GENERALIZED MATHERON VARIATIONAL IMPLICIT PROCESSES

Anonymous authors
Paper under double-blind review

ABSTRACT

Implicit-process priors specify distributions over functions through sample-forward mechanisms such as Bayesian neural networks and stochastic simulators, but their function-space densities are typically unavailable. We introduce Generalized Matheron Variational Implicit Processes (GMVIP), a pathwise variational family for posterior inference with such priors. For Gaussian-process priors, GMVIP recovers the standard inducing-variable variational GP construction; for general implicit priors, its empirical covariance construction preserves the prior mean and covariance in the population limit. GMVIP constructs posterior samples by drawing a function from the prior and applying a correction anchored at a set of inducing inputs. The effect of this correction away from the inducing inputs is determined directly from prior samples, allowing the posterior to retain the structure and variability of the original implicit process. The (surrogate) prior and variational posterior use the same pathwise construction and differ only in the distribution of whitened inducing coefficients, yielding a tractable coefficient-space Kullback-Leibler divergence. Experiments on regression, classification, and forecasting with simulator-defined and retrieval-conditioned empirical trajectory priors show that GMVIP is broadly competitive with existing methods.

1 INTRODUCTION

Bayesian neural networks are commonly formulated by placing a prior over weights and performing posterior inference in parameter space (MacKay, 1992; Neal, 1996; Blundell et al., 2015; Hernández-Lobato and Adams, 2015; Gal and Ghahramani, 2016). Although parameter-space formulations enable familiar inference algorithms, they are high-dimensional and inherit issues such as non-identifiability—including neuron-permutation symmetries—strong posterior dependencies, multi-modality, and non-injective mappings from parameters to predictors (Sun et al., 2019; Burt et al., 2020; Wild et al., 2022; Rudner et al., 2022). Since predictions depend only on the induced input–output map, it is natural to instead formulate Bayesian inference directly in function space (Sun et al., 2019; Ma et al., 2019; Ortega, 2026).

Function-space inference targets the posterior distribution over functions rather than over parameters. This perspective aligns the variational object with the predictive object, but it also introduces a central computational difficulty: for flexible functional-space priors defined by Bayesian neural networks, neural samplers, and simulator-defined stochastic processes, the density of the induced function-space prior is rarely available or tractable (Ma et al., 2019; Ortega et al., 2023). Existing approaches address the intractable process Kullback-Leibler divergence in different ways: finite measurement sets with Stein score estimates (Sun et al., 2019), local linear-Gaussian approximations at finite context sets (Rudner et al., 2022), or alternative discrepancies between Gaussian measures (Wild et al., 2022). Burt et al. (2020) analyze when process-KL objectives are ill-defined and how finite-marginal approximations behave.

Implicit processes formalize this sample-forward setting. A draw from an implicit process gives a coherent random function that can be evaluated at any finite input set, but the corresponding finite-dimensional densities are not assumed to be available (Ma et al., 2019). Variational Implicit Processes (VIPs) make inference tractable by drawing a finite collection of prior functions and forming a surrogate over linear combinations of those samples (Ma et al., 2019). Sparse Implicit Processes introduce inducing-point structure for scalability (Rodríguez-Santana et al., 2022), and

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

$$f_s(\cdot) = g_s(\cdot) + \mathbf{\Psi}_{\mathbf{Z}}(\cdot)\left[\boldsymbol{\mu} + \mathbf{D}\mathbf{a}_s - g_s(\mathbf{Z})\right]$$

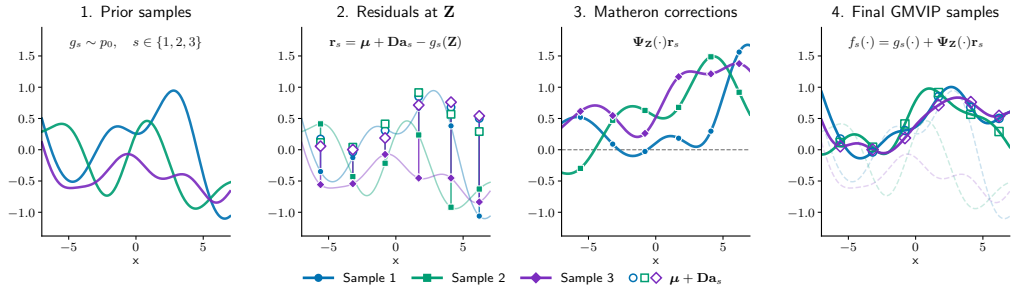


Figure 1: Pathwise construction of a GMVIP sample. Starting from a prior draw $g \sim p_0$, we evaluate the path at inducing inputs $\mathbf{Z}$ and form the residual between the sampled replacement values $\boldsymbol{\mu} + \mathbf{D}\mathbf{a}$ and $g(\mathbf{Z})$. The Matheron operator $\mathbf{\Psi}_{\mathbf{Z}}$ propagates this residual across the input domain. Adding the resulting smooth correction to g yields a sample f that matches the replacement values at $\mathbf{Z}$.

Deep Variational Implicit Processes build deeper implicit-process priors (Ortega et al., 2023). Recently, Flow-Transformed Implicit Processes (FTIP) showed that a Gaussian posterior over surrogate coefficients can smooth or collapse asymmetric, heavy-tailed, and multimodal posterior structure, and instead proposed normalizing flows over the finite surrogate variables (Ortega et al., 2026). These results suggest that posterior expressiveness is a central design choice in function-space inference.

This paper studies a complementary source of expressiveness. Instead of changing only the distribution over linear-combination coefficients, we change the pathwise construction of the surrogate itself. Conditioning jointly Gaussian variables admits a pathwise representation through a joint prior draw plus a covariance-gain correction, commonly called Matheron's rule; applying it to Gaussian-process marginals yields conditional function samples (Wilson et al., 2020). The corresponding conditional mean and covariance follow from standard Gaussian-process conditioning (Rasmussen and Williams, 2006). Sparse variational GPs introduce inducing variables and reduce the KL regularizer to $\mathrm{KL}(q(\mathbf{u})\|p(\mathbf{u}))$ (Titsias, 2009; Hensman et al., 2013). We take this inducing-value viewpoint to implicit processes: sample a prior path, replace its values at inducing locations, and retain the residual path away from those locations.

For clarity, we first present the scalar-output construction; the joint multioutput form is given in Section 4. Let $g \sim p_0$ be a prior function sample, let $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_M)^\top$ be inducing locations, and let $\mathbf{\Psi}_{\mathbf{Z}}$ be a replacement operator. For a latent vector $\mathbf{a} \in \mathbb{R}^M$, define

$$f(\cdot) = g(\cdot) + \mathbf{\Psi}_{\mathbf{Z}}(\cdot)\left[\boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z})\right]. \tag{1}$$

The vector $\boldsymbol{\mu} + \mathbf{D}\mathbf{a}$ gives replacement values at the inducing locations, while g supplies the residual path (Figure 1 shows an illustration on the idea). The surrogate prior uses $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$, and the variational posterior replaces this with $q_\phi(\mathbf{a})$. Because the prior and posterior share the same conditional map from $(g, \mathbf{a})$ to f, the variational regularizer is exactly $\mathrm{KL}(q_\phi(\mathbf{a})\|\mathcal{N}(\mathbf{0}, \mathbf{I}_M))$. Thus the method keeps the tractability of VIP-style coefficient inference while replacing linear aggregation with a Matheron-style inducing-value replacement. We call the resulting family Generalized Matheron Variational Implicit Processes (GMVIP). The construction is modular. The operator $\mathbf{\Psi}_{\mathbf{Z}}$ can be estimated from empirical prior-sample covariance, giving the closest analogue of GP Matheron conditioning, e.g., $\mathbf{\Psi}_{\mathbf{Z}} = \hat{\mathbf{K}}(\cdot, \mathbf{Z})\hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z})^{-1}$ for empirical covariance $\hat{\mathbf{K}}$.

Our contributions are: (i) We introduce a general Matheron inducing-value replacement family for variational inference with implicit-process priors. The family produces coherent posterior function samples and requires only prior sampling and likelihood evaluation. (ii) We show that the augmented KL reduces exactly to a coefficient-space KL, and that under a cardinal right-inverse condition the same identity holds for the induced function marginal. We also show process coherence for arbitrary query sets. (iii) We show exact GP/SVGP recovery for Gaussian priors and population-level preservation of the original prior mean and covariance for general implicit processes.

2 Related Work

Function-space variational inference. Function-space variational inference replaces approximate posteriors over parameters with approximate posteriors over predictors. This is especially relevant for Bayesian neural networks, where many parameter configurations can represent the same function. Functional variational Bayesian neural networks (fBNNs) formulate variational inference directly over the stochastic process induced by a network, typically approximating the process KL on finite measurement sets (Sun et al., 2019). Subsequent work clarified the measure-theoretic and optimization issues that arise when the variational object is a stochastic process rather than a finite-dimensional parameter vector (Burt et al., 2020; Wild et al., 2022). TFSVI constructs tractable finite-dimensional function-space approximations using local linearization and neural network Jacobians (Rudner et al., 2022). GMVIP shares the function-space objective, but avoids measurement-set density-ratio estimation or linearizations by defining a surrogate prior and posterior with a shared pathwise map.

Implicit processes. Implicit processes define stochastic processes through sampling mechanisms rather than explicit densities (Ma et al., 2019). VIP constructs finite surrogates from prior samples and performs variational inference over surrogate coefficients, yielding a tractable latent KL. Sparse Implicit Processes (SIP) introduce inducing variables and estimate the intractable divergence between implicit inducing distributions using adversarial density-ratio estimation (Rodríguez-Santana et al., 2022); Deep Variational Implicit Processes compose implicit-process layers to obtain richer priors (Ortega et al., 2023). FTIP makes the VIP surrogate posterior more expressive by replacing the Gaussian coefficient posterior with a normalizing flow (Ortega et al., 2026). Our method is complementary to FTIP. Rather than enriching $q(\mathbf{a})$ with an invertible transformation, we modify the pathwise surrogate: posterior samples are Matheron replacements of prior paths, not linear combinations of prior samples. Moreover, VIP/FTIP posterior draws are confined to the finite span $\mathrm{span}\{g^{(s)}\}_{s=1}^{S}$ of the S pre-sampled prior functions, whereas GMVIP draws a fresh prior path for each sample, preserving functional variation outside this finite-dimensional subspace.

Matheron sampling, inducing variables, and interpolation. Applying Matheron's rule to Gaussian-process marginals gives conditional function draws as a prior path plus a finite-dimensional covariance correction and underlies efficient pathwise GP posterior sampling (Wilson et al., 2020); the corresponding Gaussian conditioning formulas are standard (Rasmussen and Williams, 2006). Sparse variational GPs use inducing variables to obtain tractable objectives whose KL term is $\mathrm{KL}(q(\mathbf{u})\|p(\mathbf{u}))$ (Titsias, 2009; Hensman et al., 2013). Structured kernel interpolation, instantiated by KISS-GP, interpolates kernel quantities from inducing grid points to scale GP inference (Wilson and Nickisch, 2015). GMVIP borrows the pathwise inducing-value structure but applies it to priors that are only sampleable.

The empirical covariance-gain correction has the same linear-update algebra as statistical interpolation (Lorenc, 1981) and the analysis step of the ensemble Kalman filter (Evensen, 1994; Burgers et al., 1998); ensemble Kalman smoothing applies analogous ensemble-covariance corrections to earlier states (Evensen and van Leeuwen, 2000). GMVIP differs by optimizing a variational law over inducing replacement values and regularizing it with a coefficient-space KL.

3 Background: Function-Space Variational Inference and Implicit Processes

Let $\mathcal{D} = \{(\mathbf{x}_n, \mathbf{y}_n)\}_{n=1}^{N}$ denote the observed dataset, with $\mathbf{x}_n \in \mathcal{X}$ and outputs $\mathbf{y}_n \in \mathbb{R}^C$. We write $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_N)^\top$ and $\mathbf{Y} = (\mathbf{y}_1, \ldots, \mathbf{y}_N)^\top$, and assume a likelihood factorization

$$p(\mathbf{Y} \mid f, \mathbf{X}) = \prod_{n=1}^{N} p(\mathbf{y}_n \mid f(\mathbf{x}_n)). \tag{2}$$

Bayesian prediction is naturally a distribution over functions, since the likelihood depends on the predictor f rather than on a particular parameterization of f. This observation motivates function-space variational inference, where the goal is to approximate the posterior process $p(f \mid \mathcal{D})$ directly. In its ideal form, one optimizes

$$\mathcal{L}(q) = \mathbb{E}_{q(f)}[\log p(\mathbf{Y}|f, \mathbf{X})] - \mathrm{KL}(q(f)\|p_0(f)), \tag{3}$$

where p_0 is the prior process. This objective is conceptually attractive because it compares distributions over predictors, avoiding some of the redundancies and symmetries of weight-space inference. However, the process-level KL is rarely available. Existing methods use finite measurement sets with Stein score estimation (Sun et al., 2019), analyze or approximate finite-dimensional marginals (Burt et al., 2020), replace the KL with discrepancies between Gaussian measures (Wild et al., 2022), construct local linear-Gaussian approximations (Rudner et al., 2022), or introduce inducing representations with adversarial density-ratio estimation (Rodríguez-Santana et al., 2022). These approximations introduce additional choices: where to place measurement or inducing points, how to estimate finite-dimensional densities or ratios, and how to ensure that the resulting objectives define coherent posterior samples.

4 METHODOLOGY

We introduce GMVIP as a pathwise variational family for implicit-process priors. The central idea is to preserve the residual structure of a prior sample while replacing its values at a finite set of inducing inputs. This gives a posterior process that remains sample-forward, has coherent joint function samples, and admits a finite-dimensional KL.

Let $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_M)^\top$ be inducing inputs and let $g \sim p_0$ be a sample from the implicit-process prior. Given a whitened coefficient vector $\mathbf{a} \in \mathbb{R}^{d_a}$, where $d_a = M$ for scalar outputs and $d_a = MC$ for a joint C-output model, define the replacement inducing values

$$\mathbf{u}(\mathbf{a}) = \boldsymbol{\mu} + \mathbf{D}\mathbf{a}, \tag{4}$$

where $\boldsymbol{\mu}$ and $\mathbf{D}$ are fixed, prior-estimated, or determined by the chosen replacement operator. A GMVIP sample is obtained by replacing the prior path at $\mathbf{Z}$ and propagating the correction to arbitrary inputs:

$$T(g, \mathbf{a})(\cdot) = g(\cdot) + \boldsymbol{\Psi}_{\mathbf{Z}}(\cdot)\left[\mathbf{u}(\mathbf{a}) - g(\mathbf{Z})\right]. \tag{5}$$

The operator $\boldsymbol{\Psi}_{\mathbf{Z}}$ controls how inducing-value corrections are extended away from $\mathbf{Z}$. The surrogate prior and posterior share this same pathwise map:

$$p_{\mathbf{Z}}(g, \mathbf{a}, f) = p_0(g)\, p(\mathbf{a})\, \delta[f = T(g, \mathbf{a})], \tag{6}$$

$$q_{\mathbf{Z},\phi}(g, \mathbf{a}, f) = p_0(g)\, q_\phi(\mathbf{a})\, \delta[f = T(g, \mathbf{a})]. \tag{7}$$

Thus inference changes only the law of the finite coefficient vector $\mathbf{a}$; the residual path is still supplied by the original implicit prior. We use $p(\mathbf{a}) = \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_a})$ and $q_\phi(\mathbf{a}) = \mathcal{N}(\mathbf{m}_a, \mathbf{S}_a)$, with $\mathbf{m}_a \in \mathbb{R}^{d_a}$ and covariance $\mathbf{S}_a \in \mathbb{R}^{d_a \times d_a}$. Samples are drawn by reparameterization. For a minibatch $\mathcal{B}$, the Monte Carlo objective is

$$\widehat{\mathcal{L}}_t(\phi, \boldsymbol{\psi}) = \frac{N}{|\mathcal{B}| R_{\mathrm{tr}}} \sum_{r=1}^{R_{\mathrm{tr}}} \sum_{n \in \mathcal{B}} \log p_{\boldsymbol{\psi}}\big(\mathbf{y}_n \mid T(g^{(r)}, \mathbf{a}^{(r)})(\mathbf{x}_n)\big) - \mathrm{KL}\left(q_\phi(\mathbf{a}) \| p(\mathbf{a})\right), \tag{8}$$

with $g^{(r)} \sim p_0$, $\mathbf{a}^{(r)} \sim q_\phi$, and R_{tr} Monte Carlo samples per optimization step. Here $\boldsymbol{\psi}$ denotes the likelihood parameters. The most direct instantiation estimates the Matheron correction from prior samples. Given a fixed prior bank $\mathcal{G} = \{g^{(b)}\}_{b=1}^{B}$, $g^{(b)} \sim p_0$, define

$$\hat{\boldsymbol{\mu}}(\cdot) = \frac{1}{B} \sum_{b=1}^{B} g^{(b)}(\cdot), \qquad \hat{\mathbf{K}}(\cdot, \mathbf{Z}) = \frac{1}{B-1} \sum_{b=1}^{B} \big(g^{(b)}(\cdot) - \hat{\boldsymbol{\mu}}(\cdot)\big)\big(g^{(b)}(\mathbf{Z}) - \hat{\boldsymbol{\mu}}(\mathbf{Z})\big)^\top. \tag{9}$$

Let $\mathbf{L}$ be the Cholesky decomposition of $\hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z})$. The empirical replacement operator is

$$\boldsymbol{\Psi}^{\mathrm{emp}}_{\mathbf{Z}}(\cdot) = \hat{\mathbf{K}}(\cdot, \mathbf{Z})\hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z})^{-1}, \qquad \boldsymbol{\mu} = \hat{\boldsymbol{\mu}}(\mathbf{Z}), \qquad \mathbf{D} = \mathbf{L}. \tag{10}$$

Therefore,

$$f(\cdot) = g(\cdot) + \hat{\mathbf{K}}(\cdot, \mathbf{Z})\hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z})^{-1}\left[\hat{\boldsymbol{\mu}}(\mathbf{Z}) + \mathbf{L}\mathbf{a} - g(\mathbf{Z})\right]. \tag{11}$$

This is the prior-matched GMVIP operator: it uses the covariance geometry induced by the implicit prior itself. Its quality therefore depends on the prior-bank estimate, but when this estimate is reliable it provides the closest analogue of GP Matheron sampling for non-Gaussian implicit processes. The complete training and evaluation procedures are given in Algorithms 1 and 2 in Appendix B. Surrogate priors are already used in VIP (Ma et al., 2019); Appendix D.1 studies how much these priors differ from the original prior.

Multioutput case. For vector-valued outputs $f(\mathbf{x}) \in \mathbb{R}^C$, GMVIP supports both joint and channel-wise multioutput parameterizations. In the joint parameterization, we stack inducing evaluations and coefficients across outputs and construct the empirical inducing covariance $\hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z}) \in \mathbb{R}^{MC \times MC}$, including cross-output covariance blocks. The variational posterior $q_\phi(\mathbf{a}) = \mathcal{N}(\mathbf{m}_a, \mathbf{S}_a)$ may likewise use a full $MC \times MC$ covariance. For larger output spaces, we also use a channel-wise approximation in which the empirical operator and variational covariance are block-diagonal across outputs.

4.1 Theoretical properties

We now state the main properties of the construction. Proofs, together with additional results on prior-distortion bounds, prior reconstruction error, and sample-path regularity, are given in Appendix A.

Proposition 1 (Process validity)**.** *Assume p_0 is a valid stochastic process, that $\mathbf{\Psi}$ is row-consistent and that the same latent pair $(g, \mathbf{a})$ is reused for all query inputs. Then, for any coefficient law $\nu(\mathbf{a})$, the finite-dimensional distributions induced by*

$$f(\cdot) = g(\cdot) + \mathbf{\Psi}_{\mathbf{Z}}(\cdot)\big[\boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z})\big], \qquad g \sim p_0, \quad \mathbf{a} \sim \nu,$$

satisfy Kolmogorov consistency. Hence both the GMVIP *surrogate prior and variational posterior define valid stochastic processes.*

This result ensures that GMVIP is not merely a collection of unrelated finite-dimensional approximations. The same sampled path g and the same coefficient $\mathbf{a}$ define all evaluations of a function, so restrictions and permutations of finite query sets are compatible.

Proposition 2 (Coefficient KL reduction)**.** *Let $p_{\mathbf{Z}}$ and $q_{\mathbf{Z},\phi}$ be defined by Equations (6) and (7). Then*

$$\mathrm{KL}\left(q_{\mathbf{Z},\phi}(g, \mathbf{a}, f) \,\|\, p_{\mathbf{Z}}(g, \mathbf{a}, f)\right) = \mathrm{KL}\left(q_\phi(\mathbf{a}) \,\|\, p(\mathbf{a})\right). \tag{12}$$

Moreover, if $q^f_{\mathbf{Z},\phi}$ and $p^f_{\mathbf{Z}}$ denote the corresponding marginal laws over functions, then

$$\mathrm{KL}(q^f_{\mathbf{Z},\phi} \,\|\, p^f_{\mathbf{Z}}) \le \mathrm{KL}\left(q_\phi(\mathbf{a}) \,\|\, p(\mathbf{a})\right). \tag{13}$$

If $\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{Z}) = \mathbf{I}_{d_a}$ and $\mathbf{D}$ is invertible, the inequality in (13) is an equality.

Since $p_{\mathbf{Z}}$ and $q_{\mathbf{Z},\phi}$ share the same pathwise map $(g, \mathbf{a}) \mapsto f$, their augmented KL is exactly $\mathrm{KL}(q_\phi(\mathbf{a}) \| p(\mathbf{a}))$, avoiding measurement-set estimates of a functional KL. The marginal function-space KL is upper bounded by this quantity, and the bound is tight whenever $\mathbf{a}$ is recoverable from f, for example when $\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{Z}) = \mathbf{I}_{d_a}$ and $\mathbf{D}$ is invertible, which satisfies for the empirical operator.

Proposition 3 (GP and SVGP recovery)**.** *For a scalar-valued process, let $p_0 = \mathcal{GP}(\mu, \kappa)$, assume $\kappa(\mathbf{Z}, \mathbf{Z}) \succ 0$, and choose*

$$\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X}) = \kappa(\mathbf{X}, \mathbf{Z})\kappa(\mathbf{Z}, \mathbf{Z})^{-1}, \qquad \boldsymbol{\mu} = \mu(\mathbf{Z}), \qquad \mathbf{D}\mathbf{D}^\top = \kappa(\mathbf{Z}, \mathbf{Z}).$$

If $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$, then $T(g, \mathbf{a}) \sim \mathcal{GP}(\mu, \kappa)$. Moreover, for any coefficient law $q_\phi(\mathbf{a})$, with $\mathbf{u} = \boldsymbol{\mu} + \mathbf{D}\mathbf{a}$, the induced posterior is

$$q(f) = \int p_0(f|\mathbf{u})\, q_\phi(\mathbf{u})\, \mathrm{d}\mathbf{u}.$$

Thus GMVIP *recovers both the original GP prior and the standard inducing-variable GP family. In particular, any Gaussian SVGP posterior $q(\mathbf{u}) = \mathcal{N}(\mathbf{m}_u, \mathbf{S}_u)$ is recovered by taking $q_\phi(\mathbf{a}) = \mathcal{N}(\mathbf{D}^{-1}(\mathbf{m}_u - \boldsymbol{\mu}), \mathbf{D}^{-1}\mathbf{S}_u\mathbf{D}^{-\top})$.*

This proposition verifies that the construction is not an ad-hoc interpolation scheme: with the matching GP operator, GMVIP reduces exactly to classical Matheron sampling on sparse variational GPs.

Proposition 4 (Second-order prior preservation)**.** *For a scalar-valued process, assume $g \sim p_0$ has finite second moments, with mean μ and covariance function κ. If $\mathbf{\Psi}_{\mathbf{Z}}(\cdot) = \kappa(\cdot, \mathbf{Z})\kappa(\mathbf{Z}, \mathbf{Z})^{-1}$, $\boldsymbol{\mu} = \mu(\mathbf{Z})$, $\mathbf{D}\mathbf{D}^\top = \kappa(\mathbf{Z}, \mathbf{Z})$, and $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$ is independent of g, then the* GMVIP *surrogate prior has the same mean and covariance function as p_0.*

For a Gaussian prior, Proposition 3 gives equality of the full process law. For a general implicit prior, Proposition 4 shows what the empirical Matheron operator preserves the first two moments of the original implicit process when the prior is not Gaussian. Higher-order differences are controlled by the quality of the replacement operator and the choice of inducing set; Appendix A makes this dependence explicit. In these propositions, $\kappa(\mathbf{X}, \mathbf{Z})$ denotes the corresponding Gram matrix, whereas $\kappa(\mathbf{x}, \mathbf{z})$ denotes a scalar kernel value.

Table 1: UCI regression metrics (↓). Entries are mean ± standard deviation across five seeds. Best mean per dataset is shown in **purple**; second-best mean is shown in **teal**.

