Can Transformers Learn Full Bayesian Inference In Context?

Arik Reuter [1] **Tim G. J. Rudner** [2] **Vincent Fortuin** [3 4 5] **David Rügamer** [1 5]

Abstract

Transformers have emerged as the dominant architecture in the field of deep learning, with a broad range of applications and remarkable in-context learning (ICL) capabilities. While not yet fully understood, ICL has already proved to be an intriguing phenomenon, allowing transformers to learn in context—without requiring further training. In this paper, we further advance the understanding of ICL by demonstrating that transformers can perform full Bayesian inference for commonly used statistical models in context. More specifically, we introduce a general framework that builds on ideas from prior fitted networks and continuous normalizing flows and enables us to infer complex posterior distributions for models such as generalized linear models and latent factor models. Extensive experiments on real-world datasets demonstrate that our ICL approach yields posterior samples that are similar in quality to state-of-the-art MCMC or variational inference methods that do not operate in context. The source code for this paper is available at https://github.com/ArikReuter/ICL_for_Full_Bayesian_Inference

1. Introduction

In-context learning (ICL) has become a fundamental principle in natural language processing (NLP) with large language models (LLMs) as ubiquitous in-context learners. The core principle of ICL is that a system adapts to a given task based on information provided in its context. This enables the system to address complex problems, such as question answering or text summarization, using a fixed model without requiring any gradient-based fine-tuning, simply by referencing the context. Thereby, ICL enables the generation of real-time solutions through a localized understanding of data without explicit re-training (Dong et al., 2022; Garg et al., 2022).

A fundamental benefit of ICL with LLMs is its versatility. Almost every NLP task involving small data can be solved in context using LLMs, while the performance often surpasses existing baselines (Touvron et al., 2023; OpenAI, 2023; Anil et al., 2023). Additionally, achieving this performance can be very straightforward, requiring only suitably formulated prompts in natural language. Excellent results across a broad variety of tasks, combined with fast inference times and ease of usability, have made in-context learning a machine learning tool employed by millions of people (Eloundou et al., 2023).

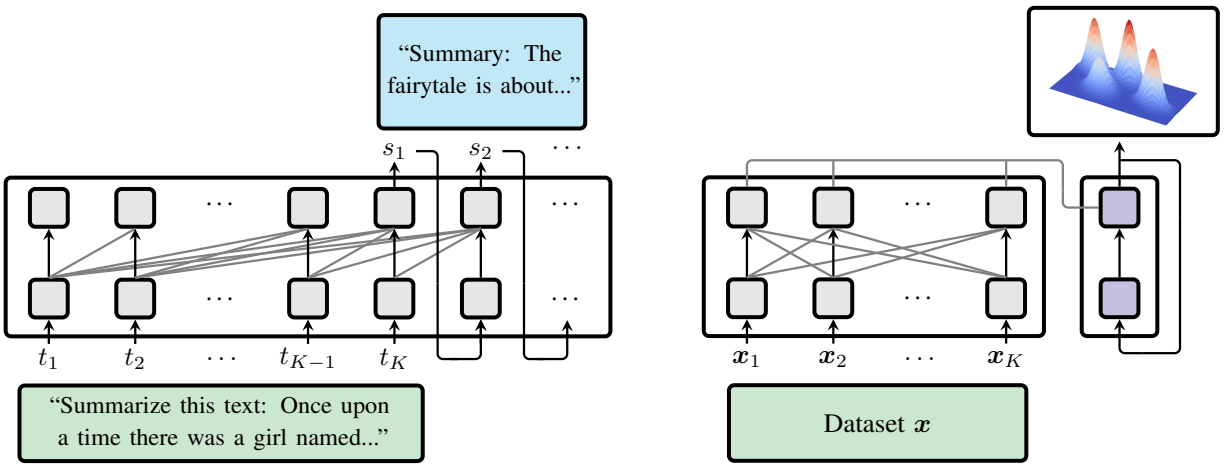
Furthermore, ICL has recently shown remarkable promise for regression and classification tasks involving tabular data, with tabular prior-data fitted networks (TabPFNs) dominating benchmarks alongside minimal prediction time (Hollmann et al., 2022; 2025; Hoo et al., 2024; Robertson et al., 2024). While the internet serves as a suitable source for the massive data needed to train in-context learners on text, TabPFNs demonstrate that training on purely synthetic data facilitates the development of in-context learners for tabular data.

While PFNs perform Bayesian inference, they target a univariate, typically discrete, posterior predictive distribution. In numerous applications, however, high-dimensional and continuous posteriors $P^{\boldsymbol{z}|\boldsymbol{x}}$ of (latent) variables $\boldsymbol{z}$ given data $\boldsymbol{x}$ play a key role.[1] This includes areas such as healthcare (Kyrimi et al., 2021; Abdullah et al., 2022; Etzioni & Kadane, 1995), physics (Gebhard et al., 2025; Brehmer & Cranmer, 2022; Dax et al., 2024), and neuroscience (Lueckmann et al., 2017; Sohn & Narain, 2021). We use the notion of *full Bayesian inference* for methods yielding potentially complex and high-dimensional posterior distributions—in contrast to, for instance, methods that yield only the pos-

[1]Department of Statistics, LMU Munich, Munich, Germany [2]Center for Data Science, New York University, New York, USA [3]Department of Computer Science, Technical University of Munich, Munich, Germany [4]Helmholtz AI, Munich, Germany. [5]Munich Center for Machine Learning (MCML), Munich, Germany. Correspondence to: Arik Reuter <arik.reuter@campus.lmu.de>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

[1]We do not assume any specific form of $\boldsymbol{z}$. That is, there can be a single $\boldsymbol{z}_j$ associated with each data point $\boldsymbol{x}_j$ in $\boldsymbol{x}$, but the case where a single "global" $\boldsymbol{z}$ governs the behavior of each $\boldsymbol{x}_j$ in $\boldsymbol{x}$ is equally included in this notation.



(a) ICL for text summarization using LLMs.

(b) ICL for full Bayesian inference.

Figure 1: **(a)** An LLM generates a summary $s_1, s_2, \ldots$ of a text $t_1, t_2, \ldots, t_K$ through autoregressive sampling while referring to the context using masked self-attention. **(b)** A dataset $\boldsymbol{x}$ is processed with a transformer encoder. Subsequently, cross attention allows generating samples from the posterior conditioned on $\boldsymbol{x}$ in context using a diffusion transformer (decoder). The samples are generated by solving a neural differential equation defining a continuous normalizing flow.

terior predictive or point estimates of the posterior as, for example Hollmann et al. (2022). However, performing full Bayesian inference can be challenging, even for relatively simple models such as generalized linear models (GLMs; Nelder & Wedderburn, 1972). Two common issues when performing full Bayesian inference include (a) slow inference time, particularly when using sampling-based methods (Sommer et al., 2025; 2024), and (b) model misspecification. Although potentially restrictive modeling assumptions are often necessary to make Bayesian inference efficient or even feasible, they can lead to suboptimal predictive performance (Wang & Blei, 2019; Walker, 2013).

In this paper, we address the following question: *Can we leverage in-context learning to effectively perform full Bayesian inference?* In doing so, we aim to obtain an in-context learner that can perform the mapping $\boldsymbol{x} \mapsto P^{\boldsymbol{z}|\boldsymbol{x}}$ for a specific probabilistic model, and, analogous to LLMs, (a) allows for the rapid generation of samples from a posterior of interest during deployment and (b) can flexibly adapt to a broad range of inputs, thereby overcoming issues arising from model misspecification. More specifically, our approach combines a TabPFN encoder (Hollmann et al., 2022) and a diffusion transformer-decoder (Peebles & Xie, 2023) that is trained via flow matching (Lipman et al., 2022).

We present the results of our in-context learning approach on extensive real-world and synthetic datasets in Section 4 and discuss the challenges and the transformative potential of in-context learning for full Bayesian inference in Section 5.

To summarize, our main contributions are as follows:

1. We develop, train, and examine a model that yields samples from the posterior distribution $P^{\boldsymbol{z}|\boldsymbol{x}}$ given data $\boldsymbol{x}$ as context without any (explicit) parameter updates or parametric assumptions about the posterior.
2. To achieve this, we propose to use synthetic samples from the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$ in order to train a large transformer model that performs ICL regarding the posterior $P^{\boldsymbol{z}|\boldsymbol{x}}$, and provide a general framework to analyze the circumstances that enable learning $P^{\boldsymbol{z}|\boldsymbol{x}}$ purely through samples from $P^{\boldsymbol{x},\boldsymbol{z}}$.
3. We then analyze the efficacy of our approach for GLMs and latent factor models, namely Gaussian mixture models (GMMs) and factor analysis (FA). For these applications, we show that including the "prior" used for TabPFNs results in reliably inferring posterior distributions on real-world data.
4. In a variety of experiments, we demonstrate that this approach yields posterior samples that are very similar to those from a Hamiltonian Monte Carlo sampler. Furthermore, we find that the quality of the samples from our ICL approach is preferable, when compared to various popular VI techniques that do not operate in context.
5. Finally, we conduct ablation studies of our approach, examining, for instance, alternative diffusion objectives and Gaussian approximations in place of flow matching, the model's performance on out-of-distribution data, and the impact of problem dimensionality.

2. Related Work

Beyond the perspective of prior-data fitted networks, the contribution of this work can be summarized from the viewpoints of recent work on in-context learning, amortized Bayesian inference, and, simulation-based inference.

In-Context Learning. ICL is a special case of meta-learning (Hospedales et al., 2021) characterized by using a large pre-trained model in order to learn from a context dataset without explicitly updating task-specific parameters. Several recent lines of work investigate the in-context learning capabilities of transformers (Garg et al., 2022; Ahuja et al., 2023; Wang et al., 2024; Chan et al., 2022).

Garg et al. (2022) show that a model similar to GPT-2 can implicitly implement various interesting function classes in context. More specifically, the model learns to reproduce the predictions of different statistical models such as (sparse) linear functions, decision trees, and even two-layer neural networks. This approach can be extended to multiple families of functions and even mixtures of tasks (Ahuja et al., 2023). Kirsch et al. (2022) investigate ICL as a general principle for meta-learning. However, the results by Garg et al. (2022) and Ahuja et al. (2023) are restricted to relatively small problem scales and scalar-valued predictions instead of multivariate posterior distributions. Additionally, the experiments are conducted exclusively on simulated data. In contrast, our results show that (large) transformer models can effectively learn multivariate posterior distributions over latent variables in context on real-world datasets. Furthermore, the focus on latent-variable models naturally steers our investigation toward unsupervised in-context learning, where the primary objective is to uncover the underlying structure of the data rather than to make predictions based on input-target paris presented in the context.

Concurrently, Mittal et al. (2025a) conduct a comparative analysis of amortized in-context Bayesian posterior estimation methods, ablating over different optimization objectives and architectural choices, and also reporting results on out-of-distribution performance and flow-matching methods. They focus on evaluating the posterior mean and downstream predictive performance, whereas we evaluate full posterior distributions. In an analogous setup, Mittal et al. (2025b) assess the effectiveness of learning point estimates versus learning entire distributions in context for the goal of predictive performance.

Amortized Inference. Amortized inference is a central paradigm in the field of variational inference (Kingma, 2013; Zhai et al., 2018; Kim et al., 2018; Margossian & Blei, 2023). A commonly used idea here is to model the posterior distribution $P^{\boldsymbol{z}|\boldsymbol{x}}$ of latent variables $\boldsymbol{z}$ given a dataset $\boldsymbol{x}$ via a factorized density $p(\boldsymbol{z}|\boldsymbol{x}) \approx \prod_{j=1}^{K} q_\theta(\boldsymbol{z}_j|h_{\boldsymbol{\phi}}(\boldsymbol{x}_j))$. In contrast to our more general assumption, each datapoint $\boldsymbol{x}_j$ in $\boldsymbol{x}$ is assumed to have a corresponding latent variable $\boldsymbol{z}_j$. While the parameter θ determines global aspects of the variational distribution, the function $h_{\boldsymbol{\phi}}$ is shared for all $\boldsymbol{x}_j$ and thus amortized across data $\boldsymbol{x}$. Variational autoencoders (Kingma, 2013; Rezende et al., 2014) and neural processes (Garnelo et al., 2018a;b; Rudner et al., 2018) are important model classes based on amortized inference.

In comparison, our ICL approach amortizes its parameters on the level of datasets, such that a single functional relationship is learned for a set $\mathcal{D} \subset (\mathcal{X} \times \mathcal{Z})^N$ of datasets. From this point of view, $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{z}_i)\}_{i=1}^N$ comprising N datasets $\boldsymbol{x}_i \in \mathcal{X}$ and the corresponding latent variables $\boldsymbol{z}_i \in \mathcal{Z}$ can be seen as a "meta-dataset" for which we perform amortized inference. This is similar in nature to the setup by Le et al. (2017), who use recurrent neural networks to "compile" inference based on execution traces of probabilistic programs by training on simulated data.

Unlike in amortized variational inference, we do not use the notion of an evidence lower bound (Blei et al., 2017) or even the Kullback-Leibler divergence to learn the posterior, but utilize ideas that also appear in the context of simulation-based inference.

Simulation-Based Inference. Analogously to latent variable models, some scientific simulations, for instance in neuroscience or astrophysics (Fan & Markram, 2019; Schmit & Pritchard, 2018), allow to draw samples from the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$ of data and latent variable of interest. Amortized posterior inference in this context is referred to as simulation-based inference (SBI; Cranmer et al., 2020). Several recent approaches focus on using neural networks to directly infer aspects of the likelihood $p(\boldsymbol{x}|\boldsymbol{z})$, the posterior $P^{\boldsymbol{z}|\boldsymbol{x}}$ or the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$. More specifically, techniques based on discrete normalizing flows (Dax et al., 2021) or flow-matching (Wildberger et al., 2024) are used to approximate the posterior $P^{\boldsymbol{z}|\boldsymbol{x}}$, while (Gloeckler et al., 2024) propose to use a transformer-based diffusion model in order to approximate the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$. In recent work, Vetter et al. (2025) directly a pre-trained TabPFN to auto-regressively sample $P^{\boldsymbol{z}|\boldsymbol{x}}$ leveraging ICL.

From a simulation-based inference viewpoint, we demonstrate that sample-based posterior estimation can be used for full Bayesian inference in complex scenarios arising in commonly used latent variable models, and demonstrate the effectiveness of this approach on real-world datasets.

3. In-Context Learning for Full Bayesian Inference

Bayesian inference is a tool of central importance for countless applications. However, exact posterior inference can

become computationally expensive when using sampling-based methods (Hastings, 1970; Hoffman et al., 2014; Betancourt, 2017) and even impossible when relying on fully factorized VI methods, which can incur substantial approximation errors (Bishop et al., 2002; Blei, 2012; Margossian & Blei, 2023). Amortized variational inference can alleviate those issues but typically requires the development of specialized and complex modeling frameworks (Kingma, 2013; Srivastava & Sutton, 2017; Garnelo et al., 2018b; Lin et al., 2021). Another issue with variational inference arises from having to choose a variational distribution. While insufficient flexibility in this respect can lead to overly simplistic posteriors, a too flexible variational distribution might overfit the given data (Cremer et al., 2018).

We propose a simple and effective solution based on ideas from ICL, which can be seen as conducting amortized inference on a dataset level. Training a model on a potentially unlimited amount of synthetic datasets yields an in-context learner that can not only approximate a vast, almost arbitrarily large, class of distributions but is also highly efficient when used for sampling. Furthermore, this does not incur the same issues with overly or insufficiently flexible distribution assumptions that are present in VI. More specifically, empirical results show that a major strength of TabPFN, for instance, is its ability to adapt flexibly to the complexity of the problem at hand, thus removing the need for extensive hyperparameter tuning (Hollmann et al., 2022).

In the following, we describe a sufficient general condition, as well as a specific framework that allows to train probabilistic in-context learners on simulated data.

The idea underlying the proposed approach is founded on two observations relating to full Bayesian inference and the working principle of PFNs: First, many Bayesian models have a generative formulation that allows the simulation of arbitrarily large amounts of training samples from the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$. We assume that samples from $P^{\boldsymbol{x},\boldsymbol{z}}$ comprise a dataset $\boldsymbol{x} = \{\boldsymbol{x}_j\}_{j=1}^K$ containing K samples $\boldsymbol{x}_j \in \mathcal{X}$ and a corresponding (latent) variable $\boldsymbol{z} \in \mathcal{Z}$.[2] This joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$ corresponds to the "prior" in PFNs and allows the training of a large neural network that implicitly learns to perform Bayesian inference. Second, Bayesian inference is especially useful for smaller datasets $\boldsymbol{x}$ that can be processed in a single forward pass. This makes an entire dataset a viable context for Bayesian ICL.

More specifically, the central goal is to develop a method allowing to infer the posterior distribution $P^{\boldsymbol{z}|\boldsymbol{x}}$ of latent variables $\boldsymbol{z} \in \mathcal{Z}$, given observations $\boldsymbol{x} \in \mathcal{X}$ using ICL. From a supervised-learning perspective, we thus aim to directly learn the mapping $f_0 : \mathcal{X} \to \mathcal{M}(\mathcal{Z}), \boldsymbol{x} \mapsto P^{\boldsymbol{z}|\boldsymbol{x}}$, where $\mathcal{M}(\mathcal{Z})$ is the space of all probability measures. Therefore, we want a model $f_\theta(\boldsymbol{x}) = Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}$ for the posterior to be as close as possible to the true posterior $P^{\boldsymbol{z}|\boldsymbol{x}} = f_0(\boldsymbol{x})$. We measure "closeness" w.r.t. some divergence $d : \mathcal{M}(\mathcal{Z}) \times \mathcal{M}(\mathcal{Z}) \to [0, \infty)$. When considering the expected divergence over data samples $\boldsymbol{x} \sim P^{\boldsymbol{x}}$, this gives rise to the following objective: $\mathcal{R}_\theta := \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}\left[d\left(f_\theta(\boldsymbol{x}), f_0(\boldsymbol{x})\right)\right]$, which can also be directly expressed as

$$\mathcal{R}_\theta = \mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}\left[d\left(Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}, P^{\boldsymbol{z}|\boldsymbol{x}}\right)\right]. \tag{1}$$

Note that we use the notion of a divergence d loosely to refer to any measure of similarity of two distributions. Although $\mathcal{R}_\theta$ itself is usually intractable, specific choices of d and the use of the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$ make Equation (1) accessible via

$$\widetilde{\mathcal{R}}_\theta := \mathbb{E}_{\boldsymbol{x},\boldsymbol{z}\sim p(\boldsymbol{x},\boldsymbol{z})}\left[\mathcal{L}_d(\boldsymbol{x}, \boldsymbol{z}, \theta)\right], \tag{2}$$

where the loss function $\mathcal{L}_d$ depends on d and the structure of $Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}$ (discussed in detail later). Performing empirical risk minimization for $\widetilde{\mathcal{R}}_\theta$ with samples from the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$ then corresponds to learning to approximate $P^{\boldsymbol{z}|\boldsymbol{x}}$. The model for the posterior $P^{\boldsymbol{z}|\boldsymbol{x}}$ is thereby only implicitly defined by the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$. While this requires the ability to sample from $P^{\boldsymbol{x},\boldsymbol{z}}$, drawing samples from the joint distribution is often a weak requirement in terms of model specification that immediately follows from specifying the generative process of a model. Furthermore, a simple sufficient condition that follows directly from the law of total expectation implies the equivalence of $\mathcal{R}_\theta$ and $\widetilde{\mathcal{R}}_\theta$:

Proposition 1. Let $d(Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}, P^{\boldsymbol{z}|\boldsymbol{x}}) = \int \gamma\left(Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}\right) dP^{\boldsymbol{z}|\boldsymbol{x}}$ for some measurable functional $\gamma : \mathcal{M}(\mathcal{Z}) \to \mathbb{R}$.
Then $\mathcal{R}_\theta = \widetilde{\mathcal{R}}_\theta$ with $\mathcal{L}_d(\boldsymbol{x}, \boldsymbol{z}, \theta) = \gamma\left(Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}\right)$.

For instance, choosing d to be the forward Kullback-Leibler divergence $d_{\mathrm{KL}}(Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}, P^{\boldsymbol{z}|\boldsymbol{x}}) = \mathbb{D}_{\mathrm{KL}}\left[p(\cdot|\boldsymbol{x})||q_\theta(\cdot|\boldsymbol{x})\right]$ implies that $\mathcal{L}_{d_{\mathrm{KL}}}(\boldsymbol{x}, \boldsymbol{z}, \theta) = -\log q_\theta(\boldsymbol{z}|\boldsymbol{x}) + const.$ (Müller et al., 2021). In this case, minimizing $\widetilde{\mathcal{R}}_\theta$ thus directly corresponds to performing maximum likelihood inference on samples from $P^{\boldsymbol{x},\boldsymbol{z}}$.

3.1. Defining the Form of the Posterior

To learn the posterior distribution $P^{\boldsymbol{z}|\boldsymbol{x}}$ in context, we use the framework of flow matching (Lipman et al., 2022). More specifically, we utilize continuous normalizing flows (CNFs) to specify and ultimately sample from $P^{\boldsymbol{z}|\boldsymbol{x}}$. CNFs, currently excelling in the field of image synthesis (Esser et al., 2024), do not only allow to flexibly learn almost arbitrary distributions, but are also found to be more sample-efficient

[2] We do not assume any specific form of $\boldsymbol{z}$. That is, there can be a single $\boldsymbol{z}_j$ associated with each data point $\boldsymbol{x}_j$ in $\boldsymbol{x}$, but the case where a single "global" $\boldsymbol{z}$ governs the behavior of each $\boldsymbol{x}_j$ in $\boldsymbol{x}$ is equally included in this notation.

in training than for instance diffusion objectives (Lipman et al., 2022; Wildberger et al., 2024). Furthermore, unlike discrete normalizing flows (Papamakarios et al., 2021a), CNF objectives do not limit the architecture of the used neural network, allowing to incorporate complex conditioning on the data $\boldsymbol{x}$ in addition to flexibly modeling the posterior, which is a crucial aspect of our ICL framework. Refer to Appendix D for more information on CNFs.

3.1.1. Normalizing flows

The key idea of modeling a distribution $P^{\boldsymbol{z}|\boldsymbol{x}}$ with normalizing flows (see, e.g., Papamakarios et al., 2021b), which are the basis of CNFs, is to assume that $P^{\boldsymbol{z}|\boldsymbol{x}}$ is the result of "pushing forward" a simple base distribution $P_{\mathcal{B}}$ into $P^{\boldsymbol{z}|\boldsymbol{x}}$ using a conditional flow $\psi_\theta(\cdot|\boldsymbol{x})$:

$$P^{\boldsymbol{z}|\boldsymbol{x}} \approx [\psi_\theta(\cdot|\boldsymbol{x})]_\sharp P_{\mathcal{B}}. \tag{3}$$

Therefore, one assumes that samples from $P^{\boldsymbol{z}|\boldsymbol{x}}$ are generated by first drawing $\boldsymbol{z}^{(0)} \sim P_{\mathcal{B}}$, and then applying $\psi_\theta(\cdot|\boldsymbol{x})$, such that $\psi_\theta(\boldsymbol{z}^{(0)}|\boldsymbol{x}) \sim P^{\boldsymbol{z}|\boldsymbol{x}}$. The base distribution $P_{\mathcal{B}}$ is commonly set to be a standard normal distribution, i.e., $P_{\mathcal{B}} = \mathcal{N}(0, I)$. The conditional flow $\psi_\theta(\cdot|\boldsymbol{x})$ is the object to be learned, such that our model of $P^{\boldsymbol{z}|\boldsymbol{x}}$ is defined as $Q_\theta^{\boldsymbol{z}|\boldsymbol{x}} := [\psi_\theta(\cdot|\boldsymbol{x})]_\sharp P_{\mathcal{B}}$.

3.1.2. Continuous Normalizing Flows

In flow matching (Lipman et al., 2022), which we will use to obtain an in-context learner for full Bayesian inference, the normalizing flow $\psi_\theta(\cdot|\boldsymbol{x})$ is implicitly defined via a (conditional) vector field $v_{t,\boldsymbol{x}}^\theta$ of an ordinary differential equation (ODE):

$$\frac{d}{dt}\psi_{\theta,t}(\boldsymbol{z}|\boldsymbol{x}) = v_{t,\boldsymbol{x}}^\theta(\psi_{\theta,t}(\boldsymbol{z}|\boldsymbol{x})),\ \psi_{\theta,0}(\boldsymbol{z}|\boldsymbol{x}) = \boldsymbol{z}, \tag{4}$$

where $0 \leq t \leq 1$. The first condition $\frac{d}{dt}\psi_{\theta,t}(\boldsymbol{z}|\boldsymbol{x}) = v_{t,\boldsymbol{x}}^\theta(\psi_{\theta,t}(\boldsymbol{z}|\boldsymbol{x}))$ means that $v_{t,\boldsymbol{x}}^\theta$ describes the change in $\psi_{\theta,t}(\boldsymbol{z}|\boldsymbol{x})$ at time t, and the second condition $\psi_{\theta,0}(\boldsymbol{z}|\boldsymbol{x}) = \boldsymbol{z}$ implies that initially the flow is just the identity. The family of vector fields $v_{t,\boldsymbol{x}}^\theta$ is parameterized by a neural network whose parameters θ will be learned. In order to ultimately compute the flow $v_{1,\boldsymbol{x}}^\theta$, that yields $Q_\theta^{\boldsymbol{z}|\boldsymbol{x}} = [\psi_{\theta,1}(\cdot|\boldsymbol{x})]_\sharp P_{\mathcal{B}}$, a numerical ODE solver can be used to forward-solve the ODE, which ultimately corresponds to evaluating $\psi_{1,\boldsymbol{x}}$ at a data point $\boldsymbol{z}^{(0)} \sim P_{\mathcal{B}}$. This construction implies very generic assumptions regarding the structure of $Q^{\boldsymbol{z}|\boldsymbol{x}}$, which include the existence of a density of the target distribution wrt. the Lebesgue measure, and the assumption that $P^{\boldsymbol{z}|\boldsymbol{x}}$ can be represented by a mixture distribution over the marginal probability paths at time point $t = 1$ (Lipman et al., 2022). Please note that the mathematical understanding of Flow Matching, its properties and assumptions, are still actively researched (Wildberger et al., 2024).

Assuming Gaussian conditional probability paths with an optimal-transport mean- and variance-function (Lipman et al., 2022), one obtains the following discrepancy measure d_{CFM} between $Q_\theta^{\boldsymbol{z}|\boldsymbol{x}} := [\psi_{\theta,1}(\cdot|\boldsymbol{x})]_\sharp P_{\mathcal{B}}$ and $P^{\boldsymbol{z}|\boldsymbol{x}}$:

$$d_{\text{CFM}}\left(Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}, P^{\boldsymbol{z}|\boldsymbol{x}}\right) := \mathbb{E}\left[\left|\left|v_{t,\boldsymbol{x}}^\theta(\gamma_t(\boldsymbol{z}^{(1)}|\boldsymbol{z}^{(0)})) - (\boldsymbol{z}^{(1)} - \omega\boldsymbol{z}^{(0)})\right|\right|_2^2\right], \tag{5}$$

where the expectation is taken w.r.t. to three random variables: a uniform time-step $t \sim \mathcal{U}([0,1])$, samples from the base distribution $\boldsymbol{z}^{(0)} \sim P_{\mathcal{B}}$, and samples from the ground-truth conditional distribution $\boldsymbol{z}^{(1)} \sim P^{\boldsymbol{z}|\boldsymbol{x}}$. We define $\gamma_t(\boldsymbol{z}^{(1)}|\boldsymbol{z}^{(0)}) := (1-\omega t)\boldsymbol{z}^{(0)} + t\boldsymbol{z}^{(1)}$.

We refer to (Wildberger et al., 2024) for mathematical results on the relationship of d_{CFM} and the (forward) Kullback-Leibler divergence. The hyperparameter $\omega = 1 - \sigma_{\min}$, where $\sigma_{\min}$ is the variance at time $t = 1$ in the Gaussian conditional probability paths, appears to have negligible influence when set to a value sufficiently close to one (Lipman et al., 2022).[3]

In order to make optimizing

$$\mathbb{E}_{\boldsymbol{x}\sim p(\boldsymbol{x})}\left[d_{\text{CFM}}\left(Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}, P^{\boldsymbol{z}|\boldsymbol{x}}\right)\right] \tag{6}$$

tractable, and thus train our in-context learner, we make use of the sufficient condition in Proposition 1. Thus, the divergence d_{CFM} admits the re-formulation as an objective $\widetilde{\mathcal{R}}_\theta$ using samples from the joint distribution $P^{\boldsymbol{x},\boldsymbol{z}}$. We can therefore optimize $\widetilde{\mathcal{R}}_\theta$ using N independent and identically distributed (i.i.d.) samples $t_i \sim \mathcal{U}([0,1])$ from the time-distribution, $\boldsymbol{z}_i^{(0)} \sim P_{\mathcal{B}}$ from the base distribution, and $(\boldsymbol{z}_i^{(1)}, \boldsymbol{x}_i) \sim P^{\boldsymbol{x},\boldsymbol{z}}$ from the joint distribution. With this, we obtain the following objective function used for the training of the ICL models:

$$\hat{\mathcal{R}}_\theta = \sum_{i=1}^N \left|\left|v_{t_i,\boldsymbol{x}_i}^\theta(\gamma_{t_i}(\boldsymbol{z}_i^{(1)}|\boldsymbol{z}_i^{(0)})) + \boldsymbol{z}_i^{(1)} - \omega\boldsymbol{z}_i^{(0)}\right|\right|_2^2 \tag{7}$$

3.2. Sampling from the Joint Distribution

In order to learn a model that can perform posterior inference according to Section 3.1, we require to sample $(\boldsymbol{x}, \boldsymbol{z}) \sim P^{\boldsymbol{x},\boldsymbol{z}}$. Given $p(\boldsymbol{x}, \boldsymbol{z}) = p(\boldsymbol{x}|\boldsymbol{z})p(\boldsymbol{z})$, this is always possible as long as one can draw samples from $P^{\boldsymbol{z}}$ and then from $P^{\boldsymbol{x}|\boldsymbol{z}}$. Hence, this is a relatively weak requirement allowing for a broad variety of priors and observation models. More specifically, for ICL, we generate a training dataset $\mathcal{D}$ which comprises i.i.d. samples $\{(\boldsymbol{x}_i, \boldsymbol{z}_i)\}_{i=1}^N$ resulting from sampling $\boldsymbol{z}_i \sim P^{\boldsymbol{z}}$ and then $\boldsymbol{x}_i \sim P^{\boldsymbol{x}|\boldsymbol{z}_i}$. We

[3] In our experiments, we follow (Wildberger et al., 2024) and set $\omega := 1 - 10^{-4}$ for all experiments.

use this simple yet fundamental and very general template to generate samples from the joint $P^{\boldsymbol{x},\boldsymbol{z}}$ for GLMs, factor analysis (FA), and Gaussian mixture models (GMMs) in our later applications. Please refer to Appendix A for more details on the data generating processes.

3.3. The Architecture

In order to implement the idea of learning full Bayesian inference in context, we extend ideas of diffusion transformers (Peebles & Xie, 2023), where the conditioning on the time t is implemented via adaptive layer norm (adaLN) blocks initialized as the identity function. As we potentially require complex conditioning on the data $\boldsymbol{x}$, an additional transformer encoder is added. The input to the decoder is a vector in the form $(1-\omega t)\boldsymbol{z}^{(0)} + t\boldsymbol{z}^{(1)}$, which is treated as a sequence with length one and processed by a transformer decoder without self-attention, but the adaLN blocks. Therefore, the decoder has an equivalent interpretation as a multilayer perceptron with skip-connections, cross-attention, and adaptive layer normalization. For the final processing in the decoder, only conditional feedforward layers with adaptive layer normalization are used. This corresponds exactly to the architecture of the decoder before, albeit without cross attention. We call this part an "MLP with Conditioning". Samples for the time $t \in [0,1]$ are mapped onto a conditioning vector using several fully connected layers, which yields a richer representation of t that is well-suited as an input to the adaLN blocks. Figure 2 depicts of the resulting architecture.

