

Three Costs of Amortizing Gaussian Process Inference with Neural Processes

Robin Young

University of Cambridge, Cambridge, UK

Abstract

Neural processes amortize Gaussian process inference, replacing the exact $O(n^3)$ posterior with a learned $O(n)$ map from context sets to predictive distributions. For a class of latent neural processes, we bound the Kullback–Leibler (KL) divergence between the GP and LNP predictives, decomposing it into three interpretable sources, namely label contamination as the neural process uses label values to estimate a quantity that is label-independent in the exact GP, an information bottleneck because the finite-dimensional representation cannot resolve the full context geometry, and amortization error from a single encoder network shared across all contexts. The bottleneck truncation term decays in the representation dimension d as $O(e^{-cd^{2/d_x}})$ for squared-exponential kernels on $\mathbb{R}^{d_x}$ where $c > 0$ is a kernel-dependent constant and as $O(d^{-2\nu/d_x})$ for Matérn- ν kernels, directly linking architecture sizing to kernel smoothness and input dimension. The label contamination term is $O(1)$ in general, with only the observation-noise component decaying as $O(1/n)$, identifying a persistent cost of routing uncertainty estimation through a label-dependent representation. These results characterize the costs of amortization within the analyzed class and yield architectural recommendations to predict variance from context locations alone in the GP-amortization regime, and replace mean aggregation with second-order pooling to close the dominant amortization gap.

Proceedings of the 2nd International Conference on Probabilistic Numerics (ProbNum) 2026, Lappeenranta, Finland. PMLR Volume 341. Copyright 2026 with the author(s)

1 Introduction

Gaussian processes (Rasmussen and Williams, 2005) define exact posterior distributions over functions given observed data. The posterior mean and variance are available in closed form, but computing them costs $O(n^3)$ in the number of context points n^1 , limiting scalability.

Neural processes (Garnelo et al., 2018a,b) address this by learning an amortized map from context sets to predictive distributions. Subsequent work has extended the framework through attention mechanisms (Kim et al., 2019), convolutional structure (Gordon et al., 2020), and autoregressive factorizations (Bruinsma et al., 2023). A latent neural process (LNP) maps each context pair through an encoder, aggregates the resulting features into a finite-dimensional representation, passes this representation through a recognition network producing a latent distribution, and decodes samples from the latent into a predictive distribution, all in $O(n)$ time for fixed architecture. When trained on samples from a GP prior, the LNP is implicitly approximating the GP posterior.

Sparse variational methods (Titsias, 2009) offer one path to scaling GP inference, reducing the cubic cost to $O(M^2n)$ by summarizing the posterior through M inducing points. However, the resulting predictive still requires $O(M^2)$ computation at test time, and the inducing locations must be optimized jointly with kernel hyperparameters for each new dataset. In settings that demand predictions across many related tasks or under real-time constraints such as sequential

experimental design (Shahriari et al., 2016), robotics (Deisenroth et al., 2015), or simulation-based inference (Cranmer et al., 2020) this per-task optimization becomes a bottleneck of its own. Neural processes eliminate per-task optimization entirely with a single forward pass mapping any context set to a predictive distribution in $O(n)$ time, with no matrix inversion and no task-specific parameters. This amortization is the source of their practical appeal, but it introduces approximation costs that have no counterpart in the sparse GP framework.

It is known that this approximation introduces error as the variational inference underlying the LNP tends to underestimate uncertainty, and the finite-dimensional bottleneck cannot capture arbitrary context geometries. However, no quantitative characterization of these errors exists. Foong et al. (2020) identify qualitative limitations of mean aggregation, showing it leads to underfitting, but do not provide rates. People choose representation dimensions, encoder architectures, and variance parameterizations without formal guidance on the resulting approximation quality. In the sparse GP literature, Burt et al. (2020) bound the KL divergence between sparse variational and exact GP posteriors, showing that $M = O((\log N)^{d_x})$ inducing points suffice for squared-exponential kernels. Our bottleneck analysis plays an analogous role for neural processes, with the representation dimension d replacing the number of inducing points M .

We provide such a characterization. Our main contributions are: (i) a decomposition of the predictive KL divergence $\text{KL}(p_{\text{GP}} \| p_{\text{LNP}})$ into three terms with distinct architectural and statistical origins; (ii) an upper bound on the truncation component of the bottleneck term that decays exponentially in d for SE kernels and

¹Specifically, $O(n^3)$ for the one-time Cholesky factorization of $K + \sigma_e^2 I$, subsequent predictions cost $O(n^2)$ per query.

polynomially for Matérn kernels, connecting representation dimension to kernel smoothness; (iii) characterization of the label contamination term as $O(1) + O(1/n)$, identifying a structural mismatch in how neural processes estimate predictive variance; and (iv) two architectural recommendations to predict variance from locations alone in the GP-amortization regime, and use second-order rather than mean aggregation.

2 Setup

2.1 Gaussian Process

Let $f \sim \mathcal{GP}(0, k)$ be a zero-mean GP with kernel $k : \mathcal{X} \times \mathcal{X} \rightarrow \mathbb{R}$ on a compact domain $\mathcal{X} \subset \mathbb{R}^{d_x}$. Given context $C = \{(x_i, y_i)\}_{i=1}^n$ with $y_i = f(x_i) + \epsilon_i$, $\epsilon_i \sim \mathcal{N}(0, \sigma_\epsilon^2)$, the GP predictive at a target x_* is:

$$p_{\text{GP}}(y_* \mid x_*, C) = \mathcal{N}(\mu_{\text{GP}}, \sigma_{\text{GP}}^2) \quad (1)$$

where

$$\mu_{\text{GP}} = \mathbf{k}_*^\top (K + \sigma_\epsilon^2 I)^{-1} \mathbf{y}, \quad \mathbf{y} = (y_1, \dots, y_n)^\top \quad (2)$$

$$\sigma_{\text{GP}}^2 = k(x_*, x_*) - \mathbf{k}_*^\top (K + \sigma_\epsilon^2 I)^{-1} \mathbf{k}_*, \quad (3)$$

with $[\mathbf{k}_*]_i = k(x_*, x_i)$ and $[K]_{ij} = k(x_i, x_j)$.

A structural property is that the GP predictive variance (3) depends on the context locations $X = \{x_i\}$ but not on the labels $\mathbf{y}$.

2.2 Latent Neural Process

Definition 1 (Latent Neural Process). *A latent neural process with mean aggregation consists of:*

1. An encoder $h : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^d$ mapping context pairs to representations.
2. Mean aggregation: $r_C = \frac{1}{n} \sum_{i=1}^n h(x_i, y_i)$.
3. A latent encoder: $q(z \mid C) = \mathcal{N}(\mu_z(r_C), \Sigma_z(r_C))$ with $z \in \mathbb{R}^{d_z}$.
4. A decoder: $p(y_* \mid x_*, z) = \mathcal{N}(w(x_*)^\top z + b(x_*), \sigma_d^2)$.

The maps $\mu_z : \mathbb{R}^d \rightarrow \mathbb{R}^{d_z}$ and $\Sigma_z : \mathbb{R}^d \rightarrow \mathbb{R}_{>0}^{d_z \times d_z}$ are typically MLPs with the covariance output parameterized via a Cholesky factor to enforce positive definiteness. The functions $w : \mathcal{X} \rightarrow \mathbb{R}^{d_z}$ and $b : \mathcal{X} \rightarrow \mathbb{R}$ are learned (typically MLP-parameterized) functions of the target location, not network weights and biases. The linear-in- z decoder is a tractability choice that yields a closed-form Gaussian posterior $p(z \mid C)$ amenable to the amortization-gap analysis of [Section 3.3](#); standard NPs use nonlinear MLP decoders. See [Appendix A](#) for a fuller introduction to NPs in case they are unfamiliar.

The marginal predictive is Gaussian:

$$p_{\text{LNP}}(y_* \mid x_*, C) = \mathcal{N}(\mu_{\text{LNP}}, \sigma_{\text{LNP}}^2), \quad (4)$$

$$\mu_{\text{LNP}} = w(x_*)^\top \mu_z(r_C) + b(x_*), \quad (5)$$

$$\sigma_{\text{LNP}}^2 = w(x_*)^\top \Sigma_z(r_C) w(x_*) + \sigma_d^2. \quad (6)$$

Unlike (3), the LNP variance (6) depends on r_C , which encodes both context locations and labels.

2.3 Predictive KL Divergence

Both predictives are Gaussian, so the KL divergence has a closed form:

$$\text{KL}(p_{\text{GP}} \parallel p_{\text{LNP}}) = \frac{1}{2} \left[\frac{\sigma_{\text{GP}}^2}{\sigma_{\text{LNP}}^2} - 1 + \frac{(\mu_{\text{GP}} - \mu_{\text{LNP}})^2}{\sigma_{\text{LNP}}^2} + \log \frac{\sigma_{\text{LNP}}^2}{\sigma_{\text{GP}}^2} \right] \quad (7)$$

We bound the expected gap $\mathbb{E}_C[\text{KL}(p_{\text{GP}} \parallel p_{\text{LNP}})]$ where the expectation is over contexts drawn from the GP prior.

Assumption 1. *There exist constants $0 < \sigma_\ell^2 \leq \sigma_u^2 < \infty$ such that $\sigma_\ell^2 \leq \sigma_{\text{LNP}}^2(x_*; C) \leq \sigma_u^2$ and $\sigma_\ell^2 \leq \sigma_{\text{GP}}^2(x_*; X) \leq \sigma_u^2$ for all contexts C , locations X , and targets $x_* \in \mathcal{X}$.*

Remark 1. *The GP side holds automatically with $\sigma_\ell^2 = \sigma_\epsilon^2 \kappa_k / (\kappa_k + \sigma_\epsilon^2)$ and $\sigma_u^2 = \kappa_k$ (where $\kappa_k = \sup_{x \in \mathcal{X}} k(x, x)$ bounds the prior signal variance), since adding context points can only reduce posterior variance from the prior. The LNP side is enforced architecturally by requiring $\sigma_d^2 \geq \sigma_\ell^2$ in the decoder and bounding the weights $\|w(x_*)\|$ and the eigenvalues of Σ_z . This assumption is used in the variance ratio expansion in [Theorem 3](#).*

3 Decomposition of the Variance Error

The variance gap is the primary source of the variational inference (VI) bias. We decompose it into three sources. Define the variance error as $\Delta\sigma^2 = \sigma_{\text{LNP}}^2 - \sigma_{\text{GP}}^2$.

3.1 Label Contamination

The GP variance $\sigma_{\text{GP}}^2(x_*; X)$ is a function of context locations only. The LNP variance $\sigma_{\text{LNP}}^2(x_*; r_C)$ depends on labels through r_C . This mismatch introduces noise in the variance estimate.

Assumption 2. *The encoder is affine in y : $h(x, y) = \phi(x) + \psi(x)y$, with positive constants B_ϕ, B_ψ such that $\|\phi(x)\| \leq B_\phi$ and $\|\psi(x)\| \leq B_\psi$ for all $x \in \mathcal{X}$.*

Under this assumption, the representation decomposes as:

$$r_C = \underbrace{\frac{1}{n} \sum_{i=1}^n \phi(x_i)}_{\bar{\phi}_X} + \underbrace{\frac{1}{n} \sum_{i=1}^n \psi(x_i) y_i}_{\delta_y}. \quad (8)$$

The first term depends only on locations; the second introduces label dependence.