RMSE	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine Red	Yacht
MAP	4.34 ± 1.41	5.45 ± 0.57	0.52 ± 0.05	0.082 ± 0.001	4.04 ± 1.97	3.94 ± 0.13	4.62 ± 0.07	0.69 ± 0.01	0.52 ± 0.20
MFVI	3.93 ± 0.70	6.23 ± 0.51	0.62 ± 0.09	0.091 ± 0.002	147.61 ± 0.90	4.12 ± 0.17	4.94 ± 0.08	**0.62 ± 0.03**	0.68 ± 0.10
VIP	6.49 ± 1.27	5.76 ± 0.97	0.58 ± 0.09	0.080 ± 0.003	2.59 ± 0.96	**3.76 ± 0.07**	4.37 ± 0.06	0.79 ± 0.05	0.84 ± 0.30
FBNN	4.17 ± 1.27	**5.21 ± 0.52**	**0.51 ± 0.05**	0.086 ± 0.001	3.95 ± 0.47	4.05 ± 0.16	4.81 ± 0.05	0.65 ± 0.04	**0.23 ± 0.06**
SIP	**3.22 ± 0.99**	5.30 ± 0.27	0.52 ± 0.04	**0.071 ± 0.001**	**1.48 ± 0.23**	3.84 ± 0.11	4.62 ± 0.05	0.66 ± 0.03	0.41 ± 0.15
TFSVI	6.39 ± 0.63	5.97 ± 0.85	0.59 ± 0.15	0.081 ± 0.002	3.95 ± 0.81	3.90 ± 0.13	4.61 ± 0.05	0.73 ± 0.02	0.83 ± 0.38
FTIP	6.68 ± 1.58	6.02 ± 1.08	0.55 ± 0.07	0.079 ± 0.002	3.36 ± 1.58	**3.77 ± 0.11**	**4.36 ± 0.05**	0.82 ± 0.03	1.17 ± 0.52
GMVIP	**3.32 ± 0.97**	**4.24 ± 0.74**	**0.51 ± 0.05**	**0.069 ± 0.001**	**1.36 ± 0.19**	3.78 ± 0.12	**4.32 ± 0.04**	**0.60 ± 0.03**	**0.32 ± 0.11**
NLL	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine Red	Yacht
MAP	7.62 ± 3.48	3.24 ± 0.17	**0.81 ± 0.14**	-1.08 ± 0.02	-6.14 ± 1.03	2.79 ± 0.03	2.95 ± 0.01	1.27 ± 0.04	16.18 ± 15.06
MFVI	**2.76 ± 0.14**	3.25 ± 0.08	0.94 ± 0.14	-0.98 ± 0.03	-2.80 ± 0.01	2.84 ± 0.04	3.02 ± 0.02	**0.94 ± 0.04**	1.08 ± 0.05
VIP	4245.41 ± 2526.37	4.80 ± 1.11	2.48 ± 0.60	-1.09 ± 0.04	-6.78 ± 0.49	**2.74 ± 0.02**	2.89 ± 0.01	3.16 ± 0.39	96.61 ± 53.56
FBNN	5.24 ± 2.45	**3.13 ± 0.16**	0.84 ± 0.22	-1.04 ± 0.01	-6.42 ± 0.11	2.82 ± 0.04	2.99 ± 0.01	1.08 ± 0.08	**0.30 ± 0.84**
SIP	3.27 ± 1.22	3.18 ± 0.09	0.84 ± 0.14	**-1.22 ± 0.02**	**-7.04 ± 0.29**	2.76 ± 0.03	2.95 ± 0.01	1.09 ± 0.07	5.65 ± 5.09
TFSVI	11.32 ± 2.79	3.38 ± 0.23	1.05 ± 0.44	-1.09 ± 0.03	-6.41 ± 0.22	2.78 ± 0.03	2.95 ± 0.01	1.29 ± 0.07	14.30 ± 10.07
FTIP	1789.45 ± 735.75	4.82 ± 1.21	2.13 ± 0.36	-1.11 ± 0.02	-6.47 ± 0.59	**2.75 ± 0.03**	**2.89 ± 0.01**	3.52 ± 0.45	56.66 ± 56.15
GMVIP	**3.11 ± 1.03**	**2.82 ± 0.16**	**0.75 ± 0.09**	**-1.26 ± 0.02**	**-7.12 ± 0.04**	2.75 ± 0.03	**2.88 ± 0.01**	**0.94 ± 0.08**	**0.18 ± 0.22**

Computational complexity For scalar outputs, let N be the number of input locations at which the function is sampled, $M = |\mathbf{Z}|$ the number of inducing locations, and B the number of prior samples used to estimate the empirical covariance operator. Let $C_{\text{bank}}(N+M, B)$ denote the cost of evaluating these B prior samples on $\mathbf{X} \cup \mathbf{Z}$, and let $C_g(N+M)$ denote the cost of drawing and evaluating the coherent prior sample $g(\mathbf{X}), g(\mathbf{Z})$ used in the Matheron update. Constructing the empirical operator costs $O\left(C_{\text{bank}}(N+M,B) + BNM + BM^2 + NM^2 + M^3\right)$. The terms BNM and BM^2 form the empirical cross-covariance and inducing covariance from the prior bank, M^3 is the inducing Cholesky factorization, and NM^2 applies the inverse inducing covariance to the $N \times M$ cross-covariance matrix. After this setup is amortized, one posterior sample costs $O\left(C_g(N+M) + M^2 + NM\right)$, where M^2 accounts for a dense coefficient transformation and NM for the pathwise correction. For a joint C-output model, replace the matrix dimension M by $d_a = MC$ and let N count scalar output evaluations.

Remark 1 (Tunable inducing locations and operators)**.** *All identities above hold conditionally on* $\mathbf{Z}$*,* $\mathbf{\Psi}_{\mathbf{Z}}$*,* $\boldsymbol{\mu}$*, and* $\mathbf{D}$*. If these quantities are fixed or learned from prior samples only,* GMVIP *is variational inference under a fixed surrogate prior. If they are optimized using the observed targets, the surrogate prior itself becomes data-adaptive. This can be useful as empirical-Bayes approximation tuning, and is used in our experiments when suitable.*

5 Experiments

In this section we showcase the experimental results obtained in a range of regression and forecasting experiments. Refer to Appendix C for further experimental details and Appendix D.4 for ablation studies on inducing locations and prior samples. Additional classification LeNet experiments are included in Appendix D.2.

5.1 Synthetic function-space diagnostics

We adapt the one-dimensional synthetic regression benchmark of Izmailov et al. (2020) to the implicit-process setting. The fixed dataset is $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{400}$ with scalar inputs and outputs. The implicit process prior is induced by a BNN. Figure 4 (Appendix C.1) shows the predictive distributions produced by the different methods on the same input grid. The experiment tests whether each approximation can represent plausible uncertainty in regions with sparse observations and extrapolation.

5.2 Small and medium regression benchmarks

We evaluate regression methods on the standard UCI suite: Boston, Concrete, Energy, Kin8nm, Naval, Power, Protein, Wine Red, and Yacht; using a two-layer with 10 units BNN as prior. Each task is scalar regression with Gaussian likelihood, and uses random 90%/10% train/test splits. Table 1 reports RMSE and NLL averaged across random seeds (CRPS and CQM (Ortega et al., 2024) in Appendix D.3). GMVIP is highly competitive across the experiment: it obtains the best or second-best mean on several datasets and metrics, including strong RMSE/CRPS results on Boston, Concrete,

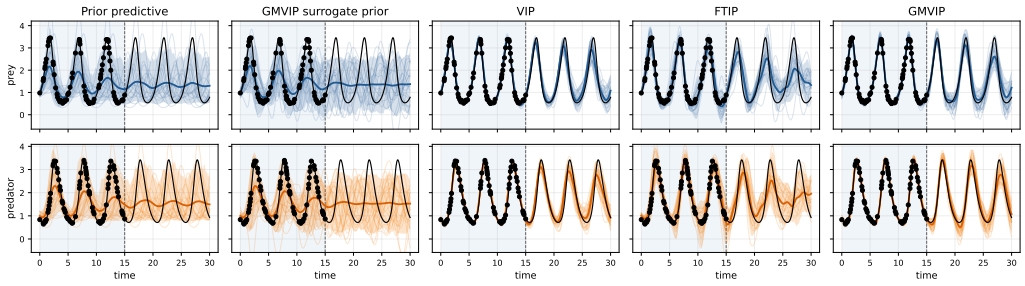


Figure 2: Lotka–Volterra forecasting with an implicit ODE prior. Black points are noisy observations. Curves and bands show posterior predictions from the final scheduled optimization iterate.

Table 2: Mean ± standard error Lotka–Volterra results on the held-out test interval $t \in (20, 30]$ across 20 target trajectories. VIP and FTIP use $S = 20$. VIP is trained for 800 iterations; FTIP uses a 400-iteration VIP warm start followed by 400 fine-tuning iterations. Best results are shown in **purple** and second-best results in **teal**. Coverage is ranked by proximity to the nominal 0.90 level.

Method	RMSE ↓	NLL ↓	CRPS ↓	Cov_{90}	ODE residual ↓
Prior predictive	0.63 ± 0.06	$\mathbf{0.90 \pm 0.11}$	0.36 ± 0.04	$\mathbf{0.90 \pm 0.03}$	$\mathbf{\approx 10^{-6}}$
GMVIP sur. prior	0.63 ± 0.06	1.95 ± 0.93	0.37 ± 0.03	0.93 ± 0.02	0.31 ± 0.00
VIP	$\mathbf{0.50 \pm 0.08}$	11.32 ± 5.01	$\mathbf{0.31 \pm 0.05}$	0.72 ± 0.06	0.65 ± 0.03
FTIP	0.57 ± 0.08	1.04 ± 0.22	0.34 ± 0.05	0.82 ± 0.04	0.62 ± 0.03
GMVIP	$\mathbf{0.32 \pm 0.06}$	$\mathbf{0.51 \pm 0.29}$	$\mathbf{0.20 \pm 0.04}$	$\mathbf{0.90 \pm 0.04}$	$\mathbf{0.31 \pm 0.02}$

Energy, Kin8nm, Naval, and Protein. For the BNN regression benchmarks, VIP, FTIP, and GMVIP jointly optimize the BNN prior parameters and therefore implement empirical-Bayes approximation tuning. Results with fixed prior in Appendix D.4.

GMVIP is particularly strong in RMSE, NLL, and CRPS, while centered interval calibration is mixed and sometimes favors simpler alternatives. This suggests that the Matheron posterior better captures predictive uncertainty while retaining the implicit-prior structure. On some datasets, notably Power and Yacht, simpler or alternative approximations remain competitive, indicating that no single posterior family dominates uniformly. Appendix D.4 shows that the residual path prevents severe overconfidence on several small-data tasks, although an inducing-only representation remains competitive on some datasets; it also quantifies the role of empirical-Bayes prior adaptation.

5.3 Forecasting With Implicit and Empirical Priors

We evaluate GMVIP in two complementary forecasting settings with sample-defined priors: a simulator-induced Lotka–Volterra prior and a retrieval-conditioned empirical prior for electricity load. Full protocols are provided in Appendices C.3 and C.4. The simulator prior remain fixed, and the electricity prior is fixed after target-specific retrieval.

Lotka–Volterra. The implicit prior is induced by the predator–prey system

$$\dot{u} = \alpha u - \beta uv, \qquad \dot{v} = \delta uv - \gamma v,$$

by sampling physical parameters and initial conditions and solving the ODE. Each method observes noisy values for $t \leq 15$, receives no data from the gap $15 < t \leq 20$, and is evaluated only on the forecast interval $20 < t \leq 30$. GMVIP uses the joint multioutput construction.

Figure 2 qualitatively illustrates the differences between the methods. The prior predictive and its GMVIP surrogate capture the oscillatory structure of the ODE prior but are not adapted to the observed trajectory. VIP and FTIP condition on the observations, although their forecasts exhibit increasing phase and amplitude errors over the held-out interval. In contrast, GMVIP more consistently preserves the coupled oscillatory dynamics of both prey and predator over the forecasting horizon.

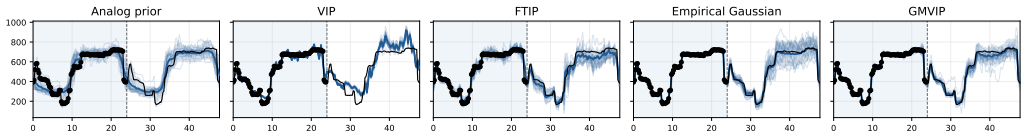


Figure 3: Electricity-load forecasting with a retrieval-conditioned empirical prior. Methods observe the first 24 hours and forecast the following 24 hours. The panels show VIP, FTIP, and GMVIP on the same target window. Black points are observations, the black curve is the held-out target, and blue curves and bands show posterior predictions.

Table 3: Electricity-load results across 75 seed–target windows. Entries are means ± standard errors across windows. Coverage and its standard error are reported in percentage points. Best results are shown in **purple** and second-best results in **teal**. Coverage is ranked by proximity to its nominal level; lower is better otherwise.

Method	RMSE ↓	NLL ↓	CRPS ↓	CQM ↓	Cov. 80%	Cov. 90%
Analog prior	55.45 ± 15.91	$\mathbf{5.31 \pm 0.44}$	30.73 ± 9.04	$\mathbf{0.12 \pm 0.01}$	74.18 ± 2.14	85.96 ± 1.70
VIP	60.93 ± 18.61	99.19 ± 55.22	40.60 ± 12.83	0.38 ± 0.01	20.94 ± 1.16	26.26 ± 1.42
FTIP	68.02 ± 22.10	10.03 ± 4.17	41.28 ± 13.99	0.17 ± 0.01	57.25 ± 2.03	68.06 ± 1.99
Empirical Gaussian	$\mathbf{48.44 \pm 15.27}$	6.63 ± 0.99	$\mathbf{28.37 \pm 9.06}$	0.13 ± 0.01	$\mathbf{79.32 \pm 2.16}$	$\mathbf{86.62 \pm 1.78}$
GMVIP	$\mathbf{49.06 \pm 15.06}$	$\mathbf{4.35 \pm 0.14}$	$\mathbf{27.48 \pm 8.74}$	$\mathbf{0.11 \pm 0.01}$	$\mathbf{76.22 \pm 2.11}$	$\mathbf{86.53 \pm 1.73}$

Quantitatively, Table 2 shows that GMVIP achieves the lowest mean RMSE, NLL, and CRPS at $S = 20$, while the prior predictive has coverage closest to 0.90. GMVIP's mean RMSE of 0.32 represents reductions of 35.5% relative to VIP and 43.6% relative to FTIP. The basis-size analysis in Appendix D.5 shows that increasing S improves VIP and FTIP's point forecasts but does not overtake GMVIP on the aggregate metrics. GMVIP has the lowest mean RMSE, NLL, CRPS, and ODE residual among the learned configurations, as well as coverage closest to 0.90.

Electricity load. We next consider day-ahead forecasting on the Electricity Load dataset (Trindade, 2015). Each target contains 48 hours at 15-minute resolution: the first 24 hours are observed and the next 24 hours are held out. For each target, we retrieve 2048 historical trajectories from 2011–2013 using calendar compatibility and nearest-neighbor similarity over the observed context; test targets are drawn from 2014. Because the observed prefix is used both to retrieve the historical bank and to fit the posterior, this experiment is a two-stage empirical-Bayes procedure rather than inference under a fixed, target-independent prior. Retrieval never accesses the held-out forecast day, and every method receives the same retrieved bank. GMVIP uses $M = 96$ inducing locations, with $\mathbf{Z} = \mathbf{X}_{\text{obs}}$: one at each observed context point and none in the forecast interval.

Figure 3 provides a representative qualitative comparison. VIP and FTIP follow the broad daily pattern but show larger forecast spread and noticeable errors around the load minimum and subsequent recovery. GMVIP more closely reproduces the sharp decline after the observed context and the later return to the higher-load regime, while maintaining coherent uncertainty across the forecast horizon.

Quantitatively, Table 3 reports means and sample standard errors across the 75 windows; the large standard errors show substantial target heterogeneity. The Empirical Gaussian has the lowest mean RMSE, with GMVIP close behind (48.44 versus 49.06). GMVIP obtains the lowest mean mixture NLL, CRPS, and CQM. The Empirical Gaussian has coverage closest to the nominal 80% and 90% levels, with GMVIP second on both. VIP is strongly under-dispersed, whereas FTIP recovers part of the missing uncertainty. Overall, GMVIP provides the strongest distributional scores, while analytic Gaussian conditioning has a marginal advantage in mean point error.

5.4 Large Scale Regression

We evaluate on the YearPredictionMSD (Bertin-Mahieux, 2011) and Airline benchmark derived from the U.S. Bureau of Transportation Statistics on-time performance data (Bureau of Transportation Statistics, 2026) and used in prior large-scale GP work (Dutordoir et al., 2020). These tasks were also considered by Ortega et al. (2024), but we use our own shared protocol. All methods use the

Table 4: Large regression metrics (↓). Entries are mean ± standard deviation across seeds. Best method is shown in **purple**; second-best is shown in **teal**.

Metric	Dataset	MFVI	VIP	FBNN	SIP	TFSVI	FTIP	GMVIP
RMSE	Year	**8.95 ± 0.01**	9.17 ± 0.01	**8.91 ± 0.02**	9.06 ± 0.01	9.05 ± 0.01	9.15 ± 0.01	8.98 ± 0.02
	Airline	37.87 ± 0.12	38.03 ± 0.16	37.90 ± 0.09	37.99 ± 0.07	**37.72 ± 0.13**	38.04 ± 0.12	**37.62 ± 0.10**
NLL	Year	**3.61 ± 0.00**	3.64 ± 0.00	**3.60 ± 0.00**	3.62 ± 0.00	3.62 ± 0.00	3.63 ± 0.00	**3.61 ± 0.00**
	Airline	**5.07 ± 0.00**	5.09 ± 0.01	**5.07 ± 0.00**	**5.08 ± 0.00**	5.09 ± 0.00	5.09 ± 0.01	**5.07 ± 0.01**
CRPS	Year	**4.717 ± 0.007**	4.859 ± 0.010	**4.675 ± 0.012**	4.785 ± 0.010	4.777 ± 0.012	4.845 ± 0.004	4.740 ± 0.014
	Airline	17.67 ± 0.13	17.79 ± 0.22	17.70 ± 0.17	17.71 ± 0.08	**17.47 ± 0.13**	17.79 ± 0.17	**17.61 ± 0.04**
CQM	Year	0.083 ± 0.002	**0.064 ± 0.002**	0.084 ± 0.003	**0.078 ± 0.001**	**0.078 ± 0.003**	**0.064 ± 0.002**	**0.078 ± 0.003**
	Airline	0.107 ± 0.009	**0.096 ± 0.015**	0.109 ± 0.014	0.106 ± 0.003	0.111 ± 0.010	**0.096 ± 0.012**	**0.102 ± 0.004**

same preprocessing and fixed train/test splits, with inputs standardized using training-set statistics and targets normalized during training. We train for a matched budget of 60 000 stochastic iterations using two-hidden-layer MLPs with 50 units for Year and 100 for Airlin.

Table 4 shows that no single method dominates all metrics. On YearPredictionMSD, FBNN gives the best predictive accuracy and distributional scores, with the lowest RMSE, NLL, and CRPS. GMVIP remains competitive, ranking close to the best methods in RMSE and CRPS, but does not improve over FBNN on this dataset. On Airline, the picture changes: GMVIP obtains the best RMSE and ties for the best rounded NLL, while also achieving the second-best CRPS. This indicates that the empirical Matheron correction is particularly effective in the larger Airline setting, where adapting function-space prior samples through learned inducing corrections improves predictive accuracy.

Calibration, as measured by CQM, favors VIP and FTIP on both datasets. These methods obtain the lowest CQM values, while GMVIP is second-best on Airline and tied for second-best on Year. Overall, the results suggest a tradeoff: VIP/FTIP provide the strongest centered interval calibration, FBNN is strongest on Year, and GMVIP gives the best or near-best predictive performance on the larger Airline benchmark while retaining competitive calibration.

6 Discussion and Conclusion

We introduced GMVIP, a pathwise variational family for implicit-process priors. Instead of forming posterior samples as linear combinations of prior functions, GMVIP draws a coherent prior path and replaces its values at inducing locations through an empirical Matheron-style correction. Since the surrogate prior and posterior share the same pathwise map and differ only in the law of the inducing coefficient, the augmented KL reduces to a finite-dimensional coefficient KL.

The construction recovers the standard inducing-variable variational GP family when the prior is Gaussian and the matching covariance operator is used. For general implicit priors, empirical GMVIP should be understood as inference under a Matheron surrogate prior: it preserves the prior mean and covariance in the population limit, but not necessarily the full process law. Empirically, GMVIP is competitive across regression and classification. In Lotka-Volterra, GMVIP has the lowest mean RMSE and CRPS and the lowest mean mixture NLL among the learned methods in the matched $S = 20$ comparison. It also retains the best mean RMSE, NLL, CRPS, calibration, and ODE residual when compared with the larger VIP and FTIP bases. These comparisons vary substantially by target, and mean NLL remains sensitive to one difficult high-amplitude trajectory. Thus the simulator results support the value of coherent prior residual paths while showing that larger coefficient bases narrow the point-forecasting gap. In electricity, GMVIP has the lowest mean NLL, CRPS, and CQM, while the Empirical Gaussian has the lowest mean RMSE and coverage closest to nominal.

Limitations GMVIP performs inference under the surrogate prior induced by the replacement map, which may differ from the original implicit-process prior for non-Gaussian or highly structured processes. The empirical operator also depends on the quality of the prior bank and on the conditioning of the inducing covariance, so poor inducing locations or too few prior samples can degrade performance. Finally, empirical GMVIP is more expensive than coefficient-only VIP-style methods because it requires prior-bank evaluations, covariance estimation, and a Matheron correction. The present experiments use a Gaussian posterior over inducing coefficients; richer posterior families may be needed for strongly multimodal posteriors.

AI USE STATEMENT

In this work, we used generative AI tools to provide feedback on research methodology and experimental design and to assist with the interpretation and presentation of experimental results. We did not use generative AI tools to generate datasets, formulate mathematical claims, develop theoretical models, provide critical ingredients for or assist in writing proofs, propose hypotheses, implement the proposed methods, clean or reformat datasets, or perform qualitative or thematic data analysis. Translation and research tasks involving surveys or interviews were not applicable to this work.

Additionally, we used generative AI tools for brainstorming, searching for and summarizing potentially relevant literature, suggesting improvements to the structure and presentation of the paper, and drafting or editing text for clarity and readability. All methodological suggestions and interpretations produced with AI assistance were independently evaluated by the authors. All references were manually verified against their original sources, all experimental results and numerical claims were checked against the outputs of experiments conducted by the authors, and all AI-assisted text was reviewed and revised by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide the details required to reproduce both the theoretical and empirical results of this work. Complete proofs of the main theoretical claims, together with additional properties of the proposed construction, are given in Appendix A. The GMVIP training and evaluation procedures are specified in Algorithms 1 and 2 in Appendix B. Experimental protocols are provided in Appendix C, including the UCI regression experiments (Appendix C.2), the Lotka–Volterra implicit-prior experiment (Appendix C.3), the electricity-load forecasting experiment (Appendix C.4), the large-scale regression experiments (Appendix C.5), and the LeNet classification experiments (Appendix C.6). These sections specify dataset preprocessing and splits, model architectures, optimization settings, random seeds, Monte Carlo sample counts, inducing-point configurations, prior-bank sizes, likelihoods, and evaluation metrics. Additional reproducibility analyses are reported for surrogate-prior fidelity in Appendix D.1, classification experiments in Appendix D.2, full UCI results in Appendix D.3, ablation studies in Appendix D.4, and Lotka–Volterra basis-size sensitivity in Appendix D.5. All reported predictive results are evaluated on held-out data according to the protocols specified in these sections.

REFERENCES

Thierry Bertin-Mahieux. Year prediction MSD. UCI Machine Learning Repository, 2011. URL https://doi.org/10.24432/C50K61. Dataset.

Charles Blundell, Julien Cornebise, Koray Kavukcuoglu, and Daan Wierstra. Weight uncertainty in neural network. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1613–1622. PMLR, 2015. URL https://proceedings.mlr.press/v37/blundell15.html.

Bureau of Transportation Statistics. Reporting carrier on-time performance (1987–present). TranStats, U.S. Department of Transportation, 2026. URL https://www.transtats.bts.gov/Fields.asp?gnoyr_VQ=FGJ. Dataset; accessed 2026-07-23.

Gerrit Burgers, Peter Jan van Leeuwen, and Geir Evensen. Analysis scheme in the ensemble Kalman filter. *Monthly Weather Review*, 126(6):1719–1724, 1998. doi: 10.1175/1520-0493(1998)126⟨1719:ASITEK⟩2.0.CO;2. URL https://journals.ametsoc.org/view/journals/mwre/126/6/1520-0493_1998_126_1719_asitek_2_0_co_2.xml.

David R. Burt, Sebastian W. Ober, Adrià Garriga-Alonso, and Mark van der Wilk. Understanding variational inference in function-space. In *Third Symposium on Advances in Approximate Bayesian Inference*, pages 1–17, 2020. URL https://openreview.net/forum?id=7P9y3sRa5Mk.

Vincent Dutordoir, Nicolas Durrande, and James Hensman. Sparse Gaussian processes with spherical harmonic features. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 2793–2802. PMLR, 2020. URL https://proceedings.mlr.press/v119/dutordoir20a.html.

Geir Evensen. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans*, 99(C5): 10143–10162, 1994. doi: 10.1029/94JC00572. URL https://agupubs.onlinelibrary.wiley.com/doi/10.1029/94JC00572.

Geir Evensen and Peter Jan van Leeuwen. An ensemble Kalman smoother for nonlinear dynamics. *Monthly Weather Review*, 128(6):1852–1867, 2000. doi: 10.1175/1520-0493(2000)128⟨1852:AEKSFN⟩2.0.CO;2. URL https://journals.ametsoc.org/view/journals/mwre/128/6/1520-0493_2000_128_1852_aeksfn_2_0_co_2.xml.

Yarin Gal and Zoubin Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059. PMLR, 2016. URL https://proceedings.mlr.press/v48/gal16.html.

