SOFTCVI: CONTRASTIVE VARIATIONAL INFERENCE WITH SELF-GENERATED SOFT LABELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Estimating a distribution given access to its unnormalized density is pivotal in Bayesian inference, where the posterior is generally known only up to an unknown normalizing constant. Variational inference and Markov chain Monte Carlo methods are the predominant tools for this task; however, both are often challenging to apply reliably, particularly when the posterior has complex geometry. Here, we introduce Soft Contrastive Variational Inference (SoftCVI), which allows a family of variational objectives to be derived through a contrastive estimation framework. The approach parameterizes a classifier in terms of a variational distribution, reframing the inference task as a contrastive estimation problem aiming to identify a single true posterior sample among a set of samples. Despite this framing, we do not require positive or negative samples, but rather learn by sampling the variational distribution and computing ground truth soft classification labels from the unnormalized posterior itself. The objectives have zero variance gradient when the variational approximation is exact, without the need for specialized gradient estimators. We empirically investigate the performance on a variety of Bayesian inference tasks, using both simple (e.g. normal) and expressive (normalizing flow) variational distributions. We find that SoftCVI can be used to form objectives which are stable to train and mass-covering, frequently outperforming inference with other variational approaches.

1 INTRODUCTION

Consider a probabilistic model with a set of parameters $\boldsymbol{\theta}$, for which given a set of observations $\boldsymbol{x}_{\text{obs}}$ we wish to infer a posterior $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$. Unless the model takes a particularly convenient form, the posterior cannot be directly computed. However, typically $p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})$ is available, related to the posterior by $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}}) = p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})/p(\boldsymbol{x}_{\text{obs}})$, where $p(\boldsymbol{x}_{\text{obs}})$ is an intractable normalizing constant $p(\boldsymbol{x}_{\text{obs}}) = \int p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})d\boldsymbol{\theta}$. In these cases, computational methods such as Markov chain Monte Carlo (MCMC) (Hastings, 1970; Metropolis et al., 1953) or variational inference (Jordan et al., 1999; Kingma & Welling, 2014) are required to perform inference.

Variational inference approaches the inference task as an optimization problem by defining a variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ and minimizing its divergence to the true posterior $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$. The performance and reliability of variational inference is dependent on numerous factors, including the choice of divergence, variational distribution and parameter initialization. Whilst choosing a divergence that favors mass-covering posterior estimates may facilitate reliable inference, in many cases these divergences introduce bias or are less stable to train, leading to worse performance (Dhaka et al., 2021; Naesseth et al., 2020). Alongside difficulties in assessing performance (Yao et al., 2018), these issues hinder practical applications of variational inference, particularly in applications such as scientific research, where accurate uncertainty quantification is crucial.

In contrast to variational inference, contrastive learning is generally used to perform inference when the likelihood function is only available through an intractable integral, such as for fitting energy-based models or for performing simulation-based inference (Gutmann et al., 2022). Learning is achieved by contrasting positive samples with negative samples, where the negative samples are often generated by augmenting positive samples or by drawing samples from a carefully chosen (possibly highly structured) noise distribution. A common theme is to leverage the invariance of classification-based objectives to unknown normalizing constants to enable learning where likelihood-based methods are infeasible.

Contribution
We show that contrastive estimation can be used to derive a family of variational objectives, terming the approach Soft Contrastive Variational Inference (SoftCVI). The task of fitting the posterior approximation is reframed as a classification task, aiming to identify a single true posterior sample among a set of samples. Instead of using explicitly positive and negative samples, we show that for arbitrary samples from a proposal distribution, we can generate ground truth soft classification labels using the unnormalized posterior density itself. The samples and corresponding labels are used for fitting a classifier parameterized in terms of the variational distribution, such that the optimal classifier recovers the true posterior. We find SoftCVI enables derivation of stable and mass-covering objectives, and demonstrate the performance across a series of experiments with both simple (normal) and flexible (normalizing flow) variational distributions. Compared to alternative variational objectives, we find SoftCVI generally yields better calibrated posteriors with a lower forward Kullback-Leibler (KL) divergence to the true posterior. We provide a pair of Python packages, **[REDACTED]** and **[REDACTED]** , which provide the implementation, and the code for reproducing the results of this paper, respectively.

2 SOFTCVI

In order to fit a variational distribution with SoftCVI, we must define a proposal distribution $\pi(\boldsymbol{\theta})$, a negative distribution $p^{-}(\boldsymbol{\theta})$, and the variational distribution itself, $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$. At each optimization step, three steps are performed which allow fitting the variational distribution:

1. Sample parameters $\{\boldsymbol{\theta}_k\}_{k=1}^K \sim \pi(\boldsymbol{\theta})$ from the proposal distribution $\pi(\boldsymbol{\theta})$.
2. Generate corresponding ground truth soft labels $\boldsymbol{y} \in (0,1)^K$ for the task of classifying between positive and negative samples, presumed to be from $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$ and $p^{-}(\boldsymbol{\theta})$, respectively.
3. Use $\{\boldsymbol{\theta}_k\}_{k=1}^K$ along with the soft labels $\boldsymbol{y}$ to optimize a classifier parameterized in terms of the variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, such that the optimal classifier recovers the true posterior.

The choice of proposal distribution in step one will influence the region in which learning is focused. Throughout the experiments here, we take the intuitive and convenient choice of using the variational distribution itself as the proposal distribution $\pi(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, which over the course of training directs learning towards regions with reasonable posterior mass. The remainder of this section is structured as follows: section 2.1 details the assignment of ground truth labels to arbitrary proposal samples; section 2.2 discusses the parameterization and optimization of the classifier; finally, section 2.3 considers the choice of negative distribution.

2.1 GENERATING GROUND TRUTH SOFT LABELS

Classification can be used to estimate the density ratio between positive and negative distributions (Gutmann & Hyvärinen, 2012; Hastie, 2009; Oord et al., 2018; Thomas et al., 2022). Conversely, if the positive and negative densities are known, the density ratio and ground truth classification labels can be directly computed. Consider we have a set of $\{\boldsymbol{\theta}_k\}_{k=1}^K$ samples from the proposal distribution $\pi(\boldsymbol{\theta})$. SoftCVI reframes inference as a classification task, where the problem is chosen such that analytical ground truth labels can be assigned to the K samples. Specifically, if we consider the true posterior $p(\boldsymbol{\theta}_k|\boldsymbol{x}_{\text{obs}})$ to be the positive distribution, and $p^{-}(\boldsymbol{\theta})$ to be a chosen negative distribution, the ratio $p(\boldsymbol{\theta}_k|\boldsymbol{x}_{\text{obs}})/p^{-}(\boldsymbol{\theta}_k)$ represents the relative likelihood of a sample being a true posterior sample under the classification problem. By contrasting the K samples, assuming a setting where $\{\boldsymbol{\theta}_k\}_{k=1}^K$ consists of exactly one positive sample from the true posterior $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$, and $K-1$ negative samples from $p^{-}(\boldsymbol{\theta})$, the optimal classifier is then given by

$$y_k = \frac{p(\boldsymbol{\theta}_k|\boldsymbol{x}_{\text{obs}})/p^{-}(\boldsymbol{\theta}_k)}{\sum_{k'=1}^K p(\boldsymbol{\theta}_{k'}|\boldsymbol{x}_{\text{obs}})/p^{-}(\boldsymbol{\theta}_{k'})}, \tag{1}$$

where y_k is the probability on the interval $(0,1)$ that $\boldsymbol{\theta}_k$ is the positive sample among all the considered samples, and $\sum_{k=1}^K y_k = 1$. This approach of contrasting samples is invariant to multiplicative scaling to the density ratio $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})/p^{-}(\boldsymbol{\theta})$, meaning that access only to unnormalized forms of $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$ and $p^{-}(\boldsymbol{\theta})$ is sufficient for computing the labels. Since typically $p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})$ is known and proportional to $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$, we can analytically compute the labels as

$$\begin{aligned} y_k &= \frac{p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})/p^{-}(\boldsymbol{\theta}_k)}{\sum_{k'=1}^{K} p(\boldsymbol{\theta}_{k'}, \boldsymbol{x}_{\text{obs}})/p^{-}(\boldsymbol{\theta}_{k'})}, \\ \boldsymbol{y} &= \operatorname{softmax}(\boldsymbol{z}), \quad \text{where } z_k = \log \frac{p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})}{p^{-}(\boldsymbol{\theta}_k)}. \end{aligned} \tag{2}$$

where $p^{-}(\boldsymbol{\theta})$ may be unnormalized. While contrastive estimation methods generally use explicitly positive and negative examples with hard labels $\boldsymbol{y} \in \{0, 1\}^K$, here, in our framework, we lack access to positive samples from $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$, and we may not have access to samples from $p^{-}(\boldsymbol{\theta})$. Nonetheless, by instead generating samples from a proposal distribution, $\{\boldsymbol{\theta}_k\}_{k=1}^K \sim \pi(\boldsymbol{\theta})$, and by assigning soft labels to these samples using eq. (2), it is still possible to train a classifier which at optimality recovers the true posterior, as we will discuss in the subsequent section.

2.2 PARAMETERIZATION AND OPTIMIZATION OF THE CLASSIFIER

Analogously to the computation of the labels in eq. (2), we can parameterize the classifier in terms of the ratio between the variational and negative distribution

$$\begin{aligned} \hat{y}_k &= \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^{-}(\boldsymbol{\theta}_k)}{\sum_{k'=1}^{K} q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_{k'})/p^{-}(\boldsymbol{\theta}_{k'})}, \\ \hat{\boldsymbol{y}} &= \operatorname{Softmax}(\hat{\boldsymbol{z}}), \quad \text{where } \hat{z}_k = \log \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^{-}(\boldsymbol{\theta}_k)}. \end{aligned} \tag{3}$$

Given a set of samples $\{\boldsymbol{\theta}_k\}_{k=1}^K \sim \pi(\boldsymbol{\theta})$ and corresponding labels $\boldsymbol{y}$ computed using eq. (2), an optimization step with respect to $\boldsymbol{\phi}$ can be taken to minimize the softmax cross-entropy loss function (i.e. the negative multinomial log-likelihood)

$$\mathcal{L}_{\text{SoftCVI}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K, \boldsymbol{y}) = -\sum_{k=1}^{K} y_k \log(\hat{y}_k) = -\sum_{k=1}^{K} y_k \log\left(\frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^{-}(\boldsymbol{\theta}_k)}{\sum_{k'=1}^{K} q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_{k'})/p^{-}(\boldsymbol{\theta}_{k'})}\right). \tag{4}$$

Let Θ represent the support of the proposal distribution, with $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$ and $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ supported on the same set, and $p^{-}(\boldsymbol{\theta})$ supported on a superset of Θ. Assume that there exists a $\boldsymbol{\phi}$ such that $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$. The optimal classifier, i.e. which minimizes eq. (4) for all $\{\boldsymbol{\theta}_k\}_{k=1}^K \in \Theta^K$, recovers the true posterior, for any valid choice of K.[1] This result follows from the fact the optimal classifier must learn the density ratio between the positive and negative distributions up to a constant across Θ

$$\begin{aligned} p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})/p^{-}(\boldsymbol{\theta}) &= c \cdot q_{\boldsymbol{\phi}}(\boldsymbol{\theta})/p^{-}(\boldsymbol{\theta}), \\ p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}}) &= c \cdot q_{\boldsymbol{\phi}}(\boldsymbol{\theta}). \end{aligned}$$

Due to the shared support, integrating both sides over the proposal support Θ gives

$$\begin{aligned} \int_{\Theta} p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}}) &= c \cdot \int_{\Theta} q_{\boldsymbol{\phi}}(\boldsymbol{\theta}), \\ c &= 1. \end{aligned} \tag{5}$$

Thus the optimal classifier recovers the true posterior

$$p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta}). \tag{6}$$

Importantly, this suggests if $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ or $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$ have non-zero regions outside the support of $\pi(\boldsymbol{\theta})$, then $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}}) = c \cdot q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ can be satisfied within Θ without recovering the true posterior, as the integrals in eq. (5) will not both evaluate to one. This is a consequence of the incentive to learn the

---

[1] Or equivalently, the optimal classifier minimizes $\mathbb{E}_{\{\boldsymbol{\theta}_k\}_{k=1}^K \sim \pi(\boldsymbol{\theta})}[\mathcal{L}_{\text{SoftCVI}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K, \boldsymbol{y})]$, for which we can view eq. (4) as a single sample Monte Carlo approximation.

density ratio (up to a constant) being localized to the sampled region. Ensuring $\pi(\boldsymbol{\theta})$, $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ and $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$ are supported on the same set Θ can be achieved by defining a variational distribution supported on the same set as the posterior $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$, alongside using the variational distribution as the proposal distribution $\pi(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$.

