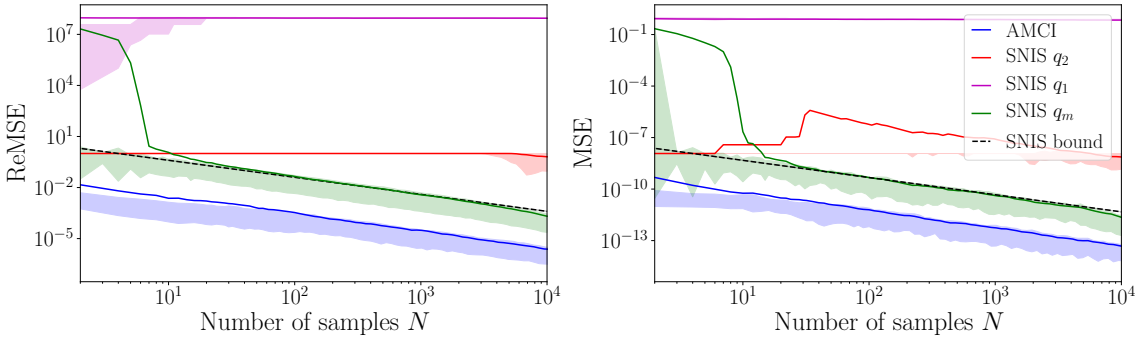


Figure 13: Additional results for one-dimensional tail integral example as per Figure 6a. [left] Relative mean squared errors (as per Eq 12) with q_1 now added to the plot and the axes rescaled. [right] MSE where we replace $\hat{\delta}(y,\theta)$ with $\mu(y,\theta) - \hat{\mu}(y,\theta)$ and otherwise proceed as before. Conventions as per Figure 6. The results for SNIS q_1 indicate that it often severely underestimates E_2 leading to very large errors. We also see that looking at the MSE and the relative MSE produce very similar qualitative comparisons.

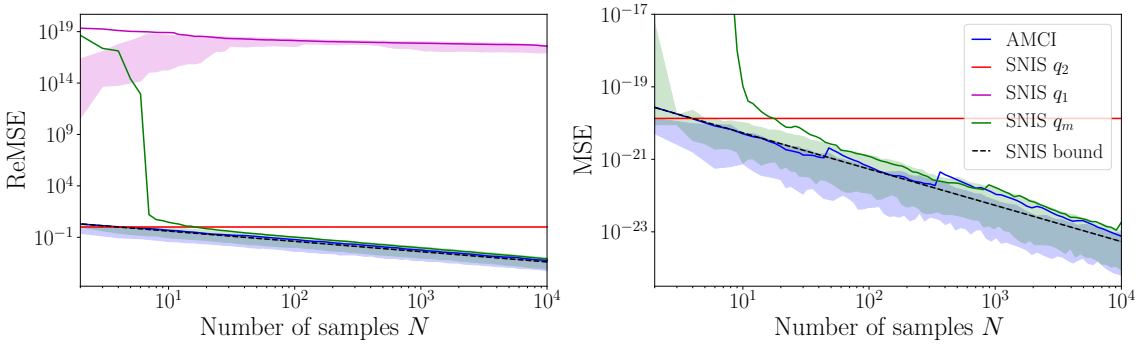


Figure 14: Additional results for five-dimensional tail integral from Figure 6b as per Figure 13. Conventions as per Figure 6b. Note the y-axis limits for the MSE have been readjusted to allow clear comparison at higher N. SNIS q_m yields an MSE of 10^{-1} at $N=2$, while the SNIS q_1 MSE is far away from the range of the plot for all N, giving an MSE of $10^{-0.9}$ at $N=2$ and $10^{-1.2}$ at $N=10^4$, with a shape very similar to the ReMSE for SNIS q_1 as per the left plot. The extremely high errors for SNIS q_m at low values of N arise in the situation when all N samples drawn happen to come from distribution q_1. We believe that the results presented for q_m underestimate the value of $\delta(y,\theta)$ between around $N=6$ and $N=100$, due to the fact that the estimation process for $\delta(y,\theta)$, though unbiased, can have a very large skew. Namely, for $N \leq 6$ there is a good chance of at least one of the 100 trials having all N samples originating from distribution q_1, such that we generate reasonable estimates for the very high errors this can induce. For $N \geq 100$ the chances of this event occurring drop to below 10^{-30}, such that it does not substantially influence the true error. For $6 \leq N \leq 100$, the chance the event will occur in our 100 trials is small, but the influence it has on the overall error is still significant, meaning it is likely we will underestimate the error. This effect could be alleviated by Rao-Blackwellizing the choice of the mixture component in a manner akin to that discussed in Appendix C, but the resulting estimator would no longer be an SNIS estimator.

