Predictive Variance Decides: Region-Aware Gaussian Processes Should Adapt the Noise, Not the Approximation

Srikar M.K.
Northeastern University
manda.k@northeastern.edu
srikarmks@gmail.com

Alekhya Bulusu
Northeastern University
bulusu.a@northeastern.edu

October 20, 2026

Abstract

Region-aware Gaussian processes adapt the model to different parts of the input space. One appealing form allocates approximation budget unevenly, routing "important" regions to high-fidelity sparse approximations and the rest to cheap ones. We find that, for the approximations we test (random Fourier features and Nyström, each with its own homoscedastic noise), this form barely changes the uncertainty a region receives, and that a second form, adapting the noise model by region, does change it.

The reason is the predictive variance decomposition: these approximations alter only the latent term, while the noise term, fitted separately by each model, comes out nearly the same. Varying Nyström rank on five benchmarks changes test RMSE by up to 155% but mean predictive standard deviation by at most 12%; on four constructed datasets, four models' per-region predictive standard deviations agree within 3% in every region and seed while the true noise varies 50×. An oracle router given the true generative partition gains nothing on average (+0.0003 ECE). Making the noise term input-dependent instead, with a latent-variance-corrected out-of-fold noise estimate, pays on the larger datasets with clearly input-dependent residual variance: after Holm correction regional noise improves ECE on the same three datasets in both of two disjoint sets of ten splits (four in the second set, five when pooled) and worsens it on none, though on the smallest dataset it worsens NLL and CRPS; a heteroscedastic GP improves NLL on seven to nine. The result survives six adversarial checks, including recalibrating the global baseline and a sweep over training-set size.

We also report a measurement hazard: an opt-in fast predictive-variance setting in a widely used GP library is inert below a training-set size threshold that never appears in user code, and above it perturbs predictive standard deviations by a median of ∼100%, inflating measured ECE by a factor of three to ten at $n = 4000$. Finally we report the research process (Section 7): five of six mechanisms we proposed did not survive, and five headline numbers, including an earlier, opposite conclusion of this paper, turned on how a *baseline* or estimator was configured. We distil these into a checklist for GP calibration studies.

1 Introduction

Exact Gaussian process regression costs $O(n^3)$, and a large literature reduces this with sparse approximations [Rasmussen and Williams, 2006, Quiñonero-Candela and Rasmussen, 2005, Titsias, 2009]. Most such methods apply a single approximation uniformly across the input space. It is natural to ask whether the budget should instead be allocated unevenly: regions that are sparsely

sampled, or where the model is uncertain, might warrant a more faithful approximation than regions that are dense and easy.

We call this *region-aware approximation*. The premise is appealing and the mechanism is intuitive. This paper shows the mechanism does not deliver on the premise, for a reason that does not depend on the partitioning criterion and holds for every homoscedastic approximation we tested: approximation fidelity and predictive uncertainty are nearly orthogonal. The same reason points to what does work. The term of the predictive variance that can differ by region is the noise, and adapting it by region improves calibration where the data warrant it.

Contributions.

1. **Fidelity and uncertainty are nearly orthogonal** (Sections 4 and 5.3). Changing Nyström rank moves accuracy far more than predictive uncertainty, and models of nominally different fidelity emit per-region predictive standard deviations within 3% of one another on our constructed datasets, so routing among them barely changes a region's uncertainty; oracle routing gains nothing on average, and the importance criterion of the method we audited is significantly worse than random routing on two of five benchmarks.
2. **Input-dependent noise pays on the larger datasets with clearly input-dependent residual variance** (Section 6). With a latent-variance-corrected out-of-fold noise estimate, regional noise beats a single global noise parameter on ECE on airfoil, sarcos and calhousing in both seed sets after Holm correction and under a resampling-corrected test, and survives global recalibration (uncorrected) and, where tested, a sweep over training-set size; a two-stage heteroscedastic GP improves NLL on most datasets.
3. **A measurement hazard** (Section 3) that silently corrupts calibration metrics for "exact" GPs above a library default, by up to a factor of ten, and is controlled by a setting other than the one that names it.
4. **A record of what failed and what caught it** (Section 7), including a reversal of this paper's own earlier conclusion, distilled into a checklist for GP calibration studies.

2 Background

Sparse GP approximations. Given data $\{(x_i, y_i)\}_{i=1}^n$ with $y = f(x) + \varepsilon$, $\varepsilon \sim \mathcal{N}(0, \sigma_n^2)$, a GP prior $f \sim \mathcal{GP}(0, k)$ yields a predictive distribution at x_* with variance

$$\mathbb{V}[y_*] \;=\; \underbrace{k(x_*, x_*) - k_*^\top (K + \sigma_n^2 I)^{-1} k_*}_{\text{latent, approximation-dependent}} + \underbrace{\sigma_n^2}_{\text{input-independent}} . \tag{1}$$

Random Fourier Features [Rahimi and Recht, 2007] and Nyström / subset-of-regressors [Williams and Seeger, 2001, Smola and Schölkopf, 2000] replace the first term with a rank-m surrogate and keep the second a single constant. (FITC [Snelson and Ghahramani, 2006] and PITC [Quiñonero-Candela and Rasmussen, 2005] are an exception: their (block-)diagonal correction acts as input-dependent noise.) Section 4 shows this is what dooms region-aware approximation.

Calibration metrics. For regression we use the expected calibration error over central predictive intervals,

$$\mathrm{ECE} = \frac{1}{B} \sum_{b=1}^{B} \Big| \tfrac{1}{n} \textstyle\sum_i \mathbf{1}\big[|y_i - \mu_i| \le z_{(1+c_b)/2}\, \sigma_i\big] - c_b \Big|, \tag{2}$$

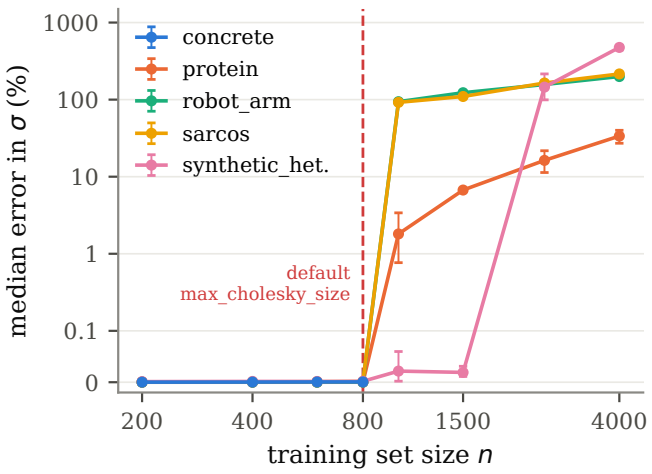


Figure 1: Median relative error in the predictive standard deviation against exact inference, as a function of training set size, with the range over three seeds. The error is below 0.002% at and below the library's default Cholesky threshold. Above it, it appears at once on robot_arm and sarcos, grows steadily on protein, and on synthetic_heteroscedastic stays below 0.6% on every seed until $n = 2500$; concrete has only 1030 points and does not extend above the threshold.

with c_b spanning $[0.1, 0.9]$, alongside Gaussian negative log-likelihood. Both depend on σ_i, which is what makes Section 3 consequential.

