PRIORFUSE: HOW PRIOR-DATA FITTED NETWORKS DEPART FROM BAYES

Anonymous authors
Paper under double-blind review

ABSTRACT

Prior-Data Fitted Networks (PFNs) let researchers specify prediction rules through synthetic task priors and are trained to approximate the corresponding Bayesian posterior predictive. However, specifying a Bayesian target does not guarantee that a finite trained network will use new evidence as that target would. To study this question, we introduce PriorFuse, an exactly enumerable benchmark in which the target posterior predictive can be computed exactly by Bayesian model averaging (BMA). For five PFNs trained only with eight context examples, supplying twelve examples improves predictive log loss while increasing excess log loss over exact BMA approximately ninefold at 100k training steps. We show that the trained PFNs exhibit a systematic Bayesian departure: fitted likelihood weighting changes from amplification at several shorter context lengths to attenuation at longer lengths. At the tested longer lengths, a post hoc inverse-length adjustment to the exponent fitted at eight examples recovers about 99% of the improvement in matching PFN predictions achieved by refitting at each length, and unchanged transfer recovers much less. The same matched-length attenuation pattern also appears under a second task prior and weakens with further training. Equivalent representations of the same prior also change fitted family parameters without changing either predictor. PriorFuse thus helps researchers distinguish better prediction from more faithful inference and identify which parts of a learned prediction rule change with training and context.

1 INTRODUCTION

Prior-Data Fitted Networks (PFNs) offer a way to specify prediction algorithms through synthetic task priors and are trained to approximate the corresponding Bayesian posterior predictive (Müller et al., 2022; 2025), which can be computed exactly by Bayesian model averaging (BMA) when the prior ranges over a finite set of hypotheses. However, prior-fitting has two distinct steps: choosing a prior defines the Bayesian predictor one wants, while a finite trained network must learn how that predictor should respond to evidence. When PFN performance improves, has the learned predictor's response to new observations actually moved closer to the intended Bayesian rule?

To study this question, we introduce PriorFuse, an exactly enumerable benchmark for identifying and characterizing Bayesian departure in trained PFNs. We use *Bayesian departure* to refer to the difference between a PFN's predictive distribution and exact BMA for the same context and query. The term is metric-independent, and we state the metric explicitly whenever reporting numerical values. We show that the trained PFNs exhibit a Bayesian departure, and that this departure follows a systematic pattern.

Our experiments show why that question cannot be answered from predictive loss alone. Five PFNs are trained only with eight context examples. When the same trained networks receive twelve examples, expected predictive log loss improves, yet excess log loss over exact BMA increases approximately ninefold at 100k training steps. Within each task, the same four unseen query inputs are scored across lengths. More observations help, but exact BMA gains more from them. The growing Bayesian departure is also structured: a likelihood-weighting description captures much of it, and a post hoc length adjustment transfers far better than carrying its exponent over unchanged.

This distinction is useful when deciding what a PFN improvement actually changed. A lower predictive loss could reflect a more faithful approximation to the intended Bayesian predictor, a

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

different weighting of context evidence, a transformation of final probabilities, or another systematic approximation. These possibilities motivate different follow-up experiments. Changing the synthetic prior addresses a different problem from improving approximation to the same prior, and output calibration is different from changing how evidence weights hypotheses. PriorFuse provides a controlled way to separate these possibilities instead of inferring the learned rule from aggregate performance.

The benchmark uses finite families of Boolean functions so that the Bayesian prediction can be computed exactly for every context and query. Its main prior contains 1,088 distinct functions across additive and shallow-tree families. The families are distinct but overlapping: a depth-two tree can represent exclusive OR of two features, which a linear threshold cannot; a majority rule over three features has the opposite representability pattern. Prediction remains uncertain within and between families after observing noisy labels. Two families suffice to compare selection with averaging while keeping the posterior family weight scalar. A *mixed PFN* is trained on a mixture of task families; the term describes its prior, not its architecture.

We fit structured candidate descriptions on calibration tasks and compare them on separate evaluation tasks. The experiments first hold the trained networks fixed and change the amount of inference-time evidence. They then follow the matched-length approximation through training, repeat that comparison under a second exactly enumerable prior, and test whether fitted family parameters survive an equivalent representation of the same Bayesian prior. The core models are designed to be small enough to support repeated independent training and exact diagnostics; a separate larger-architecture control tests one capacity change. We therefore treat the observed Bayesian departures as properties of these trained models, not as universal PFN laws or identified internal mechanisms.

2 Related Work

2.1 Bayesian In-Context Learning and PFNs

PFNs learn an amortized predictor from synthetic tasks and target the posterior predictive induced by their training prior (Müller et al., 2022). TabPFN made this paradigm practical for tabular classification (Hollmann et al., 2023; 2025). A finite trained network need not reproduce its intended Bayesian predictor exactly, motivating statistical analyses of the gap between learned and Bayesian prediction (Nagler, 2023; Wakayama & Suzuki, 2025). Bayesian accounts of in-context learning include latent-concept inference (Xie et al., 2022), mixture prediction (Pathak et al., 2024), and BMA over latent structure (Zhang et al., 2025b; Wei et al., 2026). Transformers can approximate tractable Bayesian predictors over heterogeneous task families (Panwar et al., 2024), while other work finds recognizable estimators, algorithm selection, and violations of necessary Bayesian coherence (Akyürek et al., 2023; Bai et al., 2023; Falck et al., 2024). Reuter et al. (2025) explicitly train transformers to produce full posterior distributions in context, whereas PriorFuse studies approximation of a known posterior predictive by a standard PFN. Exact likelihood-ratio settings have also revealed sufficient-statistic recovery up to monotone transformation (Chaudhry & Gadkari, 2026). Boolean function-learning benchmarks also reveal differences in task difficulty and learned algorithms (Bhattamishra et al., 2024); our finite mixtures are chosen to make the full Bayesian reference and its candidate modifications computable for every observed task.

Our comparisons draw on BMA (Hoeting et al., 1999), proper scoring rules (Gneiting & Raftery, 2007), and generalized Bayesian updating (Bissiri et al., 2016).

2.2 Training Dynamics, Tempering, and Synthetic Priors

Training time, task diversity, and model resources can change the behavior learned in context (Raventós et al., 2023; Wu et al., 2024; Bordelon et al., 2026; Qchohi & Rossi, 2026; Demir & Dogan, 2025). Park et al. (2025) compare competing algorithms as training and context size vary, while Wurgaft et al. (2025) model training-dependent strategy mixtures. Bayesian approximation across task families has already been studied by Panwar et al. (2024). PriorFuse focuses on Bayesian departure under a specified prior, rather than claiming that training-dependent behavior or Bayesian comparisons are new.

Context-length effects also have direct precedents. Liu et al. (2025) compare uncertainty predictions with Bayesian references under prompt-length shifts. Ujváry et al. (2025) study tempered Bayesian descriptions and prompt-length behavior in coin-flip tasks, including model misspecification. Kang et al. (2026) construct transformers for Gaussian-process posterior prediction and study how depth and normalization affect sample-size generalization. Here the function prior and label model remain fixed and correctly specified while the context length differs from PFN training. We compare several descriptions of the resulting errors, then test whether a likelihood exponent calibrated at the training length transfers better unchanged or with an inverse-length adjustment. The latter comparison is post hoc and does not identify a normalization mechanism.

Synthetic-prior design is central to tabular foundation models (Zhang et al., 2025a; Türkmen et al., 2026); Müller et al. (2025) frame it as a way to specify prediction algorithms. MixturePFN instead trains and routes among specialized in-context prompters for different data clusters (Xu et al., 2025). Our contribution is evidence about what a trained predictor does with additional observations, where prediction can improve while Bayesian fidelity worsens, and much of the Bayesian departure admits a compact description that transfers without target-length parameter fitting. Temperature, logistic, and isotonic transformations are established tools (Guo et al., 2017; Zadrozny & Elkan, 2002), and our contribution is what their comparison reveals, not the introduction of those tools.

3 An Exact Benchmark for Structured Bayesian Approximation

3.1 Bayesian Target and Equivalent Prior Representations

Let $\mathcal{X} = \{0,1\}^4$ be the input feature space. Fix an ordering $(u_1, \ldots, u_{16})$ of its elements. A hypothesis is a Boolean function $h : \mathcal{X} \to \{0,1\}$, represented by its truth table $(h(u_1), \ldots, h(u_{16}))$. The truth table lies in $\{0,1\}^{16}$ and fully specifies h. Parameterizations with the same truth table are counted as one hypothesis.

The additive family $\mathcal{H}_A$ enumerates thresholded linear functions with integer coefficients $w_j \in \{-2,-1,0,1,2\}$ and bias $b \in \{-4,\ldots,4\}$. Counting each distinct truth table once gives $|\mathcal{H}_A| = 980$. The tree family $\mathcal{H}_T$ enumerates binary decision trees of depth at most two, with input-feature splits and binary leaf labels, giving $|\mathcal{H}_T| = 166$. The families share 58 functions, leaving 1,088 distinct hypotheses in their union. Appendix A gives the complete formula and enumeration procedure.

A task first samples a family $z \in \{A, T\}$ with $\Pr(z = T) = \pi$ and then samples a hypothesis uniformly from $\mathcal{H}_z$. The central benchmark uses the balanced prior $\pi = 1/2$.

A labeled context is $D = \{(x_i, y_i)\}_{i=1}^n = (X_D, y_D)$, where $x_i \in \mathcal{X}$, $y_i \in \{0,1\}$, and $n = |D|$ counts labeled examples. Original training and matched-length evaluation use $n = 8$ inputs sampled uniformly without replacement, independently of h and z. Labels are independently flipped with probability $\eta = 0.10$: $p(y_i \mid x_i, h) = 1 - \eta$ when $y_i = h(x_i)$ and η otherwise. A query $x_* \in \mathcal{X} \setminus \{x_1, \ldots, x_n\}$ has an independently generated label under the same noise model.

For family z, the marginal likelihood of the context labels is

$$E_z(D) := p(y_D \mid X_D, z) = \frac{1}{|\mathcal{H}_z|} \sum_{h \in \mathcal{H}_z} \prod_{i=1}^{n} p(y_i \mid x_i, h). \tag{1}$$

The context-input sampling probability is independent of z and therefore cancels from the posterior family odds. The posterior tree-family probability is

$$r_\pi(D) = \frac{\pi E_T(D)}{(1-\pi)E_A(D) + \pi E_T(D)}.$$

Write $r(D) = r_{1/2}(D)$. Let $p_A(y_* \mid D, x_*)$ and $p_T(y_* \mid D, x_*)$ denote the exact posterior predictives within the two families; p_A and p_T abbreviate their probabilities of $y_* = 1$ when the arguments are clear. Exact Bayesian model averaging (BMA) is

$$p_{\mathrm{BMA}}(y_* \mid D, x_*) = [1 - r_\pi(D)]\, p_A(y_* \mid D, x_*) + r_\pi(D)\, p_T(y_* \mid D, x_*). \tag{2}$$

Because the family label is latent, the same Bayesian predictor can be written using the induced marginal prior over the distinct hypotheses,

$$\mu(h) = \frac{1-\pi}{|\mathcal{H}_A|}\mathbf{1}[h \in \mathcal{H}_A] + \frac{\pi}{|\mathcal{H}_T|}\mathbf{1}[h \in \mathcal{H}_T]. \tag{3}$$

A function shared by the two families receives both prior contributions. Bayesian prediction under μ is identical to Eq. 2. We use this equivalence later to test whether conclusions based on family weighting depend on the chosen factorization, without treating that factorization as an identified internal PFN representation.

3.2 Structured Candidate Predictions

We compare several structured candidate predictions to ask what kind of modification to exact BMA best reproduces the PFN's learned prediction rule. We call a candidate more faithful when it more closely reproduces the PFN's predictive probabilities on held-out queries, as measured primarily by lower PFN-matching KL. The candidates alter different parts of the Bayesian computation, including family weighting, hypothesis likelihoods, and final output probabilities. Comparing them against the same PFN predictions lets us identify where the observed Bayesian departure is best described, without treating any candidate as an internal mechanism. The family-tempering candidate rescales the log posterior odds between the two task families while leaving their family-specific posterior predictives unchanged. We refer to this candidate as family tempering:

$$g_a(D) = \mathrm{sigmoid}(a \,\mathrm{logit}\, r_\pi(D)), \qquad q_a = (1-g_a)p_A + g_a p_T, \tag{4}$$

with exact BMA at $a = 1$. Logistic and isotonic family mappings add an intercept or flexible monotone shape. Fixed family pooling, hard maximum a posteriori (MAP) selection, and output-only calibration test simpler alternatives. Fixed family pooling uses a fixed-weight linear pool of the two family-specific posterior predictives, with the same weight for every context.

In contrast, likelihood tempering applies the same likelihood exponent to every hypothesis in the complete hypothesis space, while leaving the prior unchanged,

$$p_\tau(h \mid D) \propto \mu(h)\, p(y_D \mid X_D, h)^\tau, \tag{5}$$

$$q_\tau(y_* = 1 \mid D, x_*) = \sum_h p_\tau(h \mid D)\,[\eta + (1-2\eta)h(x_*)]\,. \tag{6}$$

Exact BMA is $\tau = 1$; $\tau < 1$ attenuates and $\tau > 1$ amplifies the log likelihood. Under symmetric label noise, powering the context likelihood by τ induces exactly the same posterior over hypotheses as using

$$\eta_{\text{eff}} = \left[1 + \left((1-\eta)/\eta\right)^\tau\right]^{-1} \tag{7}$$

in the context likelihood. The query-label channel in Eq. 6 retains the original noise rate η. Thus η_{eff} is an exact reparameterization of the strength of evidence from the context, not the noise rate of a complete Bayesian model used unchanged for both context and query labels. As a post hoc robustness comparison, posterior-to-prior shrinkage uses $q_\lambda = (1-\lambda)p_{\text{BMA}} + \lambda q_{\text{prior}}$, with $0 \le \lambda \le 1$, to test retention of the unconditional prior. Its posterior form is given in Appendix C. We additionally fit a joint family tempering plus output-calibration model and a flexible output-only isotonic map. These output-only controls transform BMA probabilities to match PFN outputs; they do not recalibrate PFNs against observed labels. All candidates are predictive descriptions, not claims about internal computation. Appendix C gives the candidate definitions and context-likelihood noise derivation.

3.3 Benchmark, Training, and Evaluation

Our primary benchmark uses a balanced mixture of additive threshold functions and shallow decision trees. We refer to this as the original additive/tree study to distinguish it from the second-prior experiment introduced later. For the original additive/tree study, five nanoTabPFNs (Pfefferle et al., 2025) are independently trained under the balanced task-family prior for 20k steps, then the same models are continued to a fixed 100k horizon. Training uses contexts of eight examples. After fixing the candidate comparison and analysis protocol, we generate independent calibration and evaluation sets containing 4k and 10k contexts, respectively. Candidate parameters are fitted only on calibration contexts and evaluated unchanged on five held-out queries per evaluation context.

Let $p_\theta(y_* = 1 \mid D, x_*)$ denote the PFN probability and $q_m(y_* = 1 \mid D, x_*)$ the probability under candidate m. Here m indexes the candidate descriptions defined in Section 3.2. Behavioral fidelity is measured by *PFN-matching KL*, a Kullback-Leibler (KL) divergence,

$$K_m = \mathbb{E}_{D,x_*} D_{\mathrm{KL}} \left(\mathrm{Bern}(p_\theta) \,\|\, \mathrm{Bern}(q_m) \right), \tag{8}$$

where the PFN distribution is the first argument throughout. Lower values mean closer reproduction of PFN probabilities, not better predictive risk. Probability mean absolute error (MAE) and logit root-mean-square error (RMSE) provide distinct sensitivity metrics. Each independently trained PFN is a model-level replicate. Within a PFN, context-level cluster bootstraps with 2k draws independently resample calibration and evaluation contexts, refit candidates, and keep queries together within each sampled context. These intervals are conditional on that trained PFN and consistency across independently trained PFNs provides model-level replication. Appendix B gives the architecture, optimization, task-generation, and query protocols, and Appendix D gives the complete evaluation and bootstrap procedures.