Assumption 3. *The maps $\mu_z : \mathbb{R}^d \rightarrow \mathbb{R}^{d_z}$ and $\Sigma_z : \mathbb{R}^d \rightarrow \mathbb{R}^{d_z \times d_z}$ are L_μ -Lipschitz and L_Σ -Lipschitz respectively. The decoder satisfies $\|w(x_*)\| \leq B_w$.*

Theorem 1 (Label Contamination Bound). *Under Assumptions 2 and 3, for contexts drawn from a GP prior*

with kernel k and noise variance σ_ϵ^2 :

$$\mathbb{E}_C[\text{Var}_{\mathbf{y}|X}[\sigma_{\text{LNP}}^2(x_*; C)]] \leq L_\Sigma^2 B_w^4 B_\psi^2 \left(\frac{\sigma_\epsilon^2}{n} + \kappa_k \right) \quad (9)$$

where $\kappa_k = \sup_{x \in \mathcal{X}} k(x, x)$ bounds the signal variance. The first term arises from observation noise and decays as $O(1/n)$; the second arises from correlations under the GP prior and is $O(1)$.

Proof sketch. Decompose the label-dependent component as $\delta_y = \delta_f + \delta_\epsilon$ where $\delta_f = \frac{1}{n} \sum_i \psi(x_i) f(x_i)$ and $\delta_\epsilon = \frac{1}{n} \sum_i \psi(x_i) \epsilon_i$. The noise term is an i.i.d. sum, so $\mathbb{E}[\|\delta_\epsilon\|^2 | X] \leq B_\psi^2 \sigma_\epsilon^2 / n$. The signal term has covariance $\frac{1}{n^2} \sum_{i,j} \psi(x_i) \psi(x_j)^\top k(x_i, x_j)$. Since k is positive definite with bounded diagonal κ_k , the row-sum bound gives $\mathbb{E}[\|\delta_f\|^2 | X] \leq B_\psi^2 \kappa_k$, which is $O(1)$ because the GP-correlated $f(x_i)$ do not concentrate. Lipschitz continuity of Σ_z then transfers these bounds to σ_{LNP}^2 . Full proof in [Appendix B.1](#). $\square$

Remark 2. Only the noise component decays as $O(1/n)$. The signal component is $O(1)$ because the $f(x_i)$ are correlated under the GP prior. $\delta_f = \frac{1}{n} \sum_i \psi(x_i) f(x_i)$ converges to $\mathbb{E}_\mu[\psi(x) f(x)]$, which is a nondegenerate random variable across GP draws. This $O(1)$ term is irreducible within architectures that route uncertainty estimation through a label-dependent bottleneck, and persists even as $n \rightarrow \infty$.

3.2 Information Bottleneck

Even removing label dependence (conditioning on $\mathbf{y}$ or taking $\bar{\phi}_X$ as the representation), the d -dimensional summary $\bar{\phi}_X$ cannot capture the full context geometry that determines the GP variance.²

Definition 2 (Mercer Expansion). Let $\{(\lambda_j, e_j)\}_{j=1}^\infty$ be the eigendecomposition of the kernel integral operator T_k with respect to a base measure μ on $\mathcal{X}$, taken throughout to coincide with the distribution from which context locations are drawn:

$$k(x, x') = \sum_{j=1}^\infty \lambda_j e_j(x) e_j(x'). \quad (10)$$

Assumption 4. The location encoder uses the first d kernel eigenfunctions $\phi(x) = (\sqrt{\lambda_1} e_1(x), \dots, \sqrt{\lambda_d} e_d(x))^\top$.

This is the optimal d -dimensional encoder in the sense of minimizing integrated squared error in kernel approximation. The mean representation becomes $\bar{\phi}_X = \frac{1}{n} \sum_i \phi(x_i)$, which captures the projection of the empirical context measure onto the first d kernel eigenfunctions.

Assumption 5. The representation dimension d is large enough that the tail operator norm satisfies

$$\eta(X) = \|A^{-1/2} K_T A^{-1/2}\|_{\text{op}} < 1, \quad (11)$$

²The truncation component of this bottleneck error follows from standard Mercer approximation theory. We include it as the baseline against which the additional NP-specific costs (label contamination, amortization gap) are measured. These additional costs have no analogue in sparse variational GP theory.

where $A = K_H + \sigma_\epsilon^2 I$ with head kernel matrix $[K_H]_{il} = \sum_{j \leq d} \lambda_j e_j(x_i) e_j(x_l)$, and tail kernel matrix $[K_T]_{il} = \sum_{j > d} \lambda_j e_j(x_i) e_j(x_l)$.

Remark 3. Since $A \succeq \sigma_\epsilon^2 I$, the condition $\eta < 1$ is implied by $\|K_T\|_{\text{op}} < \sigma_\epsilon^2$. For i.i.d. context points, $\mathbb{E}[\|K_T\|_{\text{op}}] \leq n \lambda_T$ where $\lambda_T = \sum_{j > d} \lambda_j$, so the assumption holds (with high probability) once $n \lambda_T < \sigma_\epsilon^2$. For SE kernels this requires $d = \Omega((\log n)^{d_x/2})$ ³ for Matérn- ν kernels, $d = \Omega(n^{d_x/(2\nu)})$. This is the natural regime in which the d -truncated GP is a meaningful approximation: the tail kernel contributes less than the observation noise.

Theorem 2 (Information Bottleneck Bound). Under Assumptions 4 and 5, for a target x_* and context locations $X = \{x_i\}_{i=1}^n$ drawn i.i.d. from μ :

$$\mathbb{E}_X \left[(\sigma_{\text{GP}}^2(x_*; X) - g_d^*(\bar{\phi}_X))^2 \right] \leq R_d(k, n) + S_d(k, x_*), \quad (12)$$

where g_d^* is the optimal estimator of σ_{GP}^2 from $\bar{\phi}_X$, and:

$$R_d(k, n) = O\left(\frac{d^2}{n}\right) + R_d^{\text{info}}, \quad (13)$$

$$S_d(k, x_*) \leq C_S \sum_{j > d} \lambda_j e_j(x_*)^2, \quad (14)$$

with $C_S = C \kappa_k / \sigma_\epsilon^4$ for a constant C depending only on $\kappa_k / \sigma_\epsilon^2$. Here R_d^{info} is the irreducible estimation loss from $\bar{\phi}_X$ being an insufficient statistic for the head kernel matrix, and the $O(d^2/n)$ term is the variance of $\bar{\phi}_X$ about its population mean. For second-order aggregation (using $\frac{1}{n} \sum_i \phi(x_i) \phi(x_i)^\top$ in place of $\bar{\phi}_X$), $R_d^{\text{info}} = 0$ and the estimation error is $O(d^2/n)$.

Proof sketch. Split the Mercer series into head ($j \leq d$) and tail ($j > d$) components, with $A = K_H + \sigma_\epsilon^2 I$ and $M = A + K_T$. The truncated GP variance $\sigma_d^2 = k_H - (\mathbf{k}_*^H)^\top A^{-1} \mathbf{k}_*^H$ depends only on head eigenfunction evaluations. Since the optimal estimator $g_d^*(\bar{\phi}_X)$ minimizes MSE over measurable functions of $\bar{\phi}_X$, and σ_d^2 is measurable with respect to a finer σ -algebra, S_d is bounded by $\mathbb{E}_X[(\sigma_{\text{GP}}^2 - \sigma_d^2)^2]$. A Neumann expansion of M^{-1} around A^{-1} , valid under [Assumption 5](#), gives $|\sigma_{\text{GP}}^2 - \sigma_d^2| = O(\sqrt{k_T})$, hence the squared error is $O(k_T)$ with constant $C_S = C \kappa_k / \sigma_\epsilon^4$. The estimation term R_d contains an $O(d^2/n)$ CLT contribution from $\bar{\phi}_X$ converging to its population mean, plus the irreducible R_d^{info} from mean aggregation being insufficient for the head kernel matrix; under second-order aggregation $R_d^{\text{info}} = 0$. Full proof in [Appendix B.2](#). $\square$

Remark 4 (Scope of the bound). [Theorem 2](#) bounds the variance error of the Bayes-optimal estimator $g_d^*(\bar{\phi}_X)$ given the mean representation; this characterizes the information-theoretic limit of any estimator built on $\bar{\phi}_X$. The cost of replacing g_d^* with the learned amortized estimator is captured separately in [Section 3.3](#). The truncation term S_d is fully controlled and inherits the kernel's eigenvalue decay ([Corollary 1](#)). The estimation term R_d contains a controlled $O(d^2/n)$ component plus an irreducible information loss R_d^{info} .

³Throughout, $\Omega(\cdot)$ and $\Theta(\cdot)$ denote standard Bachmann-Landau lower-bound and tight-bound notation respectively.

that we do not bound under mean aggregation; under second-order aggregation $R_d^{\text{info}} = 0$.

The truncation error S_d inherits the eigenvalue decay of the kernel:

Corollary 1 (Kernel-Dependent Bottleneck Rates). *Under Assumptions 4 and 5:*

1. **Squared-exponential kernel** on $[0, 1]^{d_x}$: Eigenvalues satisfy $\lambda_j \leq C_{\text{SE}} e^{-c j^{2/d_x}}$ for constants $C_{\text{SE}}, c > 0$ depending on the lengthscale ℓ . Thus:

$$S_d \leq C'_{\text{SE}} e^{-c' d^{2/d_x}}. \quad (15)$$

2. **Matérn- ν kernel** on $[0, 1]^{d_x}$: Eigenvalues satisfy $\lambda_j \leq C_\nu j^{-(2\nu+d_x)/d_x}$. Thus:

$$S_d \leq C'_\nu d^{-2\nu/d_x}. \quad (16)$$

The constants C'_{SE} and C'_ν absorb the factor $C_S = C \kappa_k / \sigma_\epsilon^4$ from [Theorem 2](#).

Proof. For the SE kernel, the tail sum $\sum_{j>d} \lambda_j e_j(x_*)^2 \leq \|e\|_\infty^2 \sum_{j>d} \lambda_j$. The eigenvalues of the SE kernel on compact domains decay as $\lambda_j \leq C e^{-c j^{2/d_x}}$ ([Steinwart and Scovel, 2012](#)). Thus $\sum_{j>d} \lambda_j \leq C e^{-c d^{2/d_x}}$. Substituting into the linear truncation bound gives the result.

For the Matérn kernel, the eigenvalue asymptotics $\lambda_j = \Theta(j^{-(2\nu+d_x)/d_x})$ follow from the spectral theory of Sobolev-equivalent reproducing kernels ([Wendland, 2004](#)). Then $\sum_{j>d} \lambda_j \leq C \int_d^\infty t^{-(2\nu+d_x)/d_x} dt = C' d^{-2\nu/d_x}$. Substituting into the linear truncation bound gives $S_d = O(d^{-2\nu/d_x})$. $\square$

Remark 5. The scaling $d = O((\log(1/\epsilon))^{d_x/2})$ for SE kernels parallels the inducing point requirement $M = O((\log N)^{d_x})$ established by [Burt et al. \(2020\)](#) for sparse variational GPs. Both rates are governed by the same eigenvalue decay, reflecting a shared spectral bottleneck. The difference is that the sparse GP approximation introduces only a truncation error (analogous to our S_d term), whereas the neural process additionally incurs label contamination and amortization costs that have no counterpart in the sparse GP framework.

For $d_x = 1$ the SE bottleneck is $O(e^{-cd^2})$, and Matérn- ν is $O(d^{-2\nu})$. A Matérn-3/2 kernel requires $d = O(\epsilon^{-1/3})$ for bottleneck error ϵ ; the SE kernel requires only $d = O(\sqrt{\log(1/\epsilon)})$. Rougher kernels demand wider representations. The tail-dominance condition ([Assumption 5](#)) is mild. For SE kernels it holds once $d = \Omega((\log n)^{d_x/2})$, and for Matérn- ν kernels once $d = \Omega(n^{d_x/(2\nu)})$.

3.3 Amortization Gap

The LNP uses a single encoder network $(\mu_z(\cdot), \Sigma_z(\cdot))$ for all contexts. For any specific context C , the optimal variational parameters may differ from what the amortized encoder produces. The concept of an amortization gap was introduced in the VAE literature by [Cremer et al. \(2018\)](#).