3 A measurement hazard in approximate predictive variance

Before reporting calibration comparisons we must establish that they measure the model rather than the inference path.

GPyTorch [Gardner et al., 2018] provides `fast_pred_var()`, an implementation of LOVE [Pleiss et al., 2018] that approximates the predictive covariance, and separately falls back from Cholesky to iterative solves when the training set exceeds `max_cholesky_size` (default 800). We fit one model per cell and vary *only* the inference path, clearing the model's prediction cache before each path: GPyTorch caches the predictive covariance on first use, and a later call with different settings silently reuses it. All measurements use GPyTorch 1.15.2 with linear_operator 0.6.1.

Table 1: Same fitted model; only the inference path varies. Error is the median over test points of $|\sigma_{\text{mode}} - \sigma_{\text{exact}}|/\sigma_{\text{exact}}$, averaged over three seeds. Over 20 cells with $n \leq 800$ the maximum error is 0.0018%. Generated from `results/inference/corruption.json`.

dataset	n_{train}	ECE default	ECE exact	median σ error
concrete	800	0.0434	0.0434	0.00%
protein	4000	0.1076	0.0368	33.66%
robot_arm	4000	0.2595	0.0848	199.38%
sarcos	4000	0.3073	0.0307	215.30%
synthetic	4000	0.4054	0.0619	475.53%

A four-way decomposition isolates the cause. With the default settings the median error in σ across the 16 cells with $n > 800$ is 101.92%. Raising `max_cholesky_size` above n alone removes

it (0.00%; at most 0.01% in any cell), because LOVE then builds its low-rank factorisation from a Cholesky decomposition rather than Lanczos iterations. Disabling `fast_pred_var()` alone reduces the median to 5.62% but leaves the error of the iterative solves, up to 38% (sarcos, $n = 1000$). The approximation is therefore silently inert on small data and silently severe on large data, with no change in user code, and the setting that controls it is the Cholesky threshold rather than the flag that names it. GPyTorch's fits are not bit-reproducible across runs: in two further independent runs of this experiment the median error above the threshold was 100.3% and 102.0%, the maximum at or below it 0.0018%, and the ECE ratio at $n = 4000$ ranged from 2.9 to 10.2; individual cells vary more (`FINDINGS.md`).

Implication. Reported ECE or NLL for "exact" GPs trained above $n = 800$ in this library may describe an approximation. Raising the Cholesky threshold above n suffices for the predictive variance; we recommend that calibration studies also disable fast predictive variance, and report both settings.

4 Fidelity and uncertainty are nearly orthogonal

Equation (1) suggests that if σ_n^2 dominates, then changing the rank of the approximation, which affects only the first term, barely moves the predictive standard deviation. We test this in two ways: by varying fidelity over a wide range on real data with everything else fixed, and by measuring, region by region, how much the predictive uncertainty of differently approximated models differs on data with known regional structure.

Accuracy moves; uncertainty does not. Table 2 varies only the Nyström rank, from 25 to 1600 (400 on concrete), with kernel and noise hyperparameters fitted once by exact marginal likelihood and shared by every rank (`src/mechanism.py`). Over this range the lowest-rank test RMSE is 21–155% above the highest-rank one, while mean predictive standard deviation differs by 0–12%. Low-rank models are less accurate but not correspondingly less certain; their z-scores are over-dispersed (Table 10, first row).

Table 2: Nyström rank varied with hyperparameters held fixed, five seeds. Changes are relative to the highest rank. Generated from `results/mechanism/rank_sweep.json`.

dataset	ranks	RMSE (lowest)	RMSE (highest)	RMSE change	mean σ change
concrete	25–400	9.208	6.149	50%	12%
protein	25–1600	5.418	4.494	21%	4%
robot_arm	25–1600	0.1942	0.07611	155%	11%
sarcos	25–1600	8.363	3.292	154%	7%
synthetic	25–1600	0.1681	0.139	21%	0%

Region by region. We fit four models, RFF with 1000 features, Nyström with 500 and 1500 landmarks, and an exact GP, to four constructed datasets with known regional structure, including `hetero_extreme`, whose noise standard deviation varies 50× across three regions by construction. Each model fits its own hyperparameters, including σ_n, by its own marginal likelihood.

In every region of all four datasets and on every seed, the four models' mean predictive standard deviations lie within 2.7% of one another. On these dense two-dimensional datasets the four models

Table 3: Per-region ECE of four models of nominally different fidelity, evaluated at the same inputs, mean over three seeds. *ECE spread* and σ *spread* are computed within each seed (the range across the four models of the per-region ECE, and of the per-region mean predictive standard deviation relative to its mean), and the largest over seeds is shown. On `hetero_extreme` the true noise standard deviation runs from 0.020 to 1.000 across regions while every model's mean predictive standard deviation lies in $[0.563, 0.592]$ in every region and seed. Generated by `src/orthogonality.py` and `src/paper_tables.py`.

dataset	region	RFF	Nyström-500	Nyström-1500	exact GP	ECE spread	σ spread
`hetero_extreme`	0	0.4973	0.4986	0.4986	0.4985	0.0015	2.64%
	1	0.3803	0.3811	0.3811	0.3811	0.0047	2.64%
	2	0.1804	0.1797	0.1797	0.1787	0.0092	2.64%
`varying_smoothness`	0	0.2904	0.2911	0.2911	0.2912	0.0019	0.55%
	1	0.0281	0.0285	0.0285	0.0290	0.0019	0.55%
	2	0.1357	0.1356	0.1356	0.1348	0.0020	0.55%
`varying_density`	0	0.0293	0.0281	0.0283	0.0286	0.0089	2.47%
	1	0.0363	0.0333	0.0327	0.0327	0.0088	2.30%
	2	0.0054	0.0053	0.0053	0.0063	0.0038	1.93%
`combined`	0	0.4324	0.4461	0.4539	0.4510	0.0266	0.91%
	1	0.3500	0.3261	0.3300	0.2860	0.0742	0.95%
	2	0.0354	0.0340	0.0339	0.0325	0.0046	1.63%

are all accurate (their per-region RMSE agrees within 3% except in regions 0 and 1 of `combined` and the near-noiseless region 0 of `hetero_extreme`, where absolute errors are below 0.04), the latent term is small, and each model's fitted σ_n settles on the pooled noise level; this is the regime in which Eq. (1) predicts orthogonality, and Table 2 shows it persists when accuracy differs substantially. A region-aware router chooses, region by region, among such models. Because each is homoscedastic, it can shift a region's uncertainty only by the margin by which their global noise estimates differ, a few percent, against the 50× regional variation that would be needed.