When the optimal classifier is achieved, we show in appendix A.1 that the gradient is zero for any set $\{\boldsymbol{\theta}_k\}_{k=1}^K \in \Theta^K$, and consequently has zero variance. This is a generally desirable property that is not present in many variational objectives, although specialized gradient estimators have been developed which in some cases can address this issue (Roeder et al., 2017; Tucker et al., 2018). An algorithm outlining the overall approach of SoftCVI is shown in algorithm 1.

Algorithm 1: SoftCVI

```
Inputs: $p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})$, $\pi(\boldsymbol{\theta})$, $p^-(\boldsymbol{\theta})$, $q_{\boldsymbol{\phi}_1}(\boldsymbol{\theta})$, number of samples $K \geq 2$, optimization steps $N$, learning rate $\eta$
for $i$ in $1:N$ do
    Sample $\{\boldsymbol{\theta}_k\}_{k=1}^K \sim \pi(\boldsymbol{\theta})$
    Compute soft labels $\boldsymbol{y} = \text{Softmax}(\boldsymbol{z})$, where $z_k = \log \frac{p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})}{p^-(\boldsymbol{\theta}_k)}$
    Update $\boldsymbol{\phi}_{i+1} = \boldsymbol{\phi}_i - \eta \nabla_{\boldsymbol{\phi}} \mathcal{L}(\boldsymbol{\phi}_i; \{\boldsymbol{\theta}_k\}_{k=1}^K, \boldsymbol{y})$ using the cross-entropy loss, eq. (4)
end
```

2.3 CHOICE OF THE NEGATIVE DISTRIBUTION

The choice of negative distribution is a well-known challenge in contrastive learning, with the optimal negative distribution often differing significantly from the positive distribution (Chehab et al., 2022). In contrast to standard applications of contrastive estimation, in SoftCVI, the negative distribution is never sampled, meaning the impact of the choice is limited to its influence on the objective's properties. To better understand this, we can rewrite the softmax cross-entropy objective from eq. (4) by separating out the log term

$$\begin{aligned}\mathcal{L}_{\text{SoftCVI}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K, \boldsymbol{y}) &= -\sum_{k=1}^K y_k \log\left(\frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^-(\boldsymbol{\theta}_k)}\right) + \sum_{k=1}^K y_k \log \sum_{k'=1}^K \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_{k'})}{p^-(\boldsymbol{\theta}_{k'})}, \\ &= -\sum_{k=1}^K y_k \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k) + \log \sum_{k=1}^K \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^-(\boldsymbol{\theta}_k)} + \text{const},\end{aligned} \tag{7}$$

where in the last line we remove the denominator from the first term into a constant as it is independent of $\boldsymbol{\phi}$, and use $\sum_{k=1}^K y_k = 1$. The first term encourages placing posterior mass on the samples likely to be from the true posterior, and the second term penalizes the sum of the ratios. In addition to appearing in the denominator of the second term, the negative distribution choice also influences the labels y_k through eq. (2).

We focus on setting $p^-(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta})^\alpha$, where $\alpha \in [0, 1]$ is a tempering hyperparameter which interpolates between directly using the proposal distribution as the negative distribution when $\alpha = 1$ and using an improper flat negative distribution when $\alpha = 0$. Lower values of α tend to favor more mass-covering solutions by increasing the relative penalization of samples in higher density regions in $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ through the second term of eq. (7). However, a too small choice for α can lead to problematically high variances of the log ratios $z_k = \log[p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})/p^-(\boldsymbol{\theta}_k)]$ and $\hat{z}_k = \log[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^-(\boldsymbol{\theta}_k)]$. Particularly in high-dimensional problems, this can result in labels and predictions with very few non-zero values, meaning few samples contribute significantly to the loss. In terms of eq. (7), the high variance of $\hat{z}_k$ leads to a failure to sufficiently penalize the ratios in lower density regions of $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, as $\log \sum_{k=1}^K \exp(\hat{z}_k)$ is dominated by the few largest ratios. In practice, we empirically show this leads to "leakage" of mass into regions of negligible posterior density (fig. 2), and a weak signal-to-noise ratio (SNR) (see appendix A.3).

As described in the introduction to section 2, we choose the proposal distribution to equal the variational distribution, meaning the choice above is equivalent to using $p^-(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})^\alpha$. However, when computing the objective gradient, we treat $p^-(\boldsymbol{\theta})$ as independent of $\boldsymbol{\phi}$, which practically can be achieved by applying the stop-gradient operator present in automatic differentiation packages.[2] In appendix A.6, we also consider parameterizing the negative distribution using $p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})^{\alpha}$; however, this choice introduces the problematic ratio $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})/p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})$, which leads to favoring of mode-seeking solutions and poorly calibrated posteriors.

3 RELATED WORK

3.1 VARIATIONAL INFERENCE

Given access to an unnormalized posterior density, $p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})$, variational inference allows optimizing a variational distribution $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ to approximate the posterior. For certain model classes there exist closed form solutions which can be exploited during optimization (e.g. Blei et al., 2017; Ghahramani & Beal, 2000; Parisi, 1988). However, the lack of broad applicability of these methods hinders the ability of practitioners to freely alter the model and variational family. As such, we instead focus on methods which place minimal restrictions on the model form and variational family.

3.1.1 EVIDENCE LOWER BOUND

The most commonly used variational objective is to minimize the negative Evidence Lower Bound (ELBO) or equivalently, minimizing the reverse KL divergence between the posterior approximation and the true posterior

$$\begin{aligned} D_{KL}[q_{\boldsymbol{\phi}}(\boldsymbol{\theta})||p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})] &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta})}\left[\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) - \log p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})\right], \\ &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta})}[\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) - \log p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})] + \text{const.} \end{aligned} \tag{8}$$

As there is no general closed-form solution for the divergence, a Monte Carlo approximation is often used (Kingma & Welling, 2014)

$$\mathcal{L}_{\text{ELBO}}(\boldsymbol{\phi}) = \frac{1}{K}\sum_{k=1}^{K}\left[\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k) - \log p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})\right], \text{ where } \{\boldsymbol{\theta}_k\}_{k=1}^{K} \sim q_{\boldsymbol{\phi}}(\boldsymbol{\theta}). \tag{9}$$

Although straightforward to apply, the reverse KL divergence heavily punishes placing mass in regions with little mass in the true posterior leading to mode-seeking behavior and lighter tails than $p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$. In addition to underestimating uncertainty, the light tails may also lead to the approximation performing poorly in downstream tasks, such as when acting as a proposal distribution in importance sampling (Chatterjee & Diaconis, 2018; Gelman & Meng, 1998; Müller et al., 2019b; Yao et al., 2018) or for reparameterizing MCMC (Hoffman et al., 2019).

3.1.2 SELF-NORMALIZED IMPORTANCE SAMPLING FORWARD KL DIVERGENCE

In order to address the limitations of the ELBO, numerous alternative objectives have been proposed that encourage more mass-covering behavior, such as the importance weighted ELBO (Burda et al., 2015), the Rényi divergence, (Li & Turner, 2016), χ-divergence (Dieng et al., 2017), and methods targeting the forward KL divergence (Jerfel et al., 2021; Naesseth et al., 2020). In this section, we will focus on the Self-Normalized Importance Sampling Forward Kullback-Leibler (SNIS-fKL) divergence estimator introduced by Jerfel et al. (2021). Specifically, a standard importance weighted estimate of the forward KL divergence is given by

$$D_{KL}[p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})||q_{\boldsymbol{\phi}}(\boldsymbol{\theta})] = \mathbb{E}_{\pi(\boldsymbol{\theta})}\left[w(\boldsymbol{\theta})\log\frac{p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta})}\right] + \text{const}, \tag{10}$$

where $w(\boldsymbol{\theta}) = p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})/\pi(\boldsymbol{\theta})$ is the importance weights. Computing a set of self-normalized weights $\tilde{w}(\boldsymbol{\theta}_k) = \frac{p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})/\pi(\boldsymbol{\theta}_k)}{\sum_{k'=1}^{K} p(\boldsymbol{\theta}_{k'}, \boldsymbol{x}_{\text{obs}})/\pi(\boldsymbol{\theta}_{k'})}$ and using these alongside a Monte Carlo approximation of eq. (10), yields the objective

$$\mathcal{L}_{\text{SNIS-fKL}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^{K}) = \sum_{k=1}^{K}\tilde{w}(\boldsymbol{\theta}_k)\log\frac{p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}, \tag{11}$$

---

[2] Without preventing gradient flow through $p^{-}(\boldsymbol{\theta})$, the choice $p^{-}(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ leads to zero gradients as the parameterization of the classifier becomes $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})/q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$.

which introduces bias in the approximation of the forward KL vanishing with order $\mathcal{O}(1/K)$ (Agapiou et al., 2017). Generally, the proposal is chosen to equal the variational distribution $\pi(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, and the proposal parameters are held constant under differentiation.

3.2 COMPARISON OF SOFTCVI AND SNIS-FKL

In this section, we demonstrate that in the special case of choosing $p^-(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, both SoftCVI and SNIS-fKL produce gradients that are equal in expectation, but the SoftCVI objective naturally includes a control variate which ensures the variance of the gradient decreases to zero as the variational distribution approaches the true posterior. We use this result to suggest an alternative, lower-variance gradient estimator for the SNIS-fKL objective, exactly equivalent to optimizing the SoftCVI objective with $p^-(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$.

Noticing that from eq. (11), the numerator term does not depend on $\boldsymbol{\phi}$, we can equivalently write the SNIS-fKL objective as

$$\mathcal{L}_{\text{SNIS-fKL}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K) = -\sum_{k=1}^K \tilde{w}(\boldsymbol{\theta}_k) \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k) + \text{const.} \tag{12}$$

In the special case under consideration, the self-normalized weights $\tilde{w}(\boldsymbol{\theta})$ from the SNIS-fKL objective, and the ground truth labels $\boldsymbol{y}$ from SoftCVI are computed identically. This allows rewriting the SoftCVI objective from eq. (7) as the sum of the SNIS-fKL objective and the normalization term

$$\mathcal{L}_{\text{SoftCVI}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K) = \mathcal{L}_{\text{SNIS-fKL}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K) + \log \sum_{k=1}^K \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^-(\boldsymbol{\theta}_k)} + \text{const,} \tag{13}$$

where we show in appendix A.2 that when $p^-(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, the gradient of the normalization term is

$$\nabla_{\boldsymbol{\phi}} \log \sum_{k=1}^K \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^-(\boldsymbol{\theta}_k)} = \frac{1}{K} \sum_{k=1}^K \nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k). \tag{14}$$

Since $\mathbb{E}_{\pi(\boldsymbol{\theta})}[\nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})] = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta})}[\nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})] = \mathbf{0}$, the inclusion or omission of the normalization term does not change the gradient in expectation over sets of $\{\boldsymbol{\theta}_k\}_{k=1}^K$, but it significantly influences the variance. As shown in appendix A.1, the gradient variance for SoftCVI decreases to zero as the variational distribution approaches the posterior. In contrast, this implies the variance of the SNIS-fKL objective approaches the variance of $\frac{1}{K}\sum_{k=1}^K \nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)$, which is positive, scaling inversely with K.