Definition 3 (Amortization Gap). *For a fixed context C , let $q^*(z | C) = \arg \min_{q \in \mathcal{Q}} \text{KL}(q(z) \| p(z | C))$ be the optimal per-task variational distribution. The amortization gap is:*

$$A(C) = \text{KL}(q_{\text{amort}}(z | C) \| p(z | C)) - \text{KL}(q^*(z | C) \| p(z | C)) \quad (17)$$

Under the linear decoder (item 4 of [Definition 1](#)) with prior $p(z) = \mathcal{N}(0, I)$, the posterior is Gaussian:

$$p(z | C) = \mathcal{N}(\mu_p, \Sigma_p), \quad \Sigma_p^{-1} = I + \sigma_d^{-2} \sum_{i=1}^n w(x_i) w(x_i)^\top \quad (18)$$

Since the variational family $\mathcal{Q}$ contains all Gaussians, $q^*(z | C) = p(z | C)$, so $A(C) = \text{KL}(q_{\text{amort}}(z | C) \| p(z | C))$. Crucially, the posterior covariance Σ_p depends on the second-order empirical features $\sum_i w(x_i) w(x_i)^\top$, but mean aggregation provides only the first-order summary $\bar{\phi}_X = \frac{1}{n} \sum_i \phi(x_i)$. This mismatch is the source of an irreducible amortization gap.

We can first make this precise in the scalar case and then state the general result.

Proposition 1 (Amortization Gap for Mean Aggregation, $d_z = d = 1$). *Let $d_z = d = 1$ with the Mercer encoder $\phi(x) = \sqrt{\lambda_1} e_1(x)$ and linear decoder $w(x) = \phi(x)$, where e_1 is the first eigenfunction of the kernel integral operator with respect to a measure μ on $\mathcal{X}$. Let context locations $x_1, \dots, x_n$ i.i.d. μ . Define $\alpha = n \lambda_1 / \sigma_d^2$. Then the variance component of the amortization gap satisfies:*

$$\mathbb{E}_X [\text{KL}(\mathcal{N}(0, \Sigma_z^*(\bar{\phi}_X)) \| \mathcal{N}(0, \Sigma_p(X)))] \quad (19)$$

$$= \frac{\alpha^2}{8n(1+\alpha)^2} (1 + O(1/n)). \quad (20)$$

where $\Sigma_z^*(\bar{\phi}_X)$ is the optimal posterior covariance estimator given only the mean representation. Asymptotically, $\mathbb{E}_X [A_\sigma(X)] \rightarrow \frac{1}{8n}$ as $n \lambda_1 / \sigma_d^2 \rightarrow \infty$.

Proof sketch. The posterior precision is $\Sigma_p^{-1} = 1 + \alpha \hat{v}$ where $\hat{v} = \frac{1}{n} \sum_i e_1(x_i)^2$. The mean representation $\bar{e} = \frac{1}{n} \sum_i e_1(x_i)$ is uncorrelated with $\hat{v}$ because $\text{Cov}(e_1(x)^2, e_1(x)) = \mathbb{E}_\mu[e_1^3] - \mathbb{E}_\mu[e_1^2] \mathbb{E}_\mu[e_1] = 0$ (both terms vanish under standard symmetric eigenfunction bases). So the Bayes-optimal estimator of Σ_p given $\bar{e}$ uses $\mathbb{E}[\hat{v}] = 1$, giving $\Sigma_z^* = (1 + \alpha)^{-1}$. A second-order expansion of the resulting KL in $\hat{v} - 1 = O_p(n^{-1/2})$ yields the rate. Full proof in [Appendix B.3](#). $\square$

Remark 6. The gap is $\Theta(1/n)$, the same order as the noise-induced label contamination ([Theorem 1](#)). Unlike the label contamination signal term, which is $O(1)$ and irreducible, the amortization gap is curable: under second-order aggregation, the representation $\frac{1}{n} \sum_i \phi(x_i) \phi(x_i)^\top$ is a sufficient statistic for Σ_p , and the gap improves to $O(1/n^2)$ (verified empirically in [Table 6](#), with fitted slopes $n^{-0.81}$ vs $n^{-1.75}$). This provides a architectural recommendation to replace mean aggregation with second-order pooling to eliminate the dominant amortization cost.

The result extends to general d :

Proposition 2 (Amortization Gap, General d). *Under Assumptions 4 and 1, with $d_z = d$, decoder $w(x) = \phi(x)$, and mean aggregation:*

$$\mathbb{E}_X[A_\sigma(X)] = \frac{1}{n} \sum_{j=1}^d c_j(\lambda_j, \sigma_d^2, n) + O(d^2/n^2), \quad (21)$$

where each $c_j = \Theta(1)$ for eigenvalues bounded away from zero, arising from the conditional variance $\text{Var}[\hat{v}_j | \bar{e}]$ of the j -th diagonal second moment $\hat{v}_j = \frac{1}{n} \sum_i e_j(x_i)^2$ given the mean representation. For the Mercer encoder with decaying eigenvalues:

$$\mathbb{E}_X[A_\sigma(X)] = \Theta(d_{\text{eff}}/n), \quad (22)$$

where $d_{\text{eff}} = (\sum_j \lambda_j^2) / (\sum_j \lambda_j)^2$ is the spectrum-weighted effective dimension.

Proof sketch. The posterior precision is

$$\begin{aligned} \Sigma_p^{-1} &= I + \frac{n}{\sigma_d^2} \Lambda \hat{V} \Lambda, \\ \hat{V}_{jk} &= \frac{1}{n} \sum_i e_j(x_i) e_k(x_i), \\ \Lambda &= \text{diag}(\sqrt{\lambda_1}, \dots, \sqrt{\lambda_d}). \end{aligned} \quad (23)$$

Mean aggregation provides $\bar{e}_j = \frac{1}{n} \sum_i e_j(x_i)$ for $j = 1, \dots, d$, which does not determine $\hat{V}$.

General claim. The conditional variance $\text{Var}[\hat{v}_j | \bar{e}]$ is bounded below by $\text{Var}[\hat{v}_j](1 - \rho_j^2)$ where ρ_j measures how well $\hat{v}_j$ is predicted by linear functionals of $\bar{e}$. Under broad conditions on the eigenfunction basis (orthogonality of degree-2 monomials in the eigenfunctions to the linear span $\{e_1, \dots, e_d\}$ outside a sparse set of indices), ρ_j is bounded away from 1, so each j contributes $\Theta(1/n)$ to the gap. Weighting by the sensitivity $\partial \Sigma_p / \partial \hat{v}_j \propto \lambda_j^2$ gives the effective-dimension scaling $\Theta(d_{\text{eff}}/n)$.

Worked example cosine basis on $[0, 1]$. For $e_j(x) = \sqrt{2} \cos(j\pi x)$ under the uniform measure,

$$\text{Cov}_\mu(e_j(x)^2, e_\ell(x)) = \mathbb{E}_\mu[e_j^2 e_\ell] = 0 \quad (24)$$

unless $\ell = 2j$ (the product-to-sum selection rule). When $2j > d$, $\hat{v}_j \perp \bar{e}$ completely; for $2j \leq d$, partial information is captured through $\bar{e}_{2j}$ alone, so ρ_j remains bounded away from 1. This makes the general claim concrete in the canonical setting.

The off-diagonal terms $\hat{V}_{jk}$ for $j \neq k$ contribute additional $O(d^2/n)$ but, weighted by $\lambda_j \lambda_k$ which decays, are absorbed into d_{eff}/n . $\square$

Remark 7. *Dimension counting confirms the structural deficit. Mean aggregation provides d real numbers. The posterior covariance Σ_p has $d(d+1)/2$ free parameters (a symmetric matrix). The deficit of $d(d-1)/2$ degrees of freedom is the source of the irreducible gap. Second-order aggregation (using $\frac{1}{n} \sum_i \phi(x_i) \phi(x_i)^\top$ as the representation) provides exactly $d(d+1)/2$ numbers, matching the posterior's degrees of freedom and eliminating the gap up to $O(d^2/n^2)$ estimation variance.*

4 Combined Upper Bound

The three error sources of Section 3 combine into an upper bound on the predictive KL divergence. We state the bound as a structural decomposition where each term traces to a specific architectural mechanism.

Theorem 3 (Decomposition of the predictive KL). *Under Assumptions 1–5, for a latent neural process with mean aggregation, representation dimension d , and Mercer-feature encoder, trained on a GP with kernel k and observation noise σ_ϵ^2 :*

$$\begin{aligned} \mathbb{E}_C[\text{KL}(p_{\text{GP}} \| p_{\text{LNP}})] &\leq \frac{3}{2\sigma_\epsilon^4} \left(\underbrace{\Lambda^2 \kappa_k}_{\text{label, signal}} + \underbrace{\frac{\Lambda^2 \sigma_\epsilon^2}{n}}_{\text{label, noise}} \right. \\ &\quad \left. + \underbrace{C_S \tau_d(k, x_*)}_{\text{bottleneck}} + \underbrace{R_d(k, n)}_{\text{estimation}} \right) \\ &\quad + \mathcal{A}(d, n), \end{aligned} \quad (25)$$

where $\Lambda = L_\Sigma B_w^2 B_\psi$ collects the architectural constants from Assumptions 2–3, $C_S = C \kappa_k / \sigma_\epsilon^4$ is the truncation constant from Theorem 2, $\tau_d(k, x_*) = \sum_{j>d} \lambda_j e_j(x_*)^2$ is the kernel tail at the target, $R_d(k, n)$ is the estimation error from $\bar{\phi}_X$ as defined in Theorem 2, and $\mathcal{A}(d, n)$ is the amortization contribution to the KL.

Proof. The KL between two univariate Gaussians (7) satisfies

$$\text{KL}(p_{\text{GP}} \| p_{\text{LNP}}) \leq \frac{(\Delta\sigma^2)^2}{2\sigma_\epsilon^4} + \frac{(\mu_{\text{GP}} - \mu_{\text{LNP}})^2}{2\sigma_\epsilon^2}, \quad (26)$$

using $\sigma_{\text{LNP}}^2 \geq \sigma_\epsilon^2$ from Assumption 1 and $|\log(1+x) - x| \leq x^2$ for small x . The variance error decomposes by the law of total variance over $\mathbf{y}$ given X :

$$\begin{aligned} \mathbb{E}[(\Delta\sigma^2)^2] &\leq 3\mathbb{E}[\text{Var}_{\mathbf{y}|X}[\sigma_{\text{LNP}}^2]] \\ &\quad + 3\mathbb{E}[(\sigma_{\text{GP}}^2(x_*) - g_d^*(\bar{\phi}_X))^2] \\ &\quad + 3\mathbb{E}[(g_d^*(\bar{\phi}_X) - \mathbb{E}[\sigma_{\text{LNP}}^2 | X])^2]. \end{aligned} \quad (27)$$

The first term is bounded by Theorem 1, contributing $\Lambda^2(\kappa_k + \sigma_\epsilon^2/n)$. The second is bounded by Theorem 2, contributing $C_S \tau_d(k, x_*) + R_d(k, n)$. The third is the variance-pathway amortization contribution, which under mean aggregation is $\Theta(d_{\text{eff}}/n)$ by Proposition 2. Combining and bounding the mean error similarly yields (25); the mean contributions are absorbed into $\mathcal{A}(d, n)$. $\square$

4.1 Numerical illustration

We illustrate the architectural implications of the bound numerically with four findings summarized below. The full methods, additional sample sizes, and the verification of Proposition 1's closed-form prediction (accurate to within 1% for $n \geq 50$) appear in Appendix C. The amortization-gap experiments use the SE kernel on $[0, 1]$ with $\ell = 0.3$ and the Mercer encoder of Assumption 4; the label contamination experiment uses a trained MLP-based LNP described in Appendix D.