James Hensman, Nicolò Fusi, and Neil D. Lawrence. Gaussian processes for big data. In Ann Nicholson and Padhraic Smyth, editors, *Uncertainty in Artificial Intelligence: Proceedings of the Twenty-Ninth Conference*, pages 282–290, Corvallis, Oregon, 2013. AUAI Press. ISBN 978-0-9749039-9-6. URL https://auai.org/uai2013/prints/papers/244.pdf.

José Miguel Hernández-Lobato and Ryan P. Adams. Probabilistic backpropagation for scalable learning of Bayesian neural networks. In Francis Bach and David Blei, editors, *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1861–1869. PMLR, 2015. URL https://proceedings.mlr.press/v37/hernandez-lobatoc15.html.

Pavel Izmailov, Wesley J. Maddox, Polina Kirichenko, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Subspace inference for Bayesian deep learning. In Ryan P. Adams and Vibhav Gogate, editors, *Proceedings of the 35th Uncertainty in Artificial Intelligence Conference*, volume 115 of *Proceedings of Machine Learning Research*, pages 1169–1179. PMLR, 2020. URL https://proceedings.mlr.press/v115/izmailov20a.html.

A. C. Lorenc. A global three-dimensional multivariate statistical interpolation scheme. *Monthly Weather Review*, 109(4):701–721, 1981. doi: 10.1175/1520-0493(1981)109⟨0701:AGTDMS⟩2.0.CO;2. URL https://journals.ametsoc.org/view/journals/mwre/109/4/1520-0493_1981_109_0701_agtdms_2_0_co_2.xml.

Chao Ma, Yingzhen Li, and José Miguel Hernández-Lobato. Variational implicit processes. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 4222–4233. PMLR, 2019. URL https://proceedings.mlr.press/v97/ma19b.html.

David J. C. MacKay. A practical Bayesian framework for backpropagation networks. *Neural Computation*, 4(3):448–472, 1992. doi: 10.1162/neco.1992.4.3.448. URL https://doi.org/10.1162/neco.1992.4.3.448.

Radford M. Neal. *Bayesian Learning for Neural Networks*, volume 118 of *Lecture Notes in Statistics*. Springer, New York, NY, 1996. ISBN 978-0-387-94724-2. doi: 10.1007/978-1-4612-0745-0. URL https://doi.org/10.1007/978-1-4612-0745-0.

Luis A. Ortega. *Uncertainty Estimation and Generalization Bounds for Modern Deep Learning*. PhD thesis, Universidad Autónoma de Madrid, 2026. URL https://arxiv.org/abs/2606.13818.

Luis A. Ortega, Simón Rodríguez Santana, and Daniel Hernández-Lobato. Deep variational implicit processes. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=8aeSJNbmbQq.

Luis A. Ortega, Simon Rodriguez Santana, and Daniel Hernández-Lobato. Variational linearized Laplace approximation for Bayesian deep learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 38815–38836. PMLR, 2024. URL https://proceedings.mlr.press/v235/ortega24a.html.

Luis A. Ortega, Andrés R. Masegosa, and Thomas D. Nielsen. Flow-transformed implicit processes for function-space variational inference, 2026. URL https://arxiv.org/abs/2606.01954. Preprint; submitted for revision.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation and Machine Learning. The MIT Press, Cambridge, Massachusetts, 2006. ISBN 978-0-262-18253-9. doi: 10.7551/mitpress/3206.001.0001. URL https://gaussianprocess.org/gpml/.

Simón Rodríguez-Santana, Bryan Zaldívar, and Daniel Hernández-Lobato. Function-space inference with sparse implicit processes. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 18723–18740. PMLR, 2022. URL https://proceedings.mlr.press/v162/rodri-guez-santana22a.html.

Tim G. J. Rudner, Zonghao Chen, Yee Whye Teh, and Yarin Gal. Tractable function-space variational inference in Bayesian neural networks. In *Advances in Neural Information Processing Systems*, volume 35, pages 22686–22698. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/8ea50bf458f6070548b11babbe0bf89b-Abstract-Conference.html.

Shengyang Sun, Guodong Zhang, Jiaxin Shi, and Roger Grosse. Functional variational Bayesian neural networks. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rkxacs0qY7.

Michalis K. Titsias. Variational learning of inducing variables in sparse Gaussian processes. In David van Dyk and Max Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574. PMLR, 2009. URL https://proceedings.mlr.press/v5/titsias09a.html.

Artur Trindade. ElectricityLoadDiagrams20112014. UCI Machine Learning Repository, 2015. URL https://doi.org/10.24432/C58C86. Dataset.

Veit David Wild, Robert Hu, and Dino Sejdinovic. Generalized variational inference in function spaces: Gaussian measures meet Bayesian deep learning. In *Advances in Neural Information Processing Systems*, volume 35, pages 3716–3730. Curran Associates, Inc., 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/18210aa6209b9adfc97b8c17c3741d95-Abstract-Conference.html.

Andrew Gordon Wilson and Hannes Nickisch. Kernel interpolation for scalable structured Gaussian processes (KISS-GP). In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 1775–1784. PMLR, 2015. URL https://proceedings.mlr.press/v37/wilson15.html.

James T. Wilson, Viacheslav Borovitskiy, Alexander Terenin, Peter Mostowsky, and Marc Peter Deisenroth. Efficiently sampling functions from Gaussian process posteriors. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 10292–10302. PMLR, 2020. URL https://proceedings.mlr.press/v119/wilson20a.html.

A MATHEMATICAL DERIVATIONS

This appendix contains the proofs of the main results in Section 4, together with additional properties of the replacement operator used in GMVIP.

A.1 PROOFS OF THE MAIN RESULTS

Proof of Proposition 1. Let $\mathbf{X} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ be any finite ordered tuple. The induced finite-dimensional random vector is

$$f(\mathbf{X}) = g(\mathbf{X}) + \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X})\big[\boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z})\big].$$

It is a measurable function of $(g(\mathbf{X}), g(\mathbf{Z}), \mathbf{a})$, and therefore defines a probability law.

It remains to verify projective consistency. Let $\mathbf{X}_1$ be obtained from $\mathbf{X}_2$ by a selection or permutation matrix $\mathbf{\Pi}$. For multioutput functions, the same symbol denotes the block operator $\mathbf{\Pi} \otimes \mathbf{I}_C$ acting on stacked evaluations. The replacement operators used in this work are row-consistent, meaning $\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X}_1) = \mathbf{\Pi}\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X}_2)$. Hence

$$\begin{aligned} f(\mathbf{X}_1) &= g(\mathbf{X}_1) + \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X}_1)\big[\boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z})\big] \\ &= \mathbf{\Pi} g(\mathbf{X}_2) + \mathbf{\Pi}\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X}_2)\big[\boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z})\big] = \mathbf{\Pi} f(\mathbf{X}_2). \end{aligned}$$

Since $g \sim p_0$ is a valid stochastic process, its finite-dimensional laws are Kolmogorov-consistent. Because the same global pair $(g, \mathbf{a})$ is reused for all query sets, the induced laws satisfy $P^{\nu}_{\mathbf{X}_1} = \mathbf{\Pi}_{\#} P^{\nu}_{\mathbf{X}_2}$, where $\mathbf{\Pi}_{\#}$ denotes pushforward by the restriction/permutation map. Thus the family $\{P^{\nu}_{\mathbf{X}}\}_{\mathbf{X}}$ satisfies Kolmogorov consistency. By the Kolmogorov extension theorem, it defines a valid stochastic process. Taking $\nu = p(\mathbf{a})$ gives the surrogate prior and taking $\nu = q_\phi(\mathbf{a})$ gives the variational posterior. □

Proof of Proposition 2. By construction,

$$p_{\mathbf{Z}}(g, \mathbf{a}, f) = p_0(g)p(\mathbf{a})\delta[f = T(g, \mathbf{a})], \qquad q_{\mathbf{Z},\phi}(g, \mathbf{a}, f) = p_0(g)q_\phi(\mathbf{a})\delta[f = T(g, \mathbf{a})].$$

Thus the two measures have the same conditional law of (g, f) given $\mathbf{a}$. Applying the chain rule for KL divergences,

$$\begin{aligned} \mathrm{KL}(q_{\mathbf{Z},\phi}(g, \mathbf{a}, f) \| p_{\mathbf{Z}}(g, \mathbf{a}, f)) &= \mathrm{KL}(q_\phi(\mathbf{a}) \| p(\mathbf{a})) \\ &\quad + \mathbb{E}_{q_\phi(\mathbf{a})} \mathrm{KL}(q_{\mathbf{Z},\phi}(g, f \mid \mathbf{a}) \| p_{\mathbf{Z}}(g, f \mid \mathbf{a})). \end{aligned}$$

The second term is zero because both conditionals equal $p_0(g)\delta[f = T(g, \mathbf{a})]$. This proves (12).

The marginal function-space bound follows by data processing under the projection $(g, \mathbf{a}, f) \mapsto f$:

$$\mathrm{KL}(q^f_{\mathbf{Z},\phi} \| p^f_{\mathbf{Z}}) \le \mathrm{KL}(q_{\mathbf{Z},\phi}(g, \mathbf{a}, f) \| p_{\mathbf{Z}}(g, \mathbf{a}, f)) = \mathrm{KL}(q_\phi(\mathbf{a}) \| p(\mathbf{a})).$$

Finally, if $\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{Z}) = \mathbf{I}_{d_a}$ and $\mathbf{D}$ is invertible, then evaluating the replacement map at the inducing inputs gives

$$f(\mathbf{Z}) = g(\mathbf{Z}) + \boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z}) = \boldsymbol{\mu} + \mathbf{D}\mathbf{a},$$

so $\mathbf{a} = \mathbf{D}^{-1}(f(\mathbf{Z}) - \boldsymbol{\mu})$ is a measurable function of f. Hence the density ratio between the augmented posterior and prior depends only on f, and the projection to f loses no KL information. Therefore the above inequality is an equality in this case. □

Proposition 2 applies exactly to the unregularized cardinal operator. In the numerical implementation, diagonal jitter makes cardinality and the corresponding marginal-KL equality approximate up to numerical regularization.

Proof of Proposition 3. Fix a finite input set $\mathbf{X}$. The GP conditional decomposition gives

$$g(\mathbf{X}) = \mu(\mathbf{X}) + \kappa(\mathbf{X}, \mathbf{Z})\kappa(\mathbf{Z}, \mathbf{Z})^{-1}\big(g(\mathbf{Z}) - \boldsymbol{\mu}\big) + \mathbf{r}_{\mathbf{X}},$$

where $\mathbf{r}_{\mathbf{X}}$ is independent of $g(\mathbf{Z})$ and has covariance

$$\kappa(\mathbf{X}, \mathbf{X}) - \kappa(\mathbf{X}, \mathbf{Z})\kappa(\mathbf{Z}, \mathbf{Z})^{-1}\kappa(\mathbf{Z}, \mathbf{X}).$$

With $\mathbf{u} = \boldsymbol{\mu} + \mathbf{D}\mathbf{a}$, the transformed sample is

$$T(g, \mathbf{a})(\mathbf{X}) = \mu(\mathbf{X}) + \kappa(\mathbf{X}, \mathbf{Z})\kappa(\mathbf{Z}, \mathbf{Z})^{-1}\big(\mathbf{u} - \boldsymbol{\mu}\big) + \mathbf{r}_{\mathbf{X}}.$$

If $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$, then $\mathbf{u} \sim \mathcal{N}(\boldsymbol{\mu}, \kappa(\mathbf{Z}, \mathbf{Z}))$, independently of $\mathbf{r}_{\mathbf{X}}$. Therefore $T(g, \mathbf{a})(\mathbf{X})$ has mean $\mu(\mathbf{X})$ and covariance $\kappa(\mathbf{X}, \mathbf{X})$. Since this holds for every finite $\mathbf{X}$, the transformed process is exactly $\mathcal{GP}(\mu, \kappa)$.

For arbitrary $q_\phi(\mathbf{a})$, the same expression is the GP conditional $p_0(f(\mathbf{X}) \mid \mathbf{u})$. Integrating over the pushforward law $q_\phi(\mathbf{u})$ gives $q(f) = \int p_0(f \mid \mathbf{u}) q_\phi(\mathbf{u}) d\mathbf{u}$. Finally, the correspondence with a Gaussian SVGP posterior $q(\mathbf{u}) = \mathcal{N}(\mathbf{m}_u, \mathbf{S}_u)$ follows from the affine whitening relation $\mathbf{u} = \boldsymbol{\mu} + \mathbf{D}\mathbf{a}$, which gives $q_\phi(\mathbf{a}) = \mathcal{N}(\mathbf{D}^{-1}(\mathbf{m}_u - \boldsymbol{\mu}), \mathbf{D}^{-1}\mathbf{S}_u\mathbf{D}^{-\top})$. □

Proof of Proposition 4. Fix finite input sets $\mathbf{X}$ and $\mathbf{X}'$, and write $\mathbf{B}_{\mathbf{X}} = \kappa(\mathbf{X}, \mathbf{Z})\kappa(\mathbf{Z}, \mathbf{Z})^{-1}$. The transformed sample can be written as

$$f(\mathbf{X}) = g(\mathbf{X}) - \mathbf{B}_{\mathbf{X}} g(\mathbf{Z}) + \mathbf{B}_{\mathbf{X}}\mathbf{u}, \qquad \mathbf{u} = \boldsymbol{\mu} + \mathbf{D}\mathbf{a}.$$

Since $\boldsymbol{\mu} = \mu(\mathbf{Z})$, $\mathbf{D}\mathbf{D}^\top = \kappa(\mathbf{Z}, \mathbf{Z})$, and $\mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M)$, we have $\mathbb{E}[\mathbf{u}] = \mu(\mathbf{Z})$ and $\mathrm{Cov}(\mathbf{u}) = \kappa(\mathbf{Z}, \mathbf{Z})$. Moreover, $\mathbf{u}$ is independent of g. Hence

$$\mathbb{E}[f(\mathbf{X})] = \mu(\mathbf{X}) - \mathbf{B}_{\mathbf{X}}\mu(\mathbf{Z}) + \mathbf{B}_{\mathbf{X}}\mu(\mathbf{Z}) = \mu(\mathbf{X}).$$

For the covariance, independence gives

$$\begin{aligned}\mathrm{Cov}(f(\mathbf{X}), f(\mathbf{X}')) = \mathrm{Cov}(g(\mathbf{X}) - \mathbf{B}_{\mathbf{X}} g(\mathbf{Z}),\ g(\mathbf{X}') - \mathbf{B}_{\mathbf{X}'} g(\mathbf{Z})) \\ + \mathbf{B}_{\mathbf{X}}\kappa(\mathbf{Z}, \mathbf{Z})\mathbf{B}_{\mathbf{X}'}^\top.\end{aligned}$$

Expanding the first term and using $\mathbf{B}_{\mathbf{X}} = \kappa(\mathbf{X}, \mathbf{Z})\kappa(\mathbf{Z}, \mathbf{Z})^{-1}$ yields

$$\mathrm{Cov}(g(\mathbf{X}) - \mathbf{B}_{\mathbf{X}} g(\mathbf{Z}),\ g(\mathbf{X}') - \mathbf{B}_{\mathbf{X}'} g(\mathbf{Z})) = \kappa(\mathbf{X}, \mathbf{X}') - \mathbf{B}_{\mathbf{X}}\kappa(\mathbf{Z}, \mathbf{X}').$$

The second term satisfies $\mathbf{B}_{\mathbf{X}}\kappa(\mathbf{Z}, \mathbf{Z})\mathbf{B}_{\mathbf{X}'}^\top = \mathbf{B}_{\mathbf{X}}\kappa(\mathbf{Z}, \mathbf{X}')$. Therefore $\mathrm{Cov}(f(\mathbf{X}), f(\mathbf{X}')) = \kappa(\mathbf{X}, \mathbf{X}')$. Since this holds for arbitrary finite $\mathbf{X}, \mathbf{X}'$, the surrogate prior has the same mean and covariance function as p_0. □

A.2 Additional theoretical properties

Proposition 5 (Finite-dimensional prior distortion)**.** *Fix a finite input set $\mathbf{X}$, write $\mathbf{B}_{\mathbf{X}} = \boldsymbol{\Psi}_{\mathbf{Z}}(\mathbf{X})$, and define the prior residual $\mathbf{r}_{\mathbf{X}} = g(\mathbf{X}) - \mathbf{B}_{\mathbf{X}} g(\mathbf{Z})$. Let the surrogate inducing variable $\mathbf{u}$ be independent of g, with law $P_{\mathbf{u}}$. Then, whenever the KLs are well-defined,*

$$\mathrm{KL}\left(p_0(g(\mathbf{X})) \,\|\, p_{\mathbf{Z}}(f(\mathbf{X}))\right) \le I(\mathbf{r}_{\mathbf{X}}; g(\mathbf{Z})) + \mathrm{KL}\left(p_0(g(\mathbf{Z})) \,\|\, P_{\mathbf{u}}\right).$$

In particular, if $\mathbf{u} \overset{d}{=} g(\mathbf{Z})$, the distortion is bounded by $I(\mathbf{r}_{\mathbf{X}}; g(\mathbf{Z}))$, and it is zero whenever $\mathbf{r}_{\mathbf{X}}$ is independent of $g(\mathbf{Z})$.

Proof. Let $H(\mathbf{r}, \mathbf{w}) = \mathbf{r} + \mathbf{B}_{\mathbf{X}}\mathbf{w}$. Under the original prior, $g(\mathbf{X}) = H(\mathbf{r}_{\mathbf{X}}, g(\mathbf{Z}))$, where $(\mathbf{r}_{\mathbf{X}}, g(\mathbf{Z}))$ has its joint prior law. Under the surrogate prior, $f(\mathbf{X}) = H(\mathbf{r}_{\mathbf{X}}, \mathbf{u})$, where the joint law is $P_{\mathbf{r}_{\mathbf{X}}} \otimes P_{\mathbf{u}}$, because $\mathbf{u}$ is independent of g. By data processing for KL divergences,

$$\mathrm{KL}(p_0(g(\mathbf{X}))\|p_{\mathbf{Z}}(f(\mathbf{X}))) \le \mathrm{KL}(P_{\mathbf{r}_{\mathbf{X}}, g(\mathbf{Z})}\|P_{\mathbf{r}_{\mathbf{X}}} \otimes P_{\mathbf{u}}).$$

The right-hand side decomposes as

$$\mathrm{KL}(P_{\mathbf{r}_{\mathbf{X}}, g(\mathbf{Z})}\|P_{\mathbf{r}_{\mathbf{X}}} \otimes P_{g(\mathbf{Z})}) + \mathrm{KL}(P_{g(\mathbf{Z})}\|P_{\mathbf{u}}) = I(\mathbf{r}_{\mathbf{X}}; g(\mathbf{Z})) + \mathrm{KL}(p_0(g(\mathbf{Z}))\|P_{\mathbf{u}}).$$

If $P_{\mathbf{u}} = P_{g(\mathbf{Z})}$, the second term vanishes. If, in addition, $\mathbf{r}_{\mathbf{X}}$ and $g(\mathbf{Z})$ are independent, the mutual information term also vanishes. □

The finite-dimensional KL distortion bound may be vacuous for singular or empirical implicit priors; guarantees based on Wasserstein or other integral probability metrics are an important direction for future work.

Proposition 6 (Prior reconstruction error)**.** *Assume $g \sim p_0$ has mean μ, covariance function κ, and $\kappa(\mathbf{Z},\mathbf{Z}) \succ 0$. With $\mathbf{\Psi}_{\mathbf{Z}}(\mathbf{U}) = \kappa(\mathbf{U},\mathbf{Z})\kappa(\mathbf{Z},\mathbf{Z})^{-1}$, the mean-squared residual reconstruction error is*

$$\mathbb{E}\left[\left\|g(\mathbf{U}) - \mu(\mathbf{U}) - \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{U})\big(g(\mathbf{Z}) - \mu(\mathbf{Z})\big)\right\|_2^2\right] = \operatorname{tr}\left(\kappa(\mathbf{U},\mathbf{U}) - \kappa(\mathbf{U},\mathbf{Z})\kappa(\mathbf{Z},\mathbf{Z})^{-1}\kappa(\mathbf{Z},\mathbf{U})\right).$$

Proof. Let $\tilde{g}(\mathbf{U}) = g(\mathbf{U}) - \mu(\mathbf{U})$, $\tilde{g}(\mathbf{Z}) = g(\mathbf{Z}) - \mu(\mathbf{Z})$, and $\mathbf{B}_{\mathbf{U}} = \kappa(\mathbf{U},\mathbf{Z})\kappa(\mathbf{Z},\mathbf{Z})^{-1}$. The residual is $\mathbf{e}_{\mathbf{U}} = \tilde{g}(\mathbf{U}) - \mathbf{B}_{\mathbf{U}}\tilde{g}(\mathbf{Z})$. Since $\mathbb{E}[\mathbf{e}_{\mathbf{U}}] = \mathbf{0}$, $\mathbb{E}\|\mathbf{e}_{\mathbf{U}}\|_2^2 = \operatorname{tr}(\operatorname{Cov}(\mathbf{e}_{\mathbf{U}}))$. Expanding the covariance gives

$$\begin{aligned}\operatorname{Cov}(\mathbf{e}_{\mathbf{U}}) &= \kappa(\mathbf{U},\mathbf{U}) - \mathbf{B}_{\mathbf{U}}\kappa(\mathbf{Z},\mathbf{U}) - \kappa(\mathbf{U},\mathbf{Z})\mathbf{B}_{\mathbf{U}}^{\top} + \mathbf{B}_{\mathbf{U}}\kappa(\mathbf{Z},\mathbf{Z})\mathbf{B}_{\mathbf{U}}^{\top} \\ &= \kappa(\mathbf{U},\mathbf{U}) - \kappa(\mathbf{U},\mathbf{Z})\kappa(\mathbf{Z},\mathbf{Z})^{-1}\kappa(\mathbf{Z},\mathbf{U}).\end{aligned}$$

Taking the trace proves the result. □

Proposition 7 (Path-regularity inheritance)**.** *Assume g has almost surely continuous sample paths and $\mathbf{x} \mapsto \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{x})$ is continuous. Then every* GMVIP *sample*

$$f(\mathbf{x}) = g(\mathbf{x}) + \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{x})\big[\boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z})\big]$$

is almost surely continuous. The same statement holds for differentiability or higher-order smoothness whenever g and $\mathbf{x} \mapsto \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{x})$ have the corresponding regularity.

Proof. For a fixed draw of $\mathbf{a}$ and $g(\mathbf{Z})$, the vector $\mathbf{c} = \boldsymbol{\mu} + \mathbf{D}\mathbf{a} - g(\mathbf{Z})$ is finite-dimensional and constant in $\mathbf{x}$. Hence

$$f(\mathbf{x}) = g(\mathbf{x}) + \sum_{j=1}^{d_a} \mathbf{\Psi}_{\mathbf{Z},j}(\mathbf{x})c_j.$$

A finite linear combination of continuous functions is continuous, and adding it to the continuous path g preserves continuity. Here $\mathbf{\Psi}_{\mathbf{Z},j}(\mathbf{x})$ is the j-th column of the operator (a scalar in the scalar-output case and a C-vector in the joint case). The same argument applies to derivatives of any order for which both g and these columns of $\mathbf{\Psi}_{\mathbf{Z}}$ possess the required regularity. □

B Optimization and Evaluation Procedures

Let $\mathcal{G} = \{g^{(b)}\}_{b=1}^{B}$ denote the prior trajectory bank. From this bank, we estimate the empirical mean $\hat{\boldsymbol{\mu}}$ and covariance $\hat{\mathbf{K}}$. Assuming $\hat{\mathbf{K}}(\mathbf{Z},\mathbf{Z}) \succ 0$, we set

$$\mathbf{D}\mathbf{D}^{\top} = \hat{\mathbf{K}}(\mathbf{Z},\mathbf{Z}), \qquad \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X}) = \hat{\mathbf{K}}(\mathbf{X},\mathbf{Z})\hat{\mathbf{K}}(\mathbf{Z},\mathbf{Z})^{-1}.$$

The inducing values and posterior path are

$$\boldsymbol{\mu} = \hat{\boldsymbol{\mu}}(\mathbf{Z}), \qquad \mathbf{u}(\mathbf{a}) = \boldsymbol{\mu} + \mathbf{D}\mathbf{a}, \qquad f(\mathbf{X}) = g(\mathbf{X}) + \mathbf{\Psi}_{\mathbf{Z}}(\mathbf{X})\big[\mathbf{u}(\mathbf{a}) - g(\mathbf{Z})\big].$$

For vector-valued outputs, inducing evaluations and coefficients are vectorized so that $\mathbf{D}$ and $q_\phi(\mathbf{a})$ retain the joint cross-output covariance. In the electricity experiment, $\mathbf{Z} = \mathbf{X}_{\text{obs}}$ contains the 96 observed locations and does not extend into the held-out interval.

Algorithm 1 optimizes the coefficient posterior and likelihood parameters using only the observed partition. After training, Algorithm 2 draws coherent posterior paths and computes metrics exclusively on the held-out interval.

When the prior parameters or inducing locations are optimized, we keep the bank's base random draws fixed, reevaluate the bank under the current parameters, and recompute the empirical moments and replacement operator at each optimization step. Gradients are propagated through these operations.