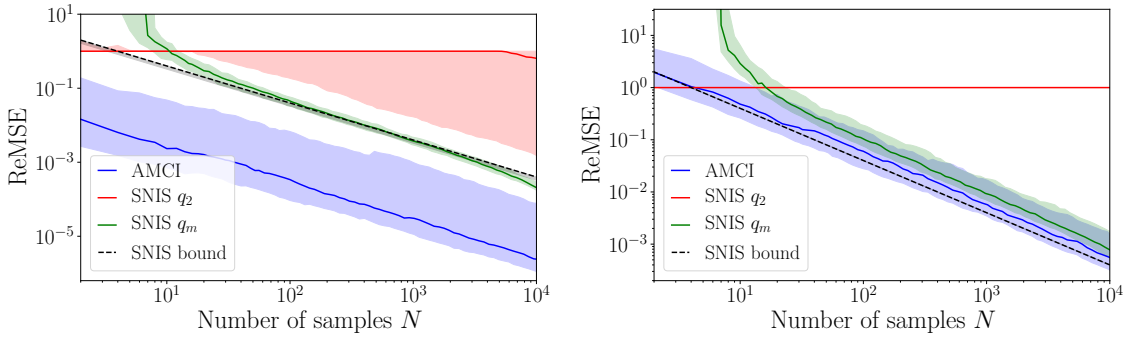


Figure 15: Investigation of the variability of the results across datapoints y, θ for [left] the one-dimensional and [right] the five-dimensional tail integral example. Unlike previous figures, the shading shows the estimates of the 25% and 75% quantiles of $\delta(y, \theta)$ estimated using a common set of 100 samples from $y, \theta \sim p(y)p(\theta)$, with each corresponding $\delta(y, \theta)$ separately estimated using 100 samples of the respective $\hat{\delta}(y, \theta)$. In other words, while the shading in previous plots has represented variability in our estimation of $\delta(y, \theta)$, we are now representing the variability in $\delta(y, \theta)$ itself over different (y, θ). The solid and dashed lines remain the same as in previous figures—they indicate the median of $\delta(y, \theta)$—but the dashed line now also has a shaded area associated with it reflecting the variability in the SNIS bound across datapoints. Qualitatively, we see similar behavior as before.

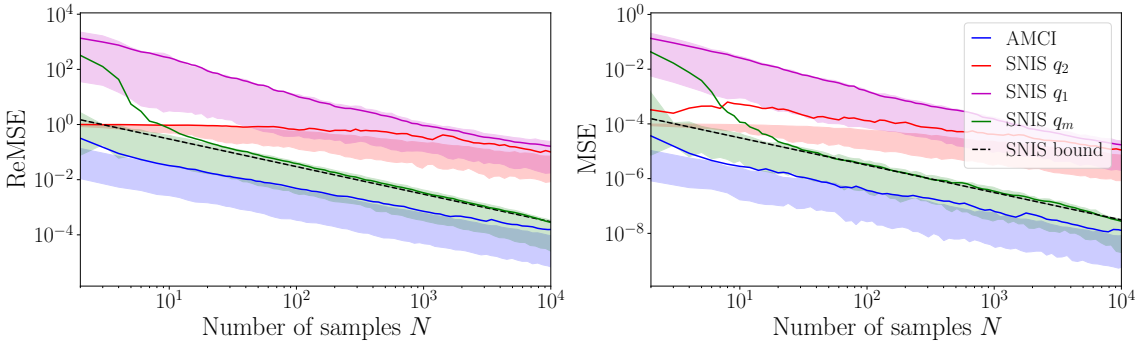


Figure 16: Additional results for cancer example from Figure 7 as per Figure 13. Conventions as per Figure 6. Here, SNIS q_1 performs much better than in the tail integral example because of smaller mismatch between $p(x|y)$ and $f(x; \theta)$, meaning the estimates for E_2 are more reasonable. Nonetheless, it still performs worse that even SNIS q_2. Qualitatively similar behavior is seen for the MSE as the ReMSE.