In addition to providing a novel perspective of the SNIS-fKL objective as training an (unnormalized) classifier, this result also implies a straightforward modification to form a lower-variance SNIS-fKL gradient estimator

$$\mathcal{L}_{\text{SNIS-fKL-LV}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K) = \sum_{k=1}^K \tilde{w}(\boldsymbol{\theta}_k) \log \frac{p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)} + \frac{1}{K} \sum_{k=1}^K \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k), \tag{15}$$

which is exactly equivalent to optimizing the SoftCVI objective with $p^-(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$. This approach of lowering the variance of a gradient estimator by utilizing $\nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ as a control variate has also been applied to other variational objectives, such as the sticking the landing estimator of the ELBO (Roeder et al., 2017; Tucker et al., 2018). There is no guarantee that the gradient variance will be lower in all instances. However, the variance provably reduces to zero as the variational distribution approaches the true posterior, which allows a reasonable SNR to be maintained even when near convergence, which is likely beneficial for performance and stable convergence. We support this claim with the empirical results in section 5, in addition to investigating the signal-to-noise ratio of the objectives in appendix A.3.

3.3 CONTRASTIVE LEARNING

Contrastive learning most commonly allows learning of distributions through the comparison of true samples to a set of negative (noise or augmented) samples (Gutmann & Hyvärinen, 2010). Contrastive learning has been applied to numerous domains, including representation learning (Chen et al., 2020; He et al., 2020; Oord et al., 2018), image generation (Aneja et al., 2021) and natural language processing (Mnih & Kavukcuoglu, 2013). Generally, and notably in contrast to the current work, applications of contrastive learning focus on problems where the likelihood is unavailable, such as for fitting energy-based models (Gao et al., 2020; Gutmann & Hyvärinen, 2010; Gutmann et al., 2022; Rhodes & Gutmann, 2019; Rhodes et al., 2020) or performing simulation-based inference (Durkan et al., 2020; Greenberg et al., 2019; Gutmann et al., 2022; Hermans et al., 2020; Miller et al., 2022; Thomas et al., 2022).

Probably the most widely used objective function in contrastive learning is the InfoNCE loss, proposed by Oord et al. (2018). Given a set of samples $\{\boldsymbol{\theta}_k\}_{k=1}^K$, containing a single true sample with index k^*, and $K-1$ negative samples, the loss can be computed as

$$\mathcal{L}_{\text{InfoNCE}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K) = -\log \frac{f_{\boldsymbol{\phi}}(\boldsymbol{\theta}_{k^*})}{\sum_{k=1}^K f_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}. \tag{16}$$

where $f_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ approximates the ratio between the positive and negative distributions (up to a constant). This is commonly also presented with an additional sum (or expectation) over different sets of $\{\boldsymbol{\theta}_k\}_{k=1}^K$. The InfoNCE loss can be derived from the softmax cross-entropy loss, eq. (4), by inputting a one-hot encoded vector of labels $\boldsymbol{y}$ with $y_{k^*}=1$ and all other elements 0. This results in only the k^*-th summation term being non-zero, recovering the InfoNCE loss.

In contrastive methods for simulation-based inference, parameters are sampled from a proposal distribution $\pi(\boldsymbol{\theta})$, and used to perform simulations. Learning is achieved through comparing positive parameter-output pairs from $p(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})$, to negative (mismatched) pairs drawn marginally from $\pi(\boldsymbol{x})\pi(\boldsymbol{\theta})$, where $\pi(\boldsymbol{x}) = \int p(\boldsymbol{x}|\boldsymbol{\theta})\pi(\boldsymbol{\theta})d\boldsymbol{\theta}$. Similar to the current work, the classifier is often parameterized in terms of a normalized approximation to the posterior, such that after training the posterior approximation is directly available for sampling and density evaluation. In these methods, the classifier is parameterized using the ratio between the approximate posterior and the prior $q_{\boldsymbol{\phi}}(\boldsymbol{\theta}|\boldsymbol{x})/p(\boldsymbol{\theta})$, and optimized using the InfoNCE objective (Durkan et al., 2019; Greenberg et al., 2019).

Recently, there has been investigation of the use of soft (or ranked) labels in contrastive learning, frequently using cross-entropy-like loss functions. In contrast to the current work, these methods focus on improving performance in the standard context of contrastive learning, where evaluation of the likelihood is infeasible. As such, the soft labels cannot be computed exactly, and instead are generated using other methods, such as label smoothing (Hugger & Uhlmann, 2024) or by utilizing a similarity metric such as cosine similarity between embeddings of positive and negative samples (Feng & Patras, 2022; Hoffmann et al., 2022; Park et al., 2024).

4 EXPERIMENTS

We focus on Bayesian inference tasks for which reference posterior samples $\{\boldsymbol{\theta}_i^*\}_{i=1}^{N_{\text{ref}}}$ are available, which allows for reliable assessment of performance. However, in appendices A.4 and A.5, we also consider application of SoftCVI for training variational autoencoders and Bayesian neural networks, respectively. We use $p^-(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta})^\alpha$ and focus results on two choices, $\alpha = 0.75$ and $\alpha = 1$. We compare performance to variational inference either with the ELBO or the often more mass-covering SNIS-fKL divergence (Jerfel et al., 2021). For each task, we perform 50 independent runs with different random seeds. Where possible (i.e. an analytical posterior is available), we use a different observation $\boldsymbol{x}_{\text{obs}}$ generated from the model for each run. For tasks where the reference posterior is created through sampling methods we rely on reference posteriors provided by PosteriorDB (Magnusson et al., 2024) or SBI-Benchmark (Lueckmann et al., 2021) and for each run we sample from the available observations if multiple options are present. We run all considered methods for 50,000 optimization steps using the Adam optimizer (Kingma & Ba, 2014), and use $K=8$ samples from $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ during computation of the objectives.

Software. Our implementation and experiments made wide use of the python packages JAX (Bradbury et al., 2018), equinox (Kidger & Garcia, 2021), numpyro (Phan et al., 2019), flowjax (Ward, 2024) and optax (DeepMind et al., 2020).

4.1 METRICS

Coverage probabilities. Posterior coverage probabilities have been widely used to assess the reliability of posteriors (e.g. Cannon et al., 2022; Cook et al., 2006; Hermans et al., 2021; Prangle et al., 2014; Talts et al., 2018; Ward et al., 2022). Given a nominal frequency $\gamma \in [0, 1]$, the metric assesses the frequency at which true posterior samples fall within the $100\gamma\%$ highest posterior density region (credible region) of the approximate posterior. For a given γ, if the actual frequency exceeds γ, the posterior is conservative for that coverage probability; if it is lower, then the posterior is overconfident. A posterior is said to be well calibrated if the actual frequency matches γ for any choice of γ. A well calibrated or somewhat conservative posterior is needed for drawing reliable scientific conclusions, and as such is an important property to investigate. We estimate the actual coverage frequency for a posterior estimate as

$$\frac{1}{N_{\text{ref}}} \sum_{i=1}^{N_{\text{ref}}} \mathbb{1}\left\{\boldsymbol{\theta}_i^* \in \text{HDR}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})}(\gamma)\right\} \tag{17}$$

where $\mathbb{1}$ is the indicator function, and HDR represents the highest posterior density region, inferred using the density quantile approach of Hyndman (1996).

Log probability of $\boldsymbol{\theta}^*$. Looking at the probability of either reference posterior samples or the ground truth parameters in a posterior approximation is a common metric for assessing performance (e.g. Greenberg et al., 2019; Lueckmann et al., 2021; Papamakarios & Murray, 2016). We compute this independently for each posterior approximation as follows

$$\frac{1}{N_{\text{ref}}} \sum_{i=1}^{N_{\text{ref}}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i^*). \tag{18}$$

This metric also approximates the negative forward KL divergence between the true and approximate posterior up to a constant, which is a quantity known to control error in importance sampling (Chatterjee & Diaconis, 2018)

$$\begin{aligned} -D_{\text{KL}}[p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})||q_{\boldsymbol{\phi}}(\boldsymbol{\theta})] &= -\mathbb{E}_{p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})}[\log p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}}) - \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})], \\ &\approx \frac{1}{N_{\text{ref}}} \sum_{i=1}^{N_{\text{ref}}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_i^*) + \text{const}, \end{aligned} \tag{19}$$

Posterior mean accuracy. A posterior that is overconfident but with the correct posterior mean would perform poorly based on the aforementioned metrics, but can form good point estimates for the parameters $\boldsymbol{\theta}$. To assess this, we choose to measure the accuracy using the negative L^2-norm of the standardized difference in posterior means

$$-\left\| \frac{\text{mean}(\boldsymbol{\theta}^*) - \text{mean}(\boldsymbol{\theta})}{\text{std}(\boldsymbol{\theta}^*)} \right\|_2, \tag{20}$$

where $\text{mean}(\boldsymbol{\theta}^*)$ and $\text{mean}(\boldsymbol{\theta})$ are the mean vectors of the reference and posterior approximation samples respectively, and $\text{std}(\boldsymbol{\theta}^*)$ is the vector of standard deviations of the reference samples.

4.2 TASKS

A brief description of each model is given below. For a complete description, see appendix A.7.

Eight schools. A classic hierarchical Bayesian inference problem, where the aim is to infer the treatment effects of a coaching program applied to eight schools, which are assumed to be exchangeable (Gelman et al., 1995; Rubin, 1981). The parameter set is $\boldsymbol{\theta} = \{\mu, \tau, \boldsymbol{m}\}$, where μ is the average treatment effect across the schools, τ is the standard deviation of the treatment effects across the schools and $\boldsymbol{m}$ is the eight-dimensional vector of treatment effects for each school. For the posterior approximation $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, we use a normal distribution for μ, a folded Student's t distribution for τ (where folding is equivalent to taking an absolute value transform), and a Student's t distribution for $\boldsymbol{m}$.

Linear regression. A Bayesian linear regression model with parameters $\boldsymbol{\theta} = \{\boldsymbol{\beta}, \mu\}$, where $\boldsymbol{\beta} \in \mathbb{R}^{50}$ is the regression coefficients, and $\mu \in \mathbb{R}$ is the bias parameter. The covariates $\boldsymbol{X} \in \mathbb{R}^{(200\times 50)}$

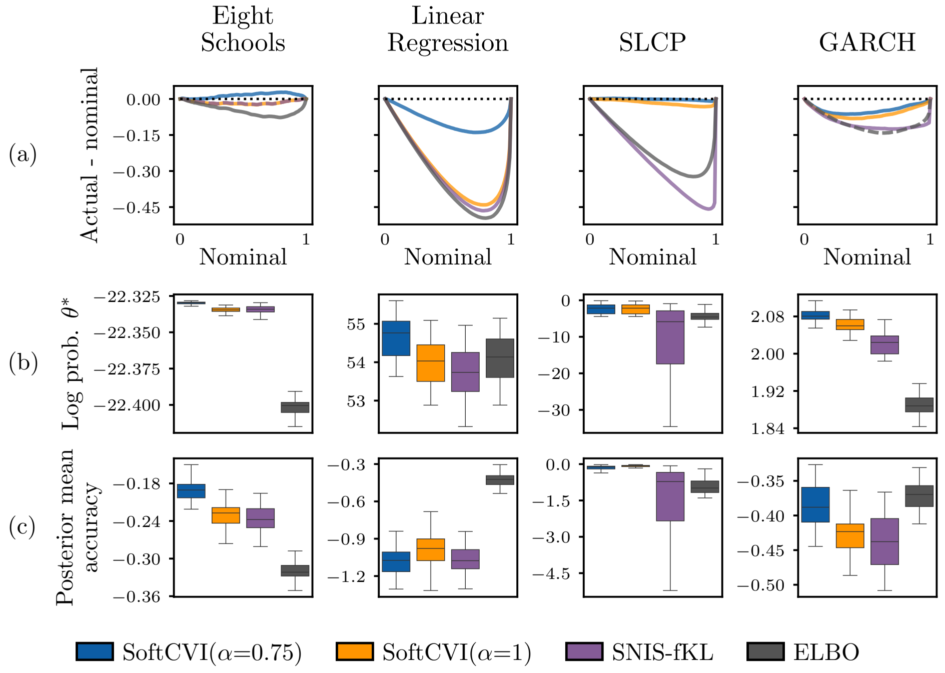

Figure 1: The posterior performance metrics (see section 4.1). a) The nominal coverage frequency against the average difference between the nominal and actual coverage frequency. Well-calibrated methods follow the black dotted line ($y = 0$), whereas conservative methods fall above, and over-confident methods below. b) The average probability of the reference posterior samples in the approximate posterior. c) The accuracy of the approximate posterior mean, calculated as the negative l2-norm between the mean of the standardized reference and approximate posterior samples.

are sampled from a standard normal distribution, with targets sampled from $\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{X}\boldsymbol{\beta} + \mu, 1)$. The posterior approximation $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ is implemented as a fully factorized normal distribution.