4 More Context Can Improve Prediction but Worsen Bayesian Fidelity

We hold the five PFNs fixed and reveal $n = 2, 3, \ldots, 12$ labeled examples, without retraining. Only $n = 8$ matches their training context length. Within each task, nested contexts share the function, input permutation, and label-noise realization. The same four unseen inputs are scored throughout. Since the domain has sixteen inputs, twelve is the largest context preserving these four common unseen queries. Extending the sweep would require fewer unseen query inputs or repeated observations. We use 4k calibration and 10k independently sampled evaluation task groups, recompute exact BMA at each length, and fit candidate parameters on calibration groups only. Bootstrap resampling preserves task-group pairing across lengths and checkpoints (Appendix N).

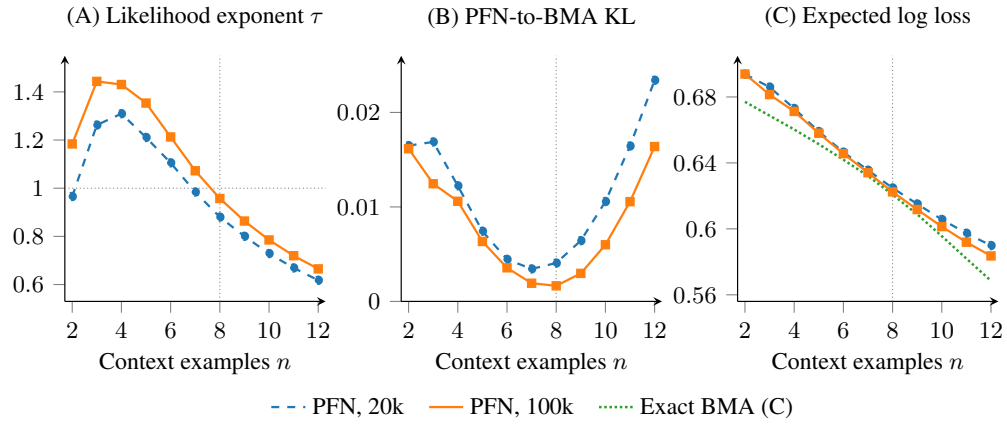


Figure 1: How the same PFNs respond to additional evidence. All were trained with eight context examples, marked by the vertical reference. Within each task, the same four unseen query inputs are scored across lengths. (A) Fitted likelihood exponents. (B,C) Beyond eight examples, PFN-to-BMA KL increases while expected predictive log loss decreases. Curves show cross-seed medians; individual models and conditional intervals appear in Appendix N, including the bound-hitting fit hidden by the two-example median.

For the 100k models, increasing n from eight to twelve lowers median expected predictive log loss from 0.622411 to 0.583601. Exact BMA improves from 0.620759 to 0.568481, so excess log loss rises from 0.001652 to 0.015120, approximately ninefold. PFN-to-BMA matching KL likewise rises from 0.001650 to 0.016385 (Figure 1). The two comparisons use opposite KL directions: excess log loss averages $D_{\mathrm{KL}}(p_{\mathrm{BMA}} \| p_\theta)$, whereas matching KL uses the PFN first. Thus the separation holds within predictive log loss itself: more observations help, but the PFN falls further behind the Bayesian predictor for those observations.

Likelihood tempering describes a substantial part of the longer-context Bayesian departure. At $n = 12$, median fitted τ is 0.618 at 20k and 0.665 at 100k, and its median relative reduction in PFN-matching KL versus BMA is 82.6% and 86.8%, respectively. The same description is not uniform across lengths, as every fitted τ exceeds one at $n = 3, 4, 5, 6$ and is below one at $n = 8, 9, 10, 11, 12$, at both checkpoints. Output isotonic mapping is closer than likelihood tempering in every model at $n = 2, 3, 4$. Because feature standardization is recomputed from the labeled context, varying n changes both the available evidence and the preprocessing applied to the inputs. This confound is strongest at very short contexts, where bound-hitting fits also occur (Appendix N); accordingly, fitted τ is a behavioral description rather than an isolated estimate of evidence weighting.

4.1 Transferring the Longer-Context Description

Next, we test whether the fitted exponent transfers across context lengths without refitting. We compare $\tau_{\text{constant}}(n) = \widehat{\tau}(8)$ with $\tau_{\text{scaled}}(n) = 8\widehat{\tau}(8)/n$ for $n = 9, \ldots, 12$. Both use the same scalar calibrated at eight examples; neither is fitted at the target length. The inverse-length rule was proposed after inspecting the length sweep, making it a post hoc comparison.

Scaled transfer recovers cross-seed medians of 98.8% to 99.2% of the PFN-matching KL improvement over BMA obtained by separately fitting the likelihood exponent at each length (Table 1). Unchanged transfer recovers substantially less, reaching only 20.2% at $n = 12$ for the 100k models. The percentages describe recovery of the fitted candidate's improvement, rather than the fractional reduction in PFN-to-BMA KL. Scaled transfer beats unchanged transfer in all 40 model/checkpoint/length conditions, but separate fitting remains measurably better. The comparison identifies a useful relationship among the longer-context predictions, not a procedure for correcting the PFN or an independently confirmed scaling law (Appendix O).

Table 1: Median across seeds of the percentage of the separately fitted likelihood candidate's improvement over BMA recovered by each transfer rule. Ratios are computed per model, not from median losses. Neither transferred rule is refitted at the target length.

	20k steps		100k steps	
n	Constant	$8/n$ scaled	Constant	$8/n$ scaled
9	82.8	99.2	51.2	99.1
10	66.6	98.9	33.9	99.0
11	55.4	98.8	25.4	98.9
12	47.0	98.9	20.2	99.0

5 Training and Bayesian Approximation Under Two Priors

5.1 Matched-Length Candidate Comparisons

Table 2 compares the candidate descriptions; Appendix E reports seed-wise results. At 20k, hard MAP selection is nearly five times farther from the PFN outputs than exact BMA, so hard MAP selection is clearly not the dominant behavior. At the same checkpoint, a fixed family-mixing weight lowers median PFN-matching KL from 0.004072 to 0.003482. Allowing that cross-family weight to depend on the context improves the fit again. The outputs therefore occupy an intermediate regime between fixed family pooling and exact posterior task family weighting.

The tested output-only logistic and isotonic mappings are less faithful than family tempering in every original-prior model at both matched-length checkpoints. Adding an output transformation jointly changes the fitted family parameter only slightly and leaves it below one (Appendix H). These comparisons favor a description that uses family evidence over the tested transformations of BMA probabilities and are not experiments in recalibrating PFN predictions against labels.

We compare family tempering with likelihood tempering using one fitted parameter for each model. On the independent evaluation set, likelihood tempering is more faithful than family tempering in four of the five observed PFNs at 20k and all five at 100k (Appendix E, Figure 4). At 20k, seed 709 is the sole exception and its interval spans zero (Appendix E, Table 7). The cross-seed median

improvement is 2.75% relative to the KL under family tempering at 20k and 1.94% at 100k. The median seed-wise relative reduction in PFN-matching KL compared with exact BMA is 22.0% at 20k and 6.4% at 100k. Thus, likelihood tempering captures a measurable but incomplete part of the Bayesian departure. The seed-wise ordering is unchanged under the two distinct sensitivity metrics, as likelihood tempering also has lower probability MAE and logit RMSE than family tempering in four of the five observed PFNs at 20k and all five at 100k, with seed 709 the sole 20k exception under all three criteria.

A post hoc posterior-to-prior shrinkage comparison improves on BMA in all five models at both checkpoints but remains less faithful than either tempering candidate (Appendix E, Table 6).

The median calibration-fitted exponent rises from $\tau = 0.883$ at 20k to 0.959 at 100k. With true label noise $\eta = 0.10$, these values induce exactly the same posterior over hypotheses as context likelihoods with $\eta_{\text{eff}} = 0.126$ and 0.109, respectively, while the query-label channel retains $\eta = 0.10$. The fitted attenuation of context evidence therefore recedes with training.

Likelihood tempering is not a complete description, as flexible family isotonic mapping is more faithful for one seed at each checkpoint. On queries inside the family-prediction convex hull at both checkpoints, family tempering and family logistic mapping are closer than likelihood tempering in all five models. Across all queries, 22.3% and 16.6% of PFN outputs lie outside that hull at 20k and 100k. Likelihood tempering can describe predictions outside the interval spanned by the two exact family predictions (Appendix G). Family reweighting cannot represent predictions outside the interval between the two exact family predictives; this restriction helps interpret the all-query comparison, but does not by itself determine which likelihood exponent fits best.

5.2 Training Changes the Matched-Length Approximation

Continuing the same five PFNs from 20k to 100k decreases PFN-to-BMA KL at every saved checkpoint on the original evaluation set (Figure 2). On the independent evaluation set, median KL falls from 0.004072 to 0.001641, while the fitted likelihood exponent rises from 0.883 to 0.959 and the family parameter from 0.397 to 0.816.

At 20k, fixed pooling is closer than BMA in four of five original runs and in the independent-evaluation median. By 40k, BMA is closer in all five models; by 100k, fixed pooling is less faithful than every context-dependent family-weighting candidate. The prior and evaluation contexts are fixed, so this reversal reflects changed PFN predictions, not a changed reference.

Improvement also occurs on contexts whose labeled signatures were never encountered during training, so exact context repetition alone does not account for it (Appendix J).

A separate architecture comparison with three models per architecture and matched optimizer settings gives median PFN-to-BMA KL of 0.005146 versus 0.001754, and family parameter $a = 0.207$ versus 0.790, for base and larger models at 20k. Appendix K gives the settings.

Table 2: Each entry is the cross-seed median KL divergence $D_{\mathrm{KL}}(\mathrm{PFN} \| \mathrm{candidate})$ over the same five PFNs; lower is closer. Isotonic candidates use data-dependent monotone fits. Among the one-parameter candidates shown, likelihood tempering has the lowest median KL at both checkpoints. Posterior-to-prior shrinkage is a post hoc point-estimate robustness comparison, outside the prospectively fixed candidate set and bootstrap analysis.

Candidate description	Fit complexity	20k KL	100k KL
MAP family selection	0	0.019974	0.014875
Exact BMA	0	0.004072	0.001641
Posterior-to-prior shrinkage	1	0.003782	0.001633
Fixed family pooling	1	0.003482	0.003010
Output-only isotonic	Nonparametric	0.003769	0.001635
Family tempering	1	0.003251	0.001578
Family isotonic	Nonparametric	0.003214	0.001568
Likelihood tempering	1	0.003220	**0.001551**
Family tempering + output calibration	3	**0.003180**	0.001577

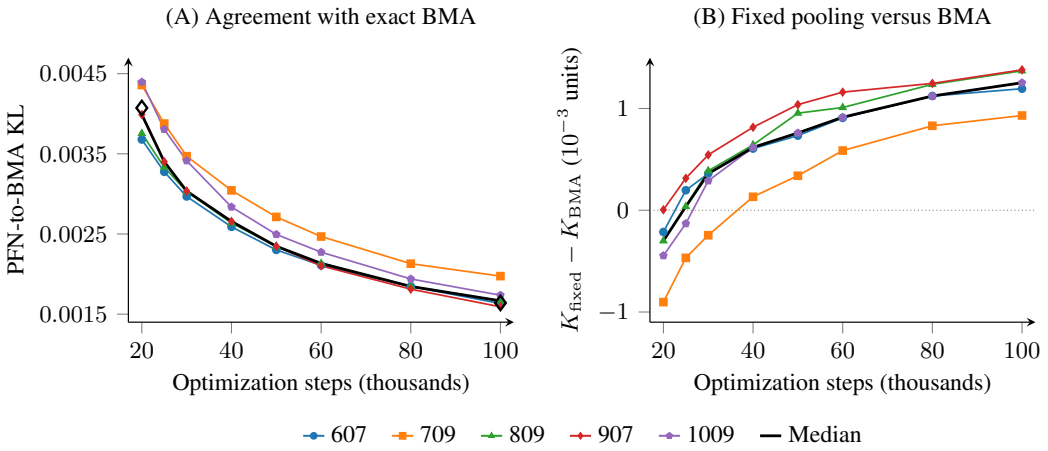


Figure 2: Panel A follows five PFNs and their median on the original evaluation set; open diamonds are independent-evaluation medians at 20k and 100k. Panel B compares fixed pooling with exact BMA: positive differences favor BMA, negative differences favor fixed pooling. By 40k, BMA is closer in every model. Colors and markers identify seeds; black lines are medians.

Table 3: Matched-length results across priors. Entries are cross-seed medians; R is the median seed-wise relative reduction in PFN-matching KL from likelihood tempering versus BMA. L/F counts models where likelihood fits more closely than family tempering. Each prior has five independent models.

Prior	Steps	PFN-to-BMA KL	τ	R (%)	L/F
Additive/tree	20k	0.004072	0.883	22.0	4/5
Additive/tree	100k	0.001641	0.959	6.4	5/5
Additive/quadratic complement	20k	0.001337	0.917	14.2	5/5
Additive/quadratic complement	100k	0.000771	0.951	7.8	5/5

5.3 Evidence Attenuation Recurs Under a Second Prior

We retain $\mathcal{H}_A$ and replace the tree family with quadratic-threshold functions outside that specified additive family. The resulting families contain 980 and 5,720 distinct functions and are disjoint. Five fresh PFNs use the same architecture, optimizer, $n = 8$, noise rate, and 20k/100k checkpoints. Candidate definitions and fitting rules are retained, except that the empty shared-component refactorization is inapplicable; shrinkage is prospectively specified as a secondary comparison. Appendix M gives the exact construction and protocol.

Likelihood tempering fits more closely than BMA, family tempering, and the tested output-only transformations in all five models at both checkpoints. All ten paired within-model 95% bootstrap intervals favor likelihood over family tempering. Further training reduces PFN-to-BMA KL and moves both fitted tempering parameters toward one in every model (Table 3). The attenuation therefore recurs under a different prior. However, fixed pooling is closer than BMA in only one model at 20k, unlike the original prior's four-of-five result. The early pooling comparison is prior-dependent, not a necessary stage of PFN training.

6 Interpreting the Fitted Family Weights

The fitted parameter depends on the prior factorization. We hold the PFN outputs and marginal hypothesis prior fixed, but replace the additive/tree hierarchy with additive-only, tree-only, and shared components. Exact BMA is unchanged. Nevertheless, the fitted component-tempering parameter increases by a median of 0.268 at 20k and 0.099 at 100k, with every paired interval positive on the independent evaluation set. The original additive/tree factorization remains more faithful in all five PFNs at both checkpoints (Appendix F, Eq. 11). A fitted family parameter can therefore change

solely because the analyst changes the prior representation, even when neither the network nor its Bayesian target changes.

Cross-query consistency improves with training but remains incomplete. Under asymmetric task-family priors, tempering the Bayes factor while preserving prior odds is more faithful than tempering posterior odds (Appendices F and I). These analyses do not identify an internal family variable.

7 Discussion

PFNs let researchers specify prediction algorithms through synthetic tasks, but the learned predictor's response to evidence still needs to be studied. PriorFuse enables that assessment to be concrete as additional training and context reveal different aspects of learned inference. Further training improves matched-length Bayesian approximation and reduces fitted attenuation under both tested priors. Additional inference-time context improves predictive log loss while increasing excess predictive log loss relative to BMA. Evaluating the response to increasing evidence therefore tests a property that predictive performance at a single context length can miss.

At the tested longer lengths, much of the Bayesian departure admits a compact description, as a likelihood exponent fitted at eight examples and rescaled by $8/n$ retains approximately 99% of the PFN-matching KL improvement over BMA achieved by separately fitting that candidate at each longer length.

The inverse-length pattern may reflect architectural or context-dependent normalization effects. Müller et al. (2025) discuss counting limitations under repeated identical observations, whereas our contexts use distinct input locations. Identifying the mechanism would require targeted interventions.

The small matched-length PFN-to-BMA KL values provide high-resolution diagnostics of the learned prediction rule. Assessing Bayesian fidelity requires measuring Bayesian departure alongside fitted coefficients, whose distance from one can change differently from agreement with BMA (Appendix N). Regrouping the same prior also changes fitted family weights while leaving both predictors unchanged. Together, these findings motivate reporting predictive performance, explicitly named measures of Bayesian departure, and fitted descriptions when assessing how training or context changes a PFN.