For sample-based C-output regression with a diagonal Gaussian likelihood, the electricity and Lotka–Volterra NLL is computed from the equal-weight Gaussian mixture

$$-\frac{1}{N_* C}\sum_{i=1}^{N_*}\sum_{c=1}^{C}\log\left[\frac{1}{R_{\text{ev}}}\sum_{r=1}^{R_{\text{ev}}}\mathcal{N}\Big(y_{*,ic};\big[\mathbf{f}_*^{(r)}\big]_{ic},\sigma_{y,c}^2\Big)\right].$$

Algorithm 1 GMVIP training loop

Require: Training data $\mathcal{D} = (\mathbf{X}, \mathbf{Y})$, prior sampler $p_0(g)$, operator $(\mathbf{Z}, \boldsymbol{\mu}, \mathbf{D}, \boldsymbol{\Psi}_{\mathbf{Z}})$, iterations T_{opt}, samples R_{tr}.
Ensure: Variational posterior $q_\phi(\mathbf{a})$ and likelihood parameters $\boldsymbol{\psi}$

Initialize $q_\phi(\mathbf{a}) = \mathcal{N}(\mathbf{m}_a, \mathbf{S}_a)$ and $\boldsymbol{\psi}$
for $t = 1, \ldots, T_{\text{opt}}$ **do**
  Draw a minibatch $(\mathbf{X}_\mathcal{B}, \mathbf{Y}_\mathcal{B})$
  for $r = 1, \ldots, R_{\text{tr}}$ **do**
    Draw $\mathbf{a}^{(r)} \sim q_\phi(\mathbf{a})$
    Draw one coherent path $g^{(r)} \sim p_0(g)$ at $\mathbf{X}_\mathcal{B} \cup \mathbf{Z}$
    $\mathbf{u}^{(r)} \leftarrow \boldsymbol{\mu} + \mathbf{D}\mathbf{a}^{(r)}$
    $\mathbf{f}_\mathcal{B}^{(r)} \leftarrow g^{(r)}(\mathbf{X}_\mathcal{B}) + \boldsymbol{\Psi}_{\mathbf{Z}}(\mathbf{X}_\mathcal{B})\big[\mathbf{u}^{(r)} - g^{(r)}(\mathbf{Z})\big]$
  end for
  $\widehat{\mathcal{L}}_t \leftarrow \dfrac{N}{|\mathcal{B}| R_{\text{tr}}} \displaystyle\sum_{r=1}^{R_{\text{tr}}} \log p_{\boldsymbol{\psi}}\big(\mathbf{Y}_\mathcal{B} \mid \mathbf{f}_\mathcal{B}^{(r)}\big) - \mathrm{KL}(q_\phi(\mathbf{a}) \,\|\, \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_a}))$
  Update $(\phi, \boldsymbol{\psi})$ with an Adam step on $-\widehat{\mathcal{L}}_t$
end for
return $q_\phi(\mathbf{a})$ and $\boldsymbol{\psi}$

Algorithm 2 GMVIP evaluation loop

Require: Fitted $q_\phi(\mathbf{a})$, prior sampler $p_0(g)$, operator $(\mathbf{Z}, \boldsymbol{\mu}, \mathbf{D}, \boldsymbol{\Psi}_{\mathbf{Z}})$, test inputs $\mathbf{X}_*$, targets $\mathbf{Y}_*$, likelihood parameters $\boldsymbol{\psi}$, and samples R_{ev}
Ensure: Posterior samples and target-level metrics

Disable gradient computation
for $r = 1, \ldots, R_{\text{ev}}$ **do**
  Draw $\mathbf{a}^{(r)} \sim q_\phi(\mathbf{a})$
  Draw one coherent path $g^{(r)} \sim p_0(g)$ at $\mathbf{X}_* \cup \mathbf{Z}$
  $\mathbf{u}^{(r)} \leftarrow \boldsymbol{\mu} + \mathbf{D}\mathbf{a}^{(r)}$
  $\mathbf{f}_*^{(r)} \leftarrow g^{(r)}(\mathbf{X}_*) + \boldsymbol{\Psi}_{\mathbf{Z}}(\mathbf{X}_*)\big[\mathbf{u}^{(r)} - g^{(r)}(\mathbf{Z})\big]$
end for
Compute $\bar{\mathbf{f}}_* = R_{\text{ev}}^{-1} \sum_{r=1}^{R_{\text{ev}}} \mathbf{f}_*^{(r)}$ and the empirical predictive quantiles
Compute RMSE, NLL, CRPS, CQM, and coverage on $(\mathbf{X}_*, \mathbf{Y}_*)$
Store the posterior samples and target-level metrics
return $\{\mathbf{f}_*^{(r)}\}_{r=1}^{R_{\text{ev}}}$ and the metrics

The mixture is evaluated with a stable log-sum-exp calculation. In the Lotka–Volterra and electricity experiments, $R_{\text{ev}} = 1024$. For VIP, FTIP, and GMVIP, $\sigma_{y,c}$ is learned for each output channel and initialized at the nominal observation noise. Training-free prior baselines and analytic Gaussian baselines retain the fixed nominal likelihood. The UCI and large-regression experiments evaluate each method's predictive density; for latent-sample methods this has the same mixture form above. CRPS is computed from the empirical posterior samples, and coverage uses their central predictive intervals. For the scalar benchmarks, let u_i denote the PIT values and define $\gamma(\ell) = N_*^{-1} \sum_i \mathbf{1}\{|2u_i - 1| \le \ell\}$. The regression benchmarks use

$$\mathrm{CQM} = \int_0^1 |\gamma(\ell) - \ell| \, d\ell,$$

evaluated by trapezoidal quadrature on 200 equally spaced levels. The electricity experiment instead uses its predefined nine-level quadrature,

$$\mathrm{CQM} = \frac{1}{9} \sum_{\ell \in \{0.1, \ldots, 0.9\}} |\hat{c}_\ell - \ell| \, .$$

Here $\hat{c}_\ell$ is the empirical coverage of the central predictive interval with nominal level ℓ.

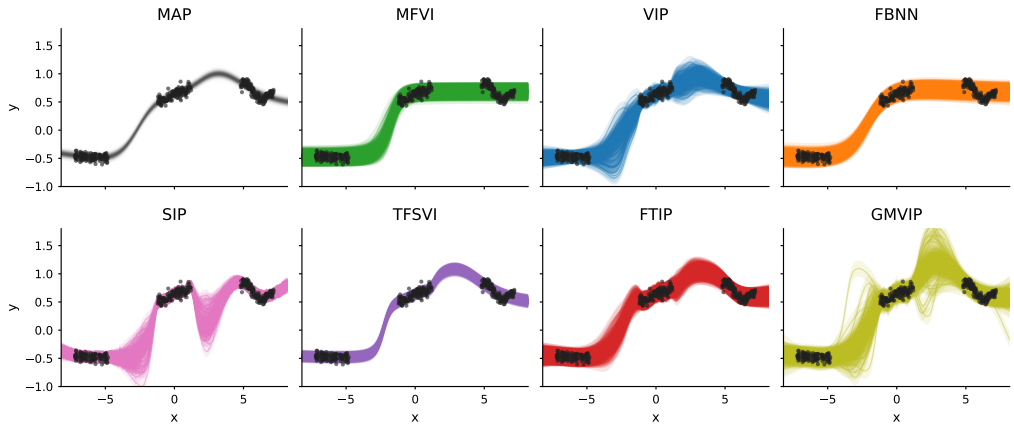


Figure 4: Synthetic one-dimensional regression. Black points denote observed data. Each panel shows the predictive behavior of one method on the same input grid. MAP gives a deterministic fit, while the Bayesian and function-space methods represent posterior uncertainty through predictive samples with learned Gaussian noise.

Electricity metrics are computed independently for each of the 75 held-out windows and summarized by the mean and sample standard errors. Lotka–Volterra metrics are evaluated only on the held-out test interval and summarized by the mean and standard error across the 20 target trajectories.

C Experimental Details

C.1 Synthetic Regression Experiment

We adapt the synthetic regression dataset introduced by Izmailov et al. (2020) and subsequently used by Ortega et al. (2024). The fixed realization contains $N = 400$ scalar observations

$$\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}, \qquad x_i, y_i \in \mathbb{R}.$$

The inputs lie approximately in $[-7.20, 7.18]$, and the outputs in $[-0.61, 0.90]$. All observed points are used for training, and posterior predictions are evaluated on a dense one-dimensional grid.

Implicit Process Prior. The implicit process prior is induced by a Bayesian neural network. Sampling weights and biases $\mathbf{w} \sim p(\mathbf{w})$ and evaluating the network gives

$$f_{\mathbf{w}}(x) = \mathrm{BNN}_{\mathbf{w}}(x), \qquad f \sim p(f).$$

The BNN has two hidden layers of width $(10, 10)$, $\tanh$ activations, and a scalar output. In the frozen-prior setting, Bayesian linear layers use zero-mean Gaussian weights and biases with unit standard deviation,

$$\mathbf{w}_\ell \sim \mathcal{N}\big(\mathbf{0}, \mathbf{I}_{\dim(\mathbf{w}_\ell)}\big)\,, \qquad \mathbf{b}_\ell \sim \mathcal{N}\big(\mathbf{0}, \mathbf{I}_{\dim(\mathbf{b}_\ell)}\big)\,.$$

Learning Problem. We use Gaussian regression,

$$y_i = f(x_i) + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma_y^2),$$

and infer the posterior predictive distribution

$$p(f(x_\star) \mid \mathcal{D})$$

on the plotting grid.

Methods and Visualization. Figure 4 compares MAP, MFVI, VIP, FBNN, SIP, TFSVI, FTIP, and GMVIP on the same grid. Black points are observations. MAP gives a deterministic curve, while the other methods show posterior uncertainty through predictive samples or bands. VIP learns a variational distribution over finite regression coefficients using samples from the BNN-induced prior; GMVIP uses the same function prior but constructs a generalized Matheron posterior through inducing evaluations.

C.2 UCI Regression Experiments

We evaluate on nine scalar UCI regression datasets:

$$\{\text{Boston}, \text{Concrete}, \text{Energy}, \text{Kin8nm}, \text{Naval}, \text{Power}, \text{Protein}, \text{Wine Red}, \text{Yacht}\}.$$

For each seed, we use a random $90/10$ train/test split. Inputs are standardized using training-split statistics. Targets are normalized during training and all reported metrics are transformed back to the original target scale. Naval RMSE and CRPS are reported in units of 10^{-4}.

Sweep design. We run five seeds, $\{0, 1, 2, 3, 4\}$, for each dataset and method variant. The reported variants are MAP, MFVI, FBNN, TFSVI, VIP, FTIP, GMVIP, and SIP. We use $T_{\text{opt}} = 30\,000$ iterations for Boston, Concrete, Energy, and Protein, and $T_{\text{opt}} = 60\,000$ iterations for Kin8nm, Naval, Power, Wine Red, and Yacht.

Shared training configuration. All runs use batch size 100 and Adam with initial learning rate 10^{-3}. A cosine schedule anneals the learning rate to 10^{-5} over the training budget. The shared network architecture is a Bayesian MLP with two hidden layers of width $(10, 10)$, $\tanh$ activations, scalar output.

Likelihood and predictive distribution. All probabilistic methods use the Gaussian regression likelihood

$$y_i = f(x_i) + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma_y^2),$$

with a learned observation-noise parameter where applicable. Predictive distributions are evaluated from the method-specific posterior predictive samples or Gaussian predictive moments.

Method-specific settings. MAP is trained as a deterministic neural-network baseline with an L_2 penalty. MFVI uses $R_{\text{tr}} = 512$ Monte Carlo samples during training and $R_{\text{ev}} = 512$ evaluation samples. FBNN uses a BNN functional prior, freezes the prior parameters, uses 512 regression coefficients, and evaluates with $R_{\text{ev}} = 512$ samples. TFSVI uses $R_{\text{tr}} = 512$ training samples and $R_{\text{ev}} = 512$ evaluation samples.

VIP uses $S = 20$ sampled-function basis coefficients. FTIP uses the same $S = 20$ basis, a `spline_1x1` normalizing flow, $R_{\text{tr}} = 512$ flow samples during training, $R_{\text{ev}} = 512$ evaluation samples, and no VIP warm start, so the optimizer budget is not inflated by pretraining.

SIP uses $M = 100$ inducing inputs initialized by k-means, learns the inducing locations, and uses $B = 512$ prior samples, $R_{\text{tr}} = 512$ training samples, and $R_{\text{ev}} = 512$ evaluation samples. GMVIP uses the empirical Matheron operator, a Gaussian inducing-coefficient posterior, a tunable BNN prior, $M = 100$ inducing inputs initialized by k-means, learned inducing locations, $B = 512$ operator-bank samples, $R_{\text{tr}} = 512$ training samples, $R_{\text{ev}} = 512$ evaluation samples, prior-sample mean mode, and prior-Cholesky inducing scale.

Metrics. Tables 1and12 reports RMSE, negative log-likelihood (NLL), continuous ranked probability score (CRPS), and centered quantile metric (CQM). All metrics are computed on the held-out test split after undoing target normalization. NLL evaluates the predictive density at the observed targets. CRPS measures the quality of the full one-dimensional predictive distribution using posterior predictive samples. CQM is computed from probability integral transform values and measures centered predictive-interval calibration error. Lower is better for all reported metrics.

C.3 Lotka–Volterra Implicit-Prior Experiment

We define an implicit process prior through the Lotka–Volterra predator–prey system. For parameters

$$\boldsymbol{\eta} = (\alpha, \beta, \delta, \gamma, u_0, v_0),$$

the physical-time state

$$\mathbf{s}_{\boldsymbol{\eta}}(t) = \big(u_{\boldsymbol{\eta}}(t), v_{\boldsymbol{\eta}}(t)\big)$$

is the solution of

$$\frac{du}{dt} = \alpha u - \beta uv,$$

$$\frac{dv}{dt} = \delta uv - \gamma v,$$

with initial condition

$$u(0) = u_0, \qquad v(0) = v_0.$$

Here $u(t)$ denotes prey population and $v(t)$ denotes predator population. We use the normalized regression coordinate $x = 2t/30 - 1$, with inverse $t(x) = 15(x+1)$, and define $f_{\boldsymbol{\eta}}(x) = \mathbf{s}_{\boldsymbol{\eta}}(t(x))$. The implicit prior is the push-forward of a parameter distribution through the ODE solver:

$$\boldsymbol{\eta} \sim p(\boldsymbol{\eta}), \qquad \mathbf{s}_{\boldsymbol{\eta}} = \mathrm{ODESolve}(\boldsymbol{\eta}), \qquad f_{\boldsymbol{\eta}}(x) = \mathbf{s}_{\boldsymbol{\eta}}(t(x)),$$

and therefore

$$f_{\boldsymbol{\eta}} \sim p(f).$$

In the experiment we use

$$\alpha \sim \mathrm{LogNormal}(\log 1.5, 0.15^2), \qquad \beta \sim \mathrm{LogNormal}(\log 1.0, 0.15^2),$$
$$\delta \sim \mathrm{LogNormal}(\log 0.75, 0.15^2), \qquad \gamma \sim \mathrm{LogNormal}(\log 1.0, 0.15^2),$$

and

$$u_0, v_0 \sim \mathrm{Uniform}(0.8, 1.2).$$

The inference methods do not receive an explicit variational distribution over $\boldsymbol{\eta}$. The prior is used as a black-box function sampler: one may draw

$$f^{(s)} \sim p(f)$$

and evaluate $f^{(s)}(x)$ at requested inputs, but the simulator parameters are not optimized.

Learning Problem Each target trajectory is generated by sampling a held-out parameter vector $\boldsymbol{\eta}^\star \sim p(\boldsymbol{\eta})$ and solving the ODE on $t \in [0, 30]$. The regression input is the normalized time defined above, so that $x \in [-1, 1]$. The output is vector-valued: $f(x) = \big(u(t), v(t)\big) \in \mathbb{R}^2$.

Training observations are noisy samples from the target trajectory:

$$\mathbf{y}_i = f_{\boldsymbol{\eta}^\star}(x_i) + \boldsymbol{\epsilon}_i, \qquad \boldsymbol{\epsilon}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_y),$$

using $N_{\mathrm{obs}} = 80$ training times restricted to $t_i \le 15$. For output channel c, the physical observation-noise standard deviation is 0.03 times the standard deviation of its clean training values; the likelihood uses this value after target normalization. The interval $15 < t \le 20$ is deliberately unused: it supplies neither observations, validation scores, nor checkpoint selection. Test points are clean trajectory values on $t \in (20, 30]$. Thus the task is sparse forecasting from early predator–prey observations rather than interpolation over the full trajectory. The reported aggregate uses 20 deterministic held-out targets. Outputs are normalized separately for each target using the noisy training observations. For each trainable method, the normalized Gaussian observation-noise scale is initialized from the known simulator noise and learned separately for prey and predator. The training-free prior predictive and GMVIP surrogate-prior controls use the known simulator noise.

Methods This experiment is designed for implicit-process inference. The object we want to use as a prior is the simulator-induced distribution $p(f)$, not a neural-network generator with trainable weights. Standard MAP or mean-field variational inference baselines require introducing an explicit parametric generator and fitting a distribution over its parameters. That changes the problem: inference is then performed over the surrogate generator rather than directly with the ODE-induced prior.

VIP, FTIP, and GMVIP are appropriate methods in this setting because they require only samples from $f \sim p(f)$. VIP constructs its approximation from a finite collection of sampled prior functions. FTIP starts from a VIP approximation and then applies a learnable transport map in function-sample space. GMVIP also uses simulator samples, but forms a generalized Matheron posterior representation through inducing evaluations.

These are the headline configurations; Appendix D.5 reports sensitivity to the sampled-function basis size S for VIP and FTIP.

In the reported setting, the standalone VIP baseline uses $S = 20$ sampled prior functions and is trained for $T_{\mathrm{opt}} = 800$ optimization steps. FTIP is warm-started from a VIP model using $S = 20$

sampled prior functions, then fine-tuned with a coupling affine flow using $R_{\mathrm{tr}} = 8$ Monte Carlo samples per step for $T_{\mathrm{opt}} = 400$ additional optimization steps. Appendix D.5 reports sensitivity to $S \in \{20, 64, 128, 256\}$ under the same forecasting protocol. GMVIP estimates its empirical operator from $B = 512$ simulator trajectories, uses $M = 96$ shared inducing times, and is trained for $T_{\mathrm{opt}} = 800$ optimization steps with $R_{\mathrm{tr}} = 4$. All trained models are evaluated at the exact final scheduled iterate: no validation loss, early stopping, or checkpoint selection is used. All methods use $R_{\mathrm{ev}} = 1024$ posterior samples for evaluation, including the equal-weight Gaussian-mixture NLL and the reported trajectory plot.

Joint multioutput parameterization. GMVIP uses $M = 96$ shared inducing times, uniformly spaced over $t \in [0, 30]$. These are input locations only. Prey and predator inducing values are stacked into a vector of dimension $2M = 192$. The empirical inducing covariance $\hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z})$ is a full 192×192 matrix containing prey–prey, predator–predator, and prey–predator covariance blocks. The variational posterior over the corresponding whitened inducing coefficients also uses a full 192×192 covariance. Thus GMVIP explicitly models posterior dependence across both time and the two state variables in this experiment.

ODE residual. The ODE residual is evaluated in physical state and time units on the held-out interval $t \in (20, 30]$. It measures whether each posterior trajectory is consistent with *some* Lotka–Volterra parameter vector. It does not use the parameters that generated the target trajectory. For each of up to 64 posterior trajectories

$$f^{(s)}(t_i) = \left(u_i^{(s)}, v_i^{(s)}\right),$$

we estimate a separate parameter vector

$$\widehat{\boldsymbol{\lambda}}^{(s)} = \left(\widehat{\alpha}^{(s)}, \widehat{\beta}^{(s)}, \widehat{\delta}^{(s)}, \widehat{\gamma}^{(s)}\right)$$

by unconstrained least squares. Derivatives are approximated at the interior test-grid points using centered finite differences:

$$\dot{u}_i^{(s)} = \frac{u_{i+1}^{(s)} - u_{i-1}^{(s)}}{t_{i+1} - t_{i-1}}, \qquad \dot{v}_i^{(s)} = \frac{v_{i+1}^{(s)} - v_{i-1}^{(s)}}{t_{i+1} - t_{i-1}}.$$

For the regression and right-hand-side evaluation, states are clipped as

$$u_i^{+} = \max(u_i^{(s)}, 10^{-8}), \qquad v_i^{+} = \max(v_i^{(s)}, 10^{-8}).$$

The prey and predator parameters are then fitted independently:

$$(\widehat{\alpha}^{(s)}, \widehat{\beta}^{(s)}) = \arg\min_{\alpha,\beta} \sum_{i=2}^{N_t-1} \left[\dot{u}_i^{(s)} - \left(\alpha u_i^{+} - \beta u_i^{+} v_i^{+}\right)\right]^2,$$

$$(\widehat{\delta}^{(s)}, \widehat{\gamma}^{(s)}) = \arg\min_{\delta,\gamma} \sum_{i=2}^{N_t-1} \left[\dot{v}_i^{(s)} - \left(\delta u_i^{+} v_i^{+} - \gamma v_i^{+}\right)\right]^2.$$

No positivity constraints are imposed on the fitted parameters. For posterior trajectory s, define

$$\mathbf{F}_i^{(s)} = \begin{bmatrix} \widehat{\alpha}^{(s)} u_i^{+} - \widehat{\beta}^{(s)} u_i^{+} v_i^{+} \\ \widehat{\delta}^{(s)} u_i^{+} v_i^{+} - \widehat{\gamma}^{(s)} v_i^{+} \end{bmatrix}, \qquad \dot{\mathbf{f}}_i^{(s)} = \begin{bmatrix} \dot{u}_i^{(s)} \\ \dot{v}_i^{(s)} \end{bmatrix}.$$

Its normalized residual is

$$\mathcal{R}^{(s)} = \frac{\dfrac{1}{2(N_t-2)} \displaystyle\sum_{i=2}^{N_t-1} \left\| \dot{\mathbf{f}}_i^{(s)} - \mathbf{F}_i^{(s)} \right\|_2^2}{\max\left\{ \dfrac{1}{2(N_t-2)} \displaystyle\sum_{i=2}^{N_t-1} \left\| \dot{\mathbf{f}}_i^{(s)} \right\|_2^2, 10^{-12} \right\}}.$$

The numerator and denominator therefore average over both prey and predator and over all interior test-grid points. The normalization makes the reported residual dimensionless. Finally, the per-target ODE residual is

$$E_{\mathrm{ODE}} = \frac{1}{R'_{\mathrm{ev}}} \sum_{s=1}^{R'_{\mathrm{ev}}} \mathcal{R}^{(s)}, \qquad R'_{\mathrm{ev}} = \min(R_{\mathrm{ev}}, 64).$$

Although the input trajectories and derivatives are evaluated in physical units, only E_{ODE}, the normalized residual ratio, is reported.

C.4 Electricity-Load Empirical-Prior Forecasting Experiment

We define a real-data empirical process prior using the Electricity Load dataset (Trindade, 2015). The dataset contains electricity-consumption trajectories sampled every 15 minutes. Let

$$y_{c,\tau}(t_j), \qquad j = 0, \dots, 191,$$

denote the 48-hour load window for customer c, starting at calendar time τ, with

$$t_j = 0.25j \quad \text{hours}.$$

For each target window, the first 24 hours are observed and the next 24 hours are held out for forecasting. Thus each window has

$$N_t = 192, \qquad N_{\mathrm{obs}} = 96, \qquad N_{\mathrm{test}} = 96$$

quarter-hourly points.

Unlike the ODE experiments, the prior is not generated by a known simulator. Instead, the prior is an empirical distribution over historical trajectories. For a target window $y^\star$, we construct a prior bank

$$\mathcal{G}(y^\star) = \{f^{(1)}, \dots, f^{(B)}\}, \qquad B = 2048,$$

from historical windows in the years 2011–2013. Test targets are sampled from 2014, so the prior bank and test targets are separated by calendar year.

All windows are normalized using statistics from their observed prefix. For a window y, define

$$\mu_y = \frac{1}{N_{\mathrm{obs}}} \sum_{j=0}^{N_{\mathrm{obs}}-1} y(t_j), \qquad s_y = \max\left\{\mathrm{std}\big(y(t_0), \dots, y(t_{N_{\mathrm{obs}}-1})\big), 10^{-3}\right\}.$$

The normalized trajectory is

$$\tilde{y}(t_j) = \frac{y(t_j) - \mu_y}{s_y}.$$

Each historical prior trajectory is normalized by its own observed prefix statistics before being added to the bank. The bank is selected by a calendar-and-prefix nearest-neighbor rule. Candidate historical windows must come from prior years, match the target's weekday/weekend status, and have month in

$$\{m^\star - 1, m^\star, m^\star + 1\},$$

with months interpreted cyclically. Among these candidates, we compute the normalized prefix distance

$$d(y, y^\star) = \frac{1}{N_{\mathrm{obs}}} \sum_{j=0}^{N_{\mathrm{obs}}-1} \left(\tilde{y}(t_j) - \tilde{y}^\star(t_j)\right)^2,$$

and select the $B = 2048$ nearest historical windows. The empirical prior is therefore

$$f \sim p_{\mathrm{emp}}(f \mid y^\star), \qquad p_{\mathrm{emp}}(f \mid y^\star) = \frac{1}{B} \sum_{b=1}^{B} \delta_{f^{(b)}}(f).$$

This is an implicit prior: the methods can draw trajectories from the historical bank and evaluate them at requested time points, but no tractable density over functions or latent calendar factors is provided.

Learning Problem For each target, the regression input is normalized time

$$x_j = 2t_j/48 - 1, \qquad x_j \in [-1, 1].$$

The output is scalar normalized load,

$$f(x_j) = \tilde{y}^\star(t_j) \in \mathbb{R}.$$

Training observations are the first 24 hours of the target window:

$$\mathcal{D}_{\text{obs}} = \{(x_j, \tilde{y}^\star(t_j)) : j = 0, \dots, 95\}.$$

The held-out forecasting grid is

$$t_j \in [24, 48), \qquad j = 96, \dots, 191.$$

Metrics are computed after transforming predictions back to the original load scale using the target-window prefix statistics $(\mu_{y^\star}, s_{y^\star})$.

