PRIOR-AMORTIZED IN-CONTEXT BAYESIAN INFERENCE FOR GENERALIZED LINEAR MIXED-EFFECTS MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Hierarchical data is ubiquitous in the empirical sciences and most commonly analyzed with generalized linear mixed-effects models (GLMMs), a flexible hierarchical model class with interpretable parameters. Bayesian inference for GLMMs yields calibrated uncertainty estimates but requires Markov Chain Monte Carlo (MCMC); the gold-standard No-U-Turn Sampler (NUTS) is slow and must restart from scratch for every new dataset, model and prior specification. We introduce `metabeta`, a pretrained neural network for prior-amortized in-context Bayesian inference over GLMMs. Unlike previous neural posterior estimators that fix the prior at training time, `metabeta` accepts prior families and hyperparameters as direct inputs at test time, enabling zero-shot generalization to new prior specifications. Two set transformers and two conditional neural spline flows mirror the posterior's two-level structure (global parameters shared across groups, local parameters per group) and are trained jointly on millions of realistic simulated datasets with continuous, binary and count outcomes. By default, the flow posterior is refined by Independence Metropolis–Hastings against the unnormalized posterior; this yields tuning-free asymptotically correct inference two to three orders of magnitude faster than NUTS. Alternatively, `metabeta` warm-starts NUTS, yielding nearly identical inference 2–12$\times$ faster while removing over 98% of its divergent transitions. On controlled benchmarks with ground-truth parameters, `metabeta` matches NUTS in parameter recovery, posterior calibration and out-of-sample prediction; on real datasets with unknown parameters, priors and model structure, its posteriors closely agree with those of NUTS; and it remains faithful where amortized models typically struggle: under misspecified likelihoods, misspecified priors, out-of-distribution predictors, collinear designs and data-poor regimes. On real data, a full prior sensitivity analysis including model comparison takes a few seconds with `metabeta` against 7–17 hours for NUTS, tracing its posteriors throughout. Our model is open-source and open-weights and thus immediately deployable.

1 INTRODUCTION

Hierarchical structure is a defining feature of many real-world tabular datasets, from longitudinal clinical trials with site-specific effects to ecological studies where the same intervention performs differently across locations. Generalized linear mixed-effects models (GLMMs) provide the standard framework for interpretable inference on such multi-level data, and are widely adopted across the empirical sciences (Gelman & Hill, 2007; Harrison et al., 2018; Gordon, 2019; Yu et al., 2022). Popular libraries such as `lme4`, `nlme`, and `brms` (Bates et al., 2015; Pinheiro et al., 1999; Bürkner, 2018) are downloaded nearly 10 million times per year[1], and GLMM analyses appear in about half of recent quantitative empirical papers (survey over 1,133 papers in Appendix F.1).

In many such applications, Bayesian inference would be the method of choice: it yields full posterior uncertainty over interpretable parameters and enables domain knowledge to enter inference through priors (Figueroa-Zúñiga et al., 2013; Gelman et al., 2013). However, the GLMM posterior over fixed effects, per-group random effects and their covariance is intractable, grows in dimension with

[1]CRAN download counts for `nlme`, `lme4` and `brms`; retrieved 2026-09-25 via `cranlogs.r-pkg.org`.

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.

the number of groups and resists simple Gaussian approximations. Markov Chain Monte Carlo (MCMC; Metropolis et al., 1953; Hastings, 1970; Gelfand & Smith, 1990) is the standard numerical approximation, with the No-U-Turn Sampler (NUTS; Hoffman & Gelman, 2014; Bürkner, 2018; Capretto et al., 2022) being the current gold standard for GLMMs. However, NUTS is poorly suited to workflows with repeated inference across datasets, model and prior specifications, for three compounding reasons. (1) It provides no amortization: every dataset, every prior change, and every model variant requires restarting the sampler from scratch (Papaspiliopoulos et al., 2007; Betancourt & Girolami, 2013; Gelman et al., 2026), making prior sensitivity analyses or Bayesian meta-analyses across many studies prohibitively slow (Blei et al., 2017; Zhang et al., 2022). (2) Wall time is governed by posterior geometry and cannot be bounded in advance (Betancourt & Girolami, 2013; Hoffman et al., 2021; Hoffman & Sountsov, 2022), complicating integration into latency-sensitive workflows. (3) Correct use demands monitoring of chain-mixing diagnostics (cross-chain variance ratios, effective sample sizes, divergent transitions), which require additional statistical expertise (Roy, 2020; Vehtari et al., 2021).

In this work, we introduce `metabeta`, a pretrained model that bridges neural posterior estimation (NPE; Cranmer et al., 2020) and MCMC, enabling asymptotically correct prior-amortized in-context Bayesian inference over GLMMs. Trained once on millions of realistic hierarchical datasets, it permits batched full posterior inference over all GLMM parameters for any new dataset and prior specification at the speed of a forward pass. Our key contributions are:

1. **Prior-amortized inference over a model class**: the prior family and its hyperparameters, together with the model structure (number of fixed effects, random effects, groups and observations per group), are inputs at inference time. One trained network therefore serves every GLMM of a likelihood family within a documented envelope, enabling zero-shot prior sensitivity analyses and inference over many datasets in a single batched pass. Prior-amortization itself is validated: a network trained at one fixed prior misstates posterior width by 11–35% whenever the analyst's prior differs (Section 3.5).
2. **Exactness of MCMC**: the flow's tractable density serves as a proposal rather than the final answer. By default, `metabeta` refines it by Independence Metropolis–Hastings (IMH; Tierney, 1994) against the unnormalized posterior. IMH acceptance rate and importance weight diagnostics flag unreliable posteriors per dataset, and the same weights yield an unbiased marginal-likelihood estimate for Bayes factors. Alternatively, the flow warm-starts NUTS, cutting its wall time 2–12× and removing over 98% of its divergent transitions.
3. **A realistic tabular data prior for hierarchical regression**: a simulator that samples model structures and prior specifications from survey-grounded ranges and predictors from structural causal models and real tabular data, released together with the source code.
4. **A verified operating envelope**: oracle, real-data and stress benchmarks (misspecified likelihoods and priors, out-of-distribution predictors, collinear designs, data-poor groups) against NUTS and existing faster approximations, showing NUTS-level posteriors at the speed of Laplace approximation.

All code is implemented in `PyTorch` (Paszke et al., 2019) and open source with open weights at `anonymous.4open.science/r/metabeta`.

1.1 Related Work

Amortized simulation-based inference. Neural Posterior Estimation (NPE) approximates Bayesian posteriors with a neural network (Papamakarios & Murray, 2016; Lueckmann et al., 2017; Greenberg et al., 2019; Cranmer et al., 2020). BayesFlow (Radev et al., 2020; 2023) implements NPE with summary encoders (Zaheer et al., 2017; Lee et al., 2019) and normalizing flows (Rezende & Mohamed, 2015; Papamakarios et al., 2021), a design `metabeta` adopts and extends. Following Rodrigues et al. (2021), BayesFlow was adapted to hierarchical models (Habermann et al., 2025) and pharmacology models (Arruda et al., 2024). Both fix the prior at training time and practitioners must train the model on their own; `metabeta` instead takes the full prior specification as in-context inputs, is highly optimized for GLMMs and released as a pretrained model.

Transformer-based amortized inference. TabPFN (Müller et al., 2021; Hollmann et al., 2025) uses transformers (Vaswani et al., 2017) for tabular predictive distributions rather than posteriors over

interpretable parameters. Methods that do target parameter posteriors still fix the prior at training time: the Simformer (Gloeckler et al., 2024) amortizes only over data for a single simulator, and in-context estimators retrain per prior (Mittal et al., 2025; Reuter et al., 2025). CogFormer (Huang et al., 2026) amortizes across families and design configurations of sequential-sampling models, but is limited to fixed priors and non-hierarchical models. Distribution Transformers (Whittle et al., 2025) do amortize over the prior, but use a Gaussian mixture over a fixed, low-dimensional parameter vector. None express the hierarchical structure of GLMMs: nested exchangeability, structured random effects, and likelihood heterogeneity across varying numbers of groups and predictors.

Flows as MCMC proposals. Normalizing flows have been used to reparameterize MCMC (Hoffman et al., 2019; Gabrié et al., 2022; Grenioux et al., 2023). We adopt and develop this methodology to serve Bayesian GLMMs: previous work fits the flow to one target model and refits it for the next; `metabeta` trains it once across datasets and priors, so a single amortized proposal serves every GLMM in its envelope.

2 METHODS

2.1 THE INFERENCE PROBLEM

Consider m groups, each contributing n_i observations $(\mathbf{X}_i, \mathbf{y}_i)$. GLMMs link predictors $\mathbf{X}_i$ to outcomes $\mathbf{y}_i$ through **fixed effects** $\boldsymbol{\beta} \in \mathbb{R}^d$ (global, shared across groups) and **random effects** $\boldsymbol{\alpha}_i \in \mathbb{R}^q$ (local, specific to each group). The outcome y_{ij} follows an exponential-family distribution $\mathcal{P}_y$, whose expectation is a function of observations and latent parameters:

$$y_{ij} \mid \eta_{ij} \sim \mathcal{P}_y\big(g^{-1}(\eta_{ij}),\, \phi\big)\,, \qquad \eta_{ij} = \mathbf{x}_{ij}^\top \boldsymbol{\beta} + \mathbf{z}_{ij}^\top \boldsymbol{\alpha}_i, \tag{1}$$

where η_{ij} is the linear predictor, $\mathbf{z}_{ij}$ is typically a subset of $\mathbf{x}_{ij}$, g is a canonical link function, and ϕ is a dispersion parameter (the residual variance $\phi = \sigma_\varepsilon^2$ for the Gaussian family, and $\phi \equiv 1$ for Bernoulli and Poisson). The random effects are viewed as samples from $\mathcal{N}_q(\mathbf{0}, \mathbf{S})$ with $\mathbf{S} = \boldsymbol{\sigma}\boldsymbol{\sigma}^\top \odot \mathbf{R}$, where $\mathbf{R}$ is the correlation matrix of the random effects.

We collect the global parameters as $\boldsymbol{\vartheta} = (\boldsymbol{\beta}, \boldsymbol{\sigma}, \mathbf{R}, \phi)$, and denote group data as $\mathbf{D}_i = [\mathbf{y}_i, \mathbf{X}_i, \mathbf{Z}_i] \in \mathbb{R}^{n_i \times (1+d+q)}$. The joint posterior $p(\boldsymbol{\vartheta}, \boldsymbol{\alpha}_{1:m} \mid \mathbf{D})$ is analytically intractable. Crucially, groups are conditionally independent given $\boldsymbol{\vartheta}$, so the posterior factorizes as

$$p(\boldsymbol{\vartheta}, \boldsymbol{\alpha}_{1:m} \mid \mathbf{D}) = \underbrace{p(\boldsymbol{\vartheta} \mid \mathbf{D})}_{\text{global}} \prod_{i=1}^{m} \underbrace{p(\boldsymbol{\alpha}_i \mid \boldsymbol{\vartheta}, \mathbf{D}_i)}_{\text{local}}. \tag{2}$$

2.2 MODEL ARCHITECTURE AND TRAINING

Architecture overview. Our model realizes the factorization in Equation 2 through a four-stage pipeline (see Figure 1). **(1) Local summarization.** Each group's observations $[\mathbf{y}_i, \mathbf{X}_i, \mathbf{Z}_i]$ of variable length n_i are compressed independently by a local summary network $\boldsymbol{\Sigma}_l$ into a fixed-size, row-permutation-invariant summary $\mathbf{s}_i \in \mathbb{R}^{h_l}$. **(2) Global summarization.** The local summaries $\mathbf{s}_{1:m}$ are stacked and processed by a global summary network $\boldsymbol{\Sigma}_g$ into a single dataset summary $\mathbf{s} \in \mathbb{R}^{h_g}$. **(3) Global posterior.** The summary $\mathbf{s}$ conditions a posterior network $\boldsymbol{\Pi}_g$, which efficiently produces samples $\hat{\boldsymbol{\vartheta}} \sim p_{\boldsymbol{\Pi}_g}(\boldsymbol{\vartheta} \mid \mathbf{s})$ and evaluates the log-density for arbitrary $\boldsymbol{\vartheta}$. **(4) Local posterior.** For each group, the local summary $\mathbf{s}_i$ and global samples $\hat{\boldsymbol{\vartheta}}$ are passed to a shared local posterior network $\boldsymbol{\Pi}_l$, which learns to sample $\hat{\boldsymbol{\alpha}}_i \sim p_{\boldsymbol{\Pi}_l}(\boldsymbol{\alpha}_i \mid \hat{\boldsymbol{\vartheta}}, \mathbf{s}_i)$. All networks are trained jointly end-to-end (Appendix E).

Summary networks. Both n_i and m vary across datasets, and observations within each level are exchangeable, so the summary networks must handle variable-length inputs and be permutation-invariant. We use Set Transformers (Lee et al., 2019), which aggregate variable-length, unordered inputs into a fixed-size vector via learned attention. A learned CLS token prepended to the input sequence serves as the pooled output. The local summarizer pools over observations per group; the global summarizer then pools over group summaries. Permutation invariance is enforced by omitting positional encodings and causal masking.

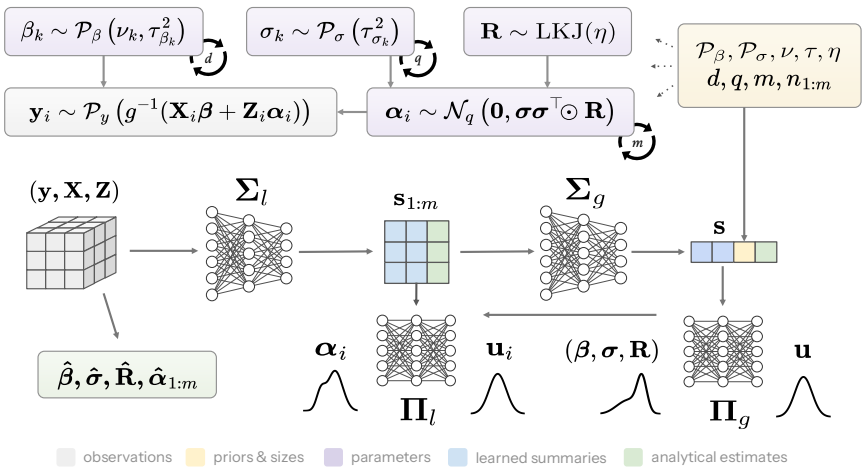


Figure 1: (Upper half) *Dataset Simulation*. Structural dimensions and prior hyperparameters are sampled; regression parameters are drawn from the chosen priors and outcomes are generated via Equation 1. Predictors come from structural causal models or real tabular datasets. (Lower half) *Model Pipeline*. $\mathbf{D}_i = (\mathbf{y}_i, \mathbf{X}_i, \mathbf{Z}_i)$ are separately summarized per group i. Local summaries are summarized over groups; the resulting global summary and prior specification condition $\boldsymbol{\Pi}_g \approx p(\boldsymbol{\vartheta} \mid \mathbf{D})$. Global posterior samples condition the local posterior $\boldsymbol{\Pi}_l \approx p(\boldsymbol{\alpha}_i \mid \boldsymbol{\vartheta}, \mathbf{D}_i)$.

Summary augmentation. Before entering the posterior networks, each summary is augmented with the prior specification: each prior family (Normal, Student-t, half-Normal, half-Student-t, Exponential) is mapped to a one-hot embedding, and the corresponding hyperparameters are appended as continuous inputs to the global summary $\mathbf{s}$. This makes the posterior networks prior-aware and enables zero-shot generalization to novel prior specifications (Appendix B.8). In addition, we append cheap analytical summary statistics to each learned summary as an inductive head start, freeing capacity for posterior geometry, tails and parameter correlations (Appendix E.3).

Posterior networks. We realize both posterior networks, $\boldsymbol{\Pi}_g$ and $\boldsymbol{\Pi}_l$, as conditional normalizing flows: learned invertible maps between a simple base distribution and an arbitrarily complex target, which enable both exact log-density evaluation and fast sampling (Papamakarios et al., 2021). We use coupling-based neural spline flows (Durkan et al., 2019) with residual multi-layer perceptron couplings (He et al., 2016). We condition the location and scale of the base distribution on the summary, so the flow need only model the residual around a data-dependent Gaussian.

Training objective. We train the network by minimizing the expected negative log-likelihood (NLL) of the neural posterior $p_{\boldsymbol{\Pi}}$ over simulated pairs $(\boldsymbol{\vartheta}, \mathbf{D})$, which is equivalent to minimizing the expected forward KL divergence to the true posterior, the mass-covering direction standard in NPE. Applied to each factor in Equation 2, the loss reads

$$\ell(f_{\boldsymbol{\Sigma}}, p_{\boldsymbol{\Pi}}) = -\mathbb{E}_{\boldsymbol{\vartheta},\boldsymbol{\alpha},\mathbf{D}}\left[\log p_{\boldsymbol{\Pi}_g}\left(\boldsymbol{\vartheta} \mid f_{\boldsymbol{\Sigma}_g}([f_{\boldsymbol{\Sigma}_l}(\mathbf{D}_i)]_{i=1}^m)\right) + \sum_{i=1}^m \log p_{\boldsymbol{\Pi}_l}(\boldsymbol{\alpha}_i \mid \boldsymbol{\vartheta},\, f_{\boldsymbol{\Sigma}_l}(\mathbf{D}_i))\right]. \quad (3)$$

At inference, the true $\boldsymbol{\vartheta}$ that conditions the local factor is replaced by global posterior samples.

2.3 Amortized Inference meets MCMC

The raw flow returns an approximate posterior in milliseconds but offers no exactness guarantee; NUTS is asymptotically exact but serial, non-amortized and restarted from scratch for every dataset. Because the trained flow supplies fast samples and a tractable proposal density, it can be scored against the exact unnormalized posterior and coupled to MCMC in two ways.

Independence Metropolis–Hastings. By default, `metabeta` refines the flow posterior with Independence Metropolis–Hastings (IMH; Tierney, 1994). The neural posterior $q(\boldsymbol{\vartheta}) \coloneqq p_{\boldsymbol{\Pi}_g}(\boldsymbol{\vartheta} \mid \mathbf{s})$ serves as a fixed proposal, and a candidate $\boldsymbol{\vartheta}'$ is accepted over the current state $\boldsymbol{\vartheta}$ with probability

$\min(1, w(\boldsymbol{\vartheta}')/w(\boldsymbol{\vartheta}))$, where $w(\boldsymbol{\vartheta}) = \tilde{p}(\boldsymbol{\vartheta} \mid \mathbf{D}) \,/\, q(\boldsymbol{\vartheta})$ is the importance weight against the unnormalized posterior. To keep the chain well-mixing, we collapse it onto the global parameters by marginalizing the random effects out of the target, so the chain explores the lower-dimensional global space alone (the same marginalization that stabilizes NUTS for latent Gaussian models; Margossian et al., 2020). This marginalization is analytical for Gaussian models; an unbiased marginal-likelihood estimate keeps it exact for Bernoulli and Poisson (Andrieu & Roberts, 2009). The random effects are then reimputed for each retained global draw from the corresponding conditional $p(\boldsymbol{\alpha}_i \mid \boldsymbol{\vartheta}, \mathbf{D}_i)$ (Rao-Blackwellization). The acceptance rate directly measures the flow–posterior overlap and doubles as a diagnostic: `metabeta` automatically flags insufficient sampling budgets and suggests one projected to be adequate (Appendix C.1).

Warm-started NUTS. Where gradient-based sampling is preferred, `metabeta` initializes the NUTS chains and their diagonal mass matrix from its draws (Zhang et al., 2022), which cuts the tuning budget fourfold and removes most divergent transitions at unchanged accuracy (Appendix C.2).

2.4 Data Simulation

Our model is trained entirely on simulated data (Figure 1) whose structure, prior specifications and size ranges follow those reported in applied Bayesian mixed-effects analyses (Appendix F.1–F.2).

Structural dimensions. For each dataset, the number of fixed effects d, the number of random effects q, the number of groups m, and the per-group observation counts $n_{1:m}$ are drawn from a meta-prior that favors smaller values consistent with real-world datasets and usage.

Priors and parameters. Prior family and hyperparameters are sampled per dataset and passed as context to the network (Section 2.2). Fixed-effect prior families are drawn from {Normal, Student-t}; scale parameter prior families from {half-Normal, half-Student-t, Exponential}. Their hyperparameters are sampled from skewed distributions, with tighter ranges for Bernoulli and Poisson to prevent extreme linear predictors (Appendix F.3). The random-effect correlation matrix follows an LKJ prior; GLMM parameters are then drawn from these priors.

Predictor sources. Fixed-effect predictors $\mathbf{X}$ are drawn from two sources that balance structural diversity and empirical realism, mitigating simulation-to-real mismatch. *(i) Structural causal model generator*: predictor matrices are sampled from structural causal models implemented by sparse multi-layer perceptrons or directed acyclic graphs, with root causes propagated through a rich pool of nonlinear activations (including randomly sampled Gaussian Process kernels). Random post-hoc transforms then yield binary, ordinal, count and censored predictors. This follows the synthetic tabular prior of TabPFN (Hollmann et al., 2025) and TabICL (Qu et al., 2025); details are in Appendix F.4. *(ii) Real tabular data*: predictor matrices are subsampled from real tabular dataset collections (PMLB, Romano et al., 2022; SRM, Lichtenberg & Şimşek, 2017; UCI, Dua & Graff, 2019; 596 datasets in total; see Appendix F.5). A separate curated collection of 40 real hierarchical datasets, mostly from `R` mixed-effects packages (`lme4`, `nlme`, `mlmRev`, `MEMSS`) is held out entirely for evaluation (see Appendix F.7).

Outcomes. Outcomes are sampled from Equation 1; continuous outcomes are standardized to unit variance, with parameters and hyperparameters rescaled analytically so that estimates return to the original scale after inference (Appendix F.6).

Test sets. Two complementary test set types (both built from the held-out real-data collection) probe orthogonal aspects of generalization: *Oracle test sets* pair real predictors with sampled parameters and outcomes. They enable direct evaluation of posterior quality given ground-truth parameters. *Real-world test sets* do not sample parameters and keep original outcomes intact. As the true parameters, priors and model structure are unknown, we measure agreement with NUTS posteriors, i.e. whether `metabeta` remains competitive with MCMC under realistic misspecification. Each test set contains 512 hierarchical datasets with varying sizes and signal-to-noise ratios. We use four versions that differ only in parameter count: *small* ($d \in$ 1–4, $q \leq 2$), *medium* ($d \in$ 5–8, $q \leq 3$), *large* ($d \in$ 9–12, $q \leq 4$) and *huge* ($d \in$ 13–16, $q \leq 5$). All four share the same data budget: at most 200 groups, at most 150 observations per group and at most 3,000 observations in total (Appendix F.2).

3 RESULTS

We organize the evaluation of `metabeta` around six claims (Section 3.1–3.6): (1) on oracle test sets with known ground truth, `metabeta` matches NUTS in parameter recovery, calibration and predictive accuracy across all three likelihood families, and outperforms Automatic Differentiation Variational Inference (ADVI; Kucukelbir et al., 2017) and the Laplace approximation (LA); (2) on real-world test sets its posteriors are nearly indistinguishable from those of NUTS; (3) the IMH head closes the raw flow's predictive gap, and the flow makes NUTS 2–12× faster while removing over 98% of its divergent transitions; (4) the importance weights of IMH yield Bayes factors that agree with bridge sampling; (5) prior conditioning makes a prior sensitivity analysis on real data forward pass that traces NUTS; a fixed-prior network reports unreliable uncertainty under prior-shift; (6) `metabeta` stays in high agreement with NUTS in stress tests including misspecification and out-of-distribution data. Throughout, `metabeta` takes 0.09–0.14 s per dataset on GPU and 0.5–2.2 s on CPU (500–1,400× and 60–210× faster than NUTS; Appendix B.5).

3.1 ORACLE BENCHMARK

On oracle test sets, we evaluate three posterior qualities. *Parameter recovery* (do posterior means track the ground truth?), measured by Pearson's r ($\rightarrow 1$) and the normalized root mean square error (NRMSE, $\rightarrow 0$). *Calibration* (do credible intervals cover the ground truth at their nominal rate?), measured by the expected coverage error and its absolute counterpart (ECE, EACE, both $\rightarrow 0$; negative ECE shows overconfidence). *Predictive accuracy* (does the inferred model explain the observed outcomes?), measured by the leave-one-out negative log likelihood (LOO-NLL, smaller is better; Vehtari et al., 2017); formal definitions are in Appendix A.1.

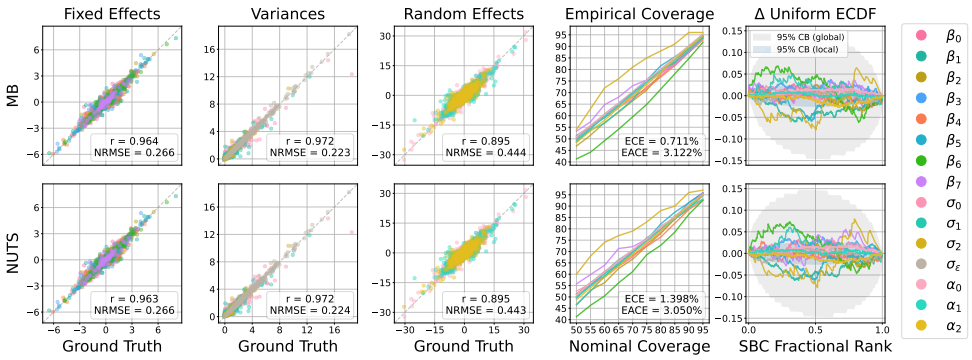


Figure 2: Performance on the *medium* Gaussian oracle benchmark. Rows compare `metabeta` (MB) and the No-U-Turn Sampler (NUTS). The first three columns show recovery for different groups of parameters. The last two show calibration by coverage analysis and simulation-based calibration checks (Talts et al., 2020; Säilynoja et al., 2022).