What a router can exploit. A router gains only where its candidate models differ in per-region calibration. Measured on held-out data within a seed, that spread is at most 0.0092 on three of the four datasets. `combined`, whose regions differ in function complexity and sampling density as well as noise, is the exception (up to 0.0266 in region 0 and 0.0742 in region 1), and it is the one constructed dataset where a non-random router improves on a single model consistently across seeds (Table 6; on `varying_density` only the random-routing control does by more than 0.0001): the true-partition oracle by 0.0009 and importance routing by 0.0022 ECE, on all three seeds, and the learned oracle by 0.0021 on average. These margins are paired, within-seed differences and lie below the unpaired ECE noise floor of Section 5. Their source is instructive. The oracle keeps the exact GP, the uniform choice, in region 1, where the spread is largest, and gains by switching region 0 to RFF. Region 1's spread is not a difference in uncertainty at all: every model there predicts $\sigma \approx 0.57$ against a true noise of 0.11, and the exact GP's lower ECE comes with a *higher* RMSE (0.227 against 0.170 for RFF), its larger errors filling intervals that are too wide for all four. The spread in calibration that routing can harvest is therefore small even where it exists, and partly reflects the mean rather than the variance.

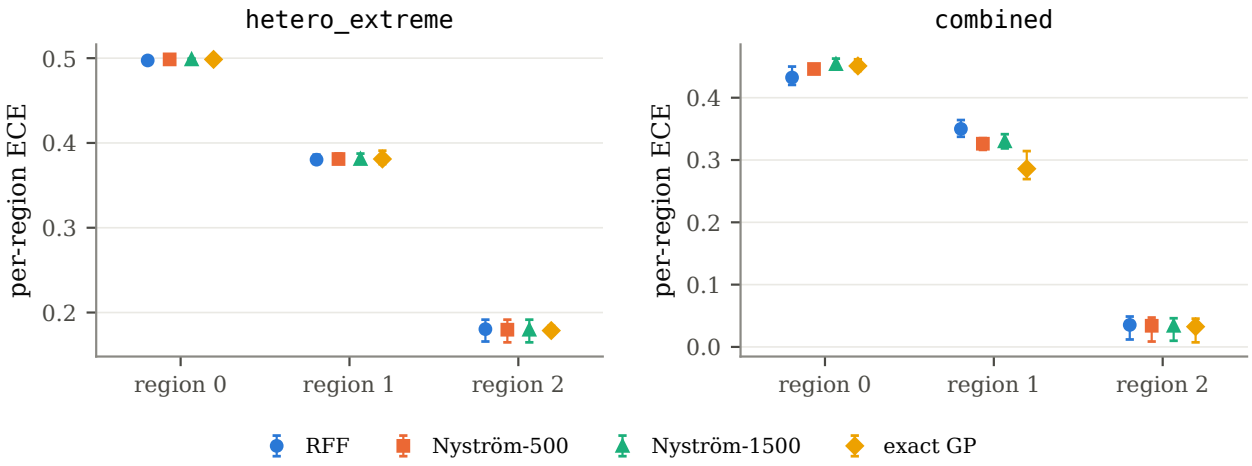


Figure 2: Per-region ECE for four models of nominally different fidelity; bars are the range over three seeds. On `hetero_extreme` (left) the four models are indistinguishable in every region. On `combined` (right) they separate in regions 0 and 1, the only place in our constructed data where a router has anything to choose between.

Scope. We test random Fourier features and Nyström/subset-of-regressors approximations, each with a homoscedastic likelihood, at $n \leq 4800$. Approximations whose correction acts as input-dependent noise, notably FITC [Snelson and Ghahramani, 2006] and PITC [Quiñonero-Candela and Rasmussen, 2005], are outside this argument.

5 Empirical study

5.1 Protocol

All methods share one split per (dataset, seed). Exact-GP hyperparameters are fitted by exact marginal likelihood on a random subsample of at most 1500 training points, and sparse models fit their own hyperparameters, including σ_n, by their own marginal likelihood on at most 2000; model and configuration choices are made on a validation split; the test split is read once. A single ECE estimator is used for every method, metrics are reported in original target units, and exact predictive variance is used throughout (Section 3).

Datasets. Table 4 lists the eleven datasets. Eight are measured, two are simulators (kin8nm, which we call robot_arm, and energy), and one, synthetic_heteroscedastic, was constructed for the original study. Four further two-dimensional datasets with known regional structure (`hetero_extreme`, `varying_smoothness`, `varying_density`, `combined`) are generated by `src/structured_data.py`; their regions are three equal-width bands of x_1 (cuts at $\pm 1/3$). Noise changes in steps at the band boundaries, while function complexity ramps smoothly in x_1 and density is concentrated in a Gaussian blob. In the noise-model comparison each dataset is subsampled to at most 2500 points and split 60/20/20, so most datasets train on 1500 points but the four smallest train on far fewer. Small test sets bound how precisely ECE can be measured: for a perfectly calibrated Gaussian our estimator has expectation 0.015 at 500 test points and 0.042 at 62 (`src/data_checks.py`), so absolute ECE values below these are not meaningful; paired differences between arms on the same split can resolve less. Known data issues are documented in

`data/README.md`: SARCOS test inputs also appear in its training file (0.6% of test points have an exact training twin at $n_{\text{train}} = 1500$), wine_white has many duplicate rows (13% of test points have a twin), and calhousing targets are censored at 5.0 (4.7%).

Table 4: Datasets. $n_{\text{train}}/n_{\text{val}}/n_{\text{test}}$ are those of the noise-model comparison (Table 7).

dataset	source	size	d	train / val / test
airfoil	Brooks and Pope [1989]	1503	5	901 / 301 / 301
calhousing	Pace and Barry [1997]	20640	8	1500 / 500 / 500
concrete	Yeh [1998]	1030	8	618 / 206 / 206
energy (simulated)	Tsanas and Xifara [2012]	768	8	460 / 154 / 154
powerplant	Tüfekci [2014]	9568	4	1500 / 500 / 500
protein	Rana [2013]	45730	9	1500 / 500 / 500
robot_arm (kin8nm, simulated)	Rasmussen et al. [1996]	8192	8	1500 / 500 / 500
sarcos	Vijayakumar and Schaal [2000]	48933	21	1500 / 500 / 500
synthetic_heteroscedastic	constructed	5000	2	1500 / 500 / 500
wine_white	Cortez et al. [2009]	4898	11	1500 / 500 / 500
yacht	Gerritsma and Onnink [1981]	308	6	184 / 62 / 62

5.2 Datasets are screened, not assumed

Because the hypothesis concerns input-dependent structure, we measure it rather than assume it. Table 5 reports het, the ratio of the 90th to 10th percentile of kNN-smoothed out-of-sample squared residuals, over ten seeds, and compares it with a permutation null. Every dataset but `varying_density` is significantly above its null on most seeds, but the four lowest real datasets (protein, robot_arm, powerplant, wine_white) are within 1.45× of it.