3.4. Implementing Flow Matching

During the training phase, a tuple $(\boldsymbol{z}^{(1)}, \boldsymbol{x})$ is drawn from the distribution $P^{\boldsymbol{z},\boldsymbol{x}}$. Additionally, a time step $t \sim \mathcal{U}[0,1]$ and a sample $\boldsymbol{z}^{(0)}$ is drawn from the base distribution $P_{\mathcal{B}}$, which is a standard Gaussian for all our applications. Subsequently, the ground-truth conditional flow $\psi(\boldsymbol{z}^{(0)}|\boldsymbol{x}) = (1-\omega t)\boldsymbol{z}^{(0)} + t\boldsymbol{z}^{(1)}$ is computed, pushing forward $P_{\mathcal{B}}$ into $P^{\boldsymbol{z}|\boldsymbol{x}}$ up to time-point t. The transformer encoder processes $\boldsymbol{x}$ and the decoder takes the representation of the encoder into account in order to output $v^{\theta}_{t,\boldsymbol{x}}(\psi(\boldsymbol{z}^{(0)}|\boldsymbol{x}))$. This output should match the vector field that describes how the ground-truth flow $\psi(\boldsymbol{z}^{(0)}|\boldsymbol{x})$ continues at time t. The discrepancy to the ground-truth vector field is measured with the MSE-loss in Equation (7).

In the sampling phase, we are given $\boldsymbol{x}$ and the goal is to sample from $P^{\boldsymbol{z}|\boldsymbol{x}}$. To do so, first a vector $\boldsymbol{z}^{(0)} \sim P_{\mathcal{B}}$ is drawn. The data $\boldsymbol{x}$ is passed through the encoder. The decoder defines a function that maps a time-point t and a vector $\boldsymbol{\nu}$ onto a vector field: $(t, \boldsymbol{\nu}) \mapsto v^{\theta}_{t,\boldsymbol{x}}(\boldsymbol{\nu})$ taking $\boldsymbol{x}$ into account. This function is given to an ODE-solver in order to forward-solve the corresponding ODE with boundary conditions $0 \leq t \leq 1$.

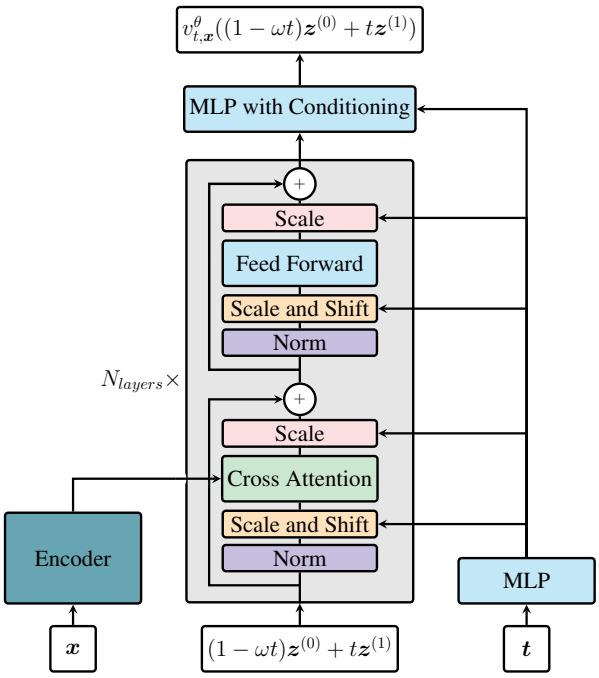


Figure 2: Architecture to perform ICL for full Bayesian inference. A relatively large transformer encoder, similar to that in TabPFN (Hollmann et al., 2022) processes a dataset $\boldsymbol{x}$ and yields a representation used in the decoder. The decoder outputs a vector field defining a flow for a given input vector conditioned on the encoder output and the time. We condition on the time in each cross-attention and each feed-forward block. Please note that the size of the parts of the architecture does not correspond to the number of allocated parameters in this figure.

4. Experiments

To show that the proposed methodology is not just an abstract concept, we derive exemplary use cases that demonstrate how well ICL is able to keep up with MCMC and VI approaches in practice.

For this, we will use two prominent statistical modeling classes, namely generalized linear models (GLMs) and latent factor models. For the latent factor models, we consider factor analysis (FA) and Gaussian mixture models (GMMs).

Modeling Scenarios. We use seven different scenarios for the GLMs, where we vary the prior distribution on the parameters, the conditional distribution of the response, and whether an intercept is included. For FA, we vary the form of the priors and dimensionalities of variables leading to four different scenarios. For the GMMs, we investigate different dimensionalities as well as prior configurations also in four different scenarios. We refer to Appendix A for details on the model structure and scenarios.

Table 1: Summarized results for GLMs. Average performance of VI methods and our ICL approach on 50 synthetic and 17 real-world datasets across 7 different GLM scenarios. Comparison to the analytical solution when available and HMC otherwise. Lower is better for all metrics. The best average result is marked in **bold**.

Model	Synthetic			Real-World		
	C2ST	MMD	$\mathcal{W}_2$	C2ST	MMD	$\mathcal{W}_2$
LA	1.000	2.770	2.049	1.000	2.091	0.849
VI: Diagonal	0.869	1.586	1.742	0.819	0.583	0.529
VI: Full	0.714	1.016	1.601	0.668	0.116	0.374
VI: Structured	0.711	0.929	1.580	0.664	0.109	**0.370**
VI: IAF	0.784	1.648	2.349	0.732	0.516	0.680
ICL (ours)	**0.657**	**0.183**	**0.556**	**0.648**	**0.090**	0.387

Table 2: Results for GLMs. Real-world Evaluation on 17 datasets: Linear regression with a gamma prior on the coefficients β, and an inverse gamma prior on the variance σ^2 of the responses (scenario 5). Comparison to HMC samples. All results within two standard errors of the best average result are marked in **bold**.

Model	C2ST	MMD	$\mathcal{W}_2$
LA	1.000 ($\pm$ 0.000)	1.982 ($\pm$ 0.126)	0.623 ($\pm$ 0.084)
VI: Diagonal	0.810 ($\pm$ 0.036)	0.441 ($\pm$ 0.252)	0.384 ($\pm$ 0.089)
VI: Full	0.711 ($\pm$ 0.038)	0.148 ($\pm$ 0.093)	**0.279** ($\pm$ 0.056)
VI: Structured	0.705 ($\pm$ 0.032)	0.140 ($\pm$ 0.081)	**0.269** ($\pm$ 0.045)
VI: IAF	0.777 ($\pm$ 0.106)	0.684 ($\pm$ 0.939)	0.625 ($\pm$ 0.525)
ICL (ours)	**0.610** ($\pm$ 0.045)	**0.046** ($\pm$ 0.020)	**0.242** ($\pm$ 0.038)

For our experiments, we train a separate model from scratch for each GLM, GMM, and FA scenario using synthetic samples from the joint distribution $P^{\boldsymbol{x},\boldsymbol{x}}$; i.e. we train seven separate models to cover the GLM scenarios, six separate models for the FA scenarios and four separate models for the GMM scenarios. Please refer to Appendix B for more details.

Datasets. We evaluate the methods on 50 synthetic datasets and 17 real-world datasets from a benchmark suite for tabular regression problems proposed by Grinsztajn et al. (2022). We refer to Appendix C for more details on the preprocessing of the datasets.

Methods. Apart from a comparison with a gold standard, we compare our ICL approach to a Laplace approximation (LA; Daxberger et al., 2021) and different established VI methods based on automatic differentiation VI (Kucukelbir et al., 2017). For the variational distribution, we use a normal distribution with 1) a diagonal and 2) a full covariance matrix, as well as 3) a structured normal distribution with linear dependencies between the latent variables, and 4) an approach based on inverse autoregressive flows (IAF; Kingma et al., 2016). Appendix F includes a discussion regarding the hyperparameters of all considered methods.

Evaluation Process. For every synthetic and real-world dataset, 1000 posterior samples from each method are compared against samples from the analytical solution, if available, or from a Hamiltonian Monte Carlo (HMC) sampler with a NUTS kernel (Hoffman et al., 2014) as the gold standard. If posteriors are unimodal, we run a single chain. In the multimodal case, we use three times the number of modes as the number of Markov chains.

Evaluation Metrics. Three metrics are employed to compare samples from different approximations of the posterior distribution. The first metric is a classifier 2-sample test (C2ST; Lueckmann et al., 2021; Lopez-Paz & Oquab, 2016), where the ROC-AUC score of a random forest classifier, trained to distinguish between samples from the gold standard and the method in question, is utilized. For random forest, we use default hyperparameters, as defined in Scikit-learn (Pedregosa et al., 2011) and 10-fold cross-validation. We use a random forest with the given hyperparameters as a highly performative classifier in order to detect small deviations in distributions, even though this incurs the risk that the C2ST quickly saturates at a value of one, especially in high-dimensional cases. The second metric is the maximum mean discrepancy (MMD) between the two distributions (gold-standard and each tested method) with an exponential kernel (Gretton et al., 2012). The third metric is the empirical Wasserstein-2 distance ($\mathcal{W}_2$; Givens & Shortt, 1984) of the two distributions, as implemented in the POT library (Flamary et al., 2021).

4.1. Generalized Linear Models

Across seven different variants of GLMs, we find that ICL yields samples that have overall the highest agreement with the gold standard (see Table 1). Specifically on the synthetic datasets, the C2ST, MMD and $\mathcal{W}_2$ metrics indicate that the posterior distribution can be approximated more accurately with ICL than via variational inference.

Particularly in cases where the posterior has a shape deviating from a normal distribution, ICL and HMC agree more closely than VI. For instance, in the case where a gamma prior, i.e. a skewed distribution, is used on the coefficients of a regression model, we find that ICL substantially outperforms VI both on synthetic and real-world data (see Table 2). On the real-world data, ICL still matches the performance of VI methods and has the best (or not significantly worse than the best) performance in terms of C2ST in four out of seven cases (see Table 2). Please refer to Appendix I for the detailed experimental results summarized in Table 1.

Table 3: Summarized results for FA. Average performance of VI methods and our ICL approach on 50 synthetic and 17 real-world datasets across 6 different FA scenarios. Comparison to HMC samples. Lower is better for all metrics. The best average result is marked in **bold**.

Model	Synthetic			Real-World		
	C2ST	MMD	$\mathcal{W}_2$	C2ST	MMD	$\mathcal{W}_2$
LA	1.000	4.115	2.543	1.000	4.127	0.597
VI: Diagonal	0.999	3.321	1.998	0.960	1.220	0.288
VI: Full	0.993	3.222	1.955	0.950	1.173	0.281
VI: Structured	0.995	3.404	2.079	0.955	1.189	0.283
VI: IAF	0.987	3.226	1.973	0.902	0.969	**0.251**
ICL (ours)	**0.568**	**0.057**	**0.409**	**0.751**	**0.673**	0.583

Table 4: Summarized Results for GMMs. Average performance of VI methods and our ICL approach on 50 synthetic and 17 real-world datasets across 4 different GMM scenarios. Comparison to HMC samples in all cases. The best average result is marked in **bold**.

Model	Synthetic			Real-World		
	C2ST	MMD	$\mathcal{W}_2$	C2ST	MMD	$\mathcal{W}_2$
LA	1.000	3.916	8.324	1.000	3.385	12.740
VI: Diagonal	0.994	2.676	7.938	0.992	2.182	11.633
VI: Full	0.995	2.556	7.947	0.987	2.143	11.696
VI: Structured	0.994	2.595	7.929	0.988	2.129	11.521
VI: IAF	0.985	2.308	7.489	0.957	1.845	11.541
ICL (ours)	**0.825**	**0.706**	**4.348**	**0.881**	**1.051**	**10.691**

4.2. Factor analysis

On the factor analysis tasks, ICL has notably lower dissimilarity scores compared to the gold standard than all other considered methods in the synthetic evaluation (Table 3). Notably, an average C2ST score of 0.568 is remarkably close to the theoretical lower bound of 0.5. Regarding the real world datasets, C2ST and MMD indicate that our ICL approach yields samples most similar to the reference, while the average $\mathcal{W}_2$ score is substantially higher. We hypothesize that this discrepancy in the metrics might be caused by numerical issues when computing the empirical $\mathcal{W}_2$ distance. Furthermore, the relatively high number of latent variables in comparison to the limited number of data-points can yield overly flexible assumptions on the variational posterior causing the VI methods to overfit. See Appendix I for the detailed experimental results summarized in Table 3.

4.3. Gaussian Mixture Models

Full Bayesian inference for GMMs is more challenging than for GLMs or FA. First, the generative process of GMMs involves discrete assignments to clusters, which poses a challenge not only for NUTS, but especially for VI methods. Second, the dimensionality of the posterior samples can be relatively large since for diagonal normal distributions, each component of the mixture has a mean and a variance parameter per dimension. Finally, the considered GMMs are not identifiable leading to multi-modal posterior distributions, which are impossible to perfectly approximate with commonly used VI methods based on Gaussian approximations.

Due to this inherent difficulty of the GMM scenarios, we find the overall performances of all models to be worse than in the GLM and FA cases. In particular, the C2ST metric is almost saturated for the VI approaches and has a value of around 83 percent for ICL (Table 4). The MMD and $\mathcal{W}_2$ metrics also indicate that ICL yields samples with higher agreement with the reference than the other approaches on synthetic data. A plot of the marginals of the posterior shows high agreement between the posterior distributions of both HMC and ICL while VI is incapable of perfectly approximating a bimodal distribution and exhibits typical mode-seeking behavior (Figure 3). Note that also the VI approach based on inverse autoregressive flows, which in theory allows flexible modeling of a wide range of posterior shapes, fails to learn the bi-modality accurately from the limited number of 50 data points in this GMM scenario. This demonstrates the strength of our ICL approach in flexibly learning distributions agnostic of the provided sample size. Please refer to Appendix I for the detailed experimental results summarized in Table 4.

4.4. Ablations and Further Experimental Results

In this subsection, we present various ablations concerning our ICL approach to full Bayesian inference. Due to limited space, most of the results are deferred to the appendix.

Alternatives to Flow Matching. Appendix L contains results from an ablation study using diffusion objectives instead of flow matching, while Appendix K investigates the use of a multivariate Gaussian to parametrize $Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}$. The empirical results from these ablations strongly indicate that flow matching is essential for achieving a close approximation of the gold-standard posterior in the scenarios we consider.

Dimensionality. In addition, we investigate the effect of the dimensionality K of the latent variable $\boldsymbol{z} \in \mathbb{R}^K$ for all seven different GLM scenarios. The key takeaway from our results is that for $K = 20$ and $K = 50$, the ICL approach performs comparably to the other methods in terms of sample similarity to HMC, but does not outperform them. Please refer to Appendix O for more details. We hypothesize that a key reason for the failure to detect meaningful differences between the methods in high dimensions is due to the curse of dimensionality affecting our metrics.

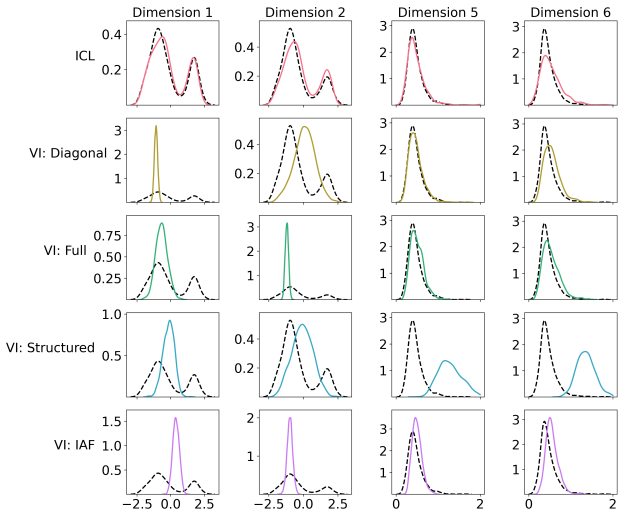


Figure 3: Density plots for the marginals of the posterior for GMM scenario 1. Comparison to HMC samples on a synthetic dataset whose density is depicted as a dotted line. Only the marginals of the first two components of the mean and the variance are shown. The density of the posterior obtained via HMC is depicted as a dotted line. While the ICL method aligns with the gold-standard HMC, the VI methods have a lower level of agreement and exhibit mode-seeking behaviour.

Out-of-distribution Performance. We further investigate the robustness of our method under mild distribution shifts in Appendix N. Our results indicate that the performance of our ICL method remains relatively stable for small distribution shifts, but increasingly degrades for a larger gap between the training and testing distribution.

Predictive Performance. Additionally, we evaluate our ICL method and all variational approaches with respect to the predictive performance of the considered GLM setups in Appendix J. Results confirm the strong performance of variational inference methods in terms of point prediction, especially for high dimensionalities, while the ICL method is generally competitive.

Architecture. Appendix M discusses results regarding the effect of using an MLP-based architecture. Our experimental findings confirm that the transformer-based encoder performs significantly better than an equally sized MLP encoder.

The Classifier in the C2ST Metric. Finally, we validate the choice of a random forest classifier for the C2ST metric (Appendix Q). We find that employing a nonlinear neural network and utilizing a random forest yields an overall analogous picture in terms of the performance of all methods.

5. Discussion

This paper explores in-context learning for the purpose of full Bayesian inference in latent variable models. We propose to use conditional flow matching as a generic and flexible framework to approximate posterior distributions and an architecture that utilizes a transformer encoder for potentially complex conditioning on the data. We find that our ICL approach yields a closer approximation of the posterior than several state-of-the-art variational inference methods across different datasets and model setups. This does not only hold for synthetic, but also real-world tabular datasets.

Limitations. While our experiments indicate the effectiveness of ICL as a Bayesian inference method, it requires an extensive up-front training routine on modern GPU hardware. Despite ICL being consistently faster at inference time than the considered HMC methods, the overall computational burden to train our approach is much higher.

Furthermore, the goal of this work is to show that ICL can effectively learn full Bayesian inference. Our experiments therefore focus on relatively simple posterior distributions where we can compare against established methods, such as HMC. Additionally, increased dimensionality of the problems considered poses a challenge to both the ICL method and the metrics we employ. Further, as with many other ICL approaches, large datasets as a context can become computationally very expensive.

Outlook and Future Work. Despite its vast up-front computational cost, ICL has not only proven fundamentally transformative in the field of NLP (Brown et al., 2020; Touvron et al., 2023), but has recently started to transform the field of tabular machine learning (Hollmann et al., 2022). Exploring the frontiers of ICL in terms of full Bayesian inference, starting from the feasibility results of this work, might therefore lead to similarly fertile territories.

Although ICL performs well even when trained on data that may differ from real-world distributions, its flexibility is limited by the structure of the training data. If the synthetic data is highly unrealistic, ICL may fail— much like any model with a misspecified hypothesis space that imposes an unsuitable inductive bias.

While flexible state-of-the-art sampling-based methods, such as HMC, serve as an efficient and highly effective reference in terms of inference for standard and statistical methods discussed in this paper, the proposed ICL approach is fundamentally more general in nature. In particular, any probabilistic model for which a generative process is conceivable can be fitted using our ICL approach—the potential for fitting models beyond the horizon of standard Bayesian methods is therefore manifold.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning. There are many potential societal consequences of our work, none of which we feel must be specifically highlighted here.

Acknowledgements

We thank Beste Aydemir and Svea Reuter for their valuable support and insightful comments. VF was supported by the Branco Weiss Fellowship. DR's research is funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) – 548823575. We also gratefully acknowledge funding provided to AR by the Munich Center for Machine Learning (MCML).

References

Abdullah, A. A., Hassan, M. M., and Mustafa, Y. T. A review on bayesian deep learning in healthcare: Applications and challenges. *IEEE Access*, 10:36538–36562, 2022.

Ahuja, K., Panwar, M., and Goyal, N. In-context learning through the bayesian prism. *arXiv preprint arXiv:2306.04891*, 2023.

Anil, R., Borgeaud, S., Alayrac, J.-B., Yu, J., Soricut, R., Schalkwyk, J., Dai, A. M., Hauth, A., Millican, K., et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Betancourt, M. A conceptual introduction to hamiltonian monte carlo. *arXiv preprint arXiv:1701.02434*, 2017.

Bingham, E., Chen, J. P., Jankowiak, M., Obermeyer, F., Pradhan, N., Karaletsos, T., Singh, R., Szerlip, P., Horsfall, P., and Goodman, N. D. Pyro: Deep universal probabilistic programming. *Journal of machine learning research*, 20(28):1–6, 2019.

Bishop, C., Spiegelhalter, D., and Winn, J. Vibes: A variational inference engine for bayesian networks. *Advances in neural information processing systems*, 15, 2002.

Blei, D. M. Probabilistic topic models. *Communications of the ACM*, 55(4):77–84, 2012.

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

Brehmer, J. and Cranmer, K. Simulation-based inference methods for particle physics. In *Artificial Intelligence for High Energy Physics*, pp. 579–611. World Scientific, 2022.

Brosse, N., Durmus, A., and Moulines, E. The promises and pitfalls of stochastic gradient langevin dynamics. *Advances in Neural Information Processing Systems*, 31, 2018.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chan, S. C., Dasgupta, I., Kim, J., Kumaran, D., Lampinen, A. K., and Hill, F. Transformers generalize differently from information stored in context vs in weights. *arXiv preprint arXiv:2210.05675*, 2022.

Chen, R. T. Q. torchdiffeq, 2018. URL https://github.com/rtqichen/torchdiffeq.

Chen, T., Fox, E., and Guestrin, C. Stochastic gradient hamiltonian monte carlo. In *International conference on machine learning*, pp. 1683–1691. PMLR, 2014.

Cranmer, K., Brehmer, J., and Louppe, G. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

Cremer, C., Li, X., and Duvenaud, D. Inference suboptimality in variational autoencoders. In *International conference on machine learning*, pp. 1078–1086. PMLR, 2018.

Dao, Q., Phung, H., Nguyen, B., and Tran, A. Flow matching in latent space. *arXiv preprint arXiv:2307.08698*, 2023.

Dax, M., Green, S. R., Gair, J., Macke, J. H., Buonanno, A., and Schölkopf, B. Real-time gravitational wave science with neural posterior estimation. *Physical review letters*, 127(24):241103, 2021.

Dax, M., Green, S. R., Gair, J., Gupte, N., Pürrer, M., Raymond, V., Wildberger, J., Macke, J. H., Buonanno, A., and Schölkopf, B. Real-time gravitational-wave inference for binary neutron stars using machine learning. *arXiv preprint arXiv:2407.09602*, 2024.

Daxberger, E., Kristiadi, A., Immer, A., Eschenhagen, R., Bauer, M., and Hennig, P. Laplace redux-effortless bayesian deep learning. *Advances in Neural Information Processing Systems*, 34:20089–20103, 2021.

Dong, Q., Li, L., Dai, D., Zheng, C., Wu, Z., Chang, B., Sun, X., Xu, J., and Sui, Z. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*, 2022.

Dormand, J. R. and Prince, P. J. A family of embedded runge-kutta formulae. *Journal of computational and applied mathematics*, 6(1):19–26, 1980.

Eloundou, T., Manning, S., Mishkin, P., and Rock, D. Gpts are gpts: An early look at the labor market impact potential of large language models. *arXiv preprint arXiv:2303.10130*, 2023.

Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first International Conference on Machine Learning*, 2024.

Etzioni, R. D. and Kadane, J. B. Bayesian statistical methods in public health and medicine. *Annual review of public health*, 16(1):23–41, 1995.

Fahrmeir, L., Kneib, T., Lang, S., Marx, B., Fahrmeir, L., Kneib, T., Lang, S., and Marx, B. *Regression models*. Springer, 2013.

Fan, X. and Markram, H. A brief history of simulation neuroscience. *Frontiers in neuroinformatics*, 13:32, 2019.

Flamary, R., Courty, N., Gramfort, A., Alaya, M. Z., Boisbunon, A., Chambon, S., Chapel, L., Corenflos, A., Fatras, K., Fournier, N., et al. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8, 2021.

Garg, S., Tsipras, D., Liang, P. S., and Valiant, G. What can transformers learn in-context? a case study of simple function classes. *Advances in Neural Information Processing Systems*, 35:30583–30598, 2022.

Garnelo, M., Rosenbaum, D., Maddison, C., Ramalho, T., Saxton, D., Shanahan, M., Teh, Y. W., Rezende, D., and Eslami, S. A. Conditional neural processes. In *International conference on machine learning*, pp. 1704–1713. PMLR, 2018a.

Garnelo, M., Schwarz, J., Rosenbaum, D., Viola, F., Rezende, D. J., Eslami, S., and Teh, Y. W. Neural processes. *arXiv preprint arXiv:1807.01622*, 2018b.

Gebhard, T. D., Wildberger, J., Dax, M., Kofler, A., Angerhausen, D., Quanz, S. P., and Schölkopf, B. Flow matching for atmospheric retrieval of exoplanets: Where reliability meets adaptive noise levels. *Astronomy & Astrophysics*, 693:A42, 2025.

Givens, C. R. and Shortt, R. M. A class of wasserstein metrics for probability distributions. *Michigan Mathematical Journal*, 31(2):231–240, 1984.

Gloeckler, M., Deistler, M., Weilbach, C., Wood, F., and Macke, J. H. All-in-one simulation-based inference. *arXiv preprint arXiv:2404.09636*, 2024.

Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B., and Smola, A. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Grinsztajn, L., Oyallon, E., and Varoquaux, G. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35:507–520, 2022.

Grünwald, P. and van Ommen, T. Inconsistency of bayesian inference for misspecified linear models, and a proposal for repairing it. *Bayesian Analysis*, 12(4):1069–1103, 2017.

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

Hoffman, M. D., Gelman, A., et al. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014.

Hollmann, N., Müller, S., Eggensperger, K., and Hutter, F. Tabpfn: A transformer that solves small tabular classification problems in a second. *arXiv preprint arXiv:2207.01848*, 2022.

Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S. B., Schirrmeister, R. T., and Hutter, F. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, 2025.

Hoo, S. B., Müller, S., Salinas, D., and Hutter, F. The tabular foundation model tabpfn outperforms specialized time series forecasting models based on simple features. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024.

Hospedales, T., Antoniou, A., Micaelli, P., and Storkey, A. Meta-learning in neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 44(9):5149–5169, 2021.

Ioffe, S. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.

Izmailov, P., Vikram, S., Hoffman, M. D., and Wilson, A. G. G. What are bayesian neural network posteriors really like? In *International conference on machine learning*, pp. 4629–4640. PMLR, 2021.

Kim, Y., Wiseman, S., Miller, A., Sontag, D., and Rush, A. Semi-amortized variational autoencoders. In *International Conference on Machine Learning*, pp. 2678–2687. PMLR, 2018.

Kingma, D. P. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kingma, D. P. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.

Kirsch, L., Harrison, J., Sohl-Dickstein, J., and Metz, L. General-purpose in-context learning by meta-learning transformers. *arXiv preprint arXiv:2212.04458*, 2022.

Kucukelbir, A., Tran, D., Ranganath, R., Gelman, A., and Blei, D. M. Automatic differentiation variational inference. *Journal of machine learning research*, 18(14):1–45, 2017.

Kyrimi, E., McLachlan, S., Dube, K., Neves, M. R., Fahmi, A., and Fenton, N. A comprehensive scoping review of bayesian networks in healthcare: Past, present and future. *Artificial Intelligence in Medicine*, 117:102108, 2021.

Lawley, D. N. and Maxwell, A. E. Factor analysis as a statistical method. *Journal of the Royal Statistical Society. Series D (The Statistician)*, 12(3):209–229, 1962.

Le, T. A., Baydin, A. G., and Wood, F. Inference compilation and universal probabilistic programming. In *Artificial Intelligence and Statistics*, pp. 1338–1348. PMLR, 2017.

Li, C., Chen, C., Carlson, D., and Carin, L. Preconditioned stochastic gradient langevin dynamics for deep neural networks. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, pp. 1788–1794, 2016.

Lienen, M., Kollovieh, M., and Günnemann, S. Generative modeling with bayesian sample inference. *arXiv preprint arXiv:2502.07580*, 2025.

Lin, X., Wu, J., Zhou, C., Pan, S., Cao, Y., and Wang, B. Task-adaptive neural process for user cold-start recommendation. In *Proceedings of the Web Conference 2021*, pp. 1306–1316, 2021.

Lipman, Y., Chen, R. T., Ben-Hamu, H., Nickel, M., and Le, M. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Lopes, H. F. and West, M. Bayesian model assessment in factor analysis. *Statistica Sinica*, pp. 41–67, 2004.

Lopez-Paz, D. and Oquab, M. Revisiting classifier two-sample tests. *arXiv preprint arXiv:1610.06545*, 2016.

Loshchilov, I. and Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Lueckmann, J.-M., Goncalves, P. J., Bassetto, G., Öcal, K., Nonnenmacher, M., and Macke, J. H. Flexible statistical inference for mechanistic models of neural dynamics. *Advances in neural information processing systems*, 30, 2017.

Lueckmann, J.-M., Boelts, J., Greenberg, D., Goncalves, P., and Macke, J. Benchmarking simulation-based inference. In *International conference on artificial intelligence and statistics*, pp. 343–351. PMLR, 2021.

Mangoubi, O. and Vishnoi, N. K. Nonconvex sampling with the metropolis-adjusted langevin algorithm. In *Conference on learning theory*, pp. 2259–2293. PMLR, 2019.

Margossian, C. C. and Blei, D. M. Amortized variational inference: When and why? *arXiv preprint arXiv:2307.11018*, 2023.

Mittal, S., Bengio, Y., Malkin, N., and Lajoie, G. In-context parametric inference: Point or distribution estimators? *arXiv preprint arXiv:2502.11617*, 2025a.

Mittal, S., Bracher, N. L., Lajoie, G., Jaini, P., and Brubaker, M. Amortized in-context bayesian posterior estimation. *arXiv preprint arXiv:2502.06601*, 2025b.

Müller, S., Hollmann, N., Arango, S. P., Grabocka, J., and Hutter, F. Transformers can do bayesian-inference by meta-learning on prior-data. In *Fifth Workshop on Meta-Learning at the Conference on Neural Information Processing Systems*, 2021.

Murphy, K. P. *Probabilistic machine learning: Advanced topics*. MIT press, 2023.

Nelder, J. A. and Wedderburn, R. W. Generalized linear models. *Journal of the Royal Statistical Society Series A: Statistics in Society*, 135(3):370–384, 1972.

OpenAI. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021a.

Papamakarios, G., Nalisnick, E., Rezende, D. J., Mohamed, S., and Lakshminarayanan, B. Normalizing flows for probabilistic modeling and inference. *Journal of Machine Learning Research*, 22(57):1–64, 2021b.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4195–4205, 2023.

Phan, D., Pradhan, N., and Jankowiak, M. Composable effects for flexible and accelerated probabilistic programming in numpyro. *arXiv preprint arXiv:1912.11554*, 2019.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *International conference on machine learning*, pp. 1278–1286. PMLR, 2014.

Robertson, J., Hollmann, N., Awad, N., and Hutter, F. Fairpfn: Transformers can do counterfactual fairness. In *ICML 2024 Next Generation of AI Safety Workshop*, 2024.

Rudner, T. G., Fortuin, V., Teh, Y. W., and Gal, Y. On the connection between neural processes and gaussian processes with deep kernels. In *Workshop on Bayesian Deep Learning, NeurIPS*, pp. 14, 2018.

Rummel, R. J. *Applied factor analysis*. Northwestern University Press, 1988.