We use a Gaussian observation model in normalized units,

$$y_i = f(x_i) + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma_y^2),$$

with

$$\sigma_y^{(0)} = 0.05.$$

For VIP, FTIP, and GMVIP, the scalar normalized observation-noise scale is initialized at 0.05 and optimized jointly with the variational parameters. The Analog prior and Empirical Gaussian baselines retain the fixed nominal value 0.05. In physical units, a normalized scale σ_y corresponds to $\sigma_y s_{y^\star}$ for each target window.

We run three target-selection seeds, $\{0, 1, 2\}$, with 25 held-out target windows per seed, for 75 total target windows. All targets are sampled from 2014, while all empirical-prior trajectories are drawn from 2011–2013.

Methods This experiment evaluates whether the same function-space inference mechanisms used for simulator priors also apply to a real empirical trajectory prior. The prior is the historical empirical process $p_{\text{emp}}(f \mid y^\star)$; the methods are not given a parametric model for electricity demand and do not optimize customer-specific latent calendar variables.

The analog baseline directly uses samples from the empirical prior bank without posterior adaptation. VIP constructs a coefficient posterior from $S = 20$ sampled prior functions. FTIP uses the same $S = 20$ basis and adds a learned transport map in coefficient space. GMVIP uses the empirical Matheron operator with a Gaussian inducing-coefficient posterior.

In the reported setting, VIP and FTIP use $S = 20$ sampled-function basis coefficients. GMVIP uses $B = 2048$ operator-bank trajectories and $M = 96$ inducing locations, with $\mathbf{Z} = \mathbf{X}_{\text{obs}}$ corresponding to the observed 24-hour input grid. All learned methods are trained for $T_{\text{opt}} = 500$ optimization steps, with $R_{\text{tr}} = 8$ Monte Carlo samples per training step. Observation noise is learned in every trainable method. Runs are evaluated at the final scheduled iterate, with no validation loss, early stopping, or checkpoint selection. All methods use $R_{\text{ev}} = 1024$ posterior samples for final metrics and plots.

Empirical Gaussian baseline. For each target window, the Empirical Gaussian baseline uses the same $B = 2048$ historical trajectories retrieved for GMVIP. Each historical trajectory is standardized using the mean and standard deviation of its own 96-point observed prefix. The target trajectory is analogously standardized using its observed prefix. All conditioning is performed in these normalized coordinates.

Let $\mathbf{h}^{(b)} \in \mathbb{R}^{192}$, $b = 1, \dots, B$, denote the retrieved historical trajectories. We estimate the full-grid mean and unbiased empirical covariance as

$$\hat{\boldsymbol{\mu}} = \frac{1}{B} \sum_{b=1}^{B} \mathbf{h}^{(b)}, \qquad \hat{\mathbf{K}} = \frac{1}{B-1} \sum_{b=1}^{B} \left(\mathbf{h}^{(b)} - \hat{\boldsymbol{\mu}}\right) \left(\mathbf{h}^{(b)} - \hat{\boldsymbol{\mu}}\right)^\top.$$

Thus, $\hat{\mathbf{K}} \in \mathbb{R}^{192 \times 192}$ contains correlations between all observed and forecast time points.

Partition the grid into the observed prefix $o = \{1, \dots, 96\}$ and the held-out forecast interval $* = \{97, \dots, 192\}$. Under the Gaussian observation model

$$\mathbf{y}_o = \mathbf{f}_o + \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \sigma_y^2 \mathbf{I}_{96}),$$

the conditional latent distribution is

$$p(\mathbf{f}_* \mid \mathbf{y}_o) = \mathcal{N}(\mathbf{m}_*, \mathbf{C}_*),$$

Table 5: Electricity-load results across 75 seed–target windows. Entries are medians with 95% percentile-bootstrap confidence intervals from 100,000 resamples. Coverage is reported in percentage points and ranked by proximity to its nominal level. Lower is better otherwise.

Method	RMSE ↓	NLL ↓	CRPS ↓	CQM ↓	Cov. 80%	Cov. 90%
Analog prior	17.46 [12.80, 25.41]	**4.30** [**4.06**, **4.70**]	8.92 [6.49, 13.78]	**0.093** [**0.077**, **0.112**]	**78.13** [**73.96**, **84.38**]	**91.67** [**88.54**, **92.71**]
VIP	22.94 [14.13, 27.18]	10.05 [8.60, 11.64]	15.41 [9.70, 18.26]	0.384 [0.365, 0.396]	20.83 [17.71, 22.92]	25.00 [21.88, 29.17]
FTIP	21.20 [15.75, 28.02]	4.88 [4.27, 5.37]	12.35 [8.81, 17.57]	0.153 [0.132, 0.174]	57.29 [54.17, 61.46]	68.75 [66.67, 73.96]
Empirical Gaussian	**16.99** [**11.14**, **25.10**]	4.36 [4.02, 5.06]	**8.39** [**6.06**, **14.05**]	0.108 [0.070, 0.151]	84.38 [80.21, 87.50]	**91.67** [**89.58**, **93.75**]
GMVIP	**16.20** [**11.14**, **24.85**]	**4.08** [**3.79**, **4.54**]	**8.14** [**5.93**, **13.52**]	**0.102** [**0.072**, **0.112**]	**80.21** [**77.08**, **84.38**]	**92.71** [**89.58**, **93.75**]

where

$$\mathbf{m}_* = \hat{\boldsymbol{\mu}}_* + \hat{\mathbf{K}}_{*o}\left(\hat{\mathbf{K}}_{oo} + \sigma_y^2 \mathbf{I}_{96}\right)^{-1}(\mathbf{y}_o - \hat{\boldsymbol{\mu}}_o),$$

and

$$\mathbf{C}_* = \hat{\mathbf{K}}_{**} - \hat{\mathbf{K}}_{*o}\left(\hat{\mathbf{K}}_{oo} + \sigma_y^2 \mathbf{I}_{96}\right)^{-1}\hat{\mathbf{K}}_{o*}.$$

Here $\mathbf{C}_*$ is the conditional covariance of the latent forecast. The baseline is analytic and uses no optimization steps.

Metrics We report RMSE, negative log-likelihood (NLL), continuous ranked probability score (CRPS), and centered quantile metric (CQM) on the held-out forecasting region $t \in [24, 48)$. All metrics are computed in the original load units after undoing the target normalization. The main paper reports the mean and sample standard errors across the 75 seed–target windows.

Results Summary As shown in Table 5, we report medians because RMSE exhibits substantial variation across forecasting windows. Electricity clients have markedly different load scales, and a few high-load or unusually difficult windows produce extreme absolute errors that disproportionately influence the mean. The median therefore better represents performance on a typical task, while the bootstrap confidence intervals quantify uncertainty across windows. GMVIP provides the strongest typical point and distributional forecasting performance, although its confidence intervals overlap those of the strongest baselines. Calibration remains dependent on the metric and nominal coverage level.

C.5 Large Regression Experiments

We evaluate on two large scalar regression datasets:

$$\{\text{Year}, \text{Airline}\}.$$

The Year dataset uses the standard fixed split with the first 463 715 examples for training and the remaining examples for testing. The Airline dataset uses the first 800 000 examples, with the first 700 000 for training and the remaining 100 000 for testing. Inputs are standardized using training-split statistics. Targets are normalized during training and all reported metrics are transformed back to the original target scale.

Sweep design. We run five seeds, $\{0, 1, 2, 3, 4\}$, for each dataset and method variant. The reported variants are MFVI, FBNN, TFSVI, VIP, FTIP, GMVIP, and SIP. MAP and the fixed-prior variants of VIP, FTIP, GMVIP, and SIP are omitted from Table 4. We use $T_{\text{opt}} = 60\,000$ optimization iterations for both Year and Airline.

Shared training configuration. All runs use batch size 100 and Adam with initial learning rate 10^{-3}. A cosine schedule anneals the learning rate to 10^{-5} over the training budget. The shared model class is a Bayesian MLP with $\tanh$ activations, scalar output. Year uses two hidden layers of width $(50, 50)$, while Airline uses two hidden layers of width $(100, 100)$.

Likelihood and predictive distribution. All probabilistic methods use the Gaussian regression likelihood

$$y_i = f(x_i) + \varepsilon_i, \qquad \varepsilon_i \sim \mathcal{N}(0, \sigma_y^2),$$

with a learned observation-noise parameter where applicable. Predictive distributions are evaluated from the method-specific posterior predictive samples or Gaussian predictive moments.

Method-specific settings. MFVI uses $R_{\mathrm{tr}} = 512$ Monte Carlo samples during training and $R_{\mathrm{ev}} = 512$ evaluation samples. FBNN uses a BNN functional prior, freezes the prior parameters and evaluates with $R_{\mathrm{ev}} = 512$ samples. TFSVI uses $R_{\mathrm{tr}} = 512$ training samples and $R_{\mathrm{ev}} = 512$ evaluation samples.

VIP uses $S = 20$ sampled-function basis coefficients and learns the prior parameters. FTIP uses the same $S = 20$ basis, a `spline_1x1` normalizing flow, $R_{\mathrm{tr}} = 512$ flow samples during training, $R_{\mathrm{ev}} = 512$ evaluation samples.

SIP uses $M = 100$ inducing inputs initialized by k-means, learns the inducing locations, and uses $B = 512$ prior samples, $R_{\mathrm{tr}} = 512$ training samples, and $R_{\mathrm{ev}} = 512$ evaluation samples. GMVIP uses the empirical Matheron operator, a Gaussian inducing-coefficient posterior, a tunable BNN prior, $M = 100$ inducing inputs initialized by k-means, learned inducing locations, $B = 512$ operator-bank samples, $R_{\mathrm{tr}} = 512$ training samples, $R_{\mathrm{ev}} = 512$ evaluation samples, prior-sample mean mode, and prior-Cholesky inducing scale.

Metrics. Table 4 reports RMSE, negative log-likelihood (NLL), continuous ranked probability score (CRPS), and centered quantile metric (CQM). All metrics are computed on the held-out test split after undoing target normalization. NLL evaluates the predictive density at the observed targets. CRPS measures the quality of the full one-dimensional predictive distribution using posterior predictive samples. CQM is computed from probability integral transform values and measures centered predictive-interval calibration error. Lower is better for all reported metrics.

C.6 LeNet Classification Experiments

We evaluate image classification on two standard multiclass datasets,

$$\{\text{FashionMNIST}, \text{CIFAR-10}\}.$$

For both datasets we use the standard train/test split provided by the dataset: FashionMNIST has $60\,000$ training examples and $10\,000$ test examples, and CIFAR-10 has $50\,000$ training examples and $10\,000$ test examples. Images are flattened by the dataset wrapper and rescaled to $[0, 1]$; the LeNet model reshapes them back to image tensors internally. No input standardization or target normalization is applied. Targets are integer class labels and all methods use the same ten-class classification task.

Sweep design. We run five seeds, $\{0, 1, 2, 3, 4\}$, for each dataset and method variant. The reported variants are MAP, MFVI, FBNN, VIP, FTIP, GMVIP, and SIP. The main classification table reports the LeNet setting, where the convolutional feature extractor is deterministic and the classifier head is Bayesian.

Shared training configuration. All runs use $T_{\mathrm{opt}} = 10\,000$ optimizer iterations, batch size 128, Adam with learning rate 10^{-3}, no cosine annealing, and evaluation batch size 512. The LeNet backbone consists of two convolutional blocks followed by a classifier head:

$$\text{Conv}(C, 6, 5) \to \text{ReLU} \to \text{AvgPool} \to \text{Conv}(6, 16, 5) \to \text{ReLU} \to \text{AvgPool} \to \tag{14}$$

$$\to \text{FC}(120) \to \text{ReLU} \to \text{FC}(84) \to \text{ReLU} \to \text{FC}(10). \tag{15}$$

In the Bayesian setting, the convolutional layers are deterministic and the three fully connected classifier layers are Bayesian layers for probabilistic methods. Bayesian weight log-standard deviations are initialized at -3.0, and the prior weight log-standard deviation is also initialized at -3.0.

Likelihood and predictive distribution. All probabilistic methods use a categorical multiclass likelihood. Predictive class probabilities are obtained by averaging softmax probabilities over posterior predictive samples. MAP is evaluated from a single deterministic network. The remaining methods use their method-specific posterior predictive samplers with $R_{\mathrm{ev}} = 512$ evaluation samples unless otherwise stated.

Method-specific settings. MAP is trained as a deterministic LeNet baseline with cross-entropy and an L_2 penalty of 10^{-4}. MFVI uses $R_{\mathrm{tr}} = 512$ Monte Carlo samples during training and $R_{\mathrm{ev}} = 512$

samples at evaluation. FBNN uses a frozen BNN functional prior, $R_{\text{ev}} = 512$ posterior samples, and $B = 512$ prior samples.

VIP uses $S = 20$ sampled-function basis coefficients and learns the Bayesian classifier-head prior parameters. FTIP uses the same $S = 20$ basis, a `spline_1x1` normalizing flow, $R_{\text{tr}} = 512$ flow samples during training, $R_{\text{ev}} = 512$ evaluation samples, and learns the prior parameters.

SIP uses $M = 100$ inducing inputs initialized by k-means, learns the inducing locations, and uses $B = 512$ prior samples, $R_{\text{tr}} = 512$ training samples, and $R_{\text{ev}} = 512$ evaluation samples. GMVIP uses the empirical Matheron operator, a Gaussian inducing-coefficient posterior, $M = 100$ inducing inputs initialized by k-means, learned inducing locations, $B = 512$ operator-bank samples, $R_{\text{tr}} = 512$ training samples, and $R_{\text{ev}} = 512$ evaluation samples.

For classification, GMVIP uses a channel-wise multioutput approximation. Each of the ten latent logits has its own inducing-coefficient block, and the variational covariance is block-diagonal across logits. With $M = 100$, a joint formulation would require a full 1000×1000 coefficient covariance, so the block-diagonal parameterization is used for scalability. The softmax likelihood jointly normalizes the sampled logits, but cross-logit covariance is not explicitly parameterized in $q_\phi(\mathbf{a})$.

Metrics. Table 9 reports negative log-likelihood (NLL), classification error, expected calibration error (ECE), and Brier score on the held-out test split. NLL evaluates the predictive categorical density assigned to the observed class. Error is the fraction of incorrect predictions under the posterior mean class probabilities. ECE measures confidence calibration using binned predicted probabilities. The Brier score measures the squared error between predictive class probabilities and one-hot labels. Lower is better for all reported metrics.

D Further Experiments

D.1 Finite-Dimensional Surrogate-Prior Fidelity

The variational objective of GMVIP is defined relative to its Matheron surrogate prior rather than directly relative to the original implicit process. Proposition 5 characterizes two sources of finite-dimensional distortion: replacing the inducing-value law and removing the dependence between the residual path and the replacement inducing values. Proposition 4 shows that the population Matheron construction nevertheless preserves the original mean and covariance. This section measures the resulting finite-sample discrepancy directly and compares it with the coefficient-space surrogate used by VIP.

Label-free prior and surrogate constructions. We use the frozen scalar BNN prior from the synthetic experiment, but do not load observations, evaluate a likelihood, or perform optimization. The architecture is 1-10-10-1, with $\tanh$ hidden activations and independent zero-mean, unit-variance Gaussian weights and biases. We evaluate every sampled function on $N_x = 301$ uniformly spaced inputs over $[-5, 5]$.

For a VIP basis of S independently sampled prior functions $\mathcal{G}_{\text{VIP}} = \{g^{(s)}\}_{s=1}^{S}$, let

$$\hat{\mu}_S(\mathbf{x}) = \frac{1}{S}\sum_{s=1}^{S} g^{(s)}(\mathbf{x}), \qquad \phi_s(\mathbf{x}) = \frac{g^{(s)}(\mathbf{x}) - \hat{\mu}_S(\mathbf{x})}{\sqrt{S-1}}.$$

The VIP surrogate-prior sample is

$$f_{\text{VIP}}(\mathbf{x}) = \hat{\mu}_S(\mathbf{x}) + \sum_{s=1}^{S} \phi_s(\mathbf{x}) a_s, \qquad \mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_S).$$

Conditional on the sampled basis, this is a Gaussian finite-rank process with the empirical basis mean and covariance.

The empirical GMVIP surrogate uses M uniformly spaced inducing inputs, the paper's prior-bank notation $\mathcal{G} = \{g^{(b)}\}_{b=1}^{B}$, and

$$\begin{aligned} f_{\mathrm{GMVIP}}(\cdot) &= T(g, \mathbf{a})(\cdot) \\ &= g(\cdot) + \boldsymbol{\Psi}_{\mathbf{Z}}^{\mathrm{emp}}(\cdot)\left[\hat{\boldsymbol{\mu}}(\mathbf{Z}) + \mathbf{L}\mathbf{a} - g(\mathbf{Z})\right], \\ &\qquad\qquad \mathbf{a} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_M). \end{aligned}$$

Here g is a fresh coherent BNN-prior path, $\mathbf{L}\mathbf{L}^\top = \hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z})$ is the stabilized empirical inducing covariance, and $\boldsymbol{\Psi}_{\mathbf{Z}}^{\mathrm{emp}}(\cdot) = \hat{\mathbf{K}}(\cdot, \mathbf{Z})\hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z})^{-1}$, as in Equation 10. We use the same numerical settings as the synthetic GMVIP experiment. Both surrogates therefore use $p(\mathbf{a}) = \mathcal{N}(\mathbf{0}, \mathbf{I}_{d_a})$, with $d_a = S$ for VIP and $d_a = M$ for scalar-output GMVIP. No variational parameters are optimized in this experiment.

Sampling protocol and finite-sample floor. We run seeds 0–4. For every seed, disjoint random streams generate an operator or basis bank, surrogate coefficients and residual paths, and three independent sets of $R_{\mathrm{ev}} = 2\,048$ original-prior functions. The first original-prior set estimates a pointwise mean $\widehat{\mu}_j$ and standard deviation $\widehat{\sigma}_j$. All functions are standardized as

$$\widetilde{f}_j^{(r)} = \frac{f^{(r)}(x_j) - \widehat{\mu}_j}{\widehat{\sigma}_j}.$$

This common coordinatewise affine map makes the distances dimensionless and prevents high-variance input regions from dominating the comparison. Its parameters are estimated only from the independent original-prior calibration set; no surrogate sample influences the normalization. The second original-prior set is the reference distribution. The third is an independent *true-prior split*, and its distance from the reference quantifies the nonzero floor caused by finite-sample estimation. Operator and basis samples are never reused in this reference comparison.

Metrics. Let $\widetilde{\mathbf{F}}, \widetilde{\mathbf{H}} \in \mathbb{R}^{R_{\mathrm{ev}} \times N_x}$ denote standardized reference and candidate samples. We report:

1. the average pointwise empirical Wasserstein distance

$$\overline{W}_1 = \frac{1}{N_x}\sum_{j=1}^{N_x}\frac{1}{R_{\mathrm{ev}}}\sum_{r=1}^{R_{\mathrm{ev}}}\left|\widetilde{F}_{(r)j} - \widetilde{H}_{(r)j}\right|,$$

   together with its maximum over x_j;

2. the joint sliced Wasserstein distance

$$\mathrm{SW}_2 = \left[\frac{1}{P}\sum_{p=1}^{P} W_2^2\left(\widetilde{\mathbf{F}}\boldsymbol{\theta}_p, \widetilde{\mathbf{H}}\boldsymbol{\theta}_p\right)\right]^{1/2}, \qquad P = 512,$$

   where the $\boldsymbol{\theta}_p$ are random unit directions over the complete discretized function;

3. the RMSE between sample means, $e_\mu = N_x^{-1/2}\|\widehat{\boldsymbol{\mu}}_F - \widehat{\boldsymbol{\mu}}_H\|_2$, and relative covariance error $e_{\mathbf{K}} = \|\hat{\mathbf{K}}_F - \hat{\mathbf{K}}_H\|_F / \|\hat{\mathbf{K}}_F\|_F$;

4. the energy distance and a three-bandwidth RBF $\mathrm{MMD}^2_{\mathrm{RBF}}$, computed on the first 512 evaluation samples to bound their quadratic cost.

All metrics are lower-is-closer. Tables report means and standard errors across the five experiment seeds. Figure 5 visualizes representative paths at the published settings, and Table 6 reports all default-setting discrepancies.

At the published settings, GMVIP reduces SW_2 by 71.4% and average marginal W_1 by 76.9% relative to VIP. Its joint and marginal distances are close to the corresponding true-split floors: 0.072 versus 0.064 for SW_2, and 0.049 versus 0.045 for $\overline{W}_1$. The maximum pointwise discrepancy is also near the floor (0.069 versus 0.066), whereas VIP reaches 0.312. Figure 6 shows that this is not an average hiding a localized failure: GMVIP tracks the split baseline across the full input interval, while VIP remains substantially farther away, especially away from the center.

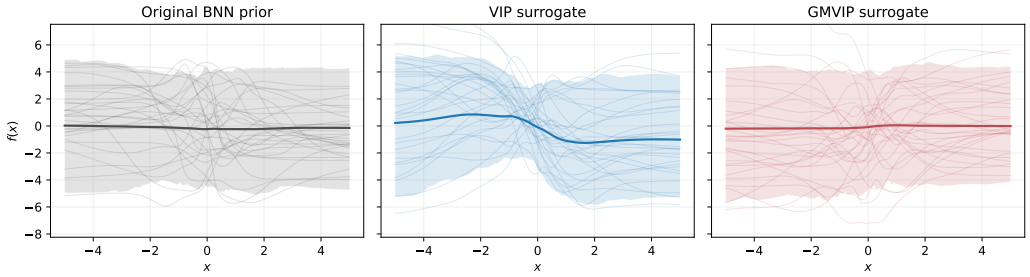


Figure 5: Original and surrogate prior paths for the published synthetic configurations. Thin curves are sampled functions, thick curves are sample means, and shaded regions are pointwise 90% intervals. VIP uses $S = 20$; GMVIP uses $M = 256$ inducing inputs and $B = 1024$ operator-bank paths. The VIP mean and path distribution visibly depend on the particular finite basis, whereas the GMVIP residual paths retain the qualitative variability of the original BNN prior.

Table 6: Surrogate-prior fidelity at the published synthetic configurations (↓). Entries are mean ± standard error across five seeds. The true-prior split is the finite-sample noise floor rather than a competing method. Bold marks the closer of the two nontrivial surrogates.

Distribution	Configuration	SW_2	$\overline{W}_1$	$\max_x W_1$	e_μ	$e_{\mathbf{K}}$	Energy	$\mathrm{MMD}^2_{\mathrm{RBF}}$
True-prior split	$R_{\mathrm{ev}} = 2048$	0.064 ± 0.004	0.045 ± 0.005	0.066 ± 0.007	0.034 ± 0.008	0.056 ± 0.010	0.0032 ± 0.0003	0.0011 ± 0.0001
VIP surrogate	$S = 20$	0.253 ± 0.032	0.214 ± 0.030	0.312 ± 0.029	0.195 ± 0.030	0.369 ± 0.073	0.0317 ± 0.0088	0.0117 ± 0.0028
GMVIP surrogate	$M = 256,\ B = 1024$	$\mathbf{0.072 \pm 0.004}$	$\mathbf{0.049 \pm 0.003}$	$\mathbf{0.069 \pm 0.006}$	$\mathbf{0.040 \pm 0.005}$	$\mathbf{0.073 \pm 0.009}$	$\mathbf{0.0056 \pm 0.0001}$	$\mathbf{0.0022 \pm 0.0001}$

Matched coefficient dimension. To separate the published configuration choice from the pathwise construction, we match the finite coefficient dimension at $S = M \in \{8, 20, 32, 64, 128, 256\}$, holding the GMVIP operator bank fixed at $B = 1024$. This matches the dimension of the standard-normal coefficient, not total sampling cost: GMVIP additionally retains a fresh residual prior path by design. Table 7 reports the two core distances, while Figure 7 shows their dimension dependence.

GMVIP is closer on both core metrics at every matched dimension. Its SW_2 advantage ranges from 77.4% at dimension 8 to 27.9% at dimension 256; the corresponding $\overline{W}_1$ reductions range from 80.5% to 33.5%. Notably, GMVIP with only $M = 8$ has lower joint and marginal distances than VIP with $S = 256$. VIP improves consistently with basis size, as expected from its empirical moment approximation, but its Gaussian finite-basis law remains detectably farther from this non-Gaussian BNN prior. Figure 8 compares the corresponding mean and covariance errors.

The moment diagnostics agree with the distributional distances. At the published settings, the GMVIP mean error is 0.040, compared with 0.034 for the true split and 0.195 for VIP. Its relative covariance error is 0.073, compared with 0.056 for the split and 0.369 for VIP. Thus the empirical results support the population second-order preservation result while also quantifying finite-bank estimation error. Figure 9 provides the energy-distance and kernel-MMD robustness checks.

The independent robustness metrics lead to the same ordering. At the published settings, the energy distance is 0.0056 for GMVIP, 0.0317 for VIP, and 0.0032 for the true split. The corresponding $\mathrm{MMD}^2_{\mathrm{RBF}}$ values are 0.0022, 0.0117, and 0.0011. The agreement across marginal, projected-joint, moment, energy, and kernel metrics makes the conclusion insensitive to any single discrepancy measure.

Operator-bank sensitivity. Finally, we fix $M = 256$ and vary the number of prior paths used to estimate the empirical operator. Table 8 and Figure 10 show that the discrepancy is not strictly monotone for the smaller banks: $B = 512$ is worse than $B = 256$ on this five-seed experiment. This is consistent with finite-bank and covariance stabilization variability rather than a deterministic monotonicity guarantee. The larger banks improve sharply. At $B = 2048$, $\mathrm{SW}_2 = 0.066$ and $\overline{W}_1 = 0.043$, which are at the scale of the true-split floors 0.064 and 0.045.