Table 5: Measured input-dependent residual variance over ten seeds: het (mean and sd), the mean of its permutation null (residuals shuffled over the inputs), and their ratio. The null is not 1: it depends on the residuals' tails and the test-set size. On small datasets a single seed can mislead (yacht ranges from 3.4 to 20.1). Generated by `src/measure_hetero_seeds.py` and `src/measure_hetero_null.py`.

origin	dataset	het	sd	null	het / null
constructed	hetero_extreme	1626.84	160.00	3.36	486.30
constructed	combined	22.85	6.31	1.96	11.61
constructed	varying_smoothness	22.54	2.49	2.02	11.19
simulated	energy	6.67	1.72	2.49	2.78
measured	airfoil	5.53	1.22	2.55	2.21
measured	yacht	9.25	5.42	4.09	2.20
constructed	synthetic_heteroscedastic	4.00	0.41	1.96	2.04
measured	concrete	4.17	0.85	2.28	1.83
measured	sarcos	4.76	0.53	2.64	1.81
measured	calhousing	4.72	0.68	2.70	1.75
measured	protein	2.69	0.27	1.86	1.45
simulated	robot_arm	2.66	0.21	2.05	1.30
measured	powerplant	2.55	0.14	2.07	1.24
measured	wine_white	2.51	0.38	2.05	1.22
constructed	varying_density	1.83	0.08	1.78	1.03

5.3 Routing gains nothing, even with the true partition

Table 6: ECE on data whose regional structure is true by construction. *true oracle* uses the true generative partition and selects the best model per region on validation data. Because selection is on validation, it bounds other routers only approximately on test, and is beaten by another router on all four datasets. Best per row in bold.

dataset	uniform best	true oracle	learned oracle	importance	random
hetero_extreme	**0.2299**	0.2315	0.2302	0.2322	0.2316
varying_smoothness	**0.0656**	0.0658	0.0659	0.0661	0.0659
varying_density	0.0069	0.0075	0.0070	0.0069	**0.0065**
combined	0.0525	0.0515	0.0504	**0.0503**	0.0537

The mean difference between the true oracle and a single model everywhere is +0.0003 ECE (+0.0016 on `hetero_extreme`). On the five benchmarks of the original study, with the oracle routing over the learned partition, the corresponding figure is +0.0001, with the oracle losing on four of five. Importance-based routing, the criterion region-aware methods such as the one we audited use, is worse than a proportion-preserving random-routing control on three of the five, significantly on robot_arm and sarcos (paired t-test over five seeds, Holm-adjusted over the five benchmarks, $p = 0.011$ and $p = 0.007$; `src/significance.py`), and nominally significantly better only on protein, by 0.0009 ($p = 0.048$, not significant after correction). A Wilcoxon test cannot reach $p < 0.0625$ with five seeds, hence the t-test here. The oracle selects per region on validation ECE, from as few as about 50 validation points per region (`varying_density`), so it bounds routing only as well as that selection resolves.

6 Making the noise model input-dependent

Equation (1) indicates the constructive move: the term that varies across regions is σ_n^2, so that is what should be made input-dependent. We compare three arms on eleven datasets: a single global noise parameter fitted by marginal likelihood; a two-stage heteroscedastic GP in the spirit of Goldberg et al. [1997] and Kersting et al. [2007], whose noise variance is a kNN smoothing of per-point noise evidence with zero or one refinement pass (re-estimating the evidence under the first-pass noise); and *regional* noise, a piecewise-constant noise variance over a partition of the input space by local residual variance. Both input-dependent arms use the same latent-variance-corrected out-of-fold noise evidence: the squared out-of-fold residual minus the fold model's latent predictive variance at that point. Pointwise this is unbiased for $\sigma^2(x_i)$ only if the fold model is well specified, that is, if its latent variance equals the actual squared error of its mean. Region means taken over a partition built from the same evidence are biased away from the overall mean, and are floored at $0.05\,\sigma_n^2$; the floor binds for 0 to 59% of test points depending on the dataset (`src/noise_controls.py`), and on yacht the mean evidence is negative in nine of ten seeds, so the floor binds for most of its test points (59% on average). Both arms select their configuration on validation NLL (six configurations for the het-GP, twelve for regional noise) and never see test labels. Significance is a paired two-sided Wilcoxon signed-rank test against the global arm with Holm correction over the eleven datasets (`src/significance.py`).

Input-dependent noise pays where there is input-dependence to model. Regional noise is the best-calibrated arm on six of eleven datasets. After Holm correction it is significantly better

Table 7: ECE by noise model, seeds 0–9, ordered by het ratio relative to its permutation null (Table 5). $^{\uparrow}/^{\downarrow}$: better/worse than global noise at Holm-adjusted $p < 0.05$ over the eleven datasets. Generated by `src/hetero_fair.py`, `src/significance.py` and `src/paper_tables.py`.

dataset	origin	n_{train}	het / null	global	het-GP	regional
energy	simulated	460	2.78	0.0385	**0.0336**	0.0392
airfoil	measured	901	2.21	0.0784	$0.0401^{\uparrow}$	$\mathbf{0.0354}^{\uparrow}$
yacht	measured	184	2.20	**0.0648**	0.1303	0.0679
synth_het	constructed	1500	2.04	0.0242	**0.0152**	0.0153
concrete	measured	618	1.83	0.0400	0.0294	**0.0267**
sarcos	measured	1500	1.81	0.0815	0.0699	$\mathbf{0.0477}^{\uparrow}$
calhousing	measured	1500	1.75	0.0693	$0.0450^{\uparrow}$	$\mathbf{0.0284}^{\uparrow}$
protein	measured	1500	1.45	0.0189	0.0250	**0.0175**
robot_arm	simulated	1500	1.30	0.0163	0.0167	**0.0156**
powerplant	measured	1500	1.24	**0.0197**	0.0204	0.0201
wine_white	measured	1500	1.22	**0.0216**	0.0223	0.0216
best-arm count (ECE)				3	2	**6**
best-arm count (NLL)				2	**9**	0
best-arm count (CRPS)				1	**9**	1

than global noise on ECE on airfoil, sarcos and calhousing in seeds 0–9, and on the same three plus synthetic_heteroscedastic in the disjoint seeds 10–19; concrete and synthetic_heteroscedastic are nominally significant in seeds 0–9 ($p = 0.014$, 0.037) and all five are Holm-significant when the twenty splits are pooled. It is significantly worse on ECE on no dataset in any of the three analyses, but on yacht, the smallest dataset, it is significantly worse on NLL and CRPS (seeds 10–19 and pooled). The heteroscedastic GP is best on NLL on nine of eleven datasets and significantly better than global noise on NLL on seven after correction in seeds 0–9 (nine in seeds 10–19), and significantly worse on yacht.