Table 1 summarizes performance across Gaussian models; Figure 2 visualizes the *medium* regime. Overall, `metabeta` closely matches NUTS in parameter recovery ($r = 0.94$–0.96, NRMSE within 0.01 of NUTS) across all regimes. Calibration is near-nominal for both `metabeta` and NUTS (EACE ≤ 0.03), whereas ADVI is consistently overconfident (ECE -0.14 to -0.20). Predictive accuracy matches NUTS at every size (LOO-NLL within 0.01). LA (Appendix B.4) recovers only the fixed effects; its random effects and their variance estimates are severely off across likelihoods (NRMSE 2.89–10.64), and its predictive accuracy is the worst of all methods in every family. On Bernoulli and Poisson data (Appendix B.1–B.2), `metabeta` matches NUTS at every size in recovery (r and NRMSE within 0.01), calibration (EACE ≤ 0.04) and predictive accuracy (LOO-NLL within 0.01). ADVI notably degrades on count data: its Poisson recovery drops to $r = 0.65$–0.71 . It remains overconfident in both families (ECE $= -0.14$ to -0.22).

3.2 REAL-WORLD BENCHMARK

On real-world test sets no ground truth is available, so we evaluate agreement with the NUTS posterior by the correlation of marginal posterior means (r), the ratio of posterior standard deviations (σ-ratio),

Table 1: Performance on the Gaussian oracle test sets (regimes as in Section 2.4), averaged over parameters: r, NRMSE, ECE and EACE are mean $\pm$ SD over parameters; LOO-NLL and time (seconds per dataset, 4,000 posterior draws) are median $\pm$ median absolute deviation (MAD) over NUTS-converged datasets (Appendix A.2). Timing details in Appendix B.5.

regime	model	r	NRMSE	ECE	EACE	LOO-NLL	time in s
small	MB	0.94 ± 0.04	0.32 ± 0.13	0.00 ± 0.02	0.02 ± 0.01	2.21 ± 0.53	0.09 ± 0.00
	NUTS	0.94 ± 0.04	0.32 ± 0.13	0.00 ± 0.02	0.02 ± 0.01	2.21 ± 0.53	75.91 ± 9.78
	ADVI	0.94 ± 0.05	0.33 ± 0.13	-0.14 ± 0.13	0.14 ± 0.13	2.22 ± 0.52	46.58 ± 4.86
medium	MB	0.95 ± 0.03	0.29 ± 0.10	0.00 ± 0.04	0.02 ± 0.03	2.26 ± 0.50	0.09 ± 0.00
	NUTS	0.95 ± 0.03	0.29 ± 0.10	0.00 ± 0.03	0.02 ± 0.03	2.26 ± 0.50	120.69 ± 39.28
	ADVI	0.95 ± 0.04	0.30 ± 0.11	-0.16 ± 0.11	0.16 ± 0.11	2.27 ± 0.49	62.93 ± 7.77
large	MB	0.96 ± 0.03	0.28 ± 0.09	0.00 ± 0.02	0.02 ± 0.01	2.27 ± 0.43	0.10 ± 0.00
	NUTS	0.96 ± 0.03	0.28 ± 0.09	0.00 ± 0.02	0.02 ± 0.01	2.27 ± 0.43	106.97 ± 26.52
	ADVI	0.95 ± 0.04	0.30 ± 0.10	-0.19 ± 0.11	0.19 ± 0.11	2.31 ± 0.43	66.31 ± 9.50
huge	MB	0.96 ± 0.03	0.29 ± 0.08	-0.02 ± 0.03	0.03 ± 0.02	2.25 ± 0.46	0.10 ± 0.00
	NUTS	0.96 ± 0.03	0.28 ± 0.08	0.00 ± 0.02	0.02 ± 0.01	2.25 ± 0.46	118.52 ± 21.89
	ADVI	0.95 ± 0.03	0.30 ± 0.08	-0.20 ± 0.12	0.20 ± 0.12	2.29 ± 0.47	80.43 ± 11.00

a rank-based measure of posterior shape (rank-MAD), and the difference in predictive accuracy (ΔLOO-NLL; Appendix A.1).

Table B4 (Appendix B.3) summarizes agreement across all regimes for Gaussian models. `metabeta` achieves near-perfect alignment with NUTS in terms of posterior means ($r \geq 0.99$), posterior uncertainty (σ-ratio 1.00), posterior shape (rank-MAD ≤ 0.01) and predictive accuracy (ΔLOO-NLL ≤ 0.01). ADVI matches NUTS in posterior means but is systematically overconfident (σ-ratio 0.83–0.89) and disagrees more in shape (rank-MAD $= 0.03$–0.04). The agreement carries over to the non-Gaussian models. On real Bernoulli and Poisson data, `metabeta` again reproduces NUTS's posterior means ($r = 1.00$), posterior uncertainty (σ-ratio 0.96–1.00), posterior shape (rank-MAD ≤ 0.01) and predictive accuracy (ΔLOO-NLL $= 0.00$). ADVI remains 13–20% overconfident on Bernoulli data and breaks down on real count data again: 32–45% of its Poisson fits exhaust the 10^5-iteration budget without reaching an ELBO plateau (vs. $\leq$2% in the other families; Appendix A.2). Here, ADVI's posteriors are 1.6–4.7$\times$ too wide and their means barely track NUTS ($r = 0.27$–0.65).

3.3 Synergy with MCMC

The IMH head is what turns the raw flow into `metabeta` (Section 2.3); we isolate its contribution by ablating it on every benchmark (Appendix C.1). The raw flow already tracks NUTS in recovery and calibration, but has an increasing predictive performance gap to NUTS (ΔLOO$-$NLL $= 0.06$–1.13, Table C2–C4). The IMH head closes this gap to $\leq$0.01 at every size and in every family, leaving recovery and calibration mostly unchanged; it adds 0.07–0.10 s per dataset on GPU and 0.2–1.8 s on CPU (Appendix B.5). IMH outperforms self-normalized importance sampling (SNIS; Tokdar & Kass, 2010; Dax et al., 2023; Barret & Dupourqué, 2026), whose finite-sample bias grows with model size and leaves a residual gap of 0.02–0.14 on the *huge* benchmarks (Appendix C.1).

The flow also accelerates MCMC: `MB+NUTS` initializes the chains and their mass matrix from `metabeta` draws and samples with a quarter of the default tuning budget (4 chains; 500 instead of 2,000 tuning steps; Appendix C.2). Across all oracle benchmarks it matches cold-start NUTS on every accuracy metric (Table C2–C4), divergent transitions drop by more than 98% (Gaussian 10,153 $\rightarrow$ 55, Bernoulli 4,951 $\rightarrow$ 21, Poisson 26,024 $\rightarrow$ 426), and sampling time per dataset falls by a factor of 2–12 (Table C5).

3.4 Model Comparison

The importance weights calculated for IMH also make model comparison a practically free byproduct: their expectation under the flow is the marginal likelihood (Kass & Raftery, 1995; Schmitt et al., 2024b): $\mathbb{E}_q[w] = \int \tilde{p}(\boldsymbol{\vartheta} \mid \mathbf{D})\, \mathrm{d}\boldsymbol{\vartheta} = \int p(\mathbf{D} \mid \boldsymbol{\vartheta})\, p(\boldsymbol{\vartheta})\, \mathrm{d}\boldsymbol{\vartheta} = p(\mathbf{D})$. Thus, the Bayes factor between

two candidate models can be estimated by the ratio of their mean importance weights, $\widehat{\mathrm{BF}}_{12} = \hat{p}(\mathbf{D} \mid \mathcal{M}_1) \,/\, \hat{p}(\mathbf{D} \mid \mathcal{M}_2)$. We validated this on the Gaussian oracle sets against bridge sampling on NUTS draws (Gronau et al., 2017): the Bayes factor falls into the same Jeffreys category (strength-of-evidence bands with cut points at $\mathrm{BF} = 3, 10, 30, 100$; Jeffreys, 1961; Wagenmakers et al., 2010) as the reference on 94–100% of datasets (Appendix B.6). Nearly all disagreements are flagged by the Pareto-smoothed importance sampling diagnostic (PSIS-$\hat{k} > 0.7$; Appendix D.6) or fall within 0.2 nats of a Jeffreys threshold. Preliminary checks for non-Gaussian models reach the reference's noise floor (Appendix B.6).

3.5 What Prior Conditioning Buys

Prior conditioning turns a sensitivity analysis into a batched forward pass. We condition `metabeta` on 132, 60 and 54 analyst priors for one held-out real dataset per likelihood family, each with a correlated random slope (`sleepstudy`, `cbpp`, `Salamanders`; Appendix B.7). Each grid takes 1.6–2.8 s with `metabeta`, model comparisons included; NUTS needs 7–17 hours per grid and fails its convergence checks at 27 of the 246 priors. Over the NUTS-converged cases, `metabeta` traces NUTS across the grid ($r = 1.00$, σ-ratio 0.99–1.00; Figure B15). The same pass yields the evidence, which matches bridge sampling on the NUTS draws within 0.021 nats at the median.

Prior-conditioning also guards the amortized model against prior-shifts. A network trained at a fixed prior P_0 will be miscalibrated on problems under a different prior P_1. We emulate this by conditioning the raw flow on a fixed P_0 while data and NUTS stay under a varying P_1 on all oracle test sets (Appendix B.9). A P_0 three times wider than P_1 makes the posterior 11–29% too wide. Three times tighter makes the posterior 9–35% too narrow, and IMH cannot repair it, because a proposal narrower than its target cannot supply the tails. Posterior means and predictions are unaffected throughout ($r \geq 0.98$): the damage is confined to uncertainty quantification, which a single dataset does not reveal without a sampler or a dedicated local calibration check (Zhao et al., 2021; Linhart et al., 2023; Säilynoja et al., 2025).

3.6 Robustness

Amortized inference is known to degrade silently when test data fall outside the training simulator (Cannon et al., 2022; Ward et al., 2022; Schmitt et al., 2024a). We stress-test `metabeta` against NUTS along five axes (Appendix D): misspecified likelihoods (outcomes regenerated from heavier-tailed or overdispersed families), misspecified priors (scale, location or family deviate from the generative prior), out-of-distribution predictors (infinite-variance marginals, tail-dependent copulas, longitudinal structure), collinear designs (binned by the condition number $\kappa_2(\mathbf{X})$ up to near-singular), and data-poor groups (fewer observations than parameters). On the first two axes both methods target the same wrong posterior; agreement with NUTS separates amortization error from modeling error. On the remaining three axes the fitted model stays correct and NUTS is an exact reference.

For each stress test, Table 2 reports the condition that degrades the raw neural posterior the most, as measured by its predictive gap to NUTS. NUTS itself only converges on 43–86% of the affected datasets, showing that the stress is substantial. Posterior inference for full `metabeta` nevertheless remains mostly indistinguishable from NUTS in every worst case ($r \geq 0.99$, σ-ratio 0.95–1.00, $|\Delta\text{LOO-NLL}| \leq 0.01$; Table D1–D12). Where the specified GLMM is wrong, `metabeta` and NUTS degrade similarly against the generating parameters (e.g. global EACE 0.37 vs. 0.36 under target overdispersion, 0.29 vs. 0.27 under falsely certain priors; Table D3–D4). The picture is the same for data poverty: in groups with fewer observations than parameters ($n_i < d + q$), `metabeta` matches NUTS in local recovery and calibration (Table D13–D15). Overall, no stress we applied leads to a meaningful disagreement between `metabeta` and NUTS.

4 Discussion

We introduced `metabeta`, a pretrained prior-amortized in-context model for Bayesian GLMMs. On oracle datasets with known ground truth, on held-out real datasets, and under every stress test we applied, its posteriors closely match those of NUTS in parameter recovery, calibration, shape and predictive accuracy. Amortization also makes it two to three orders of magnitude faster. The standard fast alternative, ADVI, is systematically overconfident in every likelihood family, a known property

Table 2: Worst-case condition per stress test and likelihood family (selection and full results in Appendix D). Entries are median ± MAD over NUTS-converged datasets; agreement measures as in Section 3.2; % conv. is the NUTS convergence rate.

stress	family	worst condition	% conv.	r	σ-ratio	ΔLOO-NLL
likelihood (D.1)	Gaussian	Student-t, ν=3	73	1.00 ± 0.00	1.00 ± 0.02	0.00 ± 0.00
	Bernoulli	logit noise, SD 2	79	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00
	Poisson	Neg. Binomial, θ=1	86	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00
prior (D.2)	Gaussian	mean +2 prior SD	72	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00
	Bernoulli	scale ×3	79	1.00 ± 0.00	0.99 ± 0.02	0.00 ± 0.00
	Poisson	scale ×3	71	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00
predictors (D.3)	Gaussian	Clayton, τ=0.9	63	1.00 ± 0.00	0.99 ± 0.01	0.00 ± 0.00
	Bernoulli	Cauchy X	82	1.00 ± 0.00	0.99 ± 0.01	0.00 ± 0.00
	Poisson	Cauchy X	55	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00
collinearity (D.4)	Gaussian	$\kappa_2 \in [10, 10^6)$	43	0.99 ± 0.01	1.00 ± 0.04	0.01 ± 0.01
	Bernoulli	$\kappa_2 \in [10, 10^6)$	72	1.00 ± 0.00	1.00 ± 0.02	0.00 ± 0.00
	Poisson	$\kappa_2 \in [6, 10)$	77	1.00 ± 0.00	0.95 ± 0.03	0.00 ± 0.00

of mean-field VI (Turner & Sahani, 2011; Giordano et al., 2018), and only 1.3–2.0× faster than NUTS (Appendix B.5).

Our results remove the three obstacles named in Section 1: (1) analyses over many datasets, model structures and priors become one batched forward pass due to amortization (Section 3.5); (2) wall time is governed by the network and IMH budget rather than by posterior geometry, so latency is bounded in advance; (3) the model is tuning-free, and the IMH acceptance rate doubles as a built-in diagnostic (Appendix D.6).

4.1 Limitations

Supported envelope. No simulation-based inference method can claim uniform accuracy; our claims are restricted to the benchmarked model class: up to 16 fixed effects, up to 5 correlated random effects, one grouping factor, and Gaussian, Bernoulli or Poisson likelihoods. This envelope covers a large majority of the reported GLMM analyses in our survey of 1,133 publications (Appendix F.1). Beyond the tested range, both actual and potential underperformance are made detectable: a low IMH acceptance rate and high PSIS-$\hat{k}$ flag the former; posteriors that drift from their analytical MAP anchors, and inputs outside the training ranges flag the latter (Appendix D.6).

Model class. Nested and crossed random effects occupy a meaningful share of GLMMs in practice but are not captured by a single set of group-level summaries; extending `metabeta` to such designs is future work. Further likelihoods (negative-binomial, ordinal, zero-inflated) require new simulators, appropriate cheap analytical estimators and trained weights, not a new architecture.

4.2 Outlook and Conclusion

Two further directions stand out. First, the forward-KL objective does not target proposal efficiency; fine-tuning the flow on the χ^2 divergence, whose Monte Carlo estimate is the variance of the importance weights (Müller et al., 2019), could raise IMH acceptance. Second, the evidence estimate of Section 3.4 makes Bayes factors between candidate model structures or priors a single batched forward pass (Section 3.5), opening a path to amortized Bayesian model comparison, for discrete outcomes too (preliminary, Appendix B.6). `metabeta` makes Bayesian GLMM inference a batched forward pass, with three operating modes on the speed–exactness spectrum: (1) the raw flow in milliseconds on GPU; (2) default `metabeta` with IMH refinement in ≈0.1 s on GPU and 0.5–2.3 s on CPU; and (3) `MB+NUTS`, a warm start making NUTS up to 12× faster and more stable. Amortized inference and MCMC come out as collaborators rather than competitors.

AI Use Statement

We used generative AI tools for the following tasks. Experiments were designed by the authors and then discussed with, reviewed by, and refined with AI tools; parts of the experiment code were drafted by AI tools, including memory optimization and debugging. The model source code was written by the authors and reviewed and refactored with the help of AI tools. Unit tests were extended by AI tools. Results were partly discussed with AI tools. Research ideas and model extensions originated with the authors and were refined in discussions with AI tools; the theoretical framework itself was developed by the authors with the help of the cited literature. The survey of Appendix F.1 was carried out by an LLM pipeline; its models, the extraction protocol and the manual audit are described there. No qualitative/thematic analysis beyond it used AI tools. We did not use generative AI to generate datasets (the simulator is described in Section 2.4 and Appendix F), to formulate mathematical claims or proofs (all follow from cited sources, are textbook results or were derived by hand), for translation, for cleaning or reformatting datasets. We have reviewed all AI-assisted work. Every piece of code written or altered by AI was inspected, and the survey extractions were checked for hallucinations by hand. We take responsibility for the final content of this work.

Ethics Statement

All datasets and software used in this work are open source and cited. `metabeta` is a tool for statistical analysis in the empirical sciences; we see no obvious opportunity for malicious use. Its amortized posteriors are approximations whose reliability is flagged per dataset by the diagnostics of Section 2.3. We recommend that practitioners whose critical decisions depend on the results prototype with `metabeta`. If IMH acceptance is low or the neural proposal is flagged by the PSIS-$\hat{k}$ diagnostic, we recommend to base final conclusions on `MB+NUTS`.

Reproducibility Statement

The anonymized source code linked in Section 1 covers every part of this paper, from the simulator (Appendix F) and model training (Appendix E) to the evaluation protocol (Appendix A), the experiments and the plots. The software is released as a Python package with demo notebooks; network weights are downloaded automatically on first use. The held-out test collection is described in Appendix F.7 and the survey data are released with the code.

References

Oriol Abril-Pla, Virgile Andreani, Colin Carroll, Larry Dong, Christopher J. Fonnesbeck, Maxim Kochurov, Ravin Kumar, Junpeng Lao, Christian C. Luhmann, Osvaldo A. Martin, Michael Osthege, Ricardo Vieira, Thomas Wiecki, and Robert Zinkov. PyMC: A modern, and comprehensive probabilistic programming framework in Python. *PeerJ Computer Science*, 9:e1516, September 2023. ISSN 2376-5992. doi: 10.7717/peerj-cs.1516.

Christophe Andrieu and Gareth O. Roberts. The pseudo-marginal approach for efficient Monte Carlo computations. *The Annals of Statistics*, 37(2):697–725, 2009. doi: 10.1214/07-AOS574.

Christophe Andrieu, Anthony Lee, and Matti Vihola. Uniform ergodicity of the iterated conditional SMC and geometric ergodicity of particle Gibbs samplers. *Bernoulli*, 24(2):842–872, 2018. doi: 10.3150/15-BEJ785.

Jonas Arruda, Yannik Schälte, Clemens Peiter, Olga Teplytska, Ulrich Jaehde, and Jan Hasenauer. An amortized approach to non-linear mixed-effects modeling based on neural posterior estimation. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 1865–1901. PMLR, July 2024.

Didier Barret and Simon Dupourqué. Simulation-based inference with neural posterior estimation applied to X-ray spectral fitting. III. Deriving exact posteriors with dimension reduction and importance sampling. *Astronomy & Astrophysics*, 708:A280, April 2026. doi: 10.1051/0004-6361/202557639.

Douglas Bates, Martin Mächler, Ben Bolker, and Steve Walker. Fitting Linear Mixed-Effects Models Using **lme4**. *Journal of Statistical Software*, 67(1), 2015. ISSN 1548-7660. doi: 10.18637/jss.v067.i01.

M. J. Betancourt and Mark Girolami. Hamiltonian Monte Carlo for Hierarchical Models. December 2013. doi: 10.48550/arXiv.1312.0906.

David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational Inference: A Review for Statisticians. *Journal of the American Statistical Association*, 112(518):859–877, April 2017. ISSN 0162-1459, 1537-274X. doi: 10.1080/01621459.2017.1285773.

N. E. Breslow and D. G. Clayton. Approximate Inference in Generalized Linear Mixed Models. *Journal of the American Statistical Association*, 88(421):9–25, 1993. ISSN 0162-1459. doi: 10.1080/01621459.1993.10594284.

Mollie E. Brooks, Kasper Kristensen, Koen J. van Benthem, Arni Magnusson, Casper W. Berg, Anders Nielsen, Hans J. Skaug, Martin Mächler, and Benjamin M. Bolker. glmmTMB balances speed and flexibility among packages for zero-inflated generalized linear mixed modeling. *The R Journal*, 9(2):378–400, 2017. doi: 10.32614/RJ-2017-066.

Paul-Christian Bürkner. Advanced Bayesian Multilevel Modeling with the R Package brms. *The R Journal*, 10 (1):395, 2018. ISSN 2073-4859. doi: 10.32614/RJ-2018-017.

Patrick Cannon, Daniel Ward, and Sebastian M. Schmon. Investigating the Impact of Model Misspecification in Neural Simulation-based Inference, September 2022.

Tomás Capretto, Camen Piho, Ravin Kumar, Jacob Westfall, Tal Yarkoni, and Osvaldo A. Martin. Bambi: A Simple Interface for Fitting Bayesian Linear Models in Python. *Journal of Statistical Software*, 103:1–29, August 2022. ISSN 1548-7660. doi: 10.18637/jss.v103.i15.

William G. Cochran. The combination of estimates from different experiments. *Biometrics. Journal of the International Biometric Society*, 10(1):101–129, 1954. ISSN 0006-341X. doi: 10.2307/3001666.

Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, December 2020. ISSN 0027-8424, 1091-6490. doi: 10.1073/pnas.1912789117.

Maximilian Dax, Stephen R. Green, Jonathan Gair, Michael Pürrer, Jonas Wildberger, Jakob H. Macke, Alessandra Buonanno, and Bernhard Schölkopf. Neural Importance Sampling for Rapid and Reliable Gravitational-Wave Inference. *Physical Review Letters*, 130(17):171403, April 2023. ISSN 0031-9007, 1079-7114. doi: 10.1103/PhysRevLett.130.171403.

Aaron Defazio, Xingyu Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, and Ashok Cutkosky. The Road Less Scheduled. *Advances in Neural Information Processing Systems*, 37:9974–10007, December 2024.

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977. ISSN 0035-9246.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using Real NVP. In *International Conference on Learning Representations*, February 2017.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2019.

Conor Durkan, Artur Bekasov, Iain Murray, and George Papamakarios. Neural Spline Flows. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Jorge I. Figueroa-Zúñiga, Reinaldo B. Arellano-Valle, and Silvia L. P. Ferrari. Mixed beta regression: A Bayesian perspective. *Computational Statistics & Data Analysis*, 61:137–147, May 2013. ISSN 0167-9473. doi: 10.1016/j.csda.2012.12.002.

Marylou Gabrié, Grant M. Rotskoff, and Eric Vanden-Eijnden. Adaptive Monte Carlo augmented with normalizing flows. *Proceedings of the National Academy of Sciences*, 119(10):e2109420119, March 2022. doi: 10.1073/pnas.2109420119.

Timothy D. Gebhard, Jonas Wildberger, Maximilian Dax, Daniel Angerhausen, Sascha P. Quanz, and Bernhard Schölkopf. Inferring Atmospheric Properties of Exoplanets with Flow Matching and Neural Importance Sampling, December 2023.

Alan E. Gelfand and Adrian F. M. Smith. Sampling-Based Approaches to Calculating Marginal Densities. *Journal of the American Statistical Association*, 85(410):398–409, 1990. ISSN 0162-1459. doi: 10.2307/2289776.

Andrew Gelman and Jennifer Hill. *Data Analysis Using Regression and Multilevel/Hierarchical Models*. Analytical Methods for Social Research. Cambridge Univ. Press, Cambridge, 23rd printing edition, 2007. ISBN 978-0-521-68689-1 978-0-521-86706-1.

Andrew Gelman, John B. Carlin, Hal S. Stern, David B. Dunson, Aki Vehtari, and Donald B. Rubin. *Bayesian Data Analysis*. Texts in Statistical Science Series. CRC Press, Taylor & Francis Group, Boca Raton London New York, third edition edition, 2013. ISBN 978-1-4398-4095-5.

Andrew Gelman, Aki Vehtari, Richard McElreath, Daniel Simpson, Charles C. Margossian, Yuling Yao, Lauren Kennedy, Jonah Gabry, Paul-Christian Bürkner, Martin Modrák, and Vianey Leos Barajas. *Bayesian Workflow*. CRC Press, Boca Raton, FL ; Abingdon, Oxon, first edition edition, 2026. ISBN 978-0-367-49018-8 978-0-367-49014-0.

Ryan Giordano, Tamara Broderick, and Michael I. Jordan. Covariances, Robustness, and Variational Bayes. *Journal of Machine Learning Research*, 19(51):1–49, 2018. ISSN 1533-7928.

Manuel Gloeckler, Michael Deistler, Christian Dietrich Weilbach, Frank Wood, and Jakob H. Macke. All-in-one simulation-based inference. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 15735–15766. PMLR, July 2024.

Katherine R. Gordon. How Mixed-Effects Modeling Can Advance Our Understanding of Learning and Memory and Improve Clinical and Educational Practice. *Journal of Speech, Language, and Hearing Research : JSLHR*, 62(3):507–524, March 2019. ISSN 1092-4388. doi: 10.1044/2018_JSLHR-L-ASTM-18-0240.

David Greenberg, Marcel Nonnenmacher, and Jakob Macke. Automatic Posterior Transformation for Likelihood-Free Inference. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 2404–2414. PMLR, May 2019.

Louis Grenioux, Alain Oliviero Durmus, Éric Moulines, and Marylou Gabrié. On sampling with approximate transport maps. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 11698–11733. PMLR, July 2023. URL https://proceedings.mlr.press/v202/grenioux23a.html.

Quentin F. Gronau, Alexandra Sarafoglou, Dora Matzke, Alexander Ly, Udo Boehm, Maarten Marsman, David S. Leslie, Jonathan J. Forster, Eric-Jan Wagenmakers, and Helen Steingroever. A tutorial on bridge sampling. *Journal of Mathematical Psychology*, 81:80–97, 2017. ISSN 0022-2496. doi: 10.1016/j.jmp.2017.09.005.

Daniel Habermann, Marvin Schmitt, Lars Kühmichel, Andreas Bulling, Stefan T. Radev, and Paul-Christian Bürkner. Amortized Bayesian Multilevel Models. *Bayesian Analysis*, -1(-1):1–30, January 2025. ISSN 1936-0975, 1931-6690. doi: 10.1214/25-BA1570.