Interpretation and limitations. The experiment isolates prior fidelity from posterior optimization and predictive performance. It shows that the residual-path construction makes the empirical GMVIP

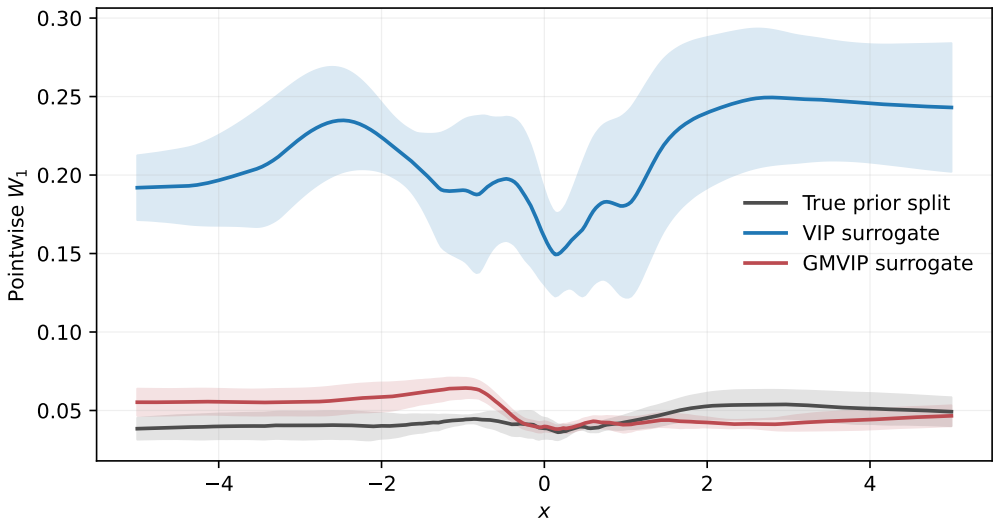


Figure 6: Pointwise marginal W_1 for the published VIP and GMVIP configurations. Curves and bands show mean $\pm$ standard error across five seeds. The GMVIP surrogate follows the true-prior split floor across $[-5, 5]$; the VIP finite-basis surrogate has a much larger discrepancy throughout the interval.

Table 7: Core distances under matched coefficient dimension (↓). GMVIP uses $B = 1024$. Entries are mean $\pm$ standard error across five seeds; bold marks the smaller distance in each pair.

Coefficient dim.	SW_2		$\overline{W}_1$	
$S = M$	VIP	GMVIP	VIP	GMVIP
8	0.367 ± 0.043	$\mathbf{0.083 \pm 0.005}$	0.303 ± 0.039	$\mathbf{0.059 \pm 0.007}$
20	0.253 ± 0.032	$\mathbf{0.078 \pm 0.004}$	0.214 ± 0.030	$\mathbf{0.055 \pm 0.007}$
32	0.189 ± 0.037	$\mathbf{0.077 \pm 0.003}$	0.144 ± 0.031	$\mathbf{0.049 \pm 0.001}$
64	0.144 ± 0.018	$\mathbf{0.081 \pm 0.005}$	0.103 ± 0.014	$\mathbf{0.054 \pm 0.006}$
128	0.119 ± 0.022	$\mathbf{0.074 \pm 0.006}$	0.086 ± 0.024	$\mathbf{0.049 \pm 0.005}$
256	0.100 ± 0.005	$\mathbf{0.072 \pm 0.004}$	0.074 ± 0.007	$\mathbf{0.049 \pm 0.003}$

surrogate substantially closer to the original BNN prior than the VIP finite-basis surrogate, and that most of the remaining detectable gap at the published setting is reduced by enlarging the operator bank. However, the true-split floor prevents interpreting a distance numerically close to that floor as proof of equality. The measurements concern one non-Gaussian BNN prior on one 301-point grid; they do not establish equality of infinite-dimensional process laws or uniform behavior for every implicit prior. They instead provide a direct finite-dimensional complement to Proposition 5 and empirical support for the second-order statement in Proposition 4.

D.2 Classification Experiments

LeNet classification. Table 9 summarizes this first setting. We first evaluate image classification on FashionMNIST and CIFAR-10 using a LeNet-style architecture with deterministic convolutional features and Bayesian classifier layers. We report NLL, classification error, ECE, and Brier score over five random seeds. This setting tests whether the proposed function-space posterior improves predictive uncertainty and accuracy while keeping the architecture lightweight enough for repeated Bayesian comparisons. GMVIP is consistently competitive in this Bayesian-head LeNet regime: on FashionMNIST it attains the best classification error and Brier score, and is tied for the best rounded NLL with FBNN. On CIFAR-10, SIP obtains the strongest NLL, error, and Brier score, but GMVIP remains among the best methods, with the second-best error and performance close to FBNN on NLL and Brier score. MFVI gives the lowest ECE, although this comes with worse predictive accuracy and likelihood on several metrics. These results indicate that GMVIP provides a favorable

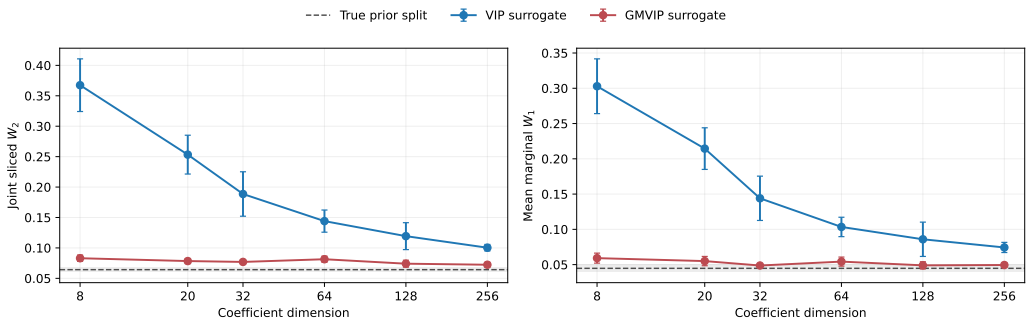


Figure 7: Joint sliced W_2 and average marginal W_1 against matched coefficient dimension. Error bars are standard errors across five seeds; the dashed line and gray band show the true-prior split mean and standard error. VIP approaches the original prior as its sampled-function basis grows, whereas GMVIP is near the finite-sample floor throughout the sweep.

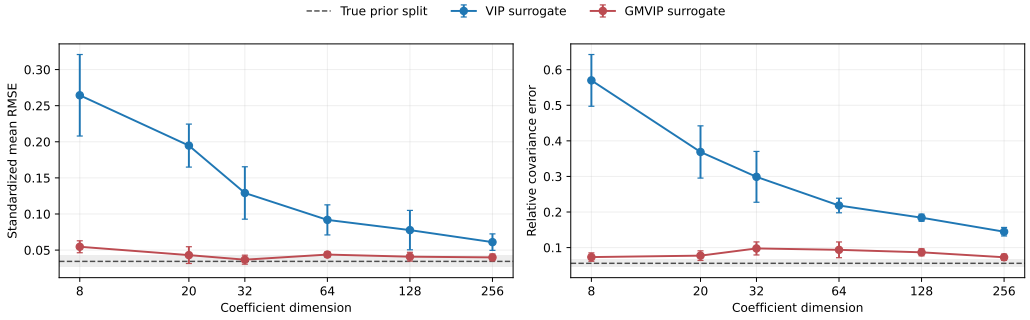


Figure 8: Standardized mean RMSE and relative covariance error under matched coefficient dimension. GMVIP remains close to the finite-sample floor because it combines an empirical prior-covariance correction with a fresh residual prior path. VIP moment errors decrease with basis size but remain larger over the tested range.

accuracy–uncertainty trade-off in the Bayesian-head LeNet setting, particularly on FashionMNIST, whereas CIFAR-10 is more challenging and favors SIP under this representation.

Frozen-CLIP protocol. We additionally evaluate all eight methods with linear classifier heads over frozen, normalized 512-dimensional CLIP ViT-B/32 embeddings. These experiments separate uncertainty in the classifier from representation learning: the encoder is never updated, and each method differs only in its linear head and posterior construction. The shared benchmark fixes the learning rate at 10^{-3} for every method. Every method is otherwise tuned independently, the selected head is retrained with 3 seeds, and Bayesian predictions average 100 posterior samples. A positive scalar temperature is fitted on a held-out calibration split after training. Temperature scaling does not change the predicted class, so the reported accuracy is unchanged, while NLL and ECE are temperature calibrated.

Frozen-CLIP CIFAR-10. For CIFAR-10, the 50 000 training images are divided into 40 000 tuning examples, 5 000 model-selection examples, and 5 000 temperature-calibration examples. After model selection, each winning head is retrained on the combined 45 000 fitting examples and evaluated once on the untouched 10 000-image test set. Table 10 shows that GMVIP is strongest overall. It obtains 94.45% accuracy, NLL 0.1667, and ECE 0.0058, giving the best accuracy and NLL and the second-best ECE. VIP ranks second in accuracy (94.42%) and NLL (0.1700), while SIP has the lowest ECE (0.0057). GMVIP gives the strongest accuracy–uncertainty trade-off on frozen-CLIP CIFAR-10.

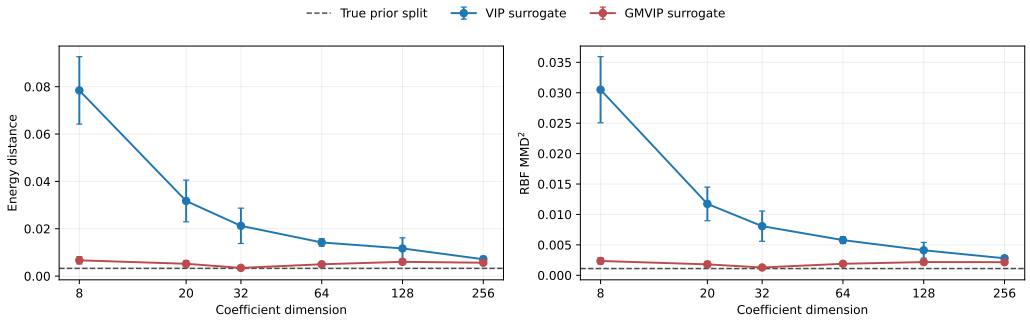


Figure 9: Energy distance and multi-bandwidth RBF $\mathrm{MMD}^2_{\mathrm{RBF}}$ under matched coefficient dimension. These metrics use a deterministic 512-sample prefix. They reproduce the Wasserstein conclusion: GMVIP stays close to the true-split floor, while VIP requires a much larger basis to reduce the discrepancy.

Table 8: GMVIP operator-bank sensitivity at $M = 256$ (↓). Entries are mean ± standard error across five seeds.

Bank B	SW_2	$\overline{W}_1$	e_μ	$e_{\mathbf{K}}$	Energy	$\mathrm{MMD}^2_{\mathrm{RBF}}$
256	0.093 ± 0.013	0.065 ± 0.011	0.058 ± 0.017	0.105 ± 0.020	0.0083 ± 0.0018	0.0028 ± 0.0006
512	0.106 ± 0.008	0.083 ± 0.009	0.071 ± 0.011	0.116 ± 0.022	0.0081 ± 0.0018	0.0029 ± 0.0006
1024	0.072 ± 0.004	0.049 ± 0.003	0.040 ± 0.005	0.073 ± 0.009	0.0056 ± 0.0001	0.0022 ± 0.0001
2048	$\mathbf{0.066 \pm 0.005}$	$\mathbf{0.043 \pm 0.005}$	$\mathbf{0.035 \pm 0.007}$	$\mathbf{0.059 \pm 0.011}$	$\mathbf{0.0044 \pm 0.0011}$	$\mathbf{0.0015 \pm 0.0005}$

Frozen-CLIP DynaSent. We next consider three-class sentiment classification on DynaSent. CLIP text embeddings are fitted with linear heads using 80 488 Round-1 training sentences for tuning and one stratified half of the Round-1 development set (1 800 examples) for model selection. The selected head is retrained on the resulting 82 288 examples, while the other 1 800 development examples are reserved for temperature calibration. We evaluate on the 3 600-example Round-1 test set (in distribution) and the 720-example Round-2 Dynabench test set (shifted). In addition to accuracy, NLL, and ECE, we report area under the risk–coverage curve (AURC), for which lower values mean that predictive confidence ranks correct examples ahead of errors more effectively.

Table 11 shows that GMVIP performs best on the Round-1 test set in accuracy, NLL, and AURC, attaining 59.48%, 0.8769, and 0.2503, respectively. Under the Round-2 shift, GMVIP retains the best NLL (0.9163) and AURC (0.3111), and has the second-best accuracy (57.41%) behind VIP (57.96%). SIP has the lowest shifted ECE, but only two of its three final seeds completed.

Taken together, Tables 9, 10, and 11 show that the value of a Bayesian classifier depends on the representation and data regime. GMVIP is strongest in the FashionMNIST LeNet comparison and is the strongest method under the common fixed-learning-rate frozen-CLIP CIFAR-10 protocol. On DynaSent, GMVIP provides the strongest overall in-distribution and shifted predictive performance. These results indicate that its function-space posterior is particularly useful when the representation retains label ambiguity, while still remaining effective with highly separable frozen image features.

D.3 Full UCI regression results

Table 12 completes the results of Table 1 by including CRPS and CQM metrics of the experiments.

D.4 Ablation Studies

We study the sensitivity of GMVIP to the initialization of the inducing locations, the number of inducing locations, and the number of prior functions used to estimate the operator moments. We use the nine UCI regression datasets and seeds 0–4, giving 45 runs per setting and 540 runs in total. Unless it is the quantity being varied, we use $M = 100$, a prior bank of $B = 512$, and k-means initialization. The inducing locations $\mathbf{Z}$ and the prior are subsequently optimized in every run; consequently, the geometry ablation measures sensitivity to the *initial* placement of $\mathbf{Z}$.

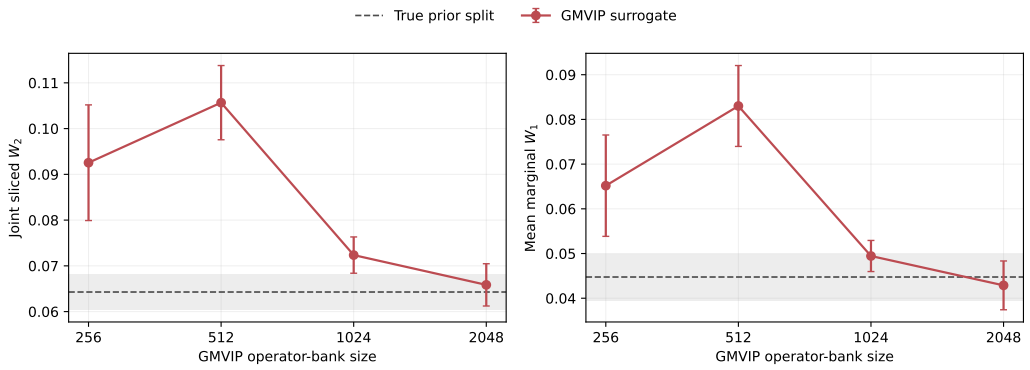


Figure 10: GMVIP joint and marginal prior discrepancy as a function of operator-bank size at $M = 256$. The dashed line and gray band denote the true-prior split floor. At $B = 2048$, the error bars overlap the floor at the resolution of these sample-based diagnostics, although the smaller-bank trend is non-monotone.

Table 9: Bayesian-head LeNet classification metrics (↓). Entries are mean ± standard deviation across five seeds. Best results are shown in **purple**; second-best are shown in **teal**.

Metric	Dataset	MAP	MFVI	VIP	FBNN	SIP	FTIP	GMVIP
NLL	FashionMNIST	0.287 ± 0.012	0.296 ± 0.014	0.282 ± 0.015	$\mathbf{0.261 \pm 0.016}$	0.277 ± 0.009	$\mathbf{0.274 \pm 0.017}$	$\mathbf{0.261 \pm 0.015}$
	CIFAR10	1.099 ± 0.054	1.189 ± 0.023	1.190 ± 0.081	$\mathbf{1.093 \pm 0.065}$	$\mathbf{1.039 \pm 0.065}$	1.344 ± 0.109	$\mathbf{1.093 \pm 0.025}$
Error	FashionMNIST	0.102 ± 0.004	0.108 ± 0.006	0.099 ± 0.006	$\mathbf{0.093 \pm 0.005}$	0.094 ± 0.005	0.096 ± 0.005	$\mathbf{0.092 \pm 0.005}$
	CIFAR10	0.374 ± 0.018	0.424 ± 0.010	0.372 ± 0.012	0.350 ± 0.009	$\mathbf{0.342 \pm 0.010}$	0.374 ± 0.016	$\mathbf{0.348 \pm 0.010}$
ECE	FashionMNIST	0.020 ± 0.006	$\mathbf{0.016 \pm 0.003}$	$\mathbf{0.019 \pm 0.004}$	0.020 ± 0.005	0.024 ± 0.009	0.021 ± 0.007	0.021 ± 0.005
	CIFAR10	$\mathbf{0.069 \pm 0.015}$	$\mathbf{0.045 \pm 0.011}$	0.121 ± 0.023	0.105 ± 0.017	0.071 ± 0.033	0.165 ± 0.019	0.102 ± 0.004
Brier	FashionMNIST	0.148 ± 0.006	0.154 ± 0.008	0.144 ± 0.008	$\mathbf{0.134 \pm 0.007}$	0.138 ± 0.007	0.140 ± 0.008	$\mathbf{0.133 \pm 0.007}$
	CIFAR10	0.505 ± 0.021	0.553 ± 0.009	0.517 ± 0.022	$\mathbf{0.485 \pm 0.018}$	$\mathbf{0.472 \pm 0.017}$	0.543 ± 0.026	0.486 ± 0.010

Tables 13–15 report the full per-dataset results as mean ± standard deviation over the five seeds. Lower values are better for RMSE, NLL, CRPS, and the centered quantile metric (CQM). Within each table and dataset, the best mean is shown in purple and the second-best mean in teal.

Inducing geometry. Table 13. The k-means initialization is the strongest choice for predictive quality. It obtains the best mean RMSE on eight of nine datasets and the best NLL and CRPS on seven. The grid initialization exhibits a different trade-off: although its predictive metrics are generally weaker, it achieves the best CQM on six datasets. Thus, k-means is preferable when predictive accuracy is primary, whereas the grid initialization can improve marginal calibration.

Number of inducing locations. Table 14. Increasing the inducing set improves predictive performance over the tested range. The largest setting, $M = 128$, wins on eight datasets for RMSE and CRPS and on seven for the NLL metric. The improvement is already visible at $M = 64$, while $M \in \{16, 32\}$ generally underfits relative to the larger sets. Calibration does not follow the same trend: $M = 16$ obtains the best CQM on five datasets. The results therefore show a clear trade-off between accuracy and calibration rather than uniform dominance as M grows.

Prior bank size. Table 15. Performance is not monotone in the bank size. A moderate bank of $B = 256$ provides the best aggregate predictive results, with five RMSE wins and four wins for each of the NLL and CRPS metrics. Increasing the bank to $B = 512$ does not improve consistently across datasets. Smaller banks occasionally improve CQM—$B = 32$ and $B = 64$ obtain four and three calibration wins, respectively—but have substantially poorer predictive metrics. Overall, $B = 256$ gives the most consistent predictive performance among the evaluated bank sizes.

Residual-path ablation. We compare full GMVIP with an inducing-only variant that retains the same empirical mean, inducing covariance, replacement operator, variational family, likelihood, and optimization protocol, but removes the coherent residual prior path. Let $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_M)^\top$ denote

Table 10: Frozen-CLIP CIFAR-10 test results with linear classifier heads. Entries are mean ± standard deviation over three seeds. Accuracy is in percentage points; NLL and ECE are temperature calibrated. Best mean is shown in **purple** and second-best mean in **teal**.

Method	Accuracy (%) ↑	NLL ↓	ECE ↓
MAP	93.93 ± 0.02	0.1855 ± 0.0001	0.0092 ± 0.0001
MFVI	93.80 ± 0.06	0.1872 ± 0.0001	0.0088 ± 0.0005
VIP	$\mathbf{94.42 \pm 0.05}$	$\mathbf{0.1700 \pm 0.0008}$	0.0093 ± 0.0021
FBNN	93.94 ± 0.03	0.1851 ± 0.0001	0.0092 ± 0.0003
SIP	93.05 ± 0.03	0.2107 ± 0.0021	$\mathbf{0.0057 \pm 0.0011}$
TFSVI	93.94 ± 0.02	0.1851 ± 0.0001	0.0105 ± 0.0005
FTIP	94.19 ± 0.03	0.1741 ± 0.0016	0.0068 ± 0.0015
GMVIP	$\mathbf{94.45 \pm 0.02}$	$\mathbf{0.1667 \pm 0.0014}$	$\mathbf{0.0058 \pm 0.0008}$

Table 11: Frozen-CLIP DynaSent results with linear three-class heads. Round 1 is the in-distribution test set and Round 2 is the adversarially collected shifted test set. Entries are mean ± standard deviation; SIP has two completed seeds and all other methods have three. Accuracy is in percentage points; NLL and ECE are temperature calibrated. Lower is better except for accuracy. AURC is the area under the risk–coverage curve. Best and second-best means are highlighted as above.

	Round 1 (in distribution)				Round 2 (shifted)			
Method	Acc. (%) ↑	NLL ↓	ECE ↓	AURC ↓	Acc. (%) ↑	NLL ↓	ECE ↓	AURC ↓
MAP	57.61 ± 0.03	0.8985 ± 0.0001	0.0437 ± 0.0003	0.2706 ± 0.0002	55.74 ± 0.08	0.9335 ± 0.0001	0.0351 ± 0.0037	0.3262 ± 0.0002
MFVI	57.23 ± 0.13	0.9031 ± 0.0003	0.0433 ± 0.0004	0.2749 ± 0.0005	55.19 ± 0.08	0.9374 ± 0.0004	$\mathbf{0.0319 \pm 0.0004}$	0.3284 ± 0.0005
VIP	$\mathbf{59.30 \pm 0.21}$	$\mathbf{0.8796 \pm 0.0005}$	$\mathbf{0.0385 \pm 0.0006}$	$\mathbf{0.2530 \pm 0.0006}$	$\mathbf{57.96 \pm 0.29}$	$\mathbf{0.9209 \pm 0.0023}$	0.0372 ± 0.0035	$\mathbf{0.3168 \pm 0.0005}$
FBNN	58.48 ± 0.40	0.8888 ± 0.0027	$\mathbf{0.0388 \pm 0.0007}$	0.2615 ± 0.0030	56.81 ± 0.28	0.9261 ± 0.0011	0.0417 ± 0.0045	0.3205 ± 0.0005
SIP	57.21 ± 0.29	0.9073 ± 0.0017	0.0418 ± 0.0019	0.2779 ± 0.0020	55.83 ± 0.20	0.9385 ± 0.0020	$\mathbf{0.0296 \pm 0.0011}$	0.3274 ± 0.0009
TFSVI	57.60 ± 0.06	0.8985 ± 0.0001	0.0441 ± 0.0003	0.2707 ± 0.0002	55.69 ± 0.00	0.9335 ± 0.0000	0.0349 ± 0.0009	0.3264 ± 0.0002
FTIP	56.61 ± 0.93	0.9095 ± 0.0077	0.0432 ± 0.0056	0.2800 ± 0.0073	56.76 ± 0.70	0.9453 ± 0.0054	0.0345 ± 0.0036	0.3319 ± 0.0011
GMVIP	$\mathbf{59.48 \pm 0.37}$	$\mathbf{0.8769 \pm 0.0032}$	0.0403 ± 0.0029	$\mathbf{0.2503 \pm 0.0032}$	$\mathbf{57.41 \pm 0.21}$	$\mathbf{0.9163 \pm 0.0012}$	0.0365 ± 0.0042	$\mathbf{0.3111 \pm 0.0000}$

the inducing matrix and let

$$g(\mathbf{Z}) = \big(g(\mathbf{z}_1), \ldots, g(\mathbf{z}_M)\big)^\top.$$

For an input $\mathbf{x}$, the residual path is

$$\mathbf{r}_g(\mathbf{x}) = g(\mathbf{x}) - \hat{\boldsymbol{\mu}}(\mathbf{x}) - \boldsymbol{\Psi}_{\mathbf{Z}}(\mathbf{x})\left(g(\mathbf{Z}) - \hat{\boldsymbol{\mu}}(\mathbf{Z})\right).$$

The full and inducing-only models are therefore

$$f_{\text{full}}(\mathbf{x}) = \hat{\boldsymbol{\mu}}(\mathbf{x}) + \boldsymbol{\Psi}_{\mathbf{Z}}(\mathbf{x})\mathbf{D}\mathbf{a} + \mathbf{r}_g(\mathbf{x}), \qquad f_{\text{ind}}(\mathbf{x}) = \hat{\boldsymbol{\mu}}(\mathbf{x}) + \boldsymbol{\Psi}_{\mathbf{Z}}(\mathbf{x})\mathbf{D}\mathbf{a},$$

where $\mathbf{a}$ is independent of g and

$$q_\phi(\mathbf{a}) = \mathcal{N}(\mathbf{m}_a, \mathbf{S}_a), \qquad \mathbf{D}\mathbf{D}^\top = \hat{\mathbf{K}}(\mathbf{Z}, \mathbf{Z}).$$

Consequently,

$$\mathrm{Cov}[\mathbf{f}_{\text{full}}(\mathbf{X})] = \mathrm{Cov}[\mathbf{f}_{\text{ind}}(\mathbf{X})] + \mathrm{Cov}[\mathbf{r}_g(\mathbf{X})].$$

The inducing-only model is thus restricted to a rank-M stochastic representation and omits the residual uncertainty outside the inducing subspace.