The datasets with a robust ECE gain have het ratios at least 1.75 times their permutation null and train on at least 901 points. The four with het ratios closest to their null (protein, robot_arm, powerplant, wine_white, at most 1.45 times) show no significant difference for regional noise, and there the global arm's ECE already sits near the estimator's noise floor (Section 5). The two smallest datasets, yacht and energy (184 and 460 training points), have high het ratios but show no ECE gain. This pattern was identified after the fact, and across all eleven datasets neither the het ratio nor its ratio to the null predicts the size of the gain (Spearman $\rho = 0.07$ and 0.09).

On constructed data the effect is larger. With the true partition, a correctly specified piecewise-constant noise model and the same out-of-fold evidence, regional noise improves ECE from 0.2374 to 0.1367 on `hetero_extreme` and from 0.0527 to 0.0285 on `combined` (`src/hetero_baseline.py`); these are upper bounds rather than estimates of what a learned partition achieves. It also improves `varying_smoothness` (0.0675 $\rightarrow$ 0.0152), whose noise is homoscedastic: there the regional term absorbs model bias that varies with function complexity, a reminder that input-dependent residual variance is not the same thing as input-dependent noise.

The earlier, opposite conclusion was an estimator bias. A previous version of this analysis found that on ECE global noise was best on nine of eleven datasets and the het-GP on none. Both input-dependent arms were mis-estimated. The regional arm used the out-of-fold squared residual as its noise estimate and then added the latent variance again, counting it twice and making regional noise under-confident, most of all where latent variance is large, at small n. The het-GP

smoothed *log* squared residuals and exponentiated, a geometric mean that for Gaussian residuals sits at $e^{-1.27} \approx 0.28$ of the noise variance. Section 7 records this as the fifth result in this project that turned on how a baseline was configured.

6.1 Adversarial audit of this section's claim

We attacked the regional-noise claim six ways: sharpness, granularity and metric choice on six real datasets with five seeds (`src/adversarial.py`), global recalibration and selection on all eleven datasets with ten seeds (`src/noise_controls.py`), and training-set size on the four datasets large enough for it (`src/adversarial_n.py`). None of the attacks overturns it.

Sharpness. A model with systematically wide intervals can win ECE without being better. Relative to the target's standard deviation, regional intervals are *narrower* than global ones on five of six datasets, and on four of those regional noise is simultaneously sharper and better calibrated; the exception to narrowness is yacht.

Granularity. Regional noise with $2, 3, 5, 10, 20$ regions gives mean ECE between 0.029 and 0.036, against 0.048 for global noise, so the result does not depend on the number of regions.

Global recalibration. The input-dependent arms see validation data and the global arm does not, so we gave the global arm three recalibrations using the same validation data: a single noise level set to the mean out-of-fold evidence (regional noise with one region), a noise multiplier chosen from 61 values in $[0.5, 2]$ on validation NLL, and a rescaling of the total predictive standard deviation by its validation-optimal factor. Only the first helps significantly, and only on airfoil ($0.078 \to 0.060$; calhousing $0.069 \to 0.065$, $p = 0.19$), while it makes concrete, sarcos and synthetic_heteroscedastic significantly worse; neither of the others improves any dataset significantly. Regional noise remains better than each recalibration on the same five datasets at uncorrected $p < 0.05$, and on airfoil and calhousing after Holm correction. The global arm's miscalibration is in the shape of its z-distribution, not its scale.

Selection. Regional noise with one fixed configuration (3 regions, $k = 50$) and no selection is still significantly better than global noise on airfoil, sarcos and calhousing; dropping the noise floor changes no conclusion.

Metric choice. Regional noise is better than global on ECE at 5, 10 and 20 bins (five, four and four of six datasets), on PIT-KS (five of six), on CRPS (five of six) and on NLL (four of six). The one metric on which it wins fewer datasets is coverage error of the 90% interval (two of six), a single point of the calibration curve.

Training-set size. A per-region variance estimated from few residuals is itself noisy, so the comparison could depend on n. We sweep n_{train} from 300 to 4000 with validation and test sets held fixed per seed and nested training sets (Table 8). The mean gap favours regional noise at every n. It is large at every n on calhousing and at $n = 600$ on airfoil, and within about 1.1 standard errors of zero at every n on powerplant and wine_white, the two datasets with the lowest het ratios.

Table 8: ECE gap (regional − global; negative favours regional) ± standard error over five seeds, with validation and test sets (400 points each) fixed across n. Cells are empty where the dataset is too small; the mean is over the datasets present.

n_{train}	calhousing	powerplant	wine_white	airfoil	mean
300	-0.0250 ± 0.0074	$+0.0036 \pm 0.0038$	$+0.0031 \pm 0.0027$	-0.0118 ± 0.0087	-0.0076
600	-0.0357 ± 0.0082	$+0.0044 \pm 0.0041$	$+0.0013 \pm 0.0029$	-0.0332 ± 0.0078	-0.0158
1200	-0.0450 ± 0.0054	$+0.0011 \pm 0.0037$	$+0.0004 \pm 0.0034$		-0.0145
2400	-0.0449 ± 0.0060	-0.0026 ± 0.0057	-0.0026 ± 0.0035		-0.0167
4000	-0.0464 ± 0.0082	-0.0007 ± 0.0051	-0.0024 ± 0.0037		-0.0165

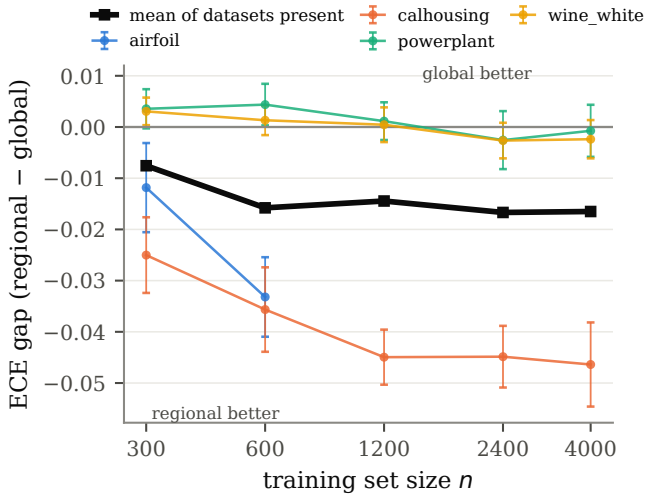


Figure 3: ECE gap between regional and global noise against training set size, with validation and test sets fixed. Negative values favour regional noise. There is no reversal: the mean gap favours regional noise at every n.

Routing approximation fidelity at larger n. The sample-size question applies equally to Sections 4 and 5.3, so we checked it there too (`src/routing_largen.py`). Sweeping n_{train} from 600 to 4800, the oracle-minus-uniform ECE gap remains at zero within noise (+0.0004, +0.0009, −0.0003, −0.0004), and the mean relative spread in predictive standard deviation across the four fidelities is below 4% above $n = 600$ (at most 4.6% for any single dataset; $n = 4800$ covers calhousing and powerplant only). At $n = 600$ it is 32%, driven by wine_white at 93%, where the sparse models are fitted on very little data. Routing headroom does not appear within this range.