SLCP. The Simple Likelihood Complex Posterior (SLCP) task introduced in (Papamakarios et al., 2019). This task parameterizes a multivariate normal distribution using a 5-dimensional vector $\boldsymbol{\theta}$. Due to squaring in the parameterization, the posterior contains four symmetric modes. For the posterior approximation $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, we use a four layer masked autoregressive flow, with a rational quadratic spline transformer (Durkan et al., 2019; Germain et al., 2015; Kingma et al., 2016; Papamakarios et al., 2017).

GARCH(1,1). A Generalized Autoregressive Conditional heteroscedasticity (GARCH) model (Bollerslev, 1986). GARCH models are used for modeling the changing volatility of time series data by accounting for time-varying and autocorrelated variance in the error terms. The observation consists of a 200-dimensional time series $\boldsymbol{x}$, where each element x_t is drawn from a normal distribution with mean μ and time-varying variance σ_t^2. The variance σ_t^2 is defined recursively by the update $\sigma_t^2 = \alpha_1 + \alpha_2(x_{t-1} - \mu)^2 + \beta_1\sigma_{t-1}^2$, where α_2 and β_1 control the contribution from the previous observation and previous variance term, respectively. To parameterize $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, we use a normal distribution for μ and a log normal distribution for α_0. For α_1 and β_1 we use uniform distributions constrained to the prior support, transformed with a rational quadratic spline (Durkan et al., 2019). To allow modeling of posterior dependencies, the distribution over β_1 is parameterized as a function of α_1 and α_2 using a neural network.

5 RESULTS

Across all tasks and metrics considered, the SoftCVI derived objectives performed competitively with the ELBO and SNIS-fKL objectives (fig. 1). Overall, SoftCVI using a negative distribution

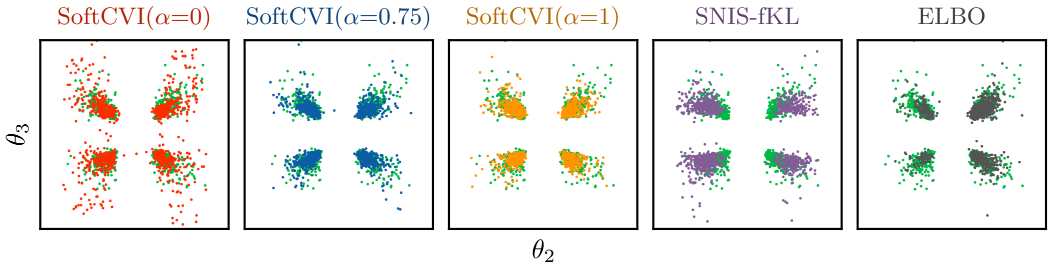

Figure 2: A 2-dimensional posterior marginal for a single run of the SLCP task, with the reference posterior samples shown in green.

$\pi(\boldsymbol{\theta})^{\alpha}$ with $\alpha = 0.75$ outperformed the other methods, giving rise to better calibrated posteriors and tending to place more mass on average on the reference posterior samples, indicating a lower forward KL divergence to the true posterior.

A key comparison, is between SoftCVI with $\alpha = 1$ and the SNIS-fKL objective, which give gradient approximators that are equal in expectation (see section 3.2). With the exception of the eight schools task, where both methods performed similarly, SoftCVI with $\alpha = 1$ tended to place more mass on the reference samples and yielded better calibrated posteriors. These results highlight the benefit of the reduced gradient variance provided by SoftCVI when the variational distribution is sufficiently close to the true posterior. Further, the performance discrepancy becomes more pronounced for a smaller choice of K (see figs. 7 and 8 and in the appendix). This can be explained due to the variance of the SNIS-fKL objective gradient scaling inversely with K, meaning the control variate naturally included by the SoftCVI objective becomes more crucial. Both SoftCVI and SNIS-fKL tended to place more mass on the reference samples when trained with a larger K, but this comes with an increase in computational cost (fig. 7).

On the SLCP task, which yields complex posterior geometry with four symmetric modes, only the SoftCVI objectives performed well. The ELBO objective often resulted in poor distribution of the mass across the modes, sometimes missing modes entirely (fig. 2; see also fig. 9 in the appendix). In contrast, SNIS-fKL, though less mode-seeking, frequently approximated the individual modes poorly. This observation aligns with previous work suggesting the unreliability of existing mass-covering objectives (Dhaka et al., 2021). To illustrate the impact of the choice of negative distribution, fig. 2 also shows a posterior trained with a flat negative distribution by setting $\alpha = 0$. The flat negative distribution does not sufficiently penalize placing mass in $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$ in regions of negligible posterior density, leading to "leakage" of mass (see section 2.3).

6 CONCLUSION

In this work, we introduced SoftCVI, a novel framework for deriving variational objectives motivated through contrastive learning. Our experiments across various Bayesian inference tasks indicate that SoftCVI often outperforms other variational objectives, producing posteriors approximations with better coverage properties and a lower forward KL divergence to the true posterior. The performance difference is particularly notable for tasks with complex posterior geometries which require flexible density estimators. SoftCVI bridges between variational inference and contrastive estimation, which we hope will open up new avenues for theoretical and experimental research. Theoretically, it would be interesting to explore if other variational objectives can be reframed using SoftCVI. Additionally, both experimental and theoretical work is needed to further guide the choice of negative distribution. Finally, it could be valuable to investigate how advances from classification and contrastive learning, such as label smoothing (Müller et al., 2019a) and temperature scaling (Wang & Liu, 2021), could be adapted to SoftCVI to enhance training stability and further control posterior calibration.

REFERENCES

Sergios Agapiou, Omiros Papaspiliopoulos, Daniel Sanz-Alonso, and Andrew M Stuart. Importance sampling: Intrinsic dimension and computational cost. *Statistical Science*, pp. 405–431, 2017.

Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. *Advances in neural information processing systems*, 34:4699–4711, 2021.

Jyoti Aneja, Alex Schwing, Jan Kautz, and Arash Vahdat. A contrastive learning approach for training variational autoencoder priors. *Advances in neural information processing systems*, 34: 480–493, 2021.

Michael Betancourt and Mark Girolami. Hamiltonian Monte Carlo for hierarchical models. *Current trends in Bayesian methodology with applications*, 79(30):2–4, 2015.

David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association*, 112(518):859–877, 2017.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In *International conference on machine learning*, pp. 1613–1622. PMLR, 2015.

Tim Bollerslev. Generalized autoregressive conditional heteroskedasticity. *Journal of econometrics*, 31(3):307–327, 1986.

James Bradbury, Roy Frostig, Peter Hawkins, Matthew James Johnson, Chris Leary, Dougal Maclaurin, George Necula, Adam Paszke, Jake VanderPlas, Skye Wanderman-Milne, and Qiao Zhang. JAX: composable transformations of Python+NumPy programs, 2018. URL `http://github.com/google/jax`.

Yuri Burda, Roger Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

Patrick Cannon, Daniel Ward, and Sebastian M Schmon. Investigating the impact of model misspecification in neural simulation-based inference. *arXiv preprint arXiv:2209.01845*, 2022.

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

Omar Chehab, Alexandre Gramfort, and Aapo Hyvärinen. The optimal noise in noise-contrastive learning is not what you think. In *Uncertainty in Artificial Intelligence*, pp. 307–316. PMLR, 2022.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Samantha R Cook, Andrew Gelman, and Donald B Rubin. Validation of software for Bayesian models using posterior quantiles. *Journal of Computational and Graphical Statistics*, 15(3):675–692, 2006.

DeepMind, Igor Babuschkin, Kate Baumli, Alison Bell, Surya Bhupatiraju, Jake Bruce, Peter Buchlovsky, David Budden, Trevor Cai, Aidan Clark, Ivo Danihelka, Antoine Dedieu, Claudio Fantacci, Jonathan Godwin, Chris Jones, Ross Hemsley, Tom Hennigan, Matteo Hessel, Shaobo Hou, Steven Kapturowski, Thomas Keck, Iurii Kemaev, Michael King, Markus Kunesch, Lena Martens, Hamza Merzic, Vladimir Mikulik, Tamara Norman, George Papamakarios, John Quan, Roman Ring, Francisco Ruiz, Alvaro Sanchez, Laurent Sartran, Rosalia Schneider, Eren Sezener, Stephen Spencer, Srivatsan Srinivasan, Miloš Stanojević, Wojciech Stokowiec, Luyu Wang, Guangyao Zhou, and Fabio Viola. The DeepMind JAX Ecosystem, 2020. URL `http://github.com/google-deepmind`.

Akash Kumar Dhaka, Alejandro Catalina, Manushi Welandawe, Michael R Andersen, Jonathan Huggins, and Aki Vehtari. Challenges and opportunities in high dimensional variational inference. *Advances in Neural Information Processing Systems*, 34:7787–7798, 2021.

Adji Bousso Dieng, Dustin Tran, Rajesh Ranganath, John Paisley, and David Blei. Variational inference via χ upper bound minimization. *Advances in Neural Information Processing Systems*, 30, 2017.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural spline flows. *Advances in neural information processing systems*, 32, 2019.

Conor Durkan, Iain Murray, and George Papamakarios. On contrastive learning for likelihood-free inference. In *International conference on machine learning*, pp. 2771–2781. PMLR, 2020.

Chen Feng and Ioannis Patras. Adaptive soft contrastive learning. In *2022 26th International Conference on Pattern Recognition (ICPR)*, pp. 2721–2727. IEEE, 2022.

Ruiqi Gao, Erik Nijkamp, Diederik P Kingma, Zhen Xu, Andrew M Dai, and Ying Nian Wu. Flow contrastive estimation of energy-based models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7518–7528, 2020.

Andrew Gelman and Xiao-Li Meng. Simulating normalizing constants: From importance sampling to bridge sampling to path sampling. *Statistical science*, pp. 163–185, 1998.

Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. *Bayesian data analysis*. Chapman and Hall/CRC, 1995.

Mathieu Germain, Karol Gregor, Iain Murray, and Hugo Larochelle. Made: Masked autoencoder for distribution estimation. In *International conference on machine learning*, pp. 881–889. PMLR, 2015.

Zoubin Ghahramani and Matthew Beal. Propagation algorithms for variational Bayesian learning. *Advances in neural information processing systems*, 13, 2000.

Manuel Glöckler, Michael Deistler, and Jakob H Macke. Variational methods for simulation-based inference. *arXiv preprint arXiv:2203.04176*, 2022.

Maria Gorinova, Dave Moore, and Matthew Hoffman. Automatic reparameterisation of probabilistic programs. In *International Conference on Machine Learning*, pp. 3648–3657. PMLR, 2020.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic posterior transformation for likelihood-free inference. In *International Conference on Machine Learning*, pp. 2404–2414. PMLR, 2019.

Michael Gutmann and Aapo Hyvärinen. Noise-contrastive estimation: A new estimation principle for unnormalized statistical models. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pp. 297–304. JMLR Workshop and Conference Proceedings, 2010.

Michael U Gutmann and Aapo Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. *Journal of machine learning research*, 13 (2), 2012.

Michael U Gutmann, Steven Kleinegesse, and Benjamin Rhodes. Statistical applications of contrastive learning. *Behaviormetrika*, 49(2):277–301, 2022.

Trevor Hastie. The elements of statistical learning: data mining, inference, and prediction, 2009.

W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9729–9738, 2020.

Joeri Hermans, Volodimir Begy, and Gilles Louppe. Likelihood-free MCMC with amortized approximate ratio estimators. In *International conference on machine learning*, pp. 4239–4248. PMLR, 2020.

Joeri Hermans, Arnaud Delaunoy, François Rozet, Antoine Wehenkel, Volodimir Begy, and Gilles Louppe. A trust crisis in simulation-based inference? Your posterior approximations can be unfaithful. *arXiv preprint arXiv:2110.06581*, 2021.

Matthew Hoffman, Pavel Sountsov, Joshua V Dillon, Ian Langmore, Dustin Tran, and Srinivas Vasudevan. Neutra-lizing bad geometry in Hamiltonian Monte Carlo using neural transport. *arXiv preprint arXiv:1903.03704*, 2019.