8 Limitations

Both priors use a sixteen-input binary domain, and the second prior changes family content, hypothesis count, and overlap together. The context-length sweep tests generalization of $n = 8$-trained PFNs, not matched training at every length; it cannot establish why their length dependence occurs. Context-dependent feature normalization, the query-label placeholder, and bound-hitting short-context fits also limit simple evidence-weighting interpretations. Likelihood tempering is a useful candidate, not a complete or unique explanation: output mappings and flexible family fits can be better in some conditions, and query-level differences between candidate and PFN predictions remain. Additional synthetic and public-data specialist-interpolation checks show no consistent benefit and lack an exact BMA reference (Appendix K). The inverse-length rule is a post hoc description over $n = 9, \ldots, 12$, not an independently selected scaling law. No internal mechanism, causal capacity scaling law, or universal PFN behavior is established.

9 Conclusion

PriorFuse provides a benchmark that helps researchers understand what PFNs have learned about using evidence, rather than infer it from predictive performance alone. Under both tested priors, further training improves matched-length Bayesian approximation. However, the same trained model can predict better with more context while falling further behind exact BMA. The constant-versus-scaled transfer comparison shows that much of the longer-context Bayesian departure has a simple quantitative description, while factorization tests show why fitted parameters need not be properties of the network alone. Our benchmark provides a controlled way to assess how training changes a learned prediction rule and how that rule responds when more observations become available.

AI Use Statement

Generative AI tools were used to assist with, review, and refine work across all of the following areas: literature discovery, information retrieval, experiment design, conceptual frameworks and mathematical claims, interpretation of results, technical terminology, grammar of author-written text, and code review, debugging, and revision. All AI-assisted outputs were treated as suggestions and were independently reviewed and verified by the authors.

Ethics Statement

The primary study uses synthetic data and does not involve human participants or sensitive personal information. Secondary boundary analyses use public tabular benchmark datasets. We report negative and heterogeneous boundary results and do not claim deployment guarantees.

Reproducibility Statement

The benchmark, exact inference, training, candidate fitting, and statistical protocols are specified in Section 3 and the appendices. The Code and Data Supplement includes scientific code, configurations, fixed seeds, per-model results, and bootstrap summaries for the original study, second-prior experiment, inference-time context-length analysis, and post hoc transfer comparison. Its self-contained lightweight workflow regenerates numerical tables and plotting-coordinate CSVs from frozen outputs, without training or inference. Full retraining and candidate refitting require the additional resources described in the supplement. Manuscript figure LaTeX sources and PDF regeneration are outside this lightweight workflow.

References

Ekin Akyürek, Dale Schuurmans, Jacob Andreas, Tengyu Ma, and Denny Zhou. What learning algorithm is in-context learning? Investigations with linear models. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=0g0X4H8yN4I.

Yu Bai, Fan Chen, Huan Wang, Caiming Xiong, and Song Mei. Transformers as statisticians: Provable in-context learning with in-context algorithm selection. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/b2e63e36c57e153b9015fece2352a9f9-Abstract-Conference.html.

Satwik Bhattamishra, Arkil Patel, Phil Blunsom, and Varun Kanade. Understanding in-context learning in transformers and LLMs by learning to learn discrete functions. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/f7e7fabd73b3df96c54a320862afcb78-Abstract-Conference.html.

Bernd Bischl, Giuseppe Casalicchio, Matthias Feurer, Pieter Gijsbers, Frank Hutter, Michel Lang, Rafael G. Mantovani, Jan N. van Rijn, and Joaquin Vanschoren. OpenML benchmarking suites. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2021. URL https://arxiv.org/abs/1708.03731.

Pier Giovanni Bissiri, Chris C. Holmes, and Stephen G. Walker. A general framework for updating belief distributions. *Journal of the Royal Statistical Society: Series B*, 78(5):1103–1130, 2016. doi: 10.1111/rssb.12158.

Blake Bordelon, Mary Letey, and Cengiz Pehlevan. Theory of scaling laws for in-context regression: Depth, width, context and time. In *International Conference on Learning Representations*, 2026.

Faris Chaudhry and Siddhant Gadkari. Implicit statistical inference in transformers: Approximating likelihood-ratio tests in-context. *arXiv preprint arXiv:2603.10573*, 2026. URL https://arxiv.org/abs/2603.10573.

Aaron Defazio, Xingyu (Alice) Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, and Ashok Cutkosky. The road less scheduled. In *Advances in Neural Information Processing Systems*, volume 37, 2024. URL https://papers.nips.cc/paper_files/paper/2024/hash/136b9a13861308c8948cd308ccd02658-Abstract-Conference.html.

Samet Demir and Zafer Dogan. How data mixing shapes in-context learning: Asymptotic equivalence for transformers with MLPs. In *Advances in Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=615vk8hmeH.

Fabian Falck, Ziyu Wang, and Christopher C. Holmes. Is in-context learning in large language models Bayesian? A martingale perspective. In *International Conference on Machine Learning*, 2024. URL https://proceedings.mlr.press/v235/falck24a.html.

Tilmann Gneiting and Adrian E. Raftery. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association*, 102(477):359–378, 2007.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *International Conference on Machine Learning*, pp. 1321–1330, 2017.

Jennifer A. Hoeting, David Madigan, Adrian E. Raftery, and Chris T. Volinsky. Bayesian model averaging: A tutorial. *Statistical Science*, 14(4):382–417, 1999.

Noah Hollmann, Samuel Müller, Katharina Eggensperger, and Frank Hutter. TabPFN: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=cp5PvcI6w8_.

Noah Hollmann, Samuel Müller, Lennart Purucker, Arjun Krishnakumar, Max Körfer, Shi Bin Hoo, Robin Tibor Schirrmeister, and Frank Hutter. Accurate predictions on small data with a tabular foundation model. *Nature*, 637:319–326, 2025. doi: 10.1038/s41586-024-08328-6.

Gyeonghun Kang, Changwoo J. Lee, and Xiang Cheng. Transformers can learn posterior predictive distributions in-context. *arXiv preprint arXiv:2605.26713*, 2026. URL https://arxiv.org/abs/2605.26713.

Shang Liu, Zhongze Cai, Guanting Chen, and Xiaocheng Li. Towards better understanding of in-context learning ability from in-context uncertainty quantification. *Transactions on Machine Learning Research*, 2025. URL https://openreview.net/forum?id=Jutpbhheoy.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers can do Bayesian inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=KSugKcbNf9.

Samuel Müller, Arik Reuter, Noah Hollmann, David Rügamer, and Frank Hutter. Position: The future of Bayesian prediction is prior-fitted. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*. PMLR, 2025. URL https://proceedings.mlr.press/v267/muller25d.html.

Thomas Nagler. Statistical foundations of prior-data fitted networks. In *International Conference on Machine Learning*, 2023. URL https://proceedings.mlr.press/v202/nagler23a.html.

Madhur Panwar, Kabir Ahuja, and Navin Goyal. In-context learning through the Bayesian prism. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/hash/d81cd83e7f6748af351485d73f305483-Abstract-Conference.html.

Core Francisco Park, Ekdeep Singh Lubana, Itamar Pres, and Hidenori Tanaka. Competition dynamics shape algorithmic phases of in-context learning. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=XgH1wfHSX8. Spotlight.

Reese Pathak, Rajat Sen, Weihao Kong, and Abhimanyu Das. Transformers can optimally learn regression mixture models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=sLkj91HIZU.

Alexander Pfefferle, Johannes Hog, Lennart Purucker, and Frank Hutter. nanoTabPFN: A lightweight and educational reimplementation of TabPFN. *arXiv preprint arXiv:2511.03634*, 2025.

Abdessamed Qchohi and Simone Rossi. A Bayesian perspective on the role of epistemic uncertainty for delayed generalization in in-context learning. *arXiv preprint arXiv:2604.12434*, 2026. URL https://arxiv.org/abs/2604.12434.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A tabular foundation model for in-context learning on large data. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 50817–50847, 2025. URL https://proceedings.mlr.press/v267/qu25d.html.

Allan Raventós, Mansheej Paul, Feng Chen, and Surya Ganguli. Pretraining task diversity and the emergence of non-Bayesian in-context learning for regression. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://papers.neurips.cc/paper_files/paper/2023/hash/2e10b2c2e1aa4f8083c37dfe269873f8-Abstract-Conference.html.

Arik Reuter, Tim G. J. Rudner, Vincent Fortuin, and David Rügamer. Can transformers learn full Bayesian inference in context? In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 51531–51582, 2025. URL https://proceedings.mlr.press/v267/reuter25a.html.

Zeynep Türkmen, Kürşat Kaya, Alexander Pfefferle, and Frank Hutter. Towards evaluating data priors for tabular foundation models. *arXiv preprint arXiv:2606.29241*, 2026. URL https://arxiv.org/abs/2606.29241.

Szilvia Ujváry, Anna Mészáros, Wieland Brendel, Patrik Reizinger, and Ferenc Huszár. Implicit Bayesian inference is an insufficient explanation of language model behavior in compositional tasks. In *DeLTa Workshop at the International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=YGcCR7uFT1.

Tomoya Wakayama and Taiji Suzuki. In-context learning is provably Bayesian inference: A generalization theory for meta-learning. *arXiv preprint arXiv:2510.10981*, 2025.

Jianzhe Wei, Siyu Chen, Jianliang He, and Zhuoran Yang. How transformers learn causal structures in-context: Explainable mechanism meets theoretical guarantee. In *International Conference on Learning Representations*, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/934ebba637be11105cff78e86a99eb08-Abstract-Conference.html.

Jingfeng Wu, Difan Zou, Zixiang Chen, Vladimir Braverman, Quanquan Gu, and Peter Bartlett. How many pretraining tasks are needed for in-context learning of linear regression? In *International Conference on Learning Representations*, 2024.

Daniel Wurgaft, Ekdeep S. Lubana, Core Francisco Park, Hidenori Tanaka, Gautam Reddy, and Noah D. Goodman. In-context learning strategies emerge rationally. In *Advances in Neural Information Processing Systems*, volume 38, 2025. doi: 10.52202/085713-3950. URL https://papers.nips.cc/paper_files/paper/2025/hash/abb22b1e4b4ab2988862be9d7ed083f9-Abstract-Conference.html.

Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit Bayesian inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=RdJVFCHjUMI.

Derek Qiang Xu, F. Olcay Cirit, Reza Asadi, Yizhou Sun, and Wei Wang. Mixture of in-context prompters for tabular PFNs. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=2fojNANZSv.

Bianca Zadrozny and Charles Elkan. Transforming classifier scores into accurate multiclass probability estimates. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 694–699, 2002. doi: 10.1145/775047.775151.

Xiyuan Zhang, Danielle Maddix Robinson, Junming Yin, Nick Erickson, Abdul Fatir Ansari, Boran Han, Shuai Zhang, Leman Akoglu, Christos Faloutsos, Michael W. Mahoney, Tony Hu, Huzefa Rangwala, George Karypis, and Yuyang Wang. Mitra: Mixed synthetic priors for enhancing tabular foundation models. In *Advances in Neural Information Processing Systems*, volume 38, 2025a. URL https://papers.nips.cc/paper_files/paper/2025/hash/177d68f4adef163b7b123b5c5adb3c60-Abstract-Conference.html.

Yufeng Zhang, Fengzhuo Zhang, Zhuoran Yang, and Zhaoran Wang. What and how does in-context learning learn? Bayesian model averaging, parameterization, and generalization. In *International Conference on Artificial Intelligence and Statistics*, 2025b. URL https://proceedings.mlr.press/v258/zhang25d.html.

A Exact Benchmark and Prior Construction

The input domain contains all 16 vectors in $\{0,1\}^4$. The additive family enumerates

$$h_{w,b}(x) = \mathbf{1}\left[\sum_{j=1}^{4} w_j x_j + b \geq 0\right], \qquad w_j \in \{-2,-1,0,1,2\}, \quad b \in \{-4,\ldots,4\}.$$

Deduplication by complete truth table leaves 980 functions. The tree family enumerates a root split on one feature, optional child splits on features distinct from the root, and every binary leaf assignment, with depth at most two. Deduplication leaves 166 functions. Fifty-eight truth tables occur in both sets, yielding 1,088 unique tables in their union.

With balanced family sampling, 58/166 = 34.9% of tree functions and 58/980 = 5.9% of additive functions are shared. Approximately $\frac{1}{2}(58/980) + \frac{1}{2}(58/166) = 20.4\%$ of task draws use a truth table represented in both families. The task family is therefore a latent variable in the generative process, not a uniquely identifiable property of every function. Among family-exclusive hypotheses, a tree-only truth table receives about $980/166 = 5.9$ times the prior mass of an additive-only truth table. Shared truth tables receive contributions from both families.

For context labels y_D and input locations X_D, hypothesis likelihood is

$$p(y_D \mid X_D, h) = \prod_{i=1}^{n} (1-\eta)^{\mathbf{1}[h(x_i)=y_i]} \eta^{\mathbf{1}[h(x_i)\neq y_i]}.$$

Equation 1 averages this likelihood under the uniform within-family prior. The family-independent input-location probability cancels from posterior odds. Equivalently, the marginal hypothesis prior in Eq. 3 sums to one, including both mass contributions for shared functions, and produces exactly the same BMA predictor.

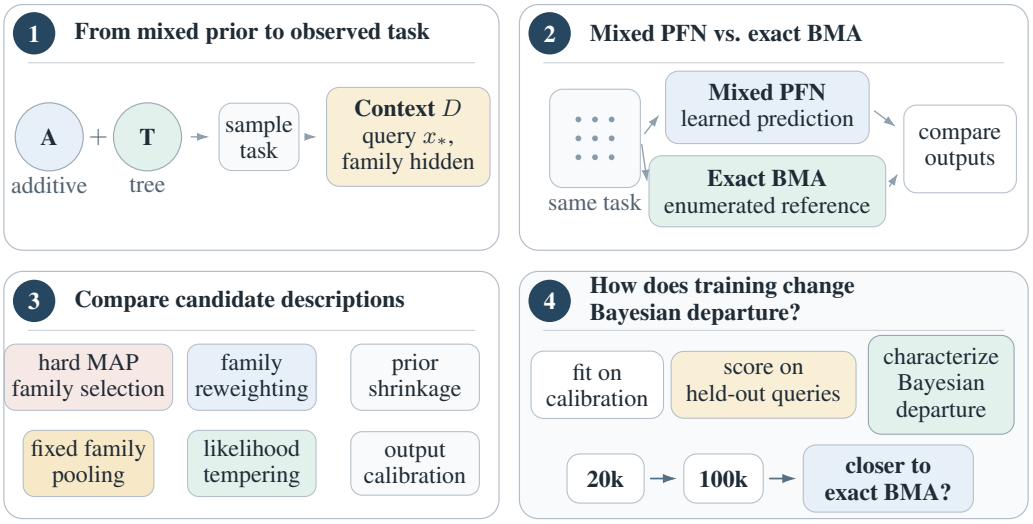


Figure 3: PriorFuse compares PFN predictions with exact BMA for the same context and query. Candidate descriptions are fitted on calibration contexts and evaluated on held-out queries. Repeating the comparison across training checkpoints shows which descriptions become more faithful and where they still fail to match PFN predictions.

B PFN Architecture, Training, and Query Protocol

The core nanoTabPFN (Pfefferle et al., 2025) has three transformer blocks, four attention heads, embedding dimension 64, MLP hidden dimension 128, and two output logits. Each block applies feature attention, row attention, and a feed-forward update. All eight query rows are supplied

simultaneously. Context rows attend only to context rows. Query rows attend to labeled context rows but not to one another, and no row-position embedding is used. For each feature, the encoder computes the mean and sample standard deviation from labeled context rows only. All context and query feature values are standardized using those statistics, with 10^{-20} added to the standard deviation, then clipped to $[-100, 100]$. Thus a zero-variance context feature does not cause division by zero, although a query with a different value can reach the clipping limit. A shared affine map from one scalar to the embedding dimension encodes each feature cell. A separate affine map encodes target cells; missing query labels are filled with the mean of the observed context labels, never the actual query labels. Feature and target cells are concatenated along the column axis. The decoder reads the target-cell embedding of each query and applies two affine maps separated by GELU to produce two class logits.