Xavier A. Harrison, Lynda Donaldson, Maria Eugenia Correa-Cano, Julian Evans, David N. Fisher, Cecily E. D. Goodwin, Beth S. Robinson, David J. Hodgson, and Richard Inger. A brief introduction to mixed effects modelling and multi-model inference in ecology. *PeerJ*, 6:e4794, May 2018. ISSN 2167-8359. doi: 10.7717/peerj.4794.

W. K. Hastings. Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika*, 57 (1):97–109, 1970. ISSN 0006-3444. doi: 10.2307/2334940.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, June 2016. doi: 10.1109/CVPR.2016.90.

C. R. Henderson. Estimation of variance and covariance components. *Biometrics. Journal of the International Biometric Society*, 9(2):226–252, 1953. ISSN 0006-341X. doi: 10.2307/3001853.

Matthew D. Hoffman and Andrew Gelman. The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(47):1593–1623, 2014. ISSN 1533-7928.

Matthew D. Hoffman and Pavel Sountsov. Tuning-Free Generalized Hamiltonian Monte Carlo. In *Proceedings of the 25th International Conference on Artificial Intelligence and Statistics*, volume 151 of *Proceedings of Machine Learning Research*, pp. 7799–7813. PMLR, 2022.

Matthew D. Hoffman, Pavel Sountsov, Joshua V. Dillon, Ian Langmore, Dustin Tran, and Srinivas Vasudevan. NeuTra-lizing Bad Geometry in Hamiltonian Monte Carlo Using Neural Transport, March 2019.

Matthew D. Hoffman, Alexey Radul, and Pavel Sountsov. An Adaptive-MCMC Scheme for Setting Trajectory Lengths in Hamiltonian Monte Carlo. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 3907–3915. PMLR, 2021.

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(8045):319–326, January 2025. ISSN 0028-0836, 1476-4687. doi: 10.1038/s41586-024-08328-6.

Jerry M. Huang, Lukas Schumacher, Niek Stevenson, and Stefan T. Radev. CogFormer: Learn All Your Models Once, June 2026.

Harold Jeffreys. *Theory of Probability*. The International Series of Monographs on Physics. Oxford University Press, Oxford, 3 edition, 1961. ISBN 978-0-19-850368-2.

Robert E. Kass and Adrian E. Raftery. Bayes Factors. *Journal of the American Statistical Association*, 90(430): 773–795, 1995. ISSN 0162-1459. doi: 10.1080/01621459.1995.10476572.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization, January 2017.

Durk P Kingma and Prafulla Dhariwal. Glow: Generative Flow with Invertible 1x1 Convolutions. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018.

Alp Kucukelbir, Dustin Tran, Rajesh Ranganath, Andrew Gelman, and David M. Blei. Automatic Differentiation Variational Inference. *Journal of Machine Learning Research*, 18(14):1–45, 2017. ISSN 1533-7928.

Nan M. Laird and James H. Ware. Random-effects models for longitudinal data. *Biometrics. Journal of the International Biometric Society*, 38(4):963–974, 1982. ISSN 0006-341X. doi: 10.2307/2529876.

Juho Lee, Yoonho Lee, Jungtaek Kim, Adam Kosiorek, Seungjin Choi, and Yee Whye Teh. Set Transformer: A Framework for Attention-based Permutation-Invariant Neural Networks. In *Proceedings of the 36th International Conference on Machine Learning*, pp. 3744–3753. PMLR, May 2019.

Daniel Lewandowski, Dorota Kurowicka, and Harry Joe. Generating random correlation matrices based on vines and extended onion method. *Journal of Multivariate Analysis*, 100(9):1989–2001, October 2009. ISSN 0047259X. doi: 10.1016/j.jmva.2009.04.008.

Jan M. Lichtenberg and Özgür Şimşek. Simple Regression Models. In *Proceedings of the NIPS 2016 Workshop on Imperfect Decision Makers*, pp. 13–25. PMLR, August 2017.

Julia Linhart, Alexandre Gramfort, and Pedro L. C. Rodrigues. L-C2ST: Local Diagnostics for Posterior Approximations in Simulation-Based Inference. In *Advances in Neural Information Processing Systems*, volume 36, 2023. URL https://arxiv.org/abs/2306.03580.

Jun S. Liu. Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, 6(2):113–119, June 1996. doi: 10.1007/BF00162521.

Jan-Matthis Lueckmann, Pedro J Goncalves, Giacomo Bassetto, Kaan Öcal, Marcel Nonnenmacher, and Jakob H Macke. Flexible statistical inference for mechanistic models of neural dynamics. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Charles C. Margossian, Aki Vehtari, Daniel Simpson, and Raj Agrawal. Hamiltonian Monte Carlo using an adjoint-differentiated Laplace approximation: Bayesian inference for latent Gaussian models and beyond. In *Advances in Neural Information Processing Systems*, volume 33, 2020.

P. McCullagh and J. A. Nelder. *Generalized Linear Models*. Chapman & Hall, London, 2 edition, 1989. ISBN 978-0-412-31760-6.

Xiao-Li Meng and Wing Hung Wong. Simulating ratios of normalizing constants via a simple identity: A theoretical exploration. *Statistica Sinica*, 6(4):831–860, 1996.

Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6): 1087–1092, June 1953. ISSN 0021-9606. doi: 10.1063/1.1699114.

Sarthak Mittal, Niels Leif Bracher, Guillaume Lajoie, Priyank Jaini, and Marcus Brubaker. Amortized In-Context Bayesian Posterior Estimation, February 2025.

Samuel Müller, Noah Hollmann, Sebastian Pineda Arango, Josif Grabocka, and Frank Hutter. Transformers Can Do Bayesian Inference. In *International Conference on Learning Representations*, October 2021.

Thomas Müller, Brian McWilliams, Fabrice Rousselle, Markus Gross, and Jan Novák. Neural Importance Sampling. *ACM Transactions on Graphics*, 38(5):145:1–145:19, 2019. doi: 10.1145/3341156.

George Papamakarios and Iain Murray. Fast \epsilon -free Inference of Simulation Models with Bayesian Conditional Density Estimation. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

George Papamakarios, Eric Nalisnick, Danilo Jimenez Rezende, Shakir Mohamed, and Balaji Lakshminarayanan. Normalizing Flows for Probabilistic Modeling and Inference. *Journal of Machine Learning Research*, 22, 2021.

Omiros Papaspiliopoulos, Gareth O. Roberts, and Martin Sköld. A General Framework for the Parametrization of Hierarchical Models. *Statistical Science*, 22(1):59–73, February 2007. ISSN 0883-4237, 2168-8745. doi: 10.1214/088342307000000014.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

José Pinheiro, Douglas Bates, and R Core Team. Nlme: Linear and Nonlinear Mixed Effects Models, November 1999.

Jingang Qu, David Holzmüller, Gaël Varoquaux, and Marine Le Morvan. TabICL: A Tabular Foundation Model for In-Context Learning on Large Data. In *Forty-Second International Conference on Machine Learning*, June 2025.

Stefan T. Radev, Ulf K. Mertens, Andreas Voss, Lynton Ardizzone, and Ullrich Kothe. BayesFlow: Learning Complex Stochastic Models With Invertible Neural Networks. *IEEE Transactions on Neural Networks and Learning Systems*, 33(4):1452–1466, 2020. ISSN 2162-237X, 2162-2388. doi: 10.1109/TNNLS.2020.3042395.

Stefan T. Radev, Marvin Schmitt, Lukas Schumacher, Lasse Elsemüller, Valentin Pratz, Yannik Schälte, Ullrich Köthe, and Paul-Christian Bürkner. BayesFlow: Amortized Bayesian Workflows With Neural Networks. *Journal of Open Source Software*, 8(89):5702, September 2023. ISSN 2475-9066. doi: 10.21105/joss.05702.

Arik Reuter, Tim G. J. Rudner, Vincent Fortuin, and David Rügamer. Can Transformers Learn Full Bayesian Inference in Context? In *Forty-Second International Conference on Machine Learning*, June 2025.

Danilo Rezende and Shakir Mohamed. Variational Inference with Normalizing Flows. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 1530–1538. PMLR, June 2015.

G. K. Robinson. That BLUP is a Good Thing: The Estimation of Random Effects. *Statistical Science*, 6(1):15–32, 1991. ISSN 0883-4237.

Pedro Rodrigues, Thomas Moreau, Gilles Louppe, and Alexandre Gramfort. HNPE: Leveraging Global Parameters for Neural Posterior Estimation. In *Advances in Neural Information Processing Systems*, volume 34, pp. 13432–13443. Curran Associates, Inc., 2021.

Joseph D Romano, Trang T Le, William La Cava, John T Gregg, Daniel J Goldberg, Praneel Chakraborty, Natasha L Ray, Daniel Himmelstein, Weixuan Fu, and Jason H Moore. PMLB v1.0: An open-source dataset collection for benchmarking machine learning methods. *Bioinformatics*, 38(3):878–880, February 2022. ISSN 1367-4803. doi: 10.1093/bioinformatics/btab727.

Vivekananda Roy. Convergence Diagnostics for Markov Chain Monte Carlo. *Annual Review of Statistics and Its Application*, 7:387–412, 2020. ISSN 2326-831X. doi: 10.1146/annurev-statistics-031219-041300.

Teemu Säilynoja, Paul-Christian Bürkner, and Aki Vehtari. Graphical Test for Discrete Uniformity and its Applications in Goodness of Fit Evaluation and Multiple Sample Comparison. *Statistics and Computing*, 32(2):32, April 2022. ISSN 0960-3174, 1573-1375. doi: 10.1007/s11222-022-10090-6.

Teemu Säilynoja, Marvin Schmitt, Paul-Christian Bürkner, and Aki Vehtari. Posterior SBC: Simulation-Based Calibration Checking Conditional on Data, February 2025.

Marvin Schmitt, Paul-Christian Bürkner, Ullrich Köthe, and Stefan T. Radev. Detecting Model Misspecification in Amortized Bayesian Inference with Neural Networks. In Ullrich Köthe and Carsten Rother (eds.), *Pattern Recognition: 45th DAGM German Conference, DAGM GCPR 2023*, pp. 541–557, Cham, 2024a. Springer Nature Switzerland. ISBN 978-3-031-54605-1. doi: 10.1007/978-3-031-54605-1_35.

Marvin Schmitt, Desi R. Ivanova, Daniel Habermann, Ullrich Köthe, Paul-Christian Bürkner, and Stefan T. Radev. Leveraging Self-Consistency for Data-Efficient Amortized Bayesian Inference. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235, pp. 43723–43741. PMLR, 2024b.

Shayle R. Searle, George Casella, and Charles E. McCulloch. *Variance Components*. Wiley, New York, 1992. ISBN 978-0-471-62834-2. doi: 10.1002/9780470316856.

Jonathan A. C. Sterne, Ian R. White, John B. Carlin, Michael Spratt, Patrick Royston, Michael G. Kenward, Angela M. Wood, and James R. Carpenter. Multiple imputation for missing data in epidemiological and clinical research: Potential and pitfalls. *BMJ*, 338:b2393, June 2009. ISSN 0959-8138, 1468-5833. doi: 10.1136/bmj.b2393.

Sean Talts, Michael Betancourt, Daniel Simpson, Aki Vehtari, and Andrew Gelman. Validating Bayesian Inference Algorithms with Simulation-Based Calibration, October 2020.

Luke Tierney. Markov Chains for Exploring Posterior Distributions. *The Annals of Statistics*, 22(4):1701–1728, December 1994. ISSN 0090-5364, 2168-8966. doi: 10.1214/aos/1176325750.

Surya T. Tokdar and Robert E. Kass. Importance sampling: A review. *WIREs Computational Statistics*, 2(1): 54–60, January 2010. ISSN 1939-5108, 1939-0068. doi: 10.1002/wics.56.

Minh-Ngoc Tran, Marcel Scharth, Michael K. Pitt, and Robert Kohn. Importance sampling squared for Bayesian inference in latent variable models, 2013.

Richard E. Turner and Maneesh Sahani. Two problems with variational expectation maximisation for time series models. In David Barber, A. Taylan Cemgil, and Silvia Chiappa (eds.), *Bayesian Time Series Models*, pp. 104–124. Cambridge University Press, Cambridge, 2011. doi: 10.1017/CBO9780511984679.006.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Aki Vehtari, Andrew Gelman, and Jonah Gabry. Practical Bayesian model evaluation using leave-one-out cross-validation and WAIC. *Statistics and Computing*, 27(5):1413–1432, 2017. doi: 10.1007/s11222-016-9696-4.

Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. Rank-Normalization, Folding, and Localization: An Improved R^ for Assessing Convergence of MCMC (with Discussion). *Bayesian Analysis*, 16(2):667–718, June 2021. ISSN 1936-0975, 1931-6690. doi: 10.1214/20-BA1221.

Eric-Jan Wagenmakers, Tom Lodewyckx, Himanshu Kuriyal, and Raoul Grasman. Bayesian hypothesis testing for psychologists: A tutorial on the Savage–Dickey method. *Cognitive Psychology*, 60(3):158–189, 2010. ISSN 0010-0285. doi: 10.1016/j.cogpsych.2009.12.001.

Daniel Ward, Patrick Cannon, Mark Beaumont, Matteo Fasiolo, and Sebastian Schmon. Robust Neural Posterior Estimation and Statistical Model Criticism. *Advances in Neural Information Processing Systems*, 35:33845–33859, December 2022.

Larry Wasserman. *All of Statistics: A Concise Course in Statistical Inference*. Springer Texts in Statistics. Springer, New York Berlin Heidelberg, corr. 2. print., [repr.] edition, 2010. ISBN 978-1-4419-2322-6.

George Whittle, Juliusz Ziomek, Jacob Rawling, and Michael A. Osborne. Distribution Transformers: Fast Approximate Bayesian Inference With On-The-Fly Prior Adaptation, October 2025.

Yuling Yao, Aki Vehtari, Daniel Simpson, and Andrew Gelman. Yes, but Did It Work?: Evaluating Variational Inference. In *Proceedings of the 35th International Conference on Machine Learning*, pp. 5581–5590. PMLR, July 2018.

Zhaoxia Yu, Michele Guindani, Steven F. Grieco, Lujia Chen, Todd C. Holmes, and Xiangmin Xu. Beyond t test and ANOVA: Applications of mixed-effects models for more rigorous statistical analysis in neuroscience research. *Neuron*, 110(1):21–35, January 2022. ISSN 1097-4199. doi: 10.1016/j.neuron.2021.10.030.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander Smola. Deep Sets. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Lu Zhang, Bob Carpenter, Andrew Gelman, and Aki Vehtari. Pathfinder: Parallel quasi-newton variational inference. *Journal of Machine Learning Research*, 23(306):1–49, 2022.

David Zhao, Niccolò Dalmasso, Rafael Izbicki, and Ann B. Lee. Diagnostics for Conditional Density Models and Bayesian Inference Algorithms. In Cassio de Campos and Marloes H. Maathuis (eds.), *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, volume 161, pp. 1830–1840. PMLR, 2021. URL https://proceedings.mlr.press/v161/zhao21b.html.

APPENDIX CONTENTS

A Evaluation Protocol **17**
A.1 Evaluation Metrics . . . 17
A.2 Reference Sampler and Baselines . . . 18

B Result Details **21**
B.1 Oracle Benchmark . . . 21
B.2 Random-Effect Correlations . . . 29
B.3 Real-World Benchmark . . . 31
B.4 Laplace Baseline . . . 32
B.5 Runtime Results . . . 32
B.6 Marginal Likelihood from Importance Weights . . . 36
B.7 Zero-Shot Prior Sensitivity Analysis . . . 38
B.8 Prior-Awareness Check . . . 40
B.9 Emulated Fixed-Prior Baseline . . . 41

C Amortized Inference meets MCMC **43**
C.1 Independence Metropolis–Hastings . . . 43
C.2 Warm-Started NUTS . . . 44

D Robustness and Stress Tests **47**
D.1 Likelihood Misspecification . . . 47
D.2 Prior Misspecification . . . 49
D.3 Out-of-Distribution Predictors . . . 51
D.4 Ill-Conditioned Designs . . . 53
D.5 Data-Poor Groups . . . 54
D.6 Deployment Safeguards . . . 56

E Architecture and Training Details **57**
E.1 Data Representation and Summary Networks . . . 57
E.2 Normalizing Flows . . . 57
E.3 Analytical Context Features . . . 58
E.4 Optimization and Hyperparameters . . . 59

F Simulator and Datasets **60**
F.1 A Survey of Published GLMM Analyses . . . 60
F.2 Size Classes . . . 60
F.3 Prior Hyperparameters . . . 61
F.4 SCM Generator . . . 62
F.5 Emulator . . . 63
F.6 Standardization . . . 64
F.7 Test Datasets . . . 66

G Preprocessing Pipeline **66**
G.1 Design Philosophy . . . 66
G.2 Pipeline Steps . . . 67

A Evaluation Protocol

This appendix describes the metrics (Appendix A.1), and the reference methods (Appendix A.2) used for the results.

A.1 Evaluation Metrics

Parameter recovery. Pearson's r measures linear agreement between posterior means $\hat{\boldsymbol{\vartheta}}$ and ground-truth parameters $\boldsymbol{\vartheta}^*$ across the test batch. Normalized RMSE, $\mathrm{NRMSE} = \mathrm{RMSE}(\hat{\boldsymbol{\vartheta}}, \boldsymbol{\vartheta}^*)/\mathrm{std}(\boldsymbol{\vartheta}^*)$, normalizes by the batch standard deviation of ground-truth values, making values comparable across scales.

Posterior calibration. Let $C(\alpha) = \frac{1}{P}\sum_{p=1}^{P} \mathbf{1}\left[\vartheta_p^* \in \left[\hat{q}_{\alpha/2}^{(p)}, \hat{q}_{1-\alpha/2}^{(p)}\right]\right]$ denote the empirical coverage at nominal level $1-\alpha$, where the sum runs over all P scalar parameters of a given type pooled across the test batch and $\hat{q}_\tau^{(p)}$ is the τ-quantile of the marginal posterior samples for parameter p. This quantifies how often the $1-\alpha$ credible interval actually contains the true parameters in that class (e.g. how accurate the credible intervals are for fixed effects). The signed expected coverage error

$$\mathrm{ECE} = \frac{1}{|A|}\sum_{\alpha\in A} C(\alpha) - (1-\alpha)$$

diagnoses the direction of miscalibration: negative values indicate overconfidence (undercoverage), positive values over-dispersion. Its absolute counterpart

$$\mathrm{EACE} = \frac{1}{|A|}\sum_{\alpha\in A} |C(\alpha) - (1-\alpha)|$$

measures the magnitude of miscalibration. Both are averaged over $A = \{0.05, 0.10, \ldots, 0.50\}$.

Simulation-based calibration (SBC). For each parameter p in a dataset with known ground truth ϑ_p^*, the *fractional rank* is the fraction of posterior samples strictly below the true value:

$$u_p = \frac{1}{S}\sum_{s=1}^{S} \mathbf{1}\left[\vartheta_p^{(s)} < \vartheta_p^*\right].$$

Under a perfectly calibrated posterior, $u_p \sim \mathrm{Uniform}(0,1)$ (Talts et al., 2020). Fractional ranks are pooled over all (parameter, dataset) pairs within each parameter type (fixed effects, random-effect scales, random-effect coefficients, etc.) and the empirical CDF is compared with the diagonal. For visual clarity, we plot the empirical CDF against the Δ to the diagonal. An S-shaped ECDF indicates over-dispersion (too-wide posteriors); an inverse-S indicates undercoverage. The displayed bands are simultaneous 95% ECDF bands following Säilynoja et al. (2022): $n_{\mathrm{sim}} = 2{,}000$ uniform samples of size n_{eff} are drawn, the minimal coverage probability is computed for each simulated path, and the $\alpha = 0.05$ quantile of these probabilities determines the band width. Simultaneous bands account for multiple comparisons across the entire ECDF.

Predictive fit. LOO-NLL is computed via PSIS leave-one-out cross-validation (Vehtari et al., 2017):

$$\text{LOO-NLL} = -\frac{1}{n}\sum_{ij} \log \hat{p}(y_{ij} \mid \mathbf{D}_{-ij}),$$

where $\hat{p}(y_{ij} \mid \mathbf{D}_{-ij}) = \sum_s \tilde{w}_s\, p(y_{ij} \mid \boldsymbol{\vartheta}^{(s)}, \boldsymbol{\alpha}_i^{(s)})$ is the importance-weighted leave-one-out predictive density with PSIS-smoothed weights $\tilde{w}_s \propto 1/p(y_{ij} \mid \boldsymbol{\vartheta}^{(s)}, \boldsymbol{\alpha}_i^{(s)})$. The PSIS computation also yields per-observation Pareto shape parameters $\hat{k}$; observations with $\hat{k} > 0.7$ indicate unreliable LOO estimates. This quantifies how likely any single left-out observation is under the GLMM, if the latter were fit without that observation.

Posterior agreement. The following metrics quantify how closely a method's posterior approximates NUTS. All three are computed after restricting to NUTS-converged datasets under the strict criterion of Appendix A.2.

Posterior mean correlation (r). For each dataset, all active posterior means $\bar{\vartheta}_{\text{method}}$ and $\bar{\vartheta}_{\text{NUTS}}$ are concatenated into flat vectors across fixed effects, random-effect scales, residual scale, and random-effect coefficients. Pearson's r is computed between these vectors; the reported statistic is median $\pm$ MAD over datasets.

Posterior standard-deviation ratio (σ-ratio). For each dataset, the per-parameter std ratio $\sigma^{(p)}_{\text{method}}/\sigma^{(p)}_{\text{NUTS}}$ is computed over all active parameters and the per-dataset median is taken. The reported statistic is median $\pm$ MAD of these per-dataset medians over datasets; values > 1 indicate a wider approximate posterior.

Rank-MAD. This metric applies the fractional-rank principle of SBC to the real-data setting: NUTS samples take the role of ϑ^*. For each active parameter and dataset, every NUTS sample is ranked within the marginal `MB` distribution and expressed as a fraction $\in [0, 1]$. Under perfect agreement these fractions follow $\text{Uniform}(0, 1)$. The metric is

$$\text{rank-MAD} = \frac{1}{|Q|} \sum_{q \in Q} \left| \hat{F}(q) - q \right|,$$

where $Q = \{0.10, 0.25, 0.50, 0.75, 0.90\}$ and $\hat{F}(q)$ is the empirical q-quantile of the pooled rank fractions across all active (parameter, dataset) pairs. A value of 0 indicates that the distributional shapes agree perfectly.

A.2 Reference Sampler and Baselines

PyMC parameterization. Since our model class explicitly includes correlated random effects (not supported by `Bambi`, Capretto et al., 2022), we implement all baseline models directly in `PyMC` (Abril-Pla et al., 2023). On the oracle and the real-world test sets alike, NUTS estimates $\mathbf{R}$ under its LKJ prior whenever $q \geq 2$. Random effects use a non-centered parameterization throughout, which improves HMC geometry in weakly-identified settings (small n_i):

Independent random effects: each dimension is parameterized separately as

$$\boldsymbol{\alpha}_i^{(j)} = \zeta_i^{(j)} \cdot \sigma_{\alpha_j}, \quad \zeta_i^{(j)} \sim \mathcal{N}(0, 1), \quad \sigma_{\alpha_j} \sim \mathcal{P}(\tau_{\sigma_j}),$$

where $\mathcal{P}$ denotes the prior family drawn at simulation time (half-Normal, half-Student-t, or Exponential).

Correlated random effects: the full covariance is $\mathbf{S} = \text{diag}(\boldsymbol{\sigma}_\alpha)\, \mathbf{R}\, \text{diag}(\boldsymbol{\sigma}_\alpha)$ with $\mathbf{R} \sim \text{LKJ}(\eta)$. The Cholesky factor $\mathbf{L}_S$ is obtained via `LKJCholeskyCov`, and random effects are drawn as

$$\boldsymbol{\alpha}_i = \boldsymbol{\zeta}_i\, \mathbf{L}_S^\top, \quad \boldsymbol{\zeta}_i \sim \mathcal{N}_q(\mathbf{0}, \mathbf{I}).$$

We verify correctness by comparing the log-posterior against `Bambi` (uncorrelated case) at 50 random unconstrained-space points across ten held-out datasets per size class; numerical differences are below 10^{-9} in both cases. Data and prior hyperparameters are passed to `PyMC` in the same standardized form that `metabeta` uses (Appendix F.6), so both target the identical posterior.

NUTS. We run 4 independent chains with 2,000 warm-up iterations and 1,000 posterior draws per chain (4,000 joint samples). Tuning adapts the step size and diagonal mass matrix using PyMC defaults: target acceptance rate 0.8, maximum tree depth 10.

NUTS convergence diagnostics. Table A1 summarizes the diagnostics of all cold-start NUTS reference fits (512 test datasets per benchmark). Four patterns stand out.

- *Divergences are common but shallow.* On the small Gaussian oracle benchmark, 35% of datasets show at least one divergent transition, but the median count among affected datasets is 3 (of 4,000 joint samples), and near-noiseless datasets (standardized $\sigma_\varepsilon < 0.10$) diverge at the same rate as the rest (32% vs. 36%).
- *The dominant failure mode is slow mixing on large random-effect blocks.* The share of datasets with $\hat{R}_{\max} > 1.01$ (Vehtari et al., 2021) grows from 8% (*small*) to 21% (*huge*) on the Gaussian oracle benchmarks, and violating datasets have roughly twice the median block size $m \times q$ of the rest. Tree-depth saturation is negligible ($\leq 1\%$ of datasets, except 5% on the *small* Poisson real-world set).
- *Real count data are harder.* Divergence prevalence is 58–74% on real Poisson data vs. 36–56% on the oracle sets (Fisher exact $p \leq 3 \times 10^{-9}$), plausibly reflecting overdispersion of real counts relative to the Poisson likelihood.
- *Poor mixing flatters NUTS.* On the Gaussian oracle benchmark, Spearman $r_s(\mathrm{ESS}_{\min}, \text{LOO-NLL}) = +0.33$, meaning that posteriors from ill-mixed chains overestimate predictive performance.

Convergence criteria. Two filters are used throughout the paper.