David T Hoffmann, Nadine Behrmann, Juergen Gall, Thomas Brox, and Mehdi Noroozi. Ranking info noise contrastive estimation: Boosting contrastive learning via ranked positives. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 897–905, 2022.

Johannes Hugger and Virginie Uhlmann. Noise contrastive estimation with soft targets for conditional models. *arXiv preprint arXiv:2404.14076*, 2024.

Rob J Hyndman. Computing and graphing highest density regions. *The American Statistician*, 50 (2):120–126, 1996.

Pavel Izmailov, Sharad Vikram, Matthew D Hoffman, and Andrew Gordon Gordon Wilson. What are bayesian neural network posteriors really like? In *International conference on machine learning*, pp. 4629–4640. PMLR, 2021.

Ghassen Jerfel, Serena Wang, Clara Wong-Fannjiang, Katherine A Heller, Yian Ma, and Michael I Jordan. Variational refinement for importance sampling using the forward Kullback-Leibler divergence. In *Uncertainty in Artificial Intelligence*, pp. 1819–1829. PMLR, 2021.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. An introduction to variational methods for graphical models. *Machine learning*, 37:183–233, 1999.

Patrick Kidger and Cristian Garcia. Equinox: neural networks in JAX via callable PyTrees and filtered transformations. *Differentiable Programming workshop at Neural Information Processing Systems 2021*, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Diederik P Kingma and Max Welling. Auto-encoding variational Bayes. In *Internation Conference on Learning Representations*, 2014.

Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.

Yingzhen Li and Richard E Turner. Rényi divergence variational inference. *Advances in neural information processing systems*, 29, 2016.

Jan-Matthis Lueckmann, Jan Boelts, David Greenberg, Pedro Gonçalves, and Jakob Macke. Benchmarking simulation-based inference. In *International conference on artificial intelligence and statistics*, pp. 343–351. PMLR, 2021.

Måns Magnusson, Jakob Torgander, Paul-Christian Bürkner, Lu Zhang, Bob Carpenter, and Aki Vehtari. posteriordb: Testing, benchmarking and developing bayesian inference algorithms. *arXiv preprint arXiv:2407.04967*, 2024.

Nicholas Metropolis, Arianna W Rosenbluth, Marshall N Rosenbluth, Augusta H Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The journal of chemical physics*, 21(6):1087–1092, 1953.

Benjamin K Miller, Christoph Weniger, and Patrick Forré. Contrastive neural ratio estimation. *Advances in Neural Information Processing Systems*, 35:3262–3278, 2022.

Andriy Mnih and Koray Kavukcuoglu. Learning word embeddings efficiently with noise-contrastive estimation. *Advances in neural information processing systems*, 26, 2013.

Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When does label smoothing help? *Advances in neural information processing systems*, 32, 2019a.

Thomas Müller, Brian McWilliams, Fabrice Rousselle, Markus Gross, and Jan Novák. Neural importance sampling. *ACM Transactions on Graphics (ToG)*, 38(5):1–19, 2019b.

Christian Naesseth, Fredrik Lindsten, and David Blei. Markovian score climbing: Variational inference with KL $(p||q)$. *Advances in Neural Information Processing Systems*, 33:15499–15510, 2020.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

George Papamakarios and Iain Murray. Fast ε-free inference of simulation models with Bayesian conditional density estimation. *Advances in neural information processing systems*, 29, 2016.

George Papamakarios, Theo Pavlakou, and Iain Murray. Masked autoregressive flow for density estimation. *Advances in neural information processing systems*, 30, 2017.

George Papamakarios, David Sterratt, and Iain Murray. Sequential neural likelihood: Fast likelihood-free inference with autoregressive flows. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 837–848. PMLR, 2019.

Omiros Papaspiliopoulos, Gareth O Roberts, and Martin Sköld. A general framework for the parametrization of hierarchical models. *Statistical Science*, pp. 59–73, 2007.

Giorgio Parisi. *Statistical Field Theory*. Addison-Wesley, 1988.

Minsu Park, Seyeon Choi, Chanyeol Choi, Jun-Seong Kim, and Jy-yong Sohn. Improving multilingual alignment through soft contrastive learning. *arXiv preprint arXiv:2405.16155*, 2024.

Du Phan, Neeraj Pradhan, and Martin Jankowiak. Composable effects for flexible and accelerated probabilistic programming in numpyro. *arXiv preprint arXiv:1912.11554*, 2019.

Dennis Prangle, Michael GB Blum, Gordana Popovic, and SA Sisson. Diagnostic tools for approximate Bayesian computation using the coverage property. *Australian & New Zealand Journal of Statistics*, 56(4):309–329, 2014.

Tom Rainforth, Adam Kosiorek, Tuan Anh Le, Chris Maddison, Maximilian Igl, Frank Wood, and Yee Whye Teh. Tighter variational bounds are not necessarily better. In *International Conference on Machine Learning*, pp. 4277–4285. PMLR, 2018.

Benjamin Rhodes and Michael U Gutmann. Variational noise-contrastive estimation. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2741–2750. PMLR, 2019.

Benjamin Rhodes, Kai Xu, and Michael U Gutmann. Telescoping density-ratio estimation. *Advances in neural information processing systems*, 33:4905–4916, 2020.

Geoffrey Roeder, Yuhuai Wu, and David K Duvenaud. Sticking the landing: Simple, lower-variance gradient estimators for variational inference. *Advances in Neural Information Processing Systems*, 30, 2017.

Donald B Rubin. Estimation in parallel randomized experiments. *Journal of Educational Statistics*, 6(4):377–401, 1981.

Donald B Rubin. Using the sir algorithm to simulate posterior distributions. In *Bayesian statistics 3. Proceedings of the third Valencia international meeting, 1-5 June 1987*, pp. 395–402. Clarendon Press, 1988.

Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. Validating Bayesian inference algorithms with simulation-based calibration. *arXiv preprint arXiv:1804.06788*, 2018.

Owen Thomas, Ritabrata Dutta, Jukka Corander, Samuel Kaski, and Michael U Gutmann. Likelihood-free inference by ratio estimation. *Bayesian Analysis*, 17(1):1–31, 2022.

George Tucker, Dieterich Lawson, Shixiang Gu, and Chris J Maddison. Doubly reparameterized gradient estimators for Monte Carlo objectives. *arXiv preprint arXiv:1810.04152*, 2018.

Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2495–2504, 2021.

Daniel Ward. Flowjax: Distributions and normalizing flows in jax, 2024. URL `https://github.com/danielward27/flowjax`. version 14.0.0.

Daniel Ward, Patrick Cannon, Mark Beaumont, Matteo Fasiolo, and Sebastian Schmon. Robust neural posterior estimation and statistical model criticism. *Advances in Neural Information Processing Systems*, 35:33845–33859, 2022.

Jiayu Yao, Weiwei Pan, Soumya Ghosh, and Finale Doshi-Velez. Quality of uncertainty quantification for bayesian neural network inference. *arXiv preprint arXiv:1906.09686*, 2019.

Yuling Yao, Aki Vehtari, Daniel Simpson, and Andrew Gelman. Yes, but did it work?: Evaluating variational inference. In *International Conference on Machine Learning*, pp. 5581–5590. PMLR, 2018.

A APPENDIX

A.1 ZERO VARIANCE GRADIENT AT OPTIMUM WITH FINITE K

When the optimal classifier is reached, we can show that the gradient is zero (and hence has zero variance), even for finite K. To avoid clashes with the parameters $\boldsymbol{\phi}$, we will use superscripts for indices. We have the objective

$$\mathcal{L}(\boldsymbol{\phi}; \{\boldsymbol{\theta}^k\}_{k=1}^K, \boldsymbol{y}) = -\sum_{k=1}^{K} y^k \log \left(\frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}^k)/p^-(\boldsymbol{\theta}^k)}{\sum_{k'=1}^K q_{\boldsymbol{\phi}}(\boldsymbol{\theta}^{k'})/p^-(\boldsymbol{\theta}^{k'})} \right).$$

Letting $\hat{y}^k_{\boldsymbol{\phi}}$ replace the term inside the log

$$= -\sum_{k=1}^{K} y^k \log \hat{y}^k_{\boldsymbol{\phi}}, \tag{21}$$

$$\nabla \mathcal{L}(\boldsymbol{\phi}; \{\boldsymbol{\theta}^k\}_{k=1}^K, \boldsymbol{y}) = -\sum_{k=1}^{K} y^k \nabla \log \hat{y}^k_{\boldsymbol{\phi}}, \tag{22}$$

$$= -\sum_{k=1}^{K} y^k \frac{\nabla \hat{y}^k_{\boldsymbol{\phi}}}{\hat{y}^k_{\boldsymbol{\phi}}}, \tag{23}$$

which due to optimality of the classifier we have $y^k = \hat{y}^k_{\boldsymbol{\phi}}$

$$= -\sum_{k=1}^{K} \nabla \hat{y}^k_{\boldsymbol{\phi}}, \tag{24}$$

due to the properties of the softmax, the labels must sum to 1

$$= -\nabla \sum_{k=1}^{K} \hat{y}^k_{\boldsymbol{\phi}} = \boldsymbol{0}. \tag{25}$$

A.2 SOFTCVI NORMALIZATION TERM GRADIENT

When $p^-(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, we claim in eq. (14) that the normalization term gradient can be written as $\frac{1}{K}\sum_{k=1}^K \nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)$. We can show this as follows

$$\nabla_{\boldsymbol{\phi}} \log \sum_{k=1}^{K} \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^-(\boldsymbol{\theta}_k)} = \frac{\nabla_{\boldsymbol{\phi}} \sum_{k=1}^K q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^-(\boldsymbol{\theta}_k)}{\sum_{k=1}^K q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^-(\boldsymbol{\theta}_k)},$$

using $p^-(\boldsymbol{\theta}) = q_{\boldsymbol{\phi}}(\boldsymbol{\theta})$, we have $\sum_{k=1}^K q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^-(\boldsymbol{\theta}_k) = K$, giving

$$= \frac{\nabla_{\boldsymbol{\phi}} \sum_{k=1}^K q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^-(\boldsymbol{\theta}_k)}{K},$$

$$= \frac{1}{K}\sum_{k=1}^{K} \frac{\nabla_{\boldsymbol{\phi}} q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^-(\boldsymbol{\theta}_k)},$$

$$= \frac{1}{K}\sum_{k=1}^{K} \frac{\nabla_{\boldsymbol{\phi}} q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)},$$

$$= \frac{1}{K}\sum_{k=1}^{K} \nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k), \tag{26}$$

where on the first and last line we make use of $\nabla_{\boldsymbol{x}} \log f(\boldsymbol{x}) = \frac{\nabla_{\boldsymbol{x}} f(\boldsymbol{x})}{f(\boldsymbol{x})}$.

A.3 GRADIENT SIGNAL-TO-NOISE RATIO

By evaluating an objective gradient over a set of random seeds (different sets of $\{\boldsymbol{\theta}_k\}_{k=1}^K$), it is possible to empirically inspect the signal-to-noise ratio (SNR) ratio of the gradient for different objectives. Following Rainforth et al. (2018), we compute the SNR as

$$\text{SNR}(\nabla_{\boldsymbol{\phi}}\mathcal{L}(\boldsymbol{\phi})) = |\mathbb{E}[\nabla_{\boldsymbol{\phi}}\mathcal{L}(\boldsymbol{\phi})]| \ /\sigma[\nabla_{\boldsymbol{\phi}}\mathcal{L}(\boldsymbol{\phi})]$$

where $\sigma[\cdot]$ denotes the element-wise standard deviation for each parameter gradient. A low SNR indicates that the gradient estimation is dominated by noise, making stochastic optimization challenging. However, it is important to recognize that whilst a high SNR is preferable, it does not alone guarantee the objective itself is practically useful (or even sensible), for example it may be biased or heavily favor mode-seeking solutions.