The test results in Table 16 show that this restriction is adequate on some datasets: inducing-only GMVIP improves all four metrics on Naval, Power, and Protein. Their train–test NLL gaps also remain small; for example, inducing-only NLL changes from -7.59 on the Naval training set to -7.58 on the test set, and from 2.82 to 2.86 on Protein. On these datasets, the learned inducing subspace appears sufficient to represent both the predictive mean and uncertainty.

In contrast, the inducing-only model exhibits pronounced overfitting on Boston, Concrete, Wine Red, and Yacht. On Boston, it achieves a training RMSE of 0.05 and training NLL of -1.38, but its test RMSE and NLL deteriorate to 4.09 and 2083.56, respectively. Full GMVIP has a substantially smaller discrepancy, with train/test RMSE $0.84/3.32$ and train/test NLL $1.55/3.11$. Similar behavior occurs on Wine Red, where inducing-only train/test NLL changes from -0.87 to 26.66, and on Yacht, where it changes from -1.18 to 6.16. On Concrete, inducing-only fits the training data more closely than full GMVIP—training RMSE 1.66 versus 2.75—but generalizes substantially worse, with test RMSE 5.39 versus 4.24.

Table 12: UCI regression metrics (↓). Entries are mean ± standard deviation across seeds. Frozen-prior variants are omitted. Best mean per dataset is shown in **purple**; second-best mean is shown in **teal**. Naval RMSE and CRPS are reported in units of 10^{-4}.

RMSE	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine Red	Yacht
MAP	4.34 ± 1.41	5.45 ± 0.57	0.52 ± 0.05	0.082 ± 0.001	4.04 ± 1.97	3.94 ± 0.13	4.62 ± 0.07	0.69 ± 0.01	0.52 ± 0.20
MFVI	3.93 ± 0.70	6.23 ± 0.51	0.62 ± 0.09	0.091 ± 0.002	147.61 ± 0.90	4.12 ± 0.17	4.94 ± 0.08	$\mathbf{0.62 \pm 0.03}$	0.68 ± 0.10
VIP	6.49 ± 1.27	5.76 ± 0.97	0.58 ± 0.09	0.080 ± 0.003	2.59 ± 0.96	$\mathbf{3.76 \pm 0.07}$	4.37 ± 0.06	0.79 ± 0.05	0.84 ± 0.30
FBNN	4.17 ± 1.27	$\mathbf{5.21 \pm 0.52}$	$\mathbf{0.51 \pm 0.05}$	0.086 ± 0.001	3.95 ± 0.47	4.05 ± 0.16	4.81 ± 0.05	0.65 ± 0.04	$\mathbf{0.23 \pm 0.06}$
SIP	$\mathbf{3.22 \pm 0.99}$	5.30 ± 0.27	0.52 ± 0.04	$\mathbf{0.071 \pm 0.001}$	$\mathbf{1.48 \pm 0.23}$	3.84 ± 0.11	4.62 ± 0.05	0.66 ± 0.03	0.41 ± 0.15
TFSVI	6.39 ± 0.63	5.97 ± 0.85	0.59 ± 0.15	0.081 ± 0.002	3.95 ± 0.81	3.90 ± 0.13	4.61 ± 0.05	0.73 ± 0.02	0.83 ± 0.38
FTIP	6.68 ± 1.58	6.02 ± 1.08	0.55 ± 0.07	0.079 ± 0.002	3.36 ± 1.58	$\mathbf{3.77 \pm 0.11}$	$\mathbf{4.36 \pm 0.05}$	0.82 ± 0.03	1.17 ± 0.52
GMVIP	$\mathbf{3.32 \pm 0.97}$	$\mathbf{4.24 \pm 0.74}$	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.069 \pm 0.001}$	$\mathbf{1.36 \pm 0.19}$	3.78 ± 0.12	$\mathbf{4.32 \pm 0.04}$	$\mathbf{0.60 \pm 0.03}$	$\mathbf{0.32 \pm 0.11}$
NLL	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine Red	Yacht
MAP	7.62 ± 3.48	3.24 ± 0.17	$\mathbf{0.81 \pm 0.14}$	-1.08 ± 0.02	-6.14 ± 1.03	2.79 ± 0.03	2.95 ± 0.01	1.27 ± 0.04	16.18 ± 15.06
MFVI	$\mathbf{2.76 \pm 0.14}$	3.25 ± 0.08	0.94 ± 0.14	-0.98 ± 0.03	-2.80 ± 0.01	2.84 ± 0.04	3.02 ± 0.02	$\mathbf{0.94 \pm 0.04}$	1.08 ± 0.05
VIP	4245.41 ± 2526.37	4.80 ± 1.11	2.48 ± 0.60	-1.09 ± 0.04	-6.78 ± 0.49	$\mathbf{2.74 \pm 0.02}$	2.89 ± 0.01	3.16 ± 0.39	96.61 ± 53.56
FBNN	5.24 ± 2.45	$\mathbf{3.13 \pm 0.16}$	0.84 ± 0.22	-1.04 ± 0.01	-6.42 ± 0.11	2.82 ± 0.04	2.99 ± 0.01	1.08 ± 0.08	$\mathbf{0.30 \pm 0.84}$
SIP	3.27 ± 1.22	3.18 ± 0.09	0.84 ± 0.14	$\mathbf{-1.22 \pm 0.02}$	$\mathbf{-7.04 \pm 0.29}$	2.76 ± 0.03	2.95 ± 0.01	1.09 ± 0.07	5.65 ± 5.09
TFSVI	11.32 ± 2.79	3.38 ± 0.23	1.05 ± 0.44	-1.09 ± 0.03	-6.41 ± 0.22	2.78 ± 0.03	2.95 ± 0.01	1.29 ± 0.07	14.30 ± 10.07
FTIP	1789.45 ± 735.75	4.82 ± 1.21	2.13 ± 0.36	-1.11 ± 0.02	-6.47 ± 0.59	$\mathbf{2.75 \pm 0.03}$	$\mathbf{2.89 \pm 0.01}$	3.52 ± 0.45	56.66 ± 56.15
GMVIP	$\mathbf{3.11 \pm 1.03}$	$\mathbf{2.82 \pm 0.16}$	$\mathbf{0.75 \pm 0.09}$	$\mathbf{-1.26 \pm 0.02}$	$\mathbf{-7.12 \pm 0.04}$	2.75 ± 0.03	$\mathbf{2.88 \pm 0.01}$	$\mathbf{0.94 \pm 0.08}$	$\mathbf{0.18 \pm 0.22}$
CRPS	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine Red	Yacht
MAP	2.33 ± 0.43	3.00 ± 0.28	0.28 ± 0.02	0.046 ± 0.001	2.46 ± 1.52	2.17 ± 0.06	2.60 ± 0.05	0.39 ± 0.01	0.20 ± 0.05
MFVI	2.06 ± 0.22	3.48 ± 0.27	0.34 ± 0.06	0.051 ± 0.001	86.17 ± 0.58	2.29 ± 0.08	2.79 ± 0.04	$\mathbf{0.34 \pm 0.02}$	0.37 ± 0.04
VIP	4.70 ± 0.88	3.26 ± 0.48	0.33 ± 0.05	0.045 ± 0.002	1.50 ± 0.64	$\mathbf{2.06 \pm 0.05}$	2.43 ± 0.03	0.48 ± 0.02	0.40 ± 0.12
FBNN	2.31 ± 0.47	$\mathbf{2.86 \pm 0.24}$	$\mathbf{0.28 \pm 0.02}$	0.048 ± 0.001	2.09 ± 0.25	2.25 ± 0.08	2.71 ± 0.04	0.37 ± 0.02	$\mathbf{0.12 \pm 0.03}$
SIP	$\mathbf{1.69 \pm 0.39}$	2.86 ± 0.15	0.28 ± 0.02	$\mathbf{0.040 \pm 0.001}$	$\mathbf{1.04 \pm 0.23}$	2.10 ± 0.05	2.60 ± 0.03	0.37 ± 0.02	0.16 ± 0.04
TFSVI	3.83 ± 0.37	3.26 ± 0.44	0.33 ± 0.07	0.045 ± 0.001	2.14 ± 0.42	2.14 ± 0.06	2.58 ± 0.03	0.40 ± 0.02	0.35 ± 0.11
FTIP	4.92 ± 1.25	3.40 ± 0.65	0.32 ± 0.04	0.044 ± 0.001	2.03 ± 1.05	$\mathbf{2.06 \pm 0.05}$	$\mathbf{2.43 \pm 0.03}$	0.48 ± 0.02	0.56 ± 0.18
GMVIP	$\mathbf{1.79 \pm 0.37}$	$\mathbf{2.26 \pm 0.35}$	$\mathbf{0.28 \pm 0.02}$	$\mathbf{0.038 \pm 0.001}$	$\mathbf{0.92 \pm 0.07}$	2.07 ± 0.06	$\mathbf{2.41 \pm 0.02}$	$\mathbf{0.33 \pm 0.01}$	$\mathbf{0.16 \pm 0.04}$
CQM	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine Red	Yacht
MAP	0.235 ± 0.053	0.051 ± 0.009	0.049 ± 0.021	0.013 ± 0.009	0.136 ± 0.142	0.021 ± 0.010	0.010 ± 0.003	0.088 ± 0.015	0.099 ± 0.028
MFVI	$\mathbf{0.047 \pm 0.013}$	$\mathbf{0.029 \pm 0.004}$	$\mathbf{0.042 \pm 0.013}$	$\mathbf{0.011 \pm 0.003}$	$\mathbf{0.072 \pm 0.005}$	$\mathbf{0.017 \pm 0.007}$	$\mathbf{0.008 \pm 0.003}$	$\mathbf{0.037 \pm 0.016}$	0.229 ± 0.050
VIP	0.495 ± 0.003	0.165 ± 0.036	0.172 ± 0.033	0.020 ± 0.013	0.130 ± 0.062	0.023 ± 0.010	0.021 ± 0.001	0.217 ± 0.009	0.342 ± 0.052
FBNN	0.211 ± 0.063	$\mathbf{0.030 \pm 0.012}$	0.051 ± 0.017	$\mathbf{0.011 \pm 0.004}$	0.095 ± 0.013	$\mathbf{0.016 \pm 0.007}$	0.008 ± 0.004	0.055 ± 0.019	$\mathbf{0.048 \pm 0.017}$
SIP	$\mathbf{0.089 \pm 0.022}$	0.049 ± 0.021	$\mathbf{0.046 \pm 0.012}$	0.016 ± 0.005	0.214 ± 0.085	0.025 ± 0.008	$\mathbf{0.007 \pm 0.004}$	0.057 ± 0.026	$\mathbf{0.067 \pm 0.018}$
TFSVI	0.292 ± 0.023	0.069 ± 0.013	0.063 ± 0.013	0.013 ± 0.008	$\mathbf{0.088 \pm 0.079}$	0.023 ± 0.005	0.015 ± 0.004	0.075 ± 0.017	0.118 ± 0.072
FTIP	0.483 ± 0.008	0.159 ± 0.043	0.213 ± 0.043	0.016 ± 0.006	0.171 ± 0.070	0.022 ± 0.008	0.017 ± 0.004	0.209 ± 0.021	0.331 ± 0.055
GMVIP	0.119 ± 0.065	0.056 ± 0.031	0.077 ± 0.026	0.034 ± 0.010	0.244 ± 0.044	0.037 ± 0.010	0.031 ± 0.005	$\mathbf{0.046 \pm 0.012}$	0.198 ± 0.053

Table 13: Inducing-location initialization ablation on the nine UCI regression datasets (↓). Entries are mean ± standard deviation across five seeds. The best mean per dataset is shown in purple; the second-best is shown in teal. Naval RMSE and CRPS are reported in units of 10^{-4}.

RMSE	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval** ($\times 10^{-4}$)	**Power**	**Protein**	**Wine red**	**Yacht**
Grid	$\mathbf{3.26 \pm 0.55}$	4.75 ± 0.55	0.52 ± 0.06	0.071 ± 0.001	2.26 ± 0.53	3.82 ± 0.12	4.49 ± 0.03	$\mathbf{0.62 \pm 0.03}$	0.49 ± 0.27
Random subset	$\mathbf{3.22 \pm 0.85}$	$\mathbf{4.27 \pm 0.54}$	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.069 \pm 0.001}$	$\mathbf{1.45 \pm 0.34}$	$\mathbf{3.79 \pm 0.12}$	$\mathbf{4.32 \pm 0.04}$	0.62 ± 0.03	$\mathbf{0.41 \pm 0.15}$
k-means	3.32 ± 0.97	$\mathbf{4.24 \pm 0.74}$	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.069 \pm 0.001}$	$\mathbf{1.36 \pm 0.19}$	$\mathbf{3.78 \pm 0.12}$	$\mathbf{4.32 \pm 0.04}$	$\mathbf{0.60 \pm 0.03}$	$\mathbf{0.32 \pm 0.11}$
NLL	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval**	**Power**	**Protein**	**Wine red**	**Yacht**
Grid	$\mathbf{2.83 \pm 0.41}$	2.96 ± 0.12	0.76 ± 0.06	-1.23 ± 0.01	-6.36 ± 0.10	2.76 ± 0.03	2.92 ± 0.01	$\mathbf{0.98 \pm 0.07}$	0.96 ± 1.48
Random subset	$\mathbf{2.87 \pm 0.58}$	$\mathbf{2.84 \pm 0.12}$	$\mathbf{0.74 \pm 0.07}$	$\mathbf{-1.25 \pm 0.02}$	$\mathbf{-7.08 \pm 0.06}$	$\mathbf{2.75 \pm 0.03}$	$\mathbf{2.88 \pm 0.01}$	1.01 ± 0.10	$\mathbf{0.35 \pm 0.32}$
k-means	3.11 ± 1.03	$\mathbf{2.82 \pm 0.16}$	$\mathbf{0.75 \pm 0.09}$	$\mathbf{-1.26 \pm 0.02}$	$\mathbf{-7.12 \pm 0.04}$	$\mathbf{2.75 \pm 0.03}$	$\mathbf{2.88 \pm 0.01}$	$\mathbf{0.94 \pm 0.08}$	$\mathbf{0.18 \pm 0.22}$
CRPS	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval** ($\times 10^{-4}$)	**Power**	**Protein**	**Wine red**	**Yacht**
Grid	1.82 ± 0.23	2.56 ± 0.26	$\mathbf{0.28 \pm 0.02}$	0.040 ± 0.001	1.84 ± 0.25	2.09 ± 0.06	2.52 ± 0.02	0.34 ± 0.02	0.21 ± 0.08
Random subset	$\mathbf{1.73 \pm 0.35}$	$\mathbf{2.29 \pm 0.27}$	$\mathbf{0.28 \pm 0.02}$	$\mathbf{0.039 \pm 0.001}$	$\mathbf{0.96 \pm 0.13}$	$\mathbf{2.08 \pm 0.06}$	$\mathbf{2.41 \pm 0.02}$	$\mathbf{0.34 \pm 0.02}$	$\mathbf{0.18 \pm 0.05}$
k-means	$\mathbf{1.79 \pm 0.37}$	$\mathbf{2.26 \pm 0.35}$	$\mathbf{0.28 \pm 0.02}$	$\mathbf{0.038 \pm 0.001}$	$\mathbf{0.92 \pm 0.07}$	$\mathbf{2.07 \pm 0.06}$	$\mathbf{2.41 \pm 0.02}$	$\mathbf{0.33 \pm 0.01}$	$\mathbf{0.16 \pm 0.04}$
CQM	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval**	**Power**	**Protein**	**Wine red**	**Yacht**
Grid	$\mathbf{0.117 \pm 0.013}$	$\mathbf{0.043 \pm 0.010}$	0.085 ± 0.041	$\mathbf{0.031 \pm 0.011}$	0.301 ± 0.044	$\mathbf{0.036 \pm 0.011}$	$\mathbf{0.020 \pm 0.007}$	$\mathbf{0.045 \pm 0.029}$	$\mathbf{0.194 \pm 0.048}$
Random subset	0.122 ± 0.060	$\mathbf{0.050 \pm 0.028}$	$\mathbf{0.076 \pm 0.027}$	$\mathbf{0.034 \pm 0.010}$	$\mathbf{0.232 \pm 0.068}$	0.038 ± 0.011	0.033 ± 0.005	0.051 ± 0.016	$\mathbf{0.183 \pm 0.022}$
k-means	$\mathbf{0.119 \pm 0.065}$	0.056 ± 0.031	$\mathbf{0.077 \pm 0.026}$	0.034 ± 0.010	$\mathbf{0.244 \pm 0.044}$	$\mathbf{0.037 \pm 0.010}$	$\mathbf{0.031 \pm 0.005}$	$\mathbf{0.046 \pm 0.012}$	0.198 ± 0.053

The overfitting cases also have much larger coefficient-space regularization terms. The final inducing-coefficient KL increases from 176 to 481 on Boston, from 237 to 417 on Wine Red, and from 117 to 253 on Yacht when the residual path is removed. This indicates that the inducing-only posterior moves further from its prior in order to explain variation that would otherwise be carried by $r_g(\mathbf{x})$. Together with the omitted residual covariance, this produces excessively concentrated predictive distributions: point errors increase moderately, whereas test NLL can deteriorate by several orders of magnitude. Overall, the residual path acts both as an additional source of predictive uncertainty and as a structural regularizer against overfitting the finite-dimensional inducing representation.

The residual path is not uniformly beneficial, but on several datasets it acts as a structural regularizer that prevents severe overconfidence and poor test likelihood.

Frozen-prior ablation. We study whether learning the implicit BNN prior is necessary for VIP, FTIP, and GMVIP. For each method, we compare a tunable-prior variant with a matched frozen-prior variant. If $g_{\boldsymbol{\theta}}(\mathbf{x})$ denotes the BNN prior, the two treatments differ only in whether $\boldsymbol{\theta}$ is optimized:

$$\boldsymbol{\theta} = \boldsymbol{\theta}_0 \quad \text{(frozen prior)}, \qquad (\boldsymbol{\theta}, \boldsymbol{\vartheta}) \in \arg\max_{\boldsymbol{\theta}, \boldsymbol{\vartheta}} \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\vartheta}) \quad \text{(tunable prior)},$$

where $\boldsymbol{\vartheta}$ collects the remaining variational, likelihood, and, when applicable, inducing parameters. All other settings are matched, including data splits, model architecture, inducing configuration,

Table 14: Number of inducing locations ablation on the nine UCI regression datasets (↓). Entries are mean ± standard deviation across five seeds. The best mean per dataset is shown in purple; the second-best is shown in teal. Naval RMSE and CRPS are reported in units of 10^{-4}.

RMSE	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine red	Yacht
$M = 16$	3.50 ± 1.30	4.75 ± 0.44	0.51 ± 0.05	0.076 ± 0.002	1.98 ± 0.19	3.94 ± 0.12	4.57 ± 0.03	$\mathbf{0.61 \pm 0.03}$	$\mathbf{0.46 \pm 0.14}$
$M = 32$	$\mathbf{3.48 \pm 0.92}$	$\mathbf{4.39 \pm 0.54}$	0.52 ± 0.05	0.072 ± 0.001	1.76 ± 0.16	3.86 ± 0.14	4.47 ± 0.04	0.61 ± 0.03	0.60 ± 0.22
$M = 64$	$\mathbf{3.24 \pm 1.34}$	4.55 ± 0.40	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.070 \pm 0.001}$	$\mathbf{1.56 \pm 0.46}$	$\mathbf{3.84 \pm 0.12}$	$\mathbf{4.37 \pm 0.04}$	0.61 ± 0.04	0.58 ± 0.36
$M = 128$	3.51 ± 1.19	$\mathbf{4.27 \pm 0.73}$	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.068 \pm 0.001}$	$\mathbf{1.54 \pm 0.38}$	$\mathbf{3.76 \pm 0.12}$	$\mathbf{4.29 \pm 0.05}$	$\mathbf{0.60 \pm 0.02}$	$\mathbf{0.40 \pm 0.24}$
NLL	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval**	**Power**	**Protein**	**Wine red**	**Yacht**
$M = 16$	$\mathbf{3.40 \pm 1.77}$	2.94 ± 0.11	0.75 ± 0.05	-1.16 ± 0.03	-6.54 ± 0.06	2.79 ± 0.03	2.94 ± 0.01	$\mathbf{0.95 \pm 0.07}$	$\mathbf{0.46 \pm 0.16}$
$M = 32$	$\mathbf{3.31 \pm 1.18}$	$\mathbf{2.86 \pm 0.11}$	0.75 ± 0.06	-1.21 ± 0.01	-6.78 ± 0.06	2.77 ± 0.03	2.92 ± 0.01	0.98 ± 0.06	0.66 ± 0.42
$M = 64$	3.47 ± 2.29	2.91 ± 0.10	$\mathbf{0.74 \pm 0.06}$	$\mathbf{-1.24 \pm 0.01}$	$\mathbf{-6.99 \pm 0.08}$	$\mathbf{2.76 \pm 0.03}$	$\mathbf{2.89 \pm 0.01}$	0.99 ± 0.09	0.60 ± 0.55
$M = 128$	3.48 ± 1.72	$\mathbf{2.85 \pm 0.16}$	$\mathbf{0.74 \pm 0.08}$	$\mathbf{-1.28 \pm 0.01}$	$\mathbf{-7.11 \pm 0.09}$	$\mathbf{2.74 \pm 0.03}$	$\mathbf{2.87 \pm 0.01}$	$\mathbf{0.95 \pm 0.05}$	$\mathbf{0.46 \pm 0.86}$
CRPS	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval** ($\times 10^{-4}$)	**Power**	**Protein**	**Wine red**	**Yacht**
$M = 16$	1.90 ± 0.66	2.52 ± 0.23	0.28 ± 0.02	0.042 ± 0.001	1.55 ± 0.10	2.17 ± 0.06	2.56 ± 0.01	$\mathbf{0.34 \pm 0.01}$	$\mathbf{0.22 \pm 0.04}$
$M = 32$	1.91 ± 0.36	$\mathbf{2.34 \pm 0.24}$	0.28 ± 0.02	0.040 ± 0.001	1.25 ± 0.08	2.13 ± 0.07	2.51 ± 0.02	0.34 ± 0.02	0.25 ± 0.07
$M = 64$	$\mathbf{1.65 \pm 0.43}$	2.42 ± 0.19	$\mathbf{0.28 \pm 0.02}$	$\mathbf{0.039 \pm 0.000}$	$\mathbf{1.05 \pm 0.17}$	$\mathbf{2.11 \pm 0.06}$	$\mathbf{2.44 \pm 0.02}$	0.34 ± 0.02	0.24 ± 0.11
$M = 128$	$\mathbf{1.85 \pm 0.41}$	$\mathbf{2.31 \pm 0.35}$	$\mathbf{0.28 \pm 0.02}$	$\mathbf{0.038 \pm 0.000}$	$\mathbf{0.97 \pm 0.16}$	$\mathbf{2.06 \pm 0.05}$	$\mathbf{2.39 \pm 0.03}$	$\mathbf{0.33 \pm 0.01}$	$\mathbf{0.17 \pm 0.07}$
CQM	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval**	**Power**	**Protein**	**Wine red**	**Yacht**
$M = 16$	$\mathbf{0.133 \pm 0.071}$	0.047 ± 0.017	0.085 ± 0.031	$\mathbf{0.028 \pm 0.006}$	0.294 ± 0.014	$\mathbf{0.027 \pm 0.008}$	$\mathbf{0.017 \pm 0.005}$	$\mathbf{0.041 \pm 0.009}$	$\mathbf{0.168 \pm 0.034}$
$M = 32$	0.161 ± 0.062	0.050 ± 0.023	0.074 ± 0.027	$\mathbf{0.034 \pm 0.009}$	0.262 ± 0.022	$\mathbf{0.032 \pm 0.013}$	$\mathbf{0.020 \pm 0.007}$	0.047 ± 0.010	0.186 ± 0.037
$M = 64$	$\mathbf{0.107 \pm 0.034}$	$\mathbf{0.043 \pm 0.019}$	$\mathbf{0.070 \pm 0.029}$	0.034 ± 0.009	$\mathbf{0.258 \pm 0.058}$	0.034 ± 0.010	0.028 ± 0.007	0.055 ± 0.023	$\mathbf{0.184 \pm 0.043}$
$M = 128$	0.138 ± 0.055	$\mathbf{0.044 \pm 0.029}$	$\mathbf{0.071 \pm 0.026}$	0.036 ± 0.010	$\mathbf{0.203 \pm 0.053}$	0.038 ± 0.010	0.032 ± 0.004	$\mathbf{0.045 \pm 0.011}$	0.207 ± 0.020

Table 15: Prior-bank size ablation on the nine UCI regression datasets (↓). Entries are mean ± standard deviation across five seeds. The best mean per dataset is shown in purple; the second-best is shown in teal. Naval RMSE and CRPS are reported in units of 10^{-4}.