6.2 Shrinking regional noise toward global

The natural regulariser for a per-region variance is to shrink it toward the global one, $\sigma_r^2(\lambda) = \lambda\,\sigma^2_{\text{global}} + (1-\lambda)\,\hat{\sigma}_r^2$, with λ selected on validation, or an empirical-Bayes $\lambda_r = (s_r^2/n_r)/(s_r^2/n_r + \tau^2)$ with τ^2 estimated by the method of moments, which needs no search over λ.

All three input-dependent arms improve CRPS over global noise by 0.4–1.3% at every n (Table 9). Shrinkage adds little on top of raw regional noise, about 0.2% (between 0.02% and 0.33%); the paired gain is 2.5–3.0 standard errors at four of the five n (0.5 at $n = 600$), though nested training sets make the five n dependent. The selected λ shows no trend in n (0.27–0.36), so the variance-reduction story that motivated shrinkage, strong shrinkage when data is scarce and weak

Table 9: CRPS relative to global noise (paired per dataset and seed, ± standard error; below 1 favours the arm), with fixed validation and test sets, five seeds, over the real datasets large enough for each n. $\bar{\lambda}$ is the mean selected shrinkage.

n_{train}	datasets	regional	shrunk	EB	$\bar{\lambda}$
300	4	0.9928±0.0024	**0.9913±0.0023**	0.9915±0.0024	0.29
600	4	0.9888±0.0027	0.9886±0.0027	**0.9880±0.0028**	0.27
1200	3	0.9960±0.0018	**0.9927±0.0022**	0.9939±0.0018	0.36
2400	3	0.9919±0.0019	**0.9903±0.0019**	0.9911±0.0020	0.27
4000	3	0.9891±0.0030	**0.9872±0.0028**	0.9888±0.0030	0.29

when it is plentiful, is not supported. A roughly constant λ is what one would expect if shrinkage instead corrects the spread that selection induces in region means built from the same evidence (Section 6), which depends on k rather than n; we have not tested this.

Region identification is not the bottleneck. Scoring regions by predictive uncertainty and local sparsity, as the method we audited does, recovers the true partition only 41–49% of the time, against about 33% for uniformly random labels; on `varying_density` and `combined`, whose true regions are very unequal in size, agreement mainly reflects label proportions. Scoring by kNN-smoothed out-of-fold residual variance raises this to 96.7% on `hetero_extreme`, whose regions differ only in noise. On `varying_smoothness` it reaches 82.0% although the noise is homoscedastic, because residual variance there tracks model bias. On `combined` it manages 44.4% and on `varying_density` 36.1%, scores that, given the unequal regions, say little. Residual-variance scoring recovers the partition when the regions differ in residual variance, whatever its source, and not otherwise.

6.3 Seed robustness

All results in Table 7 use seeds 0–9. On the disjoint seeds 10–19 (`src/diff_seeds.py`) regional noise is again nominally significantly better calibrated than global noise on five datasets (four after Holm correction) and significantly worse on none, and the het-GP is significantly better on NLL on nine.

Seeds are disjoint splits, not new data. On the four datasets used in full (yacht, energy, concrete, airfoil) any two splits share about 60% of training and 20% of test points, so seed-level tests are anti-conservative there. Under a Nadeau–Bengio corrected resampled t-test [Nadeau and Bengio, 2003], the ECE gains of regional noise on airfoil, sarcos and calhousing remain significant in seeds 0–9 ($p \leq 0.015$); those on concrete and synthetic_heteroscedastic do not ($p = 0.18$ and 0.25; synthetic_heteroscedastic reaches $p = 0.038$ in seeds 10–19). We regard the first three as the robust result. In one wine_white seed in ten the global arm's fitted noise sits at its lower bound, which inflates that seed's global ECE; wine_white shows no significant difference either way.

7 Process: which claims failed, and what caught them

This project began as an attempt to *demonstrate* region-aware approximation. Its conclusions changed several times as claims, most of them our own, failed under test. We report that sequence because the failure modes recur, and because a reader deciding how much to trust any section should know which parts were checked adversarially and which were merely not yet falsified.

7.1 Proposed mechanisms fail; direct measurements survive

Table 10: Every explanatory mechanism we proposed, its falsifiable prediction, and its fate. One of six survived, and it rests on a direct measurement.

proposed mechanism	prediction	outcome
Approximation gap inflates predictive variance, offsetting overconfidence	σ should shrink as rank grows	**Refuted.** On robot_arm z-dispersion *falls* monotonically with rank (3.34 $\rightarrow$ 1.01); low-rank models are more overconfident, not less
Exact-GP miscalibration is an undertrained optimiser	more Adam iterations should fix it	**Refuted.** Under exact predictive inference, 50 $\rightarrow$ 1000 iterations move robot_arm from 0.087 to 0.077, far from the 0.017 of an L-BFGS fit
It is hyperparameters fitted on a subsample	fitting on all n should fix it	**Moot.** Most of the gap it was proposed to explain (0.27 vs 0.017) was the inference hazard of Section 3; under exact inference the baseline reaches 0.08
Regional noise loses at small n because per-region variances are estimated from few residuals	the gap should close as n grows	**Refuted on average.** With a corrected noise estimate the mean gap favours regional noise at every n (Table 8); a small, non-significant small-n disadvantage remains on powerplant and wine_white
Shrinkage helps by reducing estimation variance at small n	selected λ should fall as n grows	**Refuted.** $\bar{\lambda}$ stays in 0.27–0.36 with no trend
Fidelity and predictive uncertainty are nearly orthogonal because the noise term dominates Eq. (1)	models of differing fidelity should emit near-identical σ	**Survived.** Across Nyström ranks RMSE moves 21–155% and σ at most 12%; per-region σ within 2.7% on constructed data

7.2 Five results that turned on baseline configuration

In each case a headline number was determined not by the method under study but by how its comparator was set up. Three were in the work we audited and two were our own; both of ours would have reversed a conclusion.

1. **Lengthscale in the wrong space.** The median heuristic was computed on raw X and applied to standardised X, wrong by 155,000$\times$ on protein, where the baseline degenerated to a constant predictor (RMSE = 6.1346 against std(y) = 6.1182, $R^2 = 0$). The original

+41.1% headline was measured against these broken baselines; with scale-correct lengthscales, fitted noise and exact inference, the method without an exact-GP tier is about −9% against uniform Nyström (−9.9% and −9.2% in two runs), and with one between +4.5% and +8.1% (`results/fair_exact`; its GPyTorch fits vary between runs).

2. **Noise never fitted.** The approximations used a hardcoded $\sigma_n = 0.1$ while the exact GP fitted its noise by marginal likelihood. One arm was tuned and the others were not.

3. **Approximate inference in the "exact" baseline** (Section 3), which perturbed every calibration number above $n = 800$ and inflated most, by up to a factor of ten.