We consider a toy normal task from Glöckler et al. (2022), with the distinction that we vary the dimensionality of the task $d \in \{1, 50\}$ and the parameterization of variational distribution. The task is defined through the model

$$\boldsymbol{\theta} \sim \mathcal{N}(\mathbf{0}, 4 \cdot \boldsymbol{I}_d), \quad \boldsymbol{x} \sim \mathcal{N}(\boldsymbol{\theta}, \boldsymbol{I}_d),$$

where the aim is to infer the mean vector $\boldsymbol{\theta}$, given an observation $\boldsymbol{x}_{\text{obs}} = \mathbf{1}_d$, where $\mathbf{1}_d$ is the d-dimensional vector of ones. The variational distribution is parameterized as a normal distribution, with mean vector $\boldsymbol{\mu} \in \mathbb{R}^d$ and log standard deviations $\log \boldsymbol{\sigma} \in \mathbb{R}^d$. When assessing the gradient properties of $\boldsymbol{\mu}$, we hold $\log \boldsymbol{\sigma}$ fixed at the closed form posterior solution, $\log \boldsymbol{\sigma} = \log(\sqrt{4/5}) \cdot \mathbf{1}_d$. Similarly, when assessing the gradient properties of $\log \boldsymbol{\sigma}$, we hold $\boldsymbol{\mu}$ fixed at the true closed form solution $\boldsymbol{\mu} = 4/5 \cdot \mathbf{1}_d$. The signal, noise and SNR for each objective are shown in fig. 3.

SoftCVI with $\alpha = 1$, and the SNIS-fKL objective yield very similar signals, which results from the two objectives producing the same gradients in expectation (section 3.2). However, the SNIS-fKL

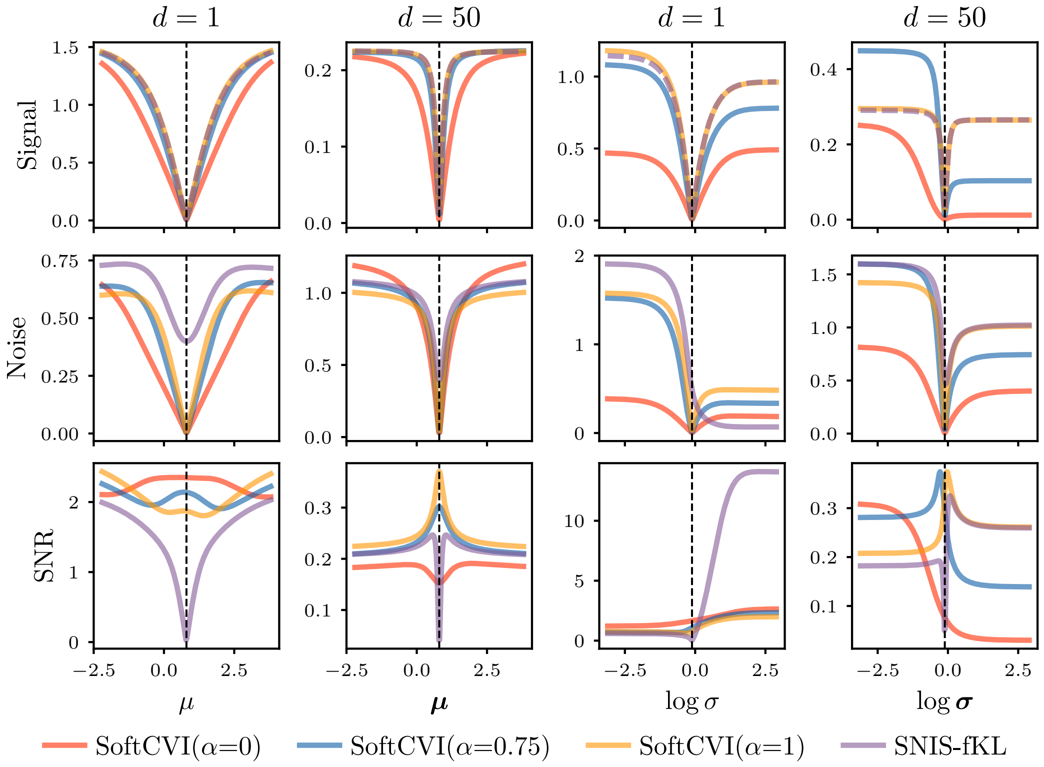

Figure 3: The signal, noise and signal-to-noise ratio of the objective gradients on a toy normal task of varying dimensionality. When $d = 50$, the gradient properties are computed parameter-wise and averaged. The vertical dashed line shows the true parameter values from the closed form posterior solution.

objective has positive noise when the variational distribution approaches the true posterior, meaning the SNR degrades to zero. In contrast, for the SoftCVI objectives, the gradient noise approaches zero (see appendix A.1), and a reasonable SNR is present as the variational distribution approaches the true posterior.

When $d = 50$ the signal and consequently the SNR, deteriorates larger values of $\log \boldsymbol{\sigma}$, and lower choices of α. In this case, both expanding the dimension of the problem, and decreasing the α value, increases the variance of $z_k = \log[p(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})/p^-(\boldsymbol{\theta}_k)]$ and $\hat{z}_k = \log[q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)/p^-(\boldsymbol{\theta}_k)]$. This can lead to degeneracy in the labels and predictions, meaning very few samples meaningfully contribute to the loss function, reducing the SNR. In an extreme case, e.g. setting $\alpha = 0$ and further expanding the dimensionality of the problem, only a single label and prediction will be significantly non-zero. In this regime, negligible signal would exist for the $\log \boldsymbol{\sigma}$ parameter, as the variational distribution, despite the incorrect $\log \boldsymbol{\sigma}$, would still result in the correct degenerate predicted labels.

A.4 TRAINING OF MODEL PARAMETERS: VARIATIONAL AUTOENCODERS

SoftCVI sets up a classification problem which allows fitting the parameters of the variational distribution $\boldsymbol{\phi}$. However, in some contexts, the model may be defined as $p_{\boldsymbol{\psi}}(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})$, with $\boldsymbol{\psi}$, being a set of model parameters which we wish to optimize in some manner alongside $\boldsymbol{\phi}$. SoftCVI does not directly provide a method for fitting such additional model parameters. One approach to resolve this is to include an additional objective, that trains $\boldsymbol{\psi}$ to (approximately) maximize the marginal likelihood.

$$\mathcal{L}(\boldsymbol{\phi}, \boldsymbol{\psi}; \{\boldsymbol{\theta}_k\}_{k=1}^K, \boldsymbol{y}) = \mathcal{L}_{\text{SoftCVI}}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^K, \boldsymbol{y}) + \mathcal{L}_{\text{model}}(\boldsymbol{\psi}; \{\boldsymbol{\theta}_k\}_{k=1}^K). \tag{27}$$

where in the above formulation we reuse the proposal distribution samples $\{\boldsymbol{\theta}_k\}_{k=1}^K$ already sampled for the SoftCVI objective, and we assume the proposal distribution is equal to the variational distribution. One could more generally consider alternating between optimizing the two objectives; however, due to the disjoint parameter sets between the objectives, in addition to the invariance of common optimizers such a Adam to diagonal rescaling of gradient elements (Kingma & Ba, 2014), we found adding the objectives to work well. In this section, we choose

$$\mathcal{L}_{\text{model}}(\boldsymbol{\psi}; \{\boldsymbol{\theta}_k\}_{k=1}^K) = -\frac{1}{K}\sum_{k=1}^{K} \log p_{\boldsymbol{\psi}}(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}}), \tag{28}$$

equivalent to training the model parameters $\boldsymbol{\psi}$ by maximizing the ELBO. As an example, we will train a variational autoencoder (Kingma & Welling, 2014), where $\boldsymbol{\psi}$ includes the parameters of the decoder in addition to the models prior parameters, and $\boldsymbol{\phi}$ consists of the parameters of the encoder. Both the encoder and decoder are parameterized as standard feed-forward networks. The manifolds

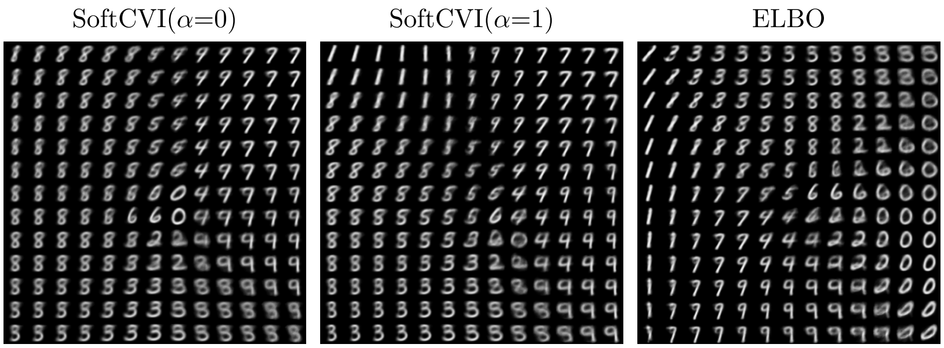

Figure 4: The manifolds learned by variational autoencoders on the MNIST dataset, trained using either the ELBO or SoftCVI. To enable training of the model parameters, the SoftCVI objective was modified by adding the model component of the ELBO, $-\frac{1}{K}\sum_{i=1}^{K} \log p_{\boldsymbol{\psi}}(\boldsymbol{\theta}_k, \boldsymbol{x}_{\text{obs}})$. In all cases, the objectives were trained for 100,000 steps with a batch size of 1, and $K = 8$.

learned using either the ELBO or the modified SoftCVI objective from eq. (27) are shown in fig. 4, showing qualitatively similar results. Whilst we do not investigate the performance, the SoftCVI method of fitting does not utilize reparameterized gradients and as such is applicable to a broader class of models.

A.5 BAYESIAN NEURAL ADDITIVE MODEL

Neural additive models enhance the interpretability of neural networks, using an additive model structure (Agarwal et al., 2021). In the simplest case, for a dataset with C features, for an input vector $\boldsymbol{x} = x_1, ..., x_C$, predictions are computed as

$$\hat{y} = \beta + f_1(x_1) + f_2(x_2) + \cdots + f_C(x_C),$$

where each f_c is a neural network corresponding to a single input feature. In this section, we consider a regression problem, in which we parameterize each f_c using a Bayesian neural network (Blundell et al., 2015). Bayesian neural networks have been suggested to reduce the risk of overfitting in addition to providing a method for assessing prediction uncertainty using the inferred posterior predictive distribution. Here, we parameterize each f_c using a single layer Bayesian neural network with a width of either 50 or 100, with a Laplace$(0, 1)$ prior on the neural network parameters. This naturally leads to high dimensional posterior distributions ($\boldsymbol{\theta} \in \mathbb{R}^{1500}$ and $\boldsymbol{\theta} \in \mathbb{R}^{3000}$ for the task considered, respectively). We use an independent Gaussian approximation to the posterior.

We consider a regression problem, with synthetic data generated using the nonlinear function

$$y = 0.1 \cdot x_1^3 + |x_2| - x_3 + \epsilon, \qquad \text{where } \epsilon \sim \mathcal{N}(0, 3^2)$$

where $\boldsymbol{x} \in \mathbb{R}^{10}$ is generated uniformly from the interval $[-4, 4]$, and the last 7 dimensions are nuisance features. We use 300 training data points, 150 validation data points and 1000 testing data points for computing the metrics.

We assess performance using the average test set log-likelihood under the posterior predictive distribution, and report the mean prediction interquartile range (IQR) across the test set and the associated prediction coverage (i.e. the average frequency with which the true underlying function value is included within the predicted IQR). We note there are numerous challenges associated with assessing performance for Bayesian neural networks. For example, high test log-likelihood or calibrated predictions is not necessarily indicative of a good posterior approximation (Yao et al., 2019). Further,

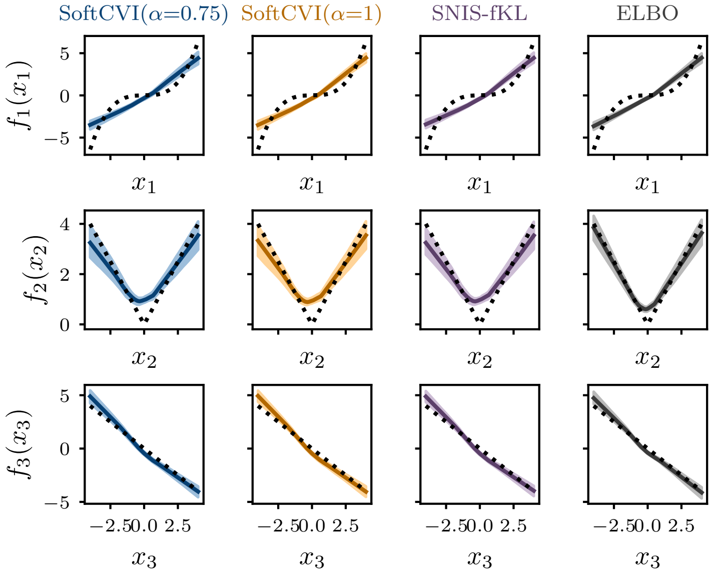

Figure 5: The means and 95% prediction intervals for the components of a Bayesian neural additive model for each method. The true underlying components are shown with the dotted black lines. We restrict to the first three dimensions, ignoring the nuisance variables.

we use early stopping based on the validation log-likelihood, in addition to choosing the learning rates with cross-validation, both of which will tend to bias results to favor models with higher test log-likelihood, without consideration of the calibration of the posterior predictive distribution.