Sahoo, S., Gokaslan, A., De Sa, C. M., and Kuleshov, V. Diffusion models with learned adaptive noise. *Advances in Neural Information Processing Systems*, 37:105730–105779, 2024.

Salazar, S. Vart: variational regression trees. *Advances in Neural Information Processing Systems*, 36:45681–45693, 2023.

Schmit, C. J. and Pritchard, J. R. Emulation of reionization simulations for bayesian inference of astrophysics parameters using neural networks. *Monthly Notices of the Royal Astronomical Society*, 475(1):1213–1223, 2018.

Sohn, H. and Narain, D. Neural implementations of bayesian inference. *Current Opinion in Neurobiology*, 70: 121–129, 2021.

Sommer, E., Wimmer, L., Papamarkou, T., Bothmann, L., Bischl, B., and Rügamer, D. Connecting the dots: Is mode-connectedness the key to feasible sample-based inference in bayesian neural networks? In *Proceedings of the 41st International Conference on Machine Learning*. PMLR, 2024.

Sommer, E., Robnik, J., Nozadze, G., Seljak, U., and Rügamer, D. Microcanonical Langevin Ensembles: Advancing the Sampling of Bayesian Neural Networks. In *The Thirteenth International Conference on Learning Representations*, 2025.

Song, Y. and Ermon, S. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems*, 32, 2019.

Song, Y., Sohl-Dickstein, J., Kingma, D. P., Kumar, A., Ermon, S., and Poole, B. Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

Srivastava, A. and Sutton, C. Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*, 2017.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Vetter, J., Gloeckler, M., Gedon, D., and Macke, J. H. Effortless, simulation-efficient bayesian inference using tabular foundation models. *arXiv preprint arXiv:2504.17660*, 2025.

Walker, S. G. Bayesian inference with misspecified models. *Journal of statistical planning and inference*, 143(10): 1621–1633, 2013.

Wang, X., Zhu, W., Saxon, M., Steyvers, M., and Wang, W. Y. Large language models are latent variable models: Explaining and finding good demonstrations for incontext learning. *Advances in Neural Information Processing Systems*, 36, 2024.

Wang, Y. and Blei, D. Variational bayes under model misspecification. *Advances in Neural Information Processing Systems*, 32, 2019.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pp. 681–688. Citeseer, 2011.

Wildberger, J., Dax, M., Buchholz, S., Green, S., Macke, J. H., and Schölkopf, B. Flow matching for scalable simulation-based inference. *Advances in Neural Information Processing Systems*, 36, 2024.

Yeo, I.-K. and Johnson, R. A. A new family of power transformations to improve normality or symmetry. *Biometrika*, 87(4):954–959, 2000.

Yim, J., Campbell, A., Foong, A. Y., Gastegger, M., Jiménez-Luna, J., Lewis, S., Satorras, V. G., Veeling, B. S., Barzilay, R., Jaakkola, T., et al. Fast protein backbone generation with se (3) flow matching. *arXiv preprint arXiv:2310.05297*, 2023.

Yim, J., Campbell, A., Mathieu, E., Foong, A. Y., Gastegger, M., Jiménez-Luna, J., Lewis, S., Satorras, V. G., Veeling, B. S., Noé, F., et al. Improved motif-scaffolding with se (3) flow matching. *arXiv preprint arXiv:2401.04082*, 2024.

Zhai, J., Zhang, S., Chen, J., and He, Q. Autoencoder and its various variants. In *2018 IEEE international conference on systems, man, and cybernetics (SMC)*, pp. 415–419. IEEE, 2018.

Zhao, W., Shi, M., Yu, X., Zhou, J., and Lu, J. Flowturbo: Towards real-time flow-based image generation with velocity refiner. *arXiv preprint arXiv:2409.18128*, 2024.

Zheng, K., Lu, C., Chen, J., and Zhu, J. Improved techniques for maximum likelihood estimation for diffusion odes. In *International Conference on Machine Learning*, pp. 42363–42389. PMLR, 2023.

Appendix

A. Data-generating Processses

This section contains more details on the data generating processes of the latent variable models we fit via ICL.

A.1. Generalized Linear Models

In this section we expand the description and explanation regarding GLMs from Section 3.2. GLMs are among the most commonly used statistical models with myriads of applications (Nelder & Wedderburn, 1972; Fahrmeir et al., 2013). In the context of GLMs, we assume that the response y follows a distribution $P^{y|\boldsymbol{u}}$ depending on the linear predictor $\eta := \boldsymbol{u}^\top \boldsymbol{\beta}$ and an additional parameter σ^2. We denote the covariates as $\boldsymbol{u}$, the regression coefficients as $\boldsymbol{\beta}$, and use σ^2 for the variance of the response. The mean of $P^{y|\boldsymbol{u}}$ depends on the linear predictor via a link function g, such that $g\left(\mathbb{E}[y|\boldsymbol{u}]\right) = \boldsymbol{u}^\top \boldsymbol{\beta}$. Ultimately, the density of distribution of the response y depending on the linear predictor and the additional parameter is denoted by $p(y|g\left(\boldsymbol{u}^\top \boldsymbol{\beta}\right), \sigma^2)$. To showcase the flexibility of our framework, we experiment with different priors P^β on the regression coefficients, P^{σ^2} on the parameter σ^2, and also different parametric distributions of the response. Additionally, to include covariates $\boldsymbol{u}$ that resemble practically relevant tabular data in the generative process, allowing for meaningful inference on real-world datasets, we utilize samples from the Tab-PFN "prior" for $P^{\boldsymbol{u}}$.

GLMs belong to the framework of latent variable models defined by data $\boldsymbol{x}$ and (latent) variables $\boldsymbol{z}$, where the data comprises covariates and response $\boldsymbol{x} := (\boldsymbol{u}, y)$. The variables of interest are the coefficients $\boldsymbol{z} := \boldsymbol{\beta}$. This yields the following generative process for a set of synthetic samples $\mathcal{D} := \{(\boldsymbol{x}_i, \boldsymbol{z}_i)\}_{i=1}^N$ from $P^{\boldsymbol{x},\boldsymbol{z}}$:

We consider seven different GLM scenarios by varying the structure of the prior distributions and the conditional distribution of the response (Table 5). In particular, we consider a normal $\mathcal{N}(0,1)$ prior, a Laplace$(0,1)$ and a gamma Ga$(1,1)$ prior that factorizes over the coefficients β_j contained in $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_p)$. In two cases we include an intercept in the model using a normal prior $\mathcal{N}(0,9)$ with a relatively large variance. We consider regression cases with a normally distributed response $\mathcal{N}(\boldsymbol{u}^\top \boldsymbol{\beta}, \sigma^2)$, a Bernoulli distributed response Bin$(1, \text{sigmoid}(\boldsymbol{u}^\top \boldsymbol{\beta}))$, i.e. logistic regression, and a response following a gamma distribution Ga$(\sigma^{-2}\exp(\boldsymbol{u}^\top \boldsymbol{\beta}), \sigma^{-2}\exp(2\boldsymbol{u}^\top \boldsymbol{\beta}))$. In the last case, we set $\exp(\boldsymbol{u}^\top \boldsymbol{\beta})$ to be the mean and σ^2 to be the conditional variance of the response. An inverse gamma prior IG$(5,2)$ is used on the variance σ^2 for each scenario except the logistic regression. We fix the number of covariates and thus also the dimensionality of $\boldsymbol{\beta}$ at $p = 5$ and set the number of data points per dataset to $K = 50$.

Algorithm 2 Generation of synthetic data for GLMs

Require: Number of datasets N, number of samples per dataset K, distributions $P^{\boldsymbol{\beta}}, P^{\boldsymbol{\sigma^2}}, P^{\boldsymbol{u}}$,
Ensure: A dataset $\mathcal{D}$ of input-output pairs $(\boldsymbol{x}_i, \boldsymbol{z}_i)$ for $i = 1, \ldots, N$.

Initialize $\mathcal{D} \leftarrow \varnothing$
for $i = 1 \rightarrow N$ **do**
  Draw $\boldsymbol{\beta}_i \sim P^{\boldsymbol{\beta}}$
  Draw $\sigma_i^2 \sim P^{\boldsymbol{\sigma^2}}$
  for $j = 1 \rightarrow K$ **do**
    Draw $\boldsymbol{u}_{i,j} \sim P^{\boldsymbol{u}}$
    Draw $y_{i,j} \sim p\Big(y \mid g^{-1}\big(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i\big),\, \sigma_i^2\Big)$
  end for
  Set $\boldsymbol{x}_i := \big((\boldsymbol{u}_{i,j},\, y_{i,j})\big)_{j=1}^K$
  Set $\boldsymbol{z}_i := \boldsymbol{\beta}_i$
  Update $\mathcal{D} \leftarrow \mathcal{D} \cup \{\big(\boldsymbol{x}_i, \boldsymbol{z}_i\big)\}$
end for

Table 5: Distribution of variables for the considered GLM scenarios.

Scenario	$\beta_{i,j}$	$\beta_{i,0}$	σ_i^2	$y_{i,j}\|(\boldsymbol{u}_{i,j}, \boldsymbol{\beta}_i, \beta_{0,i}, \sigma_i^2)$
Scenario 1	$\mathcal{N}(0,1)$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 2	$\mathcal{N}(0,1)$	$\mathcal{N}(0,9)$	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 3	Laplace$(0,1)$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 4	Laplace$(0,1)$	$\mathcal{N}(0,9)$	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 5	Ga$(1,1)$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 6	$\mathcal{N}(0,1)$	-	-	Bin$(1, \text{sigmoid}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i))$
Scenario 7	$\mathcal{N}(0,1)$	-	IG$(5,2)$	Ga$(\sigma_i^{-2} \exp(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i), \sigma_i^{-2} \exp(2\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i))$

A.2. Factor Analysis

The goal of factor analysis is to explain data $\boldsymbol{x}$ in terms of latent, typically lower-dimensional, factors $\boldsymbol{z}$ (Lawley & Maxwell, 1962; Rummel, 1988). In the Bayesian setting, one assumes a prior $P^{\boldsymbol{z}}$ on the latent variable $\boldsymbol{z}$, a prior $P^{\boldsymbol{W}}$ on the factor loading matrix $\boldsymbol{W}$ and additional priors $P^{\boldsymbol{\Psi}}$ and $P^{\boldsymbol{\mu}}$ on the covariance matrix and the mean vector. The conditional distribution $P^{\boldsymbol{z}|\boldsymbol{x}}$ of the data given $\boldsymbol{z}$ has mean $\mathbb{E}[\boldsymbol{z}|\boldsymbol{x}] = \boldsymbol{W}\boldsymbol{z} + \boldsymbol{\mu}$ and covariance matrix $\text{Cov}[\boldsymbol{z}|\boldsymbol{x}] = \boldsymbol{\Psi}$. In the case where $P^{\boldsymbol{z}}$ and $P^{\boldsymbol{z}|\boldsymbol{x}}$ are Gaussian, one can set $P^{\boldsymbol{z}} = \mathcal{N}(\boldsymbol{0}, I)$ and assume a diagonal covariance matrix $\boldsymbol{\Psi}$ without loosing expressiveness of the model (Murphy, 2023). We make the assumption that $\boldsymbol{W}$ is lower triangular with positive entries on the diagonal in order to ensure identifiability of the model (Lopes & West, 2004). Additionally, we assume that the distributions $\boldsymbol{\mu}$, $\boldsymbol{\Psi}$ and $P^{\boldsymbol{W}}$ fully factorize. In order to ensure that the diagonal of $\boldsymbol{W}$ is positive, we consider absolute values in the generative process. Algorithm 3 details the data generating process.

Table 6 summarizes the different configurations for FA. We assume a Gaussian prior on the mean components, and an inverse gamma prior on the elements of the diagonal covariance matrix $\boldsymbol{\Psi}$. For the factor loading matrix $\boldsymbol{W}$, independent normal and Laplace priors are investigated. Furthermore, we use a normal prior on the latent factors $\boldsymbol{z}_i$ in five cases and a Laplace prior in one case. We vary the number of samples K per dataset $\boldsymbol{x}$, the dimensionality P of each data point, as well as the dimensionality $\boldsymbol{z}_{dim}$.

Table 6: Distribution and dimensionalitites of variables for the considered FA scenarios.

Scenario	K	P	$\mu_{i,j}$	$\Psi_{i,j,j}$	$W_{i,j,k}$	$z_{i,j}$	$\boldsymbol{z}_{dim}$
Scenario 1	50	3	$\mathcal{N}(0,1)$	IG$(5,1)$	$\mathcal{N}(0,1)$	$\mathcal{N}(0,1)$	3
Scenario 2	50	3	$\mathcal{N}(0,0.1)$	IG$(5,1)$	Laplace$(0,10)$	$\mathcal{N}(0,1)$	3
Scenario 3	25	5	$\mathcal{N}(0,0.1)$	IG$(5,2)$	$\mathcal{N}(0,3)$	$\mathcal{N}(0,1)$	3
Scenario 4	25	15	$\mathcal{N}(0,0.1)$	IG$(5,2)$	$\mathcal{N}(0,3)$	$\mathcal{N}(0,1)$	5
Scenario 5	25	5	$\mathcal{N}(0,0.1)$	IG$(5,2)$	Laplace$(0,3)$	$\mathcal{N}(0,1)$	3
Scenario 6	25	5	$\mathcal{N}(0,0.1)$	IG$(5,2)$	$\mathcal{N}(0,3)$	Laplace$(0,1)$	3

Algorithm 3 Generation of synthetic data for FA

Require: Number of datasets N, number of samples K, and distributions $P^{\boldsymbol{\mu}}, P^{\boldsymbol{\Psi}}, P^{\boldsymbol{W}}, P^{\boldsymbol{z}}$.
Ensure: A dataset $\mathcal{D}$ containing $(\boldsymbol{x}_i, \boldsymbol{z}_i)$ for $i = 1, \ldots, N$.

```
Initialize D ← ∅
for i = 1 → N do
    Draw μ_i ~ P^μ
    Draw Ψ_i ~ P^Ψ
    Draw W_i ~ P^W
    Draw z_i ~ P^z
    for j = 1 → K do
        Draw x_{i,j} ~ N(W_i z_i + μ_i, Ψ_i)
    end for
    Update D ← D ∪ {(x_i, z_i)}
end for
```

A.3. Gaussian Mixture Models

Table 7: Distribution and dimensionalitites of variables for the considered GMM scenarios.

Scenario	K	M	L	$\boldsymbol{\phi}_i$	$\sigma^2_{i,m,l}$	$\mu_{i,m,l}\|\sigma^2_{i,m,l}$
Scenario 1	50	5	1	$\mathrm{Dir}(1)$	$\mathrm{IG}(5,2)$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$
Scenario 2	25	3	3	$\mathrm{Dir}(1)$	$\mathrm{IG}(5,2)$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$
Scenario 3	50	3	5	$\mathrm{Dir}(0.5)$	$\mathrm{IG}(5,2)$	$\mathcal{N}(0, 5\sigma^2_{i,m,l})$
Scenario 4	50	3	3	$\mathrm{Dir}(1)$	$\mathrm{IG}(5,2)$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$

In GMMs one assumes that the data of interest is generated by a convex combination of M (multivariate) normal distributions, such that $p(\boldsymbol{x}|\boldsymbol{z}) = \sum_{m=1}^{M} \boldsymbol{\phi}_m p_m(\boldsymbol{x})$, where the probability vector $\boldsymbol{\phi} = (\phi_1, \ldots, \phi_M)$ comprises the mixture weights and p_m denotes the m-th mixture component. We consider p_m to take the form of a diagonal Gaussian with mean vector $\boldsymbol{\mu}_m$ and covariance matrix with diagonal elements $\boldsymbol{\sigma}^2_m$. We assume a prior $P^{\boldsymbol{\phi}}$ on $\boldsymbol{\phi}$, a prior $P^{\boldsymbol{\sigma}^2}$ on the variances of each component and a prior $P^{\boldsymbol{\mu}|\boldsymbol{\sigma}^2}$ for the means that depends on the variance of the respective component. More specifically, we assume a symmetric Dirichlet prior on $\boldsymbol{\phi}$ such that $P^{\boldsymbol{\phi}} = \mathrm{Dir}(\alpha_{Dir})$ and an independent inverse gamma distribution as prior on each component σ^2_m of $\boldsymbol{\sigma}^2_m$. The prior on each component of $\boldsymbol{\mu}_{i,m} \in \mathbb{R}^L$ is then given by an independent normal distribution $P^{\boldsymbol{\mu}|\boldsymbol{\sigma}^2_{i,m,l}} = \mathcal{N}(0, \lambda\sigma^2_{i,m,l})$. We use $\omega_{i,j}$ to denote the assignment of datapoint j a component. Algorithm 4 details the data generating process and Table 7 summarizes the different setups regarding the prior distributions.

Algorithm 4 Generation of synthetic data for a GMM.

Require: Number of datasets N, mixture dimension parameters M, L, number of samples K, and distributions $P^{\boldsymbol{\phi}}$, $P^{\boldsymbol{\sigma}^2}$, $P^{\boldsymbol{\mu}|\boldsymbol{\sigma}^2}$.

Ensure: A dataset $\mathcal{D}$ containing $(\boldsymbol{x}_i, \boldsymbol{z}_i)$ for $i = 1, \ldots, N$.

Initialize $\mathcal{D} \leftarrow \varnothing$
for $i = 1 \rightarrow N$ **do**
  Draw $\boldsymbol{\phi}_i \sim P^{\boldsymbol{\phi}}$
  for $m = 1 \rightarrow M$ **do**
    for $l = 1 \rightarrow L$ **do**
      Draw $\sigma^2_{i,m,l} \sim P^{\boldsymbol{\sigma}^2}$
      Draw $\mu_{i,m,l} \sim P^{\boldsymbol{\mu}|\boldsymbol{\sigma}^2_{i,m,l}}$
    end for
  end for
  for $j = 1 \rightarrow K$ **do**
    Draw $\omega_{i,j} \sim \mathrm{Cat}\big(\boldsymbol{\phi}_i\big)$
    Draw $\boldsymbol{x}_{i,j} \sim \mathcal{N}\Big(\boldsymbol{\mu}_{i,\omega_{i,j}}, \boldsymbol{\sigma}^2_{i,\omega_{i,j}}\Big)$
  end for
  Set $\boldsymbol{z}_i := \Big(\big(\sigma^2_{i,m,l},\, \mu_{i,m,l}\big)\Big)_{\substack{m=1,\ldots,M \\ l=1,\ldots,L}}$
  Update $\mathcal{D} \leftarrow \mathcal{D} \cup \big\{(\boldsymbol{x}_i, \boldsymbol{z}_i)\big\}$
end for

B. Generating Realistic Data

While we assume a data-generating process such as the one in Algorithm 2, this is not necessarily the data-generating process that produces the data in the model's application as an in-context learner. Even when the generative process $P^{\boldsymbol{x},\boldsymbol{z}}$ underlying a statistical model is sophisticated and complex in nature, model misspecification is inevitable in almost every practical application. While mismatches between the real data-generating processes and model assumptions can lead to various problems in traditional Bayesian modeling (Grünwald & van Ommen, 2017), the question of model misspecification plays a somewhat different and yet an especially central role for our ICL approach.

More specifically, the ICL model learns the relationship between $P^{\boldsymbol{z}|\boldsymbol{x}}$ and a datapoint $\boldsymbol{x}$ exclusively based on synthetic samples from the marginal $P^{\boldsymbol{x}}$ implied by the statistical model with generative process $P^{\boldsymbol{x},\boldsymbol{z}}$. Given a real-world dataset $\boldsymbol{x}^* \sim P^{\boldsymbol{x}^*}$, model misspecification in terms of $P^{\boldsymbol{x}^*}$ implies that the in-context learner needs to infer the posterior based on out-of-distribution data, where the problem is aggravated the more unrealistic $P^{\boldsymbol{x}}$ is.

To be able to access a reference or ground truth distribution, the data generating processes in our experiments need to match the structure of the GLM, FA and GMM approaches. While the generative processes of FA and GMMs directly prescribe how all parts of the data are generated, this can potentially cause a discrepancy between synthetically generated and real-world datasets. However, our empirical results (Section 4.1) demonstrate that the in-context learner can generalize to real-world data despite the discrepancy to the simulated datasets.

In the aforementioned GLM case, the distribution of the covariates $P^{\boldsymbol{u}}$ does not affect the structure of $P^{\boldsymbol{z}|\boldsymbol{x}}$ in the data generating process (cf. Algorithm 2). We can therefore use a flexible prior $P^{\boldsymbol{u}}$ such as the TabPFN-"prior" (Hollmann et al., 2022) to generate covariates $\boldsymbol{u}$ and thereby effectively tackle the issue of model specification.

C. Preprocessing of the Real-world Datasets

The real-world datasets considered for the evaluation of all methods are proposed in a benchmark study by Grinsztajn et al. (2022). We standardize all features, scale and shift the target such that it has the mean and variance implied by the prior structure of the respective generative model. Furthermore, for the GLM scenarios, we apply a Yeo-Johnson transform on the target variable (Yeo & Johnson, 2000) before applying the scaling. In cases where the number of features in the real-world dataset exceeds that of our scenario, we select those features with the most distinct values in the original dataset and randomly sub-sample the appropriate number of samples from the real-world datasets for our experiments.

D. Background on Conditional Flow-matching

Flow matching, initially used in image synthesis leverages normalizing flows (Papamakarios et al., 2021b) to model arbitrary distributions. Continuous normalizing flows (Lipman et al., 2022) have emerged as a potent tool for modeling complex distributions. For example, recent advancements have shown its effectiveness in state-of-the-art image generation, outperforming diffusion-based methods in likelihood and sample quality on ImageNet (Lipman et al., 2022). Techniques like FlowTurbo have accelerated class-conditional and text-to-image generation, setting new benchmarks (Zhao et al., 2024). Additionally, applying flow matching in latent spaces of pretrained autoencoders has enhanced computational efficiency and scalability for high-resolution image synthesis (Dao et al., 2023). Similarly, flow-based models have been successfully applied to protein structure prediction, improving accuracy and efficiency in modeling complex protein conformations (Yim et al., 2024; 2023).

In the area of simulation-based inference, Wildberger et al. (2024) introduce the idea of using continuous normalizing flows in order to efficiently approximate complex posterior distributions. In particular, they apply the framework to the field of gravitational-wave inference, substantially outperforming approaches based on discrete flows. Furthermore, they demonstrate good performance on the existing SBI-Benchmark (Lueckmann et al., 2021) using a simple MLP-based architecture.

E. Relationship of our approach to density estimation methods

An alternative to flow matching for parameterizing our model $Q^{\boldsymbol{z}|\boldsymbol{x}}$ would be explicit (conditional) density estimation. While knowledge of the explicit density of a distribution can be useful for downstream tasks, we would like to reemphasize that we consider the problem of full Bayesian inference via *sampling* from the posterior in this paper.

Popular explicit density estimation methods include i-DODE (Zheng et al., 2023), which proposes several techniques for improving maximum likelihood estimation from diffusion ordinary differential equations (ODEs), including velocity parameterization and techniques for variance reduction, leading to faster convergence. Additionally,in (Sahoo et al., 2024) log-likelihood estimation is improved by casting the learned diffusion process as a variational posterior that yields a tighter evidence lower bound on the actual likelihood. (Lienen et al., 2025) propose a novel generative model using iterative Gaussian posterior inference and empirically demonstrate that it yields strong results in log-likelihood estimation. Furthermore, Salazar (2023) use variational inference to learn Bayesian regression trees, which could be used for multivariate density estimation.

Note that it is, in principle, also possible to recover the explicit density of $Q^{\boldsymbol{z}|\boldsymbol{x}}$, which is parametrized in the Flow Matching Framework that is used for our approach (Wildberger et al., 2024).

F. Hyperparameters, Software and Computational Setup

In this section, we detail the hyperparameters, used software and computational setups for all our experiments.

F.1. ICL

To ensure maximum comparability across different experiments, we fix the hyperparameters for all ICL experiments: For the architecture of the model introduced in Section 3.3, we use the following configuration: The dimensionality of encoder representations is set to 512 and is expanded to 1024 in the feed-forward blocks. We use 8 heads and 8 encoder layers with a dropout rate of 0.1. For the decoder part we also use 512 as the dimensionality of the representations and 1024 as the intermediate representation in the feed-forward layers and a dropout rate of 0.1. Furthermore, 3 simple fully connected layers with adaLN conditioning are used for final processing in the decoder. For the time conditioning, we use 3 simple fully connected layers to map the scalar-valued time t onto a 512 dimensional conditioning vector that is used for the adaLN blocks in the decoder. This yields a model of around 43.1 million parameters. We use no tokenization for either the encoder or the decoder and simple embedding layers to map the encoder- and decoder-input onto the feed-forward dimensions.

We use an Adam optimizer (Kingma, 2014) with a cosine learning rate schedule (Loshchilov & Hutter, 2016), where the maximum learning rate is $5 \cdot 10^{-4}$, the final division factor is 10^4 and 10 percent of the epochs are used for warm-up. We use a weight decay parameter of 10^{-5} and a batch size of 1024 and gradient clipping with a maximum gradient norm of one. We use in total 75 million synthetic samples for all scenarios. Of the total number, half, i.e. 37.5 million, are used for training and 10 percent for validation and the remaining 40 percent for testing. Note that we observe convergence of the loss usually much earlier than after this training duration, but fix the number of samples for consistency across experiments. A single L4 GPU is used for the GLM scenarios and a single A100 GPU for the FA and GMM cases.

To solve the ODE for the sample generation, dopri5 (Dormand & Prince, 1980) as implemented in Torchdiffeq (Chen, 2018) is used in the adjoint version. We set the relative and absolute tolerance to 10^{-7}. The σ_{min} parameter in the CNF-loss is set to 10^{-4}.

F.2. HMC

We use HMC with a NUTS kernel (Hoffman et al., 2014) as a reference for all experiments where no analytical solution is available. We set the number of burn-in samples to 500 and use one chain for all uni-modal problems and three times the number of potential modes in all other cases. More specifically, we use $M \times 3$ chains for all GMM scenarios. The Pyro implementation of NUTS is used for the GLM scenarios (Bingham et al., 2019) and the conceptually identical, albeit computationally faster implementation in Numpyro for the FA and GMM cases (Phan et al., 2019).

F.3. VI

For the variational inference methods, we utilize automatic guide generation based on the ground-truth data-generating processes (Kucukelbir et al., 2017). Pyro is used for the implementation of the probabilistic programs, which we also use to sample the synthetic training data, for the automatic guide generation, and for the implementation of the actual VI methods (Bingham et al., 2019). Default hyperparameters, as well as an Adam optimizer (Kingma, 2014) with a learning rate of 10^{-2} is used for all methods except for AutoIAF where a learning rate of 10^{-3} is used. We perform 2000 full-batch gradient update steps for each method.

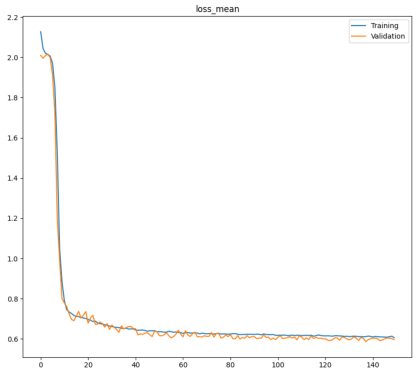


Figure 4: Learning curves for GLM scenario 1 with a Normal Prior on the coefficients $\boldsymbol{\beta}$ and an Inverse Gamma prior on σ_2.

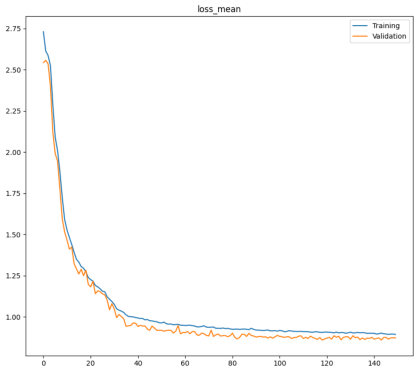


Figure 5: Learning curves for GMM scenario 1 with $M = 5$ components, $K = 50$ datapoints and $L = 1$ dimensions.

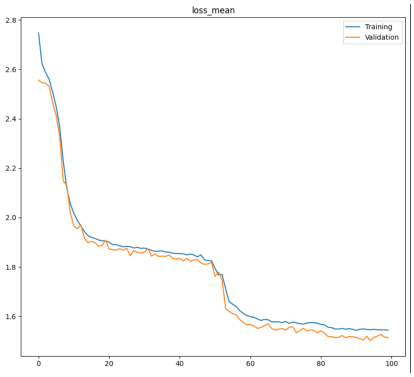


Figure 6: Learning curves for GMM scenario 3 with $M = 3$ components, $K = 50$ datapoints and $L = 5$ dimensions.

G. Runtimes

We use a single L4 GPU for generating samples based on our ICL approach and HMC in the GLM scenarios, a single A100 for our ICL approach and HMC in the FA and GMM scenarios, and an Intel(R) Xeon(R) CPU @ 2.20GHz CPU with two virtual cores and 40 gigabytes of RAM for the VI methods. Across all considered GLM scenarios, pre-training takes on average 14.89 hours with a standard error of 18.01 minutes. For the FA scenarios, on average 3.95 hours with a standard error of 11.38 minutes is used for pretraining and for the GMM scenarios 10.63 with a standard error of 72.88 minutes.

When applied in order to generate samples for a new dataset, the benchmarked VI methods have, as expected the lowest runtime. The Laplace approximation is the fastest of all methods, while our ICL appraoch has consistently a lower runtime compared to HMC. Overall, the ICL method takes around 2 minutes on the GLM tasks, around 30 seconds in the FA scenarios and less than 2 minutes for the inference regarding the GMM tasks.

This difference is especially pronounced in the FA and GMM scenarios. Please note that the runtime of the ICL method also fundamentally depends on the used precision for solving the underlying differential equation where we use a relatively high relative and absolute precision of 10^{-7}. Decreasing this value might lead to significantly faster inference time while maintaining sample quality.

Table 8: Runtime Metrics for all GLM, FA, and GMM Scenarios

Scenario	Method	Mean Runtime (s)
GLM	Laplace Approximation	$10.48\,(\pm 0.25)$
	VI: DiagonalNormal	$12.02\,(\pm 0.26)$
	VI: MultivariateNormal	$13.70\,(\pm 0.29)$
	VI: Structured Normal	$19.81\,(\pm 0.98)$
	VI:IAF	$15.44\,(\pm 0.30)$
	HMC	$120.24\,(\pm 13.94)$
	ICL (ours)	$107.79\,(\pm 17.36)$
FA	Laplace Approximation	$17.85\,(\pm 0.21)$
	VI: DiagonalNormal	$20.94\,(\pm 0.66)$
	VI: MultivariateNormal	$20.84\,(\pm 0.28)$
	VI: Structured Normal	$36.17\,(\pm 0.61)$
	VI:IAF	$23.75\,(\pm 0.38)$
	HMC	$248.26\,(\pm 57.88)$
	ICL (ours)	$31.49\,(\pm 4.97)$
GMM	Laplace Approximation	$27.52\,(\pm 0.40)$
	VI: DiagonalNormal	$29.74\,(\pm 0.57)$
	VI: MultivariateNormal	$30.50\,(\pm 0.41)$
	VI: Structured Normal	$42.44\,(\pm 0.44)$
	VI:IAF	$33.39\,(\pm 0.49)$
	HMC	$239.67\,(\pm 32.71)$
	ICL (ours)	$93.88\,(\pm 10.47)$

H. Ablation: Different Learning Rates for VI

To investigate the role of the learning rate parameter for the benchmarked VI methods, we record the performance for learning-rate values of 10^{-2}, 10^{-3} and 10^{-4} across a prototypical GLM, a FA and a GMM scenario, where we use 10 synthetic and 10 real-world datasets. In summary, while we find the VI methods to often be quite robust to the choice of the learning rate, those results also confirm our choice of setting the learning rate to 10^{-2} for the Laplace approximation, variational inference with a diagonal normal distribution, a multivariate normal distribution and a structured normal distribution, and to a value of 10^{-3} for the VI approach with inverse autoregressive flows.