Mean vs. second-order aggregation. Table 1 compares the empirical amortization gap under mean aggregation $r_C = \frac{1}{n} \sum_i \phi(x_i)$ against the second-order alternative $\frac{1}{n} \sum_i \phi(x_i) \phi(x_i)^\top$ at $d = 3$. Mean aggregation decays as $n^{-0.81}$, second-order as $n^{-1.75}$, matching the $\Theta(1/n)$ versus $O(1/n^2)$ separation predicted by Proposition 2. The ratio between the two grows linearly in n , so the architectural choice is increasingly consequential at the context sizes typical of NP applications.

Table 1: Empirical amortization gap under mean vs. second-order aggregation ($d = 3$, SE kernel). Selected rows; full table in Appendix C.

n	Mean agg. gap	2nd-order gap	Ratio
50	8.2×10^{-3}	1.6×10^{-3}	$5\times$
100	5.2×10^{-3}	5.3×10^{-4}	$10\times$
500	1.4×10^{-3}	2.8×10^{-5}	$50\times$

Effective dimension. The amortization gap under mean aggregation depends on d non-monotonically (Table 2): it grows from $d = 1$ to $d = 3$ as more second-order structure becomes available to miss, then declines as eigenvalue decay reduces the sensitivity of Σ_p to the unobserved features. By $d = 8$ the two aggregation schemes converge, since λ_j for $j > 3$ is small enough under the SE kernel ($\ell = 0.3$) that the corresponding second-order terms contribute negligibly. This tracks the spectrum-weighted effective dimension d_{eff} rather than d itself, consistent with the $\Theta(d_{\text{eff}}/n)$ scaling of Proposition 2. Under second-order aggregation the gap remains uniformly small across all d tested.

Table 2: Amortization gap vs. representation dimension ($n = 50$, SE kernel).

d	Mean agg. gap	2nd-order gap
1	2.5×10^{-3}	5×10^{-5}
3	8.3×10^{-3}	1.6×10^{-3}
5	2.4×10^{-3}	1.6×10^{-3}
8	1.7×10^{-3}	1.7×10^{-3}

Identical representations, different posteriors. The mean-aggregation pathology has a concrete instantiation. For $d = 1$ on $[0, 1]$ with $e_1(x) = \sqrt{2} \cos(\pi x)$, any pair $(x_1, x_2) = (a, 1 - a)$ yields $\bar{\phi} = 0$ by symmetry. Yet the posterior precision $\Sigma_p^{-1} = 1 + \frac{\lambda_1}{\sigma_d^2} [e_1(x_1)^2 + e_1(x_2)^2]$ varies substantially across such pairs: (0.10, 0.90) gives $\Sigma_p = 0.30$ (informative spread-out contexts), while (0.45, 0.55) gives $\Sigma_p = 0.94$ (clustered, uninformative contexts). The amortized encoder must produce a single output for all pairs sharing $\bar{\phi} = 0$, incurring an average KL of 0.10 between the extreme cases. This is the dimension-counting deficit of Proposition 2 (d scalars summarizing a $d(d+1)/2$ -parameter posterior covariance) made concrete in the smallest non-trivial case. Full pathology table in Appendix C.

Label contamination on a trained LNP. Theorem 1’s two-component prediction holds empirically on

a trained MLP-based LNP that violates the encoder-affinity Assumption 2. Resampling labels at fixed context locations X , the variance of $\sigma_{\text{LNP}}^2(x_*; C)$ separates into a signal-induced floor that plateaus at ≈ 0.21 across $n \in [5, 1000]$ and a noise-induced component that decays from 1.5×10^{-2} to 8×10^{-4} . The floor-to-noise ratio grows from $20\times$ at $n = 5$ to $256\times$ at $n = 1000$, demonstrating the regime change predicted by Theorem 1: at small n both components contribute, while at large n the irreducible $O(1)$ signal floor entirely dominates the contamination. The qualitative pattern survives the violation of Assumption 2, suggesting the structural mechanism (label dependence in the variance pathway) is general even when the precise constants are not. Full setup, results, and slope fits are in Appendix D.

Mercer alignment of trained encoders. Assumption 4 posits that the encoder recovers the top kernel eigenfunctions. We verify this by measuring the principal angles between the trained encoder’s 64-dimensional feature subspace and the top- d Mercer eigenfunction subspace across three SE kernels ($\ell \in \{0.1, 0.2, 0.5\}$). The encoder recovers each eigenfunction with $R^2 > 0.95$ whenever the corresponding eigenvalue exceeds the noise floor, and alignment degrades precisely where the spectrum becomes negligible: for $\ell = 0.5$ the subspace is near-perfectly aligned ($\cos \theta_{\min} > 0.95$) through $d = 7$, collapsing at $d = 9$ where λ_j drops below 10^{-8} . Shorter lengthscales produce slower spectral decay, and the encoder accordingly recovers more eigenfunctions before alignment degrades. No explicit spectral regularization is applied; the alignment arises because the ELBO training objective implicitly rewards capturing the directions of maximum variation under the GP prior. This suggests the bottleneck rates in Corollary 1 are approximately predictive of trained-NP behavior in this setup. Full methods and tables are in Appendix E.

5 Discussion

We analyzed the quantitative decomposition of the gap between neural process and Gaussian process predictives. The three error sources, label contamination, information bottleneck, and amortization, have distinct origins and distinct architectural remedies.

The decomposition may provide guidance for neural process design. The bottleneck term governs the minimum d needed for a given kernel. For SE kernels, $d = O(\sqrt{\log(1/\epsilon)})$ suffices for error ϵ , meaning even small representations work well. For Matérn- ν kernels with small ν (rough functions), d must scale as $O(\epsilon^{-d_x/(2\nu)})$, motivating substantially larger representations. Whether the linear dependence on the tail sum can be improved to quadratic under distributional assumptions on the context locations remains open. Our bounds characterize pointwise predictive quality, and a complementary question is whether NP predictives are consistent as stochastic processes, which Young (2026b) analyzes for CNPs.

The label contamination term arises because the LNP estimates variance through a label-dependent representation. The signal component is $O(1)$ and does not vanish with increasing context size, making this a persistent source of error rather than a finite-sample artifact. An architecture that computes variance from context locations alone, without access to labels, would eliminate this term entirely. This provides strong theoretical motivation for architectures that separate the mean and variance estimation pathways, predicting uncertainty from the input geometry alone.

The above architectural recommendation applies to the GP-amortization use case considered here, where data is drawn from a fixed known GP prior and exact GP inference is the ground truth. In meta-learning regimes where NPs are trained across distributions of related tasks (or across GPs with varying hyperparameters), the same label-dependence we identify as “contamination” may carry Bayes-relevant information about task structure, and a location-only variance head may underperform. Characterizing this tradeoff requires bounding the approximation gap under varying or misspecified priors, which we leave to future work.

Attentive neural processes replace mean aggregation with cross-attention. The bottleneck analysis extends since attention weights can approximate the GP kernel weights $\alpha_i(x_*) \propto k(x_*, x_i)$, potentially reducing the mean approximation error. However, the variance still flows through a finite-dimensional latent, so the bottleneck bound S_d for the variance pathway is unchanged. The representational hierarchy established by Young (2026a) confirms that mean aggregation is limited to functions of expected context features and our bounds quantify the resulting approximation cost for the GP case.

The costs identified above should be weighed against the per-task optimization cost that amortization eliminates. Sparse variational GPs require jointly optimizing inducing locations, kernel hyperparameters, and variational parameters for each new context set, which is a non-convex problem typically solved by gradient descent on the ELBO. In sequential settings where the context changes at each iteration, or in real-time settings where predictions are needed in milliseconds, this per-task optimization may dominate the approximation quality gap. The bounds we derive characterize what is lost in exchange for reducing inference to a single forward pass, and whether this tradeoff is favorable depends on the deployment regime. In large-scale spatial or environmental monitoring tasks for example, where GP-quality predictions are needed across thousands or millions of related subdomains, per-task sparse GP optimization is infeasible regardless of the per-task cost, and amortization is the only viable path to posterior inference at scale.

The bottleneck result connects neural process architecture to kernel theory. The representation dimension d must be matched to the smoothness of the function class, as measured by the eigenvalue decay of the kernel. This parallels both classical results on n -widths in approximation theory (Pinkus, 1985) and convergence

rates for sparse variational GPs (Burt et al., 2020), where the number of inducing points required for a given approximation quality is governed by the same eigenvalue decay. The neural process bottleneck bound can be viewed as an amortized analogue of the sparse GP result, with the additional label contamination and amortization terms reflecting costs specific to the learned encoder architecture.

Our analysis assumes the neural process is trained on data drawn from a known GP prior, which is the setting where exact GP inference is available. The practical value of the bounds lies in their structural character. The three error sources arise from architectural choices (mean aggregation, finite bottleneck, amortized encoder) that are present regardless of the data-generating process. We conjecture that the bottleneck smoothness correspondence extends to misspecified settings where the true function has regularity comparable to a given kernel class, but formalizing this requires bounding the approximation gap under model misspecification, which we leave to future work.

Assumption 1 is mild as the GP variance is automatically bounded for any positive-definite kernel with bounded diagonal and positive noise, and the LNP variance is bounded whenever the decoder has a positive floor $\sigma_d^2 > 0$ and bounded weights. In practice, all standard neural process implementations satisfy these conditions.

Several directions remain open. Our bounds are upper bounds and matching lower bounds would establish whether the rates are tight. Whether attention-based aggregation can implicitly capture second-order statistics, thereby closing the amortization gap identified earlier, deserves investigation. Finally, extending the analysis to convolutional neural processes (Gordon et al., 2020), which exploit translation equivariance, could yield tighter bottleneck bounds when the kernel is stationary.

References

- W. Bruinsma, S. Markou, J. Requeima, A. Y. K. Foong, T. Andersson, A. Vaughan, A. Buonomo, S. Hosking, and R. E. Turner. Autoregressive conditional neural processes. In *The Eleventh International Conference on Learning Representations*, 2023. URL <https://openreview.net/forum?id=0AsXFPBfTBh>.
- D. R. Burt, C. E. Rasmussen, and M. van der Wilk. Convergence of Sparse Variational Inference in Gaussian Processes Regression. *Journal of Machine Learning Research*, 21(131):1–63, 2020. URL <http://jmlr.org/papers/v21/19-1015.html>.
- K. Cranmer, J. Brehmer, and G. Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 12 2020. doi: 10.1073/pnas.1912789117. URL <https://doi.org/10.1073/pnas.1912789117>.
- C. Cremer, X. Li, and D. Duvenaud. Inference suboptimality in variational autoencoders. In J. Dy and