We train 10 networks initialized with different random seeds, and report the results in table 1. A plot of the learned components for each method is shown in fig. 5.

Method	NN Width	Test Log-Likelihood	Prediction IQR	IQR Coverage
ELBO	50	**-2.392** ± 0.005	0.648 ± 0.084	0.262 ± 0.042
	100	**-2.392** ± 0.005	0.641 ± 0.065	0.264 ± 0.031
SNIS-fKL	50	-2.422 ± 0.007	0.695 ± 0.011	0.186 ± 0.008
	100	-2.421 ± 0.005	0.938 ± 0.010	0.256 ± 0.007
SoftCVI ($\alpha = 0.75$)	50	-2.424 ± 0.007	0.743 ± 0.018	0.202 ± 0.009
	100	-2.423 ± 0.005	0.983 ± 0.017	0.271 ± 0.010
SoftCVI ($\alpha = 1$)	50	-2.422 ± 0.007	0.701 ± 0.012	0.190 ± 0.009
	100	-2.423 ± 0.005	0.943 ± 0.012	0.260 ± 0.009

Table 1: Performance metrics for the Bayesian neural additive model. Note a model producing calibrated predictions on the test set would yield an IQR coverage value of 0.5.

While all methods demonstrated comparable predictive performance, the ELBO achieved slightly better results, evidenced by the highest test log-likelihood. Both SoftCVI (with $\alpha = 0.75$ or $\alpha = 1$) and SNIS-fKL yielded similar results. Bayesian neural network posteriors are often complex and highly multimodal (Izmailov et al., 2021). In this case, it is likely that SoftCVI does not show a significant advantage over SNIS-fKL because the Gaussian posterior is heavily misspecified. As a result, the approximate posterior may never become sufficiently close to the true posterior in order to provide the variance reduction benefits associated with SoftCVI.

A.6 ALTERNATIVE NEGATIVE DISTRIBUTION CHOICES

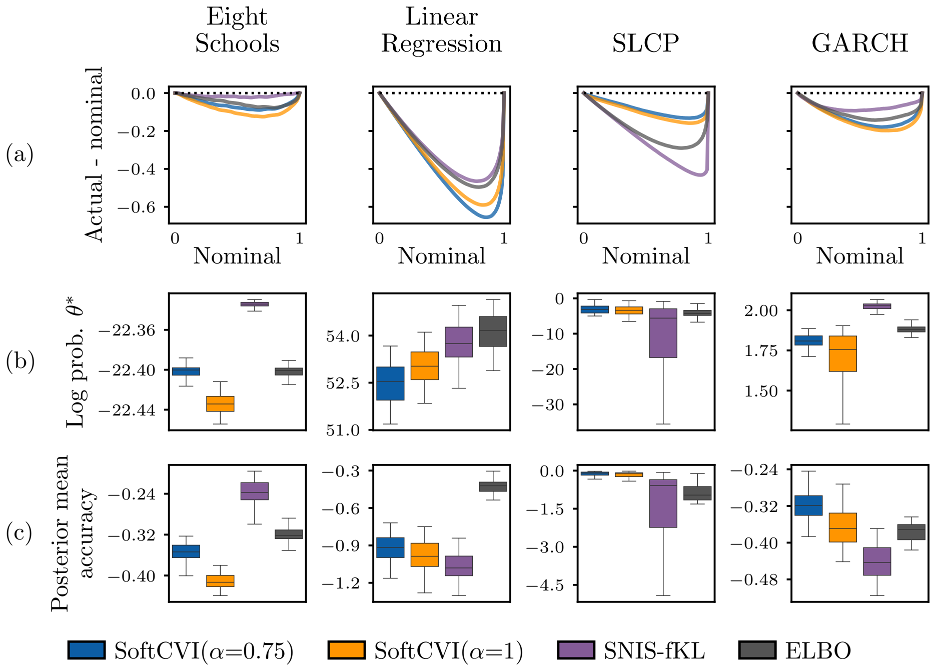

Figure 6: Additional metrics analogous to fig. 1, using $p^-(\boldsymbol{\theta}) = p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})^\alpha$ as the negative distribution choice in the SoftCVI objectives.

The main results focus on parameterizing the negative distribution as a function of the proposal distribution $p^{-}(\boldsymbol{\theta}) = \pi(\boldsymbol{\theta})^{\alpha}$. However, another possible choice is to use the unnormalized posterior to parameterize the negative distribution, for example $p^{-}(\boldsymbol{\theta}) = p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})^{\alpha}$. This choice, when $\alpha = 1$, implies equality between the assumed negative and positive distributions, meaning through eq. (2) the ground truth labels become constant $y_k = \frac{1}{k}$ for $k = 1, ..., K$. Inputting these labels into eq. (7) yields the objective

$$\mathcal{L}(\boldsymbol{\phi}; \{\boldsymbol{\theta}_k\}_{k=1}^{K}, \boldsymbol{y}) = -\frac{1}{K} \sum_{k=1}^{K} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k) + \log \sum_{k=1}^{K} \frac{q_{\boldsymbol{\phi}}(\boldsymbol{\theta}_k)}{p^{-}(\boldsymbol{\theta}_k)} + \text{const.} \tag{29}$$

Assuming the proposal distribution is chosen to match the variational distribution, since $\mathbb{E}_{\pi(\boldsymbol{\theta})}[\nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})] = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{\theta})}[\nabla_{\boldsymbol{\phi}} \log q_{\boldsymbol{\phi}}(\boldsymbol{\theta})] = 0$, the first term in eq. (7) has an expected gradient of zero. In contrast to using the proposal distribution as the negative distribution, which results in the normalization term which penalizes the ratios acting as a control variate, here, the first term instead acts as a control variate, with the normalization term providing the signal by penalizing the ratio $q_{\boldsymbol{\phi}}(\boldsymbol{\theta})/p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}}) \propto q_{\boldsymbol{\phi}}(\boldsymbol{\theta})/p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$. As such this choice heavily penalizes $q_{\boldsymbol{\phi}}(\boldsymbol{\theta}) \gg p(\boldsymbol{\theta}|\boldsymbol{x}_{\text{obs}})$, favoring overconfident posteriors. We report in fig. 6 the metrics, using $p^{-}(\boldsymbol{\theta}) = p(\boldsymbol{\theta}, \boldsymbol{x}_{\text{obs}})^{\alpha}$, with $\alpha = 0.75$ and $\alpha = 1$.

A.7 TASKS

For all tasks, where possible, we make use of reparameterizations to reduce the dependencies between the parameters $\boldsymbol{\theta}$ in the model, and to ensure variables are reasonably scaled, which is generally considered to improve performance (Betancourt & Girolami, 2015; Gorinova et al., 2020; Papaspiliopoulos et al., 2007). Below, we describe the models used and the source of the reference posterior samples.

EIGHT SCHOOLS.

A classic hierarchical inference model (Gelman et al., 1995; Rubin, 1981), aiming to infer the treatment effects of a training program applied to eight schools. The parameter set is $\boldsymbol{\theta} = \{\mu, \tau, \boldsymbol{m}\}$, where μ is the average treatment effect across the schools, τ is the standard deviation of the treatment effects across the schools and $\boldsymbol{m}$ is the treatment effects for each school. The model is given by

$$\begin{aligned}
\mu &\sim \mathcal{N}(0, 5^2), \\
\tau &\sim \text{HalfCauchy}(0, 5^2), \\
m_i &\sim \mathcal{N}(\mu, \tau^2), \quad i = 1, ..., 8, \\
x_i &\sim \mathcal{N}(m_i, \sigma_i^2), \quad i = 1, ..., 8,
\end{aligned}$$

where $\boldsymbol{\sigma}^2$ is treated as known, estimated using the standard errors in the data. We use the reference posterior samples available from PosteriorDB, which are sampled using MCMC (Magnusson et al., 2024).

LINEAR REGRESSION.

A linear regression model, defined as

$$\begin{aligned}
\beta_i &\sim \mathcal{N}(0, 1), \quad i = 1, ..., 50, \\
\mu &\sim \mathcal{N}(0, 1), \\
x_i &\sim \mathcal{N}(\boldsymbol{X}\boldsymbol{\beta} + \mu, 1), \quad j = 1, ..., 200,
\end{aligned}$$

For each run of the task, we sampled a dataset $\boldsymbol{X} \in \mathbb{R}^{200 \times 50}$ from a standard normal distribution, and drew reference posterior samples from the analytical posterior solution.

SIMPLE LIKELIHOOD COMPLEX POSTERIOR

The SLCP task was introduced by Papamakarios et al. (2019), and is designed to be a challenging inference problem with a multimodal posterior. The data is a set of four samples from a two-dimensional multivariate Gaussian likelihood function. The likelihood is parameterized using $\boldsymbol{\theta}$

using squaring which introduces a complex multimodal structure fig. 2. Specifically, the model is defined as

$$\theta_i \sim \text{Uniform}(-3, 3), \quad i = 1, ..., 5 \tag{30}$$

$$\boldsymbol{x}_i \sim \mathcal{N}(\boldsymbol{\theta}_{1:2}, \boldsymbol{\Sigma}), \quad j = 1, ..., 4, \tag{31}$$

where the covariance matrix $\boldsymbol{\Sigma}$ is

$$\boldsymbol{\Sigma} = \begin{bmatrix} \mathrm{s}_1^2 & p \cdot \mathrm{s}_1 \cdot \mathrm{s}_2 \\ p \cdot \mathrm{s}_1 \cdot \mathrm{s}_2 & \mathrm{s}_2^2 \end{bmatrix}, \text{ where } s_1 = \theta_3^2, \; s_2 = \theta_4^2 \text{ and } p = \tanh(\theta_5)$$

Despite being used extensively in the simulation-based inference (SBI) literature, this task has a tractable likelihood so can be used in the current work. The reference posterior is from the SBI Benchmark python package (Lueckmann et al., 2021), and was inferred using sampling/importance resampling using the analytical likelihood function (Rubin, 1988).

GARCH

The GARCH model is a widely used statistical model for analyzing time series data with time-varying volatility (Bollerslev, 1986). It extends the basic autoregressive framework by allowing the conditional variance of the observations to depend on both past observations (controlled via the α_2 parameter) and variances (controlled by the β_1 parameter). GARCH and similar models are often used for modeling financial data, where autocorrelated variances are common. The priors are defined as

$$\begin{aligned} \mu &\sim \text{ImproperUniform}(-\infty, \infty) \\ \alpha_1 &\sim \text{ImproperUniform}(0, \infty) \\ \alpha_2 &\sim \text{Uniform}(0, 1) \\ \beta_1 &\sim \text{Uniform}(0, 1 - \alpha_1), \end{aligned}$$

Where ImproperUniform represents an improper flat prior over the specified region. For $t = 1, \ldots, 200$, the variance evolves recursively through the update

$$\sigma_t^2 = \alpha_1 + \alpha_2(x_{t-1} - \mu)^2 + \beta_1 \sigma_{t-1}^2$$

and the likelihood is given by

$$x_t \sim \mathcal{N}(\mu, \sigma_t)$$

At initialization, y_0 is set to the first observation element, and $\sigma_0^2 = 0.25$. For this task, a reference posterior is available in PosteriorDB, sampled using MCMC (Magnusson et al., 2024).