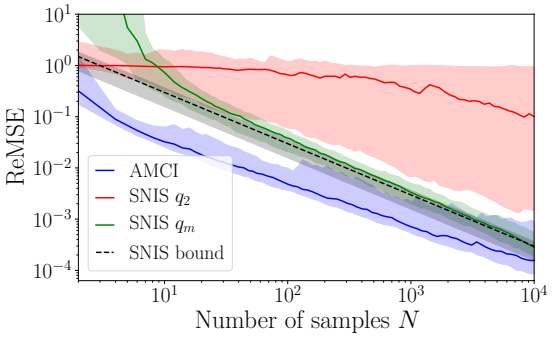


Figure 17: Investigation of the variability of the results across datapoints y, θ for the cancer example. Conventions as per Figure 15. The fact that the upper quantile of the AMCI error is larger than the upper quantile of the SNIS q_m error suggests that there are datapoints for which AMCI yields higher MSE than SNIS q_m. However, AMCI is still always better than our main baseline SNIS q_2.

References

Monica F Bugallo, Victor Elvira, Luca Martino, David Luengo, Joaquin Miguez, and Petar M Djuric. Adaptive importance sampling: the past, the present, and the future. *IEEE Signal Processing Magazine*, 34(4):60–79, 2017.

Olivier Cappé, Arnaud Guillin, Jean-Michel Marin, and Christian P Robert. Population Monte Carlo. *Journal of Computational and Graphical Statistics*, 13(4):907–929, 2004.

Olivier Cappé, Randal Douc, Arnaud Guillin, Jean-Michel Marin, and Christian P Robert. Adaptive importance sampling in general mixture classes. *Statistics and Computing*, 18(4):447–459, 2008.

Sourav Chatterjee, Persi Diaconis, et al. The sample size required in importance sampling. *The Annals of Applied Probability*, 28(2):1099–1135, 2018.

Ming-Hui Chen and Qi-Man Shao. On Monte Carlo methods for estimating ratios of normalizing constants. *The Annals of Statistics*, 25(4):1563–1594, 08 1997.

Junesang Choi and HM Srivastava. Some summation formulas involving harmonic numbers and generalized harmonic numbers. *Mathematical and Computer Modelling*, 54(9-10):2220–2234, 2011.

Nicolas Chopin and Christian P Robert. Properties of nested sampling. *Biometrika*, 97(3):741–755, 2010.

Adam D Cobb, Stephen J Roberts, and Yarin Gal. Loss-calibrated approximate inference in Bayesian neural networks. *arXiv:1805.03901*, 2018.

Julien Cornebise, Éric Moulines, and Jimmy Olsson. Adaptive methods for sequential importance sampling with application to state space models. *Statistics and Computing*, 18(4):461–480, 2008.

Jean Cornuet, Jean-Michel Marin, Antonietta Mira, and Christian P Robert. Adaptive multiple importance sampling. *Scandinavian Journal of Statistics*, 39(4):798–812, 2012.

Chris Cremer, Xuechen Li, and David Duvenaud. Inference suboptimality in variational autoencoders. *International Conference on Machine Learning (ICML)*, 2018.

Randal Douc, Arnaud Guillin, J-M Marin, and Christian P Robert. Convergence of adaptive mixtures of importance sampling schemes. *The Annals of Statistics*, pages 420–448, 2007.

Arnaud Doucet, Nando De Freitas, and Neil Gordon. An introduction to sequential Monte Carlo methods. In *Sequential Monte Carlo Methods in Practice*, pages 3–14. Springer, 2001.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. *Neural Information Processing Systems (NeurIPS)*, 2019.

Víctor Elvira, Luca Martino, David Luengo, and Jukka Corander. A gradient adaptive population importance sampler. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015.

Víctor Elvira, Luca Martino, David Luengo, and Mónica F Bugallo. Generalized multiple importance sampling. *Statistical Science*, 34(1):129–155, 2019.

Heiko Enderling and Mark AJ Chaplain. Mathematical modeling of tumor growth and treatment. *Current Pharmaceutical Design*, 20–30:4934–40, 2014.

Michael Evans. Discussion of nested sampling for Bayesian computations by John Skilling. *Bayesian Statistics*, 8:491–524, 2007.

Michael Evans and Tim Swartz. Methods for approximating integrals in statistics with special emphasis on Bayesian integration problems. *Statistical Science*, 1995.

Farhan Feroz, Michael P. Hobson, Ewan Cameron, and Anthony N. Pettitt. Importance nested sampling and the MultiNest algorithm. *The Open Journal of Astrophysics*, 2019.