For the GLM-scenario, we find in terms of the C2ST metric that VI with an ordinary multivariate normal distribution and VI with a structured normal distribution and a learning rate of 10^{-2} are the best models on the synthetic data. While MMD also indicates that this learning rate yields ideal results for those models, VI with inverse auoregressive flows has good values across the different learning rates with the minimum for 10^{-3}. The $\mathcal{W}_2$ metric indicates a similar tendency.

Regarding the learning rate for the FA scenario, one can first see that no single learning rate seems to dominate substantially given the variance of the results. However, on the synthetic data for the Laplace approximation, as well as VI with a diagonal normal distribution, a multivariate normal and a structured normal distribution, the lowest average result is obtained for a learning rate of 10^{-2}, while for VI with inverse autoregressive flows the best performance is obtained when the learning rate equals 10^{-3}. The real-world results are the best for VI with a structured normal distribution and a learning rate of 10^{-2}.

For the GMM scenario, we find that VI with a diagonal, structured and ordinary normal distribution obtain the best results, namely for learning rates of 10^{-2} and 10^{-3}, taking the variance into account. Just considering the averages leads to the conclusion that 10^{-2} is the best choice here. The results on the real-world data confirm that 10^{-2} is the optimal choice for VI with a diagonal normal and ordinary multivariate normal, while VI with inverse autoregressive flows has good results across all choices regarding the learning rate.

Table 9: Results of VI methods with different learning rates on 10 synthetic and 10 real-world datasets: Linear regression with a normal prior on the coefficients $\boldsymbol{\beta}$ and an inverse gamma prior on the variance σ^2 (scenario 1). Comparison to HMC samples. All results within two standard errors of the best average result are marked in **bold**.

Model	LR	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Laplace Approximation	1e-2	1.000 (± 0.000)	2.342 (± 0.390)	2.121 (± 0.100)	1.000 (± 0.000)	2.134 (± 0.107)	2.095 (± 0.062)
Laplace Approximation	1e-3	1.000 (± 0.000)	2.341 (± 0.389)	2.121 (± 0.100)	1.000 (± 0.000)	2.133 (± 0.108)	2.095 (± 0.062)
Laplace Approximation	1e-4	1.000 (± 0.000)	2.341 (± 0.389)	2.121 (± 0.100)	1.000 (± 0.000)	2.133 (± 0.108)	2.095 (± 0.062)
VI: DiagonalNormal	1e-2	0.892 (± 0.074)	0.921 (± 0.374)	**1.411** (± 0.174)	0.889 (± 0.062)	0.819 (± 0.343)	1.339 (± 0.190)
VI: DiagonalNormal	1e-3	0.966 (± 0.024)	1.588 (± 0.540)	**1.672** (± 0.203)	0.981 (± 0.017)	1.685 (± 0.331)	1.739 (± 0.139)
VI: DiagonalNormal	1e-4	0.971 (± 0.010)	1.572 (± 0.300)	1.666 (± 0.081)	0.849 (± 0.030)	0.575 (± 0.127)	**1.221** (± 0.098)
VI: MultivariateNormal	1e-2	**0.725** (± 0.064)	**0.523** (± 0.242)	**1.114** (± 0.261)	**0.625** (± 0.051)	**0.470** (± 0.066)	**0.918** (± 0.119)
VI: MultivariateNormal	1e-3	0.964 (± 0.008)	1.455 (± 0.327)	1.617 (± 0.100)	0.853 (± 0.052)	0.634 (± 0.266)	**1.238** (± 0.151)
VI: MultivariateNormal	1e-4	0.984 (± 0.005)	1.848 (± 0.324)	1.773 (± 0.079)	0.899 (± 0.020)	0.807 (± 0.094)	**1.345** (± 0.079)
VI: Structured Normal	1e-2	**0.734** (± 0.063)	**0.541** (± 0.254)	**1.119** (± 0.264)	**0.670** (± 0.047)	**0.467** (± 0.086)	**1.060** (± 0.130)
VI: Structured Normal	1e-3	0.882 (± 0.042)	0.719 (± 0.315)	1.335 (± 0.149)	0.776 (± 0.045)	**0.473** (± 0.081)	**1.064** (± 0.131)
VI: Structured Normal	1e-4	0.890 (± 0.027)	0.710 (± 0.290)	1.347 (± 0.138)	0.771 (± 0.049)	**0.468** (± 0.078)	**1.062** (± 0.128)
VI: IAF	1e-2	0.840 (± 0.036)	**0.502** (± 0.262)	**1.272** (± 0.170)	**0.614** (± 0.045)	**0.455** (± 0.048)	**0.957** (± 0.105)
VI: IAF	1e-3	0.797 (± 0.065)	**0.485** (± 0.556)	**1.169** (± 0.313)	**0.619** (± 0.036)	**0.469** (± 0.064)	**0.989** (± 0.124)
VI: IAF	1e-4	0.803 (± 0.068)	**0.475** (± 0.535)	**1.162** (± 0.291)	**0.612** (± 0.034)	**0.457** (± 0.055)	**0.977** (± 0.113)

Table 10: Results of VI methods with different learning rates on 10 synthetic and 10 real-world datasets: Factor analysis with Gaussian priors on the weights and the latents and $K = 25$ datapoints, $P = 5$ features, and dimensionality of the latents $\mathbf{z}_{dim} = 3$ (scenario 3). Comparison to HMC samples. All results within two standard errors of the best average result are marked in **bold**.

Model	LR	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Laplace Approximation	1e-2	1.000 (± 0.000)	3.449 (± 0.821)	**1.773** (± 0.539)	1.000 (± 0.000)	2.703 (± 0.312)	**0.362** (± 0.017)
Laplace Approximation	1e-3	1.000 (± 0.000)	4.288 (± 0.853)	**2.263** (± 0.732)	1.000 (± 0.000)	2.896 (± 0.238)	**0.376** (± 0.022)
Laplace Approximation	1e-4	1.000 (± 0.000)	4.252 (± 0.611)	**2.122** (± 0.430)	1.000 (± 0.000)	2.805 (± 0.181)	**0.368** (± 0.017)
VI: DiagonalNormal	1e-2	0.998 (± 0.002)	2.880 (± 1.046)	**1.457** (± 0.559)	0.944 (± 0.008)	1.022 (± 0.067)	**0.230** (± 0.010)
VI: DiagonalNormal	1e-3	0.998 (± 0.002)	2.973 (± 0.834)	**1.465** (± 0.540)	0.941 (± 0.006)	0.997 (± 0.056)	**0.229** (± 0.010)
VI: DiagonalNormal	1e-4	1.000 (± 0.001)	3.416 (± 0.761)	**1.602** (± 0.437)	0.943 (± 0.009)	0.997 (± 0.057)	**0.229** (± 0.010)
VI: MultivariateNormal	1e-2	0.993 (± 0.007)	2.969 (± 1.089)	**1.506** (± 0.659)	**0.929** (± 0.007)	**0.957** (± 0.048)	**0.224** (± 0.010)
VI: MultivariateNormal	1e-3	0.996 (± 0.004)	3.140 (± 0.910)	**1.570** (± 0.625)	0.934 (± 0.009)	**0.971** (± 0.054)	**0.225** (± 0.010)
VI: MultivariateNormal	1e-4	0.997 (± 0.007)	3.464 (± 0.791)	**1.639** (± 0.426)	0.934 (± 0.005)	**0.962** (± 0.049)	**0.225** (± 0.010)
VI: Structured Normal	1e-2	0.998 (± 0.002)	3.005 (± 0.871)	**1.481** (± 0.504)	0.947 (± 0.005)	1.003 (± 0.066)	**0.230** (± 0.009)
VI: Structured Normal	1e-3	0.999 (± 0.001)	3.244 (± 0.665)	**1.619** (± 0.559)	0.948 (± 0.007)	1.033 (± 0.078)	**0.232** (± 0.009)
VI: Structured Normal	1e-4	0.999 (± 0.001)	3.119 (± 0.612)	**1.487** (± 0.400)	0.943 (± 0.007)	0.998 (± 0.056)	**0.229** (± 0.010)
VI: IAF	1e-2	**0.939** (± 0.040)	**2.836** (± 0.293)	**1.247** (± 0.297)	0.944 (± 0.008)	1.518 (± 0.048)	1.332 (± 0.027)
VI: IAF	1e-3	**0.927** (± 0.047)	**2.758** (± 0.342)	**1.195** (± 0.331)	0.949 (± 0.009)	1.560 (± 0.031)	**1.392** (± 0.024)
VI: IAF	1e-4	**0.842** (± 0.038)	**2.862** (± 0.296)	**1.281** (± 0.292)	0.943 (± 0.008)	1.493 (± 0.039)	1.302 (± 0.039)

Table 11: Results of VI methods with different learning rates on 10 synthetic and 10 real-world datasets: Gaussian Mixture Model with $K = 50$ datapoints, $L = 1$ features (univariate case), $M = 5$ components, $\lambda = 3$, and $\alpha_{dir} = 1$ (scenario 1). Comparison to HMC samples.All results within two standard errors of the best average result are marked in **bold**.

Model	**LR**	**Synthetic Evaluation**			**Real-World Evaluation**		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Laplace Approximation	1e-2	1.000 (± 0.000)	4.380 (± 1.386)	**4.838** (± 1.521)	1.000 (± 0.000)	4.588 (± 1.229)	**6.813** (± 1.697)
Laplace Approximation	1e-3	1.000 (± 0.000)	3.893 (± 1.433)	**4.010** (± 1.233)	1.000 (± 0.000)	4.699 (± 1.193)	**6.986** (± 0.981)
Laplace Approximation	1e-4	1.000 (± 0.000)	4.463 (± 1.117)	**4.610** (± 1.027)	1.000 (± 0.000)	4.710 (± 1.205)	**6.995** (± 0.869)
VI: DiagonalNormal	1e-2	**0.979** (± 0.138)	**1.370** (± 1.394)	**3.522** (± 1.634)	**0.985** (± 0.030)	2.384 (± 1.318)	**6.202** (± 1.747)
VI: DiagonalNormal	1e-3	**0.990** (± 0.096)	**1.454** (± 1.454)	**3.650** (± 1.743)	0.999 (± 0.002)	3.026 (± 0.977)	**6.959** (± 0.890)
VI: DiagonalNormal	1e-4	1.000 (± 0.001)	2.390 (± 1.177)	**4.903** (± 1.278)	0.998 (± 0.007)	2.830 (± 1.001)	**7.007** (± 0.987)
VI: MultivariateNormal	1e-2	**0.978** (± 0.119)	**1.351** (± 1.410)	**3.474** (± 1.604)	**0.987** (± 0.024)	2.375 (± 1.304)	**6.189** (± 1.761)
VI: MultivariateNormal	1e-3	**0.980** (± 0.089)	**1.476** (± 1.480)	**3.681** (± 1.734)	0.997 (± 0.008)	2.808 (± 1.014)	**6.964** (± 0.944)
VI: MultivariateNormal	1e-4	1.000 (± 0.001)	2.114 (± 1.140)	**4.532** (± 1.187)	0.997 (± 0.007)	2.799 (± 1.012)	**6.963** (± 0.950)
VI: Structured Normal	1e-2	**0.958** (± 0.129)	**1.246** (± 1.615)	**3.225** (± 1.701)	1.000 (± 0.001)	2.911 (± 0.753)	**6.675** (± 1.403)
VI: Structured Normal	1e-3	**0.979** (± 0.092)	**1.593** (± 1.561)	**3.395** (± 1.440)	0.998 (± 0.007)	2.882 (± 1.070)	**6.968** (± 0.941)
VI: Structured Normal	1e-4	1.000 (± 0.001)	2.270 (± 1.133)	**4.733** (± 1.162)	0.997 (± 0.009)	2.802 (± 1.012)	**6.953** (± 0.948)
VI: IAF	1e-2	0.998 (± 0.003)	**1.539** (± 0.691)	**8.371** (± 0.750)	**0.987** (± 0.022)	**1.376** (± 0.799)	**8.082** (± 1.352)
VI: IAF	1e-3	0.997 (± 0.004)	**1.443** (± 0.564)	**8.517** (± 0.820)	**0.988** (± 0.020)	**1.304** (± 0.855)	**8.425** (± 1.281)
VI: IAF	1e-4	0.997 (± 0.004)	**1.602** (± 0.628)	**7.888** (± 0.783)	**0.987** (± 0.020)	**1.380** (± 0.848)	**7.729** (± 1.322)

I. Detailed Experimental Results

In this section, we describe our experimental results in detail, discussing how different scenarios for GLMs, FA and GMMs affect the performance of different approaches.

I.1. Generalized Linear Models

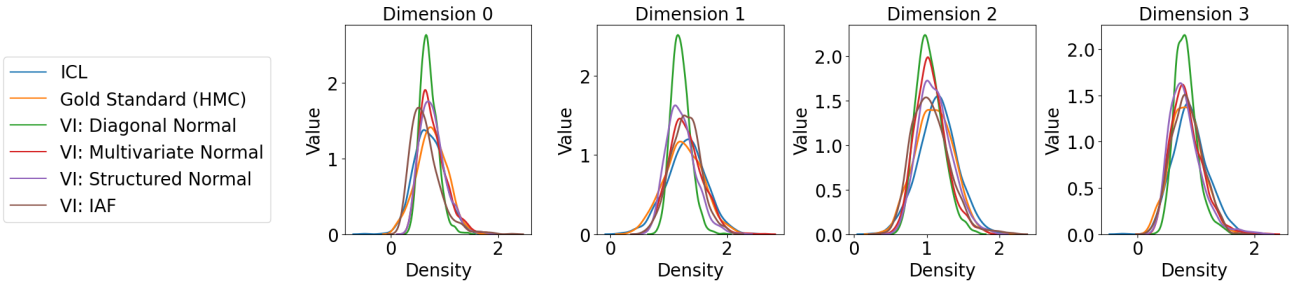


Figure 7: Density plots for first three the marginals of the posterior in a GLM with a gamma prior on the coefficients β, and an inverse gamma prior on the variance σ^2 of the responses. The data is part of the Miami housing 2016 dataset.

Table 45 contains detailed results regarding the performance of the proposed ICL and the reference VI approaches. In summary, we find that on the synthetic data, our ICL method has the overall best performance, or a performance not significantly worse than that of the best model, with respect to the C2ST metric.[4] More specifically, ICL significantly outperforms all other models in 5 out of seven cases w.r.t. the C2ST and also the MMD metric. While the $\mathcal{W}_2$ metric exhibits a larger variance, it also indicates that on the synthetic data, ICL yields the significantly best result in those 5 cases.

On the real-world data, the differences between ICL and VI are less pronounced, and ICL attains the best average result without any other model within two standard errors in three scenarios in terms of the C2ST metric. ICL is among those models not significantly worse than the best in four cases with respect to the C2ST metric, in six cases in terms of the MMD metric, and also in six cases in terms of $\mathcal{W}_2$.

In scenario 1, which is a linear regression scenario with a normal prior on the coefficients $\boldsymbol{\beta}$ and an inverse gamma prior on the variance σ^2, ICL and HMC show a similarly large agreement with the analytical solution. Furthermore, the VI approaches with an ordinary multivariate normal distribution, a structured normal distribution as well as the approach based on inverse autoregressive flows also show a large agreement with the analytical solution, which is to be expected since scenario 1 is has a conjugate prior structure yielding a multivariate t-distribution for the posterior of the coefficients (Murphy, 2023).

Scenario 2 and scenario 4 are those where an intercept is included in the generative structure of the GLM. The notably superior performance of the ICL approach in those two cases might be explained by its ability to model distributions with substantially different variances in different dimensions better than VI. Similarly, the posterior in scenario 5 is determined by the gamma prior on the coefficients leading to a (slightly) skewed posterior distribution, which might explain the good relative performance of ICL. See Figure 7 for a plot of the marginals of the posterior in this scenario on the Miami housing 2016 dataset.

Finally, scenarios 6 and 7 demonstrate the versatility of the ICL method in terms of posterior inference for logistic regression and regression with a gamma response.

[4] We refer to a difference that is larger than two standard deviations as "significant".

Table 12: Generalized Linear Models: Evaluation on 50 synthetic and 17 real-world datasets for seven different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	Laplace Approximation	1.000 (± 0.000)	2.738 (± 0.721)	**0.825** (± 0.279)	1.000 (± 0.000)	2.150 (± 0.323)	**0.642** (± 0.124)
	VI: DiagonalNormal	0.904 (± 0.076)	1.452 (± 0.984)	**0.669** (± 0.301)	0.797 (± 0.083)	0.612 (± 0.511)	**0.414** (± 0.152)
	VI: MultivariateNormal	**0.750** (± 0.128)	**0.735** (± 0.733)	**0.565** (± 0.292)	**0.607** (± 0.070)	**0.167** (± 0.196)	**0.301** (± 0.123)
	VI: Structured Normal	**0.753** (± 0.126)	**0.736** (± 0.737)	**0.570** (± 0.310)	**0.600** (± 0.070)	**0.169** (± 0.214)	**0.306** (± 0.131)
	VI: IAF	**0.777** (± 0.122)	**0.864** (± 0.844)	0.725 (± 0.523)	0.683 (± 0.132)	0.440 (± 0.559)	0.503 (± 0.383)
	HMC	**0.745** (± 0.130)	**0.722** (± 0.732)	**0.569** (± 0.301)	**0.595** (± 0.075)	**0.173** (± 0.213)	**0.321** (± 0.140)
	ICL (ours)	**0.765** (± 0.123)	**0.767** (± 0.727)	**0.585** (± 0.301)	**0.614** (± 0.074)	**0.175** (± 0.219)	**0.310** (± 0.138)
Scenario 2	Laplace Approximation	1.000 (± 0.000)	4.853 (± 2.333)	5.770 (± 5.946)	1.000 (± 0.000)	2.572 (± 0.206)	0.809 (± 0.149)
	VI: DiagonalNormal	0.957 (± 0.091)	3.906 (± 2.679)	5.628 (± 6.092)	0.892 (± 0.044)	0.847 (± 0.389)	**0.530** (± 0.175)
	VI: MultivariateNormal	0.910 (± 0.131)	3.407 (± 2.781)	5.584 (± 6.104)	0.820 (± 0.031)	0.243 (± 0.148)	**0.408** (± 0.118)
	VI: Structured Normal	0.908 (± 0.119)	3.139 (± 2.763)	5.480 (± 6.164)	0.824 (± 0.023)	0.215 (± 0.110)	**0.392** (± 0.109)
	VI: IAF	0.968 (± 0.063)	4.416 (± 2.473)	7.474 (± 6.235)	0.888 (± 0.067)	0.921 (± 0.860)	0.942 (± 0.733)
	ICL (ours)	**0.839** (± 0.072)	**0.707** (± 0.658)	**1.111** (± 0.300)	**0.768** (± 0.033)	**0.143** (± 0.089)	**0.411** (± 0.094)
Scenario 3	Laplace Approximation	1.000 (± 0.000)	2.203 (± 0.997)	1.170 (± 0.949)	1.000 (± 0.000)	1.841 (± 0.185)	0.729 (± 0.175)
	VI: DiagonalNormal	0.866 (± 0.101)	1.069 (± 1.150)	0.846 (± 0.747)	0.797 (± 0.083)	0.526 (± 0.361)	0.480 (± 0.207)
	VI: MultivariateNormal	0.656 (± 0.131)	0.445 (± 1.061)	0.660 (± 0.737)	**0.560** (± 0.035)	**0.032** (± 0.028)	**0.249** (± 0.069)
	VI: Structured Normal	0.653 (± 0.125)	0.421 (± 0.993)	0.659 (± 0.736)	**0.552** (± 0.028)	**0.027** (± 0.015)	**0.239** (± 0.055)
	VI: IAF	0.751 (± 0.148)	0.939 (± 1.349)	0.964 (± 0.924)	0.673 (± 0.141)	0.399 (± 0.543)	0.563 (± 0.433)
	ICL (ours)	**0.611** (± 0.070)	**0.089** (± 0.114)	**0.423** (± 0.348)	0.576 (± 0.027)	**0.037** (± 0.026)	**0.257** (± 0.044)
Scenario 4	Laplace Approximation	1.000 (± 0.000)	3.511 (± 2.025)	2.166 (± 1.722)	1.000 (± 0.000)	2.011 (± 0.058)	0.993 (± 0.144)
	VI: DiagonalNormal	0.968 (± 0.036)	2.798 (± 2.255)	2.065 (± 1.745)	0.916 (± 0.040)	0.928 (± 0.339)	0.732 (± 0.181)
	VI: MultivariateNormal	0.855 (± 0.123)	1.648 (± 2.052)	1.853 (± 1.745)	0.771 (± 0.017)	**0.087** (± 0.030)	**0.539** (± 0.070)
	VI: Structured Normal	0.847 (± 0.116)	1.505 (± 1.978)	1.889 (± 1.883)	0.769 (± 0.012)	**0.083** (± 0.018)	**0.543** (± 0.070)
	VI: IAF	0.942 (± 0.077)	3.029 (± 2.210)	3.554 (± 2.715)	0.833 (± 0.069)	0.636 (± 0.756)	0.978 (± 0.600)
	ICL (ours)	**0.753** (± 0.049)	**0.171** (± 0.153)	**0.631** (± 0.294)	**0.762** (± 0.015)	**0.105** (± 0.046)	**0.597** (± 0.104)
Scenario 5	Laplace Approximation	1.000 (± 0.000)	2.060 (± 0.472)	0.797 (± 0.577)	1.000 (± 0.000)	1.982 (± 0.126)	0.623 (± 0.084)
	VI: DiagonalNormal	0.866 (± 0.085)	0.954 (± 1.022)	0.651 (± 0.549)	0.810 (± 0.036)	0.441 (± 0.252)	0.384 (± 0.089)
	VI: MultivariateNormal	0.765 (± 0.100)	0.537 (± 1.019)	0.633 (± 1.067)	0.711 (± 0.038)	0.148 (± 0.093)	**0.279** (± 0.056)
	VI: Structured Normal	0.758 (± 0.098)	0.447 (± 0.818)	0.572 (± 0.816)	0.705 (± 0.032)	0.140 (± 0.081)	**0.269** (± 0.045)
	VI: IAF	0.814 (± 0.105)	0.953 (± 1.165)	0.881 (± 1.067)	0.777 (± 0.106)	0.684 (± 0.939)	0.625 (± 0.525)
	ICL (ours)	**0.621** (± 0.063)	**0.067** (± 0.080)	**0.299** (± 0.195)	**0.610** (± 0.045)	**0.046** (± 0.020)	**0.242** (± 0.038)
Scenario 6	Laplace Approximation	1.000 (± 0.000)	2.026 (± 0.027)	1.612 (± 0.162)	1.000 (± 0.000)	1.993 (± 0.032)	1.299 (± 0.106)
	VI: DiagonalNormal	0.724 (± 0.060)	0.185 (± 0.082)	**0.787** (± 0.078)	0.703 (± 0.039)	0.147 (± 0.063)	0.637 (± 0.089)
	VI: MultivariateNormal	**0.534** (± 0.018)	**0.014** (± 0.006)	**0.581** (± 0.074)	**0.538** (± 0.019)	**0.016** (± 0.007)	**0.466** (± 0.029)
	VI: Structured Normal	**0.536** (± 0.016)	**0.014** (± 0.005)	**0.583** (± 0.071)	**0.536** (± 0.019)	**0.017** (± 0.009)	**0.469** (± 0.033)
	VI: IAF	0.542 (± 0.026)	0.031 (± 0.031)	0.613 (± 0.092)	**0.535** (± 0.015)	**0.015** (± 0.006)	**0.467** (± 0.031)
	ICL (ours)	**0.532** (± 0.019)	0.016 (± 0.008)	**0.590** (± 0.066)	0.556 (± 0.017)	0.035 (± 0.015)	**0.504** (± 0.038)
Scenario 7	Laplace Approximation	1.000 (± 0.000)	3.559 (± 1.933)	1.347 (± 1.067)	1.000 (± 0.000)	2.016 (± 0.080)	0.763 (± 0.174)
	VI: DiagonalNormal	0.938 (± 0.074)	2.536 (± 2.097)	1.142 (± 0.993)	0.936 (± 0.024)	1.029 (± 0.255)	0.579 (± 0.181)
	VI: MultivariateNormal	0.814 (± 0.181)	1.999 (± 2.283)	1.033 (± 0.969)	**0.741** (± 0.020)	0.093 (± 0.025)	**0.391** (± 0.074)
	VI: Structured Normal	0.824 (± 0.177)	1.891 (± 2.127)	1.041 (± 0.934)	**0.734** (± 0.025)	**0.072** (± 0.019)	**0.385** (± 0.065)
	VI: IAF	0.939 (± 0.091)	2.707 (± 1.712)	1.590 (± 0.820)	0.864 (± 0.093)	0.830 (± 0.697)	1.064 (± 0.616)
	ICL (ours)	**0.700** (± 0.116)	**0.317** (± 0.355)	**0.400** (± 0.286)	0.773 (± 0.048)	**0.294** (± 0.457)	0.559 (± 0.256)

I.2. Factor Analysis

Table 46 contains detailed results regarding FA for 50 synthetic and 17 real-world datasets across 6 different scenarios. We find that overall the ICL method has a very high agreement with the gold standard HMC reference with scores of more than than 56 percent in five scenarios on the synthetic data. In comparison, the C2ST metric is almost saturated for all considered VI methods. For MMD and $\mathcal{W}_2$ the ICL method is again the best.

The real-world datasets show a similar picture except for scenario 4 where C2ST and MMD indicate that VI with inverse autoregressive flows performs best. The $\mathcal{W}_2$ metric, however exhibits a relatively large variance in those cases and does not yield significant results regarding the best performance.

Table 13: Factor Analysis: Evaluation on 50 synthetic and 17 real-world datasets for six different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	Laplace Approximation	1.000 (± 0.000)	3.459 (± 1.553)	1.987 (± 1.363)	1.000 (± 0.000)	2.487 (± 0.454)	**0.875** (± 0.036)
	VI: DiagonalNormal	1.000 (± 0.001)	4.695 (± 1.488)	2.865 (± 1.681)	0.979 (± 0.008)	1.283 (± 0.225)	**0.625** (± 0.058)
	VI: MultivariateNormal	0.998 (± 0.003)	4.163 (± 1.473)	2.603 (± 1.959)	0.966 (± 0.010)	1.213 (± 0.260)	**0.608** (± 0.047)
	VI: Structured Normal	0.997 (± 0.004)	4.655 (± 1.189)	2.700 (± 1.333)	0.979 (± 0.010)	1.231 (± 0.132)	**0.611** (± 0.041)
	VI: IAF	0.953 (± 0.104)	3.992 (± 2.089)	2.750 (± 1.838)	0.849 (± 0.075)	0.772 (± 0.335)	**0.503** (± 0.063)
	ICL (ours)	**0.552** (± 0.028)	**0.034** (± 0.034)	**0.289** (± 0.083)	**0.606** (± 0.038)	**0.068** (± 0.069)	0.265 (± 0.078)
Scenario 2	Laplace Approximation	1.000 (± 0.000)	3.687 (± 1.661)	1.954 (± 1.129)	1.000 (± 0.000)	1.690 (± 0.182)	**0.598** (± 0.058)
	VI: DiagonalNormal	0.998 (± 0.002)	3.135 (± 1.482)	1.629 (± 0.938)	0.975 (± 0.010)	1.156 (± 0.068)	**0.496** (± 0.052)
	VI: MultivariateNormal	0.989 (± 0.009)	2.945 (± 1.019)	1.482 (± 0.683)	0.951 (± 0.025)	0.764 (± 0.053)	**0.421** (± 0.052)
	VI: Structured Normal	0.984 (± 0.031)	3.790 (± 1.572)	2.106 (± 1.429)	0.958 (± 0.025)	1.001 (± 0.126)	**0.465** (± 0.056)
	VI: IAF	0.966 (± 0.066)	3.523 (± 1.340)	2.153 (± 0.968)	0.799 (± 0.058)	0.462 (± 0.226)	**0.342** (± 0.070)
	ICL (ours)	**0.542** (± 0.006)	**0.017** (± 0.006)	**0.244** (± 0.033)	**0.622** (± 0.032)	**0.098** (± 0.039)	**0.287** (± 0.046)
Scenario 3	Laplace Approximation	1.000 (± 0.000)	4.137 (± 0.932)	2.188 (± 1.011)	1.000 (± 0.000)	3.653 (± 0.183)	**0.473** (± 0.026)
	VI: DiagonalNormal	0.999 (± 0.002)	3.339 (± 0.985)	1.722 (± 0.870)	0.951 (± 0.007)	1.114 (± 0.080)	**0.245** (± 0.016)
	VI: MultivariateNormal	0.994 (± 0.007)	3.189 (± 0.960)	1.644 (± 0.859)	0.945 (± 0.007)	1.085 (± 0.082)	**0.242** (± 0.015)
	VI: Structured Normal	0.997 (± 0.003)	3.159 (± 0.968)	1.614 (± 0.793)	0.942 (± 0.009)	1.084 (± 0.071)	**0.242** (± 0.018)
	VI: IAF	0.990 (± 0.011)	3.145 (± 1.203)	1.705 (± 0.990)	0.928 (± 0.015)	1.022 (± 0.093)	**0.235** (± 0.018)
	ICL (ours)	**0.537** (± 0.023)	**0.024** (± 0.021)	**0.259** (± 0.088)	**0.609** (± 0.019)	**0.124** (± 0.037)	**0.179** (± 0.018)
Scenario 4	Laplace Approximation	1.000 (± 0.000)	4.354 (± 0.572)	3.339 (± 0.932)	1.000 (± 0.000)	6.617 (± 0.259)	0.598 (± 0.135)
	VI: DiagonalNormal	1.000 (± 0.000)	3.396 (± 0.591)	2.420 (± 0.720)	0.977 (± 0.003)	1.499 (± 0.066)	**0.096** (± 0.003)
	VI: MultivariateNormal	0.999 (± 0.001)	3.447 (± 0.567)	2.479 (± 0.848)	0.973 (± 0.008)	1.484 (± 0.097)	0.096 (± 0.005)
	VI: Structured Normal	1.000 (± 0.000)	3.421 (± 0.610)	2.481 (± 0.884)	0.973 (± 0.007)	1.474 (± 0.078)	**0.095** (± 0.004)
	VI: IAF	0.999 (± 0.001)	3.269 (± 0.552)	2.307 (± 0.779)	**0.961** (± 0.018)	**1.337** (± 0.142)	**0.092** (± 0.005)
	ICL (ours)	**0.684** (± 0.060)	**0.198** (± 0.141)	**0.918** (± 0.246)	0.988 (± 0.003)	1.764 (± 0.026)	1.248 (± 0.008)
Scenario 5	Laplace Approximation	1.000 (± 0.000)	4.456 (± 0.785)	2.608 (± 0.946)	1.000 (± 0.000)	4.559 (± 0.494)	0.663 (± 0.127)
	VI: DiagonalNormal	0.999 (± 0.002)	3.520 (± 1.073)	2.012 (± 0.886)	0.944 (± 0.010)	1.007 (± 0.129)	**0.261** (± 0.036)
	VI: MultivariateNormal	0.995 (± 0.007)	3.472 (± 1.021)	1.982 (± 0.814)	0.930 (± 0.017)	0.964 (± 0.111)	**0.255** (± 0.038)
	VI: Structured Normal	0.998 (± 0.005)	3.369 (± 1.044)	1.916 (± 0.852)	0.934 (± 0.011)	0.996 (± 0.133)	**0.259** (± 0.035)
	VI: IAF	0.992 (± 0.012)	3.166 (± 0.967)	1.761 (± 0.671)	0.910 (± 0.011)	**0.892** (± 0.094)	**0.247** (± 0.037)
	ICL (ours)	**0.535** (± 0.016)	**0.021** (± 0.011)	**0.279** (± 0.060)	**0.886** (± 0.017)	1.207 (± 0.101)	**1.002** (± 0.042)
Scenario 6	Laplace Approximation	1.000 (± 0.000)	3.942 (± 0.971)	2.624 (± 1.682)	1.000 (± 0.000)	3.319 (± 0.196)	**0.377** (± 0.020)
	VI: DiagonalNormal	0.998 (± 0.002)	3.214 (± 1.072)	2.209 (± 1.543)	0.949 (± 0.008)	1.196 (± 0.093)	**0.210** (± 0.011)
	VI: MultivariateNormal	0.991 (± 0.013)	3.056 (± 1.237)	2.189 (± 1.698)	0.938 (± 0.009)	1.121 (± 0.075)	**0.205** (± 0.012)
	VI: Structured Normal	0.997 (± 0.005)	3.279 (± 1.071)	2.276 (± 1.787)	0.944 (± 0.006)	1.161 (± 0.066)	**0.208** (± 0.012)
	VI: IAF	0.989 (± 0.029)	3.027 (± 0.910)	1.936 (± 1.060)	0.865 (± 0.027)	0.822 (± 0.106)	**0.179** (± 0.015)
	ICL (ours)	**0.543** (± 0.021)	**0.023** (± 0.015)	**0.345** (± 0.173)	**0.666** (± 0.020)	**0.200** (± 0.034)	**0.224** (± 0.014)

I.3. Gaussian Mixture Models

We summarize the results of the ICL approach and the different VI methods regarding the GMM scenarios in Table 47. First, one can note that on the synthetic data, the ICL approach has a much lower C2ST score for scenario 1 and scenario 2 than the other methods. However, for scenarios 3 and 4, C2ST saturates, or at least almost saturates for all approaches. The MMD metric, however, shows that ICL not only has a high agreement with HMC in scenarios 1 and 2, but that it attains the significantly best result in scenarios 3 and 4 as well. This is supported by the $\mathcal{W}_2$ metric, which has the significantly lowest values for ICL in scenarios 2,3 and 4.