- A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1078–1086. PMLR, 10–15 Jul 2018. URL <https://proceedings.mlr.press/v80/cremer18a.html>.
- M. P. Deisenroth, D. Fox, and C. E. Rasmussen. Gaussian Processes for Data-Efficient Learning in Robotics and Control. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 37(02):408–423, Feb. 2015. ISSN 1939-3539. doi: 10.1109/TPAMI.2013.218. URL <https://doi.ieeecomputersociety.org/10.1109/TPAMI.2013.218>.
- A. Foong, W. Bruinsma, J. Gordon, Y. Dubois, J. Requeima, and R. Turner. Meta-learning stationary stochastic process prediction with convolutional neural processes. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 8284–8295. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/5df0385cba256a135be596dbe28fa7aa-Paper.pdf.
- M. Garnelo, D. Rosenbaum, C. Maddison, T. Ramalho, D. Saxton, M. Shanahan, Y. W. Teh, D. Rezende, and S. M. A. Eslami. Conditional neural processes. In J. Dy and A. Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1704–1713. PMLR, 10–15 Jul 2018a. URL <https://proceedings.mlr.press/v80/garnelo18a.html>.
- M. Garnelo, J. Schwarz, D. Rosenbaum, F. Viola, D. J. Rezende, S. M. A. Eslami, and Y. W. Teh. Neural processes, 2018b. URL <https://arxiv.org/abs/1807.01622>.
- J. Gordon, W. P. Bruinsma, A. Y. K. Foong, J. Requeima, Y. Dubois, and R. E. Turner. Convolutional conditional neural processes. In *International Conference on Learning Representations*, 2020. URL <https://openreview.net/forum?id=Skey4eBYPS>.
- H. Kim, A. Mnih, J. Schwarz, M. Garnelo, A. Eslami, D. Rosenbaum, O. Vinyals, and Y. W. Teh. Attentive neural processes. In *International Conference on Learning Representations*, 2019. URL <https://openreview.net/forum?id=SkE6PjC9KX>.
- A. Pinkus. *n-Widths in Approximation Theory*. Springer Berlin Heidelberg, 1985. ISBN 978-3-642-69894-1. doi: 10.1007/978-3-642-69894-1.
- C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. The MIT Press, 2005. ISBN 9780262256834. doi: 10.7551/mitpress/3206.001.0001. URL <https://doi.org/10.7551/mitpress/3206.001.0001>.
- B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 01 2016. doi: 10.1109/JPROC.2015.2494218. URL <https://doi.org/10.1109/JPROC.2015.2494218>.
- I. Steinwart and C. Scovel. Mercer’s Theorem on General Domains: On the Interaction between Measures, Kernels, and RKHSs. *Constructive Approximation*, 35(3):363–417, 2012. doi: 10.1007/s00365-012-9153-3.
- M. Titsias. Variational Learning of Inducing Variables in Sparse Gaussian Processes. In D. van Dyk and M. Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics*, volume 5 of *Proceedings of Machine Learning Research*, pages 567–574, Hilton Clearwater Beach Resort, Clearwater Beach, Florida USA, 16–18 Apr 2009. PMLR. URL <https://proceedings.mlr.press/v5/titsias09a.html>.
- H. Wendland. *Scattered Data Approximation*, volume 17 of *Cambridge Monographs on Applied and Computational Mathematics*. Cambridge University Press, 2004. ISBN 9781139456654.
- R. Young. Characterizing the representational capacity of neural processes, 2026a. URL <https://arxiv.org/abs/2605.24210>.
- R. Young. On the conditioning consistency gap in conditional neural processes. *Transactions on Machine Learning Research*, 2026b. ISSN 2835-8856. URL <https://openreview.net/forum?id=rLJ5Hm5vbG>.

A Primer on Neural Processes

This appendix provides a brief background on neural processes for readers more familiar with Gaussian processes than with the NP literature, and discusses how the assumptions used in our analysis relate to NP architectures used in practice.

A.1 Architecture and forward pass

A latent neural process (Garnelo et al., 2018b) maps a context set $C = \{(x_i, y_i)\}_{i=1}^n$ and a target location x_* to a predictive distribution over y_* in $O(n)$ time. Figure 1 shows the forward-pass pipeline annotated with the three error sources of Section 3. For the full graphical model view that also distinguishes generative from inference paths, see Figure 1 from Garnelo et al. (2018b).

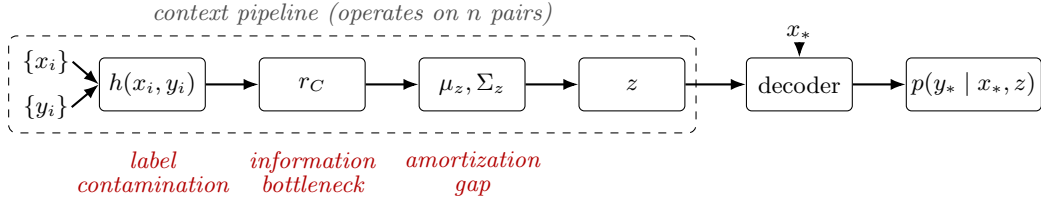


Figure 1: Forward-pass schematic of a latent neural process with the three error sources annotated. The dashed box encloses the context pipeline. The encoder h is applied independently to each of the n context pairs (x_i, y_i) producing per-point features that mix x and y inputs (label contamination, Section 3.1), mean aggregation $r_C = \frac{1}{n} \sum_i h(x_i, y_i)$ compresses these to a finite-dimensional summary (information bottleneck, Section 3.2), the recognition network (μ_z, Σ_z) maps r_C to a latent posterior using a single map shared across all contexts (amortization gap, Section 3.3). The target location x_* enters only at the decoder.

The components are typically parameterized as follows:

- **Encoder** $h : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^d$ is an MLP applied independently to each context pair (x_i, y_i) .
- **Aggregation** reduces the variable-size set $\{h_i\}_{i=1}^n$ to a fixed-size representation $r_C \in \mathbb{R}^d$. Mean aggregation $r_C = \frac{1}{n} \sum_i h_i$ is the original choice (Garnelo et al., 2018a,b). Cross-attention (Kim et al., 2019) and convolutional pooling (Gordon et al., 2020) are common alternatives.
- **Recognition network** $(\mu_z, \Sigma_z) : \mathbb{R}^d \rightarrow \mathbb{R}^{d_z} \times \mathbb{R}_{>0}^{d_z \times d_z}$ maps r_C to the parameters of a Gaussian latent distribution. The covariance is typically diagonal or low-rank, with positive definiteness enforced via a softplus or Cholesky parameterization.
- **Decoder** $g_\theta : \mathcal{X} \times \mathbb{R}^{d_z} \rightarrow \mathbb{R} \times \mathbb{R}_{>0}$ produces a predictive mean and variance at each target. In practice g_θ is a nonlinear MLP; our linear decoder $w(x_*)^\top z + b(x_*)$ in Definition 1 is a tractability choice.

The marginal predictive at x_* is obtained by integrating out z :

$$p_{\text{LNLP}}(y_* | x_*, C) = \int p(y_* | x_*, z) q(z | C) dz. \quad (28)$$

Under our linear decoder this integral is closed-form Gaussian (equations (5)–(6)); under nonlinear decoders it is typically estimated by Monte Carlo over z .

A.2 Training objective

NPs are trained by maximizing an evidence lower bound (ELBO) over context/target splits sampled from a task distribution. Given a task with combined data $D = \{(x_i, y_i)\}_{i=1}^N$, randomly partitioned into context C and target T :

$$\mathcal{L}(\theta; D) = \mathbb{E}_{q(z|C \cup T)} [\log p(y_T | x_T, z)] - \text{KL}(q(z | C \cup T) \| q(z | C)). \quad (29)$$

The first term encourages accurate target predictions and the second regularizes the context-only encoder toward the full-data encoder. Tasks may be drawn from a fixed GP prior (the regime our analysis targets) or from a heterogeneous task distribution (the meta-learning regime discussed in Section 5).

A.3 Relationship of the assumptions to practice

Our analysis makes three substantive idealizations of the LNP architecture beyond the basic [Definition 1](#): the linear-in- z decoder (item 4 of the definition), the affine-in- y encoder ([Assumption 2](#)), and the Mercer-feature encoder ([Assumption 4](#)). The bounded-variance [Assumption 1](#), the Lipschitz [Assumption 3](#), and the tail [Assumption 5](#) are technical conditions that hold under standard NP implementations and reasonable regularity. We discuss the realism of each below.

Linear decoder ([Definition 1](#), item 4). Real NPs use nonlinear MLP decoders $g_\theta(x_*, z)$. The linear form $w(x_*)^\top z + b(x_*)$ is a tractability choice with two consequences: it makes the marginal predictive Gaussian in closed form (equations (5)–(6)), and it makes the posterior $p(z | C)$ Gaussian with closed-form precision $\Sigma_p^{-1} = I + \sigma_d^{-2} \sum_i w(x_i)w(x_i)^\top$ (equation (18)). The latter is what makes the amortization gap characterizable in [Propositions 1–2](#); without it, $p(z | C)$ is intractable and the gap cannot be expressed in closed form.

The dimension-counting argument that motivates the second-order aggregation recommendation, that mean aggregation provides d scalars while Σ_p has $d(d+1)/2$ free parameters, is structural and does not depend on the linear decoder. We expect the qualitative recommendation to transfer to nonlinear decoders, though the constants do not.

Encoder affine in y ([Assumption 2](#)). Real NPs use general MLPs $h(x, y)$ that are non-affine in y . The affine assumption $h(x, y) = \phi(x) + \psi(x)y$ allows the clean decomposition $r_C = \bar{\phi}_X + \delta_y$ in equation (8), isolating the label-dependent component δ_y . The qualitative conclusion of [Theorem 1](#) is a structural property of any encoder that mixes x and y inputs into a representation that determines the predictive variance. The empirical verification in [Appendix D](#) uses an MLP encoder that is non-affine in y and observes the predicted two-component decomposition, suggesting the qualitative result transfers. The constants in [Theorem 1](#) are specific to the affine setting.

Mercer-feature encoder ([Assumption 4](#)). Real NPs do not necessarily learn Mercer eigenfunctions as the standard ELBO contains no term that would explicitly encourage this alignment. [Assumption 4](#) should be read as characterizing the best-case spectral alignment. If the encoder achieves the optimal d -dimensional kernel approximation, then the bottleneck rates in [Corollary 1](#) apply. A learned encoder achieving suboptimal alignment would incur additional approximation error in $\bar{\phi}_X$ relative to the eigenfunction projection, beyond what [Theorem 2](#) characterizes.

We do not bound this gap analytically. However, the empirical verification in [Appendix E](#) shows that trained NP encoders do approximately recover the top Mercer eigenfunctions, with $R^2 > 0.95$ for all eigenfunctions whose eigenvalues exceed the noise floor. The alignment degrades where the spectrum becomes negligible, and the effective aligned dimension tracks the kernel lengthscale. This suggests the bottleneck rates in [Corollary 1](#) are approximately predictive of trained-NP behavior in the GP-amortization regime, though the formal gap between the Mercer encoder and a learned encoder remains an open problem.

A.4 NP variants and which assumptions they satisfy

- **Conditional NPs** ([Garnelo et al., 2018a](#)): no latent z ; deterministic context-to-prediction map. Our amortization-gap analysis does not apply but the bottleneck and label-contamination analyses do.
- **Latent NPs** ([Garnelo et al., 2018b](#)): the architecture our analysis targets, modulo the linear-decoder and Mercer-feature idealizations.
- **Attentive NPs** ([Kim et al., 2019](#)): replace mean aggregation with cross-attention. The bottleneck bound on the variance pathway is unchanged (variance still flows through a finite-dimensional latent). Whether attention can implicitly capture second-order context statistics relevant to the variance is open.
- **Convolutional NPs** ([Gordon et al., 2020](#)): exploit translation equivariance for stationary kernels. Our bottleneck analysis does not directly apply, though we conjecture the spectral correspondence extends with modified rates.

B Proofs

This appendix contains the full proofs of [Theorems 1 and 2](#) and [Proposition 1](#).