A seeded random permutation of all 16 inputs determines eight labeled context rows and eight query rows for every task. Training samples a family, a truth table uniformly within that family, a fresh row permutation, and independent label flips with $\eta = 0.10$. Cross-entropy is averaged over the batch and all eight query labels. Training uses Schedule-Free AdamW (Defazio et al., 2024), learning rate 0.004, weight decay 0, batch size 64, gradient clipping at 1.0, and deterministic algorithm settings. The core 20k run contains 1.28 million sampled tasks per model. The five mixed PFNs continue from complete model, optimizer, random, and task-stream states to 100k, or 6.40 million total tasks per model.

The original evaluation set uses 4k calibration and 6k evaluation contexts. On calibration contexts, all eight query positions fit candidate parameters. On evaluation contexts, the first three positions after the seeded permutation are reserved for per-context diagnostic weight fitting and the final five support the headline evaluation. The independent calibration and evaluation sets contain 4k and 10k contexts, respectively, generated with a new seed after the candidate equations, fitting routines, metrics, and subgroup definitions were fixed; the same analysis is applied without further candidate selection.

C Candidate Predictions and Context-Likelihood Noise Reparameterization

Hard maximum a posteriori (MAP) family selection uses the family with the higher posterior probability. Fixed family pooling fits one global $g \in [0, 1]$. Family logistic mapping uses $g(D) = \mathrm{sigmoid}(\alpha + \beta \,\mathrm{logit}\, r(D))$. Family isotonic mapping first estimates one bounded context target on the eight calibration queries and fits a nondecreasing map from $r(D)$ with equal context weights. It is a flexible monotone comparator, not a test against every unconstrained function.

Both isotonic candidates use scikit-learn's increasing isotonic regression with fitted values constrained to $[0, 1]$. Prediction linearly interpolates between fitted threshold points and clips inputs outside the calibration range to the corresponding endpoint. Family isotonic mapping retains its two-stage construction: context-specific weights are fitted first, followed by least-squares isotonic regression with equal context weights.

Posterior-to-prior shrinkage is defined by

$$\widetilde{p}_\lambda(h \mid D) = (1 - \lambda)p(h \mid D) + \lambda\mu(h), \qquad 0 \le \lambda \le 1. \tag{9}$$

Its predictive form is $q_\lambda = (1 - \lambda)p_{\mathrm{BMA}} + \lambda q_{\mathrm{prior}}$, where q_{prior} is the unconditional predictive under $\mu(h)$. In this benchmark q_{prior} lies very close to 0.5 for every query input, so this control is also close to direct probability shrinkage toward 0.5.

Output-only logistic and isotonic candidates transform p_{BMA} directly. The joint model is

$$m_a = (1 - g_a)p_A + g_a p_T, \qquad q_{\mathrm{joint}} = \mathrm{sigmoid}(c_0 + c_1 \,\mathrm{logit}\, m_a),$$

and fits a, c_0, c_1 jointly. A low-complexity component-aware output model is also evaluated in the supplement.

For m context-label mismatches among n examples, the powered likelihood is proportional to $(1 - \eta)^{\tau(n-m)}\eta^{\tau m}$. Defining η_{eff} by

$$\frac{1 - \eta_{\mathrm{eff}}}{\eta_{\mathrm{eff}}} = \left(\frac{1 - \eta}{\eta}\right)^\tau$$

gives Eq. 7. The prediction under likelihood tempering and the context-likelihood noise reparameterization therefore agree to numerical precision because they induce the same posterior over hypotheses and both retain the original query-label channel.

D Evaluation and Statistical Analysis

Except for the metric-sensitivity fits in Appendix G and the two isotonic candidates, candidate descriptions are fitted by minimizing PFN soft-target cross-entropy on calibration queries. The output-only isotonic candidate uses least-squares isotonic regression directly on BMA and PFN probabilities. Family isotonic mapping first fits a bounded mixture weight separately for each calibration context, then applies least-squares isotonic regression from posterior task family probability to those context-level weights. For fixed PFN targets,

$$\mathrm{CE}(p_\theta, q) = H(p_\theta) + D_{\mathrm{KL}}(p_\theta \| q),$$

so soft-target cross-entropy differs from $D_{\mathrm{KL}}(p_\theta \| q)$ only by the entropy of the PFN distribution and therefore does not provide an independent metric. Probability MAE and logit RMSE are reported as distinct sensitivities. Excess predictive log-loss risk relative to exact BMA is evaluated separately, using KL with BMA as its first argument and the predictor as its second.

For the original matched-length evaluation, within each seed and checkpoint, 2k context-level cluster-bootstrap draws independently resample calibration and evaluation contexts, refit the candidates in the locked comparison, and keep queries together within each sampled context. Intervals are conditional on one trained PFN; consistency across independently trained PFNs supplies model-level replication. Paired long-horizon checkpoint comparisons reuse the same resampled context indices across the two checkpoints. The independent-evaluation 20k and 100k endpoint analyses use separate bootstrap draws. Original post hoc shrinkage results are point estimates only; the new experiments have separately specified protocols in Appendices M and N. The independent evaluation set contains 50k held-out query predictions per seed and checkpoint.

Calibration and evaluation contexts were sampled independently from a finite discrete context distribution rather than forced to be disjoint. After canonicalizing context-row order, 23 of 10k independent-evaluation contexts also occur in the calibration sample; the original 6k-context evaluation has 9 such overlaps. The independent evaluation contains 9,974 unique unordered contexts. These repeats arise naturally under independent sampling and were retained.

E Candidate Results on the Independent Evaluation Set

Table 4: Independent-evaluation PFN-matching KL at 20k. Lower is more faithful.

Seed	BMA	Fixed	Family tempering	Family isotonic	Likelihood tempering	Joint
607	.003708	.003448	.002990	.002955	.002908	.002946
709	.004378	.003482	.003251	.003214	.003260	.003180
809	.003772	.003456	.003026	.003000	.002944	.002970
907	.004072	.004073	.003490	.003456	.003220	.003357
1009	.004438	.003938	.003572	.003549	.003450	.003487
Median	.004072	.003482	.003251	.003214	.003220	.003180

Table 5: Independent-evaluation PFN-matching KL at 100k.

Seed	BMA	Fixed	Family tempering	Family isotonic	Likelihood tempering	Joint
607	.001625	.002861	.001540	.001516	.001511	.001535
709	.001909	.002892	.001754	.001733	.001708	.001732
809	.001641	.003028	.001578	.001568	.001551	.001577
907	.001601	.003010	.001550	.001545	.001499	.001541
1009	.001700	.003018	.001641	.001618	.001625	.001641
Median	.001641	.003010	.001578	.001568	.001551	.001577

Table 6: Posterior-to-prior shrinkage on the independent evaluation set. These are post hoc point-estimate robustness results, not part of the locked candidate bootstrap analysis; lower KL is more faithful. Likelihood tempering is shown for the matched comparison.

	20k steps			100k steps		
Seed	λ	Shrinkage KL	Likelihood KL	λ	Shrinkage KL	Likelihood KL
607	.041	.003529	.002908	.011	.001612	.001511
709	.051	.004103	.003260	.018	.001874	.001708
809	.045	.003572	.002944	.009	.001633	.001551
907	.054	.003782	.003220	.014	.001577	.001499
1009	.051	.004173	.003450	.006	.001695	.001625
Median	.051	.003782	.003220	.011	.001633	.001551

Table 7: Likelihood exponents τ and paired family-minus-likelihood KL intervals on the independent evaluation set. Positive differences favor likelihood tempering.

	20k		100k	
Seed	τ	Difference [95% CI]	τ	Difference [95% CI]
607	.887	.000082 [.000055,.000111]	.956	.000030 [.000023,.000037]
709	.870	-.000009 [-.000044,.000026]	.941	.000047 [.000036,.000056]
809	.884	.000083 [.000055,.000109]	.961	.000027 [.000021,.000033]
907	.883	.000270 [.000245,.000298]	.959	.000052 [.000046,.000058]
1009	.873	.000121 [.000089,.000156]	.964	.000016 [.000010,.000023]

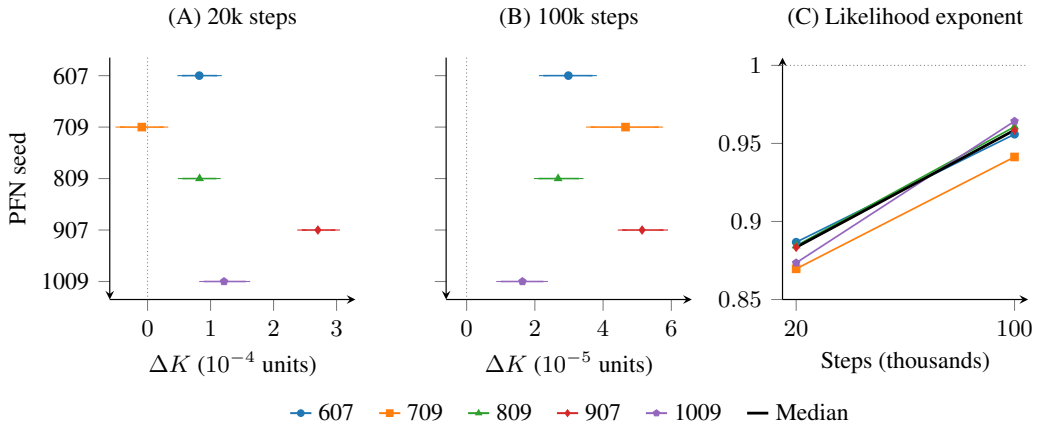


Figure 4: Panels A and B show the paired KL difference $\Delta K = K_{\text{family}} - K_{\text{likelihood}}$ on the independent evaluation set; positive values favor likelihood tempering. Bars are 95% context-level cluster-bootstrap intervals within each PFN. Panel C shows the likelihood exponent for each model and the cross-seed median; the dotted reference is exact BMA at $\tau = 1$. Colors and markers identify PFN seeds.

F Factorization and Cross-Query Diagnostics

For evaluation context c and query q, let $p_{\theta,c,q}$ denote the mixed PFN probability and let $p_{A,c,q}$ and $p_{T,c,q}$ denote the exact additive and tree posterior predictives. The implied tree-family weight is

$$\widehat{g}_{c,q} = \frac{p_{\theta,c,q} - p_{A,c,q}}{p_{T,c,q} - p_{A,c,q}}. \tag{10}$$

It is defined only when $|p_{T,c,q} - p_{A,c,q}| \geq \delta$ with $\delta = 0.20$ and $p_{\theta,c,q} \in [\min(p_{A,c,q}, p_{T,c,q}) - \epsilon, \max(p_{A,c,q}, p_{T,c,q}) + \epsilon]$ with $\epsilon = 10^{-7}$. These are the *informative inside hull* rows used in the main text. Rows failing either condition are excluded, not clipped. Within-context consistency summaries require at least two eligible queries. With $z_{c,q} = \mathbf{1}[\widehat{g}_{c,q} > 1/2]$. Writing $\mathcal{E}_c$ for the eligible queries in context c, define $\overline{z}_c = |\mathcal{E}_c|^{-1} \sum_{q \in \mathcal{E}_c} z_{c,q}$. The majority family-preference fraction is $\max(\overline{z}_c, 1 - \overline{z}_c)$; family preference is unanimous when $\overline{z}_c \in \{0, 1\}$. For these two summaries, a tied implied weight is grouped with additive preference. Writing $\overline{\widehat{g}}_c = |\mathcal{E}_c|^{-1} \sum_{q \in \mathcal{E}_c} \widehat{g}_{c,q}$, agreement with the posterior-favored family compares $\operatorname{sign}(\overline{\widehat{g}}_c - 1/2)$ with $\operatorname{sign}(r(D_c) - 1/2)$, retaining exact ties as a separate outcome.

Each evaluation context has eight query positions after the fixed seeded permutation. The prespecified split uses the first three as diagnostic fitting queries and the final five for held-out evaluation. As sensitivity analyses, we also enumerate all 70 four-query/four-query partitions and 20 prespecified unique three-query/five-query partitions. For each context and partition, a single $g_c \in [0, 1]$ is fitted by PFN soft-target cross-entropy on diagnostic fitting queries and applied unchanged to the complementary queries. The scalar fitted directly on the evaluation queries is an oracle fit, not a held-out prediction. Both diagnostics use PFN outputs for fitting, so neither is a standalone predictor based only on the labeled context.

The disjoint additive-only, tree-only, and shared components have prior masses

$$(\pi_A, \pi_T, \pi_S) = \left(\tfrac{1}{2}\tfrac{922}{980}, \tfrac{1}{2}\tfrac{108}{166}, \tfrac{1}{2}\left[\tfrac{58}{980} + \tfrac{58}{166}\right]\right) \approx (0.470408, 0.325301, 0.204291).$$

For component $k \in \{A, T, S\}$, let $E_k(D) = \sum_{h \in \mathcal{H}_k} \mu(h \mid k) p(D \mid h)$ and let $p_k(y_* \mid x_*, D)$ be its exact posterior predictive. The generalized component reweighting is

$$w_{k,a}(D) = \frac{\pi_k E_k(D)^a}{\sum_j \pi_j E_j(D)^a}, \qquad q_{3,a}(y_* \mid x_*, D) = \sum_k w_{k,a}(D) p_k(y_* \mid x_*, D). \tag{11}$$

At $a = 1$ this is exactly the original marginal-prior BMA prediction.

An equivalent three-component factorization separates additive-only, tree-only, and shared truth tables while preserving the same marginal hypothesis prior. At $a = 1$, it produces exactly the same BMA prediction. Its fitted coefficient is systematically higher, but its KL is also systematically worse than the original additive/tree mapping.

Table 8: Factorization sensitivity on the independent evaluation set. a_2 is the original additive/tree coefficient, a_3 the equivalent three-component coefficient, and $\Delta K = K_3 - K_2$.

	20k			100k		
Seed	a_2	a_3	ΔK	a_2	a_3	ΔK
607	.397	.665	.000138	.783	.891	.000029
709	.290	.601	.000201	.704	.834	.000027
809	.407	.661	.000113	.820	.905	.000016
907	.474	.692	.000078	.825	.909	.000014
1009	.343	.636	.000167	.816	.915	.000023

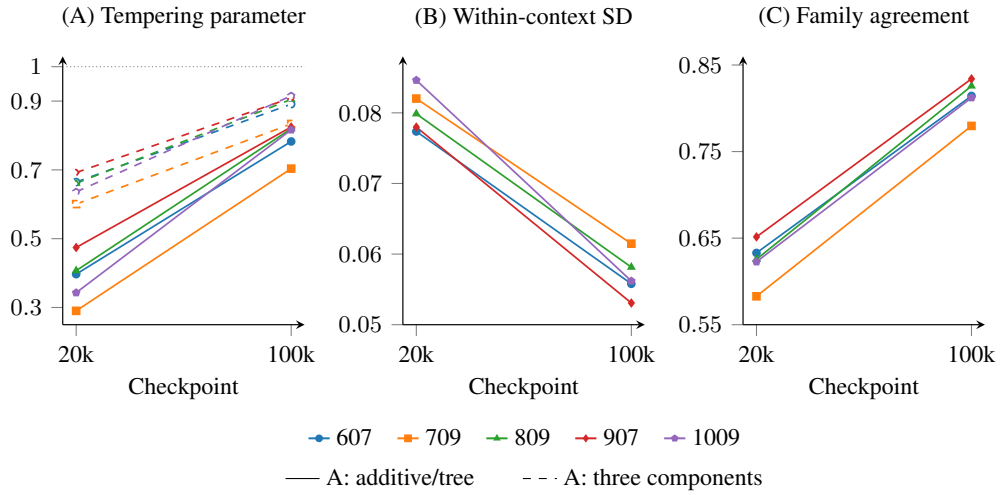


Figure 5: Independent-evaluation diagnostics. (A) The fitted family tempering parameter changes under an equivalent three-component factorization. (B) Within-context standard deviation of implied family weights. (C) Agreement with the posterior-favored family. Each line follows one PFN; colors and markers identify seeds. Only panel A compares two factorizations.

Table 9: Cross-seed median cross-query consistency diagnostics on the independent evaluation set.