Sylvia Frühwirth-Schnatter. Estimating marginal likelihoods for mixture and markov switching models using bridge sampling techniques. *The Econometrics Journal*, 7(1):143–167, 2004.

Andrew Gelman and Xiao-Li Meng. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical Science*, 13(2):163–185, 05 1998.

Samuel Gershman and Noah Goodman. Amortized inference in probabilistic reasoning. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, 2014.

Adam Goliński, Frank Wood, and Tom Rainforth. Amortized Monte Carlo Integration. *International Conference on Machine Learning (ICML)*, 2019.

Quentin F Gronau, Alexandra Sarafoglou, Dora Matzke, Alexander Ly, Udo Boehm, Maarten Marsman, David S Leslie, Jonathan J Forster, Eric-Jan Wagenmakers, and Helen Steingroever. A tutorial on bridge sampling. *Journal of Mathematical Psychology*, 81:80–97, 2017.

Philip Hahnfeldt, Dipak Panigrahy, Judah Folkman, and Lynn Hlatky. Tumor development under angiogenic signaling. *Cancer Research*, 59(19):4770–4775, 1999.

Timothy Classen Hesterberg. *Advances in Importance Sampling*. PhD thesis, Stanford University, 1988.

Matthew D. Hoffman, David M. Blei, Chong Wang, and John Paisley. Stochastic variational inference. *Journal of Machine Learning Research (JMLR)*, 2013.

Herman Kahn and Andy W Marshall. Methods of reducing sample size in Monte Carlo computations. *Journal of the Operations Research Society of America*, 1(5):263–278, 1953.

Johannes Kästner. Umbrella sampling. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 1(6):932–942, 2011.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *International Conference on Learning Representations (ICLR)*, 2015.

Diederik P. Kingma and Max Welling. Auto-encoding variational Bayes. *International Conference on Learning Representations (ICLR)*, 2014.

Simon Lacoste-Julien, Ferenc Huszár, and Zoubin Ghahramani. Approximate inference for the loss-calibrated Bayesian. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2011.

Roland Lamberti, Yohan Petetin, François Septier, and François Desbouvries. A double proposal normalized importance sampling estimator. *IEEE Statistical Signal Processing Workshop (SSP)*, 2018.

Tuan Anh Le, Atılım Güneş Baydin, and Frank Wood. Inference compilation and universal probabilistic programming. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2017.

Tuan Anh Le, Maximilian Igl, Tom Jin, Tom Rainforth, and Frank Wood. Auto-encoding sequential Monte Carlo. *International Conference on Learning Representations (ICLR)*, 2018.

Tuan Anh Le, Adam R Kosiorek, N Siddharth, Yee Whye Teh, and Frank Wood. Revisiting reweighted wake-sleep. *Uncertainty in Artificial Intelligence (UAI)*, 2019.

Xiaoyu Lu, Tom Rainforth, Yuan Zhou, Jan-Willem van de Meent, and Yee Whye Teh. On exploration, exploitation and learning in adaptive importance sampling. *arXiv:1810.13296*, 2018.

Luca Martino, Victor Elvira, David Luengo, and Jukka Corander. Layered adaptive importance sampling. *Statistics and Computing*, 27(3):599–623, 2017.

Charles Matthews, Jonathan Weare, Andrey Kravtsov, and Elise Jennings. Umbrella sampling: A powerful method to sample tails of distributions. *Monthly Notices of the Royal Astronomical Society*, 480(3):4069–4079, 2018.

Xiao-Li Meng and Stephen Schilling. Warp bridge sampling. *Journal of Computational and Graphical Statistics*, 11(3):552–586, 2002.

Xiao-Li Meng and Wing Hung Wong. Simulating ratios of normalizing constants via a simple identity: A theoretical exploration. *Statistica Sinica*, 6:831–860, 1996.

Mihaly Mezei. Adaptive umbrella sampling: Self-consistent determination of the non-Boltzmann bias. *Journal of Computational Physics*, 68(1):237–248, 1987.

Thomas Müller, Brian McWilliams, Fabrice Rousselle, Markus Gross, and Jan Novák. Neural importance sampling. *ACM Transactions on Graphics*, 2019.

Radford M Neal. Annealed importance sampling. *Statistics and Computing*, 11(2):125–139, 2001.

Man-Suk Oh and James O Berger. Adaptive importance sampling in Monte Carlo integration. *Journal of Statistical Computation and Simulation*, 41(3-4):143–168, 1992.

Art Owen. *Monte Carlo Theory, Methods and Examples.* 2013.