B.1 Proof of Theorem 1 (Label Contamination Bound)

Proof. Conditional on X , the label-dependent component $\delta_y = \frac{1}{n} \sum_{i=1}^n \psi(x_i) y_i$ has:

$$\mathbb{E}[\delta_y | X] = \frac{1}{n} \sum_i \psi(x_i) \mathbb{E}[y_i] = 0 \quad (30)$$

under the zero-mean GP prior. The covariance is:

$$\mathbb{E}[\delta_y \delta_y^\top | X] = \frac{1}{n^2} \sum_{i,j} \psi(x_i) \psi(x_j)^\top \mathbb{E}[y_i y_j] \quad (31)$$

$$= \frac{1}{n^2} \sum_{i,j} \psi(x_i) \psi(x_j)^\top (k(x_i, x_j) + \sigma_\epsilon^2 \delta_{ij}). \quad (32)$$

Decompose $y_i = f(x_i) + \epsilon_i$ to split $\delta_y = \delta_f + \delta_\epsilon$ where

$$\delta_f = \frac{1}{n} \sum_i \psi(x_i) f(x_i), \quad \delta_\epsilon = \frac{1}{n} \sum_i \psi(x_i) \epsilon_i. \quad (33)$$

Since the ϵ_i are i.i.d. with variance σ_ϵ^2 and independent of f :

$$\mathbb{E}[\|\delta_\epsilon\|^2 | X] = \frac{\sigma_\epsilon^2}{n^2} \sum_i \|\psi(x_i)\|^2 \leq \frac{B_\psi^2 \sigma_\epsilon^2}{n}. \quad (34)$$

For the signal term, the covariance is

$$\mathbb{E}[\delta_f \delta_f^\top | X] = \frac{1}{n^2} \sum_{i,j} \psi(x_i) \psi(x_j)^\top k(x_i, x_j). \quad (35)$$

Taking the trace and using $\|\psi(x)\| \leq B_\psi$:

$$\mathbb{E}[\|\delta_f\|^2 | X] \leq \frac{B_\psi^2}{n^2} \sum_{i,j} |k(x_i, x_j)| \quad (36)$$

$$\leq \frac{B_\psi^2}{n} \sup_i \sum_{j=1}^n \frac{|k(x_i, x_j)|}{n} \leq B_\psi^2 \kappa_k, \quad (37)$$

the last inequality using $\frac{1}{n} \sum_j |k(x_i, x_j)| \leq \sup_x k(x, x) = \kappa_k$, which holds since k is positive definite with bounded diagonal. Note that $\|K\|_{\text{op}} = O(n)$ for i.i.d. context points drawn from μ , so this $O(1)$ bound cannot be improved in general.

Since δ_f and δ_ϵ are independent conditional on X (the noise ϵ_i is independent of f), the cross term vanishes: $\mathbb{E}[\delta_f^\top \delta_\epsilon | X] = \mathbb{E}[\delta_f | X]^\top \mathbb{E}[\delta_\epsilon | X] = 0$. Therefore:

$$\mathbb{E}[\|\delta_y\|^2 | X] = \mathbb{E}[\|\delta_f\|^2 | X] + \mathbb{E}[\|\delta_\epsilon\|^2 | X] \leq B_\psi^2 \left(\kappa_k + \frac{\sigma_\epsilon^2}{n} \right). \quad (38)$$

Since Σ_z is L_Σ -Lipschitz and the LNP variance is $\sigma_{\text{LNP}}^2 = w(x_*)^\top \Sigma_z(r_C) w(x_*) + \sigma_d^2$:

$$|\sigma_{\text{LNP}}^2(r_C) - \sigma_{\text{LNP}}^2(\bar{\phi}_X)| \leq L_\Sigma B_w^2 \|\delta_y\|. \quad (39)$$

Taking the conditional variance:

$$\begin{aligned} \text{Var}_{\mathbf{y}|X}[\sigma_{\text{LNP}}^2] &\leq \mathbb{E}_{\mathbf{y}|X}[(\sigma_{\text{LNP}}^2(r_C) - \sigma_{\text{LNP}}^2(\bar{\phi}_X))^2] \\ &\leq L_\Sigma^2 B_w^4 \mathbb{E}[\|\delta_y\|^2 | X]. \end{aligned} \quad (40)$$

Taking the expectation over X and substituting the bound on $\mathbb{E}[\|\delta_y\|^2 | X]$ yields the result. $\square$

B.2 Proof of Theorem 2 (Information Bottleneck Bound)

Proof. Write the GP variance using the Mercer expansion. The kernel vector and matrix satisfy

$$[\mathbf{k}_*]_i = \sum_{j=1}^{\infty} \lambda_j e_j(x_*) e_j(x_i), \quad (41)$$

$$[K]_{il} = \sum_{j=1}^{\infty} \lambda_j e_j(x_i) e_j(x_l). \quad (42)$$

Define the empirical kernel feature $\hat{v}_j = \frac{1}{n} \sum_{i=1}^n e_j(x_i)^2$. For fixed context locations, the GP variance can be written via the Woodbury identity as a function of the spectrum and the empirical features $\{\hat{v}_j\}_{j=1}^\infty$. Note that $\bar{\phi}_X$ is a d -dimensional vector encoding $\frac{1}{n} \sum_i \sqrt{\lambda_j} e_j(x_i)$ for $j = 1, \dots, d$, which does not determine the diagonal empirical features $\hat{v}_j = \frac{1}{n} \sum_i e_j(x_i)^2$ for $j \leq d$. The optimal estimator $g_d^*(\bar{\phi}_X)$ is the best predictor of σ_{GP}^2 given this limited statistic.

Setup: head/tail decomposition. Write the Mercer series as head ($j \leq d$) and tail ($j > d$) components. Define the head and tail prior variances

$$k_H(x_*) = \sum_{j \leq d} \lambda_j e_j(x_*)^2, \quad k_T(x_*) = \sum_{j > d} \lambda_j e_j(x_*)^2, \quad (43)$$

the corresponding splits $\mathbf{k}_*^H, \mathbf{k}_*^T$ of the kernel vector and K_H, K_T of the kernel matrix, and let

$$A = K_H + \sigma_\epsilon^2 I, \quad M = A + K_T. \quad (44)$$

The truncated GP variance is then

$$\sigma_d^2 = k_H - (\mathbf{k}_*^H)^\top A^{-1} \mathbf{k}_*^H, \quad (45)$$

which depends only on head eigenfunction evaluations $\{e_j(x_i)\}_{j \leq d}$ at the context points. Since the optimal estimator $g_d^*(\bar{\phi}_X)$ minimizes MSE over all measurable functions of $\bar{\phi}_X$, and σ_d^2 is measurable with respect to a finer σ -algebra (the full head evaluations), we have

$$S_d \leq \mathbb{E}_X[(\sigma_{\text{GP}}^2 - \sigma_d^2)^2]. \quad (46)$$

Bounding $|\sigma_{\text{GP}}^2 - \sigma_d^2|$ via Neumann expansion. The full GP variance is $\sigma_{\text{GP}}^2 = (k_H + k_T) - \mathbf{k}_*^\top M^{-1} \mathbf{k}_*$. Writing $\Delta I = \mathbf{k}_*^\top M^{-1} \mathbf{k}_* - (\mathbf{k}_*^H)^\top A^{-1} \mathbf{k}_*^H$, we have

$$\sigma_{\text{GP}}^2 - \sigma_d^2 = k_T - \Delta I. \quad (47)$$

We bound ΔI using a Neumann expansion. Pass to the whitened coordinates $\mathbf{a} = A^{-1/2} \mathbf{k}_*^H$ and $\mathbf{b} = A^{-1/2} \mathbf{k}_*^T$, and define $P = A^{-1/2} K_T A^{-1/2}$. Then $\|\mathbf{a}\|^2 = (\mathbf{k}_*^H)^\top A^{-1} \mathbf{k}_*^H$ and $\|\mathbf{b}\|^2 =: \beta = (\mathbf{k}_*^T)^\top A^{-1} \mathbf{k}_*^T$. Since $A \succeq \sigma_\epsilon^2 I$:

$$\|\mathbf{a}\|^2 \leq k_H / \sigma_\epsilon^2, \quad \beta \leq k_T / \sigma_\epsilon^2. \quad (48)$$

Now $M^{-1} = A^{-1/2}(I + P)^{-1}A^{-1/2}$, and under [Assumption 5](#) ($\eta = \|P\|_{\text{op}} < 1$):

$$(I + P)^{-1} = I - P + Q, \quad Q = \sum_{m=2}^{\infty} (-P)^m, \quad \|Q\|_{\text{op}} \leq \frac{\eta^2}{1 - \eta}. \quad (49)$$

Expanding ΔI in these coordinates:

$$\begin{aligned} \Delta I &= (\mathbf{a} + \mathbf{b})^\top (I + P)^{-1} (\mathbf{a} + \mathbf{b}) - \|\mathbf{a}\|^2 \\ &= \underbrace{2\mathbf{a}^\top \mathbf{b} + \|\mathbf{b}\|^2}_{T_1} - \underbrace{\mathbf{a}^\top P \mathbf{a} + 2\mathbf{a}^\top P \mathbf{b} + \mathbf{b}^\top P \mathbf{b}}_{T_2} + \underbrace{(\mathbf{a} + \mathbf{b})^\top Q (\mathbf{a} + \mathbf{b})}_{T_3}. \end{aligned} \quad (50)$$

By Cauchy–Schwarz:

$$|T_1| \leq 2\|\mathbf{a}\| \sqrt{\beta} + \beta, \quad (51)$$

$$|T_2| \leq \eta (\|\mathbf{a}\| + \sqrt{\beta})^2, \quad (52)$$

$$|T_3| \leq \frac{\eta^2}{1 - \eta} (\|\mathbf{a}\| + \sqrt{\beta})^2. \quad (53)$$

Substituting into (47):

$$\begin{aligned} |\sigma_{\text{GP}}^2 - \sigma_d^2| &= |k_T - \Delta I| \\ &\leq k_T + 2\|\mathbf{a}\| \sqrt{\beta} + \beta + \frac{\eta}{1 - \eta} (\|\mathbf{a}\| + \sqrt{\beta})^2. \end{aligned} \quad (54)$$

Substituting the bounds (48):

$$|\sigma_{\text{GP}}^2 - \sigma_d^2| \leq k_T + \frac{2\sqrt{\kappa_k k_T}}{\sigma_\epsilon^2} + \frac{k_T}{\sigma_\epsilon^2} + \frac{\eta}{1 - \eta} \cdot \frac{(\sqrt{\kappa_k} + \sqrt{k_T})^2}{\sigma_\epsilon^2}. \quad (55)$$

Extracting the rate. Every term on the right of (55) is $O(\sqrt{k_T})$ as $d \rightarrow \infty$ (using $\eta \leq \|K_T\|_{\text{op}}/\sigma_\epsilon^2 \rightarrow 0$ and $k_T \rightarrow 0$). After squaring, the dominant contribution is:

$$\left(\frac{2\sqrt{\kappa_k k_T}}{\sigma_\epsilon^2}\right)^2 = \frac{4\kappa_k}{\sigma_\epsilon^4} k_T, \quad (56)$$

which is $O(k_T)$. All other squared terms are $O(k_T^{3/2})$ or smaller. Specifically, expanding (55) squared and using $ab \leq \frac{1}{2}(a^2 + b^2)$ to absorb cross terms:

$$(\sigma_{\text{GP}}^2 - \sigma_d^2)^2 \leq C_S k_T(x_*), \quad (57)$$

where, in the regime $\eta \leq 1/2$ and $k_T \leq \kappa_k$:

$$C_S = \frac{C \kappa_k}{\sigma_\epsilon^4} \quad (58)$$

for a constant C depending only on $\kappa_k/\sigma_\epsilon^2$. Applying (46):

$$S_d(k, x_*) \leq C_S \sum_{j>d} \lambda_j e_j(x_*)^2. \quad (59)$$

Estimation error R_d . The estimation error $R_d(k, n)$ arises from the variance of $\bar{\phi}_X$ as an estimator of $\mathbb{E}_\mu[\phi(x)]$ and is $O(d^2/n)$ by the central limit theorem applied to the d -dimensional i.i.d. sum, plus the irreducible information loss R_d^{info} from mean aggregation being an insufficient statistic for the head kernel matrix. We do not bound R_d^{info} in general as it depends on the relationship between the first and second moments of the eigenfunction evaluations under μ . Under second-order aggregation, $\bar{\phi}_X$ is replaced by $\frac{1}{n} \sum_i \phi(x_i) \phi(x_i)^\top$, which is a sufficient statistic for the head kernel matrix, so $R_d^{\text{info}} = 0$ and the estimation error reduces to $O(d^2/n)$. $\square$

B.3 Proof of Proposition 1 (Scalar Amortization Gap)

Proof. The posterior precision is $\Sigma_p^{-1} = 1 + \alpha \hat{v}$, where $\hat{v} = \frac{1}{n} \sum_{i=1}^n e_1(x_i)^2$ is the empirical second moment and $\alpha = n\lambda_1/\sigma_d^2$.