Analogously, on the real-world data, MMD shows that ICL is the best approach in all four scenarios without any other model coming into the two standard-deviation range. While the C2ST score is the lowest in scenario 1 and scenario 2 for ICL, it saturates for cases 3 and 4.

Table 14: Gaussian Mixture Models: Evaluation on 50 synthetic and 17 real-world datasets for six different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	Laplace Approximation	1.000 (± 0.000)	3.367 (± 1.030)	**4.341** (± 2.018)	1.000 (± 0.000)	3.374 (± 0.941)	**6.440** (± 1.994)
	VI: DiagonalNormal	0.988 (± 0.013)	1.175 (± 1.189)	**2.961** (± 1.669)	0.995 (± 0.006)	1.919 (± 1.217)	**5.145** (± 2.489)
	VI: MultivariateNormal	0.988 (± 0.013)	1.135 (± 1.149)	**2.926** (± 1.651)	0.994 (± 0.007)	2.007 (± 1.367)	**5.379** (± 2.845)
	VI: Structured Normal	0.987 (± 0.015)	1.126 (± 1.145)	**2.944** (± 1.663)	0.993 (± 0.009)	1.943 (± 1.359)	**5.313** (± 2.737)
	VI: IAF	0.989 (± 0.013)	1.017 (± 1.036)	**3.104** (± 1.523)	0.995 (± 0.010)	1.888 (± 1.051)	**5.402** (± 2.310)
	ICL (ours)	**0.760 (± 0.092)**	**0.303** (± 0.548)	**2.095** (± 1.692)	**0.847** (± 0.082)	**0.486** (± 0.623)	**4.054** (± 2.782)
Scenario 2	Laplace Approximation	1.000 (± 0.000)	2.864 (± 0.607)	5.407 (± 2.320)	1.000 (± 0.000)	2.928 (± 0.438)	**7.228** (± 1.323)
	VI: DiagonalNormal	0.989 (± 0.024)	1.425 (± 0.829)	4.933 (± 2.379)	0.998 (± 0.003)	1.525 (± 0.356)	**6.091** (± 0.931)
	VI: MultivariateNormal	0.991 (± 0.021)	1.532 (± 0.940)	5.119 (± 2.521)	0.999 (± 0.002)	1.619 (± 0.269)	**6.258** (± 0.872)
	VI: Structured Normal	0.992 (± 0.017)	1.487 (± 0.899)	5.085 (± 2.530)	0.999 (± 0.002)	1.580 (± 0.337)	**6.241** (± 0.960)
	VI: IAF	0.992 (± 0.021)	1.319 (± 0.854)	5.265 (± 2.534)	0.998 (± 0.004)	1.256 (± 0.320)	**6.201** (± 0.892)
	ICL (ours)	**0.812 (± 0.061)**	**0.159** (± 0.154)	**2.314** (± 0.926)	**0.937** (± 0.041)	**0.282** (± 0.131)	**3.947** (± 1.055)
Scenario 3	Laplace Approximation	1.000 (± 0.000)	3.631 (± 1.362)	16.387 (± 19.604)	1.000 (± 0.000)	3.009 (± 0.768)	**37.034** (± 7.178)
	VI: DiagonalNormal	**0.996** (± 0.011)	2.127 (± 1.479)	16.864 (± 19.301)	**0.992** (± 0.018)	2.429 (± 0.516)	**35.355** (± 6.608)
	VI: MultivariateNormal	0.997 (± 0.009)	2.076 (± 1.388)	16.938 (± 19.636)	**0.993** (± 0.016)	2.427 (± 0.510)	**35.312** (± 6.655)
	VI: Structured Normal	**0.995** (± 0.017)	2.049 (± 1.462)	16.723 (± 19.093)	**0.993** (± 0.016)	2.301 (± 0.549)	**34.217** (± 5.461)
	VI: IAF	**0.994** (± 0.018)	1.675 (± 1.049)	14.311 (± 9.266)	**0.993** (± 0.017)	2.148 (± 0.528)	**34.336** (± 5.398)
	ICL (ours)	1.000 (± 0.000)	**0.582** (± 0.280)	**8.708** (± 4.945)	1.000 (± 0.000)	**1.869** (± 0.342)	**33.230** (± 8.095)
Scenario 4	Laplace Approximation	1.000 (± 0.000)	6.260 (± 1.427)	13.497 (± 29.702)	1.000 (± 0.000)	5.924 (± 1.145)	**12.400** (± 4.313)
	VI: DiagonalNormal	**1.000** (± 0.002)	3.958 (± 1.641)	12.068 (± 21.301)	1.000 (± 0.000)	3.879 (± 1.061)	**11.080** (± 3.341)
	VI: MultivariateNormal	**1.000** (± 0.002)	3.875 (± 1.691)	12.150 (± 22.198)	1.000 (± 0.000)	3.896 (± 1.057)	**11.112** (± 3.321)
	VI: Structured Normal	**1.000** (± 0.001)	3.661 (± 1.717)	12.195 (± 22.874)	**0.996** (± 0.016)	3.822 (± 1.302)	**11.368** (± 4.216)
	VI: IAF	**1.000** (± 0.002)	3.536 (± 1.597)	12.015 (± 20.884)	1.000 (± 0.000)	3.471 (± 1.036)	**11.421** (± 3.233)
	ICL (ours)	1.000 (± 0.000)	**2.451** (± 0.868)	**8.333** (± 4.202)	**1.000** (± 0.000)	**2.518 (± 0.694)**	**11.938** (± 2.956)

J. Evaluating Predictive Performance

In this section, we discuss the results of our ICL approach in terms of predictive performance. In scenarios one to four, TabPFN gives the best overall performance, which is expected since it is not limited to the GLM structure. Besides that, the MAP approach obtains consistently the best results, while our ICL method performs on par (scenarios 1,2,3) or better than (scenario 4) compared to the fully Bayesian methods on the real-world data. On the synthetic data there is no significant difference to the other fully Bayesian methods, except for the real-world data in scenario where HMC is clearly the best method. In scenario 5 (gamma prior on the regression coefficients), the in-context learner performs significantly worse than all other methods, while this difference is less pronounced in scenario 7. In scenario 6, TabPFN also has a substantially better performance than all other methods. The MAP approach performs on average better than all fully Bayesian methods, which themselves do not differ significantly.

Table 15: Evaluating the predictive performance across 50 synthetic and 17 real-world datasets for scenarios 1-4 in terms of Root Mean Squared Error (RMSE). The best result among all fully Bayesian methods is marked in **bold**. For the fully Bayesian approaches, we use the posterior mean as a point estimate for the response. MAP denotes the predictive performance of the model with the maximum a posteriori estimate for the latents.

Scenario	Model	RMSE Real-World (↓)	RMSE Synthetic (↓)
Scenario 1	HMC	**0.591** (± 0.023)	**0.510** (± 0.040)
	Laplace Approximation	**0.594** (± 0.023)	**0.510** (± 0.040)
	VI: DiagonalNormal	**0.591** (± 0.023)	**0.509** (± 0.040)
	VI: MultivariateNormal	**0.591** (± 0.023)	**0.510** (± 0.040)
	VI: Structured Normal	**0.629** (± 0.017)	0.555 (± 0.039)
	VI: IAF	**0.593** (± 0.023)	**0.510** (± 0.040)
	ICL (ours)	**0.593** (± 0.020)	**0.524** (± 0.038)
	MAP	0.555 (± 0.024)	0.491 (± 0.038)
	TabPFN	0.483 (± 0.036)	0.453 (± 0.036)
Scenario 2	HMC	**0.559** (± 0.023)	**0.556** (± 0.049)
	Laplace Approximation	**0.561** (± 0.022)	**0.557** (± 0.049)
	VI: DiagonalNormal	**0.560** (± 0.023)	**0.557** (± 0.049)
	VI: MultivariateNormal	**0.559** (± 0.023)	**0.556** (± 0.049)
	VI: Structured Normal	**0.604** (± 0.016)	0.685 (± 0.054)
	VI: IAF	**0.563** (± 0.023)	**0.557** (± 0.049)
	ICL (ours)	**0.561** (± 0.019)	**0.653** (± 0.049)
	MAP	0.513 (± 0.023)	0.522 (± 0.048)
	TabPFN	0.449 (± 0.034)	0.498 (± 0.047)
Scenario 3	HMC	**0.684** (± 0.027)	**0.512** (± 0.040)
	Laplace Approximation	**0.688** (± 0.026)	0.516 (± 0.040)
	VI: DiagonalNormal	**0.686** (± 0.027)	**0.513** (± 0.040)
	VI: MultivariateNormal	**0.685** (± 0.027)	**0.512** (± 0.040)
	VI: Structured Normal	**0.733** (± 0.016)	0.607 (± 0.043)
	VI: IAF	**0.686** (± 0.027)	**0.512** (± 0.040)
	ICL (ours)	**0.690** (± 0.023)	**0.588** (± 0.045)
	MAP	0.646 (± 0.028)	0.495 (± 0.039)
	TabPFN	0.556 (± 0.041)	0.462 (± 0.037)
Scenario 4	HMC	**0.642** (± 0.027)	**0.559** (± 0.051)
	Laplace Approximation	0.737 (± 0.048)	2.457 (± 0.493)
	VI: DiagonalNormal	0.751 (± 0.038)	2.046 (± 0.399)
	VI: MultivariateNormal	**0.690** (± 0.037)	2.155 (± 0.454)
	VI: Structured Normal	**0.686** (± 0.015)	3.019 (± 0.545)
	VI: IAF	**0.643** (± 0.027)	1.751 (± 0.422)
	ICL (ours)	**0.649** (± 0.023)	1.464 (± 0.151)
	MAP	0.626 (± 0.038)	2.377 (± 0.529)
	TabPFN	0.522 (± 0.037)	0.496 (± 0.047)

Table 16: Evaluating the predictive performance across 50 synthetic and 17 real-world datasets for scenarios 5 and 7 in terms of Root Mean Squared Error (RMSE). The best result among all fully Bayesian methods is marked in **bold**. For the fully Bayesian approaches, we use the posterior mean as a point estimate for the response. MAP denotes the predictive performance of the model with the maximum a posteriori estimate for the latents.

Scenario	Model	RMSE Real-World (↓)	RMSE Synthetic (↓)
Scenario 5	HMC	**0.699** (± 0.022)	**0.490** (± 0.036)
	Laplace Approximation	**0.699** (± 0.022)	**0.491** (± 0.036)
	VI: DiagonalNormal	**0.702** (± 0.022)	**0.491** (± 0.036)
	VI: MultivariateNormal	**0.698** (± 0.021)	**0.491** (± 0.036)
	VI: Structured Normal	1.507 (± 0.089)	0.741 (± 0.053)
	VI: IAF	**0.699** (± 0.022)	**0.490** (± 0.036)
	ICL (ours)	0.769 (± 0.020)	0.701 (± 0.049)
	MAP	0.658 (± 0.022)	0.471 (± 0.035)
	TabPFN	0.534 (± 0.040)	0.442 (± 0.035)
Scenario 7	HMC	**0.953** (± 0.015)	**0.719** (± 0.041)
	Laplace Approximation	**0.950** (± 0.016)	**0.719** (± 0.041)
	VI: DiagonalNormal	**0.954** (± 0.015)	**0.718** (± 0.041)
	VI: MultivariateNormal	**0.953** (± 0.015)	**0.718** (± 0.041)
	VI: Structured Normal	1.082 (± 0.026)	1.028 (± 0.118)
	VI: IAF	**0.954** (± 0.014)	**0.720** (± 0.041)
	ICL (ours)	1.019 (± 0.017)	**0.765** (± 0.041)
	MAP	0.945 (± 0.017)	0.686 (± 0.048)
	TabPFN	0.817 (± 0.040)	0.654 (± 0.039)

Table 17: Evaluating the predictive performance across 50 synthetic and 17 real-world datasets for scenarios 5 and 7 in terms of accuracy (Acc.). The best result among all fully Bayesian methods is marked in **bold**. For the fully Bayesian approaches, we use the posterior mean as a point estimate for the response. MAP denotes the predictive performance of the model with the maximum a posteriori estimate for the latents.

Scenario	Model	Acc. Real-World (↑)	Acc. Synthetic (↑)
Scenario 6	HMC	**0.694** (± 0.028)	**0.546** (± 0.015)
	Laplace Approximation	**0.692** (± 0.027)	**0.547** (± 0.015)
	VI: DiagonalNormal	**0.700** (± 0.028)	**0.546** (± 0.015)
	VI: MultivariateNormal	**0.691** (± 0.029)	**0.546** (± 0.015)
	VI: Structured Normal	**0.686** (± 0.028)	**0.546** (± 0.015)
	VI: IAF	**0.689** (± 0.029)	**0.545** (± 0.015)
	ICL (ours)	**0.688** (± 0.027)	**0.545** (± 0.015)
	MAP	0.723 (± 0.025)	0.610 (± 0.016)
	TabPFN	0.862 (± 0.021)	0.673 (± 0.011)

K. Ablation: Using a Gaussian Approximation

In this section, we present results on using a Gaussian approximation instead of Flow Matching to parameterize the approximation of the posterior.

The key takeaway from these results is that the in-context learning approach performs substantially better with Flow Matching (Lipman et al., 2022) than when a Gaussian approximation of the posterior is employed.

Table 18: Generalized Linear Models: Comparing the in-context learner with a Gaussian approximation, fitted via the forward KL-divergence, to the proposed flow matching method. Evaluation on 50 synthetic and 17 real-world datasets for seven different scenarios. If one method is by more than two standard errors better than the other, it is marked in **bold**. Overall, the ICL + Flow Matching method clearly outperforms the Gaussian approximation, fitted via the forward KL-divergence,: it yields significantly better results (according to the two-standard-error criterion) in 6 out of 7 scenarios on synthetic datasets and in all 7 scenarios on real-world datasets, across at least two of the three considered metrics (C2ST, MMD, or $\mathcal{W}_2$). In addition, the flow matching method consistently achieves lower or comparable standard errors, indicating more stable and reliable performance across datasets.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	ICL + Gaussian	**0.845** (± 0.213)	**1.601** (± 1.213)	2.024 (± 0.874)	0.980 (± 0.007)	1.715 (± 0.295)	1.976 (± 0.238)
	ICL + Flow Matching	**0.765** (± 0.123)	**0.767** (± 0.727)	**0.585** (± 0.301)	**0.614** (± 0.074)	**0.175** (± 0.219)	**0.310** (± 0.138)
Scenario 2	ICL + Gaussian	**0.941** (± 0.056)	**1.000** (± 0.953)	1.943 (± 0.657)	0.969 (± 0.013)	1.490 (± 0.310)	2.068 (± 0.259)
	ICL + Flow Matching	**0.839** (± 0.072)	**0.707** (± 0.658)	**1.111** (± 0.300)	**0.768** (± 0.033)	**0.143** (± 0.089)	**0.411** (± 0.094)
Scenario 3	ICL + Gaussian	0.907 (± 0.138)	1.779 (± 1.363)	4.713 (± 1.560)	0.985 (± 0.006)	1.526 (± 0.198)	4.144 (± 0.438)
	ICL + Flow Matching	**0.611** (± 0.070)	**0.089** (± 0.114)	**0.423** (± 0.348)	**0.576** (± 0.027)	**0.037** (± 0.026)	**0.257** (± 0.044)
Scenario 4	ICL + Gaussian	0.989 (± 0.011)	3.544 (± 0.343)	23.035 (± 6.549)	0.990 (± 0.003)	3.858 (± 0.061)	13.601 (± 0.427)
	ICL + Flow Matching	**0.753** (± 0.049)	**0.171** (± 0.153)	**0.631** (± 0.294)	**0.762** (± 0.015)	**0.105** (± 0.046)	**0.597** (± 0.104)
Scenario 5	ICL + Gaussian	0.962 (± 0.037)	1.444 (± 1.640)	3.299 (± 1.614)	0.991 (± 0.005)	1.666 (± 0.387)	2.963 (± 0.239)
	ICL + Flow Matching	**0.621** (± 0.063)	**0.067** (± 0.080)	**0.299** (± 0.195)	**0.610** (± 0.045)	**0.046** (± 0.020)	**0.242** (± 0.038)
Scenario 6	ICL + Gaussian	0.909 (± 0.048)	1.020 (± 0.505)	1.515 (± 0.358)	0.939 (± 0.047)	1.799 (± 0.751)	1.904 (± 0.541)
	ICL + Flow Matching	**0.532** (± 0.019)	**0.016** (± 0.008)	**0.590** (± 0.066)	**0.556** (± 0.017)	**0.035** (± 0.015)	**0.504** (± 0.038)
Scenario 7	ICL + Gaussian	0.970 (± 0.030)	2.169 (± 1.473)	1.707 (± 0.480)	0.993 (± 0.006)	2.390 (± 0.414)	1.362 (± 0.152)
	ICL + Flow Matching	**0.700** (± 0.116)	**0.317** (± 0.355)	**0.400** (± 0.286)	**0.773** (± 0.048)	**0.294** (± 0.457)	**0.559** (± 0.256)

Table 19: Factor Analysis: Comparing the in-context learner with a Gaussian approximation, fitted via the forward KL-divergence, to the proposed flow matching method. Evaluation on 50 synthetic and 17 real-world datasets for seven different scenarios. If one method is by more than two standard errors better than the other, it is marked in **bold**. The flow matching approach shows favorable performance in the majority of cases. Specifically, it achieves statistically significant improvements in all 6 scenarios on synthetic data and in 5 out of 6 scenarios on real-world data. Notably, it often reduces discrepancy measures such as MMD and Wasserstein-2 distance by a large margin. In addition, the variability of the flow matching estimates is generally lower, leading to more reliable and consistent results across different datasets. In scenario 4, the Gaussian in-context learner learned a singular covariance matrix.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	ICL + Gaussian	0.974 (± 0.028)	1.838 (± 0.778)	1.450 (± 0.607)	**0.589** (± 0.015)	**0.080** (± 0.010)	0.459 (± 0.017)
	ICL + Flow Matching	**0.552** (± 0.028)	**0.034** (± 0.034)	**0.289** (± 0.083)	**0.606** (± 0.038)	**0.068** (± 0.069)	**0.265** (± 0.078)
Scenario 2	ICL + Gaussian	0.835 (± 0.040)	0.813 (± 0.276)	1.250 (± 0.316)	0.889 (± 0.027)	0.778 (± 0.109)	1.074 (± 0.073)
	ICL + Flow Matching	**0.542** (± 0.006)	**0.017** (± 0.006)	**0.244** (± 0.033)	**0.622** (± 0.032)	**0.098** (± 0.039)	**0.287** (± 0.046)
Scenario 3	ICL + Gaussian	0.826 (± 0.035)	0.826 (± 0.226)	1.210 (± 0.239)	0.942 (± 0.008)	1.466 (± 0.078)	1.317 (± 0.038)
	ICL + Flow Matching	**0.537** (± 0.023)	**0.024** (± 0.021)	**0.259** (± 0.088)	**0.609** (± 0.019)	**0.124** (± 0.037)	**0.179** (± 0.018)
Scenario 4	ICL + Gaussian	0.870 (± 0.043)	0.706 (± 0.218)	1.635 (± 0.297)	0.999 (± 0.001)	2.025 (± 0.017)	2.013 (± 0.019)
	ICL + Flow Matching	**0.684** (± 0.060)	**0.198** (± 0.141)	**0.918** (± 0.246)	**0.988** (± 0.003)	**1.764** (± 0.026)	**1.248** (± 0.008)
Scenario 5	ICL + Gaussian	0.838 (± 0.029)	0.831 (± 0.219)	1.248 (± 0.249)	0.944 (± 0.009)	1.477 (± 0.073)	1.316 (± 0.031)
	ICL + Flow Matching	**0.535** (± 0.016)	**0.021** (± 0.011)	**0.279** (± 0.060)	**0.886** (± 0.017)	**1.207** (± 0.101)	**1.002** (± 0.042)
Scenario 6	ICL + Gaussian	0.837 (± 0.030)	0.831 (± 0.219)	1.248 (± 0.249)	0.944 (± 0.008)	1.477 (± 0.073)	1.316 (± 0.031)
	ICL + Flow Matching	**0.543** (± 0.021)	**0.023** (± 0.015)	**0.345** (± 0.173)	**0.666** (± 0.020)	**0.200** (± 0.034)	**0.224** (± 0.014)

Table 20: Gaussian Mixture Models: Comparing the in-context learner with a Gaussian approximation, fitted via the forward KL-divergence, to the proposed flow matching method. Evaluation on 50 synthetic and 17 real-world datasets for seven different scenarios. If one method is by more than two standard errors better than the other, it is marked in **bold**. While the differences are less clear-cut than in the previous models, ICL + Flow Matching demonstrates favorable performance in several scenarios, particularly for the Wasserstein-2 distance and MMD. Notably, its advantage is most visible in lower-dimensional settings (Scenario 1 and 2), where it consistently improves upon the Gaussian approximation, fitted via the forward KL-divergence, across most metrics. However, as the dimensionality increases, the performance gap tends to narrow, and in some cases, the inherent variability of the datasets, especially for the Gaussian approximation, fitted via the forward KL-divergence,, makes it difficult to conclusively determine a clear winner. Nonetheless, the flow matching approach often achieves smaller standard errors and lower discrepancy measures, underlining its potential for more stable modeling.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	ICL + Gaussian	**0.926** (± 0.029)	**0.555** (± 0.452)	**2.586** (± 0.560)	**0.957** (± 0.034)	**0.765** (± 0.958)	**3.717** (± 1.709)
	ICL + Flow Matching	**0.760** (± 0.092)	**0.303** (± 0.548)	**2.095** (± 1.692)	**0.847** (± 0.082)	**0.486** (± 0.623)	**4.054** (± 2.782)
Scenario 2	ICL + Gaussian	0.985 (± 0.010)	0.761 (± 0.227)	5.022 (± 0.945)	**0.999** (± 0.001)	0.801 (± 0.256)	7.525 (± 1.513)
	ICL + Flow Matching	**0.812** (± 0.061)	**0.159** (± 0.154)	**2.314** (± 0.926)	**0.937** (± 0.041)	**0.282** (± 0.131)	**3.947** (± 1.055)
Scenario 3	ICL + Gaussian	**0.998** (± 0.002)	**0.829** (± 0.241)	**11.536** (± 2.365)	**1.000** (± 0.000)	**1.500** (± 0.251)	**26.242** (± 4.171)
	ICL + Flow Matching	**1.000** (± 0.000)	**0.582** (± 0.280)	**8.708** (± 4.945)	**1.000** (± 0.000)	**1.869** (± 0.342)	**33.230** (± 8.095)
Scenario 4	ICL + Gaussian	**0.998** (± 0.001)	**6.314** (± 0.449)	**13.404** (± 0.609)	**0.997** (± 0.001)	**2.770** (± 1.201)	22.596 (± 5.717)
	ICL + Flow Matching	1.000 (± 0.000)	**2.451** (± 0.868)	**8.333** (± 4.202)	1.000 (± 0.000)	**2.518** (± 0.694)	**11.938** (± 2.956)

L. Ablation: Using a Diffusion Objective

To validate choosing the flow matching objective with optimal transport (OT) paths resulting in the objective in equation Equation (7), we also conduct experiments using a diffusion-objective with variance preserving paths introduced by Song et al. (2020). We choose three selected GLM, FA and GMM scenarios with the same 50 synthetic and 17 real-world datasets for each scenario as in the other benchmarks.

L.1. Diffusion with Flow-Matching

First, we use the diffusion objective learned via flow matching, as described in (Lipman et al., 2022), where we choose the same hyperparameters as (Lipman et al., 2022).

Table 21: GLMs: Comparison of the OT flow matching and the VP diffusion objective on 50 synthetic and 17 real-world datasets for three different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 2	Diffusion paths + FM	0.961 (± 0.040)	**1.525** (± 0.777)	3.354 (± 1.333)	0.961 (± 0.016)	1.347 (± 0.365)	2.025 (± 0.270)
	OT paths	**0.839** (± 0.072)	**0.707** (± 0.658)	**1.111** (± 0.300)	**0.768** (± 0.033)	**0.143** (± 0.089)	**0.411** (± 0.094)
Scenario 3	Diffusion paths + FM	0.903 (± 0.111)	1.080 (± 0.564)	1.733 (± 0.408)	0.936 (± 0.013)	1.002 (± 0.203)	1.442 (± 0.103)
	OT paths	**0.611** (± 0.070)	**0.089** (± 0.114)	**0.423** (± 0.348)	**0.576** (± 0.027)	**0.037** (± 0.026)	**0.257** (± 0.044)
Scenario 5	Diffusion paths + FM	**0.691** (± 0.074)	0.211 (± 0.143)	0.708 (± 0.233)	**0.681** (± 0.038)	0.182 (± 0.093)	**0.554 (± 0.090)**
	OT paths + FM	**0.621** (± 0.063)	**0.067** (± 0.080)	**0.299** (± 0.195)	**0.610** (± 0.045)	**0.046** (± 0.020)	**0.242** (± 0.038)

In summary, the empirical results demonstrate that using the OT paths consistently outperforms the VP diffusion objective across all scenarios for both GLMs and FAs. For GLMs, OT paths achieve significantly lower C2ST values in all scenarios. In Scenario 2, OT paths reduce C2ST from 0.961 to 0.839 on synthetic data and from 0.961 to 0.768 on real-world data. Similarly, in Scenario 3, OT paths achieve substantial improvements, with C2ST dropping from 0.903 to 0.611 on synthetic data and from 0.936 to 0.576 on real-world data. This trend is complemented by consistent improvements in other metrics such as $\mathcal{W}_2$, where OT paths often achieve reductions by over 50%.

Table 22: FA: Comparison of the OT flow matching and the VP diffusion objective on 50 synthetic and 17 real-world datasets for three different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	Diffusion paths + FM	0.622 (± 0.043)	0.207 (± 0.121)	0.692 (± 0.192)	**0.595** (± 0.012)	0.089 (± 0.011)	0.475 (± 0.019)
	OT paths + FM	**0.552** (± 0.028)	**0.034** (± 0.034)	**0.289** (± 0.083)	**0.606** (± 0.038)	**0.068** (± 0.069)	**0.265** (± 0.078)
Scenario 2	Diffusion paths + FM	0.826 (± 0.036)	0.768 (± 0.238)	1.219 (± 0.276)	0.878 (± 0.028)	0.793 (± 0.154)	1.056 (± 0.084)
	OT paths + FM	**0.542** (± 0.006)	**0.017** (± 0.006)	**0.244** (± 0.033)	**0.622** (± 0.032)	**0.098** (± 0.039)	**0.287** (± 0.046)
Scenario 3	Diffusion paths + FM	0.751 (± 0.048)	0.387 (± 0.216)	0.834 (± 0.163)	0.944 (± 0.008)	1.514 (± 0.056)	1.332 (± 0.028)
	OT paths + FM	**0.537** (± 0.023)	**0.024** (± 0.021)	**0.259** (± 0.088)	**0.609** (± 0.019)	**0.124** (± 0.037)	**0.179** (± 0.018)

For FA, the performance gap in C2ST remains notable. In Scenario 1, OT paths achieve the best results on synthetic data, reducing C2ST from 0.622 to 0.552, while also delivering improvements in $\mathcal{W}_2$ (0.289 compared to 0.692). On real-world datasets, OT paths maintain competitive results, matching or exceeding the performance of diffusion paths. The advantage is even more pronounced in Scenario 2, where OT paths consistently lead across all metrics, with a particularly striking reduction in MMD on synthetic data (0.017 compared to 0.768) and strong results for C2ST on real-world data (0.622 vs. 0.878). Similarly, in Scenario 3, OT paths achieve the lowest C2ST values, with synthetic results improving from 0.751 to 0.537 and real-world results from 0.944 to 0.609.