4. **In-sample residuals.** Our first heteroscedastic-GP comparison estimated noise from in-sample residuals of a GP conditioned on the same points, which underestimate it, and the het-GP was worse than global noise on three of four constructed datasets.

5. **A biased noise estimator in both input-dependent arms.** Moving to out-of-fold residuals was not enough. An out-of-fold squared residual is noise plus latent variance; the regional arm then added the latent variance again, and the het-GP took a geometric mean of squared residuals, 0.28× too small. Together these produced a negative result on ECE (global noise best on nine of eleven datasets, the het-GP on none) and a spurious sample-size reversal that an earlier draft of this paper reported as a finding. With the estimator corrected, the result is the opposite (Section 6).

A comparison is only as meaningful as the effort spent on the arm one is not advocating. Reporting, for every arm, what was fitted, what was selected, on which split, and what the estimator's expectation is would have caught all five. The fifth was found only by an independent audit of the code against the paper.

7.3 Verifications that passed without running

Four checks reported success while failing:

- An audit script printed `CLEAN: no selection on test labels` three times. Under `zsh` an unquoted array variable reaches `grep` as a single argument; the greps errored and a `|| echo` fallback reported the pass.
- A crashing script reported `EXIT=0`, because a pipeline's exit status is its last stage and the traceback had been piped to `tail`.
- Table 3, the paper's central mechanism, was generated by an ad-hoc command that was never committed. Regenerating it from a script changed the numbers.
- The decomposition of Section 3 first reported that raising the Cholesky threshold alone left the error unchanged. Every inference path was evaluated on the same model object, and GPyTorch reused the covariance cached by the first path; a statistical review of the code found it.

Every number in Sections 3 to 6 is produced by, or recomputable from, results written by committed scripts reachable from `reproduce.sh`. Tables 1 to 3, 5 and 7 to 9 are written directly from stored results by `src/paper_tables.py`; Tables 4, 6 and 10 transcribe `results/` files. The audit figures in this section come from `AUDIT.md` and `RESULTS_FAIR.md`, and the earlier, superseded results from the repository history.

7.4 A checklist for GP calibration studies

1. Confirm the inference path is exact. Raise any Cholesky-size threshold above n and disable fast predictive-variance approximations, for training as well as prediction; clear the prediction cache between configurations; verify that results below the threshold are unchanged by the setting.

2. Fit kernel hyperparameters in the space the kernel operates in, and verify a fitted lengthscale against the median pairwise distance of the *transformed* inputs.

3. Give every arm the same tuning budget, and state what was fitted and selected for each.

4. Estimate any residual-derived quantity out-of-fold, and check its expectation: an out-of-fold squared residual includes latent variance, and the log of a squared residual is biased by -1.27.

5. Screen datasets for the property under study over several seeds, and calibrate the screening statistic against a permutation null; do not assume the property. (Table 5: the het ratio's null is 1.8–4.1, not 1.)

6. Report a proper scoring rule alongside any binned calibration metric, and the ECE noise floor for the test-set size.

7. Average metrics across datasets only in scale-free form; CRPS, NLL differences aside, is in target units, and one dataset can dominate a raw mean.

8. When sweeping n, hold the validation and test sets fixed.

9. Correct per-dataset significance for the number of datasets, and treat resampled splits of one dataset as dependent.

8 Threats to validity

- Our constructed datasets reach het ratios of $\sim$1600 where measured data averages at most $\sim$9 (Table 5). Two of them vary noise in steps at equal-width bands of x_1 with three regions, which structurally favours a piecewise-constant noise model and makes the partition easy to recover; they are dense and two-dimensional, which makes the latent term small and favours orthogonality. The claims of Section 6 rest on the real-data results.

- Every GP in Sections 4 to 6 uses an isotropic RBF kernel (one lengthscale, even for 21-dimensional sarcos) fitted by L-BFGS-B with at most 60 iterations from each of three starts. Misspecification of the mean adds input-dependent residual variance, which regional noise can absorb; an ARD kernel might narrow the gap on the datasets where regional noise wins.

- The heteroscedastic GP is a two-stage kNN construction, not a full implementation of Kersting et al. [2007] or of the variational heteroscedastic GP of Lázaro-Gredilla and Titsias [2011]. An independent GPyTorch implementation with a GP noise model (`src/hetero_independent.py`, five seeds) agrees where the effect is large, improving on its own global baseline on airfoil (0.077 $\rightarrow$ 0.048 ECE) and calhousing (0.068 $\rightarrow$ 0.029), but is worse on ECE on the other four, most of all on the two smallest (energy and yacht). How much an input-dependent noise model gains therefore depends on how it is built; that it gains on the larger heteroscedastic datasets does not.

- Residual variance that varies with x may reflect model bias rather than noise (Section 6, `varying_smoothness`); we therefore use the term *input-dependent residual variance*.
- Some datasets carry known issues (Section 5, `data/README.md`): duplicated SARCOS test rows, duplicate inputs in wine_white, censored calhousing targets, and two simulated datasets. Small test sets limit how finely ECE can separate arms.
- Training-set sizes run from $n = 184$ to $n = 4800$. Behaviour at the 10^5–10^6 scale where sparse approximations are most needed is untested.

9 Related work

Sparse GP approximation. Inducing-point and spectral methods [Quiñonero-Candela and Rasmussen, 2005, Titsias, 2009, Rahimi and Recht, 2007, Williams and Seeger, 2001, Smola and Schölkopf, 2000] reduce the $O(n^3)$ cost by replacing the kernel matrix with a low-rank surrogate. This literature is overwhelmingly concerned with the *latent* term of Eq. (1); our observation is that predictive *uncertainty* is usually dominated by the other term, which is why methods that differ substantially in approximation quality can be nearly indistinguishable in predictive standard deviation.

Locally adaptive and mixture-of-experts GPs. The closest prior art partitions the input space and fits separate models per region [Rasmussen and Ghahramani, 2002, Tresp, 2000, Deisenroth and Ng, 2015, Gramacy and Lee, 2008]. Our results are not in tension with it, and the distinction matters. Those methods give each region its own *kernel and noise hyperparameters*, so per-region predictive variance genuinely differs; the mechanism we identify, and our noise-model results, are precisely why that is the right design. What we rule out is the weaker variant (one global model, with only *approximation fidelity* varying by region), which leaves the dominant term of Eq. (1) untouched. Treed GPs [Gramacy and Lee, 2008] additionally learn the partition, which our result on residual-variance scoring complements.

Heteroscedastic GPs. Input-dependent noise has been modelled by a second GP on log residuals [Goldberg et al., 1997], by EM [Kersting et al., 2007], and variationally [Lázaro-Gredilla and Titsias, 2011]. Our regional noise is a coarse, piecewise approximation to these, and Section 6 finds that even this coarse form significantly improves calibration on three of our ten real datasets after correction, four (five of eleven including the constructed one) when twenty splits are pooled. Our experience adds a practical caution for two-stage constructions: both the latent variance in out-of-fold residuals and the log-of-squared-residual bias are large enough to reverse a comparison.