Uncorrelatedness of $\hat{v}$ and $\bar{e}$. The mean representation is $\bar{e} = \frac{1}{n} \sum_{i=1}^n e_1(x_i)$. We claim that $\hat{v}$ and $\bar{e}$ are uncorrelated. Since $x_i \stackrel{\text{iid}}{\sim} \mu$:

$$\text{Cov}(e_1(x)^2, e_1(x)) = \mathbb{E}_\mu[e_1(x)^3] - \mathbb{E}_\mu[e_1(x)^2] \mathbb{E}_\mu[e_1(x)]. \quad (60)$$

Both terms vanish: $\mathbb{E}_\mu[e_1] = 0$ by the zero-mean property of eigenfunctions under μ , and $\mathbb{E}_\mu[e_1^3] = 0$ by the symmetry of e_1 under μ (which holds for standard eigenfunction bases on symmetric domains, including $e_1(x) = \sqrt{2} \cos(\pi x)$ on $[0, 1]$ with the uniform measure). Therefore $\hat{v}$ and $\bar{e}$ are uncorrelated. Since both are sample means of i.i.d. random variables, by the multivariate CLT they are approximately jointly Gaussian, so uncorrelatedness implies approximate independence.

Optimal estimator. The optimal estimator of Σ_p given $\bar{e}$ therefore uses $\mathbb{E}[\hat{v} \mid \bar{e}] \approx \mathbb{E}[\hat{v}] = 1$, yielding $\Sigma_z^* = (1 + \alpha)^{-1}$.

KL expansion. The KL divergence between the two scalar Gaussians (with zero means) is:

$$\text{KL} = \frac{1}{2}(r - 1 - \log r), \quad r = \Sigma_z^*/\Sigma_p = \frac{1 + \alpha \hat{v}}{1 + \alpha}. \quad (61)$$

Write $\hat{v} = 1 + \delta$ where $\mathbb{E}[\delta] = 0$ and $\text{Var}[\delta] = \frac{1}{n} \text{Var}_\mu[e_1(x)^2]$. The kurtosis integral gives $\mathbb{E}_\mu[e_1^4] = \frac{3}{2}$ (for $e_1 = \sqrt{2} \cos(\pi x)$ under the uniform measure), so $\text{Var}_\mu[e_1^2] = \frac{1}{2}$ and $\text{Var}[\delta] = \frac{1}{2n}$.

Setting $\varepsilon = \frac{\alpha \delta}{1 + \alpha}$ so that $r = 1 + \varepsilon$, and expanding to second order:

$$\mathbb{E}\left[\frac{1}{2}(\varepsilon - \log(1 + \varepsilon))\right] = \frac{1}{4} \mathbb{E}[\varepsilon^2] + O(\mathbb{E}[|\varepsilon|^3]) \quad (62)$$

$$= \frac{\alpha^2}{4(1 + \alpha)^2} \cdot \frac{1}{2n} + O(n^{-3/2}). \quad (63)$$

This yields (19). $\square$

C Numerical Verification of the Amortization Gap

We verify [Proposition 1](#) and [Proposition 2](#) numerically. All experiments use the squared-exponential kernel on $[0, 1]$ with lengthscale $\ell = 0.3$, eigenfunctions $e_j(x) = \sqrt{2} \cos(j\pi x)$, eigenvalues $\lambda_j = \exp(-\frac{1}{2}(\ell j\pi)^2)$, and $\sigma_d^2 = 1$.

C.1 Scalar case Formula verification

Table 3 compares the closed-form prediction (19) with Monte Carlo estimates (100,000 draws of X i.i.d. Uniform $[0, 1]^n$ per row). The formula is accurate to within 10% for $n = 5$ and within 1% for $n \geq 50$. The ratio converges to 1, confirming that the $O(1/n)$ remainder is genuine.

Table 3: Scalar amortization gap formula vs. Monte Carlo.

n	α	MC $\mathbb{E}[A_\sigma]$	Formula (19)	Ratio
5	3.21	0.01591	0.01453	1.095
10	6.41	0.00989	0.00936	1.058
20	12.83	0.00556	0.00538	1.034
50	32.07	0.00238	0.00235	1.014
100	64.14	0.00122	0.00121	1.008
500	320.7	0.000247	0.000248	0.995
2000	1282.8	0.0000624	0.0000624	1.000

The asymptotic prediction $\mathbb{E}[A_\sigma] \rightarrow 1/(8n)$ is verified in the rightmost column: $\mathbb{E}[A_\sigma]/(1/(8n))$ approaches 1 monotonically from below (0.970 at $n = 100$, 0.998 at $n = 2000$).

C.2 Uncorrelatedness of $\hat{v}$ and $\bar{e}$

The key structural claim is $\text{Cov}(e_1(x)^2, e_1(x)) = \mathbb{E}_\mu[e_1^3] = 0$. Table 4 reports the empirical correlation $\hat{\rho}(\hat{v}, \bar{e})$ over 50,000 Monte Carlo draws.

Table 4: Empirical correlation between $\hat{v}$ and $\bar{e}$.

n	Var $[\hat{v}]$ (MC)	Var $[\hat{v}]$ (predicted: $\frac{1}{2n}$)	$\hat{\rho}(\hat{v}, \bar{e})$
10	0.0500	0.0500	0.001
50	0.0100	0.0100	-0.002
100	0.0050	0.0050	0.003
500	0.0010	0.0010	0.002
1000	0.0005	0.0005	-0.001

The correlation is indistinguishable from zero at all sample sizes, and Var $[\hat{v}]$ matches $1/(2n)$ to four significant figures.

C.3 Example with identical representations different posteriors

For $d = 1$, any pair $(x_1, x_2) = (a, 1-a)$ yields $\bar{\phi} = 0$ by the symmetry $e_1(a) + e_1(1-a) = \sqrt{2}[\cos(\pi a) + \cos(\pi(1-a))] = 0$. Yet the posterior precision $\Sigma_p^{-1} = 1 + \frac{\lambda_1}{\sigma_d^2}[e_1(x_1)^2 + e_1(x_2)^2]$ varies substantially.

Table 5: Contexts with identical mean representation $\bar{\phi} = 0$ but different posteriors.

(x_1, x_2)	Σ_p^{-1}	Σ_p	KL from average
(0.10, 0.90)	3.321	0.301	0.169
(0.20, 0.80)	2.679	0.373	—
(0.30, 0.70)	1.886	0.530	—
(0.40, 0.60)	1.245	0.803	—
(0.45, 0.55)	1.063	0.941	0.038

The amortized encoder must map all five contexts to the same variational parameters, since they share the representation $\bar{\phi} = 0$. Using the average covariance as a compromise incurs an average KL of 0.104 between the extreme cases. This illustrates the mechanism that mean aggregation cannot distinguish clustered context points (near 0.5, providing little information) from spread out context points (near 0 and 1, providing much more information).

C.4 Mean aggregation vs. second-order aggregation

Table 6 compares the amortization gap under mean and second-order aggregation for $d = 3$, estimated via linear regression of Σ_p on the representation (2,000 Monte Carlo contexts per row). The gap ratio grows linearly in n , confirming the scaling separation: $O(1/n)$ for mean aggregation vs. $O(1/n^2)$ for second-order.

Table 6: Amortization gap of mean vs. second-order aggregation ($d = 3$).

n	Mean agg. gap	2nd-order gap	Ratio
10	0.0389	0.0240	$1.6\times$
50	0.0082	0.0016	$5.1\times$
100	0.0052	0.00053	$9.9\times$
200	0.0030	0.00015	$20.9\times$
500	0.0014	0.000028	$49.5\times$

Fitting power laws, the mean aggregation gap decays as $n^{-0.81}$ (consistent with $O(1/n)$ up to the linear-predictor lower bound), while the second-order gap decays as $n^{-1.75}$ (consistent with $O(1/n^2)$).

C.5 Scaling with representation dimension

Table 7 shows how the gap depends on d for fixed $n = 50$. The mean aggregation gap initially grows with d (more second-order structure to miss) then declines (eigenvalue decay reduces the sensitivity). The second-order gap remains small throughout, confirming that it is a sufficient statistic for the head kernel matrix.

Table 7: Amortization gap vs. representation dimension ($n = 50$).

d	Rep. dim (mean)	Posterior d.o.f.	Mean agg. gap	2nd-order gap
1	1	1	0.00250	0.00005
2	2	3	0.00696	0.00055
3	3	6	0.00831	0.00161
5	5	15	0.00243	0.00163
8	8	36	0.00167	0.00169

At $d = 8$ the two methods converge, because the eigenvalues λ_j for $j > 3$ are small enough under the SE kernel ($\ell = 0.3$) that the corresponding second-order terms contribute negligibly, and both representations capture the relevant structure equally well. This is consistent with the effective dimension scaling predicted by Proposition 2.

D Empirical Verification of Theorem 1

We verify the two-component decomposition of Theorem 1 on a trained LNP. The theorem predicts

$$\mathbb{E}_C [\text{Var}_{y|X} [\sigma_{\text{LNP}}^2(x_*; C)]] \leq L_\Sigma^2 B_w^4 B_\psi^2 \left(\underbrace{\frac{\sigma_\epsilon^2}{n}}_{\text{noise } \delta_\epsilon} + \underbrace{\kappa_k}_{\text{signal } \delta_f} \right), \quad (64)$$

with the signal component $O(1)$ and irreducible, and the noise component $O(1/n)$ and vanishing.

D.1 Setup

We train a small LNP on samples from $\mathcal{GP}(0, k_{\text{SE}})$ with lengthscale $\ell = 0.2$, signal variance 1, and observation noise $\sigma_\epsilon^2 = 0.05$ on $\mathcal{X} = [0, 1]$. The architecture follows Definition 1 but uses MLP components rather than the analytical idealizations of Assumptions 2 and 4:

- Encoder: $h(x, y) = \text{MLP}([x, y]) \rightarrow \mathbb{R}^{64}$ (two hidden layers of width 64, ReLU). Notably, h is non-affine in y , so Assumption 2 is violated.
- Aggregation: mean.
- Recognition: $(\mu_z, \log \sigma_z^2) = (\text{Linear}(r_C), \text{Linear}(r_C))$ with $z \in \mathbb{R}^{32}$ and diagonal covariance.

- Decoder: $\text{MLP}([x_*, z]) \rightarrow (\mu, \text{softplus}(\nu))$ with $\sigma_d^2 = 0.05$ floor on the output variance.

Training uses the standard NP ELBO over context/target splits with sizes drawn uniformly from $\{5, \dots, 49\}$, batch size 16, Adam at learning rate 10^{-3} , 4000 steps. Using a non-affine encoder is intentional: it tests whether the qualitative two-component prediction of [Theorem 1](#) survives the violation of [Assumption 2](#).

D.2 Protocol

To isolate the two components of (64) we evaluate the trained model under two label-resampling protocols at fixed context locations X :

Full ($\delta_f + \delta_\epsilon$). For each context location set X , draw $f \sim \mathcal{GP}$ and $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2 I)$ independently across $R = 400$ resamples; set $y^{(r)} = f^{(r)}(X) + \epsilon^{(r)}$. Both label-dependent components contribute.

Noise-only (δ_ϵ). Draw a single GP function f_0 , fix it across resamples, and resample only $\epsilon^{(r)}$; set $y^{(r)} = f_0(X) + \epsilon^{(r)}$. Only the noise component contributes.