- *Strict* ($\hat{R}_{\max} \leq 1.01$, divergence rate $\leq 0.5\%$, $\mathrm{ESS}_{\min} \geq 400$, saturation rate $\leq 5\%$; retains 41–89% of datasets): wherever NUTS serves as the reference posterior, i.e. the real-world agreement metrics, the stress tests of Appendix D, and the cost-per-usable-posterior analysis (Appendix B.5).
- *Liberal* ($\hat{R}_{\max} \leq 1.05$, divergence rate $\leq 2\%$, $\mathrm{ESS}_{\min} \geq 200$, saturation rate $\leq 20\%$; retains 72–97%): on the oracle benchmarks, where NUTS is scored against the ground truth like every other method and only clear failures are removed.

ADVI. We use mean-field ADVI (Kucukelbir et al., 2017) with the Adam optimizer (Kingma & Ba, 2017), learning rate 10^{-3}, up to 10^5 iterations with early stopping (relative ELBO change $< 2 \times 10^{-3}$ over a 10^4-step window, checked after 5,000 steps), and 4,000 posterior draws. The budget floor of 5,000 steps is half of `PyMC`'s default of 10^4 fixed iterations.

Laplace approximation. The LA baseline represents the unnormalized log posterior in an unconstrained non-centered parameterization (Papaspiliopoulos et al., 2007): scale parameters are log-transformed and the random-effect covariance uses an unconstrained Cholesky factor. A MAP estimate is obtained via L-BFGS (with a lower-learning-rate retry on failure), the dense Hessian at the MAP via automatic differentiation, and 4,000 samples are drawn from the resulting Gaussian and transformed back to the constrained space. Non-positive-definite Hessians are repaired by jittering the diagonal; fit failures and repairs are recorded per dataset (results in Appendix B.4).

Table A1: Convergence diagnostics of the cold-start NUTS reference fits on every test set (512 datasets each; 4 chains, 2,000 warm-up iterations, 1,000 draws, target acceptance 0.8). Test sets are the oracle and real-world sets of each likelihood family and size regime (Section 2.4). divg.: total divergent transitions over all datasets; % ≥1 divg.: share of datasets with at least one divergent transition; $\hat{R}_{\text{worst}}$: largest $\hat{R}$ over all datasets and parameters; tree-sat.: share of datasets whose tree-depth saturation rate exceeds 5% of post-warm-up draws; strict / liberal: share of datasets passing the respective convergence filter.

family	regime	test set	divg.	% ≥1 divg.	% $\hat{R}_{\max}$>1.01	$\hat{R}_{\text{worst}}$	% tree-sat.	% strict	% liberal
Gaussian	small	oracle	1,737	35	8.2	1.60	0.4	83	92
	small	real	2,089	45	3.9	1.23	0.0	84	96
	medium	oracle	3,789	28	11.7	1.39	0.0	76	88
	medium	real	2,616	28	12.5	1.20	0.0	68	89
	large	oracle	2,424	25	14.8	1.61	0.4	71	88
	large	real	4,414	25	16.2	1.60	1.0	64	86
	huge	oracle	2,203	14	21.3	1.60	0.2	66	83
	huge	real	1,315	15	20.1	1.60	0.2	62	82
Bernoulli	small	oracle	988	34	1.2	1.13	0.0	89	97
	small	real	2,029	35	3.9	1.28	0.0	88	95
	medium	oracle	1,450	20	6.6	1.53	0.0	79	90
	medium	real	922	30	8.0	1.29	0.0	75	89
	large	oracle	632	21	11.7	1.22	0.0	74	88
	large	real	1,219	15	15.2	1.63	0.2	66	84
	huge	oracle	1,881	17	13.5	1.54	0.0	73	84
	huge	real	196	11	16.2	1.53	0.0	66	85
Poisson	small	oracle	5,112	56	5.9	1.08	0.0	81	93
	small	real	40,579	74	35.9	1.62	5.3	41	72
	medium	oracle	8,001	45	10.4	1.29	0.8	74	87
	medium	real	10,423	72	11.3	1.60	0.2	68	93
	large	oracle	7,830	36	12.1	1.21	0.8	73	85
	large	real	6,409	58	8.6	1.15	0.0	74	91
	huge	oracle	5,081	30	14.6	1.20	0.6	70	85

B Result Details

This appendix collects the remaining per-family results: the remaining oracle results (Appendix B.1), the separate random-effect correlation results (Appendix B.2), the non-Gaussian real-world tables (Appendix B.3), the Laplace baseline (Appendix B.4), the runtime breakdown (Appendix B.5), the marginal-likelihood validation (Appendix B.6), the prior-sensitivity use case on real data (Appendix B.7), a check that the network actually uses its prior-family input (Appendix B.8), and an emulated fixed-prior baseline (Appendix B.9). Metrics and convergence criteria are defined in Appendix A.

B.1 Oracle Benchmark

The figures on the following pages complement Figure 2; rows, columns and descriptors are reused throughout. Table 1 in the main text summarizes the Gaussian regimes; Table B1 and Table B2 do so for Bernoulli and Poisson.

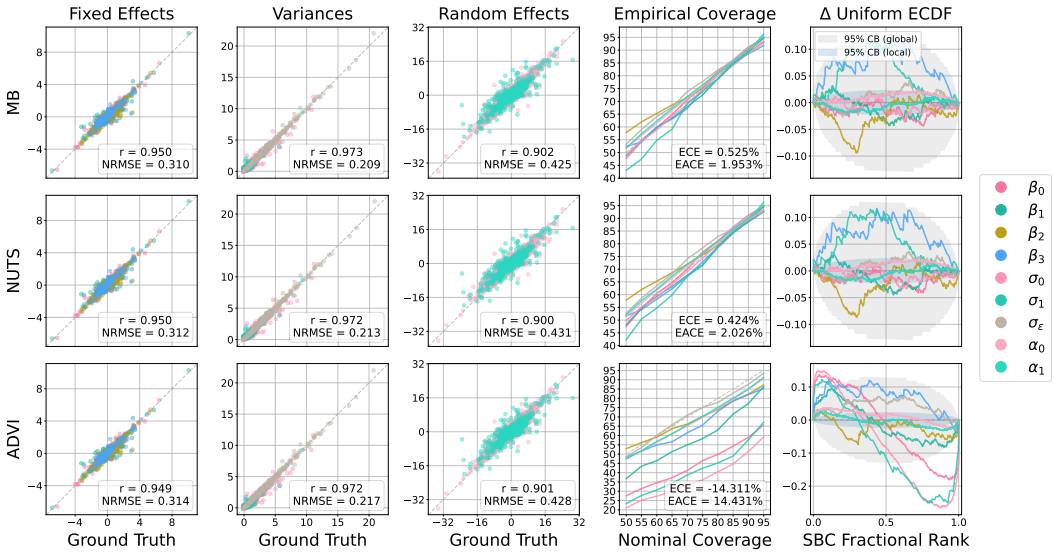


Figure B1: Performance on the *small* Gaussian oracle benchmark; layout as in Figure 2.

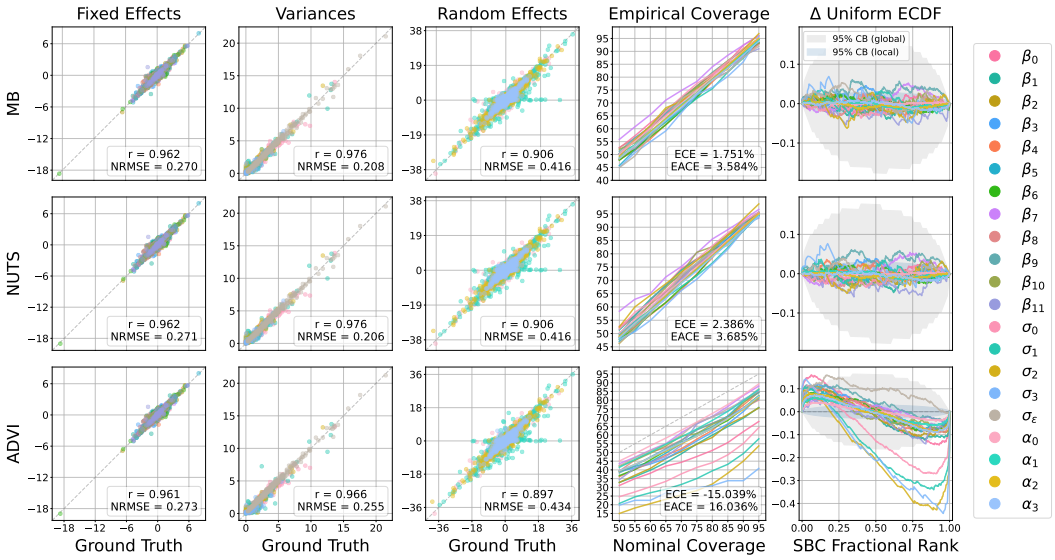


Figure B2: Performance on the *large* Gaussian oracle benchmark; layout as in Figure 2.

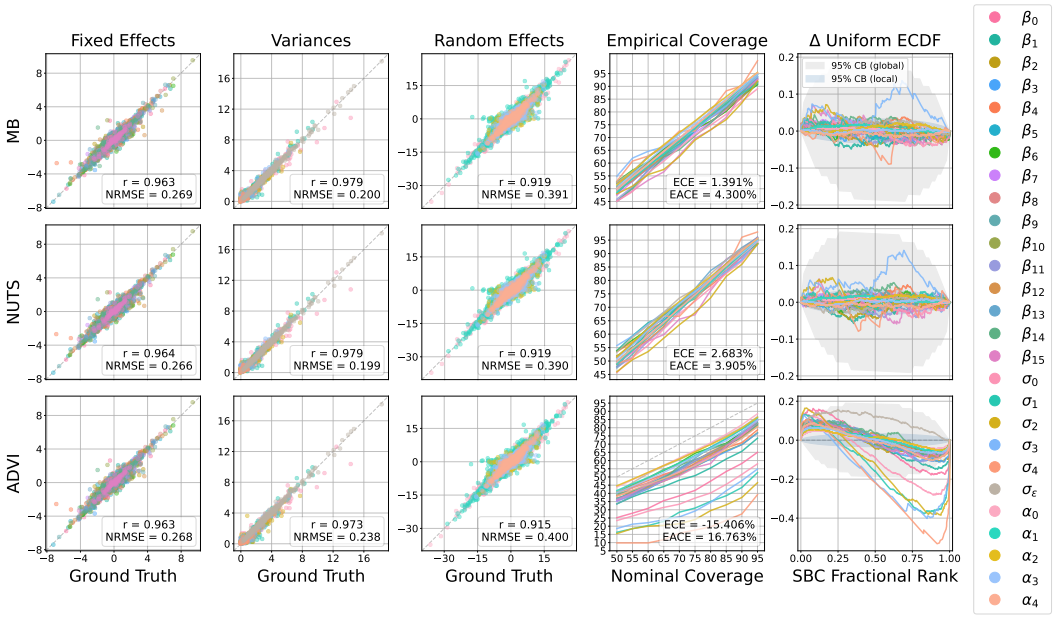


Figure B3: Performance on the *huge* Gaussian oracle benchmark; layout as in Figure 2.

Table B1: Performance on each Bernoulli oracle test set averaged over all parameters. Layout as in Table 1.

regime	model	r	NRMSE	ECE	EACE	LOO-NLL	time
small	MB	0.89 ± 0.09	0.42 ± 0.17	0.00 ± 0.02	0.02 ± 0.01	0.55 ± 0.09	0.10 ± 0.00
	NUTS	0.89 ± 0.09	0.42 ± 0.17	0.00 ± 0.02	0.02 ± 0.01	0.55 ± 0.09	73.90 ± 9.27
	ADVI	0.89 ± 0.10	0.44 ± 0.17	-0.14 ± 0.08	0.14 ± 0.08	0.55 ± 0.09	43.85 ± 4.84
medium	MB	0.90 ± 0.09	0.41 ± 0.16	0.00 ± 0.02	0.02 ± 0.01	0.45 ± 0.06	0.11 ± 0.01
	NUTS	0.90 ± 0.09	0.41 ± 0.16	0.00 ± 0.02	0.02 ± 0.01	0.45 ± 0.06	102.59 ± 21.47
	ADVI	0.90 ± 0.08	0.40 ± 0.16	-0.14 ± 0.08	0.14 ± 0.08	0.45 ± 0.06	59.10 ± 8.35
large	MB	0.88 ± 0.11	0.43 ± 0.18	0.00 ± 0.02	0.02 ± 0.01	0.43 ± 0.05	0.11 ± 0.01
	NUTS	0.88 ± 0.11	0.43 ± 0.18	0.01 ± 0.02	0.02 ± 0.01	0.43 ± 0.05	88.38 ± 15.93
	ADVI	0.88 ± 0.12	0.44 ± 0.19	-0.15 ± 0.09	0.15 ± 0.09	0.43 ± 0.05	62.62 ± 8.42
huge	MB	0.86 ± 0.12	0.46 ± 0.18	-0.01 ± 0.03	0.03 ± 0.02	0.41 ± 0.05	0.12 ± 0.01
	NUTS	0.86 ± 0.12	0.46 ± 0.18	0.00 ± 0.03	0.02 ± 0.02	0.41 ± 0.05	118.95 ± 29.12
	ADVI	0.85 ± 0.13	0.48 ± 0.20	-0.17 ± 0.07	0.17 ± 0.07	0.42 ± 0.06	77.91 ± 13.14

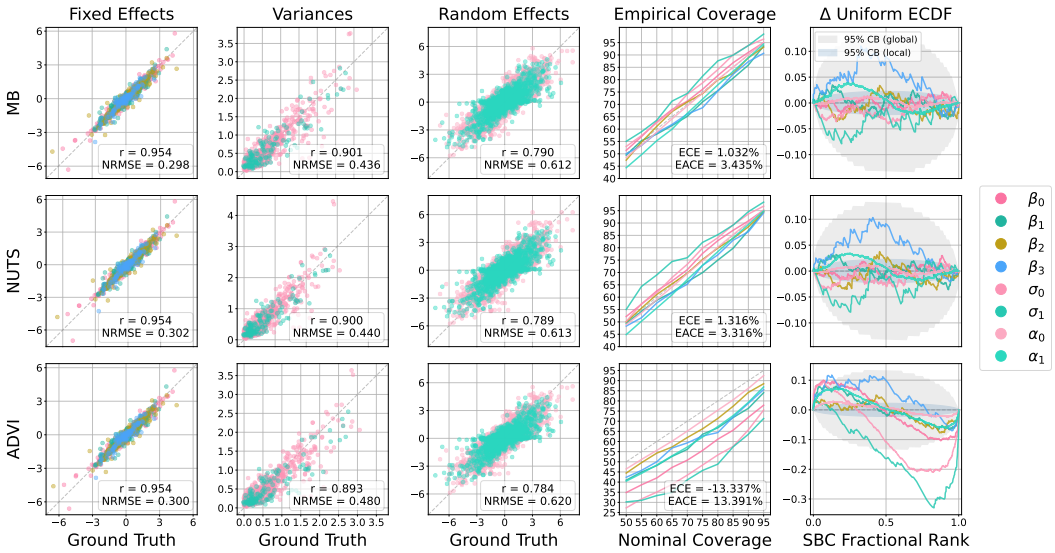


Figure B4: Performance on the *small* Bernoulli oracle benchmark; layout as in Figure 2.

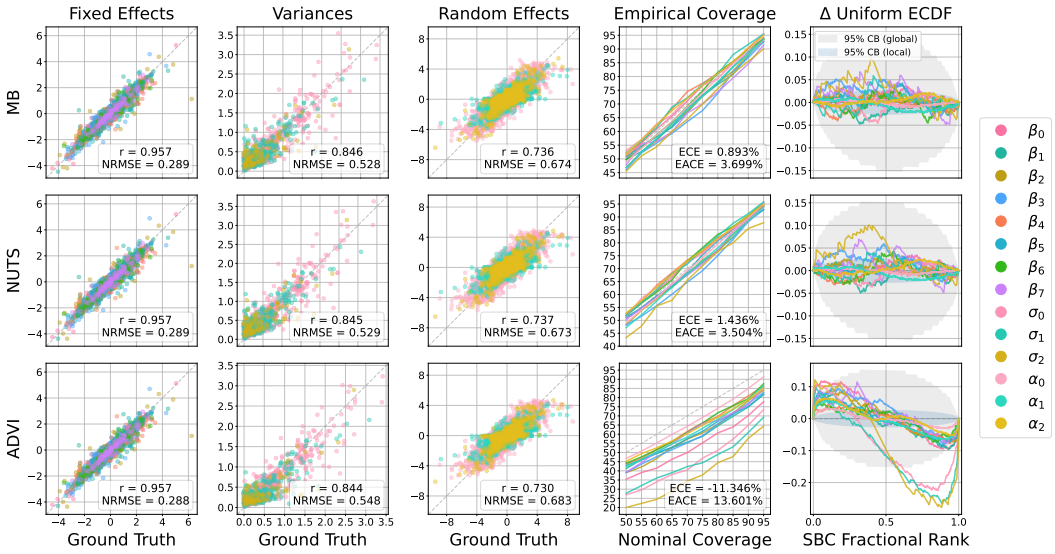


Figure B5: Performance on the *medium* Bernoulli oracle benchmark; layout as in Figure 2.

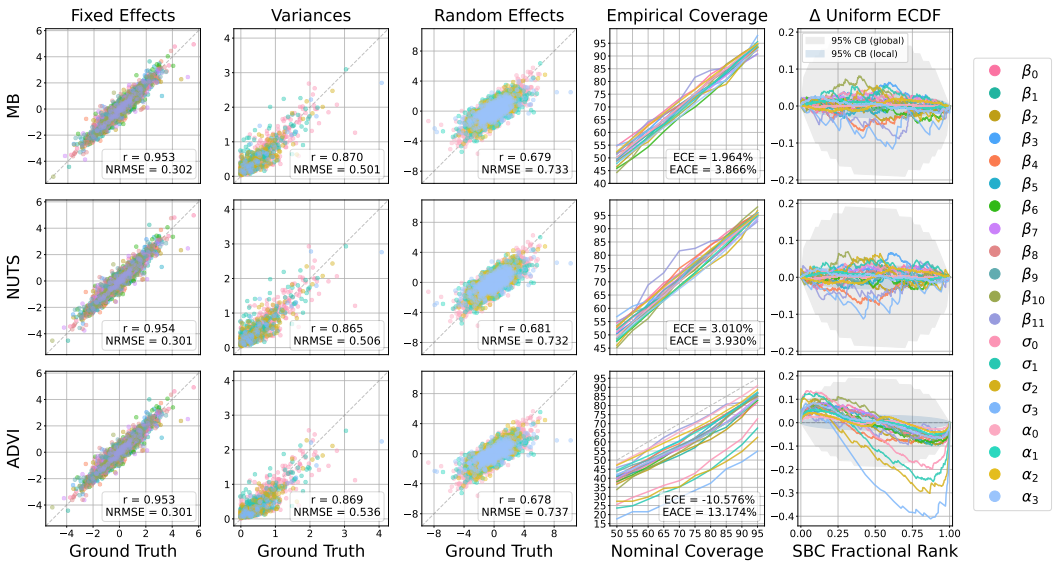


Figure B6: Performance on the *large* Bernoulli oracle benchmark; layout as in Figure 2.

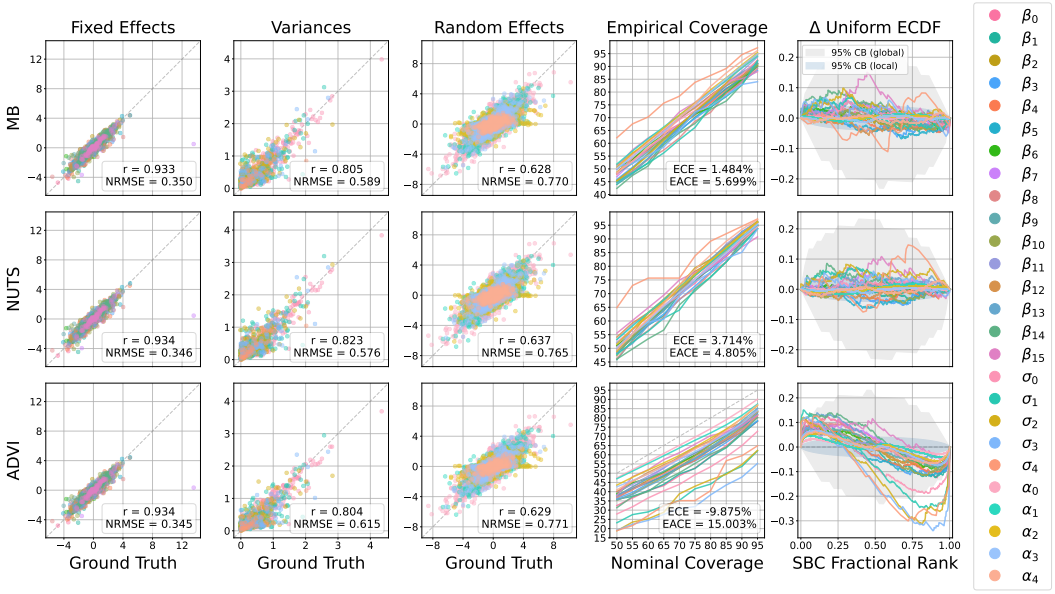


Figure B7: Performance on the *huge* Bernoulli oracle benchmark; layout as in Figure 2.

Table B2: Performance on each Poisson oracle test set averaged over all parameters. Layout as in Table 1.

regime	model	r	NRMSE	ECE	EACE	LOO-NLL	time
small	MB	0.92 ± 0.08	0.35 ± 0.16	0.00 ± 0.04	0.03 ± 0.03	1.34 ± 0.22	0.11 ± 0.00
	NUTS	0.92 ± 0.08	0.35 ± 0.16	0.00 ± 0.04	0.03 ± 0.02	1.34 ± 0.22	63.65 ± 8.41
	ADVI	0.71 ± 0.26	0.80 ± 0.59	-0.18 ± 0.13	0.18 ± 0.13	1.38 ± 0.27	47.73 ± 6.34
medium	MB	0.94 ± 0.06	0.32 ± 0.15	0.00 ± 0.02	0.02 ± 0.01	1.36 ± 0.17	0.11 ± 0.01
	NUTS	0.94 ± 0.06	0.32 ± 0.15	0.00 ± 0.02	0.02 ± 0.01	1.36 ± 0.17	110.71 ± 30.74
	ADVI	0.68 ± 0.28	0.95 ± 0.79	-0.20 ± 0.14	0.20 ± 0.13	1.46 ± 0.27	68.81 ± 11.54
large	MB	0.94 ± 0.07	0.30 ± 0.15	-0.02 ± 0.02	0.03 ± 0.02	1.39 ± 0.16	0.12 ± 0.01
	NUTS	0.94 ± 0.07	0.29 ± 0.16	-0.01 ± 0.02	0.02 ± 0.02	1.39 ± 0.16	97.01 ± 21.32
	ADVI	0.67 ± 0.29	1.05 ± 0.88	-0.21 ± 0.15	0.21 ± 0.15	1.50 ± 0.25	65.90 ± 11.37
huge	MB	0.93 ± 0.09	0.33 ± 0.17	-0.03 ± 0.03	0.04 ± 0.03	1.41 ± 0.14	0.12 ± 0.01
	NUTS	0.93 ± 0.09	0.32 ± 0.18	0.00 ± 0.02	0.02 ± 0.01	1.41 ± 0.14	131.29 ± 31.16
	ADVI	0.65 ± 0.30	1.31 ± 1.52	-0.22 ± 0.14	0.23 ± 0.14	1.55 ± 0.25	95.38 ± 18.53

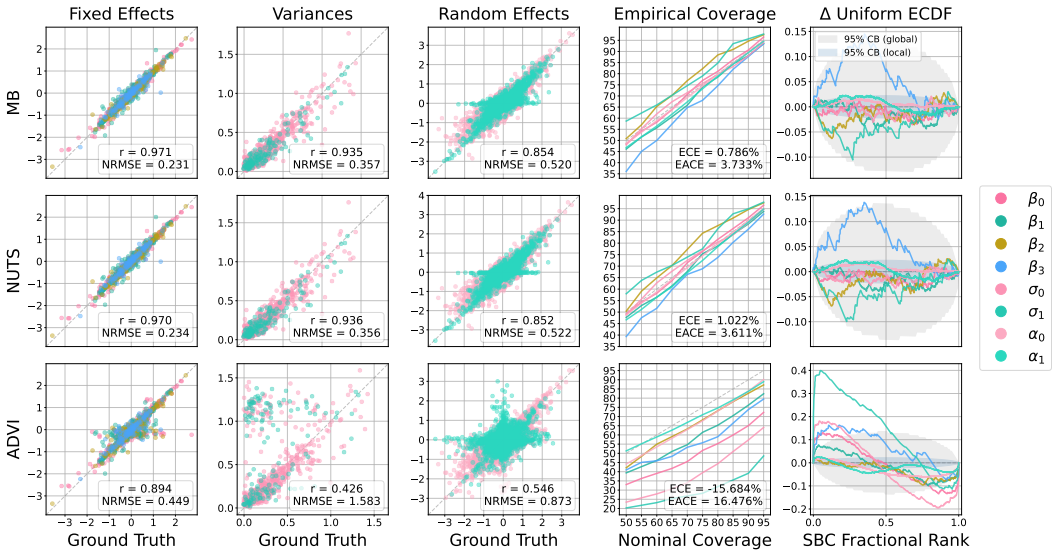


Figure B8: Performance on the *small* Poisson oracle benchmark; layout as in Figure 2.