In the case of Gaussian Mixture Models (GMMs), the empirical results indicate that the OT paths generally outperform the VP diffusion objective across most scenarios and metrics, though the differences are not always statistically significant in pair-wise comparisons. For example, in Scenario 1, OT paths achieve notably better results for C2ST on both synthetic and real-world datasets, with reductions from 0.924 to 0.760 and from 0.958 to 0.847, respectively. Similarly, for $\mathcal{W}_2$, OT paths

Table 23: GMMs: Comparison of the OT flow matching and the VP diffusion objective on 50 synthetic and 17 real-world datasets for three different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	Diffusion paths + FM	0.924 (± 0.024)	**0.241** (± 0.381)	**2.195** (± 1.431)	0.958 (± 0.030)	0.890 (± 0.912)	**5.328** (± 2.544)
	OT paths + FM	**0.760** (± 0.092)	**0.303** (± 0.548)	**2.095** (± 1.692)	**0.847** (± 0.082)	**0.486** (± 0.623)	**4.054** (± 2.782)
Scenario 2	Diffusion paths + FM	0.942 (± 0.020)	**0.213** (± 0.187)	**2.748** (± 0.659)	**0.984** (± 0.012)	**0.411** (± 0.162)	**5.397** (± 1.458)
	OT paths + FM	**0.812** (± 0.061)	**0.159** (± 0.154)	**2.314** (± 0.926)	**0.937** (± 0.041)	**0.282** (± 0.131)	**3.947** (± 1.055)
Scenario 3	Diffusion paths + FM	**1.000** (± 0.000)	0.582 (± 0.280)	**8.708** (± 4.945)	**1.000** (± 0.000)	1.869 (± 0.342)	**33.230** (± 8.095)
	OT paths + FM	**0.999** (± 0.001)	**0.267** (± 0.154)	**7.234** (± 2.974)	**1.000** (± 0.000)	**1.155** (± 0.258)	**26.956** (± 3.114)

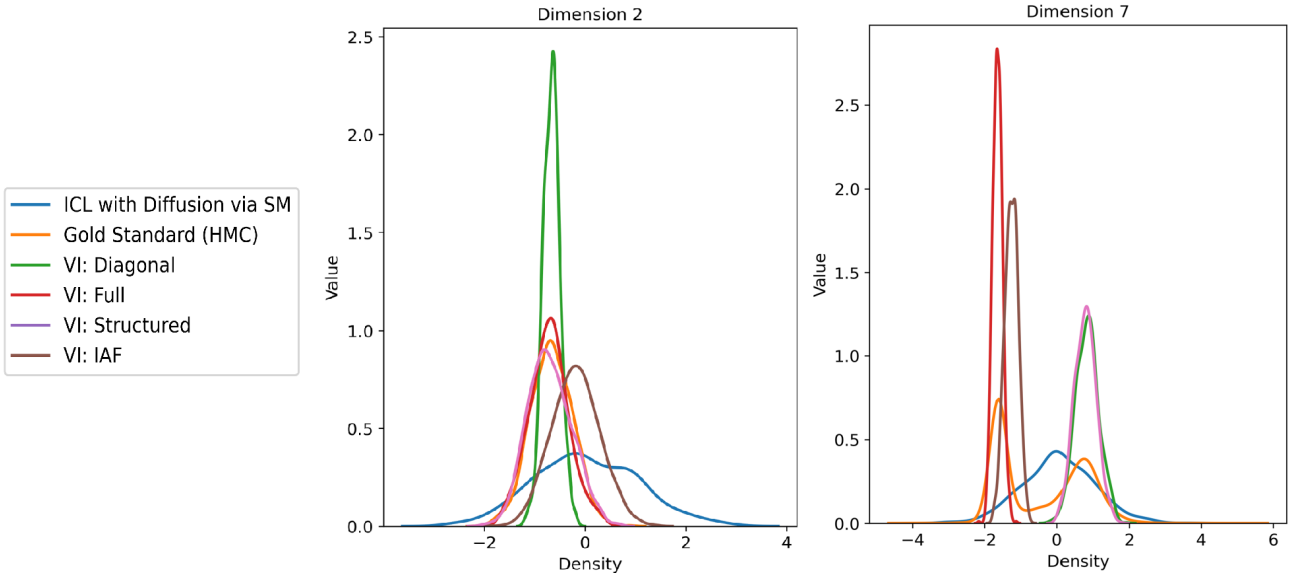


Figure 8: Marginal distribution for GLM scenario 2 (left) and GMM scenario 1 (right). The in-context learner is trained with a diffusion objective using VP paths.

exhibit better performance on real-world data (4.054 vs. 5.328). In Scenario 2, OT paths maintain a consistent advantage in metrics such as C2ST and $\mathcal{W}_2$. For instance, synthetic data shows a C2ST improvement from 0.942 to 0.812, while real-world data improves from 0.984 to 0.937. The OT paths also achieve lower MMD on synthetic data (0.159 vs. 0.213), supporting their effectiveness in this scenario. For Scenario 3, OT paths achieve better results for $\mathcal{W}_2$ on both synthetic and real-world data, reducing it from 8.708 to 7.234 and from 33.230 to 26.956, respectively.

L.2. Diffusion with Score-Matching

Second, we compare the results of using OT paths with flow matching to the results obtained when using VP paths and score matching. We use the score matching objective introduced by Song & Ermon (2019) and maintain the VP hyperparameters from Lipman et al. (2022) that we previously used for the diffusion objective with flow matching.

We find that, across all three considered GLM scenarios, using OT paths and flow matching yields substantially better results than using Diffusion VP paths and score matching, where the score-matching objective sometimes yields results comparable to those obtained using a Laplace approximation. We observe similar overall results for FA and GMMs, although the effect is less pronounced. Note that the inferiority of score matching compared to flow matching is consistent with findings by Lipman et al. (2022) and Dax et al. (2024), who also report that flow matching produces more stable and less noisy training trajectories.

The large quantity of noise in the diffusion objective might prevent the model from learning complex conditioning on datasets $\boldsymbol{x}$, which is arguably the main challenge for performing in-context learning for the posteriors of latent variable

models. We find visually that using the diffusion objective leads to a form of collapse where the model only learns a constant posterior distribution $Q_\theta^{\boldsymbol{z}|\boldsymbol{x}}$ that has a relatively large variance and is centered around zero, while largely ignoring the conditioning on $\boldsymbol{x}$ (Please refer to figure 8).

Table 24: GLMs: Comparison of the OT flow matching and the VP diffusion objective with score matching on 50 synthetic and 17 real-world datasets for three different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 2	Diffusion paths + SM	0.996 (± 0.011)	4.121 (± 1.625)	8.761 (± 4.415)	0.998 (± 0.002)	1.574 (± 0.906)	8.483 (± 1.580)
	OT paths + FM	**0.839** (± 0.072)	**0.707** (± 0.658)	**1.111** (± 0.300)	**0.768** (± 0.033)	**0.143** (± 0.089)	**0.411** (± 0.094)
Scenario 3	Diffusion paths + SM	0.965 (± 0.075)	2.466 (± 1.224)	3.947 (± 1.323)	0.994 (± 0.002)	2.018 (± 0.206)	3.301 (± 0.260)
	OT paths + FM	**0.611** (± 0.070)	**0.089** (± 0.114)	**0.423** (± 0.348)	**0.576** (± 0.027)	**0.037** (± 0.026)	**0.257** (± 0.044)
Scenario 5	Diffusion paths + SM	0.998 (± 0.002)	3.163 (± 0.651)	8.684 (± 1.135)	0.999 (± 0.001)	3.004 (± 0.056)	8.547 (± 0.177)
	OT paths + FM	**0.621** (± 0.063)	**0.067** (± 0.080)	**0.299** (± 0.195)	**0.610** (± 0.045)	**0.046** (± 0.020)	**0.242** (± 0.038)

Table 25: FA: Comparison of the OT flow matching and the VP diffusion objective with score matching on 50 synthetic and 17 real-world datasets for three different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	Diffusion paths + SM	0.880 (± 0.024)	0.875 (± 0.134)	1.787 (± 0.155)	0.906 (± 0.007)	0.845 (± 0.026)	1.723 (± 0.029)
	OT paths + FM	**0.552** (± 0.028)	**0.034** (± 0.034)	**0.289** (± 0.083)	**0.606** (± 0.038)	**0.068** (± 0.069)	**0.265** (± 0.078)
Scenario 2	Diffusion paths + SM	0.932 (± 0.022)	1.459 (± 0.128)	2.798 (± 0.141)	0.980 (± 0.008)	1.772 (± 0.065)	2.927 (± 0.085)
	OT paths + FM	**0.542** (± 0.006)	**0.017** (± 0.006)	**0.244** (± 0.033)	**0.622** (± 0.032)	**0.098** (± 0.039)	**0.287** (± 0.046)
Scenario 3	Diffusion paths + SM	0.925 (± 0.021)	1.747 (± 0.382)	3.028 (± 0.646)	0.989 (± 0.003)	2.101 (± 0.050)	2.882 (± 0.050)
	OT paths + FM	**0.537** (± 0.023)	**0.024** (± 0.021)	**0.259** (± 0.088)	**0.609** (± 0.019)	**0.124** (± 0.037)	**0.179** (± 0.018)

Table 26: GMMs: Comparison of the OT flow matching and the VP diffusion objective with score matching on 50 synthetic and 17 real-world datasets for three different scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	Diffusion paths + SM	1.000 (± 0.001)	1.412 (± 0.365)	7.038 (± 0.655)	0.998 (± 0.002)	1.574 (± 0.906)	8.483 (± 1.580)
	OT paths + FM	**0.760** (± 0.092)	**0.303** (± 0.548)	**2.095** (± 1.692)	**0.847** (± 0.082)	**0.486** (± 0.623)	**4.054** (± 2.782)
Scenario 2	Diffusion paths + SM	1.000 (± 0.000)	1.275 (± 0.240)	6.621 (± 1.091)	1.000 (± 0.000)	1.032 (± 0.163)	7.931 (± 0.748)
	OT paths + FM	**0.812** (± 0.061)	**0.159** (± 0.154)	**2.314** (± 0.926)	**0.937** (± 0.041)	**0.282** (± 0.131)	**3.947** (± 1.055)
Scenario 3	Diffusion paths + SM	**1.000** (± 0.000)	1.337 (± 0.476)	**10.877** (± 5.262)	**1.000** (± 0.000)	2.277 (± 0.245)	**24.269** (± 3.841)
	OT paths + FM	**0.999** (± 0.001)	**0.267** (± 0.154)	**7.234** (± 2.974)	**1.000** (± 0.000)	**1.155** (± 0.258)	**26.956** (± 3.114)

M. Ablation: Using an MLP-based Encoder

To further justify choosing a transformer encoder in our ICL approach, we conduct an ablation study comparing the performance of our original ICL method with the performance obtained when the transformer encoder is replaced by an MLP with batch normalization (Ioffe, 2015) and skip-connections. To ensure a fair comparison, we use an MLP encoder with a hidden dimension of 1250 to give the overall model approximately the same number of parameters as in the transformer-based approach. Concretely, our MLP-approach has 43.3 million parameters compared to 43.1 million parameters with the transformer encoder. We choose three selected GLM, FA and GMM scenarios with 50 synthetic and 17 real-world datasets for each scenario.

In summary, we find that the transformer encoder yields consistently better, results than the mlp encoder across all scenarios. While the difference is especially pronounced for the GLM scenarios, the difference become smaller for FA and GMMs.

Table 27: GLMs: Comparison when using an MLP-based encoder and a transformer encoder on 50 synthetic and 17 real-world datasets for three different scenarios.

Scenario	Type of Encoder	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 2	MLP	0.942 (± 0.093)	1.783 (± 1.048)	2.503 (± 0.814)	0.968 (± 0.012)	1.528 (± 0.394)	2.271 (± 0.315)
	Transformer	0.839 (± 0.072)	0.707 (± 0.658)	1.111 (± 0.300)	0.768 (± 0.033)	0.143 (± 0.089)	0.411 (± 0.094)
Scenario 3	MLP	0.957 (± 0.075)	2.236 (± 1.218)	2.681 (± 1.130)	0.972 (± 0.012)	1.658 (± 0.450)	2.076 (± 0.427)
	Transformer	0.611 (± 0.070)	0.089 (± 0.114)	0.423 (± 0.348)	0.576 (± 0.027)	0.037 (± 0.026)	0.257 (± 0.044)
Scenario 5	MLP	0.845 (± 0.115)	1.066 (± 0.859)	1.166 (± 0.996)	0.890 (± 0.055)	1.223 (± 0.791)	1.102 (± 0.383)
	Transformer	0.621 (± 0.063)	0.067 (± 0.080)	0.299 (± 0.195)	0.610 (± 0.045)	0.046 (± 0.020)	0.242 (± 0.038)

In Table 27, the transformer encoder consistently outperforms the MLP encoder across all metrics and scenarios. In Scenario 2, C2ST drops from 0.942 (MLP) to 0.839 (Transformer) on synthetic data and from 0.968 to 0.768 on real-world data. Similarly, $\mathcal{W}_2$ improves significantly, decreasing from 2.503 to 1.111 on synthetic data and from 2.271 to 0.411 on real-world data. In Scenario 3, transformers achieve substantial improvements, reducing C2ST from 0.957 (MLP) to 0.611 on synthetic data and from 0.972 to 0.576 on real-world data. $\mathcal{W}_2$ also sees notable reductions, dropping from 2.681 to 0.423 on synthetic data and from 2.076 to 0.257 on real-world data. Finally, in Scenario 5, transformers maintain their superiority, achieving reductions in C2ST from 0.845 (MLP) to 0.621 on synthetic data and from 0.890 to 0.610 on real-world data. Improvements in $\mathcal{W}_2$ are similarly remarkable, with reductions from 1.166 to 0.299 on synthetic data and from 1.102 to 0.242 on real-world data.

Table 28: FA: Comparison when using an MLP-based encoder and a transformer encoder on 50 synthetic and 17 real-world datasets for three different scenarios.

Scenario	Type of Encoder	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	MLP	0.579 (± 0.015)	0.017 (± 0.006)	0.364 (± 0.029)	0.634 (± 0.014)	0.013 (± 0.004)	0.331 (± 0.010)
	Transformer	0.552 (± 0.028)	0.034 (± 0.034)	0.289 (± 0.083)	0.606 (± 0.038)	0.068 (± 0.069)	0.265 (± 0.078)
Scenario 2	MLP	0.562 (± 0.038)	0.037 (± 0.042)	0.308 (± 0.097)	0.632 (± 0.068)	0.182 (± 0.407)	0.339 (± 0.174)
	Transformer	0.542 (± 0.006)	0.017 (± 0.006)	0.244 (± 0.033)	0.622 (± 0.032)	0.098 (± 0.039)	0.287 (± 0.046)
Scenario 3	MLP	0.539 (± 0.025)	0.023 (± 0.022)	0.278 (± 0.116)	0.680 (± 0.019)	0.268 (± 0.044)	0.253 (± 0.017)
	Transformer	0.537 (± 0.023)	0.024 (± 0.021)	0.259 (± 0.088)	0.609 (± 0.019)	0.124 (± 0.037)	0.179 (± 0.018)

For the factor analysis cases (Table 28), the transformer encoder still has better average performances even though the differences are substantially less pronounced than for the GLMs. In Scenario 1, transformers slightly outperform MLPs, reducing C2ST from 0.579 to 0.552 on synthetic data and from 0.634 to 0.606 on real-world data. $\mathcal{W}_2$ also sees moderate improvements, dropping from 0.364 to 0.289 on synthetic data and from 0.331 to 0.265 on real-world data. In Scenario 2, the advantage of the transformer encoder remains consistent, with C2ST decreasing from 0.562 (MLP) to 0.542 on synthetic data and from 0.632 to 0.622 on real-world data. $\mathcal{W}_2$ also improves slightly, dropping from 0.308 to 0.244 on synthetic data and from 0.339 to 0.287 on real-world data. Scenario 3 shows the smallest differences, where transformers marginally improve C2ST from 0.539 (MLP) to 0.537 on synthetic data and from 0.680 to 0.609 on real-world data. For $\mathcal{W}_2$, the reductions are minor but consistent, dropping from 0.278 to 0.259 on synthetic data and from 0.253 to 0.179 on real-world data.

Table 29: GMMs: Comparison when using an MLP-based encoder and a transformer encoder on 50 synthetic and 17 real-world datasets for three different scenarios.

Scenario	Type of Encoder	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	MLP	0.873 (± 0.045)	0.242 (± 0.363)	2.203 (± 1.098)	0.917 (± 0.067)	0.891 (± 1.150)	4.528 (± 2.701)
	Transformer	0.760 (± 0.092)	0.303 (± 0.548)	2.095 (± 1.692)	0.847 (± 0.082)	0.486 (± 0.623)	4.054 (± 2.782)
Scenario 2	MLP	0.921 (± 0.035)	0.291 (± 0.205)	2.870 (± 0.710)	0.992 (± 0.005)	0.399 (± 0.127)	5.505 (± 1.144)
	Transformer	0.812 (± 0.061)	0.159 (± 0.154)	2.314 (± 0.926)	0.937 (± 0.041)	0.282 (± 0.131)	3.947 (± 1.055)
Scenario 3	MLP	0.999 (± 0.000)	0.438 (± 0.181)	11.502 (± 9.719)	1.000 (± 0.000)	1.001 (± 0.149)	26.282 (± 3.731)
	Transformer	0.999 (± 0.001)	0.267 (± 0.154)	7.234 (± 2.974)	1.000 (± 0.000)	1.155 (± 0.258)	26.956 (± 3.114)

For the Gaussian Mixture Models (GMMs), the results indicate a more mixed performance where the transformer still performs slightly better (Table 29): In Scenario 1, transformer encoders slightly outperform MLPs on synthetic data, with C2ST improving from 0.873 (MLP) to 0.760 and $\mathcal{W}_2$ decreasing slightly from 2.203 to 2.095. However, on real-world data, MLPs perform marginally better in terms of MMD, reducing it from 0.486 to 0.242, while transformers show minor improvements in $\mathcal{W}_2$ from 4.528 to 4.054. In Scenario 2, transformers show a more noticeable advantage. On synthetic data, C2ST improves from 0.921 (MLP) to 0.812, and $\mathcal{W}_2$ decreases significantly from 2.870 to 2.314. On real-world data, transformers reduce C2ST from 0.992 to 0.937 and MMD from 0.399 to 0.282, along with a considerable improvement in $\mathcal{W}_2$ from 5.505 to 3.947. In Scenario 3, the differences between the two encoders are relatively small but still favor the transformers on synthetic data, with $\mathcal{W}_2$ decreasing from 11.502 (MLP) to 7.234. For real-world data, the results are nearly identical for C2ST (1.000 for both) but show a slight increase in $\mathcal{W}_2$ for the transformer from 26.282 to 26.956. Overall, for the GMMs, the transformer encoders demonstrate consistent improvements across scenarios for synthetic data, particularly in Scenarios 1 and 2. However, for real-world data, the performance differences are less pronounced.

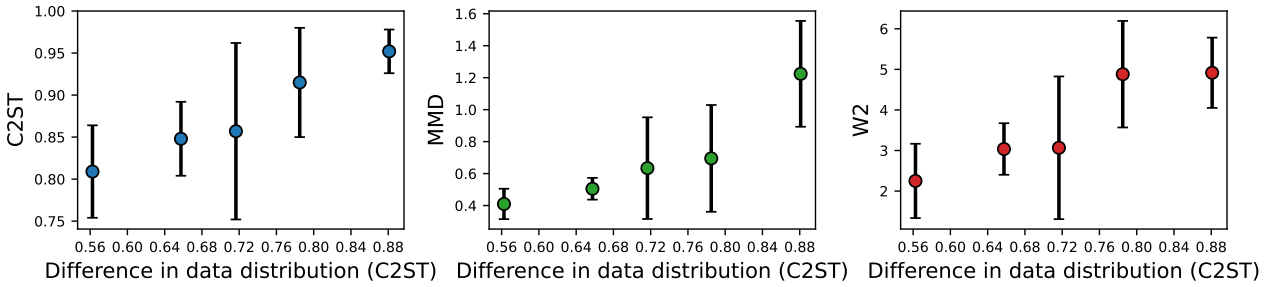


Figure 9: Out-of-distribution (OOD) performance of the ICL method in GLM Scenario 2. The x-axis shows the distribution shift between training and test distributions, quantified by C2ST. The y-axis displays the performance of the in-context learner, evaluated via C2ST, MMD, and $\mathcal{W}_2$ distances against HMC samples — where higher values indicate worse performance. OOD data is generated by gradually increasing the variance of the prior on the regression coefficients from an initial value of 1.0 to $\sqrt{2}$, 1.7, 2, 2.5, and 3.5.

N. Robustness to Out-of-distribution Data

To investigate how our ICL approach behaves under mismatches between the distribution of synthetic training data and the data used to infer the posterior, we conduct an ablation study by changing aspects of the distribution of training and testing data.

In summary, the results in Tables 31, 33 and 35 show that our ICL approach is, in most cases, capable of robustly generalizing beyond its specific pre-training distribution when various aspects of this distribution are changed. While the performance sometimes decreases when a mismatch between training and testing data occurs, the drops in performance are almost always modest and, in many cases, almost negligible.

N.1. GLM Scenarios

For scenario 2, we change the variance of the prior on the covariates from a value of $\mathbb{V}(\boldsymbol{\beta}_{i,j}) = 1$ to $\mathbb{V}(\boldsymbol{\beta}_{i,j}) = 2$ for scenario 2.B and $\mathbb{V}(\boldsymbol{\beta}_{i,j}) = 4$ for scenario 2.C. In scenarios 2.D and 2.E we change the scale parameter of the prior on the variance σ^2 of the noise—thereby changing its mean from $\mathbb{E}[\sigma^2] = 0.5$ to a value of $\mathbb{E}[\sigma^2] \approx 0.7071$ for 2.D and $\mathbb{E}[\sigma^2] = 1$ for 2.E. The variance is changed from $\mathbb{V}[\sigma^2] \approx 0.0833$ to $\mathbb{V}[\sigma^2] \approx 0.1667$ and $\mathbb{V}[\sigma^2] \approx 0.333$.

For scenarios 3.B and 3.C, the variance of the coefficients is doubled from scenario 3 to scenario 3.B and from 3.B to 3.C again, analogously to scenarios 2.B and 2.C.0

For scenario 5, the rate parameter of the gamma distribution is changed. This leads to a decrease in the variance from $\mathbb{V}(\boldsymbol{\beta}_{i,j}) = 1$ to $\mathbb{V}(\boldsymbol{\beta}_{i,j}) = 0.5$ for scenario 5.B and $\mathbb{V}(\boldsymbol{\beta}_{i,j}) = 0.25$ for scenario 5.C. Notably, we also change the mean in the distribution of the covariates from mean from $\mathbb{E}[\boldsymbol{\beta}_{i,j}] = 1$ to a value of $\mathbb{E}[\boldsymbol{\beta}_{i,j}] \approx 0.7071$ for 2.D and $\mathbb{E}[\boldsymbol{\beta}_{i,j}] = 0.5$ for 2.E.

Table 30 shows that our ICL approach only exhibits modest degradation in performance when the variance of the coefficients is doubled or quadruple while the mean stays the same (Scenarios 2.B, 2.C and 3.B, 3.C). Increasing the variance of the noise term by a factor of two only has a small effect while multiplying it by four causes a drop in C2ST by 9.3%. However, decreasing the variance of the gamma prior in scenario 5, combined with decreasing the mean, leads to a notable drop in performance across all metrics.

N.2. FA Scenarios

To construct the mismatch between training and test distribution, we vary the variance of the factor loading $W_{i,j,k}$ for scenarios 1, 2 and 3. Concretely, the variance is doubled and quadrupled.

For the FA cases (refer to Table 33), there is a notable drop in performance in the first scenario when OOD data is used. Please note that even in the most misspecified scenario (1.C), the performance, as measured in C2ST is still around ten

Table 30: Distribution of variables for the OOD analysis on GLM scenarios.

Scenario	$\beta_{i,j}$	$\beta_{i,0}$	σ_i^2	$y_{i,j}\|(\boldsymbol{u}_{i,j}, \boldsymbol{\beta}_i, \beta_{0,i}, \sigma_i^2)$
Scenario 2	$\mathcal{N}(0,1)$	$\mathcal{N}(0,9)$	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 2.B	$\mathcal{N}(0,2)$	$\mathcal{N}(0,9)$	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 2.C	$\mathcal{N}(0,4)$	$\mathcal{N}(0,9)$	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 2.D	$\mathcal{N}(0,1)$	$\mathcal{N}(0,9)$	IG$(5,2\sqrt{2})$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 2.E	$\mathcal{N}(0,1)$	$\mathcal{N}(0,9)$	IG$(5,4)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 3	Laplace$(0,1)$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 3.B	Laplace$(0,\sqrt{2})$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 3.C	Laplace$(0,2)$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 5	Ga$(1,1)$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 5.B	Ga$(1,\sqrt{2})$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$
Scenario 5.C	Ga$(1,2)$	-	IG$(5,2)$	$\mathcal{N}(\boldsymbol{u}_{i,j}^\top \boldsymbol{\beta}_i, \sigma_i^2)$

Table 31: OOD Performance: Evaluation on 50 synthetic datasets for 8 different GLM scenarios. All results within two standard errors of the non-OOD result for each scenario are marked in **bold**.

Scenario	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 2	**0.839** ($\pm$ 0.072)	**0.707** ($\pm$ 0.658)	**1.111** ($\pm$ 0.300)
Scenario 2.B	**0.809** ($\pm$ 0.055)	**0.410** ($\pm$ 0.095)	**2.250** ($\pm$ 0.916)
Scenario 2.C	**0.857** ($\pm$ 0.105)	**0.634** ($\pm$ 0.318)	**3.067** ($\pm$ 1.759)
Scenario 2	**0.839** ($\pm$ 0.072)	**0.707** ($\pm$ 0.658)	**1.111** ($\pm$ 0.300)
Scenario 2.D	**0.840** ($\pm$ 0.109)	**0.916** ($\pm$ 1.123)	**4.007** ($\pm$ 3.261)
Scenario 2.E	**0.932** ($\pm$ 0.120)	**1.556** ($\pm$ 1.127)	**4.850** ($\pm$ 2.261)
Scenario 3	**0.611** ($\pm$ 0.070)	**0.089** ($\pm$ 0.114)	**0.423** ($\pm$ 0.348)
Scenario 3.B	**0.667** ($\pm$ 0.080)	**0.210** ($\pm$ 0.117)	1.172 ($\pm$ 0.258)
Scenario 3.C	**0.720** ($\pm$ 0.108)	**0.362** ($\pm$ 0.248)	1.891 ($\pm$ 0.678)
Scenario 5	**0.621** ($\pm$ 0.063)	**0.067** ($\pm$ 0.080)	**0.299** ($\pm$ 0.195)
Scenario 5.B	0.831 ($\pm$ 0.121)	0.479 ($\pm$ 0.200)	1.762 ($\pm$ 0.541)
Scenario 5.C	0.920 ($\pm$ 0.064)	0.753 ($\pm$ 0.424)	3.159 ($\pm$ 1.254)

percent better than the best VI method in this scenario (Table 46). While the absolute difference between performance on the training distribution and the test distribution is very small for scenarios 2 and 3, the difference is still not within two standard errors of the non-OOD performance because the standard error itself is quite small. The performance on the OOD data is still better than all other VI methods (see Table 3).

N.3. GMM Scenarios

To generate several distinct OOD scenarios based on the generative processes of GMMs, we vary scenario 2 in various ways. Note that the structure of the distributions is the same for all GMM scenarios—focusing on this specific scenario thus makes sense when considering OOD generalization. First, in scenario 2.B, we decrease the symmetric parameter of the Dirichlet prior on the assignments from 1 to 0.5 causing larger discrepancy in the number of points per cluster. In scenario 2.C we make the opposite change.

In scenarios 2.D and 2.E we first double and then quadruple the variance of the prior on the per-component variances $\sigma_{i,m,l}$. Finally, in scenarios 2.F and 2.G, the prior on the mean is made more dispersed compared to the training data.

On the GMM scenarios (Table 35), the sample quality obtained via ICL is surprisingly stable under various changes to the data-generating process. It is relatively unsurprising that changing the Dirichlet prior, i.e., making the cluster more or less uniform in their number of samples, might lead to cases the ICL method can generalize to relatively easily, as demonstrated in scenarios 2.B and 2.C. The most pronounced drop in performance results from increasing the variance of the prior on the

Table 32: Distribution of variables for the OOD analysis on the FA scenarios.

Scenario	K	P	$\mu_{i,j}$	$\Psi_{i,j,j}$	$W_{i,j,k}$	$z_{i,j}$	$\boldsymbol{z}_{dim}$
Scenario 1	50	3	$\mathcal{N}(0,1)$	IG$(5,1)$	$\mathcal{N}(0,1)$	$\mathcal{N}(0,1)$	3
Scenario 1.B	50	3	$\mathcal{N}(0,1)$	IG$(5,1)$	$\mathcal{N}(0,2)$	$\mathcal{N}(0,1)$	3
Scenario 1.C	50	3	$\mathcal{N}(0,1)$	IG$(5,1)$	$\mathcal{N}(0,4)$	$\mathcal{N}(0,1)$	3
Scenario 2	50	3	$\mathcal{N}(0,0.1)$	IG$(5,1)$	Laplace$(0,10)$	$\mathcal{N}(0,1)$	3
Scenario 2.B	50	3	$\mathcal{N}(0,0.1)$	IG$(5,1)$	Laplace$(0,10\cdot\sqrt{2})$	$\mathcal{N}(0,1)$	3
Scenario 2.C	50	3	$\mathcal{N}(0,0.1)$	IG$(5,1)$	Laplace$(0,20)$	$\mathcal{N}(0,1)$	3
Scenario 3	25	5	$\mathcal{N}(0,0.1)$	IG$(5,2)$	$\mathcal{N}(0,3)$	$\mathcal{N}(0,1)$	3
Scenario 3	25	5	$\mathcal{N}(0,0.1)$	IG$(5,2)$	$\mathcal{N}(0,3\cdot\sqrt{2})$	$\mathcal{N}(0,1)$	3
Scenario 3	25	5	$\mathcal{N}(0,0.1)$	IG$(5,2)$	$\mathcal{N}(0,6)$	$\mathcal{N}(0,1)$	3

Table 33: OOD Performance: Evaluation on 50 synthetic datasets for 6 different FA scenarios. All results within two standard errors of the non-OOD result for each scenario are marked in **bold**.

Scenario	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	**0.552** ($\pm$ 0.028)	**0.034** ($\pm$ 0.034)	**0.289** ($\pm$ 0.083)
Scenario 1.B	0.826 ($\pm$ 0.066)	0.656 ($\pm$ 0.384)	0.929 ($\pm$ 0.321)
Scenario 1.C	0.855 ($\pm$ 0.060)	0.837 ($\pm$ 0.494)	1.135 ($\pm$ 0.461)
Scenario 2	**0.542** ($\pm$ 0.006)	**0.017** ($\pm$ 0.006)	**0.244** ($\pm$ 0.033)
Scenario 2.B	0.580 ($\pm$ 0.069)	0.087 ($\pm$ 0.191)	0.393 ($\pm$ 0.291)
Scenario 2.C	0.589 ($\pm$ 0.076)	0.089 ($\pm$ 0.113)	0.446 ($\pm$ 0.233)
Scenario 3	**0.537** ($\pm$ 0.023)	**0.024** ($\pm$ 0.021)	**0.259** ($\pm$ 0.088)
Scenario 3.B	**0.544** ($\pm$ 0.028)	0.030 ($\pm$ 0.021)	**0.285** ($\pm$ 0.094)
Scenario 3.C	**0.533** ($\pm$ 0.025)	0.021 ($\pm$ 0.015)	**0.347** ($\pm$ 0.152)

standard deviation of the components of the mixture model (scenario 2.E), while increasing the variance of the mean vector relative to the standard deviation of the components has a less pronounced effect.

Table 34: Distribution for the OOD analysis of the GMM scenarios.

Scenario	K	M	L	$\boldsymbol{\phi}_i$	$\sigma^2_{i,m,l}$	$\mu_{i,m,l}\|\sigma^2_{i,m,l}$
Scenario 2	25	3	3	$\text{Dir}(1)$	$\text{IG}(5,2)$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$
Scenario 2.B	25	3	3	$\text{Dir}(0.5)$	$\text{IG}(5,2)$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$
Scenario 2.C	25	3	3	$\text{Dir}(2)$	$\text{IG}(5,2)$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$
Scenario 2.D	25	3	3	$\text{Dir}(1)$	$\text{IG}(5, 2\cdot\sqrt{2})$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$
Scenario 2.E	25	3	3	$\text{Dir}(1)$	$\text{IG}(5,4)$	$\mathcal{N}(0, 3\sigma^2_{i,m,l})$
Scenario 2.F	25	3	3	$\text{Dir}(1)$	$\text{IG}(5,2)$	$\mathcal{N}(0, 4\sigma^2_{i,m,l})$
Scenario 2.G	25	3	3	$\text{Dir}(1)$	$\text{IG}(5,2)$	$\mathcal{N}(0, 5\sigma^2_{i,m,l})$

Table 35: OOD Performance: Evaluation on 50 synthetic datasets for 6 different GMM scenarios. All results within two standard errors of the non-OOD result for each scenario are marked in **bold**.