RMSE	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine red	Yacht
$B = 32$	4.15 ± 1.28	4.47 ± 0.89	0.51 ± 0.06	0.073 ± 0.003	$\mathbf{1.29 \pm 0.17}$	3.81 ± 0.12	4.36 ± 0.05	0.68 ± 0.01	0.39 ± 0.17
$B = 64$	3.71 ± 1.17	$\mathbf{4.37 \pm 0.47}$	0.51 ± 0.05	0.071 ± 0.001	$\mathbf{1.32 \pm 0.20}$	$\mathbf{3.78 \pm 0.12}$	4.33 ± 0.05	0.65 ± 0.02	0.44 ± 0.22
$B = 128$	3.86 ± 1.25	4.37 ± 0.70	0.52 ± 0.04	0.070 ± 0.002	1.39 ± 0.18	3.78 ± 0.14	$\mathbf{4.31 \pm 0.05}$	0.65 ± 0.01	0.76 ± 0.88
$B = 256$	$\mathbf{3.68 \pm 1.35}$	4.40 ± 0.60	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.069 \pm 0.001}$	1.52 ± 0.19	$\mathbf{3.77 \pm 0.13}$	$\mathbf{4.31 \pm 0.04}$	$\mathbf{0.64 \pm 0.03}$	$\mathbf{0.33 \pm 0.12}$
$B = 512$	$\mathbf{3.32 \pm 0.97}$	$\mathbf{4.24 \pm 0.74}$	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.069 \pm 0.001}$	1.36 ± 0.19	3.78 ± 0.12	4.32 ± 0.04	$\mathbf{0.60 \pm 0.03}$	$\mathbf{0.32 \pm 0.11}$
NLL	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval**	**Power**	**Protein**	**Wine red**	**Yacht**
$B = 32$	6.30 ± 4.87	2.94 ± 0.26	$\mathbf{0.72 \pm 0.10}$	-1.21 ± 0.03	$\mathbf{-7.29 \pm 0.08}$	2.76 ± 0.03	2.89 ± 0.01	1.34 ± 0.07	0.31 ± 0.31
$B = 64$	4.52 ± 2.89	2.86 ± 0.11	$\mathbf{0.72 \pm 0.08}$	-1.23 ± 0.02	$\mathbf{-7.26 \pm 0.06}$	2.75 ± 0.03	2.88 ± 0.01	1.21 ± 0.05	0.49 ± 0.20
$B = 128$	4.36 ± 2.58	2.88 ± 0.18	0.75 ± 0.06	-1.24 ± 0.03	-7.20 ± 0.03	$\mathbf{2.75 \pm 0.04}$	$\mathbf{2.88 \pm 0.01}$	1.16 ± 0.02	0.41 ± 0.62
$B = 256$	$\mathbf{3.75 \pm 1.84}$	$\mathbf{2.86 \pm 0.13}$	0.73 ± 0.07	$\mathbf{-1.26 \pm 0.01}$	-7.13 ± 0.06	$\mathbf{2.75 \pm 0.03}$	$\mathbf{2.88 \pm 0.01}$	$\mathbf{1.09 \pm 0.08}$	$\mathbf{0.19 \pm 0.31}$
$B = 512$	$\mathbf{3.11 \pm 1.03}$	$\mathbf{2.82 \pm 0.16}$	0.75 ± 0.09	$\mathbf{-1.26 \pm 0.02}$	-7.12 ± 0.04	2.75 ± 0.03	2.88 ± 0.01	$\mathbf{0.94 \pm 0.08}$	$\mathbf{0.18 \pm 0.22}$
CRPS	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval** ($\times 10^{-4}$)	**Power**	**Protein**	**Wine red**	**Yacht**
$B = 32$	2.32 ± 0.52	2.38 ± 0.38	$\mathbf{0.27 \pm 0.03}$	0.041 ± 0.001	$\mathbf{0.81 \pm 0.08}$	2.09 ± 0.05	2.43 ± 0.03	0.38 ± 0.01	0.19 ± 0.07
$B = 64$	$\mathbf{2.00 \pm 0.40}$	2.33 ± 0.25	0.28 ± 0.02	0.039 ± 0.001	$\mathbf{0.83 \pm 0.08}$	$\mathbf{2.07 \pm 0.06}$	2.41 ± 0.03	0.36 ± 0.00	0.20 ± 0.06
$B = 128$	2.09 ± 0.54	$\mathbf{2.31 \pm 0.31}$	0.28 ± 0.02	0.039 ± 0.001	0.88 ± 0.06	2.07 ± 0.07	$\mathbf{2.40 \pm 0.02}$	0.36 ± 0.01	0.28 ± 0.24
$B = 256$	2.01 ± 0.47	2.33 ± 0.29	$\mathbf{0.28 \pm 0.02}$	$\mathbf{0.038 \pm 0.000}$	0.96 ± 0.08	$\mathbf{2.07 \pm 0.06}$	$\mathbf{2.40 \pm 0.03}$	$\mathbf{0.35 \pm 0.02}$	$\mathbf{0.15 \pm 0.03}$
$B = 512$	$\mathbf{1.79 \pm 0.37}$	$\mathbf{2.26 \pm 0.35}$	0.28 ± 0.02	$\mathbf{0.038 \pm 0.001}$	0.92 ± 0.07	2.07 ± 0.06	2.41 ± 0.02	$\mathbf{0.33 \pm 0.01}$	$\mathbf{0.16 \pm 0.04}$
CQM	**Boston**	**Concrete**	**Energy**	**Kin8nm**	**Naval**	**Power**	**Protein**	**Wine red**	**Yacht**
$B = 32$	0.210 ± 0.062	0.046 ± 0.012	$\mathbf{0.056 \pm 0.028}$	$\mathbf{0.024 \pm 0.009}$	$\mathbf{0.200 \pm 0.029}$	$\mathbf{0.033 \pm 0.011}$	$\mathbf{0.029 \pm 0.008}$	0.104 ± 0.026	$\mathbf{0.171 \pm 0.064}$
$B = 64$	0.184 ± 0.041	$\mathbf{0.033 \pm 0.010}$	0.067 ± 0.027	$\mathbf{0.024 \pm 0.007}$	0.205 ± 0.036	$\mathbf{0.034 \pm 0.010}$	$\mathbf{0.028 \pm 0.007}$	0.076 ± 0.012	0.201 ± 0.061
$B = 128$	0.166 ± 0.052	0.042 ± 0.009	$\mathbf{0.066 \pm 0.026}$	0.024 ± 0.012	0.218 ± 0.039	0.036 ± 0.010	0.031 ± 0.007	0.082 ± 0.009	0.219 ± 0.040
$B = 256$	$\mathbf{0.163 \pm 0.021}$	$\mathbf{0.038 \pm 0.015}$	0.075 ± 0.025	0.030 ± 0.009	$\mathbf{0.204 \pm 0.038}$	0.037 ± 0.011	0.032 ± 0.007	$\mathbf{0.063 \pm 0.023}$	$\mathbf{0.194 \pm 0.032}$
$B = 512$	$\mathbf{0.119 \pm 0.065}$	0.056 ± 0.031	0.077 ± 0.026	0.034 ± 0.010	0.244 ± 0.044	0.037 ± 0.010	0.031 ± 0.005	$\mathbf{0.046 \pm 0.012}$	0.198 ± 0.053

Monte Carlo budgets, optimization schedule, and seeds. Table 17 reports RMSE, NLL, CRPS, and CQM over the nine UCI datasets.

Prior learning is especially important for GMVIP. As shown in Table 17, tunable-prior GMVIP outperforms its frozen counterpart on every dataset and every reported metric. The difference is substantial on Concrete, where RMSE increases from 4.24 to 7.34 when the prior is frozen, and on Energy, where it increases from 0.51 to 2.80. The largest degradation occurs on Yacht: RMSE increases from 0.32 to 4.11 and NLL from 0.18 to 3.28. These results indicate that the empirical GMVIP operator cannot, in general, compensate for a poorly adapted function prior merely by learning the inducing locations and coefficient posterior.

The behavior of VIP and FTIP is more nuanced. Their tunable-prior variants obtain lower test RMSE on eight of the nine datasets; the only exception is Wine Red, where freezing the prior reduces RMSE from 0.79 to 0.70 for VIP and from 0.82 to 0.70 for FTIP. The final training metrics nevertheless show clear overfitting on several datasets. On Boston, VIP attains train/test RMSEs of $0.055/6.49$ and train/test NLLs of $-1.41/4245.41$; FTIP exhibits the same failure, with RMSEs of $0.135/6.68$ and NLLs of $-0.40/1789.45$. On Yacht, the corresponding train/test gaps are $0.045/0.84$ in RMSE and $-1.62/96.61$ in NLL for VIP, and $0.100/1.17$ and $-0.84/56.66$ for FTIP. Wine Red shows a milder version of the same pattern. Thus, the extreme test NLLs are accompanied by near-interpolation of the training data and very favorable training NLLs. Learning the prior can make VIP and FTIP severely under-dispersed: the predictive distribution contracts around the training observations and assigns negligible mass to modest held-out residuals.

Table 16: GMVIP path ablation on UCI regression (↓). Entries are mean ± standard deviation across five seeds. The full-path model retains the residual prior path, whereas the inducing-only variant removes it. The best mean per dataset is shown in bold. Naval RMSE and CRPS are reported in units of 10^{-4}.

RMSE	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine Red	Yacht
Full	$\mathbf{3.32 \pm 0.97}$	$\mathbf{4.24 \pm 0.74}$	0.51 ± 0.05	$\mathbf{0.069 \pm 0.001}$	1.36 ± 0.19	3.78 ± 0.12	4.32 ± 0.04	$\mathbf{0.60 \pm 0.03}$	$\mathbf{0.32 \pm 0.11}$
Induc. only	4.09 ± 1.10	5.39 ± 0.65	$\mathbf{0.45 \pm 0.03}$	0.085 ± 0.003	$\mathbf{1.16 \pm 0.25}$	$\mathbf{3.72 \pm 0.08}$	$\mathbf{4.22 \pm 0.05}$	0.87 ± 0.05	0.46 ± 0.15
NLL	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine Red	Yacht
Full	$\mathbf{3.11 \pm 1.03}$	$\mathbf{2.82 \pm 0.16}$	$\mathbf{0.75 \pm 0.09}$	$\mathbf{-1.26 \pm 0.02}$	-7.12 ± 0.04	2.75 ± 0.03	2.88 ± 0.01	$\mathbf{0.94 \pm 0.08}$	$\mathbf{0.18 \pm 0.22}$
Induc. only	2083.56 ± 1408.57	5.57 ± 1.00	1.49 ± 0.43	-1.02 ± 0.04	$\mathbf{-7.58 \pm 0.12}$	$\mathbf{2.73 \pm 0.02}$	$\mathbf{2.86 \pm 0.01}$	26.66 ± 2.95	6.16 ± 5.43
CRPS	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine Red	Yacht
Full	$\mathbf{1.79 \pm 0.37}$	$\mathbf{2.26 \pm 0.35}$	0.28 ± 0.02	$\mathbf{0.038 \pm 0.001}$	0.92 ± 0.07	2.07 ± 0.06	2.41 ± 0.02	$\mathbf{0.33 \pm 0.01}$	$\mathbf{0.16 \pm 0.04}$
Induc. only	2.86 ± 0.40	3.06 ± 0.27	$\mathbf{0.24 \pm 0.02}$	0.048 ± 0.002	$\mathbf{0.64 \pm 0.11}$	$\mathbf{2.02 \pm 0.04}$	$\mathbf{2.33 \pm 0.03}$	0.56 ± 0.03	0.20 ± 0.06
CQM	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine Red	Yacht
Full	$\mathbf{0.119 \pm 0.065}$	$\mathbf{0.056 \pm 0.031}$	$\mathbf{0.077 \pm 0.026}$	$\mathbf{0.034 \pm 0.010}$	0.244 ± 0.044	0.037 ± 0.010	0.031 ± 0.005	$\mathbf{0.046 \pm 0.012}$	0.198 ± 0.053
Induc. only	0.483 ± 0.009	0.167 ± 0.011	0.139 ± 0.037	0.040 ± 0.011	$\mathbf{0.166 \pm 0.041}$	$\mathbf{0.021 \pm 0.008}$	$\mathbf{0.029 \pm 0.004}$	0.302 ± 0.012	$\mathbf{0.109 \pm 0.024}$

Table 17: Frozen-prior ablation for VIP, FTIP, and GMVIP on UCI regression (↓). Entries are mean ± standard deviation across five matched seeds. Tunable variants optimize the BNN prior jointly with the remaining model parameters, whereas frozen variants keep the prior fixed at initialization. The best mean per dataset among the compared variants is shown in bold. Naval RMSE and CRPS are reported in units of 10^{-4}.

RMSE	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine Red	Yacht
VIP (tunable)	6.49 ± 1.27	5.76 ± 0.97	0.58 ± 0.09	0.080 ± 0.003	2.59 ± 0.96	$\mathbf{3.76 \pm 0.07}$	4.37 ± 0.06	0.79 ± 0.05	0.84 ± 0.30
VIP (frozen)	6.70 ± 1.26	11.79 ± 0.90	4.97 ± 0.59	0.234 ± 0.007	96.20 ± 8.64	7.01 ± 0.94	5.52 ± 0.05	0.70 ± 0.04	10.79 ± 2.21
FTIP (tunable)	6.68 ± 1.58	6.02 ± 1.08	0.55 ± 0.07	0.079 ± 0.002	3.36 ± 1.58	3.77 ± 0.11	4.36 ± 0.05	0.82 ± 0.03	1.17 ± 0.52
FTIP (frozen)	6.71 ± 1.27	11.77 ± 0.86	4.97 ± 0.60	0.234 ± 0.006	90.54 ± 9.39	7.02 ± 0.93	5.53 ± 0.06	0.70 ± 0.04	10.80 ± 2.22
GMVIP (tunable)	$\mathbf{3.32 \pm 0.97}$	$\mathbf{4.24 \pm 0.74}$	$\mathbf{0.51 \pm 0.05}$	$\mathbf{0.069 \pm 0.001}$	$\mathbf{1.36 \pm 0.19}$	3.78 ± 0.12	$\mathbf{4.32 \pm 0.04}$	$\mathbf{0.60 \pm 0.03}$	$\mathbf{0.32 \pm 0.11}$
GMVIP (frozen)	3.86 ± 0.97	7.34 ± 0.45	2.80 ± 0.13	0.144 ± 0.005	4.10 ± 0.56	4.01 ± 0.11	4.64 ± 0.05	0.60 ± 0.03	4.11 ± 1.07
NLL	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine Red	Yacht
VIP (tunable)	4245.41 ± 2526.37	4.80 ± 1.11	2.48 ± 0.60	-1.09 ± 0.04	-6.78 ± 0.49	$\mathbf{2.74 \pm 0.02}$	2.89 ± 0.01	3.16 ± 0.39	96.61 ± 53.56
VIP (frozen)	3.34 ± 0.17	3.89 ± 0.07	3.02 ± 0.12	-0.04 ± 0.03	-3.23 ± 0.09	3.36 ± 0.13	3.13 ± 0.01	1.07 ± 0.05	3.83 ± 0.17
FTIP (tunable)	1789.45 ± 735.75	4.82 ± 1.21	2.13 ± 0.36	-1.11 ± 0.02	-6.47 ± 0.59	2.75 ± 0.03	2.89 ± 0.01	3.52 ± 0.45	56.66 ± 56.15
FTIP (frozen)	3.34 ± 0.18	3.89 ± 0.07	3.02 ± 0.12	-0.03 ± 0.03	-3.29 ± 0.10	3.36 ± 0.13	3.13 ± 0.01	1.07 ± 0.05	3.84 ± 0.17
GMVIP (tunable)	$\mathbf{3.11 \pm 1.03}$	$\mathbf{2.82 \pm 0.16}$	$\mathbf{0.75 \pm 0.09}$	$\mathbf{-1.26 \pm 0.02}$	$\mathbf{-7.12 \pm 0.04}$	2.75 ± 0.03	$\mathbf{2.88 \pm 0.01}$	$\mathbf{0.94 \pm 0.08}$	$\mathbf{0.18 \pm 0.22}$
GMVIP (frozen)	3.62 ± 0.02	4.04 ± 0.03	3.45 ± 0.01	0.34 ± 0.00	-5.15 ± 0.02	3.42 ± 0.01	3.11 ± 0.00	1.46 ± 0.00	3.28 ± 0.03
CRPS	Boston	Concrete	Energy	Kin8nm	Naval ($\times 10^{-4}$)	Power	Protein	Wine Red	Yacht
VIP (tunable)	4.70 ± 0.88	3.26 ± 0.48	0.33 ± 0.05	0.045 ± 0.002	1.50 ± 0.64	$\mathbf{2.06 \pm 0.05}$	2.43 ± 0.03	0.48 ± 0.02	0.40 ± 0.12
VIP (frozen)	3.74 ± 0.62	6.67 ± 0.50	2.78 ± 0.38	0.132 ± 0.004	53.79 ± 5.37	3.94 ± 0.54	3.15 ± 0.03	0.39 ± 0.02	6.10 ± 1.02
FTIP (tunable)	4.92 ± 1.25	3.40 ± 0.65	0.32 ± 0.04	0.044 ± 0.001	2.03 ± 1.05	2.06 ± 0.05	2.43 ± 0.03	0.48 ± 0.02	0.56 ± 0.18
FTIP (frozen)	3.75 ± 0.64	6.66 ± 0.47	2.78 ± 0.38	0.132 ± 0.004	50.24 ± 5.36	3.95 ± 0.54	3.15 ± 0.03	0.39 ± 0.02	6.10 ± 1.02
GMVIP (tunable)	$\mathbf{1.79 \pm 0.37}$	$\mathbf{2.26 \pm 0.35}$	$\mathbf{0.28 \pm 0.02}$	$\mathbf{0.038 \pm 0.001}$	$\mathbf{0.92 \pm 0.07}$	2.07 ± 0.06	$\mathbf{2.41 \pm 0.02}$	$\mathbf{0.33 \pm 0.01}$	$\mathbf{0.16 \pm 0.04}$
GMVIP (frozen)	3.82 ± 0.20	6.09 ± 0.14	3.16 ± 0.02	0.143 ± 0.001	5.69 ± 0.14	3.27 ± 0.03	2.79 ± 0.02	0.46 ± 0.00	2.90 ± 0.21
CQM	Boston	Concrete	Energy	Kin8nm	Naval	Power	Protein	Wine Red	Yacht
VIP (tunable)	0.495 ± 0.003	0.165 ± 0.036	0.172 ± 0.033	0.020 ± 0.013	0.130 ± 0.062	0.023 ± 0.010	0.021 ± 0.001	0.217 ± 0.009	0.342 ± 0.052
VIP (frozen)	0.054 ± 0.028	$\mathbf{0.025 \pm 0.009}$	$\mathbf{0.034 \pm 0.020}$	$\mathbf{0.009 \pm 0.004}$	$\mathbf{0.026 \pm 0.022}$	$\mathbf{0.014 \pm 0.003}$	0.031 ± 0.009	$\mathbf{0.034 \pm 0.015}$	0.069 ± 0.038
FTIP (tunable)	0.483 ± 0.008	0.159 ± 0.043	0.213 ± 0.043	0.016 ± 0.006	0.171 ± 0.070	0.022 ± 0.008	$\mathbf{0.017 \pm 0.004}$	0.209 ± 0.021	0.331 ± 0.055
FTIP (frozen)	$\mathbf{0.053 \pm 0.028}$	0.026 ± 0.009	0.035 ± 0.023	0.010 ± 0.005	0.035 ± 0.022	0.014 ± 0.003	0.033 ± 0.011	0.034 ± 0.015	$\mathbf{0.068 \pm 0.039}$
GMVIP (tunable)	0.119 ± 0.065	0.056 ± 0.031	0.077 ± 0.026	0.034 ± 0.010	0.244 ± 0.044	0.037 ± 0.010	0.031 ± 0.005	0.046 ± 0.012	0.198 ± 0.053
GMVIP (frozen)	0.353 ± 0.014	0.305 ± 0.013	0.371 ± 0.007	0.340 ± 0.005	0.407 ± 0.008	0.293 ± 0.005	0.144 ± 0.005	0.278 ± 0.013	0.310 ± 0.037

Freezing the prior removes this train–test discrepancy, but typically by underfitting both splits. For example, frozen VIP has train/test RMSEs of $6.49/6.70$ on Boston and $11.73/10.79$ on Yacht, with nearly identical FTIP values. Consequently, freezing reduces the Yacht test NLL to 3.83 for VIP and 3.84 for FTIP, but increases test RMSE to 10.79 and 10.80, respectively. The frozen variants also obtain lower CQM on eight datasets. Their improved NLL and CQM should therefore not be read as uniformly better predictions: they often reflect much more diffuse distributions with poor point accuracy. Table 17 exposes a sharp trade-off for VIP and FTIP between overfit, overconfident tunable priors and underfit, overly diffuse frozen priors, motivating the joint interpretation of NLL and CQM with RMSE and CRPS.

Finally, frozen VIP and frozen FTIP produce nearly identical results: across the matched runs, their mean absolute differences are only 0.013 in RMSE and 0.009 in NLL. Thus, under the frozen BNN prior used here, the additional FTIP flow provides little measurable benefit over VIP. Overall, Table 17 shows that prior adaptation is essential for GMVIP and generally important for the point-prediction accuracy of VIP and FTIP.

D.5 VIP and FTIP Basis-Size Sensitivity on Lotka–Volterra

To test whether the forecasting comparison is limited by the finite sampled-function basis, we rerun VIP and FTIP with $S \in \{20, 64, 128, 256\}$ on the same 20 target trajectories and evaluation protocol as Appendix C.3. The bases are nested within a common master prior bank, and all other method-specific settings are held fixed. Each coefficient-space method receives 800 total optimization steps:

Table 18: Sensitivity of VIP and FTIP to the number S of sampled prior basis functions in the Lotka–Volterra experiment. Entries are means $\pm$ standard errors across 20 target trajectories on the test interval $t \in (20, 30]$. VIP is trained for 800 iterations. FTIP is warm-started from a separate 400-iteration VIP fit at the same S and then fine-tuned for 400 additional iterations. GMVIP uses 96 inducing locations and is included as a reference. Coverage is ranked by proximity to the nominal 90% level; lower is better otherwise. Best and second-best results, determined using unrounded values, are highlighted.

Method	Representation size	RMSE ↓	NLL ↓	CRPS ↓	Cov_{90}	ODE residual ↓
VIP	$S = 20$	0.50 ± 0.08	11.32 ± 5.01	0.31 ± 0.05	0.72 ± 0.06	0.65 ± 0.03
VIP	$S = 64$	0.37 ± 0.08	6.20 ± 5.87	0.23 ± 0.05	0.87 ± 0.04	0.69 ± 0.02
VIP	$S = 128$	0.36 ± 0.09	7.93 ± 7.62	0.23 ± 0.06	$\mathbf{0.89 \pm 0.04}$	0.71 ± 0.03
VIP	$S = 256$	0.35 ± 0.09	9.97 ± 9.71	$\mathbf{0.22 \pm 0.06}$	0.89 ± 0.04	0.69 ± 0.03
FTIP	$S = 20$	0.57 ± 0.08	1.04 ± 0.22	0.34 ± 0.05	0.82 ± 0.04	$\mathbf{0.62 \pm 0.03}$
FTIP	$S = 64$	0.43 ± 0.13	1.80 ± 1.41	0.29 ± 0.10	0.92 ± 0.04	0.67 ± 0.02
FTIP	$S = 128$	0.37 ± 0.10	1.22 ± 0.84	0.25 ± 0.07	0.95 ± 0.04	0.68 ± 0.02
FTIP	$S = 256$	$\mathbf{0.33 \pm 0.08}$	$\mathbf{0.72 \pm 0.37}$	0.23 ± 0.06	0.95 ± 0.03	0.68 ± 0.02
GMVIP	$M = 96$	$\mathbf{0.32 \pm 0.06}$	$\mathbf{0.51 \pm 0.29}$	$\mathbf{0.20 \pm 0.04}$	$\mathbf{0.90 \pm 0.04}$	$\mathbf{0.31 \pm 0.02}$

Table 19: GPU efficiency diagnostics on Airline. Setup includes model construction and method-specific preparation; train time is mean wall-clock time per measured optimizer step after warmup. Memory reports peak CUDA allocated/reserved memory.

Method	Setup	Train / step	Eval time	CUDA memory	Latent dim.	Prior samples
VIP	0.04s	0.006s	0.247s	28 / 48 MB	20	–
FTIP	0.16s	0.023s	0.216s	84 / 112 MB	20	512
FBNN	0.03s	0.044s	0.247s	176 / 290 MB	512	512
TFSVI	4.09s	0.067s	0.232s	249 / 378 MB	512	–
SIP	2.34s	0.021s	0.357s	626 / 672 MB	100	512
GMVIP	1.17s	0.031s	0.489s	626 / 672 MB	100	512

VIP is trained directly for 800 steps, while FTIP uses a separate 400-step VIP warm start followed by 400 fine-tuning steps. Every run is evaluated at its final scheduled iterate, without validation or checkpoint selection, using the same 1,024-sample evaluation protocol. Because S counts global sampled-function basis elements, whereas GMVIP uses inducing values, this experiment is a within-method capacity sensitivity analysis rather than a direct equality of representation dimensions.

The results in Table 18 show that increasing S generally improves point forecasting, but does not yield monotone gains in every metric. GMVIP has the lowest mean RMSE, NLL, CRPS, and ODE residual among the learned configurations, together with coverage closest to the nominal 0.90. FTIP at $S = 256$ is second in RMSE and NLL, while VIP at $S = 128$ is second in coverage.

D.6 Efficiency and capacity diagnostics

To contextualize the empirical gains, Table 19 reports GPU wall-clock setup time, per-step training time, evaluation time, peak CUDA memory, latent or coefficient dimension, and the number of prior samples used by each method on Airline; on a NVIDIA RTX 5070. This diagnostic is especially important for empirical GMVIP and SIP: both use inducing-variable representations with 100 inducing variables and 512 prior samples, while empirical GMVIP additionally constructs a Matheron operator from prior-bank evaluations and empirical covariance estimates. The table therefore separates predictive performance from the computational capacity and overhead required to obtain it.