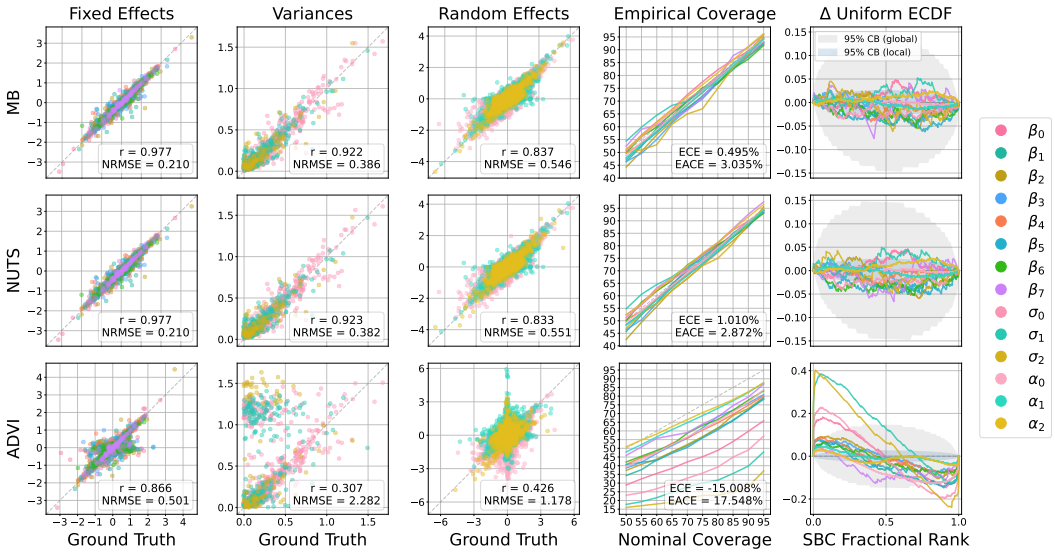


Figure B9: Performance on the *medium* Poisson oracle benchmark; layout as in Figure 2.

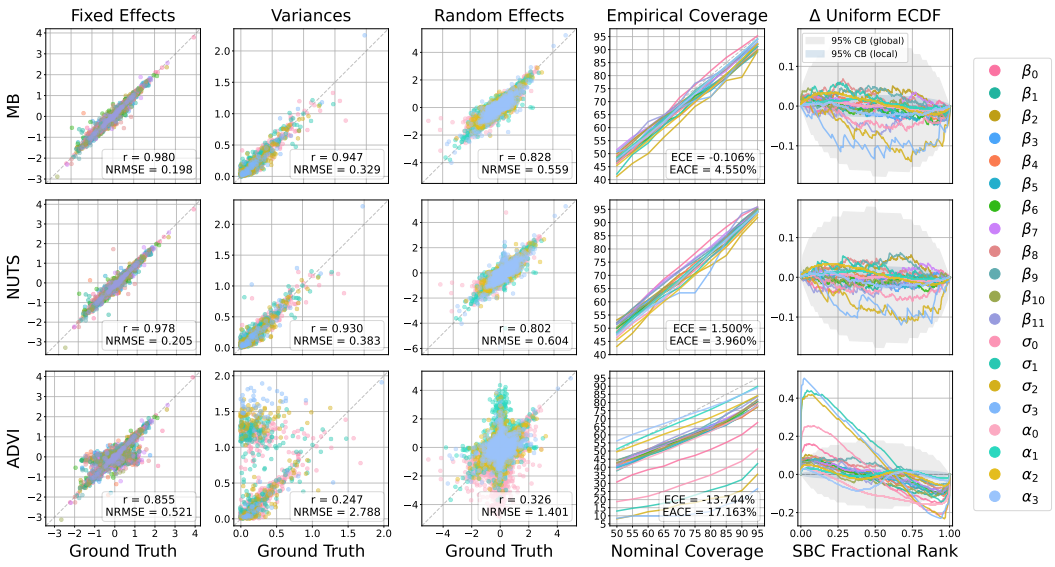


Figure B10: Performance on the *large* Poisson oracle benchmark; layout as in Figure 2.

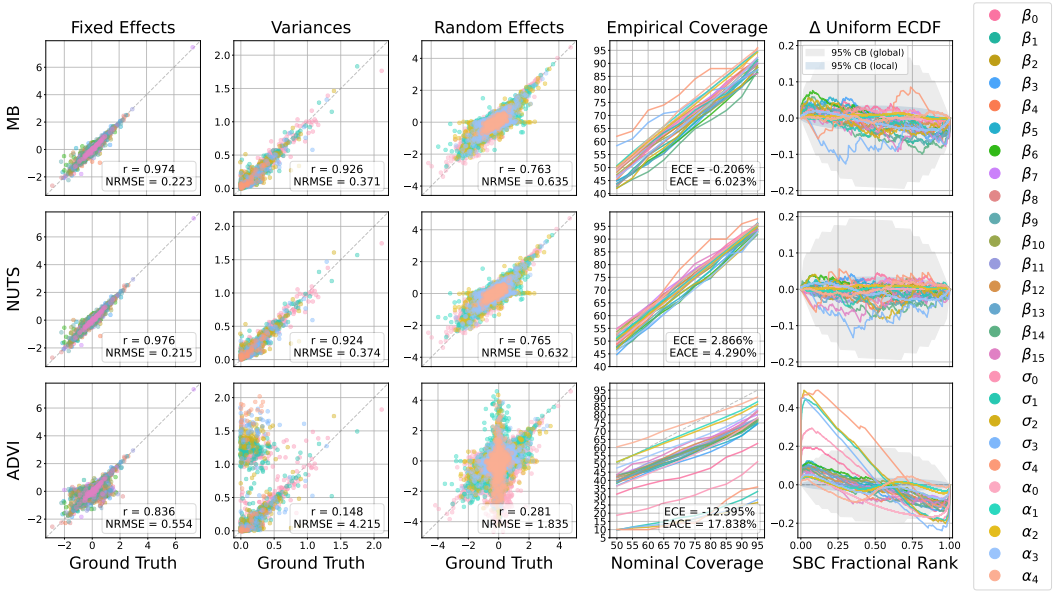


Figure B11: Performance on the *huge* Poisson oracle benchmark; layout as in Figure 2.

B.2 Random-Effect Correlations

Table 1, Table B1 and Table B2 exclude the random-effect correlations ($\mathbf{R}$) from their averages. Table B3 reports them separately for two reasons: (1) The correlations are the least identified GLMM parameters (each is estimated from m noisy pairs of group-level effects), and the up to $q(q-1)/2$ pairs would dominate the dimension-weighted averages of the oracle tables. (2) NUTS notably struggles here: of the 110–200 datasets per test set with correlated random effects (Appendix F.3), NUTS converges on only 60–86% (against 95–100% with independent random effects) under the liberal convergence criterion of Appendix A.2. We restrict the evaluation to the correlated datasets NUTS converged on (81–137 per regime), computing recovery and calibration per correlation pair and reporting mean ± SD over the pairs (a single pair in *small*, no spread possible; up to ten in *huge*). Figure B12 shows the recovery for the *huge* Gaussian benchmark.

All inference methods recover the correlations substantially worse than any other parameter class. On the Gaussian sets, `metabeta` and NUTS reach $r = 0.63$–0.70 and NRMSE 0.73–0.78 (against $r \geq 0.94$ and NRMSE ≤ 0.32 for the parameter average in Table 1). The Poisson sets are similar ($r = 0.53$–0.68), and the Bernoulli sets degrade with size, down to $r = 0.31$ for NUTS in the *huge* regime, where the spread over pairs (± 0.38) already exceeds the mean. Throughout, `metabeta` tracks NUTS within 0.03 in r on the Gaussian and Poisson sets and stays within this pair-wise spread on the Bernoulli sets, sharing the common failure mode of Figure B12: every method shrinks the posterior means toward zero relative to the ground truth. The raw flow ($\texttt{MB}^0$) is the best-calibrated method for the correlations (EACE ≤ 0.08 throughout, ECE between -0.06 and 0.05); the IMH head leaves the correlations close to that calibration, with only a small residual overconfidence in the hardest regime (ECE -0.10 against -0.02 for NUTS in the *huge* Gaussian regime). ADVI remains strongly overconfident on the Gaussian sets (ECE -0.24 to -0.32) and loses the correlations on count data ($r = 0.09$–0.38); LA is worse still (Appendix B.4).

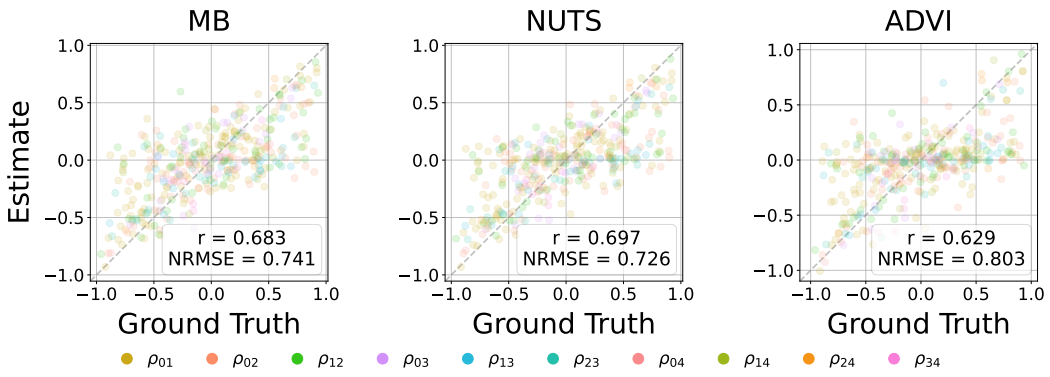


Figure B12: Recovery of the random-effect correlations on the *huge* Gaussian oracle benchmark (correlated, NUTS-converged datasets): posterior mean against ground truth, one color per pair ρ_{jk}. All methods shrink their estimates toward zero; `metabeta` and NUTS are nearly indistinguishable, ADVI loses the large $|\rho|$.

Table B3: Recovery and calibration of the random-effect correlations ρ_{jk} on the oracle test sets (correlated, NUTS-converged datasets; see text). Entries are mean $\pm$ SD over the $q(q-1)/2$ correlation pairs; metrics as in Table 1; $\texttt{MB}^0$ is the raw flow without the IMH head (Appendix C.1).

family	regime	model	r	NRMSE	ECE	EACE
Gaussian	small	$\texttt{MB}^0$	0.67 ± 0.00	0.74 ± 0.00	-0.03 ± 0.00	0.03 ± 0.00
		MB	0.68 ± 0.00	0.73 ± 0.00	-0.03 ± 0.00	0.03 ± 0.00
		NUTS	0.68 ± 0.00	0.74 ± 0.00	-0.03 ± 0.00	0.03 ± 0.00
		ADVI	0.64 ± 0.00	0.78 ± 0.00	-0.28 ± 0.00	0.28 ± 0.00
	medium	$\texttt{MB}^0$	0.67 ± 0.04	0.75 ± 0.04	0.00 ± 0.04	0.04 ± 0.02
		MB	0.69 ± 0.05	0.73 ± 0.06	-0.06 ± 0.03	0.06 ± 0.03
		NUTS	0.69 ± 0.07	0.73 ± 0.08	-0.04 ± 0.01	0.04 ± 0.02
		ADVI	0.63 ± 0.06	0.78 ± 0.05	-0.32 ± 0.03	0.32 ± 0.03
	large	$\texttt{MB}^0$	0.64 ± 0.12	0.78 ± 0.07	-0.01 ± 0.06	0.06 ± 0.03
		MB	0.63 ± 0.09	0.78 ± 0.07	-0.07 ± 0.08	0.08 ± 0.06
		NUTS	0.65 ± 0.07	0.77 ± 0.05	-0.05 ± 0.08	0.08 ± 0.05
		ADVI	0.53 ± 0.07	0.88 ± 0.06	-0.29 ± 0.06	0.29 ± 0.06
	huge	$\texttt{MB}^0$	0.68 ± 0.10	0.74 ± 0.09	0.02 ± 0.09	0.08 ± 0.06
		MB	0.68 ± 0.11	0.74 ± 0.11	-0.10 ± 0.13	0.14 ± 0.07
		NUTS	0.70 ± 0.10	0.73 ± 0.08	-0.02 ± 0.11	0.10 ± 0.05
		ADVI	0.63 ± 0.10	0.80 ± 0.13	-0.24 ± 0.11	0.24 ± 0.10
Bernoulli	small	$\texttt{MB}^0$	0.55 ± 0.00	0.83 ± 0.00	0.05 ± 0.00	0.05 ± 0.00
		MB	0.59 ± 0.00	0.81 ± 0.00	0.05 ± 0.00	0.05 ± 0.00
		NUTS	0.60 ± 0.00	0.81 ± 0.00	0.05 ± 0.00	0.05 ± 0.00
		ADVI	0.56 ± 0.00	0.84 ± 0.00	-0.13 ± 0.00	0.13 ± 0.00
	medium	$\texttt{MB}^0$	0.39 ± 0.12	0.93 ± 0.03	-0.01 ± 0.07	0.05 ± 0.04
		MB	0.41 ± 0.12	0.93 ± 0.03	-0.06 ± 0.09	0.07 ± 0.07
		NUTS	0.41 ± 0.12	0.93 ± 0.03	-0.06 ± 0.10	0.08 ± 0.08
		ADVI	0.34 ± 0.11	0.95 ± 0.03	-0.15 ± 0.09	0.15 ± 0.09
	large	$\texttt{MB}^0$	0.40 ± 0.16	0.91 ± 0.07	0.00 ± 0.08	0.07 ± 0.03
		MB	0.47 ± 0.14	0.88 ± 0.06	-0.05 ± 0.07	0.08 ± 0.03
		NUTS	0.46 ± 0.19	0.89 ± 0.07	-0.05 ± 0.08	0.08 ± 0.05
		ADVI	0.45 ± 0.14	0.92 ± 0.05	-0.18 ± 0.08	0.18 ± 0.08
	huge	$\texttt{MB}^0$	0.28 ± 0.20	0.99 ± 0.07	-0.02 ± 0.08	0.08 ± 0.03
		MB	0.24 ± 0.38	0.99 ± 0.11	-0.08 ± 0.06	0.09 ± 0.05
		NUTS	0.31 ± 0.28	0.97 ± 0.09	-0.06 ± 0.06	0.08 ± 0.05
		ADVI	0.48 ± 0.24	0.99 ± 0.11	-0.14 ± 0.09	0.15 ± 0.08
Poisson	small	$\texttt{MB}^0$	0.65 ± 0.00	0.77 ± 0.00	0.02 ± 0.00	0.02 ± 0.00
		MB	0.65 ± 0.00	0.77 ± 0.00	0.01 ± 0.00	0.01 ± 0.00
		NUTS	0.65 ± 0.00	0.77 ± 0.00	0.00 ± 0.00	0.01 ± 0.00
		ADVI	0.38 ± 0.00	0.96 ± 0.00	-0.06 ± 0.00	0.06 ± 0.00
	medium	$\texttt{MB}^0$	0.50 ± 0.18	0.86 ± 0.10	-0.06 ± 0.10	0.07 ± 0.09
		MB	0.53 ± 0.14	0.85 ± 0.08	-0.09 ± 0.11	0.09 ± 0.11
		NUTS	0.54 ± 0.11	0.85 ± 0.06	-0.10 ± 0.10	0.10 ± 0.10
		ADVI	0.12 ± 0.19	1.03 ± 0.07	-0.14 ± 0.11	0.14 ± 0.11
	large	$\texttt{MB}^0$	0.65 ± 0.10	0.76 ± 0.10	0.04 ± 0.05	0.05 ± 0.04
		MB	0.68 ± 0.12	0.73 ± 0.12	-0.01 ± 0.04	0.05 ± 0.01
		NUTS	0.67 ± 0.12	0.74 ± 0.11	0.00 ± 0.04	0.04 ± 0.02
		ADVI	0.09 ± 0.16	1.03 ± 0.06	-0.07 ± 0.06	0.08 ± 0.06
	huge	$\texttt{MB}^0$	0.50 ± 0.14	0.90 ± 0.10	0.01 ± 0.05	0.06 ± 0.02
		MB	0.57 ± 0.13	0.87 ± 0.10	-0.04 ± 0.04	0.07 ± 0.02
		NUTS	0.58 ± 0.11	0.86 ± 0.09	-0.04 ± 0.05	0.06 ± 0.03
		ADVI	0.16 ± 0.17	1.02 ± 0.07	-0.07 ± 0.07	0.09 ± 0.04

B.3 Real-World Benchmark

Table B4, Table B5 and Table B6 report the real-world agreement with NUTS for the Gaussian, Bernoulli and Poisson families (Section 3.2).

Table B4: Posterior agreement on each Gaussian real-world test set aggregated over all parameters. Entries are median ± MAD over datasets. Regime descriptions are in Section 2.4.

regime	model	r	σ-ratio	rank-MAD	ΔLOO-NLL	Δtime
small	MB	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	-60.31 ± 5.33
	ADVI	1.00 ± 0.00	0.89 ± 0.07	0.03 ± 0.01	0.00 ± 0.00	-13.73 ± 4.82
medium	MB	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	-74.51 ± 12.37
	ADVI	1.00 ± 0.00	0.86 ± 0.09	0.03 ± 0.01	0.00 ± 0.00	-21.20 ± 10.28
large	MB	1.00 ± 0.00	1.00 ± 0.01	0.01 ± 0.00	0.00 ± 0.00	-82.11 ± 17.46
	ADVI	1.00 ± 0.00	0.83 ± 0.09	0.04 ± 0.01	0.00 ± 0.00	-26.33 ± 12.33
huge	MB	1.00 ± 0.00	1.00 ± 0.02	0.01 ± 0.00	0.00 ± 0.00	-82.21 ± 10.67
	ADVI	1.00 ± 0.00	0.88 ± 0.07	0.03 ± 0.01	0.00 ± 0.00	-27.37 ± 9.02

Table B5: Posterior agreement on each Bernoulli real-world test set aggregated over all parameters. Layout as in Table B4.

regime	model	r	σ-ratio	rank-MAD	ΔLOO-NLL	Δtime
small	MB	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	-61.12 ± 7.12
	ADVI	1.00 ± 0.00	0.87 ± 0.07	0.03 ± 0.01	0.00 ± 0.00	-16.13 ± 5.30
medium	MB	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	-76.60 ± 14.76
	ADVI	0.99 ± 0.01	0.80 ± 0.11	0.04 ± 0.02	0.00 ± 0.00	-23.56 ± 8.96
large	MB	1.00 ± 0.00	1.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	-80.49 ± 14.09
	ADVI	1.00 ± 0.00	0.85 ± 0.09	0.03 ± 0.01	0.00 ± 0.00	-28.04 ± 11.12
huge	MB	1.00 ± 0.00	0.99 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	-91.83 ± 18.69
	ADVI	0.99 ± 0.01	0.83 ± 0.09	0.04 ± 0.01	0.00 ± 0.00	-36.44 ± 10.80

Table B6: Posterior agreement on each Poisson real-world test set aggregated over all parameters. Layout as in Table B4. ADVI's large ΔLOO-NLL values reflect fits that did not converge within the iteration budget of Appendix A.2: 32%, 41% and 45% of the *small*, *medium* and *large* fits exhaust the 10^5 iterations without meeting the ELBO plateau criterion ($\leq$2% on Gaussian and Bernoulli real-world data), and the affected posteriors are far too wide (σ-ratios up to $\approx$4.7). No *huge* regime exists for count outcomes: no real hierarchical datasets with sufficiently many predictors were found.

regime	model	r	σ-ratio	rank-MAD	ΔLOO-NLL	Δtime
small	MB	1.00 ± 0.00	1.00 ± 0.01	0.01 ± 0.00	0.00 ± 0.00	-67.44 ± 11.85
	ADVI	0.27 ± 0.17	4.71 ± 3.96	0.13 ± 0.05	5279.69 ± 5218.54	-10.38 ± 9.34
medium	MB	1.00 ± 0.00	0.98 ± 0.03	0.01 ± 0.00	0.00 ± 0.00	-86.67 ± 18.44
	ADVI	0.65 ± 0.32	2.07 ± 1.37	0.13 ± 0.05	596.35 ± 596.34	-21.95 ± 13.36
large	MB	1.00 ± 0.00	0.96 ± 0.03	0.01 ± 0.01	0.00 ± 0.00	-97.08 ± 22.39
	ADVI	0.63 ± 0.34	1.62 ± 1.02	0.11 ± 0.03	236.60 ± 236.59	-27.63 ± 12.32

B.4 Laplace Baseline

The Laplace approximation (LA; implementation in Appendix A.2) is the classical fast approximate inference method, so we compare it on the oracle benchmarks with recovery broken down by parameter class. Table B7 confirms the standard intuition only for fixed effects: LA's $\boldsymbol{\beta}$ estimates are usable in every family (NRMSE within 0.05 of NUTS). Everything else fails. Variance components are off by up to an order of magnitude (NRMSE 2.89 Gaussian, 10.64 Bernoulli, 6.12 Poisson, against $\leq$0.49 for `metabeta`), random effects follow (0.82–2.46), and predictive accuracy is the worst of all methods in every family.

The failures are structural rather than a tuning issue: the posterior of a variance component near zero is strongly skewed, and a Gaussian centered at the joint MAP places substantial mass at negative (transformed) scales, which then corrupts the random-effect conditional. Figure B13 makes the failure mode visible on the *medium* Bernoulli benchmark: LA's variance estimates collapse toward zero regardless of the ground truth, the credible intervals of the variance parameters cover far below their nominal level, and the SBC ranks pile up at the extremes. The default `metabeta` pipeline runs at Laplace-class latency: 1.2–5.1$\times$ faster than LA on a GPU and within 2–10$\times$ of it on the same four CPU cores (Appendix B.5, Table B8). `metabeta` matches NUTS on every parameter class, so LA is not competitive even in the large-data regime it is usually recommended for.

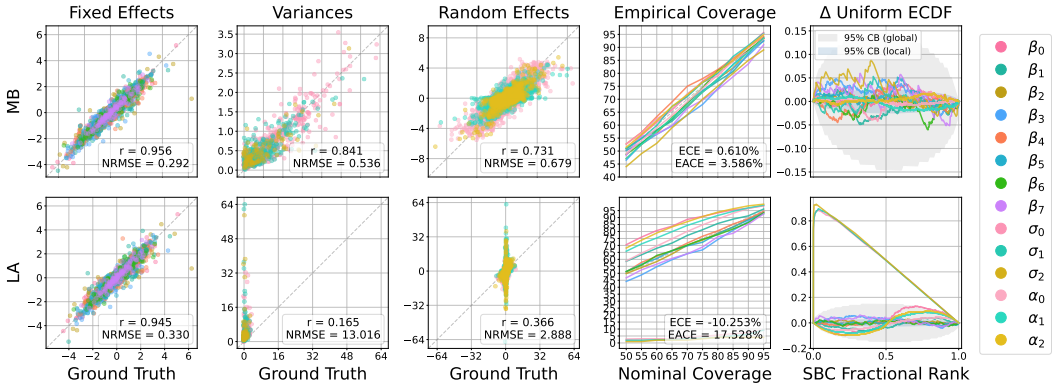


Figure B13: Performance of the default `metabeta` pipeline (`MB`) and the Laplace approximation (`LA`) on the *medium* Bernoulli oracle benchmark; layout as in Figure 2. `LA` recovers the fixed effects but collapses the variance components toward zero irrespective of the ground truth, which corrupts the random-effect conditional, the coverage of the affected parameters, and the SBC ranks.

B.5 Runtime Results

Setup. Runtimes were measured on the twelve oracle benchmarks (three likelihood families $\times$ four size regimes, 512 datasets each). `metabeta` draws 4,000 posterior samples per dataset, matching NUTS (4 chains $\times$ 1,000 draws); the IMH head runs 4 chains of 1,000 steps on these draws. NUTS, ADVI and LA run via `PyMC` on four Intel Xeon Gold 6248R CPU cores, as in all other experiments; their times are recorded during the main evaluation runs (ADVI excluding failed fits). `metabeta` is timed on one NVIDIA H100 GPU and on the same four CPU cores. Every dataset is processed separately (batch of one) after a warm-up pass. We use device synchronization around the timed region. Two savings available in deployment, batching several datasets per pass and caching the analytical statistics, are not used.

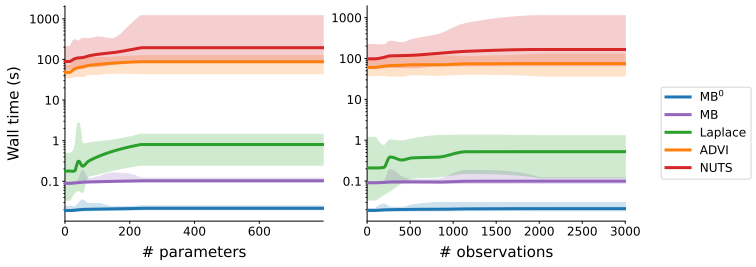
What is timed. `MB` is the default pipeline: one forward pass through the summary networks and the global posterior network $\boldsymbol{\Pi}_g$, which draws 4,000 global samples, followed by the IMH head which reimputes the random effects from their analytical conditional (Section 2.3). The IMH head is listed separately for comparison. $\texttt{MB}^0$ is the raw flow with refinement switched off; for the discrete families it draws the random effects from the local posterior network $\boldsymbol{\Pi}_l$, which the default pipeline never evaluates (Gaussian $\texttt{MB}^0$ uses the closed-form conditional instead, Appendix E).

Results. Table B8 and Figure B14 give the breakdown. Three take-aways follow:

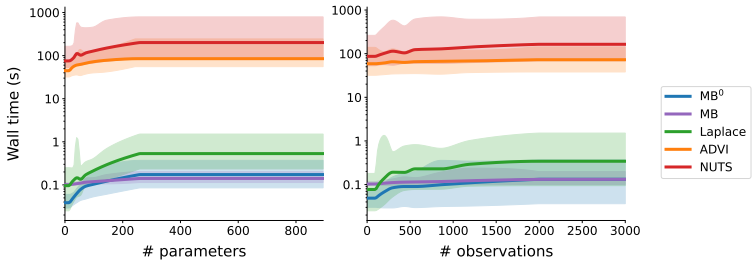
1. **The IMH head permits a shortcut around the local posterior network**. `metabeta` returns a full posterior in 0.09–0.12 s per dataset on the GPU and 0.54–2.3 s on four CPU cores. The $\mathbf{\Pi}_g$ (global posterior) pass costs 0.02 s (GPU) resp. 0.2–0.5 s (CPU) in every family. The IMH head adds 0.07–0.10 s resp. 0.2–1.8 s. The head is also what makes the global pass sufficient: by marginalizing the random effects out, it avoids $\mathbf{\Pi}_l$, whose evaluation per group and draw makes $\texttt{MB}^0$ 4–8× slower than `MB` on CPU for Bernoulli and Poisson (4.3–10.6 s).
2. **Wall-clock time scales more favorably for `metabeta`**. The flow pass grows by at most 13% (GPU) and 80% (CPU) from *small* to *huge*; only the head's per-group work on CPU grows significantly with the group count (2–3×). The 95th percentile of `MB` stays within 1.5× its median on GPU, where the slowest of all 6,144 datasets took 0.32 s. The 95th percentile is 0.8–6.2 s on CPU. NUTS' 95th percentile is 1.7–4.0× its median; its slowest dataset took 3,127 s.
3. **`metabeta` works best on GPU and remains useful on CPU**. At the median, `MB` is 600–1,400× faster than NUTS on the GPU and 50–210× faster on the baselines' own CPU cores; ADVI is only 1.3–2.0× faster than NUTS. Against LA, `MB` is 1.2–5.1× faster on the GPU and within 2–10× on the CPU. Counting only converged NUTS runs (strict criterion, Appendix A.2) widens the gap, since the slowest runs converge least often: 93–256 s per usable NUTS posterior, 900–2,200× (GPU) and 90–300× (CPU) the cost of `MB`.