Scenario	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 2	**0.812** ($\pm$ 0.061)	**0.159** ($\pm$ 0.154)	**2.314** ($\pm$ 0.926)
Scenario 2.B	**0.829** ($\pm$ 0.050)	**0.233** ($\pm$ 0.161)	**2.595** ($\pm$ 0.998)
Scenario 2.C	**0.816** ($\pm$ 0.057)	0.149 ($\pm$ 0.135)	2.272 ($\pm$ 0.654)
Scenario 2	**0.812** ($\pm$ 0.061)	**0.159** ($\pm$ 0.154)	**2.314** ($\pm$ 0.926)
Scenario 2.D	**0.812** ($\pm$ 0.076)	**0.148** ($\pm$ 0.091)	**2.557** ($\pm$ 0.837)
Scenario 2.E	**0.880** ($\pm$ 0.057)	**0.231** ($\pm$ 0.109)	**3.535** ($\pm$ 1.003)
Scenario 2	**0.812** ($\pm$ 0.061)	**0.159** ($\pm$ 0.154)	**2.314** ($\pm$ 0.926)
Scenario 2.F	**0.821** ($\pm$ 0.076)	**0.216** ($\pm$ 0.214)	**2.700** ($\pm$ 1.044
Scenario 2.G	**0.844** ($\pm$ 0.046)	**0.197** ($\pm$ 0.124)	**2.675** ($\pm$ 0.552)

O. Ablation: Dimensionality of the Problem

In this section, the effect of the dimensionality K of the latent variable $\boldsymbol{z} \in \mathbb{R}^K$ for the GLM scenarios is investigated.

In summary, our results show that forward-KL based VI and in particular MAP solutions perform strongly in terms of predictive performance, which is in line with results first presented by Mittal et al. (2025b;a).

In table 36, we find that the advantages of the in-context learning approach to deteriorate for higher dimensionalities, with the variational inference methods using a Gaussian approximation performing well for 20 dimensions. This finding is line with work by Mittal et al. (2025b;a). For 50 dimensions we find that in many cases the used metrics do not allow to significantly discriminate the performance of the different approaches. Note that the randomness in the results, especially for higher dimensionalities, can in rare cases lead to better mean values. This is most likely not significant when taking the standard error into account.

Similar to scenarios 1,2 and 3, we find that in scenarios 4,5 and 6 (Table 37) the advantages of the in-context learning approach to deteriorate for higher dimensionalities, with the variational inference methods using a Gaussian approximation performing well for 20 dimensions. For 50 dimensions we find that in many cases the used metrics do not allow to significantly discriminate the performance of the different approaches. Note that the randomness in the results, especially for higher dimensionalities, can in rare cases lead to better mean values. This is most likely not significant when taking the standard error into account.

Finally, the results in Table 38 show that also for this scenario 7, the advantages of the in-context learning approach to deteriorate for higher dimensionalities. However, in this specific scenario the in-context learner is not significantly different from the VI methods in terms of C2ST and MMD for 20 dimensions. For 50 dimensions we find that the VI method using IAF performs well, together with the in-context learning approach in terms of MMD while the C2ST score does not indicate a clear winner and $\mathcal{W}_2$ favors the other methods. Note that the randomness in the results, especially for higher dimensionalities, can in rare cases lead to better mean values. This is most likely not significant when taking the standard error into account.

Table 36: Generalized Linear Models: Ablation with respect to the dimensionality of the problem on 50 synthetic and 17 real-world datasets for scenarios 1,2 and 3. All results within two standard errors of the best average result for each scenario are marked in **bold**. Due to the limitations of the number of features in the real-world data, we can only use 5 datasets for 20 and one dataset for 50 dimensions.

Scenario	Dim.	Model	Synthetic Evaluation			Real-World Evaluation		
			C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	5	Laplace Approximation	1.000 (± 0.000)	2.738 (± 0.721)	**0.825** (± 0.279)	1.000 (± 0.000)	2.150 (± 0.323)	**0.642** (± 0.124)
		VI: DiagonalNormal	0.904 (± 0.076)	1.452 (± 0.984)	**0.669** (± 0.301)	0.797 (± 0.083)	0.612 (± 0.511)	**0.414** (± 0.152)
		VI: MultivariateNormal	**0.750** (± 0.128)	**0.735** (± 0.733)	**0.565** (± 0.292)	**0.607** (± 0.070)	**0.167** (± 0.196)	**0.301** (± 0.123)
		VI: Structured Normal	**0.753** (± 0.126)	**0.736** (± 0.737)	**0.570** (± 0.310)	**0.600** (± 0.070)	**0.169** (± 0.214)	**0.306** (± 0.131)
		VI: IAF	**0.777** (± 0.122)	**0.864** (± 0.844)	0.725 (± 0.523)	0.683 (± 0.132)	0.440 (± 0.559)	0.503 (± 0.383)
		HMC	**0.745** (± 0.130)	**0.722** (± 0.732)	**0.569** (± 0.301)	**0.595** (± 0.075)	**0.173** (± 0.213)	**0.321** (± 0.140)
		ICL (ours)	**0.765** (± 0.123)	**0.767** (± 0.727)	**0.585** (± 0.301)	**0.614** (± 0.074)	**0.175** (± 0.219)	**0.310** (± 0.138)
Scenario 1	20	Laplace Approximation	1.000 (± 0.000)	2.237 (± 0.024)	**3.252** (± 1.172)	1.000 (± 0.000)	2.206 (± 0.021)	2.792 (± 0.339)
		VI: DiagonalNormal	**0.843** (± 0.204)	1.056 (± 0.869)	**2.976** (± 0.927)	0.983 (± 0.019)	1.217 (± 0.463)	2.406 (± 0.348)
		VI: MultivariateNormal	**0.789** (± 0.140)	**0.714** (± 0.351)	**2.774** (± 0.995)	**0.768** (± 0.120)	**0.180** (± 0.159)	**2.064** (± 0.306)
		VI: Structured Normal	**0.792** (± 0.109)	**0.708** (± 0.153)	**2.703** (± 1.069)	**0.668** (± 0.080)	**0.168** (± 0.071)	**2.052** (± 0.275)
		VI: IAF	**0.832** (± 0.196)	**0.847** (± 1.016)	**4.015** (± 0.415)	0.939 (± 0.024)	0.508 (± 0.200)	3.140 (± 0.290)
		ICL (ours)	**0.849** (± 0.171)	**0.844** (± 0.905)	4.564 (± 0.622)	0.970 (± 0.020)	0.724 (± 0.287)	4.250 (± 0.312)
Scenario 1	50	Laplace Approximation	1.000 (± 0.000)	2.401 (± 0.282)	**5.152** (± 2.268)	1.000 (± nan)	2.339 (± nan)	6.642 (± nan)
		VI: DiagonalNormal	**0.812** (± 0.197)	**0.956** (± 1.016)	**5.541** (± 2.356)	0.915 (± nan)	0.508 (± nan)	6.200 (± nan)
		VI: MultivariateNormal	**0.839** (± 0.148)	**0.926** (± 0.682)	**5.514** (± 2.370)	0.905 (± nan)	0.790 (± nan)	6.258 (± nan)
		VI: Structured Normal	**0.823** (± 0.160)	**0.844** (± 0.480)	**5.752** (± 2.098)	0.910 (± nan)	1.122 (± nan)	6.898 (± nan)
		VI: IAF	**0.820** (± 0.182)	**0.814** (± 0.987)	**6.696** (± 1.207)	0.938 (± nan)	0.256 (± nan)	6.869 (± nan)
		ICL (ours)	**0.787** (± 0.217)	**1.015** (± 1.255)	**8.278** (± 0.821)	0.979 (± nan)	0.413 (± nan)	8.368 (± nan)
Scenario 2	5	Laplace Approximation	1.000 (± 0.000)	4.853 (± 2.333)	5.770 (± 5.946)	1.000 (± 0.000)	2.572 (± 0.206)	0.809 (± 0.149)
		VI: DiagonalNormal	0.957 (± 0.091)	3.906 (± 2.679)	5.628 (± 6.092)	0.892 (± 0.044)	0.847 (± 0.389)	**0.530** (± 0.175)
		VI: MultivariateNormal	0.910 (± 0.131)	3.407 (± 2.781)	5.584 (± 6.104)	0.820 (± 0.031)	0.243 (± 0.148)	**0.408** (± 0.118)
		VI: Structured Normal	0.908 (± 0.119)	3.139 (± 2.763)	5.480 (± 6.164)	0.824 (± 0.023)	0.215 (± 0.110)	**0.392** (± 0.109)
		VI: IAF	0.968 (± 0.063)	4.416 (± 2.473)	7.474 (± 6.235)	0.888 (± 0.067)	0.921 (± 0.860)	0.942 (± 0.733)
		ICL (ours)	**0.839** (± 0.072)	**0.707** (± 0.658)	**1.111** (± 0.300)	**0.768** (± 0.033)	**0.143** (± 0.089)	**0.411** (± 0.094)
Scenario 2	20	Laplace Approximation	1.000 (± 0.000)	2.314 (± 0.237)	3.069 (± 1.168)	1.000 (± 0.000)	2.222 (± 0.018)	2.847 (± 0.305)
		VI: DiagonalNormal	0.904 (± 0.168)	1.292 (± 0.937)	**2.863** (± 0.919)	0.990 (± 0.009)	1.277 (± 0.452)	2.483 (± 0.318)
		VI: MultivariateNormal	0.851 (± 0.134)	**0.492** (± 0.547)	**2.694** (± 0.916)	0.843 (± 0.069)	0.243 (± 0.170)	**2.166** (± 0.266)
		VI: Structured Normal	**0.697** (± 0.065)	**0.070** (± 0.099)	**2.497** (± 0.993)	**0.655** (± 0.031)	**0.029** (± 0.025)	**2.191** (± 0.271)
		VI: IAF	0.916 (± 0.110)	1.062 (± 1.076)	4.191 (± 0.623)	0.952 (± 0.025)	0.515 (± 0.242)	3.331 (± 0.371)
		ICL (ours)	0.955 (± 0.057)	1.131 (± 1.035)	4.945 (± 0.836)	0.968 (± 0.020)	0.724 (± 0.278)	4.356 (± 0.302)
Scenario 2	50	Laplace Approximation	**1.000** (± 0.000)	2.437 (± 0.271)	**5.728** (± 1.358)	1.000 (± nan)	2.350 (± nan)	5.620 (± nan)
		VI: DiagonalNormal	**0.853** (± 0.182)	0.787 (± 0.687)	**6.224** (± 1.225)	0.996 (± nan)	1.080 (± nan)	5.426 (± nan)
		VI: MultivariateNormal	**0.878** (± 0.150)	**0.688** (± 0.620)	**6.206** (± 1.244)	0.994 (± nan)	0.791 (± nan)	5.305 (± nan)
		VI: Structured Normal	**0.865** (± 0.081)	**0.186** (± 0.169)	5.874 (± 1.233)	0.819 (± nan)	0.093 (± nan)	5.660 (± nan)
		VI: IAF	**0.909** (± 0.130)	0.649 (± 0.650)	7.465 (± 0.335)	0.985 (± nan)	0.426 (± nan)	6.426 (± nan)
		ICL (ours)	**0.972** (± 0.039)	0.741 (± 0.713)	8.313 (± 0.608)	0.971 (± nan)	0.405 (± nan)	7.718 (± nan)
Scenario 3	5	Laplace Approximation	1.000 (± 0.000)	2.203 (± 0.997)	1.170 (± 0.949)	1.000 (± 0.000)	1.841 (± 0.185)	0.729 (± 0.175)
		VI: DiagonalNormal	0.866 (± 0.101)	1.069 (± 1.150)	0.846 (± 0.747)	0.797 (± 0.083)	0.526 (± 0.361)	0.480 (± 0.207)
		VI: MultivariateNormal	**0.656** (± 0.131)	**0.445** (± 1.061)	**0.660** (± 0.737)	**0.560** (± 0.035)	**0.032** (± 0.028)	**0.249** (± 0.069)
		VI: Structured Normal	**0.653** (± 0.125)	**0.421** (± 0.993)	**0.659** (± 0.736)	**0.552** (± 0.028)	**0.027** (± 0.015)	**0.239** (± 0.055)
		VI: IAF	0.751 (± 0.148)	0.939 (± 1.349)	0.964 (± 0.924)	0.673 (± 0.141)	0.399 (± 0.543)	0.563 (± 0.433)
		ICL (ours)	**0.611** (± 0.070)	**0.089** (± 0.114)	**0.423** (± 0.348)	**0.576** (± 0.027)	**0.037** (± 0.026)	**0.257** (± 0.044)
Scenario 3	20	Laplace Approximation	1.000 (± 0.000)	2.726 (± 1.116)	4.127 (± 1.927)	1.000 (± 0.000)	2.234 (± 0.092)	3.589 (± 0.519)
		VI: DiagonalNormal	**0.912** (± 0.134)	**1.704** (± 1.467)	**3.933** (± 1.574)	0.983 (± 0.014)	1.298 (± 0.443)	3.147 (± 0.557)
		VI: MultivariateNormal	**0.863** (± 0.113)	**0.937** (± 1.174)	**3.754** (± 1.650)	**0.796** (± 0.099)	**0.268** (± 0.226)	**2.645** (± 0.466)
		VI: Structured Normal	**0.768** (± 0.109)	**0.302** (± 0.518)	**3.151** (± 1.663)	**0.722** (± 0.073)	**0.131** (± 0.141)	**2.579** (± 0.399)
		VI: IAF	**0.908** (± 0.133)	**1.657** (± 1.476)	5.543 (± 1.120)	0.936 (± 0.041)	0.548 (± 0.341)	3.678 (± 0.670)
		ICL (ours)	**0.902** (± 0.076)	**1.053** (± 0.782)	6.206 (± 0.783)	0.932 (± 0.019)	0.635 (± 0.183)	5.281 (± 0.317)
Scenario 3	50	Laplace Approximation	**1.000** (± 0.000)	2.700 (± 0.789)	**8.841** (± 1.691)	1.000 (± nan)	2.348 (± nan)	7.049 (± nan)
		VI: DiagonalNormal	**0.870** (± 0.127)	**1.154** (± 1.321)	**9.180** (± 1.513)	0.997 (± nan)	1.393 (± nan)	6.791 (± nan)
		VI: MultivariateNormal	**0.896** (± 0.101)	**1.027** (± 1.157)	**9.175** (± 1.555)	0.998 (± nan)	1.092 (± nan)	6.667 (± nan)
		VI: Structured Normal	**0.873** (± 0.112)	**0.539** (± 0.667)	**9.118** (± 1.538)	0.958 (± nan)	0.420 (± nan)	6.665 (± nan)
		VI: IAF	**0.869** (± 0.124)	**0.751** (± 0.939)	**9.917** (± 0.870)	0.971 (± nan)	0.417 (± nan)	7.411 (± nan)
		ICL (ours)	**0.931** (± 0.062)	**0.784** (± 0.884)	10.063 (± 0.930)	0.965 (± nan)	0.347 (± nan)	8.482 (± nan)

Table 37: Generalized Linear Models: Ablation with respect to the dimensionality of the problem on 50 synthetic and 17 real-world datasets for scenarios 4, 5 and 6. All results within two standard errors of the best average result for each scenario are marked in **bold**. Due to the limitations of the number of features in the real-world data, we can only use 5 datasets for 20 and one dataset for 50 dimensions.

Scenario	Dim.	Model	Synthetic Evaluation			Real-World Evaluation		
			C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 4	5	Laplace Approximation	1.000 (± 0.000)	3.511 (± 2.025)	2.166 (± 1.722)	1.000 (± 0.000)	2.011 (± 0.058)	0.993 (± 0.144)
		VI: DiagonalNormal	0.968 (± 0.036)	2.798 (± 2.255)	2.065 (± 1.745)	0.916 (± 0.040)	0.928 (± 0.339)	0.732 (± 0.181)
		VI: MultivariateNormal	0.855 (± 0.123)	1.648 (± 2.052)	1.853 (± 1.745)	0.771 (± 0.017)	**0.087** (± 0.030)	**0.539** (± 0.070)
		VI: Structured Normal	0.847 (± 0.116)	1.505 (± 1.978)	1.889 (± 1.883)	0.769 (± 0.012)	**0.083** (± 0.018)	**0.543** (± 0.070)
		VI: IAF	0.942 (± 0.077)	3.029 (± 2.210)	3.554 (± 2.715)	0.833 (± 0.069)	0.636 (± 0.756)	0.978 (± 0.600)
		ICL (ours)	**0.753** (± 0.049)	**0.171** (± 0.153)	**0.631** (± 0.294)	**0.762** (± 0.015)	**0.105** (± 0.046)	**0.597** (± 0.104)
Scenario 4	20	Laplace Approximation	1.000 (± 0.000)	4.929 (± 1.611)	**8.863** (± 3.796)	**1.000** (± 0.000)	3.196 (± 0.841)	5.186 (± 1.533)
		VI: DiagonalNormal	**0.988** (± 0.060)	4.418 (± 2.013)	**9.364** (± 4.281)	0.997 (± 0.007)	3.095 (± 1.417)	6.098 (± 2.435)
		VI: MultivariateNormal	**0.986** (± 0.054)	3.388 (± 1.907)	**7.910** (± 4.070)	0.893 (± 0.087)	**0.534** (± 0.469)	**3.175** (± 0.751)
		VI: Structured Normal	**0.954** (± 0.076)	**2.254** (± 1.515)	**7.475** (± 4.224)	**0.727** (± 0.034)	**0.074** (± 0.070)	**2.877** (± 0.379)
		VI: IAF	**0.987** (± 0.059)	3.258 (± 1.415)	**9.865** (± 3.515)	0.955 (± 0.030)	0.629 (± 0.308)	4.098 (± 0.341)
		ICL (ours)	**0.978** (± 0.038)	**1.185** (± 0.720)	**11.335** (± 1.378)	0.972 (± 0.018)	0.668 (± 0.199)	9.937 (± 0.466)
Scenario 4	50	Laplace Approximation	**1.000** (± 0.000)	6.695 (± 1.329)	**12.323** (± 4.091)	1.000 (± nan)	5.491 (± nan)	7.518 (± nan)
		VI: DiagonalNormal	**0.965** (± 0.084)	2.395 (± 1.958)	**12.022** (± 3.673)	0.996 (± nan)	4.368 (± nan)	6.951 (± nan)
		VI: MultivariateNormal	**0.984** (± 0.054)	5.395 (± 1.847)	**12.141** (± 3.079)	1.000 (± nan)	5.146 (± nan)	9.002 (± nan)
		VI: Structured Normal	**0.982** (± 0.026)	**4.261** (± 1.191)	**11.126** (± 3.396)	0.869 (± nan)	3.181 (± nan)	7.065 (± nan)
		VI: IAF	**0.981** (± 0.048)	**4.609** (± 1.412)	**12.567** (± 3.131)	0.988 (± nan)	3.558 (± nan)	7.849 (± nan)
		ICL (ours)	**0.960** (± 0.045)	**3.792** (± 0.758)	**14.071** (± 0.894)	0.974 (± nan)	3.443 (± nan)	12.546 (± nan)
Scenario 5	5	Laplace Approximation	1.000 (± 0.000)	2.060 (± 0.472)	0.797 (± 0.577)	1.000 (± 0.000)	1.982 (± 0.126)	0.623 (± 0.084)
		VI: DiagonalNormal	0.866 (± 0.085)	0.954 (± 1.022)	0.651 (± 0.549)	0.810 (± 0.036)	0.441 (± 0.252)	0.384 (± 0.089)
		VI: MultivariateNormal	0.765 (± 0.100)	0.537 (± 1.019)	0.633 (± 1.067)	0.711 (± 0.038)	0.148 (± 0.093)	**0.279** (± 0.056)
		VI: Structured Normal	0.758 (± 0.098)	0.447 (± 0.818)	0.572 (± 0.816)	0.705 (± 0.032)	0.140 (± 0.081)	**0.269** (± 0.045)
		VI: IAF	0.814 (± 0.105)	0.953 (± 1.165)	0.881 (± 1.067)	0.777 (± 0.106)	0.684 (± 0.939)	0.625 (± 0.525)
		ICL (ours)	**0.621** (± 0.063)	**0.067** (± 0.080)	**0.299** (± 0.195)	**0.610** (± 0.045)	**0.046** (± 0.020)	**0.242** (± 0.038)
Scenario 5	20	Laplace Approximation	**1.000** (± 0.000)	2.367 (± 0.555)	2.780 (± 1.271)	**1.000** (± 0.000)	2.200 (± 0.041)	2.444 (± 0.619)
		VI: DiagonalNormal	**0.938** (± 0.098)	**1.153** (± 0.954)	**2.552** (± 1.147)	**0.967** (± 0.012)	0.547 (± 0.233)	**1.973** (± 0.452)
		VI: MultivariateNormal	**0.929** (± 0.082)	**0.710** (± 0.768)	**2.473** (± 1.145)	**0.928** (± 0.016)	**0.250** (± 0.079)	**1.776** (± 0.399)
		VI: Structured Normal	**0.909** (± 0.082)	**0.397** (± 0.442)	**2.246** (± 1.244)	**0.924** (± 0.018)	**0.202** (± 0.094)	**1.775** (± 0.430)
		VI: IAF	**0.934** (± 0.092)	1.325 (± 1.161)	4.899 (± 1.320)	**0.980** (± 0.016)	0.892 (± 0.404)	3.593 (± 0.597)
		ICL (ours)	**0.961** (± 0.046)	1.330 (± 1.125)	5.084 (± 1.297)	**0.981** (± 0.014)	1.162 (± 0.461)	4.804 (± 0.578)
Scenario 5	50	Laplace Approximation	**1.000** (± 0.000)	2.582 (± 0.606)	**5.765** (± 1.540)	1.000 (± nan)	2.322 (± nan)	3.485 (± nan)
		VI: DiagonalNormal	**0.925** (± 0.074)	0.925 (± 1.056)	**6.461** (± 1.877)	0.972 (± nan)	0.186 (± nan)	3.251 (± nan)
		VI: MultivariateNormal	**0.934** (± 0.064)	**0.825** (± 0.972)	**6.404** (± 1.882)	0.969 (± nan)	0.165 (± nan)	3.223 (± nan)
		VI: Structured Normal	**0.927** (± 0.068)	**0.481** (± 0.588)	**6.420** (± 1.970)	0.961 (± nan)	0.072 (± nan)	3.324 (± nan)
		VI: IAF	**0.925** (± 0.069)	**0.792** (± 0.975)	8.458 (± 0.864)	0.996 (± nan)	0.519 (± nan)	4.645 (± nan)
		ICL (ours)	**0.998** (± 0.002)	**0.762** (± 0.987)	8.195 (± 0.820)	1.000 (± nan)	0.984 (± nan)	7.288 (± nan)
Scenario 6	5	Laplace Approximation	1.000 (± 0.000)	2.026 (± 0.027)	1.612 (± 0.162)	1.000 (± 0.000)	1.993 (± 0.032)	1.299 (± 0.106)
		VI: DiagonalNormal	0.724 (± 0.060)	0.185 (± 0.082)	**0.787** (± 0.078)	0.703 (± 0.039)	0.147 (± 0.063)	0.637 (± 0.089)
		VI: MultivariateNormal	**0.534** (± 0.018)	**0.014** (± 0.006)	**0.581** (± 0.074)	**0.538** (± 0.019)	**0.016** (± 0.007)	**0.466** (± 0.029)
		VI: Structured Normal	**0.536** (± 0.016)	**0.014** (± 0.005)	**0.583** (± 0.071)	**0.536** (± 0.019)	**0.017** (± 0.009)	**0.469** (± 0.033)
		VI: IAF	0.542 (± 0.026)	0.031 (± 0.031)	0.613 (± 0.092)	**0.535** (± 0.015)	**0.015** (± 0.006)	**0.467** (± 0.031)
		ICL (ours)	**0.532** (± 0.019)	0.016 (± 0.008)	**0.590** (± 0.066)	0.556 (± 0.017)	0.035 (± 0.015)	**0.504** (± 0.038)
Scenario 6	20	Laplace Approximation	1.000 (± 0.000)	2.247 (± 0.006)	4.158 (± 0.243)	1.000 (± 0.000)	2.240 (± 0.007)	3.714 (± 0.127)
		VI: DiagonalNormal	**0.747** (± 0.138)	**0.136** (± 0.123)	**3.460** (± 0.361)	**0.836** (± 0.053)	**0.203** (± 0.086)	**2.977** (± 0.112)
		VI: MultivariateNormal	**0.621** (± 0.016)	**0.016** (± 0.002)	**3.564** (± 0.290)	**0.608** (± 0.017)	**0.015** (± 0.003)	**3.101** (± 0.115)
		VI: Structured Normal	**0.599** (± 0.015)	**0.012** (± 0.002)	**3.592** (± 0.267)	**0.584** (± 0.028)	**0.012** (± 0.002)	**3.120** (± 0.107)
		VI: IAF	**0.625** (± 0.040)	**0.019** (± 0.009)	**3.572** (± 0.266)	**0.636** (± 0.021)	**0.020** (± 0.005)	**3.106** (± 0.128)
		ICL (ours)	0.747 (± 0.148)	0.163 (± 0.144)	4.063 (± 0.184)	0.928 (± 0.030)	0.463 (± 0.162)	4.425 (± 0.314)
Scenario 6	50	Laplace Approximation	1.000 (± 0.000)	2.291 (± 0.003)	**6.742** (± 0.362)	1.000 (± nan)	2.293 (± nan)	6.587 (± nan)
		VI: DiagonalNormal	**0.761** (± 0.138)	**0.087** (± 0.083)	**6.909** (± 0.743)	0.905 (± nan)	0.175 (± nan)	6.403 (± nan)
		VI: MultivariateNormal	**0.797** (± 0.100)	**0.069** (± 0.055)	**6.956** (± 0.736)	0.891 (± nan)	0.110 (± nan)	6.473 (± nan)
		VI: Structured Normal	**0.647** (± 0.017)	**0.013** (± 0.002)	7.218 (± 0.506)	0.654 (± nan)	0.013 (± nan)	6.890 (± nan)
		VI: IAF	**0.639** (± 0.038)	**0.014** (± 0.006)	7.204 (± 0.463)	0.692 (± nan)	0.024 (± nan)	6.887 (± nan)
		ICL (ours)	**0.742** (± 0.178)	0.115 (± 0.124)	7.713 (± 0.120)	0.935 (± nan)	0.203 (± nan)	7.846 (± nan)

Table 38: Generalized Linear Models: Ablation with respect to the dimensionality of the problem on 50 synthetic and 17 real-world datasets for scenario 7. All results within two standard errors of the best average result for each scenario are marked in **bold**. Due to the limitations of the number of features in the real-world data, we can only use 5 datasets for 20 and one dataset for 50 dimensions.

Scenario	Dim.	Model	Synthetic Evaluation			Real-World Evaluation		
			C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 7	5	Laplace Approximation	1.000 (± 0.000)	3.559 (± 1.933)	1.347 (± 1.067)	1.000 (± 0.000)	2.016 (± 0.080)	0.763 (± 0.174)
		VI: DiagonalNormal	0.938 (± 0.074)	2.536 (± 2.097)	1.142 (± 0.993)	0.936 (± 0.024)	1.029 (± 0.255)	0.579 (± 0.181)
		VI: MultivariateNormal	0.814 (± 0.181)	1.999 (± 2.283)	1.033 (± 0.969)	**0.741** (± 0.020)	0.093 (± 0.025)	**0.391** (± 0.074)
		VI: Structured Normal	0.824 (± 0.177)	1.891 (± 2.127)	1.041 (± 0.934)	**0.734** (± 0.025)	**0.072** (± 0.019)	**0.385** (± 0.065)
		VI: IAF	0.939 (± 0.091)	2.707 (± 1.712)	1.590 (± 0.820)	0.864 (± 0.093)	0.830 (± 0.697)	1.064 (± 0.616)
		ICL (ours)	**0.700** (± 0.116)	**0.317** (± 0.355)	**0.400** (± 0.286)	0.773 (± 0.048)	**0.294** (± 0.457)	0.559 (± 0.256)
Scenario 7	20	Laplace Approximation	1.000 (± 0.000)	3.581 (± 2.147)	**3.365** (± 1.583)	1.000 (± 0.000)	2.213 (± 0.024)	2.539 (± 0.378)
		VI: DiagonalNormal	**0.887** (± 0.184)	2.819 (± 2.732)	3.637 (± 1.371)	**0.996** (± 0.005)	1.734 (± 0.314)	**2.348** (± 0.423)
		VI: MultivariateNormal	**0.881** (± 0.164)	2.265 (± 2.573)	**3.524** (± 1.392)	**0.916** (± 0.085)	0.766 (± 0.535)	**2.043** (± 0.516)
		VI: Structured Normal	**0.850** (± 0.162)	**1.667** (± 2.266)	**3.186** (± 1.315)	**0.849** (± 0.105)	**0.391** (± 0.244)	**1.880** (± 0.367)
		VI: IAF	**0.867** (± 0.184)	**1.629** (± 1.584)	4.875 (± 1.239)	0.986 (± 0.007)	0.895 (± 0.361)	4.096 (± 0.319)
		ICL (ours)	**0.867** (± 0.185)	**1.428** (± 1.352)	4.836 (± 1.032)	**0.982** (± 0.010)	**0.820** (± 0.324)	4.177 (± 0.368)
Scenario 7	50	Laplace Approximation	1.000 (± 0.000)	4.768 (± 1.171)	**6.573** (± 1.038)	1.000 (± nan)	2.312 (± nan)	5.270 (± nan)
		VI: DiagonalNormal	**0.771** (± 0.191)	3.263 (± 1.853)	**6.919** (± 1.257)	1.000 (± nan)	2.237 (± nan)	5.417 (± nan)
		VI: MultivariateNormal	**0.816** (± 0.154)	3.245 (± 1.793)	6.978 (± 1.226)	0.997 (± nan)	2.117 (± nan)	5.781 (± nan)
		VI: Structured Normal	**0.795** (± 0.171)	3.126 (± 1.677)	**6.918** (± 1.260)	1.000 (± nan)	1.879 (± nan)	5.461 (± nan)
		VI: IAF	**0.769** (± 0.189)	**2.534** (± 0.894)	7.895 (± 0.843)	0.994 (± nan)	0.584 (± nan)	7.626 (± nan)
		ICL (ours)	**0.732** (± 0.216)	**2.451** (± 0.790)	7.787 (± 0.661)	0.980 (± nan)	0.411 (± nan)	7.461 (± nan)

P. Comparison to SGLD

Besides comparing the samples from our ICL approach to samples from various VI methods, we additionally compare it against samples generated via stochastic gradient Langevin dynamics (SGLD) (Welling & Teh, 2011). We run SGLD with a learning rate of 10^{-3} for the GLM and GMM cases and a learning rate of 10^{-4} for FA and use 1000 gradient steps for warmup and partition the data into ten minibatches. We implement the preconditioning method introduced by (Li et al., 2016) for more stable sampling behavior. Despite the preconditioning, SGLD consistently fails for GLMs scenario 7 because the sampler diverges causing singular covariance matrices. To facilitate running SGLD for the GMMs, which also include discrete variables, we marginalize over the discrete variables.

In summary, we find that ICL yields samples with much higher quality than SGLD compared to the gold standard HMC samples across almost all scenarios on both synthetic and real-world data. The poor sample quality with SGLD is expected given that numerous theoretical and empirical findings confirm that, while SGLD is computationally very cheap, it is substantially outperformed by, for instance, HMC, in terms of sample quality, which is especially pronounced when the posterior distributions are complex and parameters are correlated (Chen et al., 2014; Mangoubi & Vishnoi, 2019; Izmailov et al., 2021; Brosse et al., 2018) .

For GLMs (Table 42), ICL achieves significantly better results, with notable improvements in C2ST. In Scenario 1, synthetic C2ST drops from 0.992 to 0.765 and real-world C2ST from 0.980 to 0.614. Similarly, Scenario 3 shows substantial gains, with synthetic C2ST improving from 0.997 to 0.611 and real-world C2ST from 0.983 to 0.576. These trends extend to metrics like $\mathcal{W}_2$, where ICL yields consistent reductions.