Quantity	20k	100k
Within-context SD of implied weight	0.080	0.056
Within-context range	0.190	0.135
Majority family-preference fraction	0.800	0.857
Fraction with unanimous family preference	0.446	0.606
Agreement with posterior-favored family	0.626	0.814
All-query outside-hull rate	0.223	0.166

Across all 70 balanced 4/4 fitting/evaluation splits, a fitted shared weight beats exact BMA in all five observed PFNs at 20k but beats fixed family pooling in only two. Under the prespecified 3/5 split it beats neither baseline. At 100k it beats fixed family pooling in all five observed PFNs but never exact BMA. This indicates greater consistency across queries while showing that a single context-level weight remains incomplete.

G Evaluation Subsets and Metric Sensitivity

The family tempering candidate $g_a(D) = \mathrm{sigmoid}(a \,\mathrm{logit}\, r(D))$ is refitted on calibration rows under four objectives: $D_{\mathrm{KL}}(p_\theta \| q_a)$, $D_{\mathrm{KL}}(q_a \| p_\theta)$, probability MAE, and logit RMSE. We cross these with three weighting regimes. Equal-context weighting normalizes the included queries to total weight one within each context. Posterior-probability-stratified weighting divides $r(D)$ into ten fixed bins and gives every occupied bin equal total weight; it does not renormalize within context because $r(D)$ is constant across a context's queries. High-disagreement weighting retains only rows with $|p_T - p_A| \geq 0.20$ and then normalizes the retained rows within each context. Parameters are fitted only on calibration data and applied unchanged to held-out evaluation rows. Five seeds, two checkpoints, four objectives, and three weighting regimes give the 120 reported fits.

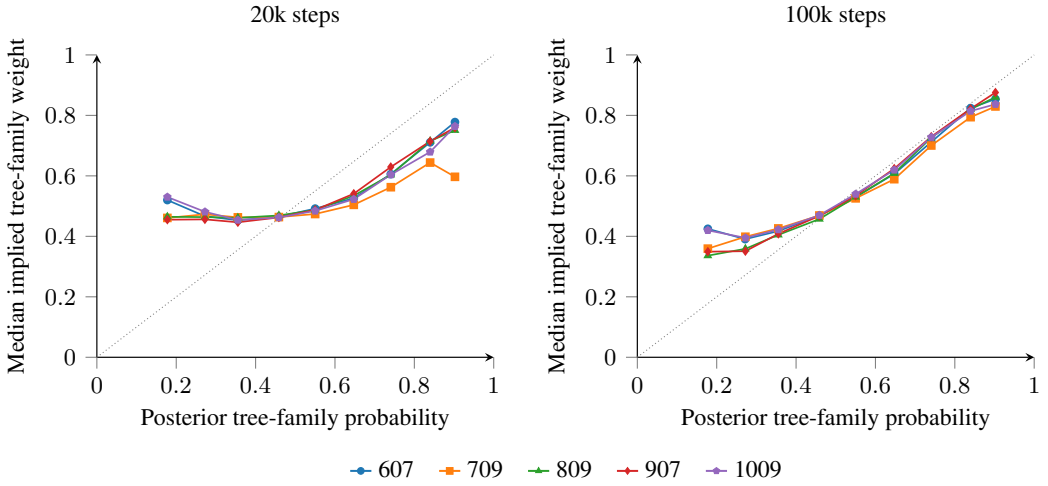


Figure 6: Binned median implied family weights on informative queries inside the family-prediction convex hull at both checkpoints, on the original evaluation set. The same query rows are used at 20k and 100k. Colors and markers identify PFNs; the dotted line is exact posterior weighting.

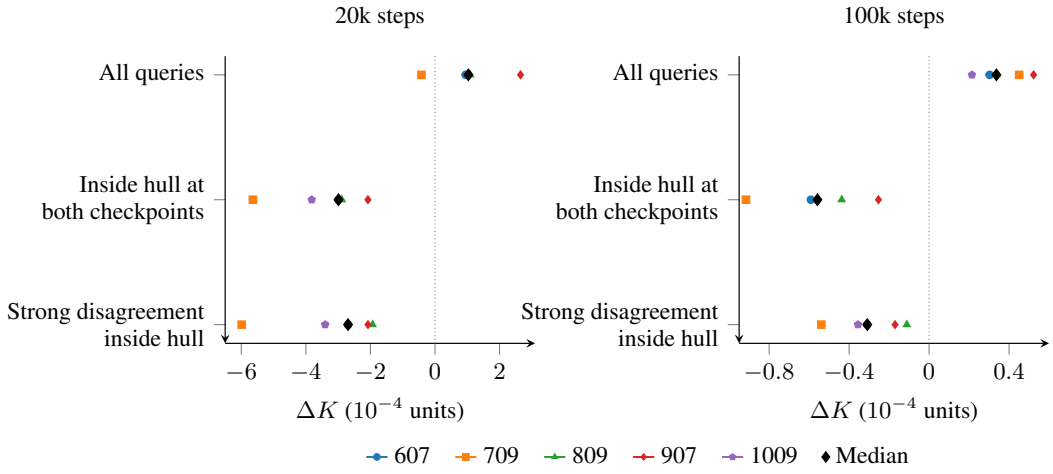


Figure 7: Family-minus-likelihood KL on the original evaluation set, with parameters fitted for each evaluation regime as specified in the analysis. Positive values favor likelihood tempering. The high-disagreement subset also requires $|p_T - p_A| \geq 0.20$. Colors and markers identify models; black diamonds are cross-seed medians.

The fraction of evaluation queries outside the family-prediction convex hull falls from a median of 22.35% at 20k to 16.60% at 100k on the independent evaluation set. On rows inside the family-prediction convex hull at both checkpoints, family tempering and family logistic mapping are more

faithful than likelihood tempering in all five models at both checkpoints. Across 120 combinations of four fitting objectives and three weighting regimes, every fitted family coefficient is below one and every matched 20k-to-100k coefficient increases. Values range from 0.244 to 0.487 at 20k and 0.632 to 0.828 at 100k, showing qualitative robustness but substantial metric dependence.

H Output-Level Controls

Table 10: Cross-seed median independent-evaluation PFN-matching KL for output controls.

Candidate	20k	100k
Exact BMA	.004072	.001641
Output-only logistic	.003774	.001633
Output-only isotonic	.003769	.001635
Family tempering	.003251	.001578
Family logistic	.003227	.001578
Joint family tempering + output	.003180	.001577

Output-only logistic and isotonic mappings are less faithful than family tempering in every seed/checkpoint comparison. Joint output calibration changes the family coefficient by a median $+0.024$ at 20k and $+0.006$ at 100k. Thus ordinary output calibration does not remove the attenuated task family weighting, although the joint model obtains the lowest independent-evaluation cross-seed median at 20k among the candidates in Table 2.

I Asymmetric Task Family Priors

The asymmetric-prior experiment uses three independently trained 20k PFNs at each of $\pi_T = 0.25$ and 0.75. All fitted parameters use 4k calibration contexts and are evaluated unchanged on 6k common contexts with exact posterior quantities recomputed under the corresponding training prior.

Let $\ell_\pi = \mathrm{logit}(\pi_T)$ and let the context log Bayes factor be $\ell_B(D) = \log p(D \mid T) - \log p(D \mid A) = \mathrm{logit}\, r_\pi(D) - \ell_\pi$. The two one-parameter candidates use

$$\begin{aligned} g_{\mathrm{post}}(D) &= \mathrm{sigmoid}(a_{\mathrm{post}}[\ell_\pi + \ell_B(D)])\,, \\ g_{\mathrm{evid}}(D) &= \mathrm{sigmoid}(\ell_\pi + a_{\mathrm{evid}}\ell_B(D))\,, \end{aligned} \qquad q_g = (1 - g(D))p_A + g(D)p_T. \tag{12}$$

Posterior-odds tempering scales both the prior log odds and the log Bayes factor by a_{post}; when $a_{\mathrm{post}} < 1$, both contract toward zero. Bayes-factor tempering preserves the prior odds and raises the context Bayes factor to the power a_{evid}. They coincide when $\pi_T = 0.5$. Both candidates reweight task families rather than individual hypotheses.

At $\pi_T = 0.25$, fixed family pooling near the training prior is already highly competitive and Bayes-factor tempering adds little. At $\pi_T = 0.75$, the added context response is clearer. Bayes-factor tempering nevertheless fits PFN probabilities more closely than posterior-odds tempering in all six models and on evaluation sets generated from the corresponding training prior. A descriptive pooled fit gives $c = -0.079$, prior-log-odds coefficient $b = 1.003$, and log-Bayes-factor coefficient $a = 0.341$; these summarize baseline alignment across separately trained models rather than identify an internal prior variable.

Table 11: Behavioral fidelity under asymmetric task family priors on the common evaluation set. Constant g is the fitted fixed family weight.

π_T	Seed	g	a_{evid}	Fixed	Posterior odds	Bayes factor	Logistic
0.25	2101	.252	.049	.002333	.004137	.002328	.002328
0.25	2201	.231	.069	.002670	.004615	.002684	.002648
0.25	2301	.247	.199	.002576	.003909	.002503	.002493
0.75	3101	.748	.486	.003444	.003218	.003094	.003071
0.75	3201	.730	.321	.003944	.003942	.003802	.003724
0.75	3301	.734	.302	.004014	.004073	.003872	.003816

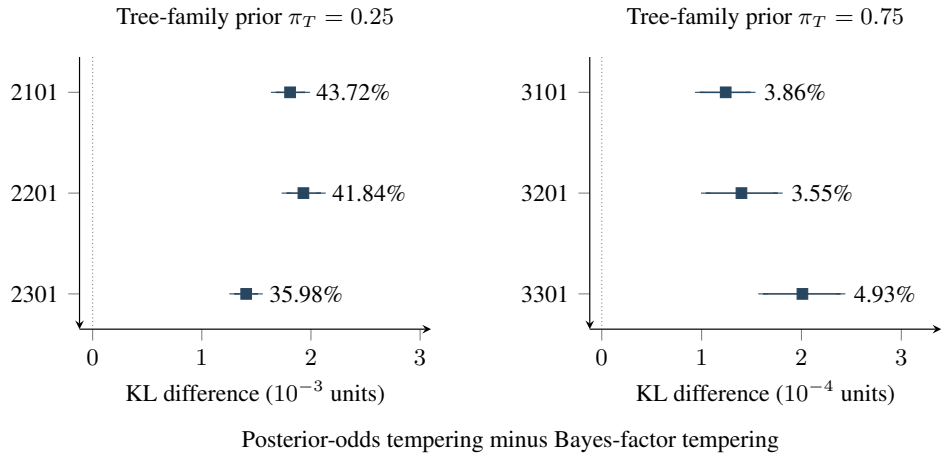


Figure 8: Paired posterior-odds-tempering minus Bayes-factor-tempering KL under asymmetric priors, evaluated on the common contexts. Positive values favor preserving prior odds while tempering the Bayes factor. Bars are 95% context-bootstrap intervals; percentages are relative KL reductions from posterior-odds tempering.

J Long-Horizon Training and Context Repetition

Table 12: Cross-seed medians on the original evaluation set.

Step	PFN-to-BMA KL	Fixed pooling KL	Family a
20k	.003989	.003465	.390
25k	.003402	.003472	.459
30k	.003036	.003418	.526
40k	.002657	.003279	.609
50k	.002347	.003254	.664
60k	.002134	.003143	.701
80k	.001846	.003058	.765
100k	.001666	.002971	.801

All five PFN-to-BMA KL point trajectories decrease at every saved checkpoint. The 100k family-tempering coefficient intervals remain below one, while every paired 80k-to-100k coefficient change is positive. The trajectory has therefore not plateaued empirically.

A canonical context signature records the unordered set of eight context input indices together with their noisy binary labels. Replaying each deterministic training stream through 100k leaves 768 to 809 evaluation contexts per PFN whose exact signatures never occurred. On these fixed subsets,

median PFN-to-BMA KL falls from 0.004814 at 20k to 0.002095 at 100k, with the same direction in all five observed models. This rules out exact context replay as the sole source of the trend, but not interpolation among closely related contexts or repeated latent functions.

K Predictive Risk, Specialists, and Boundary Probes

At 20k optimization steps, replacing exact components with leave-one-seed-out ensembles of finite additive and tree specialists preserves qualitative attenuation: median learned-component BMA KL is 0.002739, family-tempering KL is 0.002229, and fitted $a = 0.475$. This control is not exposure-matched. Each specialist sees 1.28 million tasks from one family, while a balanced mixed PFN sees about 0.64 million from each family, and four specialists are ensembled.

Table 13: Expected predictive scores on the independent evaluation set. Label expectations are analytic; contexts and queries are averaged over the fixed set.

Predictor	Expected log loss	Expected Brier score
Exact BMA	**0.621687**	**0.216153**
Mixed PFN, 20k median	0.625755	0.217999
Mixed PFN, 100k median	0.623365	0.216908

The supporting architecture comparison uses a six-layer model with embedding dimension 128, eight attention heads, MLP dimension 256, and two output logits, against the three-layer base model with embedding dimension 64, four heads, and MLP dimension 128. Both use learning rate 0.001, batch size 64, and 20k optimization steps, with three independent mixed-model seeds per architecture. The matched base seeds are 1401, 1501, and 1601; the larger-model seeds are 1101, 1201, and 1301. Median PFN-to-BMA KL is 0.005146 for the matched base and 0.001754 for the larger model; fitted family-tempering a is 0.207 and 0.790, respectively. These are family, not likelihood, exponents. The comparison shows that Bayesian departure can depend on model resources, but does not establish a capacity scaling law or the cause of the inference-time context-length profile.

Additional boundary analyses use TabICL's MLP-SCM and tree-SCM generators (Qu et al., 2025) at commit `54abea15b3379ab12b2924bc7ffc89ea1102d9d4`, with eight features, 16 context rows, 16 query rows, and binary labels. Three model triplets use the larger architecture, training MLP specialists, tree specialists, and balanced mixed models for 10k steps. Each mixed-model comparison uses the other two specialist seeds as component references. Calibration and evaluation have 1,000 and 2,000 independent generator groups. Shared interpolation beats fixed pooling in one of three mixed-model seeds; the median relative KL gain is -1.4%. The exact generator call, fixed settings, and sampled-task records are supplied in `configs/richer_prior.yaml`, `experiments/richer_prior.py`, and `data/frozen/richer_prior_v2_independent/`.

The public-data probes use the eight datasets retrieved from OpenML (Bischl et al., 2021) and listed in Table 14. Source rows are split into disjoint calibration and evaluation pools; preprocessing is fitted only on the calibration pool. Each dataset supplies 80 calibration and 120 evaluation tasks, with 16 context examples, 16 queries, and at most eight features. Sampling is conditioned on both classes being present in the context. Shared interpolation is closer than fixed pooling in 11 of 24 model/dataset comparisons, with no consistent overall advantage. These comparisons reproduce mixed-model outputs using learned specialist references; they have no exact BMA reference and do not test likelihood-tempering transfer. Complete scores, preprocessing, generator settings, and uncertainty qualifications are in the supplement's `reports/06_richer_prior_validation.md`, `reports/07_real_data_external_validity.md`, and corresponding frozen result and dataset records.

Table 14: Existing public-data boundary probes. Wins count mixed-model seeds for which shared interpolation has lower held-out PFN-matching KL than fixed pooling, using leave-one-seed-out specialist references. These are descriptive counts, not independent statistical-significance tests.

Dataset	OpenML ID	Shared wins
banknote-authentication	1462	3/3
blood-transfusion-service-center	1464	2/3
churn	40701	0/3
credit-g	31	0/3
diabetes	37	2/3
ilpd	1480	2/3
qsar-biodeg	1494	2/3
seismic-bumps	45562	0/3

L Analysis Status and Reproducibility

The initial 20k family-tempering result was exploratory. The twelve-candidate comparison, long-horizon analysis, flexible-family analyses, asymmetric-prior analyses, and independent evaluation protocol were specified before their corresponding new outcomes were examined. Posterior-to-prior shrinkage was added afterward as a post hoc point-estimate robustness comparison and was not included in the locked candidate bootstrap analysis. The second-prior and context-length extensions have separate frozen protocols described in Appendices M and N. The constant and inverse-length transfer comparison in Appendix O was added after inspecting the context-length results and is post hoc; only the anchor exponent is calibrated at $n = 8$.