Table B7: Recovery by parameter class on the oracle benchmarks, averaged over the four size regimes (NUTS-converged datasets, as in Table 1). Entries are NRMSE (r in parentheses) for fixed effects ($\boldsymbol{\beta}$), variance components ($\boldsymbol{\sigma}$) and random effects ($\boldsymbol{\alpha}$), median per-dataset LOO-NLL, and median wall-clock time in seconds per dataset (`MB` on one GPU, the baselines on four CPU cores; Table B8). `MB` is the default pipeline; LOO-NLL and time are the regime averages of Table 1, Table B1 and Table B2. Bold marks the salient failures.

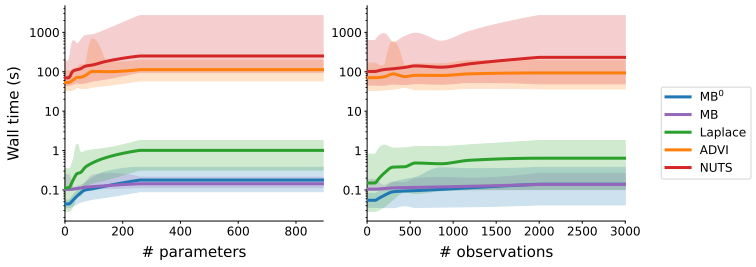
family	model	$\boldsymbol{\beta}$: NRMSE (r)	$\boldsymbol{\sigma}$: NRMSE (r)	$\boldsymbol{\alpha}$: NRMSE (r)	LOO-NLL	time [s]
Gaussian	`MB`	0.28 (0.96)	0.22 (0.97)	0.44 (0.90)	2.25	0.09
	`NUTS`	0.28 (0.96)	0.22 (0.97)	0.44 (0.90)	2.25	106
	`ADVI`	0.28 (0.96)	0.25 (0.97)	0.45 (0.89)	2.27	64.1
	`LA`	0.31 (0.95)	**2.89** (0.77)	**0.82** (0.75)	**2.96**	0.25
Bernoulli	`MB`	0.31 (0.95)	0.49 (0.88)	0.70 (0.71)	0.46	0.11
	`NUTS`	0.31 (0.95)	0.49 (0.87)	0.70 (0.71)	0.46	96
	`ADVI`	0.31 (0.95)	0.53 (0.87)	0.70 (0.70)	0.47	60.9
	`LA`	0.35 (0.94)	**10.64** (0.27)	**2.46** (0.40)	**0.60**	0.16
Poisson	`MB`	0.21 (0.98)	0.36 (0.93)	0.58 (0.81)	1.38	0.11
	`NUTS`	0.21 (0.98)	0.36 (0.93)	0.58 (0.81)	1.37	101
	`ADVI`	0.45 (0.89)	**2.44** (0.33)	**1.27** (0.43)	1.47	69.5
	`LA`	0.23 (0.97)	**6.12** (0.45)	**1.43** (0.56)	**2.48**	0.31



(a) Gaussian



(b) Bernoulli



(c) Poisson

Figure B14: Wall-clock time per dataset vs. model complexity ($n_{\text{params}} = d + q + mq$, left) and total observations (n, right), pooled over the four size regimes of each likelihood family. Lines show binned means; bands span the 0.5th–99.5th percentile. `metabeta` appears as the raw flow (`MB`^{0}) and the default pipeline (`MB`), one dataset per forward pass on the GPU; `NUTS`, `ADVI` and `LA` as recorded during the main evaluation runs on four CPU cores. Note the log ordinate: `MB` is flat in both panels because its cost tracks the architecture, whereas the sampler baselines track the data; `MB`^{0} of the discrete families inherits a mild dependence on the group count from the local posterior network.

Table B8: Median wall-clock time in seconds per dataset (512 datasets per cell, one dataset per forward pass), for `metabeta` on one H100 GPU (left) and on the same four Xeon CPU cores as the baselines (right). `MB` is the default pipeline; its IMH head is listed separately. $\texttt{MB}^0$ is the raw flow. `LA`, `ADVI` and `NUTS` run on the CPU only.

	H100 GPU				4 CPU cores			
method	*small*	*medium*	*large*	*huge*	*small*	*medium*	*large*	*huge*
Gaussian								
MB	0.090	0.093	0.095	0.098	0.539	0.605	0.704	0.857
IMH only	0.074	0.076	0.078	0.081	0.196	0.218	0.280	0.361
$\texttt{MB}^0$	0.019	0.020	0.020	0.021	0.371	0.458	0.512	0.629
LA	–	–	–	–	0.158	0.240	0.312	0.394
ADVI	–	–	–	–	47.1	64.5	68.6	83.7
NUTS	–	–	–	–	77.0	126	112	125
Bernoulli								
MB	0.104	0.108	0.114	0.118	0.781	1.03	1.34	2.21
IMH only	0.086	0.090	0.096	0.099	0.552	0.749	1.01	1.79
$\texttt{MB}^0$	0.048	0.068	0.086	0.102	4.26	6.78	10.0	10.3
LA	–	–	–	–	0.129	0.141	0.180	0.221
ADVI	–	–	–	–	44.2	61.1	65.4	80.6
NUTS	–	–	–	–	74.2	107	95.0	129
Poisson								
MB	0.105	0.109	0.116	0.120	1.15	1.55	2.01	2.26
IMH only	0.087	0.091	0.098	0.102	0.870	1.22	1.64	1.82
$\texttt{MB}^0$	0.054	0.073	0.093	0.106	4.39	7.47	9.24	10.6
LA	–	–	–	–	0.137	0.230	0.406	0.615
ADVI	–	–	–	–	48.3	71.3	71.0	101
NUTS	–	–	–	–	64.9	117	107	143

B.6 Marginal Likelihood from Importance Weights

Every marginal importance weight $w_s = p(\mathbf{D} \mid \boldsymbol{\vartheta}_s)\, p(\boldsymbol{\vartheta}_s)/q(\boldsymbol{\vartheta}_s)$ divides the unnormalized posterior by the flow density, so $\mathbb{E}_q[w] = p(\mathbf{D})$ (Section 3.4) and $\hat{p}(\mathbf{D}) = S^{-1} \sum_s w_s$ is an unbiased estimate of the marginal likelihood (Kass & Raftery, 1995). The identity rests on three requirements:

1. the proposal q must be normalized, which the flow guarantees by construction;
2. the numerator must retain every normalizing constant of likelihood and prior: the analytical marginalization of the random effects provides this for Gaussian outcomes, and the unbiased IS^2 estimate of Appendix C.1 for Bernoulli and Poisson outcomes;
3. q must cover the support of the posterior, which the mass-covering forward-KL objective favors.

We validated the estimator on all four Gaussian oracle sets against iterative bridge sampling (Meng & Wong, 1996; Gronau et al., 2017) on the NUTS draws (warped-Gaussian proposal fitted on one half of the draws and evaluated on the other; swapping the halves changes the reference by 0.006–0.012 nats in the median). Table B9 reports the agreement for pool sizes $S = 1000$ and $S = 4000$, the share of datasets whose PSIS shape parameter exceeds 0.7, and a nested comparison in which every dataset with a random slope is also fit with a random intercept only.

Table B9: Marginal likelihood from importance weights vs. bridge sampling on `NUTS` draws, on each Gaussian oracle test set (512 datasets per regime, of which #corr have correlated random effects). $|\Delta| = |\Delta \log p(\mathbf{D})|$ is the absolute difference between the two estimates (median and 90% quantile over datasets); frac $\hat{k} > 0.7$ is the share of datasets flagged by the PSIS shape parameter; nested: number of datasets with a random slope that were additionally fit without it, share of them whose $\ln \mathrm{BF}$ (slope vs. no slope) falls into the same Jeffreys category as the reference, and median $|\Delta \ln \mathrm{BF}|$.

regime	#ds	#corr	S	med $\lvert\Delta\rvert$	q_{90} $\lvert\Delta\rvert$	frac $\hat{k} > 0.7$	#nested	BF cat. agree	med $\lvert\Delta \ln \mathrm{BF}\rvert$
`small`	512	113	1000	0.012	0.041	0.01			
			4000	0.008	0.028	0.00	137	1.00	0.016
`medium`	512	187	1000	0.028	0.092	0.05			
			4000	0.016	0.058	0.02	259	0.97	0.033
`large`	512	189	1000	0.055	0.266	0.16			
			4000	0.031	0.163	0.11	294	0.98	0.068
`huge`	512	200	1000	0.112	1.035	0.39			
			4000	0.067	0.653	0.34	319	0.94	0.163

Continuous outcomes. On *small* and *medium* the estimate sits at the reference's noise floor (0.008 and 0.016 nats median error at $S = 4000$), and the Bayes factor from two batch elements of one forward pass falls into the same Jeffreys category as the bridge reference on 97–100% of datasets. The error grows with model size (0.031 and 0.067 nats on *large* and *huge*) together with the flagged share (11% and 34% of datasets with $\hat{k} > 0.7$). Datasets with $\hat{k} \leq 0.5$ stay within 0.03 nats in every regime, whereas flagged datasets carry median errors of 0.2–0.3 nats. They account for nearly all discrepancies above one nat, so the PSIS diagnostic that `metabeta` already reports identifies the estimates that should be viewed with caution. Category agreement follows the same pattern (98% on *large*, 94% on *huge*, against 98% between two bridge estimates), and the median error shrinks by about 40% from $S = 1000$ to $S = 4000$ in every regime. Of the 1,008 nested comparisons, 31 land in a different Jeffreys category than the reference; all 31 are either flagged by $\hat{k} > 0.7$ in the full or the reduced fit or have a reference $\ln \mathrm{BF}$ within 0.2 nats of a category boundary (28 of 31 when only the full fit's $\hat{k}$ counts). A disagreement is either announced by the diagnostic or a borderline call.

Discrete outcomes (preliminary). With the pseudo-marginal weights of Appendix C.1, the same estimator applies to Bernoulli and Poisson outcomes: an unbiased estimate of the likelihood inside an unbiased average keeps $\hat{p}(\mathbf{D})$ unbiased. As the reference, we run bridge sampling on the NUTS draws of the global parameters, with the marginal likelihood evaluated by adaptive Gauss–Hermite quadrature (20, 12, 5, 4 and 3 nodes per dimension for $q = 1, \ldots, 5$ random effects). It converges

geometrically in the number of nodes. A preliminary analysis on *small* and *medium* Bernoulli oracle sets shows that the estimate agrees with the reference to a median of 0.010 and 0.017 nats at $S = 4000$ (90% quantile 0.036 and 0.056). This is close to the noise floor of the reference (swapped halves: 0.009 and 0.011 nats), and only 2.5% and 2.8% of datasets have $\hat{k} > 0.7$. In the nested comparison (random slope vs. intercept only; 128 and 109 datasets), every Bayes factor falls into the same Jeffreys category as the reference (median $|\Delta \ln \mathrm{BF}| = 0.026$ and 0.036). The seven comparisons off by more than 0.5 nats all have a reference $\ln \mathrm{BF} > 8$ (decisive either way) and are flagged by $\hat{k} > 0.7$ in the intercept-only fit.

B.7 Zero-Shot Prior Sensitivity Analysis

Prior conditioning promises a prior sensitivity analysis in a single batched forward pass (Section 3.5). We test this promise on real data against NUTS, which is refit at every prior.

Setup.

- **Data.** One held-out dataset per likelihood family (Table B10; Bates et al., 2015; Brooks et al., 2017), each with a correlated random intercept and slope: `Reaction ~ Days + (1 + Days | Subject)`, `incidence ~ period + (1 + period4 | herd)` and `count ~ mined + wtemp + (1 + wtemp | site)`. The *key effects* are `Days`, `period4` and `mined`.
- **Prior grid.** Twelve log-spaced Normal slope-prior scales $\tau_\beta \in [0.05, 5]$ crossed with half-Normal, half-Student-t_5 and Exponential priors on the random-effect SDs at scales $\tau_\sigma \in \{0.25, 1, 2.5, 5\}$; all other priors at the Bambi defaults. Priors above a family's training hyper-prior range (Appendix F.3) are dropped, leaving 132, 60 and 54 priors.
- **Methods.** `metabeta` (4,000 draws per prior, one batched call per SD-prior family) and the raw flow `MB`^{0}. NUTS (four chains, 2,000 warm-up and 1,000 draws) refits every grid point.
- **Disagreement flag.** The posterior median of the key effect shifts by more than 0.25 SDs of the NUTS posterior (about three Monte Carlo standard errors of the difference of two medians) or the 90%-interval width ratio leaves $[0.8, 1.25]$. Only priors at which NUTS passes the liberal convergence filter of Appendix A.2 are compared (219 of 246).

Table B10: Prior-sensitivity datasets, grid sizes and reference cost. conv.: priors at which `NUTS` passes the liberal convergence filter. `NUTS`: median seconds per prior with four chains on four Xeon CPU cores and core-hours for the full grid. `metabeta` runs each full grid in 1.6–2.8 s on one H100 GPU, IMH included.

dataset	m	n	priors grid	priors conv.	NUTS per prior [s]	NUTS full grid [core-h]
`sleepstudy` (Gaussian)	18	180	132	132	97	16.9
`cbpp` (Bernoulli)	15	842	60	42	106	7.1
`Salamanders` (Poisson)	23	644	54	45	124	7.6

Results. Table B11 gives the agreement over the 219 converged priors, Figure B15 shows prior sensitivity for the key effect. Three take-aways:

1. **The grid is interactive with `metabeta`.** Each grid takes 1.6–2.8 s on one GPU against 7–17 core-hours for NUTS (Table B10).
2. **`metabeta` reproduces NUTS across the whole grid.** On every dataset $r = 1.00$, σ-ratio 0.99–1.00, and the key effect's posterior median lies within 0.03–0.04 NUTS SDs of the reference. The flag fires at 14 of 219 priors for `metabeta`, against 73 for the raw flow.
3. **Where `metabeta` is off, it usually says so.** Nine of the 14 flags are the two tightest slope priors of `sleepstudy` ($\tau_\beta \leq 0.076$), where the posterior comes out too narrow and IMH acceptance collapses to 0.02–0.13 (the pale markers in Figure B15). Ten of the 14 are announced at deployment by acceptance below 0.2 or $\hat{k} > 0.7$. The rest are at the Monte Carlo noise floor: 10 of 14 flagged shifts lie within three standard errors, two reruns with new seeds flag 12 and 7 priors, and of the five priors flagged in all three runs, four fire a diagnostic.

The same forward pass yields the marginal likelihood (Appendix B.6): on `sleepstudy` it matches bridge sampling on the NUTS draws within 0.021 nats at the median over 42 reference priors (bridge noise floor 0.007), with the two errors above one nat flagged by $\hat{k} > 1$. Figure B16 shows the resulting Bayes-factor curve: the data prefer $\tau_\beta \approx 0.4$–0.6 over the Bambi-like default ($\ln \mathrm{BF} = +0.9$) and reject $\tau_\beta = 0.05$ ($\ln \mathrm{BF} \leq -6$). For `metabeta` this means that a full prior sensitivity analysis, including the evidence, is a two-second call whose failures are almost always announced by the diagnostics it already reports.

Table B11: Agreement with `NUTS` over all grid priors at which `NUTS` converged, median ± MAD: r, σ-ratio, rank-MAD and ΔLOO-NLL over all parameters, the median shift of the key effect in NUTS posterior SDs, the number of flagged priors, and the IMH acceptance of `MB`.

dataset	model	r	σ-ratio	rank-MAD	ΔLOO-NLL	$\lvert\Delta\text{med}\rvert/\text{sd}$	flagged	IMH acc.
`sleepstudy`	`MB`	1.00 ± 0.00	1.00 ± 0.01	0.01 ± 0.00	0.00 ± 0.00	0.03 ± 0.02	$9/132$	0.74 ± 0.07
	`MB`^{0}	1.00 ± 0.00	1.03 ± 0.01	0.01 ± 0.01	0.00 ± 0.00	0.09 ± 0.07	$36/132$	–
`cbpp`	`MB`	1.00 ± 0.00	0.99 ± 0.01	0.00 ± 0.00	0.00 ± 0.00	0.04 ± 0.02	$0/42$	0.61 ± 0.05
	`MB`^{0}	0.99 ± 0.00	0.93 ± 0.01	0.02 ± 0.00	0.00 ± 0.00	0.08 ± 0.07	$8/42$	–
`Salamanders`	`MB`	1.00 ± 0.00	0.99 ± 0.02	0.00 ± 0.00	0.00 ± 0.00	0.04 ± 0.03	$5/45$	0.48 ± 0.09
	`MB`^{0}	1.00 ± 0.00	1.02 ± 0.07	0.02 ± 0.02	0.01 ± 0.00	0.13 ± 0.08	$29/45$	–

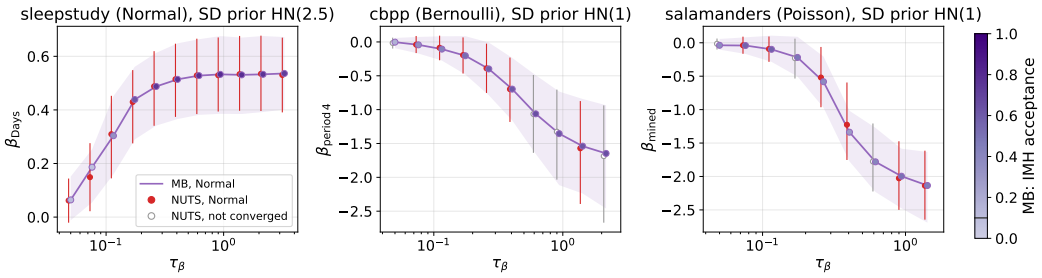


Figure B15: Prior sensitivity in one pass: posterior median and 90% interval of the key effect against the slope-prior scale τ_β, with the random-effect SD prior fixed at half-Normal(2.5) (`sleepstudy`) or half-Normal(1). `metabeta` as lines and bands, markers shaded by IMH acceptance (the colorbar line marks the warning threshold 0.1); NUTS as red markers, hollow where its convergence checks fail. The Bayes-factor curve is in Figure B16.

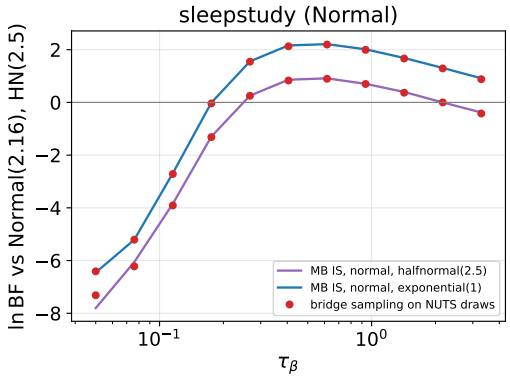


Figure B16: `sleepstudy`: $\ln \text{BF}$ of each slope prior against the Bambi-like reference (Normal, $\tau_\beta = 2.16$, half-Normal(2.5) SD prior) from the importance weights of one forward pass (lines), and from bridge sampling on the NUTS draws (markers).

B.8 PRIOR-AWARENESS CHECK

A model that ignored the prior-family encoding could still appear calibrated on average, so we test its use directly: holding the data and all other hyperparameters fixed, we overwrite one family index across the whole test set, resample the posterior, and compare posterior standard deviations per parameter entry. Families are ordered by their prior SD in units of the scale hyperparameter τ; the ratio of prior SDs (*prior ratio*) is the ceiling the posterior SD-ratio would attain without data. Since the prior's pull should vanish as the data become informative, entries are split into quantile bins of the information available per parameter: observations per fixed effect (n/d) for $\boldsymbol{\beta}$, groups per random effect (m/q) for $\boldsymbol{\sigma}_\alpha$, and observations ($n$) for σ_ε.

Table B12 shows the expected dose–response: heavier-tailed families widen the posterior in every group and likelihood. The effect decays monotonically across all four information bins in all nine group–family–likelihood combinations. We note that the σ_ε encoding is effectively unused (ratio $\approx$1.00 at every information level): the residual scale is well identified even at the smallest n tested, so this input carries negligible training signal.

Table B12: Posterior SD-ratio of the heavier-tailed relative to the narrowest prior family, by quantile bin of the per-parameter information (Q1 least, Q4 most informative). Entries are median $\pm$ MAD over parameter entries. *Prior ratio* is the no-data ceiling implied by the prior SDs; σ_ε exists only for Gaussian likelihoods.

Group	Family pair	Info bin	Prior ratio	Gaussian	Bernoulli	Poisson
β	Normal $\to$ Student-t	n/d Q1	1.291	1.075 ± 0.050	1.069 ± 0.042	1.044 ± 0.039
		n/d Q2		1.048 ± 0.042	1.045 ± 0.037	1.025 ± 0.034
		n/d Q3		1.037 ± 0.040	1.034 ± 0.033	1.018 ± 0.031
		n/d Q4		1.027 ± 0.036	1.023 ± 0.031	1.013 ± 0.031
σ_α	Half-Normal $\to$ Half-Student-t	m/q Q1	1.452	1.043 ± 0.044	1.076 ± 0.063	1.045 ± 0.049
		m/q Q2		1.032 ± 0.038	1.059 ± 0.056	1.040 ± 0.041
		m/q Q3		1.022 ± 0.034	1.044 ± 0.044	1.025 ± 0.039
		m/q Q4		1.013 ± 0.030	1.030 ± 0.040	1.019 ± 0.032
σ_α	Half-Normal $\to$ Exponential	m/q Q1	1.659	1.056 ± 0.055	1.103 ± 0.090	1.063 ± 0.060
		m/q Q2		1.044 ± 0.049	1.081 ± 0.076	1.055 ± 0.053
		m/q Q3		1.035 ± 0.041	1.061 ± 0.064	1.039 ± 0.046
		m/q Q4		1.019 ± 0.035	1.041 ± 0.051	1.029 ± 0.037
σ_ε	Half-Normal $\to$ Half-Student-t	n Q1	1.452	1.004 ± 0.025	—	—
		n Q2		1.000 ± 0.022	—	—
		n Q3		1.003 ± 0.020	—	—
		n Q4		0.999 ± 0.023	—	—
σ_ε	Half-Normal $\to$ Exponential	n Q1	1.659	1.002 ± 0.023	—	—
		n Q2		1.001 ± 0.023	—	—
		n Q3		1.003 ± 0.023	—	—
		n Q4		0.999 ± 0.022	—	—

B.9 Emulated Fixed-Prior Baseline

The closest amortized alternatives to `metabeta` fix the prior at training time (Habermann et al., 2025; Arruda et al., 2024). We quantify what this costs an analyst whose prior differs from the training prior. This isolates what prior conditioning buys without training a reference network with a different architecture, capacity and convergence.

Setup.

- **Data.** The oracle sets of all four size regimes (128 datasets per family and regime, strict NUTS-converged subset). Unlike the stress tests of Appendix D, NUTS is not refit: data and NUTS stay under the stored prior P_1, only the prior passed to the network changes.
- **Reference priors P_0.** Derived from each dataset's P_1 (all scales three times wider or tighter; fixed-effect locations shifted by $2\tau_1$; prior families rotated as in Appendix D.2) or fixed for all datasets at the mode of the training hyper-prior (Appendix F.3), the prior a fixed-prior network would sensibly be trained at. The LKJ concentration stays at P_1. A dataset enters a condition only if every scale of P_0 lies within the training hyper-prior range, so the network never extrapolates; this keeps most datasets for tighter and few for wider P_0 (n in Table B13).
- **Arms.** $\texttt{MB}^0(P_0)$; the same flow as IMH proposal towards P_1 (Appendix C.1); `metabeta` at P_1 as reference. Emulation check: the flow as IMH proposal towards P_0, whose acceptance relative to `metabeta` measures how well the network fits at P_0.
- **Metrics**, fixed before the larger size regimes were run: σ-ratio against $\texttt{NUTS}(P_1)$, overall and in the lowest quartile of groups per random effect (m/q); EACE against the generating parameters; IMH acceptance.

Results. Table B13. Three take-aways:

1. **A fixed prior distorts uncertainty exactly where data are scarce.** A P_0 three times wider than P_1 makes the posterior 11–22% too wide overall, 20–30% in the lowest m/q quartile and 4–10% in the highest. A P_0 three times tighter makes it 9–35% too narrow and raises global EACE to 0.17–0.32, against $\leq$0.007 for NUTS; for binary outcomes, whose variance components stay prior-dominated, this overconfidence grows with the size regime (σ-ratio 0.73 at small to 0.54 at huge). Posterior means and predictions are insensitive throughout ($r \geq 0.98$, ΔLOO-NLL ≤ 0.06), and so is rotating the prior families at equal scale (σ-ratio 1.00–1.02): the damage is confined to uncertainty quantification.
2. **IMH repairs a vague fixed prior, not a tight one.** As proposal towards P_1, the wider-prior flow recovers NUTS at 0.3–0.8 times the acceptance of `metabeta`. The tighter-prior flow cannot supply the tails of its target: acceptance falls to 0.04–0.07 and the posterior stays 4–16% too narrow (global EACE 0.11–0.20). For both, the acceptance gap to `metabeta` widens as posteriors sharpen, from 1.1–4 times at small to 2–12 times at huge.
3. **Even a well-chosen fixed prior costs proposal quality.** The hyper-prior mode and a location shift of two prior SDs leave the raw posterior nearly intact (σ-ratio 1.00–1.03, global EACE 0.003–0.09) but cut acceptance towards P_1 to 0.08–0.23, against 0.47–0.52 for `metabeta`.