A.8 ADDITIONAL FIGURES

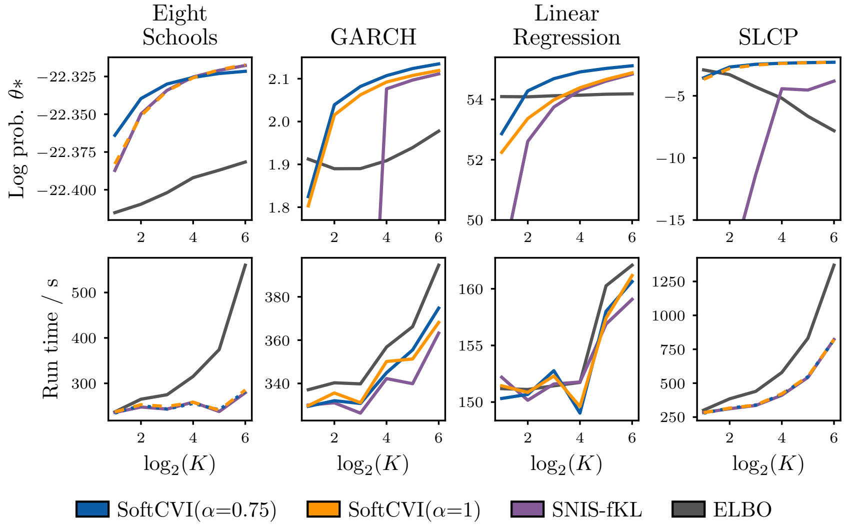

Figure 7: Average log-probability of the reference posterior samples as a function of K (ranging from 2 to 64), along with the associated run times (measured on a CPU with 8GB RAM, including compilation time). The poor performance of SLCP for higher values of K was due to increased mode-seeking behavior. Some results for SNIS-fKL are truncated on the axes to improve visualization of other methods. Note that the run times will be dependent on the architecture of the variational distribution. For example, the ELBO showed significantly slower run times for the SLCP task due to the cost of computing reparameterization gradients for a masked autoregressive flow. However, using an inverse autoregressive flow (Kingma et al., 2016) would likely mitigate this issue.

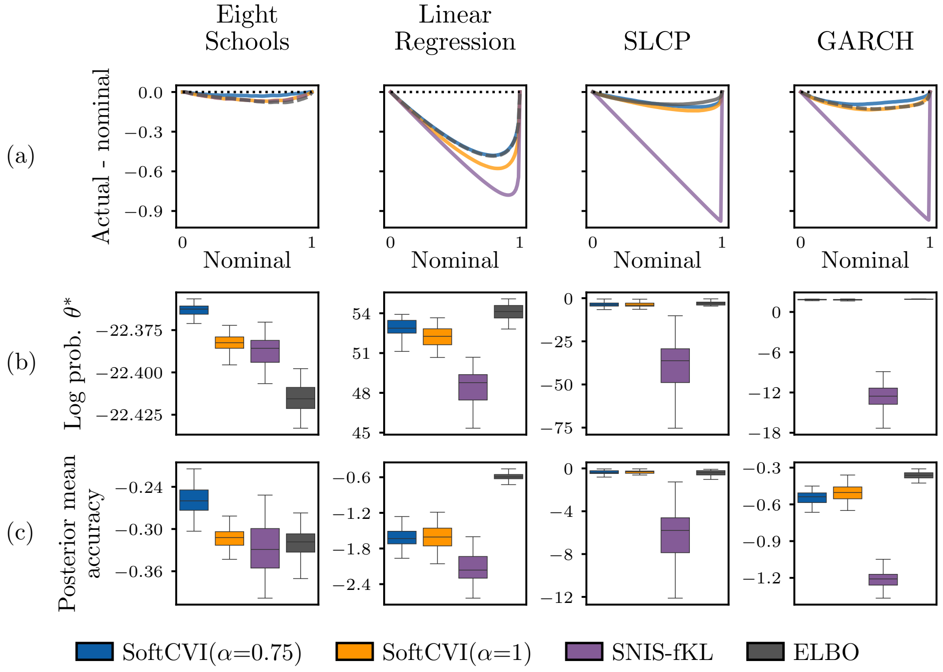

Figure 8: Additional metrics analogous to fig. 1, using $K = 2$, instead of $K = 8$ samples when approximating the objectives. The SNIS-fKL objective performance degrades substantially when K is small.

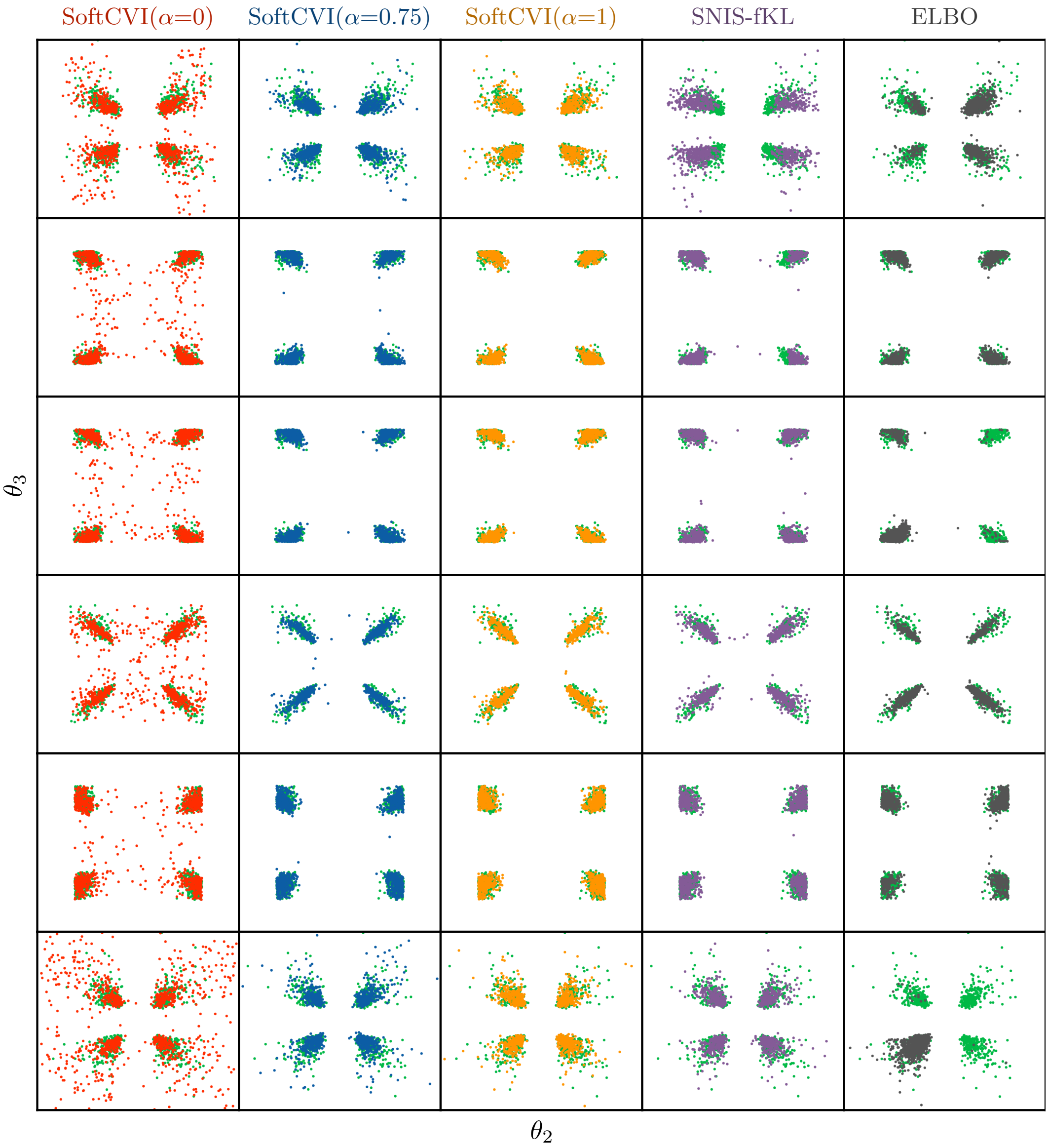

Figure 9: The multimodal 2-dimensional posterior marginals for the first six runs of the SLCP task for each method, with the reference posterior samples shown in green.

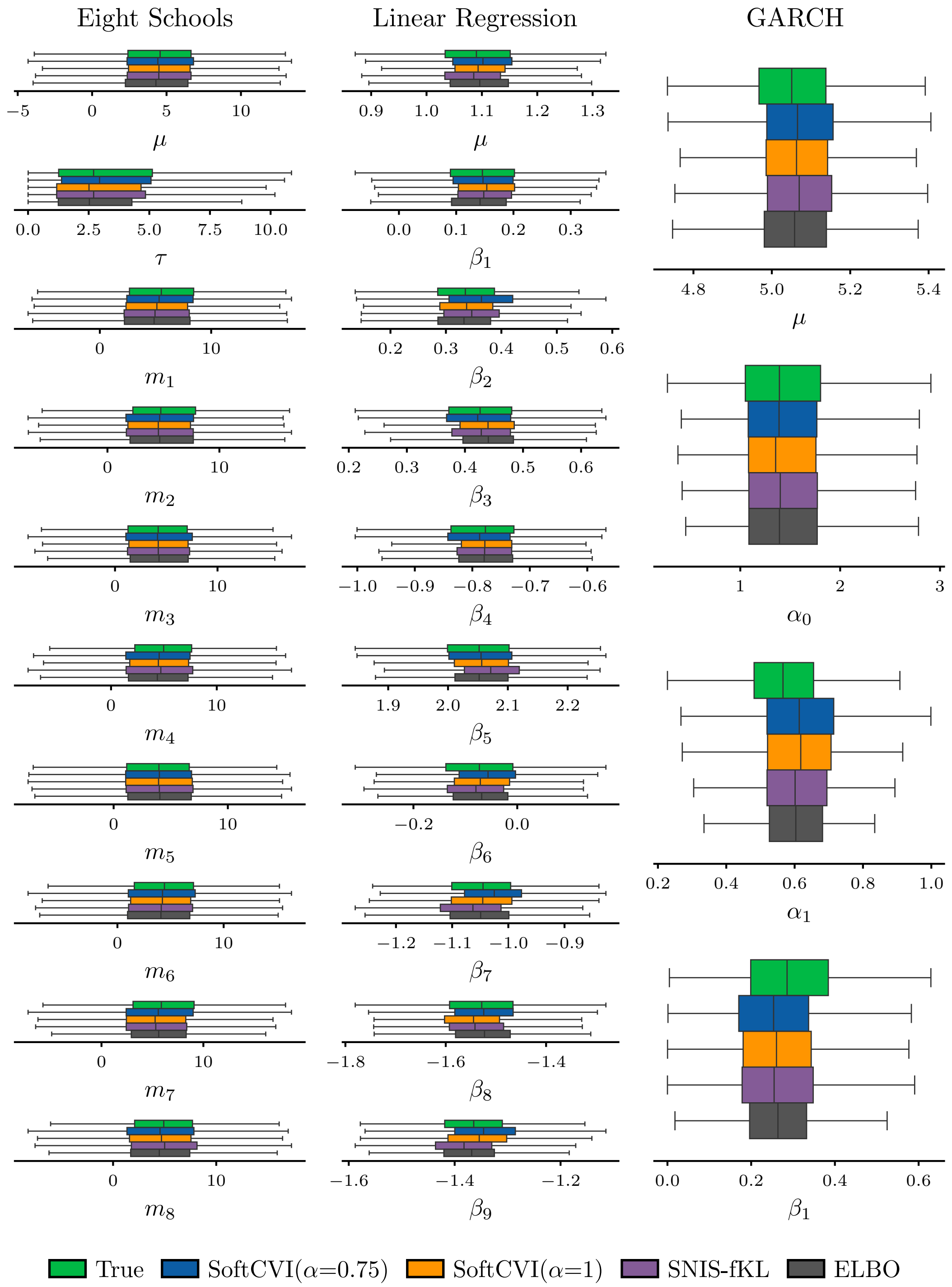

Figure 10: The distribution of the posterior marginals for a single run of the eight schools, linear regression and GARCH(1,1) tasks. For the linear regression task, we restrict the plot to the first 10 parameters of the model.