The Code and Data Supplement includes exact enumeration, model and candidate implementations, fixed configurations, compact per-model results, and uncertainty summaries for all analyses reported here. Tier 1 regenerates numerical tables and plot coordinates from included frozen artifacts; it does not execute PFNs or refit candidates. Tier 2 refitting requires omitted prediction archives and, where relevant, exact caches. Tier 3 training and inference require the full environment, compute, and canonical provenance resources listed in the supplement. Package checks and original experiment checks are documented separately; reproducing a summary from frozen artifacts is not a new run of the underlying experiment.

M Second Exactly Enumerable Prior

Let $\mathcal{H}_{\text{quad}}$ contain the distinct truth tables generated by

$$h(x) = \mathbf{1}\left[b + \sum_{j=1}^{4} w_j x_j + \sum_{1 \le j < k \le 4} v_{jk} x_j x_k \ge 0\right], \quad w_j, v_{jk} \in \{-1, 0, 1\}, \quad b \in \{-3, \ldots, 3\}.$$

Enumerating 413,343 parameterizations gives 6,700 distinct functions, including all 980 members of the original $\mathcal{H}_A$. Define $\mathcal{H}_Q = \mathcal{H}_{\text{quad}} \setminus \mathcal{H}_A$, with $|\mathcal{H}_Q| = 5,720$. We call $\mathcal{H}_Q$ the quadratic complement of the bounded additive family within the enumerated quadratic-threshold class; this does not assert that every member lacks any linear-threshold representation. The two families are disjoint and sampled with equal probability, then uniformly within each family. Exact inference uses prior mass $1/(2|\mathcal{H}_z|)$ per hypothesis in family z.

Seeds 10607, 10709, 10809, 10907, and 11009 train fresh models with the main three-layer architecture, Schedule-Free AdamW at learning rate 0.004, batch size 64, zero weight decay, and gradient clipping at 1.0. Each model continues from 20k to 100k with model, optimizer, and random states preserved. Each task has eight context and eight query inputs, with noise $\eta = 0.10$. Calibration uses 4,000 contexts (seed 271828) and all eight queries; evaluation uses 10,000 contexts (seed 314159) and the last five queries. The original candidate definitions, numerical bounds, and fit procedures are reused. The shared-component refactorization is omitted because that component is empty. The effective-noise parameterization duplicates likelihood tempering up to numerical precision and is not an independent conceptual competitor. Posterior-to-prior shrinkage is specified before this experiment's outcomes as a secondary candidate, unlike its post hoc status in the original study.

The source protocol and input hashes were frozen before training. An orderly system shutdown interrupted the first seed at step 250 without saved endpoints; the same seed restarted from initialization, with no scientific specification change or seed substitution. Independently sampled contexts were not forced to be disjoint: 21 evaluation contexts also occur in calibration and were retained. Two thousand paired context-level bootstrap draws per model and checkpoint independently resample calibration and evaluation contexts, refit the two tempering candidates, and retain all queries within each context. Intervals condition on the trained model, not on a population of training runs.

Fixed pooling fits more closely than BMA for only one seed at 20k and none at 100k. Family isotonic mapping outperforms likelihood tempering for two seeds at 100k. In the prespecified secondary analysis, each checkpoint uses its own inside-hull subset; family tempering is closer on that subset for four of five models at each checkpoint. This is not the original study's common-at-both-checkpoints subset. The held-out outside-hull fractions have medians 19.5% and 15.4%. BMA expected log loss is 0.639350; median PFN values are 0.640684 and 0.640121 at 20k and 100k. Fidelity comparisons do not imply better predictive risk than BMA. The different family content, hypothesis count, and overlap structure change together, so this is a cross-prior comparison rather than a causal isolation of interaction order. The accompanying numerical records retain every candidate and seed.

Table 15: Second-prior results per model. R is the relative likelihood-tempering KL reduction versus BMA; $\Delta K = K_{\text{family}} - K_{\text{likelihood}}$. The last column reports $10^5 \Delta K$ and its interval. Positive differences favor likelihood tempering.

Seed	Steps	BMA KL	a	τ	R (%)	$10^5 \Delta K$ [95% CI]
10607	20k	0.001236	0.616	0.917	14.6	10.08 [9.09, 11.10]
10709	20k	0.001249	0.642	0.925	10.8	7.22 [6.33, 8.14]
10809	20k	0.001337	0.605	0.917	12.6	8.30 [7.29, 9.36]
10907	20k	0.001431	0.602	0.911	14.2	12.08 [11.01, 13.17]
11009	20k	0.001742	0.490	0.902	14.8	11.88 [10.48, 13.33]
10607	100k	0.000771	0.765	0.951	7.8	3.36 [2.93, 3.84]
10709	100k	0.000658	0.819	0.967	4.0	1.13 [0.83, 1.43]
10809	100k	0.000672	0.821	0.963	4.6	1.56 [1.22, 1.90]
10907	100k	0.000791	0.748	0.950	7.8	3.15 [2.66, 3.62]
11009	100k	0.001000	0.688	0.939	9.7	4.85 [4.23, 5.55]

Table 16: Second-prior cross-seed median PFN-matching KL for all reported candidates. The effective-noise and likelihood rows are equivalent parameterizations.

Candidate	20k	100k
Hard MAP family selection	0.009469	0.008380
Exact BMA	0.001337	0.000771
Fixed family pooling	0.001488	0.001114
Family tempering	0.001252	0.000744
Family logistic	0.001251	0.000742
Family isotonic	0.001230	0.000731
Likelihood tempering	0.001169	0.000711
Effective noise (equivalent)	0.001169	0.000711
Output-only logistic	0.001302	0.000767
Output-only isotonic	0.001300	0.000770
Family tempering + output	0.001238	0.000744
Component-aware output	0.001286	0.000762
Posterior-to-prior shrinkage	0.001299	0.000767

N Inference-Time Context-Length Generalization

The five additive/tree PFNs (seeds 607, 709, 809, 907, 1009) are reused without training or fine-tuning. Primary checkpoints are 20k and 100k; existing 25k, 30k, 40k, 50k, 60k, and 80k checkpoints give

secondary descriptive trajectories. A task group samples one family, function, input permutation $u_1, \ldots, u_{16}$, and noisy labels. For every integer $n = 2, \ldots, 12$, the context is the prefix of length n and focal queries are $u_{13}, \ldots, u_{16}$. Shared inputs keep identical labels across lengths. All sixteen inputs remain present, with only the first n labels exposed to the network; unscored queries remain unlabeled. Implemented invariance checks remove or change non-focal query rows and test focal predictions at absolute/relative tolerances $10^{-6}/10^{-5}$; context statistics alone determine normalization. Twelve is the maximum number of distinct labeled inputs compatible with four common unseen query locations.

There are 4,000 calibration groups (seed 424242) and 10,000 independently sampled evaluation groups (seed 515151). Each supplies four scored queries at every length. The same exact prior is used at every n, but its posterior and predictive are recomputed for each context. All candidate parameters are refitted separately by model, checkpoint, and length on calibration groups only. Ten central candidates reuse their original equations and fit routines; shrinkage is a prospectively specified secondary comparison. Family isotonic mapping retains its two-stage context-weight/least-squares construction. No monotonicity in n is imposed on the fitted parameters or metrics.

Two thousand context-level bootstrap draws independently resample calibration and evaluation groups, refit all fitted candidates (including the secondary shrinkage comparison), and preserve identical group indices across lengths and the two primary checkpoints within each model. Queries from one group remain together. The resulting intervals condition on the trained PFN; the five seeds supply model-level replication. Neighboring lengths are paired measurements, not independent replications. Intervals are pointwise, not a simultaneous confidence band over the entire sweep. The frozen protocol was retained through documented changes to empty-subset reporting and row-identity export; these did not change the scientific specification.

The new task samples and four-query protocol differ from the original five-query evaluation. Their median PFN-to-BMA KL values are 0.004061 and 0.001650 at 20k and 100k, versus the original 0.004072 and 0.001641. Median τ is 0.880 and 0.957 versus 0.883 and 0.959. Likelihood tempering is closer than family tempering in four of five models at 20k and all five at 100k under both protocols. These are separate estimates; the original headline numbers are not replaced by the length-sweep estimates.

Table 17: Full context-length trajectory. Values are cross-seed medians; R is computed within each seed before taking the median. At $n = 2$ the median hides bound-hitting and heterogeneous fits.

	τ		PFN-to-BMA KL		R (%)	
n	20k	100k	20k	100k	20k	100k
2	0.966	1.184	0.016484	0.016156	0.1	0.6
3	1.262	1.443	0.016879	0.012453	2.6	5.9
4	1.309	1.431	0.012237	0.010598	8.8	18.5
5	1.211	1.353	0.007430	0.006343	14.7	33.5
6	1.105	1.213	0.004450	0.003555	7.5	33.2
7	0.984	1.072	0.003440	0.001923	0.3	10.6
8	0.880	0.957	0.004061	0.001650	21.4	6.6
9	0.800	0.864	0.006430	0.002964	49.9	47.3
10	0.729	0.785	0.010573	0.005995	67.7	71.9
11	0.669	0.720	0.016451	0.010557	77.4	82.0
12	0.618	0.665	0.023409	0.016385	82.6	86.8

At $n = 6$, all five models have lower PFN-to-BMA KL at 100k than at 20k, while the median fitted likelihood exponent moves from 1.105 to 1.213, farther from one. Likelihood tempering's median relative KL reduction versus BMA rises from 7.5% to 33.2%. The fitted description therefore captures a larger relative reduction from a smaller PFN-to-BMA KL baseline. At $n = 2$ and $n = 3$, only two of five models improve in PFN-to-BMA KL. These findings concern the same models outside their training context length, not matched-length retraining.

The inherited family and likelihood exponent bounds are $[0, 4]$. At $n = 2$, nine of ten family fits and one likelihood fit reach the upper bound. Bootstrap concentration at a bound is not an estimate of an unrestricted parameter. Context-based feature normalization also varies with context length; at short lengths some binary features can be constant in context. This experiment does not isolate preprocess-

ing from the learned inference rule or establish a mechanism for the amplification/attenuation pattern. The bound and preprocessing qualifications are particularly important for $n = 2$, rather than grounds for excluding it.

Calibration and evaluation are sampled independently, not constrained to have distinct observed contexts. The order-invariant overlap audit hashes sorted input/observed-label pairs, excluding query identities. At $n = 2$, all 10,000 evaluation contexts match a calibration context: both samples cover all 480 possible unordered two-input binary-label contexts. This does not imply that the sampled query sets or task groups coincide. The corresponding counts at every length are reported in Table 18; all sampled repeats are retained under the frozen protocol.

The tables and figures below show the supplied individual-model estimates and conditional intervals, separated by training checkpoint; the compact main figure displays medians only. Short-context output-isotonic advantages, the length-dependent sign of $\tau - 1$, and nonuniform training improvements remain visible in the complete results. Secondary analyses cover within-checkpoint hull subsets, adjacent-length common hull subsets, and query consistency. These subset comparisons are conditional diagnostics, not additional independent replications. Intermediate-checkpoint point estimates are retained in the numerical records and are not needed for the endpoint claims above. Since feature standardization and the missing-query-label placeholder are computed from the labeled context, varying n changes both the available evidence and aspects of the PFN input representation. We therefore interpret the fitted τ behaviorally rather than as an isolated estimate of evidence weighting.

Table 18: Order-invariant context overlap in the length sweep. Each length has 4,000 calibration and 10,000 evaluation draws. Matching contexts do not imply matching focal query sets.

n	Unique calibration	Unique evaluation	Evaluation matching calibration
2	480	480	10000
3	2630	3949	6119
4	3749	8368	1379
5	3930	9589	355
6	3978	9875	95
7	3997	9958	42
8	3998	9982	20
9	3999	9990	13
10	3995	9992	5
11	3996	9985	7
12	3998	9985	10

Table 19: Expected predictive risk for the length sweep. Label expectation is analytic under the corresponding exact BMA; contexts and queries are averaged over the sampled evaluation groups. PFN entries are cross-seed medians.

	Expected log loss			Expected Brier score		
n	BMA	PFN 20k	PFN 100k	BMA	PFN 20k	PFN 100k
2	0.677057	0.693797	0.693888	0.242072	0.250006	0.249775
3	0.668665	0.685989	0.681431	0.238024	0.245899	0.243870
4	0.660279	0.673005	0.671229	0.234029	0.239593	0.238760
5	0.651365	0.659036	0.657999	0.229835	0.233160	0.232595
6	0.641950	0.646522	0.645609	0.225450	0.227430	0.226984
7	0.632088	0.635573	0.634047	0.220911	0.222471	0.221758
8	0.620759	0.624797	0.622411	0.215731	0.217556	0.216475
9	0.608826	0.615096	0.611734	0.210310	0.213101	0.211594
10	0.595593	0.605698	0.601363	0.204357	0.208797	0.206834
11	0.581957	0.597354	0.591901	0.198288	0.204944	0.202457
12	0.568481	0.589895	0.583601	0.192378	0.201487	0.198600

Table 20: Per-length model counts, not independent length replications. Columns 2 and 3 compare 100k with 20k. L/F: likelihood closer than family tempering. O/L: output isotonic closer than likelihood tempering.

	Training change		L/F		O/L	
n	BMA KL: 100k $<$ 20k	τ closer to 1	20k	100k	20k	100k
2	2/5	2/5	2/5	2/5	5/5	5/5
3	2/5	1/5	5/5	5/5	5/5	5/5
4	4/5	0/5	5/5	5/5	5/5	5/5
5	4/5	0/5	5/5	5/5	5/5	2/5
6	5/5	0/5	5/5	5/5	5/5	3/5
7	5/5	0/5	0/5	5/5	4/5	5/5
8	5/5	5/5	4/5	5/5	0/5	0/5
9	5/5	5/5	5/5	5/5	0/5	0/5
10	5/5	5/5	5/5	5/5	0/5	0/5
11	5/5	5/5	5/5	5/5	0/5	0/5
12	5/5	5/5	5/5	5/5	0/5	0/5

Table 21: Context-length candidate comparison at 20k: cross-seed median PFN-matching KL. Complete per-seed results for all eleven candidates accompany the numerical records.

n	BMA	MAP	Fixed	Family	Likelihood	Output isotonic
2	0.016484	0.017048	0.015819	0.016224	0.015271	0.013006
3	0.016879	0.021959	0.014454	0.016877	0.016515	0.014580
4	0.012237	0.018803	0.010923	0.012228	0.010838	0.010128
5	0.007430	0.015536	0.007192	0.007424	0.006294	0.005701
6	0.004450	0.013866	0.004632	0.004424	0.004116	0.003791
7	0.003440	0.015534	0.003453	0.003113	0.003421	0.003390
8	0.004061	0.019736	0.003399	0.003184	0.003223	0.003787
9	0.006430	0.026193	0.004516	0.004406	0.003281	0.004734
10	0.010573	0.034359	0.006431	0.006384	0.003436	0.006117
11	0.016451	0.043871	0.008872	0.008862	0.003678	0.007652
12	0.023409	0.054060	0.011677	0.011702	0.003960	0.008977

Table 22: Context-length candidate comparison at 100k: cross-seed median PFN-matching KL. Complete per-seed results for all eleven candidates accompany the numerical records.

n	BMA	MAP	Fixed	Family	Likelihood	Output isotonic
2	0.016156	0.017267	0.013799	0.016074	0.015798	0.013297
3	0.012453	0.017476	0.011488	0.012454	0.012207	0.011220
4	0.010598	0.016513	0.009873	0.010581	0.007929	0.007557
5	0.006343	0.012926	0.006899	0.006279	0.004096	0.004152
6	0.003555	0.011157	0.004790	0.003429	0.002387	0.002423
7	0.001923	0.011869	0.003378	0.001900	0.001703	0.001644
8	0.001650	0.014814	0.002931	0.001587	0.001555	0.001648
9	0.002964	0.019852	0.003434	0.002316	0.001576	0.002285
10	0.005995	0.026647	0.004777	0.003872	0.001685	0.003288
11	0.010557	0.034724	0.006809	0.006114	0.001897	0.004392
12	0.016385	0.043802	0.009353	0.008816	0.002152	0.005520

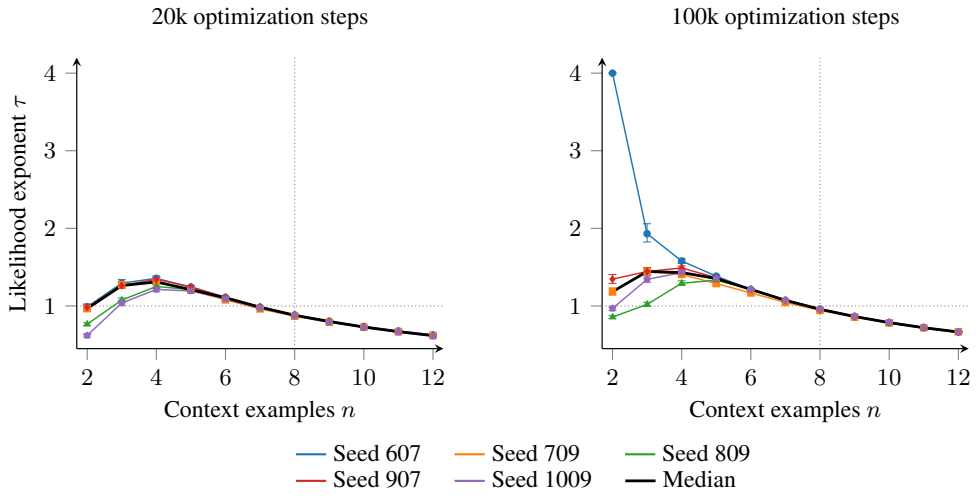


Figure 9: Per-model likelihood exponents and conditional intervals across context length. The full vertical range includes the bound-hitting two-example fit.