For `metabeta` this means that prior conditioning buys correct uncertainty quantification without a sampler on the datasets where an analyst cannot check it (few groups, tight priors), and the proposal quality on which the exactness path rests: a fixed-prior network used as IMH proposal pays in acceptance (up to thirteen-fold here) and cannot be repaired once its prior is tighter than the analyst's.

Table B13: Emulated fixed-prior baseline on the oracle sets (all four size regimes; NUTS-converged datasets whose P_0 lies within the training hyper-prior, n per row). $\texttt{MB}^0(P_0)$ is the raw flow conditioned on the reference prior P_0 while data and NUTS stay under the analyst's prior P_1; τ_1/τ_0 is the realised median scale ratio. Medians over datasets; σ-ratio$_{Q1}$ restricted to the lowest m/q quartile; EACE_g against the generating parameters (NUTS: ≤ 0.007). +IMH(P_1): the same flow refined as IMH proposal towards P_1. IMH acceptance of $\texttt{MB}^0(P_0)$ as proposal towards P_1 and towards P_0 (emulation check).

				$\texttt{MB}^0(P_0)$ vs. $\texttt{NUTS}(P_1)$			+IMH(P_1)		IMH acceptance	
family	P_0	n	τ_1/τ_0	σ-ratio	σ-ratio$_{Q1}$	EACE_g	σ-ratio	EACE_g	$\to P_1$	$\to P_0$
Gaussian	$P_0 = P_1$ (MB)	371	1	1.012	1.020	0.004	0.995	0.027	0.50	—
	$\tau_0 = 3\tau_1$	64	1/3	1.113	1.242	0.035	0.994	0.030	0.41	0.63
	$\tau_0 = \tau_1/3$	365	3	0.915	0.874	0.225	0.951	0.189	0.04	0.22
	$\nu_0 = \nu_1 + 2\tau_1$	371	1	1.031	1.046	0.089	0.992	0.091	0.08	0.33
	hyper-prior mode	371	1.4	1.011	1.022	0.088	0.994	0.100	0.09	0.38
Bernoulli	$P_0 = P_1$ (MB)	397	1	0.999	0.989	0.004	0.997	0.013	0.52	—
	$\tau_0 = 3\tau_1$	31	1/3	1.219	1.295	0.063	1.002	0.012	0.30	0.63
	$\tau_0 = \tau_1/3$	394	3	0.646	0.610	0.321	0.843	0.200	0.04	0.43
	$\nu_0 = \nu_1 + 2\tau_1$	397	1	1.000	1.013	0.063	0.992	0.048	0.14	0.41
	hyper-prior mode	397	1	0.997	1.008	0.021	0.995	0.029	0.22	0.51
Poisson	$P_0 = P_1$ (MB)	382	1	1.009	1.003	0.010	1.000	0.015	0.47	—
	$\tau_0 = 3\tau_1$	45	1/3	1.171	1.200	0.025	1.002	0.048	0.16	0.21
	$\tau_0 = \tau_1/3$	375	3	0.889	0.829	0.169	0.961	0.110	0.07	0.34
	$\nu_0 = \nu_1 + 2\tau_1$	382	1	1.018	1.018	0.036	0.996	0.044	0.17	0.34
	hyper-prior mode	382	1/1.2	1.027	1.025	0.003	0.998	0.029	0.23	0.35

C AMORTIZED INFERENCE MEETS MCMC

The amortized posterior can be combined with likelihood-based refinement in two ways: as a proposal distribution for Independence Metropolis–Hastings (asymptotically exact at no MCMC tuning cost, Appendix C.1), or as an initialization for NUTS (inheriting NUTS' asymptotic guarantees at a reduced tuning budget, Appendix C.2). Both use the exact unnormalized posterior, so their quality does not hinge on network generalization alone. Each subsection describes the procedure first and its empirical evaluation second.

C.1 INDEPENDENCE METROPOLIS–HASTINGS

Procedure. The flow $q(\boldsymbol{\vartheta}) = p_{\boldsymbol{\Pi}_g}(\boldsymbol{\vartheta} \mid \mathbf{s})$ serves as a fixed proposal for an IMH chain (Tierney, 1994). The chain runs over the global parameters $\boldsymbol{\vartheta}$ only; the random effects are marginalized out of the log-weight

$$\log w(\boldsymbol{\vartheta}) = \log p(\mathbf{D} \mid \boldsymbol{\vartheta}) + \log p(\boldsymbol{\vartheta}) - \log q(\boldsymbol{\vartheta}).$$

All $S = C \times T$ proposals are drawn and weighted upfront. Each of the C chains starts at the first draw of its block, runs T accept/reject steps and discards the first T_{burn}.

Gaussian outcomes. The marginal likelihood $p(\mathbf{D} \mid \boldsymbol{\vartheta})$ has a closed form. For every retained state, the random effects are redrawn from their exact conditional $p(\boldsymbol{\alpha}_i \mid \boldsymbol{\vartheta}, \mathbf{D}_i)$ (Rao–Blackwellization).

Bernoulli and Poisson outcomes. The marginal likelihood has no closed form. We replace it by the unbiased importance-sampling estimate (IS^2; Tran et al., 2013)

$$\hat{p}(\mathbf{D} \mid \boldsymbol{\vartheta}) = \prod_{i=1}^{m} \frac{1}{K} \sum_{k=1}^{K} \frac{p(\mathbf{y}_i \mid \boldsymbol{\alpha}_{ik}, \boldsymbol{\vartheta})\, p(\boldsymbol{\alpha}_{ik} \mid \boldsymbol{\vartheta})}{g_i(\boldsymbol{\alpha}_{ik})}, \qquad \boldsymbol{\alpha}_{ik} \sim g_i.$$

The inner proposal $g_i = (1-\epsilon)\,\mathcal{N}(\boldsymbol{\alpha}_i^*, \mathbf{H}_i^{-1}) + \epsilon\,\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\vartheta}})$ is a Laplace approximation of group i's conditional (mode $\boldsymbol{\alpha}_i^*$ by damped Newton steps, Hessian $\mathbf{H}_i$), mixed with a small share of the random-effect prior for safety. Because $\hat{p}$ is unbiased and each state keeps its own estimate, the chain is pseudo-marginal (Andrieu & Roberts, 2009). Its $\boldsymbol{\vartheta}$-marginal is the exact posterior; the estimator noise only lowers the acceptance rate. Each state also stores one inner draw per group, picked in proportion to its weight. A rejection would repeat these draws, so every retained step refreshes them with one sampling–importance-resampling move (Andrieu et al., 2018), which leaves $p(\boldsymbol{\alpha}_i \mid \boldsymbol{\vartheta}, \mathbf{D}_i)$ invariant. The same estimate $\hat{p}$ gives the marginal likelihood of Appendix B.6 for the discrete families. A plain Laplace approximation in place of $\hat{p}$ would be cheaper, but it targets a pseudo-posterior. On the real Bernoulli test sets it leaves posteriors 2–3% too narrow (σ-ratio 0.97–0.98 against 0.99–1.00 for the pseudo-marginal chain). On the *large* and *huge* oracle sets it leaves a predictive gap of 0.004–0.008 nats to NUTS, which the pseudo-marginal chain closes to $\leq$0.002.

Pool size and acceptance. The mean acceptance rate $\bar{a}$ measures how well the flow overlaps the posterior. Rates below $\approx$0.1 indicate a large gap. We swept the pool size in the hardest regime (Bernoulli, *huge*; $S \in \{1, 2, 4\} \times 10^3$). The fixed-effect calibration error of IMH decays as $(\bar{a}S)^{-0.6}$ and reaches the level of the easy regimes at $\bar{a}S \approx 700$ accepted draws. `metabeta` therefore reports a suggested pool size $S_{\text{sugg}} = \lceil 700/\bar{a} \rceil$ with its diagnostics, rounded up to a multiple of 500 and clamped to $[10^3, 1.6 \times 10^4]$ (Appendix D.6).

Defaults. $C = 4$ chains and $T_{\text{burn}} = 25$. For the discrete families, $K = 8$ inner draws and $\epsilon = 0.01$; under the posterior, this keeps the standard deviation of $\log \hat{p}(\mathbf{D} \mid \boldsymbol{\vartheta})$ at a median of 0.09–0.12 nats. The released implementation over-draws the pool to $S = C\,(\lceil n/C \rceil + T_{\text{burn}})$, so that exactly the n requested samples remain after burn-in (e.g. $n = 1000$: $S = 1{,}100$, $T = 275$). The benchmarks use a fixed pool of $S = 4{,}000$ ($T = 1{,}000$, 3,900 samples), matching the $4 \times 1{,}000$ draws of NUTS.

Refinement ablations. We compare the full refinement ladder on the oracle benchmarks (512 datasets per size regime): the raw amortized posterior (`MB`^{0}), i.e. `metabeta` with its IMH head ablated; self-normalized importance sampling (SNIS; Tokdar & Kass, 2010) toward the same target (`MB`^{0}+IS, marginal weights for Gaussian, Laplace weights for the discrete families); IMH (`MB`^{0}+IMH, the default pipeline `MB` of the main text); warm-started NUTS (`MB+NUTS`, treated in Appendix C.2); and

default NUTS as the reference (Appendix A.2). SNIS is an established method for NPE refinement (Dax et al., 2023; Gebhard et al., 2023; Barret & Dupourqué, 2026) and a close relative of IMH (Liu, 1996): IMH uses ratios of w in an accept–reject step and returns unweighted draws, whereas SNIS reweights the draws by w and carries a finite-sample bias that grows with the weight variance. In both cases unreliable proposals are flagged by the Pareto-smoothed importance sampling shape diagnostic (PSIS-$\hat{k} > 0.7$; Vehtari et al., 2017; Yao et al., 2018) (Appendix D.6).

Results. Table C2–C4 support two observations. First, the two weighted refinements act very similarly on recovery and calibration, and both shrink the raw posterior's predictive gap, with IMH ahead of IS on the largest regimes (Gaussian *huge* LOO-NLL $2.409 \to 2.247$ vs. 2.379, NUTS 2.244; Poisson *huge* $2.527 \to 1.397$ vs. 1.474, NUTS 1.397). SNIS matches IMH up to the *medium* regime, but its residual predictive gap to NUTS reaches 0.02–0.14 on the *huge* benchmarks, which is why IMH is the default head. Second, MB+NUTS matches cold-start NUTS on every accuracy metric at a quarter of the tuning budget. Table C1 reports the corresponding ablation on the real-world test sets of Section 3.2. The raw posterior agrees with NUTS in means and shape everywhere ($r \geq 0.99$, rank-MAD ≤ 0.04) but shows family-specific width biases: over-dispersion that grows with size for Gaussian outcomes (σ-ratio up to 1.28) and mild narrowing for Poisson (down to 0.87). This goes along with a predictive gap that grows with size (up to 0.45 for Gaussian and 1.70 for Poisson outcomes). The default IMH head removes the width biases (σ-ratio 0.96–1.00) and closes the predictive gap (≤ 0.01) at every size and in every family.

Table C1: Posterior agreement of the raw flow (MB^0) and the default pipeline (MB) with NUTS on each real-world test set. Entries are median ± MAD over datasets; r and rank-MAD are MB^0 values (MB matches or improves them; Table B4 and Appendix B.3). No *huge* regime exists for count outcomes (Table B6).

				σ-ratio		ΔLOO-NLL	
family	regime	r	rank-MAD	MB^0	MB	MB^0	MB
Gaussian	small	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.02	1.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00
	medium	1.00 ± 0.00	0.01 ± 0.01	1.04 ± 0.04	1.00 ± 0.01	0.01 ± 0.01	0.00 ± 0.00
	large	0.99 ± 0.00	0.02 ± 0.01	1.17 ± 0.14	1.00 ± 0.01	0.08 ± 0.06	0.00 ± 0.00
	huge	0.99 ± 0.00	0.03 ± 0.01	1.28 ± 0.15	1.00 ± 0.02	0.45 ± 0.35	0.00 ± 0.00
Bernoulli	small	1.00 ± 0.00	0.01 ± 0.00	1.02 ± 0.02	1.00 ± 0.01	0.00 ± 0.00	0.00 ± 0.00
	medium	1.00 ± 0.00	0.01 ± 0.00	1.00 ± 0.03	1.00 ± 0.01	0.01 ± 0.01	0.00 ± 0.00
	large	1.00 ± 0.00	0.01 ± 0.00	0.99 ± 0.03	1.00 ± 0.01	0.02 ± 0.01	0.00 ± 0.00
	huge	1.00 ± 0.00	0.01 ± 0.00	0.99 ± 0.03	0.99 ± 0.01	0.04 ± 0.02	0.00 ± 0.00
Poisson	small	1.00 ± 0.00	0.03 ± 0.01	0.97 ± 0.08	1.00 ± 0.01	0.04 ± 0.04	0.00 ± 0.00
	medium	0.99 ± 0.00	0.03 ± 0.01	0.93 ± 0.05	0.98 ± 0.03	0.19 ± 0.13	0.00 ± 0.00
	large	0.99 ± 0.01	0.04 ± 0.01	0.87 ± 0.06	0.96 ± 0.03	1.70 ± 1.55	0.00 ± 0.00

C.2 Warm-Started NUTS

Procedure. Warm-started NUTS (MB+NUTS) targets the exact posterior, while metabeta's proposal reduces the tuning budget needed to reach the typical set: 4 chains with 500 tuning steps and 500 draws, against $2{,}000$ and $1{,}000$ for the cold-start baseline on the identical PyMC model (Appendix A.2). We use $S = 4{,}000$ metabeta samples for initialization. The C chain starts are spaced evenly along the samples, and the initial diagonal mass matrix is the per-coordinate variance of 64 draws. Each start is back-transformed to PyMC's non-centered parameterization (Appendix A.2).

Results. MB+NUTS and cold-start NUTS are compared on the twelve oracle benchmarks (512 datasets each), both timed on the same four CPU cores (Appendix B.5); the reported MB+NUTS wall-clock time covers the complete procedure per dataset including metabeta sampling. Accuracy is at parity: MB+NUTS matches cold-start NUTS in recovery, calibration and predictive accuracy on every benchmark (Table C2–C4). Table C5 summarizes the sampler diagnostics. The warm start stabilizes sampling, cutting divergent transitions by more than 98% in each family, and accelerates it: the mean wall-clock time per dataset falls by 4–$9\times$ for Gaussian, 5–$12\times$ for Bernoulli and 2–$9\times$ for Poisson outcomes, with the smallest gain on the *huge* Poisson benchmark ($2.3\times$). The escalation fallback is triggered on 12–27% (Gaussian), 4–13% (Bernoulli) and 5–22% (Poisson) of datasets.

Table C2: Refinement ablation on each Gaussian oracle test set (512 datasets per regime). Recovery (r, NRMSE) and calibration (ECE, EACE) entries are averaged over parameters as in Table 1, but over all 512 datasets rather than the NUTS-converged subset; LOO-NLL is the median over datasets. $\texttt{MB}^0$+IMH is the default pipeline MB of the main text; the NUTS budgets are those of Table C5.

regime	model	r	NRMSE	ECE	EACE	LOO-NLL
small	$\texttt{MB}^0$	0.943	0.314	0.002	0.020	2.178
	$\texttt{MB}^0$+IS	0.947	0.302	0.002	0.017	2.177
	$\texttt{MB}^0$+IMH	0.947	0.302	0.001	0.018	2.177
	MB+NUTS	0.947	0.303	−0.000	0.019	2.176
	NUTS	0.946	0.305	0.001	0.018	2.177
medium	$\texttt{MB}^0$	0.951	0.295	0.008	0.020	2.284
	$\texttt{MB}^0$+IS	0.952	0.290	0.001	0.021	2.276
	$\texttt{MB}^0$+IMH	0.952	0.290	−0.001	0.020	2.260
	MB+NUTS	0.952	0.291	0.001	0.021	2.260
	NUTS	0.952	0.291	0.000	0.020	2.260
large	$\texttt{MB}^0$	0.952	0.295	0.007	0.017	2.354
	$\texttt{MB}^0$+IS	0.952	0.293	0.001	0.017	2.303
	$\texttt{MB}^0$+IMH	0.954	0.284	−0.007	0.019	2.272
	MB+NUTS	0.955	0.283	−0.000	0.019	2.272
	NUTS	0.955	0.283	0.002	0.018	2.274
huge	$\texttt{MB}^0$	0.956	0.284	0.007	0.016	2.409
	$\texttt{MB}^0$+IS	0.957	0.281	−0.003	0.015	2.379
	$\texttt{MB}^0$+IMH	0.958	0.276	−0.016	0.021	2.247
	MB+NUTS	0.959	0.274	−0.001	0.015	2.240
	NUTS	0.959	0.274	−0.001	0.015	2.244

Table C3: Refinement ablation on each Bernoulli oracle test set. Layout as in Table C2.

regime	model	r	NRMSE	ECE	EACE	LOO-NLL
small	$\texttt{MB}^0$	0.897	0.414	−0.002	0.024	0.551
	$\texttt{MB}^0$+IS	0.899	0.411	−0.003	0.021	0.550
	$\texttt{MB}^0$+IMH	0.899	0.412	0.001	0.021	0.550
	MB+NUTS	0.897	0.416	0.001	0.020	0.550
	NUTS	0.898	0.414	0.002	0.020	0.550
medium	$\texttt{MB}^0$	0.881	0.430	−0.005	0.018	0.460
	$\texttt{MB}^0$+IS	0.884	0.426	−0.006	0.019	0.452
	$\texttt{MB}^0$+IMH	0.884	0.426	−0.006	0.020	0.449
	MB+NUTS	0.883	0.427	−0.006	0.019	0.449
	NUTS	0.882	0.428	−0.006	0.020	0.449
large	$\texttt{MB}^0$	0.883	0.427	0.007	0.016	0.460
	$\texttt{MB}^0$+IS	0.882	0.426	0.001	0.015	0.434
	$\texttt{MB}^0$+IMH	0.883	0.427	0.000	0.015	0.427
	MB+NUTS	0.883	0.427	0.004	0.016	0.428
	NUTS	0.882	0.427	0.005	0.015	0.427
huge	$\texttt{MB}^0$	0.850	0.474	0.003	0.018	0.471
	$\texttt{MB}^0$+IS	0.855	0.469	−0.003	0.021	0.425
	$\texttt{MB}^0$+IMH	0.857	0.466	−0.011	0.025	0.409
	MB+NUTS	0.860	0.464	0.003	0.019	0.409
	NUTS	0.859	0.466	0.004	0.019	0.408

Table C4: Refinement ablation on each Poisson oracle test set. Layout as in Table C2.

regime	model	r	NRMSE	ECE	EACE	LOO-NLL
small	MB^0	0.931	0.339	−0.002	0.032	1.360
	MB^0+IS	0.933	0.335	0.002	0.031	1.345
	MB^0+IMH	0.933	0.336	0.002	0.031	1.341
	MB+NUTS	0.932	0.338	−0.001	0.033	1.339
	NUTS	0.932	0.337	0.002	0.031	1.339
medium	MB^0	0.936	0.318	−0.002	0.015	1.449
	MB^0+IS	0.935	0.320	−0.004	0.018	1.372
	MB^0+IMH	0.936	0.317	−0.006	0.019	1.357
	MB+NUTS	0.935	0.321	−0.006	0.018	1.357
	NUTS	0.935	0.320	−0.005	0.017	1.358
large	MB^0	0.943	0.298	−0.001	0.023	1.740
	MB^0+IS	0.943	0.295	−0.015	0.022	1.425
	MB^0+IMH	0.943	0.296	−0.019	0.024	1.394
	MB+NUTS	0.933	0.321	−0.012	0.020	1.394
	NUTS	0.933	0.321	−0.012	0.020	1.395
huge	MB^0	0.921	0.335	0.009	0.024	2.527
	MB^0+IS	0.922	0.333	−0.001	0.021	1.474
	MB^0+IMH	0.924	0.331	−0.023	0.030	1.397
	MB+NUTS	0.925	0.326	−0.002	0.019	1.397
	NUTS	0.925	0.326	0.001	0.020	1.397

Table C5: Sampler diagnostics of warm-started (MB+NUTS: 4 chains, 500 tuning steps, 500 draws) and cold-start NUTS (4 chains, 2,000 tuning steps, 1,000 draws) on each oracle test set (512 datasets per regime). Divergences are total divergent transitions over all datasets and chains; time is the mean wall-clock time in seconds per dataset over all 512 datasets on four CPU cores, speed-up is the ratio of the two means.

		divergences		time [s]		
family	regime	NUTS	MB+NUTS	NUTS	MB+NUTS	speed-up
Gaussian	small	1,737	16	95.7	12.5	7.7×
	medium	3,789	11	138.1	14.9	9.3×
	large	2,424	26	129.1	23.8	5.4×
	huge	2,203	2	141.0	33.8	4.2×
Bernoulli	small	988	7	83.4	7.2	11.6×
	medium	1,450	4	115.1	17.3	6.7×
	large	632	1	100.8	21.9	4.6×
	huge	1,881	9	187.2	36.4	5.1×
Poisson	small	5,112	64	82.2	10.1	8.1×
	medium	8,001	81	166.2	18.1	9.2×
	large	7,830	129	143.0	30.1	4.8×
	huge	5,081	152	176.6	76.7	2.3×

D Robustness and Stress Tests

This appendix collects all experiments that deliberately place the test distribution outside the training simulator and probes regimes where amortized inference is expected to fail: misspecified likelihoods (Appendix D.1), misspecified priors (Appendix D.2), out-of-distribution predictors (Appendix D.3), ill-conditioned designs (Appendix D.4), and data-poor groups (Appendix D.5). It closes with the safeguards shipped that automatically flag unreliable posteriors (Appendix D.6).

Common protocol. NUTS is refit under the same perturbation in every suite. Under likelihood and prior misspecification both methods therefore target the same wrong posterior, so their agreement separates amortization error from modeling error; under out-of-distribution predictors, ill-conditioned designs and data poverty the fitted GLMM stays correctly specified and NUTS is an exact reference. Agreement metrics (Appendix A.1) are computed pairwise on the strict NUTS-converged subset (Appendix A.2). Where the generating parameters are known, quality against them is reported for global (g) and local (l) parameters separately. Each suite reports `metabeta` next to the ablated raw flow $\texttt{MB}^0$ (Appendix C.1), whose predictive gap to NUTS measures how hard a condition hits the amortized proposal.

Common outcome. The same picture emerges in every suite, so we state it once and report only the suite-specific numbers below. `metabeta` stays faithful to NUTS under every perturbation: $r \geq 0.99$, posterior widths within a few percent of NUTS, rank-MAD ≤ 0.025 and $|\Delta\text{LOO-NLL}| \leq 0.006$ in every condition. NUTS itself converges on only 43–87% of the perturbed datasets (43–86% in the worst-case conditions of Table 2). Where the fitted GLMM is wrong, `metabeta` and NUTS degrade in lockstep against the generating parameters. The raw flow $\texttt{MB}^0$ agrees with NUTS in means and largely in marginal shape, but its predictive gap grows with the severity of the stress, and its posterior widths drift where the stress is strongest. The IMH head closes the predictive gap and removes the width bias in every suite, at the price of a mild finite-pool overconfidence ($\leq$0.04 EACE over NUTS).

D.1 Likelihood Misspecification

Setup. We use the oracle test sets (32 datasets per size regime, 128 per condition and family) and keep the predictors and generating parameters. We regenerate the outcomes $\mathbf{y}$ under contaminated likelihoods at three severities per family:

- Gaussian $\rightarrow$ Student-t ($\nu \in \{10, 5, 3\}$);
- Bernoulli $\rightarrow$ latent logit noise ($\sigma_c \in \{0.5, 1, 2\}$), the binary analogue of overdispersion;
- Poisson $\rightarrow$ Negative Binomial (size $\theta \in \{10, 3, 1\}$; smaller θ = more overdispersion).

$\texttt{MB}^0$, `metabeta` and NUTS fit the original (misspecified) GLMM, and their posteriors are compared.

Results. Table D1–D3. The raw flow's predictive gap grows with severity for Poisson outcomes (median ΔLOO-NLL 0.04 $\rightarrow$ 0.39 at θ=1), and the corresponding per-dataset *means* (not shown) reach 23.2 vs. 2.1 at θ=1, exposing rare material failures of $\texttt{MB}^0$on count data. Calibration against the generating parameters degrades at the same pace for `metabeta` and NUTS (e.g. global EACE 0.24 vs. 0.24 at σ_c=2; 0.37 vs. 0.36 at θ=1): when the GLMM is wrong, `metabeta` is as wrong as NUTS, but not worse.