For each protocol we compute the empirical variance of $\sigma_{\text{LNP}}^2(x_* = 0.5; X, y^{(r)})$ across the R label resamples, then average over 30 independent context location sets X (with $x_i \sim \text{Uniform}[0, 1]$). The marginal predictive variance is computed as $\mathbb{E}_z[\text{Var}(y_* | z)] + \text{Var}_z[\mathbb{E}(y_* | z)]$ using 64 samples from $q(z | C)$.

D.3 Results

Table 8: Conditional variance of the LNP predictive variance under the two protocols. The Full protocol measures $\delta_f + \delta_\epsilon$; the Noise-only protocol measures δ_ϵ alone. The Floor/Noise ratio quantifies how much of the contamination is irreducible at each n .

n	Full ($\delta_f + \delta_\epsilon$)	Noise only (δ_ϵ)	Floor/Noise ratio
5	2.98×10^{-1}	1.52×10^{-2}	20×
10	2.42×10^{-1}	1.19×10^{-2}	20×
20	2.30×10^{-1}	3.47×10^{-3}	66×
50	2.25×10^{-1}	2.59×10^{-3}	87×
100	2.21×10^{-1}	1.64×10^{-3}	134×
200	2.08×10^{-1}	1.06×10^{-3}	197×
500	2.08×10^{-1}	7.40×10^{-4}	282×
1000	2.14×10^{-1}	8.38×10^{-4}	256×

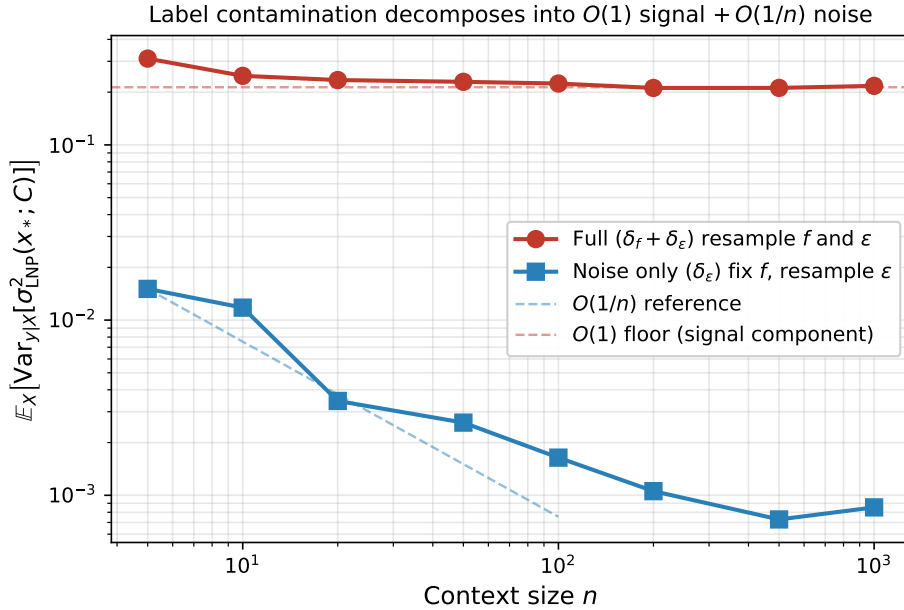


Figure 2: Empirical verification of [Theorem 1](#). The Full protocol (red) plateaus at an $O(1)$ floor across two orders of magnitude in n . The Noise-only protocol (blue) decays as $1/n$ on the early- n side, consistent with the σ_ϵ^2/n term, before saturating at the latent- z Monte Carlo floor of the variance estimator.

Fitting power laws $\text{Var} \propto n^\alpha$ on the early- n range $n \in \{5, 10, 20, 50, 100\}$ where the noise contribution dominates the latent- z Monte Carlo floor:

- Noise-only protocol: $\alpha = -0.78$ (theory: -1).
- Full protocol: $\alpha = -0.09$ (theory: $\rightarrow 0$).

D.4 Interpretation

Signal floor. The full protocol plateaus near 0.21 across $n \in [5, 1000]$, providing direct empirical evidence for the $O(1)$ signal floor predicted by [Theorem 1](#). This contamination does not vanish with increasing context size. At $n = 1000$, label dependence still injects variance of order σ_{LNP}^2 itself (which is $O(10^{-1})$ at $x_* = 0.5$ on this trained model).

Noise decay. The noise-only protocol decays from 1.5×10^{-2} at $n = 5$ to $\sim 8 \times 10^{-4}$ at $n = 1000$, consistent with the predicted σ_ϵ^2/n scaling. The fitted slope of -0.78 is shallower than the theoretical -1 , and this is consistent with a finite-sample floor introduced by the latent- z Monte Carlo expectation in the marginal predictive (computed here with 64 samples per evaluation), which lower-bounds the measurable variance at roughly 10^{-3} on this network. The early- n portion, where the signal-vs-MC ratio is largest, gives the cleanest test of the rate.

Regime change. The Floor/Noise ratio grows from $20\times$ at $n = 5$ to $256\times$ at $n = 1000$, demonstrating exactly the regime change the theorem predicts: at small n both components contribute non-trivially, while at large n the signal floor entirely dominates the contamination.

Robustness to [Assumption 2](#). The trained encoder is a non-affine MLP, so [Assumption 2](#) is violated and the precise constants in [Theorem 1](#) do not apply. The qualitative two-component decomposition nonetheless appears robustly. This supports the interpretation of [Appendix A.3](#). The structural mechanism (label contamination through any encoder that mixes x and y inputs into the variance pathway) is general, even when the analytical decomposition $r_C = \bar{\phi}_X + \delta_y$ does not hold exactly.

E Empirical Verification of [Assumption 4](#)

[Assumption 4](#) posits that the encoder learns the first d kernel eigenfunctions, which is the optimal d -dimensional kernel approximation. This is the strongest assumption in our analysis, as the standard NP training objective contains no explicit term encouraging Mercer alignment. We test whether trained NP encoders approximately recover the Mercer eigenfunction subspace.

E.1 Setup

For each of three SE kernels on $[0, 1]$ with lengthscales $\ell \in \{0.1, 0.2, 0.5\}$ (signal variance 1, noise $\sigma_\epsilon^2 = 0.05$), we train the same MLP-based LNP architecture used in [Appendix D](#) (encoder MLP $\rightarrow \mathbb{R}^{64}$, mean aggregation, latent $z \in \mathbb{R}^{32}$, MLP decoder) via the standard ELBO for 4000 steps.

Ground-truth eigenfunctions. The top- $d_{\text{max}} = 16$ Mercer eigenfunctions $\{e_j\}$ and eigenvalues $\{\lambda_j\}$ of each kernel under the uniform measure on $[0, 1]$ are computed via a Nyström decomposition on a dense grid of $N = 2000$ equispaced points, with eigenfunctions normalized to unit L^2 -norm.

Encoder features. The trained encoder $h(x, y) : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}^{64}$ produces a 64-dimensional feature vector for each input pair. To isolate the x -dependent subspace we evaluate the encoder in two modes:

$y = 0$: Set $y = 0$ for all inputs, giving the direct readout of the x -pathway: $h(x, 0)$.

Marginal: Average $h(x, y)$ over $y \sim p(y | x)$ under the GP prior, approximated with 200 joint GP samples. This is the theoretically cleaner “location-only” representation $\bar{\phi}(x) = \mathbb{E}_{y|x}[h(x, y)]$.

Subspace alignment. For each $d \in \{1, \dots, 16\}$, we measure how well the d -dimensional Mercer eigenfunction subspace $\text{span}(e_1, \dots, e_d)$ is contained in the 64-dimensional encoder feature subspace. Both subspaces are orthonormalized with respect to the L^2 -uniform inner product, and we compute the principal angles between them. The minimum cosine $\cos \theta_{\min}$ across the d principal angles equals 1 if and only if the Mercer subspace is exactly contained in the encoder subspace; values near 1 indicate near-containment.

Per-axis recovery. For each eigenfunction e_j individually, we compute $R_j^2 = \|P_Q e_j\|^2 / \|e_j\|^2$, where P_Q is the orthogonal projector onto the encoder subspace. $R_j^2 = 1$ means e_j is exactly recoverable from the encoder features.

E.2 Results

Table 9: Per-axis R^2 for the top eigenfunction recovery by the trained encoder ($y = 0$ mode) across three SE kernels. Bold indicates $R^2 \geq 0.95$.

j	$\ell = 0.1$		$\ell = 0.2$		$\ell = 0.5$	
	λ_j	R_j^2	λ_j	R_j^2	λ_j	R_j^2
1	2.4e-1	1.00	4.4e-1	1.00	7.7e-1	1.00
2	2.1e-1	1.00	3.0e-1	1.00	2.0e-1	1.00
3	1.8e-1	1.00	1.6e-1	0.99	2.7e-2	1.00
4	1.3e-1	0.98	6.8e-2	0.98	2.3e-3	1.00
5	9.4e-2	0.99	2.4e-2	0.95	1.5e-4	0.99
6	6.2e-2	0.94	6.8e-3	0.91	7.4e-6	0.97
7	3.7e-2	0.96	1.7e-3	0.90	3.1e-7	0.93
8	2.1e-2	0.90	3.6e-4	0.79	1.1e-8	0.79
9	1.1e-2	0.81	6.9e-5	0.82	$< 10^{-8}$	0.02
10	5.6e-3	0.89	1.2e-5	0.63	$< 10^{-8}$	0.03

Table 10: Minimum cosine of principal angles between the top- d Mercer subspace and the full encoder subspace ($y = 0$ mode). $\cos \theta_{\min} = 1$ means exact containment.

d	$\ell = 0.1$	$\ell = 0.2$	$\ell = 0.5$
1	1.000	1.000	1.000
2	1.000	1.000	1.000
3	0.999	0.993	1.000
4	0.989	0.986	0.999
5	0.989	0.960	0.996
6	0.958	0.918	0.985
7	0.948	0.887	0.959
8	0.913	0.784	0.883

E.3 Interpretation

Approximate Mercer alignment. The trained encoder recovers the top Mercer eigenfunctions with high fidelity with $R^2 > 0.95$ for all eigenfunctions whose eigenvalues exceed approximately 10^{-2} (Table 9). The subspace containment analysis (Table 10) confirms this at the subspace level, with $\cos \theta_{\min} > 0.9$ for $d \leq 6$ ($\ell = 0.2$), $d \leq 8$ ($\ell = 0.1$), and $d \leq 7$ ($\ell = 0.5$).

Alignment tracks the spectrum. Recovery degrades where the eigenvalue spectrum becomes negligible relative to the observation noise $\sigma_\epsilon^2 = 0.05$. For $\ell = 0.5$, the spectrum drops below 10^{-8} at $j = 9$, and R^2 simultaneously collapses from 0.79 to 0.02. For $\ell = 0.1$, the slower spectral decay means more eigenfunctions carry signal, and the encoder accordingly recovers more of them before alignment degrades. This is consistent with the ELBO training objective implicitly encouraging the encoder to capture the directions of maximum variation under the GP prior which are the top Mercer eigenfunctions.

The ELBO encourages Mercer alignment. No explicit spectral regularization is applied during training. The alignment arises because the top eigenfunctions are the features that minimize prediction error under the GP prior. GP samples vary primarily along the directions e_j with large λ_j , so an encoder trained to predict well on GP samples recovers these directions. This provides empirical justification for Assumption 4 in the regime where the NP is trained on samples from a fixed GP prior, which is the regime our analysis targets.

Implications for the bottleneck rates. Since the trained encoder approximately spans the top- d Mercer subspace for d up to the effective spectral dimension of the kernel, the bottleneck rates in Corollary 1 are not properties of an idealized encoder but are approximately predictive of trained-NP behavior. The effective aligned dimension tracks the kernel lengthscale as the theory predicts. Smoother kernels ($\ell = 0.5$, faster eigenvalue decay) require fewer encoder dimensions for high-fidelity alignment than rougher kernels ($\ell = 0.1$, slower decay).