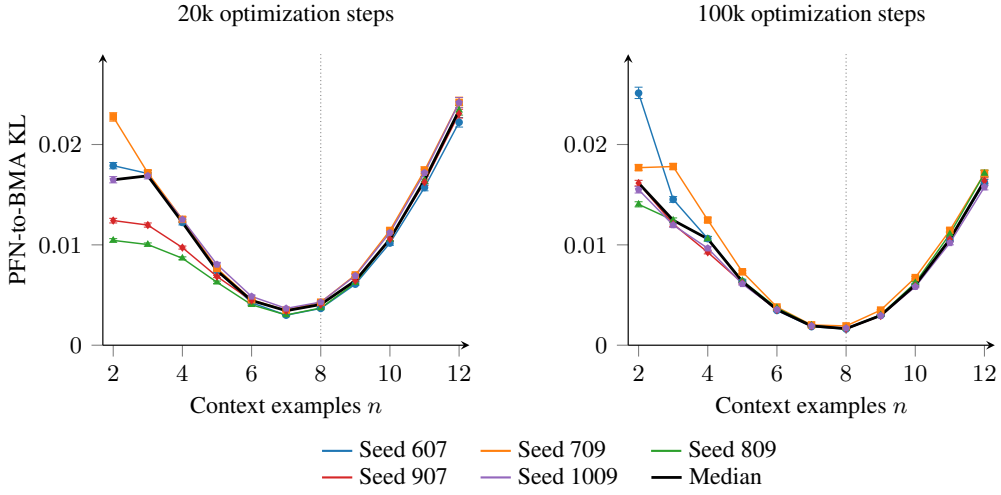


Figure 10: Per-model PFN-to-BMA KL across context length. More training improves every model for $n = 6$ through 12, but not at every shorter length.

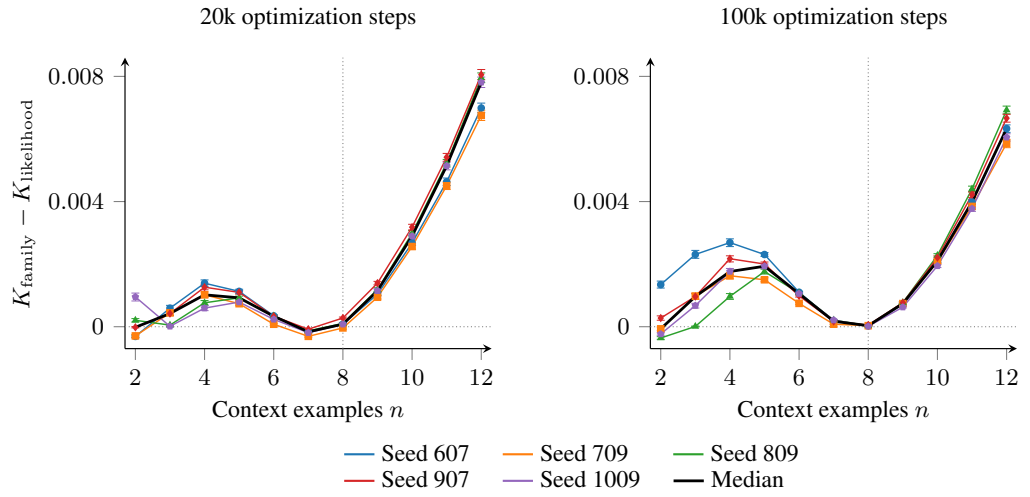


Figure 11: Family-minus-likelihood KL across context length. Positive values favor likelihood tempering.

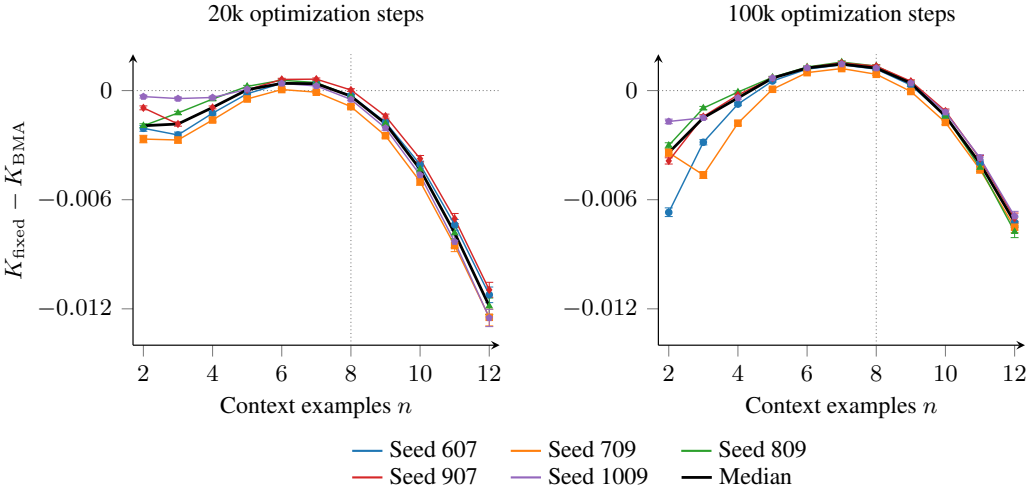


Figure 12: Fixed-pooling-minus-BMA KL across context length. Positive values favor exact BMA.

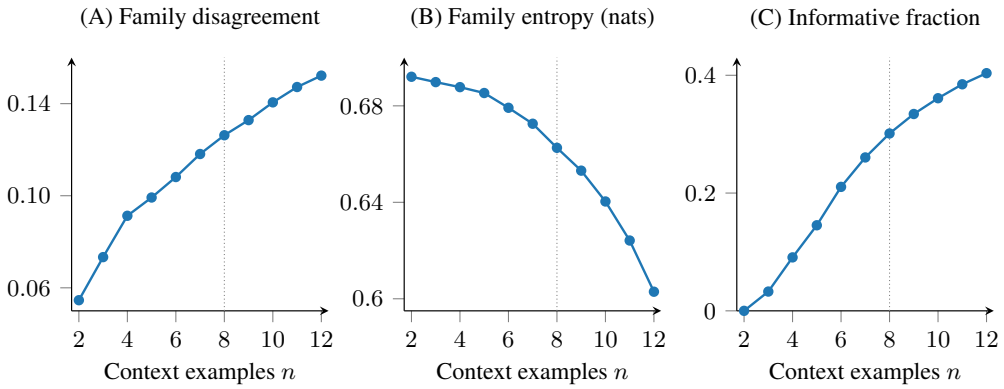


Figure 13: Evidence diagnostics across context length. Panel A shows median $|p_T - p_A|$; panel B shows median posterior family entropy in nats; panel C shows the fraction of focal queries with $|p_T - p_A| \geq 0.20$. The vertical marker is the training length $n = 8$.

O Post Hoc Constant and Inverse-Length Exponent Transfer

This comparison tests two ways to transfer a likelihood exponent calibrated at the eight-example training context. For each of the five original PFNs and each checkpoint, we fit $\widehat{\tau}(8)$ using only $n = 8$ calibration task groups, then evaluate

$$\tau_{\text{constant}}(n) = \widehat{\tau}(8), \tag{13}$$

$$\tau_{\text{scaled}}(n) = \widehat{\tau}(8)\frac{8}{n}, \qquad n \in \{9, 10, 11, 12\}. \tag{14}$$

Neither rule is fitted at its target length. Both use likelihoods recomputed from the corresponding target-length context and retain the original query-label noise rate. Comparators are exact BMA, likelihood tempering calibrated separately at each target length, and family tempering calibrated at that length. All use the same four held-out queries and saved PFN probabilities. No network is retrained or corrected.

The inverse-length rule was proposed after inspecting the context-length results. It is therefore a post hoc behavioral comparison, not an independently prespecified discovery. The unchanged-transfer result is retained as a baseline, using the same fitted scalar and no target-length parameters. This comparison tests the value of the length adjustment, rather than adding flexibility through additional calibration parameters.

Let K_j denote $D_{\text{KL}}(\text{PFN} \| j)$, averaged over the held-out task groups and their four queries. We report the fraction of the length-specific likelihood candidate's improvement recovered by each transfer rule $j \in \{\text{constant}, \text{scaled}\}$:

$$R_j = \frac{K_{\text{BMA}} - K_j}{K_{\text{BMA}} - K_{\text{fitted}}}. \tag{15}$$

It is computed separately for each model before taking a cross-seed median. It is not explained variance, is not clipped, and does not measure the fractional reduction in PFN-to-BMA KL. A value near one means recovery of nearly all the improvement obtained by the separately fitted likelihood candidate. Probability MAE and logit RMSE are retained as sensitivity metrics.

Each of 2,000 bootstrap draws independently resamples calibration and evaluation task groups and keeps the four queries within each group together. The anchor $\widehat{\tau}(8)$ is refitted first; both transfer rules are then constructed from that draw's anchor. Target-length likelihood and family parameters are refitted separately. The same resampled indices are used across lengths, checkpoints, and models, preserving dependence when taking medians and differences. Percentile intervals measure task-sampling uncertainty conditional on these five networks, not variation over new training runs. Every recorded improvement denominator is positive.

Scaled transfer fits more closely than constant transfer, BMA, and family tempering in all 40 model/checkpoint/length conditions under KL, MAE, and logit RMSE. Separately fitted likelihood

tempering remains more faithful in all 40 conditions. Every within-model 95% KL interval for scaled-minus-constant and scaled-minus-BMA is negative; every interval for scaled-minus-fitted-likelihood is positive. These are repeated comparisons on five models, not 40 independent training replications.

Across the eight checkpoint/length cells, median recovery under scaled transfer ranges from 98.8% to 99.2%. At 100k and $n = 12$, it is 99.0% [98.8%, 99.1%], compared with 20.2% [18.7%, 21.2%] for unchanged transfer. At that endpoint, the median relative KL reduction versus BMA is 85.9% for scaled transfer and 86.8% for separate fitting. Thus, nearly all the fitted candidate's gain is recovered, but separate fitting still yields a small, statistically detectable improvement.

The rule rescales the summed log likelihood as

$$\frac{8\widehat{\tau}(8)}{n}\sum_{i=1}^{n}\log p(y_i \mid x_i, h) = 8\widehat{\tau}(8)\left[\frac{1}{n}\sum_{i=1}^{n}\log p(y_i \mid x_i, h)\right].$$

This is an algebraic property of the candidate. It does not show that the network computes an average log likelihood, remembers eight examples, or implements a specific normalization mechanism. The reported comparison is restricted to the original $n = 8$-trained models and $n = 9, \ldots, 12$.

Table 23: Post hoc transfer: cross-seed median PFN-matching KL. The scaled and constant candidates use the same exponent fitted at $n = 8$; the last column instead calibrates the exponent separately at each target length.

Steps	n	BMA	Constant	$8/n$ scaled	Separately fitted
20k	9	0.006430	0.003824	0.003305	0.003281
20k	10	0.010573	0.005822	0.003512	0.003436
20k	11	0.016451	0.009458	0.003836	0.003678
20k	12	0.023409	0.014291	0.004228	0.003960
100k	9	0.002964	0.002321	0.001580	0.001576
100k	10	0.005995	0.004606	0.001728	0.001685
100k	11	0.010557	0.008354	0.001982	0.001897
100k	12	0.016385	0.013383	0.002281	0.002152

Table 24: Median recovery percentages with 95% paired task-group-bootstrap intervals conditional on the five models. These are medians of individual ratios, not ratios of median losses.

Steps	n	Constant recovery [95% CI]	Scaled recovery [95% CI]
20k	9	82.8 [81.4, 84.5]	99.2 [98.8, 99.5]
20k	10	66.6 [65.3, 68.3]	98.9 [98.6, 99.2]
20k	11	55.4 [54.2, 57.0]	98.8 [98.6, 99.1]
20k	12	47.0 [45.9, 48.4]	98.9 [98.7, 99.1]
100k	9	51.2 [48.0, 53.4]	99.1 [98.7, 99.5]
100k	10	33.9 [31.6, 35.4]	99.0 [98.6, 99.2]
100k	11	25.4 [23.6, 26.6]	98.9 [98.7, 99.1]
100k	12	20.2 [18.7, 21.2]	99.0 [98.8, 99.1]

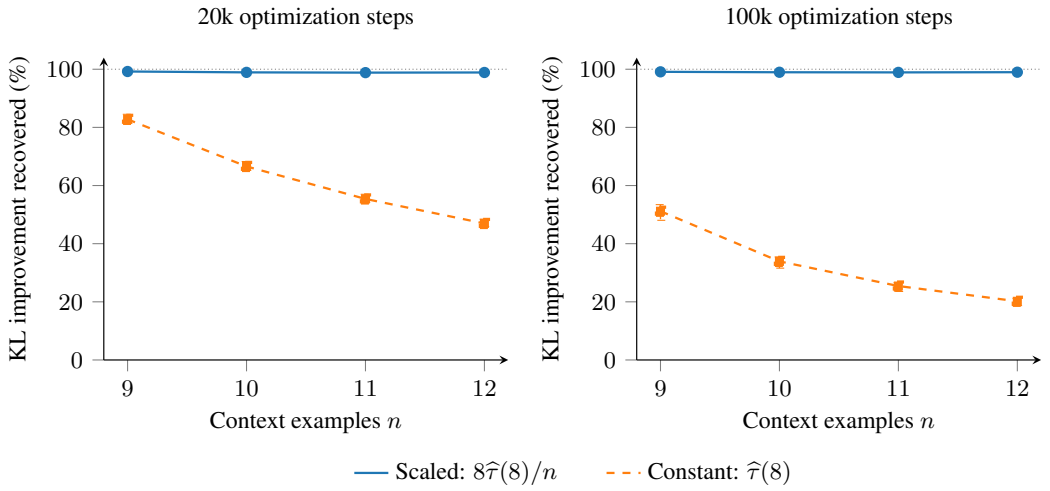


Figure 14: Recovery of the length-specific likelihood candidate's improvement over BMA under constant and inverse-length transfer. Points are medians of model-specific ratios; bars are 95% conditional paired task-group-bootstrap intervals. The dotted reference is 100% recovery, not a perfect match to PFN predictions.