Art Owen and Yi Zhou. Safe and effective importance sampling. *Journal of the American Statistical Association*, 95(449):135–143, 2000.

Brooks Paige and Frank Wood. Inference networks for sequential Monte Carlo in graphical models. *International Conference on Machine Learning (ICML)*, 2016.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. *Neural Information Processing Systems (NeurIPS)*, 2017.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing flows for probabilistic modeling and inference. *arXiv:1912.02762*, 2019.

François Portier and Bernard Delyon. Asymptotic optimality of adaptive importance sampling. *Neural Information Processing Systems (NeurIPS)*, 2018.

Tom Rainforth. *Automating Inference, Learning, and Design Using Probabilistic Programming*. PhD thesis, 2017.

Tom Rainforth, Robert Cornish, Hongseok Yang, Andrew Warrington, and Frank Wood. On nesting Monte Carlo estimators. *International Conference on Machine Learning (ICML)*, 2018a.

Tom Rainforth, Yuan Zhou, Xiaoyu Lu, Yee Whye Teh, Frank Wood, Hongseok Yang, and Jan-Willem van de Meent. Inference trees: Adaptive inference with exploration. *arXiv:1806.09550*, 2018b.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. *International Conference on Machine Learning (ICML)*, 2015.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. *International Conference on Machine Learning (ICML)*, 2014.

Daniel Ritchie, Paul Horsfall, and Noah D Goodman. Deep amortized inference for probabilistic programs. *arXiv:1610.05735*, 2016.

Christian Robert. *The Bayesian Choice: From Decision-Theoretic Foundations to Computational Implementation*. Springer Science & Business Media, 2007.

Christian Robert and George Casella. *Monte Carlo Statistical Methods*. Springer Science & Business Media, 2013.

Ruslan Salakhutdinov and Hugo Larochelle. Efficient learning of deep Boltzmann machines. *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

John Skilling. Nested sampling. In *AIP Conference Proceedings*, volume 735, pages 395–405. AIP, 2004.

John Skilling. Nested sampling's convergence. In *AIP Conference Proceedings*, volume 1193, pages 277–291. AIP, 2009.

Evgeny Slutsky. Uber stochastische asymptoten und grenzwerte. *Metron*, 5(3):3–89, 1925.

Andreas Stuhlmüller, Jacob Taylor, and Noah Goodman. Learning stochastic inverses. *Neural Information Processing Systems (NeurIPS)*, 2013.

Erik H Thiede, Brian Van Koten, Jonathan Weare, and Aaron R Dinner. Eigenvector method for umbrella sampling enables error analysis. *The Journal of Chemical Physics*, 145(8):084115, 2016.

Glenn M Torrie and John P Valleau. Nonphysical sampling distributions in Monte Carlo free-energy estimation: Umbrella sampling. *Journal of Computational Physics*, 23(2):187–199, 1977a.

GM Torrie and JP Valleau. Monte Carlo study of a phase-separating liquid mixture by umbrella sampling. *The Journal of Chemical Physics*, 66(4):1402–1408, 1977b.

Eric Veach and Leonidas J Guibas. Optimally combining sampling techniques for Monte Carlo rendering. *Conference on Computer Graphics and Interactive Techniques*, 1995.

Peter Virnau and Marcus Müller. Calculation of free energy through successive umbrella sampling. *The Journal of Chemical Physics*, 120(23):10925–10930, 2004.

Hanna M Wallach, Iain Murray, Ruslan Salakhutdinov, and David Mimno. Evaluation methods for topic models. *International Conference on Machine Learning (ICML)*, 2009.

Lazhi Wang and Xiao-Li Meng. Warp bridge sampling: The next generation. *arXiv:1609.07690*, 2016.

Stefan Webb, Adam Goliński, Robert Zinkov, N. Siddharth, Tom Rainforth, Yee Whye Teh, and Frank Wood. Faithful inversion of generative models for effective amortized inference. *Neural Information Processing Systems (NeurIPS)*, 2018.

Robert L Wolpert. Monte Carlo integration in Bayesian statistical analysis. *Contemporary Mathematics*, 115:101–116, 1991.

Yuhuai Wu, Yuri Burda, Ruslan Salakhutdinov, and Roger Grosse. On the quantitative analysis of decoder-based generative models. *International Conference on Learning Representations (ICLR)*, 2017.

Ping Zhang. Nonparametric importance sampling. *Journal of the American Statistical Association*, 91(435):1245–1253, 1996.