For FA (Table 43), ICL also achieves superior performance, particularly in Scenarios 1 and 2. For example, in Scenario 1, synthetic C2ST decreases from 0.996 to 0.552, accompanied by improvements in $\mathcal{W}_2$ from 1.776 to 0.289. Scenario 2 sees further enhancements, with synthetic MMD dropping from 2.950 to 0.017 and real-world C2ST improving from 0.995 to 0.622.

For GMMs (Table 44), ICL demonstrates a clear advantage in most scenarios. In Scenario 1, ICL reduces synthetic C2ST from 1.000 to 0.760 and real-world $\mathcal{W}_2$ from 6.510 to 4.054. Scenario 2 shows synthetic C2ST improving from 1.000 to 0.812, and MMD from 3.046 to 0.159. While in scenarios 3, ICL has a singificantly lower MMD score on the synthetic data, the other differences are not significant.

Table 39: Evaluating the predictive performance across 50 synthetic and 17 real-world datasets in GLM scenario 2 for different dimensionalities. All results within two standard errors of the best average result for each scenario are marked in **bold**. Due to the limitations of the number of features in the real-world data, we can only use 5 datasets for 20 and one dataset for 50 dimensions. We find that the quality of the samples by the in-context learner, when evaluated based on predictive performance, decreases consistently with an increase in the dimensionality of the problem. Note that the randomness in the results, especially for higher dimensionalities, can in rare cases lead to better mean values. This is most likely not significant when taking the standard error into account.

Scenario	Dim.	Model	RMSE Real-World (↓)	RMSE Synthetic (↓)
Scenario 2	5	HMC	**0.559** (± 0.023)	**0.556** (± 0.049)
		Laplace Approximation	**0.561** (± 0.022)	**0.557** (± 0.049)
		VI: DiagonalNormal	**0.560** (± 0.023)	**0.557** (± 0.049)
		VI: MultivariateNormal	**0.559** (± 0.023)	**0.556** (± 0.049)
		VI: Structured Normal	**0.604** (± 0.016)	0.685 (± 0.054)
		VI: IAF	**0.563** (± 0.023)	**0.557** (± 0.049)
		ICL (ours)	**0.561** (± 0.019)	**0.653** (± 0.049)
		MAP	0.513 (± 0.023)	0.522 (± 0.048)
		TabPFN	0.449 (± 0.034)	0.498 (± 0.047)
Scenario 2	20	HMC	**0.682** (± 0.029)	**0.536** (± 0.041)
		Laplace Approximation	**0.682** (± 0.030)	**0.538** (± 0.040)
		VI: DiagonalNormal	**0.680** (± 0.029)	**0.539** (± 0.041)
		VI: MultivariateNormal	**0.685** (± 0.029)	**0.537** (± 0.041)
		VI: Structured Normal	0.746 (± 0.019)	0.681 (± 0.041)
		VI: IAF	**0.683** (± 0.029)	**0.539** (± 0.041)
		ICL (ours)	0.777 (± 0.011)	1.122 (± 0.078)
		MAP	0.578 (± 0.025)	0.472 (± 0.039)
		TabPFN	0.470 (± 0.044)	0.446 (± 0.038)
Scenario 2	50	HMC	0.669 (± nan)	**0.713** (± 0.060)
		Laplace Approximation	0.594 (± nan)	0.878 (± 0.068)
		VI: DiagonalNormal	0.582 (± nan)	0.870 (± 0.065)
		VI: MultivariateNormal	0.729 (± nan)	**0.764** (± 0.066)
		VI: Structured Normal	0.922 (± nan)	1.116 (± 0.074)
		VI: IAF	0.695 (± nan)	**0.770** (± 0.060)
		ICL (ours)	1.256 (± nan)	2.343 (± 0.230)
		MAP	0.301 (± nan)	0.398 (± 0.047)
		TabPFN	0.235 (± nan)	0.570 (± 0.053)

Table 40: Evaluating the predictive performance across 50 synthetic and 17 real-world datasets in GLM scenario 2 for different dimensionalities. All results within two standard errors of the best average result for each scenario are marked in **bold**. Due to the limitations of the number of features in the real-world data, we can only use 5 datasets for 20 and one dataset for 50 dimensions. We find that the quality of the samples by the in-context learner, when evaluated based on predictive performance, decreases consistently with an increase in the dimensionality of the problem.

Scenario	Dim.	Model	RMSE Real-World (↓)	RMSE Synthetic (↓)
Scenario 3	5	HMC	**0.684** (± 0.027)	**0.512** (± 0.040)
		Laplace Approximation	**0.688** (± 0.026)	**0.516** (± 0.040)
		VI: DiagonalNormal	**0.686** (± 0.027)	**0.513** (± 0.040)
		VI: MultivariateNormal	**0.685** (± 0.027)	**0.512** (± 0.040)
		VI: Structured Normal	**0.733** (± 0.016)	0.607 (± 0.043)
		VI: IAF	**0.686** (± 0.027)	**0.512** (± 0.040)
		ICL (ours)	**0.690** (± 0.023)	**0.588** (± 0.045)
		MAP	0.646 (± 0.028)	0.495 (± 0.039)
		TabPFN	0.556 (± 0.041)	0.462 (± 0.037)
Scenario 3	20	HMC	**1.030** (± 0.045)	**0.621** (± 0.046)
		Laplace Approximation	**1.053** (± 0.047)	0.755 (± 0.052)
		VI: DiagonalNormal	**1.035** (± 0.043)	0.734 (± 0.053)
		VI: MultivariateNormal	**1.033** (± 0.039)	**0.705** (± 0.055)
		VI: Structured Normal	**1.095** (± 0.045)	1.033 (± 0.063)
		VI: IAF	**1.026** (± 0.045)	**0.653** (± 0.047)
		ICL (ours)	1.770 (± 0.048)	2.160 (± 0.217)
		MAP	0.861 (± 0.038)	0.581 (± 0.050)
		TabPFN	0.654 (± 0.062)	0.475 (± 0.039)
Scenario 3	50	HMC	0.858 (± nan)	**0.645** (± 0.051)
		Laplace Approximation	0.866 (± nan)	0.865 (± 0.083)
		VI: DiagonalNormal	0.788 (± nan)	0.870 (± 0.084)
		VI: MultivariateNormal	0.819 (± nan)	0.778 (± 0.066)
		VI: Structured Normal	0.812 (± nan)	1.040 (± 0.103)
		VI: IAF	0.802 (± nan)	0.846 (± 0.078)
		ICL (ours)	1.686 (± nan)	3.477 (± 0.604)
		MAP	0.539 (± nan)	0.618 (± 0.054)
		TabPFN	0.322 (± nan)	0.534 (± 0.038)

Table 41: Evaluating the predictive performance across 50 synthetic and 17 real-world datasets in GLM scenario 2 for different dimensionalities. All results within two standard errors of the best average result for each scenario are marked in **bold**. Due to the limitations of the number of features in the real-world data, we can only use 5 datasets for 20 and one dataset for 50 dimensions. We find that the quality of the samples by the in-context learner, when evaluated based on predictive performance, decreases consistently with an increase in the dimensionality of the problem.

Scenario	Dim.	Model	RMSE Real-World (↓)	RMSE Synthetic (↓)
Scenario 5	5	HMC	**0.699** (± 0.022)	**0.490** (± 0.036)
		Laplace Approximation	**0.699** (± 0.022)	**0.491** (± 0.036)
		VI: DiagonalNormal	**0.702** (± 0.022)	**0.491** (± 0.036)
		VI: MultivariateNormal	**0.698** (± 0.021)	**0.491** (± 0.036)
		VI: Structured Normal	1.507 (± 0.089)	0.741 (± 0.053)
		VI: IAF	**0.699** (± 0.022)	**0.490** (± 0.036)
		ICL (ours)	0.769 (± 0.020)	0.701 (± 0.049)
		MAP	0.658 (± 0.022)	0.471 (± 0.035)
		TabPFN	0.534 (± 0.040)	0.442 (± 0.035)
Scenario 5	20	HMC	**1.527** (± 0.055)	**0.553** (± 0.044)
		Laplace Approximation	**1.585** (± 0.065)	**0.586** (± 0.043)
		VI: DiagonalNormal	**1.554** (± 0.058)	**0.586** (± 0.042)
		VI: MultivariateNormal	**1.530** (± 0.058)	**0.564** (± 0.043)
		VI: Structured Normal	2.109 (± 0.156)	1.054 (± 0.067)
		VI: IAF	**1.548** (± 0.057)	**0.562** (± 0.043)
		ICL (ours)	3.545 (± 0.288)	1.626 (± 0.140)
		MAP	1.254 (± 0.027)	0.464 (± 0.035)
		TabPFN	0.668 (± 0.064)	0.413 (± 0.032)
Scenario 5	50	HMC	1.626 (± nan)	**0.521** (± 0.028)
		Laplace Approximation	1.541 (± nan)	0.655 (± 0.040)
		VI: DiagonalNormal	1.576 (± nan)	0.639 (± 0.041)
		VI: MultivariateNormal	1.659 (± nan)	0.592 (± 0.035)
		VI: Structured Normal	2.076 (± nan)	1.018 (± 0.102)
		VI: IAF	1.706 (± nan)	0.627 (± 0.040)
		ICL (ours)	10.319 (± nan)	1.458 (± 0.193)
		MAP	1.318 (± nan)	0.416 (± 0.018)
		TabPFN	0.330 (± nan)	0.443 (± 0.024)

Table 42: SGLD vs. ICL: Evaluation on 50 synthetic and 17 real-world datasets for six different GLM scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	SGLD	0.992 (± 0.015)	2.846 (± 1.411)	1.951 (± 0.917)	0.980 (± 0.013)	2.191 (± 1.183)	0.865 (± 0.438)
	ICL (ours)	**0.765** (± 0.123)	**0.767** (± 0.727)	**0.585** (± 0.301)	**0.614** (± 0.074)	**0.175** (± 0.219)	**0.310** (± 0.138)
Scenario 2	SGLD	0.999 (± 0.004)	5.650 (± 1.762)	8.295 (± 5.629)	0.994 (± 0.006)	2.699 (± 1.093)	1.289 (± 0.454)
	ICL (ours)	**0.839** (± 0.072)	**0.707** (± 0.658)	**1.111** (± 0.300)	**0.768** (± 0.033)	**0.143** (± 0.089)	**0.411** (± 0.094)
Scenario 3	SGLD	0.997 (± 0.008)	3.320 (± 1.595)	3.011 (± 1.036)	0.983 (± 0.013)	2.152 (± 1.194)	0.935 (± 0.523)
	ICL (ours)	**0.611** (± 0.070)	**0.089** (± 0.114)	**0.423** (± 0.348)	**0.576** (± 0.027)	**0.037** (± 0.026)	**0.257** (± 0.044)
Scenario 4	SGLD	1.000 (± 0.000)	6.626 (± 1.215)	15.674 (± 8.100)	0.994 (± 0.006)	2.927 (± 1.564)	1.606 (± 1.022)
	ICL (ours)	**0.753** (± 0.049)	**0.171** (± 0.153)	**0.631** (± 0.294)	**0.762** (± 0.015)	**0.105** (± 0.046)	**0.597** (± 0.104)
Scenario 5	SGLD	0.999 (± 0.003)	3.308 (± 1.728)	2.216 (± 1.247)	1.000 (± 0.000)	4.012 (± 1.413)	0.996 (± 0.406)
	ICL (ours)	**0.621** (± 0.063)	**0.067** (± 0.080)	**0.299** (± 0.195)	**0.610** (± 0.045)	**0.046** (± 0.020)	**0.242** (± 0.038)
Scenario 6	SGLD	0.998 (± 0.001)	2.681 (± 0.565)	2.419 (± 0.510)	0.998 (± 0.002)	2.845 (± 0.590)	1.851 (± 0.319)
	ICL (ours)	**0.532** (± 0.019)	**0.016** (± 0.008)	**0.590** (± 0.066)	**0.556** (± 0.017)	**0.035** (± 0.015)	**0.504** (± 0.038)

Table 43: SGLD vs. ICL: Evaluation on 50 synthetic and 17 real-world datasets for six different FA scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	SGLD	0.996 (± 0.006)	2.883 (± 1.552)	1.776 (± 0.694)	0.995 (± 0.003)	2.676 (± 0.710)	1.608 (± 0.381)
	ICL (ours)	**0.552** (± 0.028)	**0.034** (± 0.034)	**0.289** (± 0.083)	**0.606** (± 0.038)	**0.068** (± 0.069)	**0.265** (± 0.078)
Scenario 2	SGLD	0.997 (± 0.003)	2.950 (± 0.786)	1.892 (± 0.533)	0.995 (± 0.003)	2.517 (± 0.583)	1.500 (± 0.268)
	ICL (ours)	**0.542** (± 0.006)	**0.017** (± 0.006)	**0.244** (± 0.033)	**0.622** (± 0.032)	**0.098** (± 0.039)	**0.287** (± 0.046)
Scenario 3	SGLD	0.998 (± 0.005)	3.662 (± 1.099)	2.086 (± 0.919)	0.956 (± 0.025)	1.580 (± 0.819)	0.311 (± 0.108)
	ICL (ours)	**0.537** (± 0.023)	**0.024** (± 0.021)	**0.259** (± 0.088)	**0.609** (± 0.019)	**0.124** (± 0.037)	**0.179** (± 0.018)
Scenario 4	SGLD	1.000 (± 0.000)	4.127 (± 0.635)	3.047 (± 0.972)	**0.950** (± 0.021)	**1.520** (± 0.512)	**0.141** (± 0.031)
	ICL (ours)	**0.684** (± 0.060)	**0.198** (± 0.141)	**0.918** (± 0.246)	0.988 (± 0.003)	1.764 (± 0.026)	1.248 (± 0.008)
Scenario 5	SGLD	0.999 (± 0.001)	3.465 (± 0.939)	1.981 (± 0.938)	0.962 (± 0.024)	1.945 (± 1.383)	**0.393** (± 0.243)
	ICL (ours)	**0.535** (± 0.016)	**0.021** (± 0.011)	**0.279** (± 0.060)	**0.886** (± 0.017)	**1.207** (± 0.101)	1.002 (± 0.042)
Scenario 6	SGLD	0.997 (± 0.004)	3.395 (± 1.199)	2.358 (± 1.458)	0.950 (± 0.040)	2.177 (± 1.643)	0.342 (± 0.224)
	ICL (ours)	**0.543** (± 0.021)	**0.023** (± 0.015)	**0.345** (± 0.173)	**0.666** (± 0.020)	**0.200** (± 0.034)	**0.224** (± 0.014)

Table 44: SGLD vs. ICL: Evaluation on 50 synthetic and 17 real-world datasets for four different GMM scenarios. All results within two standard errors of the best average result for each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation			Real-World Evaluation		
		C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)	C2ST (↓)	MMD (↓)	$\mathcal{W}_2$ (↓)
Scenario 1	SGLD	1.000 (± 0.001)	2.629 (± 0.868)	3.279 (± 1.330)	1.000 (± 0.000)	3.421 (± 0.877)	6.510 (± 1.763)
	ICL (ours)	**0.760** (± 0.092)	**0.303** (± 0.548)	**2.095** (± 1.692)	**0.847** (± 0.082)	**0.486** (± 0.623)	**4.054** (± 2.782)
Scenario 2	SGLD	1.000 (± 0.000)	3.046 (± 1.041)	6.015 (± 4.265)	1.000 (± 0.000)	2.487 (± 0.521)	6.858 (± 1.618)
	ICL (ours)	**0.812** (± 0.061)	**0.159** (± 0.154)	**2.314** (± 0.926)	**0.937** (± 0.041)	**0.282** (± 0.131)	**3.947** (± 1.055)
Scenario 3	SGLD	1.000 (± 0.000)	4.631 (± 1.169)	23.247 (± 30.646)	1.000 (± 0.000)	2.655 (± 0.437)	26.356 (± 2.699)
	ICL (ours)	**1.000** (± 0.000)	**0.582** (± 0.280)	**8.708** (± 4.945)	**1.000** (± 0.000)	**1.869** (± 0.342)	**33.230** (± 8.095)
Scenario 4	SGLD	**1.000** (± 0.000)	3.464 (± 1.098)	**6.995** (± 5.554)	**1.000** (± 0.000)	**2.555** (± 0.494)	**9.477** (± 3.432)
	ICL (ours)	**1.000** (± 0.000)	2.451 (± 0.868)	**8.333** (± 4.202)	1.000 (± 0.000)	**2.518** (± 0.694)	**11.938** (± 2.956)

Q. Evaluation the choice of classifier for the C2ST metric

In this section, we validate the choice of the classifier for the C2ST metric by comparing the ROC characteristic of a random forest (our default choice) and a neural network in distinguishing posterior samples. In summary, we find that despite minor differences, the two metrics yield the same overall results. Across all scenarios, both Random Forest (RF) and Neural Network NN classifiers yield quite consistent rankings of model performance with only insubstantial deviations in terms of the big picture. In particular, ICL is consistently among the top-performing approaches under both evaluation metrics. Out of the 14 total scenario–domain combinations (7 scenarios $\times$ 2 dataset types), the RF and NN metrics identify the same best-performing model in 12 cases.

Table 45: Generalized Linear Models: Comparison of C2ST scores with a Random Forest (RF) and a Neural Network (NN). For the NN we follow the setup of Lueckmann et al., 2021. Evaluation across seven distinct scenarios on 50 synthetic and 17 real-world datasets. All results within two standard errors of the best average result in each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation		Real-World Evaluation	
		C2ST RF (↓)	C2ST NN (↓)	C2ST RF (↓)	C2ST NN (↓)
Scenario 1	Laplace Approximation	1.000 (± 0.000)	0.998 (± 0.000)	1.000 (± 0.000)	0.998 (± 0.000)
	VI: DiagonalNormal	0.904 (± 0.076)	0.857 (± 0.001)	0.797 (± 0.083)	0.803 (± 0.004)
	VI: MultivariateNormal	**0.750** (± 0.128)	0.780 (± 0.002)	**0.607** (± 0.070)	0.713 (± 0.004)
	VI: Structured Normal	**0.753** (± 0.126)	0.781 (± 0.002)	**0.600** (± 0.070)	0.705 (± 0.004)
	VI: IAF	**0.777** (± 0.122)	0.793 (± 0.002)	0.683 (± 0.132)	0.746 (± 0.006)
	HMC	**0.745** (± 0.130)	0.777 (± 0.002)	**0.595** (± 0.075)	**0.702** (± 0.004)
	ICL (ours)	**0.765** (± 0.123)	**0.712** (± 0.002)	**0.614** (± 0.074)	**0.701** (± 0.004)
Scenario 2	Laplace Approximation	1.000 (± 0.000)	0.998 (± 0.000)	1.000 (± 0.000)	0.998 (± 0.000)
	VI: DiagonalNormal	0.957 (± 0.091)	0.883 (± 0.002)	0.892 (± 0.044)	0.851 (± 0.003)
	VI: MultivariateNormal	0.910 (± 0.131)	0.860 (± 0.002)	**0.820** (± 0.031)	0.815 (± 0.003)
	VI: Structured Normal	0.908 (± 0.119)	0.859 (± 0.002)	**0.824** (± 0.023)	0.817 (± 0.003)
	VI: IAF	0.968 (± 0.063)	0.889 (± 0.001)	0.888 (± 0.067)	0.849 (± 0.004)
	ICL (ours)	**0.839** (± 0.072)	**0.824** (± 0.001)	**0.768** (± 0.033)	**0.789** (± 0.003)
Scenario 3	Laplace Approximation	1.000 (± 0.000)	0.998 (± 0.000)	1.000 (± 0.000)	0.998 (± 0.000)
	VI: DiagonalNormal	0.866 (± 0.101)	0.838 (± 0.002)	0.797 (± 0.083)	0.803 (± 0.004)
	VI: MultivariateNormal	0.656 (± 0.131)	0.733 (± 0.002)	**0.590** (± 0.035)	0.685 (± 0.003)
	VI: Structured Normal	0.653 (± 0.125)	0.731 (± 0.002)	**0.582** (± 0.028)	0.681 (± 0.003)
	VI: IAF	0.751 (± 0.148)	0.780 (± 0.002)	0.673 (± 0.141)	0.741 (± 0.006)
	ICL (ours)	**0.611** (± 0.070)	**0.710** (± 0.001)	**0.576** (± 0.027)	**0.693** (± 0.003)
Scenario 4	Laplace Approximation	1.000 (± 0.000)	0.998 (± 0.000)	1.000 (± 0.000)	0.998 (± 0.000)
	VI: DiagonalNormal	0.968 (± 0.036)	0.889 (± 0.001)	0.916 (± 0.040)	0.863 (± 0.003)
	VI: MultivariateNormal	0.855 (± 0.123)	0.832 (± 0.002)	0.771 (± 0.017)	0.790 (± 0.002)
	VI: Structured Normal	0.847 (± 0.116)	0.828 (± 0.002)	0.769 (± 0.012)	0.789 (± 0.002)
	VI: IAF	0.942 (± 0.077)	0.876 (± 0.001)	0.833 (± 0.069)	0.821 (± 0.004)
	ICL (ours)	**0.753** (± 0.049)	**0.781** (± 0.001)	**0.762** (± 0.015)	**0.786** (± 0.002)
Scenario 5	Laplace Approximation	1.000 (± 0.000)	0.998 (± 0.000)	1.000 (± 0.000)	0.998 (± 0.000)
	VI: DiagonalNormal	0.866 (± 0.085)	0.838 (± 0.002)	0.810 (± 0.036)	0.810 (± 0.003)
	VI: MultivariateNormal	0.765 (± 0.100)	0.787 (± 0.002)	0.711 (± 0.038)	0.760 (± 0.003)
	VI: Structured Normal	0.758 (± 0.098)	0.784 (± 0.002)	0.705 (± 0.032)	0.757 (± 0.003)
	VI: IAF	0.814 (± 0.105)	0.812 (± 0.002)	0.777 (± 0.106)	0.793 (± 0.005)
	ICL (ours)	**0.621** (± 0.063)	**0.715** (± 0.001)	**0.610** (± 0.045)	**0.710** (± 0.003)
Scenario 6	Laplace Approximation	1.000 (± 0.000)	0.998 (± 0.000)	1.000 (± 0.000)	0.998 (± 0.000)
	VI: DiagonalNormal	0.724 (± 0.060)	0.767 (± 0.001)	0.703 (± 0.039)	0.756 (± 0.003)
	VI: MultivariateNormal	**0.534** (± 0.018)	**0.672** (± 0.001)	**0.538** (± 0.019)	0.674 (± 0.002)
	VI: Structured Normal	**0.536** (± 0.016)	**0.673** (± 0.001)	**0.536** (± 0.019)	0.673 (± 0.002)
	VI: IAF	0.542 (± 0.026)	0.676 (± 0.001)	**0.535** (± 0.015)	0.672 (± 0.002)
	ICL (ours)	**0.532** (± 0.019)	**0.671** (± 0.001)	0.556 (± 0.017)	**0.653** (± 0.002)
Scenario 7	Laplace Approximation	1.000 (± 0.000)	0.998 (± 0.000)	1.000 (± 0.000)	0.998 (± 0.000)
	VI: DiagonalNormal	0.938 (± 0.074)	0.874 (± 0.001)	0.936 (± 0.024)	0.873 (± 0.003)
	VI: MultivariateNormal	0.814 (± 0.181)	0.812 (± 0.002)	**0.741** (± 0.020)	0.775 (± 0.003)
	VI: Structured Normal	0.824 (± 0.177)	0.817 (± 0.002)	**0.734** (± 0.025)	0.772 (± 0.003)
	VI: IAF	0.939 (± 0.091)	0.874 (± 0.002)	0.864 (± 0.093)	0.837 (± 0.005)
	ICL (ours)	**0.700** (± 0.116)	**0.721** (± 0.002)	0.773 (± 0.048)	**0.751** (± 0.003)

Table 46: Factor Analysis: Comparison of C2ST scores using a Random Forest (RF) and a Neural Network (NN) classifier across six different scenarios on 50 synthetic and 17 real-world datasets. For the NN we follow the setup of Lueckmann et al., 2021. All results within two standard errors of the best average result in each scenario are marked in **bold**.

Scenario	Model	Synthetic Evaluation		Real-World Evaluation	
		C2ST RF (↓)	C2ST NN (↓)	C2ST RF (↓)	C2ST NN (↓)
Scenario 1	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	1.000 (± 0.001)	0.997 (± 0.000)	0.979 (± 0.008)	0.950 (± 0.001)
	VI: MultivariateNormal	0.998 (± 0.003)	0.960 (± 0.000)	0.966 (± 0.010)	0.944 (± 0.001)
	VI: Structured Normal	0.997 (± 0.004)	0.959 (± 0.000)	0.979 (± 0.010)	0.950 (± 0.001)
	VI: IAF	0.953 (± 0.104)	0.937 (± 0.001)	0.849 (± 0.075)	0.885 (± 0.003)
	ICL (ours)	**0.552** (± 0.028)	**0.737** (± 0.000)	**0.606** (± 0.038)	**0.764** (± 0.001)
Scenario 2	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	0.998 (± 0.002)	0.960 (± 0.000)	0.975 (± 0.010)	0.948 (± 0.001)
	VI: MultivariateNormal	0.989 (± 0.009)	0.955 (± 0.000)	0.951 (± 0.025)	0.936 (± 0.001)
	VI: Structured Normal	0.984 (± 0.031)	0.953 (± 0.000)	0.958 (± 0.025)	0.940 (± 0.001)
	VI: IAF	0.966 (± 0.066)	0.944 (± 0.001)	0.799 (± 0.058)	0.860 (± 0.002)
	ICL (ours)	**0.542** (± 0.006)	**0.732** (± 0.000)	**0.622** (± 0.032)	**0.772** (± 0.001)
Scenario 3	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	0.999 (± 0.002)	0.960 (± 0.000)	0.951 (± 0.007)	0.936 (± 0.001)
	VI: MultivariateNormal	0.994 (± 0.007)	0.958 (± 0.000)	0.945 (± 0.007)	0.933 (± 0.001)
	VI: Structured Normal	0.997 (± 0.003)	0.959 (± 0.000)	0.942 (± 0.009)	0.932 (± 0.001)
	VI: IAF	0.990 (± 0.011)	0.987 (± 0.000)	0.928 (± 0.015)	0.925 (± 0.001)
	ICL (ours)	**0.537** (± 0.023)	**0.729** (± 0.000)	**0.609** (± 0.019)	**0.765** (± 0.001)
Scenario 4	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	1.000 (± 0.000)	0.997 (± 0.000)	0.977 (± 0.003)	0.949 (± 0.000)
	VI: MultivariateNormal	0.999 (± 0.001)	0.960 (± 0.000)	0.973 (± 0.008)	0.947 (± 0.001)
	VI: Structured Normal	1.000 (± 0.000)	0.997 (± 0.000)	0.973 (± 0.007)	0.947 (± 0.001)
	VI: IAF	0.999 (± 0.001)	0.960 (± 0.000)	**0.961** (± 0.018)	0.941 (± 0.001)
	ICL (ours)	**0.684** (± 0.060)	**0.803** (± 0.001)	0.988 (± 0.003)	**0.955** (± 0.000)
Scenario 5	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	0.999 (± 0.002)	0.960 (± 0.000)	0.944 (± 0.010)	0.933 (± 0.001)
	VI: MultivariateNormal	0.995 (± 0.007)	0.958 (± 0.000)	0.930 (± 0.017)	0.926 (± 0.001)
	VI: Structured Normal	0.998 (± 0.005)	0.960 (± 0.000)	0.934 (± 0.011)	0.928 (± 0.001)
	VI: IAF	0.992 (± 0.012)	0.957 (± 0.000)	0.910 (± 0.011)	0.916 (± 0.001)
	ICL (ours)	**0.535** (± 0.016)	**0.728** (± 0.000)	**0.886** (± 0.017)	**0.904** (± 0.001)
Scenario 6	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	0.998 (± 0.002)	0.960 (± 0.000)	0.949 (± 0.008)	0.935 (± 0.001)
	VI: MultivariateNormal	0.991 (± 0.013)	0.956 (± 0.000)	0.938 (± 0.009)	0.930 (± 0.001)
	VI: Structured Normal	0.997 (± 0.005)	0.959 (± 0.000)	0.944 (± 0.006)	0.933 (± 0.001)
	VI: IAF	0.989 (± 0.029)	0.955 (± 0.000)	0.865 (± 0.027)	0.893 (± 0.001)
	ICL (ours)	**0.543** (± 0.021)	**0.732** (± 0.000)	**0.666** (± 0.020)	**0.794** (± 0.001)

Table 47: Gaussian Mixture Models: Comparison of C2ST scores using a Random Forest (RF) and a Neural Network (NN) classifier across six distinct scenarios on 50 synthetic and 17 real-world datasets. All results within two standard errors of the best average result in each scenario are marked in **bold**. For the NN we follow the setup of Lueckmann et al., 2021. Both RF and NN classifiers yield consistent rankings, with ICL emerging as the top method in scenarios with more pronounced model mismatch.

Scenario	Model	Synthetic Evaluation		Real-World Evaluation	
		C2ST RF (↓)	C2ST NN (↓)	C2ST RF (↓)	C2ST NN (↓)
Scenario 1	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	0.988 (± 0.013)	1.012 (± 0.000)	0.995 (± 0.006)	0.996 (± 0.001)
	VI: MultivariateNormal	0.988 (± 0.013)	1.012 (± 0.000)	0.994 (± 0.007)	0.993 (± 0.001)
	VI: Structured Normal	0.987 (± 0.015)	0.982 (± 0.000)	0.993 (± 0.009)	0.992 (± 0.001)
	VI: IAF	0.989 (± 0.013)	0.983 (± 0.000)	0.995 (± 0.010)	0.996 (± 0.001)
	ICL (ours)	**0.760** (± 0.092)	**0.825** (± 0.001)	**0.847** (± 0.082)	**0.869** (± 0.003)
Scenario 2	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	0.989 (± 0.024)	0.983 (± 0.000)	0.998 (± 0.003)	0.997 (± 0.001)
	VI: MultivariateNormal	0.991 (± 0.021)	0.991 (± 0.000)	0.999 (± 0.002)	1.002 (± 0.001)
	VI: Structured Normal	0.992 (± 0.017)	0.988 (± 0.000)	0.999 (± 0.002)	1.002 (± 0.001)
	VI: IAF	0.992 (± 0.021)	0.988 (± 0.000)	0.998 (± 0.004)	0.997 (± 0.001)
	ICL (ours)	**0.812** (± 0.061)	**0.851** (± 0.001)	**0.937** (± 0.041)	**0.915** (± 0.002)
Scenario 3	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	**0.996** (± 0.011)	1.004 (± 0.000)	**0.992** (± 0.018)	0.988 (± 0.001)
	VI: MultivariateNormal	0.997 (± 0.009)	1.007 (± 0.000)	**0.993** (± 0.016)	0.992 (± 0.001)
	VI: Structured Normal	**0.995** (± 0.017)	0.996 (± 0.000)	**0.993** (± 0.016)	0.992 (± 0.001)
	VI: IAF	**0.994** (± 0.018)	0.993 (± 0.000)	**0.993** (± 0.017)	0.992 (± 0.001)
	ICL (ours)	1.000 (± 0.000)	**0.997** (± 0.000)	1.000 (± 0.000)	**0.997** (± 0.000)
Scenario 4	Laplace Approximation	1.000 (± 0.000)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: DiagonalNormal	**1.000** (± 0.002)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: MultivariateNormal	**1.000** (± 0.002)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	VI: Structured Normal	**1.000** (± 0.001)	0.997 (± 0.000)	**0.996** (± 0.016)	1.004 (± 0.001)
	VI: IAF	**1.000** (± 0.002)	0.997 (± 0.000)	1.000 (± 0.000)	0.997 (± 0.000)
	ICL (ours)	1.000 (± 0.000)	**0.997** (± 0.000)	**1.000** (± 0.000)	**0.997** (± 0.000)