Calibration in regression. Interval-coverage calibration [Kuleshov et al., 2018] and proper scoring rules [Gneiting and Raftery, 2007] measure different things. In our final results they mostly agree (on yacht regional noise is no worse on ECE but significantly worse on NLL and CRPS in seeds 10–19 and pooled), whereas an earlier, biased version of the comparison had them disagree systematically, which is one reason we report both.

Reproducibility and measurement validity. Section 3 belongs to a line of work showing that reported gains can originate in evaluation setup rather than method [Henderson et al., 2018, Musgrave et al., 2020, Lipton and Steinhardt, 2019]. Our instance is unusual in being a silent,

threshold-triggered approximation inside a trusted library rather than a protocol choice, which makes it invisible in code review.

10 Conclusion

Region-aware Gaussian processes can adapt two things by region: the fidelity of the approximation, or the noise model. In our experiments only the second changed the predictive uncertainty a region receives. The random-feature and Nyström approximations we tested alter only the latent term of the predictive variance, so their predictive standard deviations move far less than their accuracy and agree within 3% in every region of our constructed data, and an oracle router given the true partition gains nothing on average (+0.0003 ECE). Making the noise term regional, by contrast, improves calibration robustly on the larger datasets whose residual variance is clearly input-dependent (airfoil, sarcos and calhousing under every correction we apply, five datasets in pooled analyses) and does not worsen ECE on any, though it worsens NLL and CRPS on the smallest.

The practical readings are that routing approximation fidelity among homoscedastic approximations is not worth building, that the same regional machinery pointed at the *noise* model is worth trying when there is enough data and the residual variance is clearly input-dependent relative to a permutation null (a screen that separated, after the fact, the datasets that gained from those that did not, but did not predict the size of the gain), and that the partition such methods need is recoverable by residual-variance scoring when the regions differ in residual variance.

Separately, practitioners measuring GP calibration should verify that their inference path is exact. Above a library default that appears nowhere in user code, ours was not, and the calibration numbers we computed above it before noticing were inflated by up to a factor of ten.

Reproducibility. Every number above is regenerated by `./reproduce.sh` from the code, data and results at `https://github.com/Srikarmk/region-aware-gp`.

References

Thomas Brooks and D. Pope. Airfoil self-noise. UCI Machine Learning Repository, 1989.

Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4): 547–553, 2009. doi: 10.1016/j.dss.2009.05.016.

Marc Peter Deisenroth and Jun Wei Ng. Distributed Gaussian processes. In *ICML*, 2015.

Jacob R. Gardner, Geoff Pleiss, David Bindel, Kilian Q. Weinberger, and Andrew Gordon Wilson. GPytorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration. In *NeurIPS*, 2018.

J. Gerritsma and R. Onnink. Yacht hydrodynamics. UCI Machine Learning Repository, 1981.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *JASA*, 102(477):359–378, 2007.

Paul W. Goldberg, Christopher K. I. Williams, and Christopher M. Bishop. Regression with input-dependent noise: A Gaussian process treatment. In *NeurIPS*, 1997.

Robert B. Gramacy and Herbert K. H. Lee. Bayesian treed Gaussian process models with an application to computer modeling. *JASA*, 103(483):1119–1130, 2008.

Peter Henderson, Riashat Islam, Philip Bachman, Joelle Pineau, Doina Precup, and David Meger. Deep reinforcement learning that matters. In *AAAI*, 2018.

Kristian Kersting, Christian Plagemann, Patrick Pfaff, and Wolfram Burgard. Most likely heteroscedastic Gaussian process regression. In *ICML*, 2007.

Volodymyr Kuleshov, Nathan Fenner, and Stefano Ermon. Accurate uncertainties for deep learning using calibrated regression. In *ICML*, 2018.

Miguel Lázaro-Gredilla and Michalis K. Titsias. Variational heteroscedastic Gaussian process regression. In *ICML*, 2011.

Zachary C. Lipton and Jacob Steinhardt. Troubling trends in machine learning scholarship. *ACM Queue*, 17(1), 2019.

Kevin Musgrave, Serge Belongie, and Ser-Nam Lim. A metric learning reality check. In *ECCV*, 2020.

Claude Nadeau and Yoshua Bengio. Inference for the generalization error. *Machine Learning*, 52 (3):239–281, 2003. doi: 10.1023/A:1024068626366.

R. Kelley Pace and Ronald Barry. Sparse spatial autoregressions. *Statistics & Probability Letters*, 33(3):291–297, 1997. doi: 10.1016/S0167-7152(96)00140-X.

Geoff Pleiss, Jacob R. Gardner, Kilian Q. Weinberger, and Andrew Gordon Wilson. Constant-time predictive distributions for Gaussian processes. In *ICML*, 2018.

Joaquin Quiñonero-Candela and Carl Edward Rasmussen. A unifying view of sparse approximate Gaussian process regression. *JMLR*, 6:1939–1959, 2005.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *NeurIPS*, 2007.

Prashant Rana. Physicochemical properties of protein tertiary structure. UCI Machine Learning Repository, 2013.

Carl Edward Rasmussen and Zoubin Ghahramani. Infinite mixtures of Gaussian process experts. In *NeurIPS*, 2002.

Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

Carl Edward Rasmussen, Radford M. Neal, Geoffrey E. Hinton, Drew van Camp, Michael Revow, Zoubin Ghahramani, Rafal Kustra, and Robert Tibshirani. The DELVE manual. Technical report, University of Toronto, 1996.

Alex J. Smola and Bernhard Schölkopf. Sparse greedy matrix approximation for machine learning. In *ICML*, 2000.

Edward Snelson and Zoubin Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *NeurIPS*, 2006.

Michalis Titsias. Variational learning of inducing variables in sparse Gaussian processes. In *AISTATS*, 2009.

Volker Tresp. A Bayesian committee machine. *Neural Computation*, 12(11):2719–2741, 2000.

Athanasios Tsanas and Angeliki Xifara. Accurate quantitative estimation of energy performance of residential buildings using statistical machine learning tools. *Energy and Buildings*, 49:560–567, 2012. doi: 10.1016/j.enbuild.2012.03.003.

Pınar Tüfekci. Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. *International Journal of Electrical Power & Energy Systems*, 60:126–140, 2014. doi: 10.1016/j.ijepes.2014.02.027.

Sethu Vijayakumar and Stefan Schaal. Locally weighted projection regression: An O(n) algorithm for incremental real time learning in high dimensional space. In *ICML*, pages 1079–1086, 2000.

Christopher K. I. Williams and Matthias Seeger. Using the Nyström method to speed up kernel machines. In *NeurIPS*, 2001.

I-Cheng Yeh. Modeling of strength of high-performance concrete using artificial neural networks. *Cement and Concrete Research*, 28(12):1797–1808, 1998. doi: 10.1016/S0008-8846(98)00165-3.