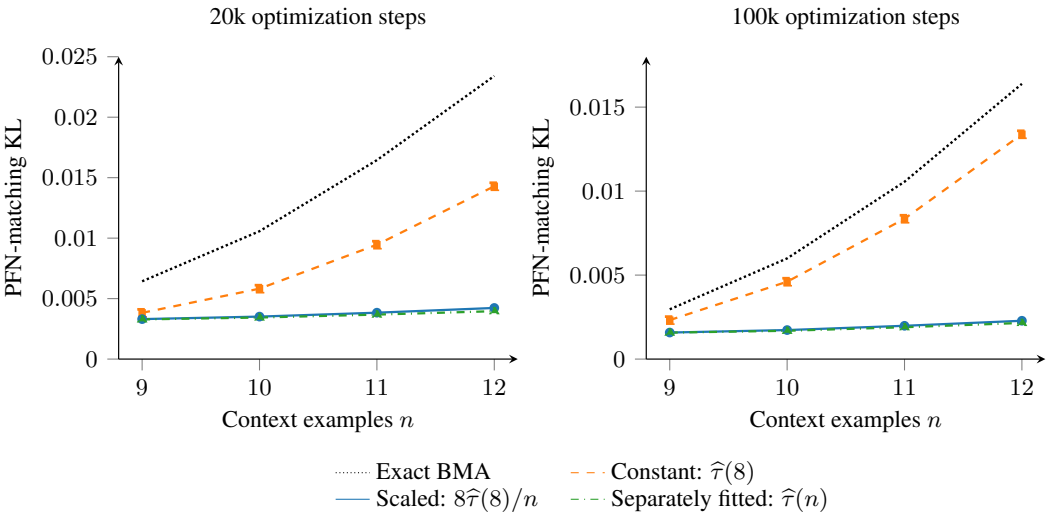


Figure 15: Cross-seed median PFN-matching KL under BMA, constant transfer, inverse-length transfer, and separate likelihood fitting. This is a comparison of fidelity to PFN probabilities, not label-prediction risk.

Table 25: Transfer results for each model at 20k steps. KL values use 10^{-3} units. R_c and R_s are constant and scaled recovery percentages relative to the separately fitted likelihood candidate.

Seed	n	BMA	Constant	Scaled	Fitted	R_c (%)	R_s (%)
607	9	6.0815	3.5739	3.0805	3.0479	82.66	98.92
607	10	10.1786	5.6046	3.3877	3.2831	66.33	98.48
607	11	15.7147	9.0382	3.8372	3.6252	55.23	98.25
607	12	22.2139	13.6608	4.2865	3.9598	46.86	98.21
709	9	6.9986	3.9758	3.4971	3.4638	85.52	99.06
709	10	11.4324	6.0535	3.9324	3.8205	70.66	98.53
709	11	17.4096	9.6150	4.6087	4.3827	59.83	98.27
709	12	24.1955	14.2909	5.3593	4.9878	51.57	98.07
809	9	6.2203	3.5945	2.9853	2.9720	80.84	99.59
809	10	10.5700	5.7791	3.1732	3.1262	64.36	99.37
809	11	16.4515	9.4582	3.4449	3.3462	53.36	99.25
809	12	23.4093	14.4553	3.7555	3.6013	45.20	99.22
907	9	6.4305	3.8237	3.3055	3.2810	82.77	99.22
907	10	10.5733	5.8219	3.5125	3.4356	66.57	98.92
907	11	16.2481	9.2853	3.8249	3.6783	55.39	98.83
907	12	23.0767	14.1257	4.2279	4.0156	46.96	98.89
1009	9	6.9195	4.0677	3.5402	3.5170	83.81	99.32
1009	10	11.2149	6.0382	3.6536	3.5859	67.86	99.11
1009	11	17.1811	9.5700	3.8360	3.7193	56.54	99.13
1009	12	24.1757	14.4073	4.0656	3.8925	48.16	99.15

Table 26: Transfer results for each model at 100k steps. KL values use 10^{-3} units. R_c and R_s are constant and scaled recovery percentages relative to the separately fitted likelihood candidate.

Seed	n	BMA	Constant	Scaled	Fitted	R_c (%)	R_s (%)
607	9	2.9638	2.2332	1.5455	1.5358	51.17	99.33
607	10	5.9947	4.5332	1.7279	1.6849	33.91	99.00
607	11	10.5083	8.3074	1.9817	1.8882	25.53	98.92
607	12	16.2119	13.3517	2.2814	2.1383	20.32	98.98
709	9	3.4965	2.5022	1.9019	1.8822	61.59	98.78
709	10	6.7326	4.7977	2.3462	2.2603	43.26	98.08
709	11	11.4452	8.5457	2.9417	2.7609	33.39	97.92
709	12	17.1138	13.3829	3.6506	3.3489	27.10	97.81
809	9	3.0233	2.3206	1.5802	1.5758	48.55	99.70
809	10	6.2167	4.7872	1.6952	1.6777	31.49	99.61
809	11	11.0799	8.9101	1.8629	1.8297	23.46	99.64
809	12	17.1121	14.2989	2.0469	1.9966	18.61	99.67
907	9	2.9214	2.2062	1.5510	1.5387	51.73	99.11
907	10	5.9034	4.4582	1.7047	1.6607	34.06	98.96
907	11	10.5567	8.3541	1.9761	1.8965	25.43	99.08
907	12	16.3855	13.5121	2.2659	2.1520	20.19	99.20
1009	9	2.9491	2.3514	1.7052	1.6902	47.48	98.81
1009	10	5.8232	4.6058	1.9012	1.8464	30.61	98.62
1009	11	10.1925	8.3409	2.1505	2.0432	22.72	98.68
1009	12	15.7635	13.3437	2.4260	2.2693	17.93	98.84

Table 27: Scaled-transfer KL contrasts at 20k, in 10^{-4} units. $\Delta_B = K_s - K_{\mathrm{BMA}}$, $\Delta_C = K_s - K_c$, and $\Delta_F = K_s - K_{\mathrm{fitted}}$. Brackets give 95% conditional paired task-group-bootstrap intervals.

Seed	n	Δ_B [95% CI]	Δ_C [95% CI]	Δ_F [95% CI]
607	9	-30.01 [-31.03, -28.96]	-4.93 [-5.65, -4.26]	0.33 [0.21, 0.44]
607	10	-67.91 [-69.94, -65.88]	-22.17 [-23.80, -20.50]	1.05 [0.82, 1.28]
607	11	-118.78 [-122.15, -115.48]	-52.01 [-54.86, -49.29]	2.12 [1.79, 2.47]
607	12	-179.27 [-184.19, -174.52]	-93.74 [-97.90, -89.62]	3.27 [2.84, 3.72]
709	9	-35.01 [-36.22, -33.80]	-4.79 [-5.53, -4.07]	0.33 [0.20, 0.47]
709	10	-75.00 [-77.23, -72.76]	-21.21 [-22.86, -19.57]	1.12 [0.87, 1.37]
709	11	-128.01 [-131.53, -124.47]	-50.06 [-53.01, -47.32]	2.26 [1.89, 2.62]
709	12	-188.36 [-193.56, -183.21]	-89.32 [-93.65, -85.18]	3.72 [3.23, 4.18]
809	9	-32.35 [-33.37, -31.32]	-6.09 [-6.82, -5.46]	0.13 [0.04, 0.22]
809	10	-73.97 [-75.97, -71.98]	-26.06 [-27.73, -24.59]	0.47 [0.30, 0.62]
809	11	-130.07 [-133.33, -126.70]	-60.13 [-63.00, -57.53]	0.99 [0.75, 1.21]
809	12	-196.54 [-201.45, -191.70]	-107.00 [-111.19, -103.08]	1.54 [1.25, 1.82]
907	9	-31.25 [-32.29, -30.17]	-5.18 [-5.94, -4.47]	0.25 [0.12, 0.38]
907	10	-70.61 [-72.58, -68.66]	-23.09 [-24.75, -21.48]	0.77 [0.57, 0.98]
907	11	-124.23 [-127.46, -121.13]	-54.60 [-57.42, -51.84]	1.47 [1.18, 1.78]
907	12	-188.49 [-193.10, -183.90]	-98.98 [-103.18, -94.87]	2.12 [1.76, 2.51]
1009	9	-33.79 [-34.96, -32.55]	-5.27 [-6.07, -4.46]	0.23 [0.13, 0.35]
1009	10	-75.61 [-77.83, -73.36]	-23.85 [-25.67, -22.02]	0.68 [0.47, 0.89]
1009	11	-133.45 [-137.00, -129.96]	-57.34 [-60.36, -54.22]	1.17 [0.89, 1.44]
1009	12	-201.10 [-206.27, -195.92]	-103.42 [-107.88, -98.92]	1.73 [1.39, 2.08]

Table 28: Scaled-transfer KL contrasts at 100k, in 10^{-4} units. $\Delta_B = K_s - K_{\mathrm{BMA}}$, $\Delta_C = K_s - K_c$, and $\Delta_F = K_s - K_{\mathrm{fitted}}$. Brackets give 95% conditional paired task-group-bootstrap intervals.

Seed	n	Δ_B [95% CI]	Δ_C [95% CI]	Δ_F [95% CI]
607	9	-14.18 [-14.65, -13.72]	-6.88 [-7.43, -6.33]	0.10 [0.04, 0.16]
607	10	-42.67 [-43.75, -41.59]	-28.05 [-29.37, -26.73]	0.43 [0.31, 0.55]
607	11	-85.27 [-87.26, -83.31]	-63.26 [-65.56, -61.02]	0.93 [0.76, 1.12]
607	12	-139.30 [-142.53, -136.23]	-110.70 [-114.14, -107.31]	1.43 [1.21, 1.66]
709	9	-15.95 [-16.49, -15.41]	-6.00 [-6.58, -5.44]	0.20 [0.11, 0.29]
709	10	-43.86 [-45.09, -42.68]	-24.52 [-25.90, -23.08]	0.86 [0.69, 1.03]
709	11	-85.04 [-87.31, -82.87]	-56.04 [-58.46, -53.70]	1.81 [1.55, 2.06]
709	12	-134.63 [-137.91, -131.37]	-97.32 [-100.76, -93.87]	3.02 [2.66, 3.38]
809	9	-14.43 [-14.88, -13.97]	-7.40 [-7.94, -6.86]	0.04 [0.00, 0.09]
809	10	-45.21 [-46.30, -44.08]	-30.92 [-32.22, -29.61]	0.18 [0.10, 0.25]
809	11	-92.17 [-94.25, -90.09]	-70.47 [-72.75, -68.18]	0.33 [0.23, 0.45]
809	12	-150.65 [-154.04, -147.36]	-122.52 [-126.08, -118.90]	0.50 [0.38, 0.65]
907	9	-13.70 [-14.17, -13.25]	-6.55 [-7.09, -6.00]	0.12 [0.06, 0.19]
907	10	-41.99 [-43.04, -40.98]	-27.54 [-28.80, -26.28]	0.44 [0.33, 0.56]
907	11	-85.81 [-87.69, -83.92]	-63.78 [-66.00, -61.70]	0.80 [0.64, 0.96]
907	12	-141.20 [-144.09, -138.06]	-112.46 [-115.74, -109.16]	1.14 [0.94, 1.34]
1009	9	-12.44 [-12.91, -12.00]	-6.46 [-7.06, -5.90]	0.15 [0.08, 0.22]
1009	10	-39.22 [-40.25, -38.20]	-27.05 [-28.48, -25.70]	0.55 [0.41, 0.68]
1009	11	-80.42 [-82.31, -78.58]	-61.90 [-64.22, -59.62]	1.07 [0.89, 1.28]
1009	12	-133.37 [-136.28, -130.53]	-109.18 [-112.51, -105.86]	1.57 [1.33, 1.82]

Table 29: Constant transfer at 20k: per-model paired KL differences. $\Delta_B = K_{\text{constant}} - K_{\text{BMA}}$ and $\Delta_F = K_{\text{constant}} - K_{\text{fitted}}$. Brackets give 95% conditional task-group-bootstrap intervals.

Seed	n	Δ_B [95% CI]	Δ_F [95% CI]
607	9	-0.002508 [-0.002587, -0.002430]	0.000526 [0.000470, 0.000587]
607	10	-0.004574 [-0.004730, -0.004423]	0.002321 [0.002177, 0.002464]
607	11	-0.006676 [-0.006899, -0.006447]	0.005413 [0.005172, 0.005667]
607	12	-0.008553 [-0.008839, -0.008259]	0.009701 [0.009322, 0.010077]
709	9	-0.003023 [-0.003116, -0.002925]	0.000512 [0.000454, 0.000573]
709	10	-0.005379 [-0.005548, -0.005214]	0.002233 [0.002092, 0.002375]
709	11	-0.007795 [-0.008056, -0.007545]	0.005232 [0.004988, 0.005492]
709	12	-0.009905 [-0.010254, -0.009580]	0.009303 [0.008928, 0.009698]
809	9	-0.002626 [-0.002709, -0.002539]	0.000623 [0.000568, 0.000686]
809	10	-0.004791 [-0.004951, -0.004634]	0.002653 [0.002519, 0.002806]
809	11	-0.006993 [-0.007233, -0.006762]	0.006112 [0.005869, 0.006378]
809	12	-0.008954 [-0.009267, -0.008649]	0.010854 [0.010480, 0.011251]
907	9	-0.002607 [-0.002690, -0.002519]	0.000543 [0.000485, 0.000606]
907	10	-0.004751 [-0.004905, -0.004596]	0.002386 [0.002244, 0.002532]
907	11	-0.006963 [-0.007205, -0.006727]	0.005607 [0.005358, 0.005866]
907	12	-0.008951 [-0.009264, -0.008624]	0.010110 [0.009731, 0.010499]
1009	9	-0.002852 [-0.002943, -0.002754]	0.000551 [0.000480, 0.000620]
1009	10	-0.005177 [-0.005343, -0.004997]	0.002452 [0.002291, 0.002615]
1009	11	-0.007611 [-0.007869, -0.007337]	0.005851 [0.005565, 0.006127]
1009	12	-0.009768 [-0.010113, -0.009410]	0.010515 [0.010094, 0.010928]

Table 30: Constant transfer at 100k: per-model paired KL differences. $\Delta_B = K_{\text{constant}} - K_{\text{BMA}}$ and $\Delta_F = K_{\text{constant}} - K_{\text{fitted}}$. Brackets give 95% conditional task-group-bootstrap intervals.

Seed	n	Δ_B [95% CI]	Δ_F [95% CI]
607	9	-0.000731 [-0.000775, -0.000684]	0.000697 [0.000649, 0.000747]
607	10	-0.001462 [-0.001554, -0.001363]	0.002848 [0.002725, 0.002969]
607	11	-0.002201 [-0.002343, -0.002048]	0.006419 [0.006209, 0.006636]
607	12	-0.002860 [-0.003053, -0.002659]	0.011213 [0.010883, 0.011542]
709	9	-0.000994 [-0.001040, -0.000944]	0.000620 [0.000572, 0.000669]
709	10	-0.001935 [-0.002028, -0.001832]	0.002537 [0.002412, 0.002661]
709	11	-0.002899 [-0.003045, -0.002740]	0.005785 [0.005571, 0.006005]
709	12	-0.003731 [-0.003926, -0.003526]	0.010034 [0.009715, 0.010354]
809	9	-0.000703 [-0.000749, -0.000656]	0.000745 [0.000694, 0.000795]
809	10	-0.001430 [-0.001531, -0.001330]	0.003110 [0.002986, 0.003232]
809	11	-0.002170 [-0.002328, -0.002017]	0.007080 [0.006859, 0.007299]
809	12	-0.002813 [-0.003016, -0.002608]	0.012302 [0.011955, 0.012650]
907	9	-0.000715 [-0.000759, -0.000671]	0.000667 [0.000620, 0.000716]
907	10	-0.001445 [-0.001537, -0.001348]	0.002798 [0.002683, 0.002914]
907	11	-0.002203 [-0.002352, -0.002052]	0.006458 [0.006260, 0.006666]
907	12	-0.002873 [-0.003074, -0.002668]	0.011360 [0.011046, 0.011674]
1009	9	-0.000598 [-0.000641, -0.000554]	0.000661 [0.000612, 0.000714]
1009	10	-0.001217 [-0.001314, -0.001121]	0.002759 [0.002636, 0.002889]
1009	11	-0.001852 [-0.002000, -0.001700]	0.006298 [0.006090, 0.006511]
1009	12	-0.002420 [-0.002620, -0.002221]	0.011074 [0.010756, 0.011393]

The transfer comparison is restricted to $n = 9, \ldots, 12$. The independently fitted shorter-context coefficients do not follow the same numerical pattern: for example, the 100k median anchor $\widehat{\tau}(8) \approx 0.957$ would give $8\widehat{\tau}(8)/4 \approx 1.91$, whereas the median independently fitted coefficient at $n = 4$ is 1.431. This parameter comparison does not evaluate the transferred candidate's shorter-context predictive fit, but it rules out presenting the reported coefficient pattern as uniform over the full sweep.