Table D1: Likelihood misspecification on the Gaussian oracle sets (outcomes regenerated from Student-t with ν degrees of freedom; 128 datasets per condition). Left: agreement with NUTS fitting the same misspecified model, median $\pm$ MAD over the conv. of 128 datasets that pass the strict criterion (Appendix A.2); r and rank-MAD as medians only (their MAD is ≤ 0.005 and ≤ 0.01 throughout). Right: quality against the generating parameters, recovery (NRMSE) and calibration (EACE) for global (g) and local (l) parameters.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
baseline	MB^0	94	1.000	1.009 ± 0.014	0.005	0.005 ± 0.005	0.195	0.004	0.339	0.010
	MB		1.000	0.995 ± 0.012	0.004	0.000 ± 0.001	0.194	0.028	0.338	0.008
	NUTS		—	—	—	—	0.191	0.008	0.338	0.006
$t(\nu{=}10)$	MB^0	95	1.000	1.012 ± 0.019	0.005	0.003 ± 0.003	0.257	0.050	0.366	0.003
	MB		1.000	0.995 ± 0.009	0.004	0.000 ± 0.001	0.256	0.082	0.366	0.005
	NUTS		—	—	—	—	0.253	0.054	0.365	0.004
$t(\nu{=}5)$	MB^0	95	1.000	1.010 ± 0.019	0.006	0.004 ± 0.004	0.364	0.063	0.359	0.019
	MB		1.000	0.997 ± 0.011	0.004	0.000 ± 0.001	0.358	0.086	0.356	0.017
	NUTS		—	—	—	—	0.358	0.065	0.356	0.016
$t(\nu{=}3)$	MB^0	93	1.000	1.011 ± 0.016	0.006	0.004 ± 0.004	0.595	0.074	0.468	0.012
	MB		1.000	0.999 ± 0.012	0.004	-0.001 ± 0.002	0.576	0.089	0.467	0.014
	NUTS		—	—	—	—	0.574	0.068	0.467	0.015

Table D2: Likelihood misspecification on the Bernoulli oracle sets (unmodeled latent logit noise of scale σ_c). Layout as in Table D1.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
baseline	MB^0	99	0.998	1.002 ± 0.025	0.007	0.007 ± 0.006	0.374	0.009	0.683	0.005
	MB		1.000	0.999 ± 0.011	0.003	0.000 ± 0.000	0.374	0.009	0.681	0.006
	NUTS		—	—	—	—	0.374	0.009	0.680	0.007
$\sigma_c{=}0.5$	MB^0	101	0.998	0.996 ± 0.027	0.008	0.007 ± 0.006	0.390	0.006	0.710	0.016
	MB		1.000	1.001 ± 0.010	0.002	0.000 ± 0.000	0.382	0.013	0.693	0.012
	NUTS		—	—	—	—	0.378	0.008	0.687	0.012
$\sigma_c{=}1$	MB^0	99	0.998	1.001 ± 0.025	0.007	0.007 ± 0.007	0.409	0.068	0.739	0.025
	MB		1.000	0.996 ± 0.009	0.002	0.000 ± 0.000	0.401	0.080	0.735	0.021
	NUTS		—	—	—	—	0.402	0.070	0.734	0.017
$\sigma_c{=}2$	MB^0	101	0.998	1.007 ± 0.035	0.008	0.007 ± 0.006	0.536	0.222	0.773	0.084
	MB		1.000	1.000 ± 0.008	0.002	0.000 ± 0.000	0.532	0.244	0.770	0.092
	NUTS		—	—	—	—	0.531	0.236	0.770	0.091

Table D3: Likelihood misspecification on the Poisson oracle sets (outcomes regenerated from a Negative Binomial with size θ; smaller θ = more overdispersion). Layout as in Table D1.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
baseline	MB^0	93	0.999	1.009 ± 0.029	0.008	0.038 ± 0.038	0.241	0.007	0.588	0.004
	MB		1.000	0.997 ± 0.009	0.003	0.000 ± 0.000	0.241	0.041	0.582	0.002
	NUTS		—	—	—	—	0.241	0.013	0.582	0.003
$\theta{=}10$	MB^0	96	0.999	1.005 ± 0.024	0.008	0.075 ± 0.075	0.351	0.097	0.642	0.013
	MB		1.000	1.000 ± 0.006	0.003	0.000 ± 0.001	0.352	0.114	0.640	0.011
	NUTS		—	—	—	—	0.352	0.108	0.640	0.012
$\theta{=}3$	MB^0	90	0.998	1.009 ± 0.020	0.011	0.136 ± 0.136	0.491	0.179	0.828	0.042
	MB		1.000	1.000 ± 0.005	0.002	0.000 ± 0.001	0.496	0.203	0.837	0.045
	NUTS		—	—	—	—	0.499	0.190	0.840	0.044
$\theta{=}1$	MB^0	110	0.996	1.005 ± 0.022	0.016	0.393 ± 0.390	0.786	0.332	1.184	0.127
	MB		1.000	0.998 ± 0.005	0.003	0.000 ± 0.003	0.818	0.365	1.237	0.139
	NUTS		—	—	—	—	0.824	0.355	1.243	0.138

D.2 Prior Misspecification

Setup. The prior hyperparameters stored with generated test sets are rewritten while the observations and the true parameters remain untouched: the fitted prior systematically deviates from the one that generated the data. Five conditions cover wrong scale ($\tau\times c$, all scale hyperparameters multiplied by $c \in \{1/3, 3\}$), wrong location ($\nu_\beta + k\tau_\beta$, fixed-effect prior means shifted by k prior SDs), and wrong family (family+1, all prior-family indices rotated, e.g. Normal $\rightarrow$ Student-t); each uses 128 datasets (32 per size regime) per likelihood family.

Results. Table D4–D6. Prior misspecification opens no gap between the amortized and the exact posterior (σ-ratio within 1% of one, $|\Delta\text{LOO-NLL}| \leq 0.001$), while NUTS converges on only 67–81% of the datasets. Quality against the generating parameters degrades in step with NUTS, and asymmetrically. An over-tight prior ($\tau\times 1/3$) is the most damaging for all models (global EACE rises to 0.22–0.32 for `metabeta` and 0.20–0.31 for NUTS, from $\leq$0.04 and $\leq$0.01 under the correct prior). An over-wide prior ($\tau\times 3$) is largely benign. The raw flow's predictive gap stays at most 0.02 for Gaussian and Bernoulli outcomes and reaches 0.15 for Poisson.

Table D4: Prior misspecification on the Gaussian oracle sets (prior scale $\tau\times c$, fixed-effect prior location $\nu_\beta + k\tau_\beta$, or prior family rotated; `NUTS` refit under the same perturbed prior; 128 datasets per condition, pooled over the four size regimes). Layout as in Table D1.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
correct	MB^0	94	0.999	1.014 ± 0.023	0.006	0.006 ± 0.005	0.196	0.004	0.338	0.011
	MB		1.000	0.996 ± 0.012	0.005	0.000 ± 0.001	0.190	0.036	0.338	0.005
	NUTS		—	—	—	—	0.191	0.008	0.338	0.006
$\tau\times 1/3$	MB^0	96	0.999	1.017 ± 0.031	0.010	0.009 ± 0.008	0.317	0.239	0.485	0.068
	MB		1.000	0.996 ± 0.016	0.006	0.000 ± 0.001	0.304	0.286	0.478	0.069
	NUTS		—	—	—	—	0.297	0.265	0.477	0.070
$\tau\times 3$	MB^0	97	0.999	1.018 ± 0.024	0.007	0.008 ± 0.007	0.233	0.019	0.368	0.032
	MB		1.000	0.999 ± 0.016	0.005	0.000 ± 0.001	0.231	0.033	0.367	0.020
	NUTS		—	—	—	—	0.228	0.013	0.366	0.026
$\nu_\beta + 1\tau_\beta$	MB^0	99	0.999	1.015 ± 0.023	0.006	0.007 ± 0.007	0.199	0.029	0.341	0.004
	MB		1.000	0.997 ± 0.014	0.004	0.000 ± 0.001	0.193	0.068	0.339	0.004
	NUTS		—	—	—	—	0.193	0.033	0.339	0.003
$\nu_\beta + 2\tau_\beta$	MB^0	92	0.999	1.015 ± 0.020	0.007	0.010 ± 0.009	0.251	0.084	0.367	0.005
	MB		1.000	0.995 ± 0.013	0.006	0.000 ± 0.001	0.233	0.132	0.366	0.006
	NUTS		—	—	—	—	0.230	0.099	0.369	0.006
family+1	MB^0	100	1.000	1.008 ± 0.021	0.007	0.006 ± 0.006	0.195	0.005	0.329	0.009
	MB		1.000	0.997 ± 0.012	0.004	0.000 ± 0.001	0.196	0.033	0.330	0.009
	NUTS		—	—	—	—	0.192	0.005	0.328	0.009

Table D5: Prior misspecification on the Bernoulli oracle sets. Layout as in Table D4.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
correct	MB^0	99	0.998	1.001 ± 0.024	0.007	0.007 ± 0.006	0.375	0.011	0.682	0.004
	MB		1.000	0.999 ± 0.011	0.003	0.000 ± 0.000	0.374	0.009	0.681	0.006
	NUTS		—	—	—	—	0.374	0.009	0.680	0.007
$\tau \times 1/3$	MB^0	101	0.998	1.001 ± 0.079	0.013	0.004 ± 0.004	0.568	0.303	0.768	0.175
	MB		1.000	1.002 ± 0.012	0.003	0.000 ± 0.000	0.555	0.318	0.758	0.172
	NUTS		—	—	—	—	0.555	0.310	0.758	0.172
$\tau \times 3$	MB^0	101	0.997	0.958 ± 0.054	0.014	0.021 ± 0.018	0.522	0.052	0.726	0.032
	MB		0.999	0.992 ± 0.019	0.004	0.000 ± 0.001	0.596	0.020	0.733	0.047
	NUTS		—	—	—	—	0.736	0.025	0.782	0.053
$\nu_\beta + 1\tau_\beta$	MB^0	104	0.998	0.991 ± 0.037	0.009	0.009 ± 0.008	0.411	0.015	0.725	0.004
	MB		1.000	0.998 ± 0.011	0.003	0.000 ± 0.000	0.410	0.036	0.710	0.006
	NUTS		—	—	—	—	0.415	0.028	0.708	0.007
$\nu_\beta + 2\tau_\beta$	MB^0	96	0.998	0.971 ± 0.037	0.011	0.010 ± 0.009	0.467	0.080	0.703	0.014
	MB		1.000	0.996 ± 0.007	0.003	0.000 ± 0.000	0.487	0.104	0.719	0.002
	NUTS		—	—	—	—	0.509	0.098	0.728	0.002
family+1	MB^0	100	0.998	0.988 ± 0.037	0.009	0.009 ± 0.008	0.386	0.031	0.725	0.006
	MB		1.000	0.999 ± 0.013	0.003	0.000 ± 0.000	0.369	0.014	0.701	0.005
	NUTS		—	—	—	—	0.369	0.022	0.700	0.009

Table D6: Prior misspecification on the Poisson oracle sets. Layout as in Table D4.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
correct	MB^0	93	0.999	1.009 ± 0.033	0.009	0.043 ± 0.042	0.246	0.007	0.589	0.003
	MB		1.000	0.997 ± 0.009	0.003	0.000 ± 0.000	0.241	0.040	0.583	0.002
	NUTS		—	—	—	—	0.241	0.013	0.582	0.003
$\tau \times 1/3$	MB^0	94	0.999	1.036 ± 0.045	0.013	0.030 ± 0.029	0.366	0.168	0.603	0.075
	MB		1.000	0.998 ± 0.008	0.003	0.000 ± 0.000	0.356	0.216	0.591	0.091
	NUTS		—	—	—	—	0.352	0.203	0.588	0.088
$\tau \times 3$	MB^0	91	0.997	1.012 ± 0.054	0.012	0.152 ± 0.151	0.298	0.011	0.606	0.034
	MB		1.000	1.000 ± 0.013	0.004	0.000 ± 0.001	0.285	0.042	0.598	0.028
	NUTS		—	—	—	—	0.291	0.014	0.595	0.031
$\nu_\beta + 1\tau_\beta$	MB^0	91	0.998	1.021 ± 0.046	0.012	0.031 ± 0.031	0.305	0.025	0.608	0.004
	MB		1.000	0.997 ± 0.011	0.003	0.000 ± 0.000	0.296	0.062	0.600	0.007
	NUTS		—	—	—	—	0.293	0.037	0.597	0.005
$\nu_\beta + 2\tau_\beta$	MB^0	86	0.998	1.025 ± 0.045	0.012	0.057 ± 0.056	0.318	0.059	0.599	0.008
	MB		1.000	1.001 ± 0.011	0.003	0.000 ± 0.000	0.305	0.099	0.592	0.014
	NUTS		—	—	—	—	0.308	0.080	0.590	0.015
family+1	MB^0	94	0.998	1.013 ± 0.037	0.010	0.064 ± 0.064	0.307	0.008	0.638	0.003
	MB		1.000	1.001 ± 0.009	0.003	0.000 ± 0.000	0.308	0.028	0.634	0.003
	NUTS		—	—	—	—	0.305	0.009	0.633	0.003

D.3 Out-of-Distribution Predictors

Setup. The training design space is broad but structurally constrained: all simulated and real training designs have finite variance, asymptotically independent tails, and no temporal structure within groups. We probe each constraint directly. Taking the same held-out test sets as in Appendix D.1 and keeping the priors, likelihood and parameters the same, we re-sample the predictors from distributions strictly outside the training support and regenerate the outcomes:

- heavy-tailed marginals ($X \sim t(\nu{=}2)$ and $X \sim \mathrm{Cauchy}$, i.e. infinite variance);
- Clayton-copula dependence (Kendall $\tau \in \{0.5, 0.9\}$, inducing tail dependence);
- and longitudinal growth-curve designs (time as an additional predictor with a random slope).

The fitted GLMM therefore remains correctly specified, and any gap isolates the amortization error.

Results. Table D7–D9. The raw flow's predictive performance decreases on the extremes (median ΔLOO-NLL 0.19 for Bernoulli and 0.26 for Poisson under Cauchy predictors), which the IMH head repairs. Even strictly outside the training support there is barely a gap between `metabeta` and NUTS (posterior widths within 1%, $|\Delta\text{LOO-NLL}| \leq 0.001$).

Table D7: Out-of-distribution predictors on the Gaussian oracle sets: predictors re-sampled from $t(\nu{=}2)$ or Cauchy marginals, Clayton copulas with Kendall $\tau \in \{0.5, 0.9\}$, or longitudinal designs, outcomes regenerated under the original likelihood (128 datasets per condition). Layout as in Table D1.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
baseline	MB^0	94	1.000	1.009 ± 0.014	0.005	0.005 ± 0.005	0.195	0.004	0.339	0.010
	MB		1.000	0.995 ± 0.012	0.004	0.000 ± 0.001	0.194	0.028	0.338	0.008
	NUTS		—	—	—	—	0.191	0.008	0.338	0.006
$t(\nu{=}2)$	MB^0	88	1.000	1.007 ± 0.018	0.005	0.002 ± 0.002	0.185	0.008	0.455	0.019
	MB		1.000	0.997 ± 0.011	0.004	0.000 ± 0.001	0.183	0.025	0.454	0.021
	NUTS		—	—	—	—	0.181	0.011	0.454	0.016
Cauchy	MB^0	75	1.000	1.013 ± 0.017	0.007	0.004 ± 0.004	0.218	0.009	0.553	0.004
	MB		1.000	0.997 ± 0.009	0.004	0.000 ± 0.001	0.214	0.004	0.552	0.011
	NUTS		—	—	—	—	0.213	0.008	0.552	0.004
Clayton 0.5	MB^0	93	1.000	1.009 ± 0.020	0.007	0.003 ± 0.003	0.210	0.032	0.357	0.005
	MB		1.000	0.998 ± 0.009	0.004	0.000 ± 0.001	0.203	0.005	0.355	0.006
	NUTS		—	—	—	—	0.198	0.018	0.354	0.004
Clayton 0.9	MB^0	81	0.999	1.018 ± 0.029	0.009	0.014 ± 0.014	0.304	0.039	0.400	0.017
	MB		1.000	0.995 ± 0.012	0.005	0.000 ± 0.001	0.298	0.018	0.397	0.007
	NUTS		—	—	—	—	0.285	0.010	0.396	0.011
longitudinal	MB^0	96	1.000	1.008 ± 0.015	0.005	0.002 ± 0.002	0.168	0.022	0.395	0.008
	MB		1.000	0.995 ± 0.009	0.003	0.000 ± 0.001	0.167	0.010	0.395	0.005
	NUTS		—	—	—	—	0.165	0.016	0.394	0.007

Table D8: Out-of-distribution predictors on the Bernoulli oracle sets. Layout as in Table D7.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
baseline	MB^0	99	0.998	1.002 ± 0.025	0.007	0.007 ± 0.006	0.374	0.009	0.683	0.005
	MB		1.000	0.999 ± 0.011	0.003	0.000 ± 0.000	0.374	0.009	0.681	0.006
	NUTS		—	—	—	—	0.374	0.009	0.680	0.007
$t(\nu{=}2)$	MB^0	99	0.998	0.999 ± 0.023	0.007	0.013 ± 0.012	0.322	0.009	0.692	0.012
	MB		1.000	0.998 ± 0.007	0.002	0.000 ± 0.000	0.309	0.012	0.687	0.008
	NUTS		—	—	—	—	0.306	0.013	0.687	0.006
Cauchy	MB^0	105	0.994	1.000 ± 0.035	0.010	0.191 ± 0.120	0.483	0.049	0.776	0.015
	MB		1.000	0.994 ± 0.013	0.003	0.000 ± 0.000	0.450	0.010	0.765	0.011
	NUTS		—	—	—	—	0.442	0.015	0.766	0.006
Clayton 0.5	MB^0	99	0.999	0.997 ± 0.027	0.007	0.005 ± 0.005	0.321	0.014	0.685	0.021
	MB		1.000	0.997 ± 0.009	0.002	0.000 ± 0.000	0.309	0.005	0.672	0.010
	NUTS		—	—	—	—	0.308	0.005	0.669	0.007
Clayton 0.9	MB^0	98	0.997	1.014 ± 0.056	0.011	0.012 ± 0.010	0.502	0.047	0.720	0.006
	MB		1.000	0.998 ± 0.010	0.003	0.000 ± 0.000	0.506	0.004	0.712	0.013
	NUTS		—	—	—	—	0.507	0.005	0.715	0.015
longitudinal	MB^0	105	0.999	0.991 ± 0.027	0.007	0.004 ± 0.004	0.279	0.020	0.634	0.001
	MB		1.000	0.998 ± 0.008	0.002	0.000 ± 0.000	0.268	0.017	0.620	0.014
	NUTS		—	—	—	—	0.268	0.011	0.621	0.017

Table D9: Out-of-distribution predictors on the Poisson oracle sets. Layout as in Table D7.

			agreement with NUTS				quality vs. generating parameters			
condition	model	conv.	r	σ-ratio	rank-MAD	ΔLOO-NLL	NRMSE$_g$	EACE$_g$	NRMSE$_l$	EACE$_l$
baseline	MB^0	93	0.999	1.009 ± 0.029	0.008	0.038 ± 0.038	0.241	0.007	0.588	0.004
	MB		1.000	0.997 ± 0.009	0.003	0.000 ± 0.000	0.241	0.041	0.582	0.002
	NUTS		—	—	—	—	0.241	0.013	0.582	0.003
$t(\nu{=}2)$	MB^0	84	0.999	0.996 ± 0.018	0.008	0.150 ± 0.149	0.223	0.013	0.543	0.015
	MB		1.000	0.999 ± 0.007	0.002	0.000 ± 0.000	0.219	0.013	0.545	0.020
	NUTS		—	—	—	—	0.238	0.020	0.557	0.013
Cauchy	MB^0	70	0.999	1.012 ± 0.021	0.007	0.257 ± 0.256	0.250	0.015	0.622	0.010
	MB		1.000	1.000 ± 0.007	0.003	0.000 ± 0.000	0.241	0.007	0.616	0.006
	NUTS		—	—	—	—	0.243	0.011	0.616	0.007
Clayton 0.5	MB^0	96	0.999	1.014 ± 0.029	0.009	0.035 ± 0.034	0.246	0.034	0.515	0.004
	MB		1.000	1.002 ± 0.008	0.002	0.000 ± 0.000	0.241	0.010	0.507	0.002
	NUTS		—	—	—	—	0.244	0.008	0.510	0.002
Clayton 0.9	MB^0	88	0.999	1.016 ± 0.030	0.011	0.143 ± 0.141	0.372	0.062	0.590	0.008
	MB		1.000	0.999 ± 0.010	0.003	0.000 ± 0.001	0.367	0.013	0.584	0.007
	NUTS		—	—	—	—	0.376	0.023	0.589	0.013
longitudinal	MB^0	92	0.999	1.012 ± 0.025	0.008	0.032 ± 0.032	0.208	0.015	0.564	0.006
	MB		1.000	1.001 ± 0.006	0.002	0.000 ± 0.000	0.205	0.005	0.558	0.006
	NUTS		—	—	—	—	0.206	0.005	0.558	0.007

D.4 Ill-Conditioned Designs

Setup. Real design matrices are often collinear, so we quantify agreement with NUTS as a function of the condition number $\kappa_2(\mathbf{X}) = \sigma_{\max}/\sigma_{\min}$ of each real dataset's fixed-effects design. Here, σ_* denotes singular values of the design's SVD, and a large ratio indicates high collinearity. Unlike the preceding suites this is a post-hoc analysis: simulated designs are near-orthogonal ($\kappa_2 \approx 1$). In the real-world benchmarks of each family, collinearity arises naturally and κ_2 spans 1 to $\sim 10^{15}$ with a near-singular tail. Results are binned by κ_2.

Results. Table D10–D12. Collinearity is the axis that stresses the amortized proposal most. The raw flow remains faithful in posterior means ($r \geq 0.99$ at all κ_2) but degrades in width and predictive fit: for Gaussian outcomes it widens (σ-ratio up to 1.38) and its predictive gap grows to 0.54 with κ_2; for the discrete families it narrows instead (σ-ratio down to 0.95 for Bernoulli, 0.87 for Poisson). Predictive gaps stay small for Bernoulli ($\leq$0.06) but grow large for Poisson (3.01 in the $[6, 10)$ bin). NUTS itself struggles increasingly on collinear data (Gaussian: $87\% \to 43\%$ converged before the near-singular tail). `metabeta` holds NUTS-level agreement across the whole range, including the near-singular tail at $\kappa_2 \approx 10^{15}$, with one visible residue: the discrete families narrow mildly at high κ_2 (σ-ratio down to 0.99 for Bernoulli and 0.95 for Poisson).

A bin-free analysis confirms this pattern on the Gaussian benchmarks: Spearman rank correlations over all converged datasets show the raw flow degrading monotonically with κ_2 ($r_s(\kappa_2, \Delta\text{LOO}) = +0.83$, $r_s(\kappa_2, \sigma\text{-ratio}) = +0.63$), while the IMH head removes the width dependence entirely ($r_s = -0.01$) and strongly attenuates the predictive one ($+0.38$).

Table D10: Posterior agreement vs. design condition number $\kappa_2(\mathbf{X})$ on the Gaussian real-world benchmarks. Entries are median $\pm$ MAD over the NUTS-converged datasets of each bin; $\%$ conv. is the NUTS convergence rate per bin (strict criterion) and $\tilde{\kappa}$ the median condition number within the bin.

$\kappa_2(X)$	model	n	n_{conv}	% conv.	$\tilde{\kappa}$	r	σ-ratio	rank-MAD	ΔLOO-NLL
$[1, 3)$	MB^0	376	326	87	1.00	1.000 ± 0.000	1.001 ± 0.017	0.005 ± 0.003	0.001 ± 0.001
	MB					1.000 ± 0.000	0.999 ± 0.011	0.004 ± 0.002	0.000 ± 0.001
$[3, 6)$	MB^0	555	401	72	4.44	0.999 ± 0.001	1.038 ± 0.041	0.010 ± 0.006	0.008 ± 0.007
	MB					1.000 ± 0.000	0.998 ± 0.011	0.004 ± 0.002	0.000 ± 0.001
$[6, 10)$	MB^0	919	577	63	7.95	0.992 ± 0.005	1.188 ± 0.131	0.024 ± 0.011	0.114 ± 0.091
	MB					0.996 ± 0.003	0.999 ± 0.024	0.009 ± 0.005	0.001 ± 0.002
$[10, 10^6)$	MB^0	127	54	43	11.01	0.993 ± 0.005	1.363 ± 0.252	0.029 ± 0.016	0.541 ± 0.428
	MB					0.991 ± 0.004	0.998 ± 0.038	0.012 ± 0.006	0.005 ± 0.005
$[10^6, \infty)$	MB^0	71	62	87	4.2×10^{15}	0.987 ± 0.009	1.378 ± 0.259	0.030 ± 0.014	0.150 ± 0.097
	MB					0.993 ± 0.004	0.995 ± 0.028	0.018 ± 0.006	0.001 ± 0.001

Table D11: Posterior agreement vs. $\kappa_2(\mathbf{X})$ on the Bernoulli real-world benchmarks. Layout as in Table D10.

$\kappa_2(X)$	model	n	n_{conv}	% conv.	$\tilde{\kappa}$	r	σ-ratio	rank-MAD	ΔLOO-NLL
$[1, 3)$	MB^0	354	310	88	1.05	0.999 ± 0.001	1.026 ± 0.025	0.009 ± 0.005	0.001 ± 0.001
	MB					1.000 ± 0.000	1.000 ± 0.007	0.002 ± 0.001	0.000 ± 0.000
$[3, 6)$	MB^0	637	495	78	4.65	0.997 ± 0.002	1.004 ± 0.029	0.009 ± 0.004	0.006 ± 0.004
	MB					1.000 ± 0.000	0.999 ± 0.007	0.002 ± 0.001	0.000 ± 0.000
$[6, 10)$	MB^0	961	636	66	7.68	0.996 ± 0.003	0.988 ± 0.027	0.008 ± 0.003	0.025 ± 0.014
	MB					0.999 ± 0.000	0.995 ± 0.008	0.003 ± 0.001	0.000 ± 0.000
$[10, 10^6)$	MB^0	67	48	72	10.50	0.995 ± 0.003	0.951 ± 0.025	0.010 ± 0.003	0.061 ± 0.027
	MB					0.998 ± 0.001	0.995 ± 0.016	0.004 ± 0.002	0.000 ± 0.001
$[10^6, \infty)$	MB^0	29	24	83	3.9×10^{15}	0.997 ± 0.002	0.947 ± 0.030	0.011 ± 0.004	0.015 ± 0.006
	MB					0.999 ± 0.000	0.989 ± 0.012	0.003 ± 0.001	0.000 ± 0.001

Table D12: Posterior agreement vs. $\kappa_2(\mathbf{X})$ on the Poisson real-world benchmarks. Layout as in Table D10. The near-singular bin is absent: only two Poisson datasets exceed $\kappa_2 = 10$.

$\kappa_2(X)$	model	n	n_{conv}	% conv.	$\tilde{\kappa}$	r	σ-ratio	rank-MAD	ΔLOO-NLL
$[1, 3)$	MB^0	479	208	43	1.00	0.996 ± 0.002	0.957 ± 0.058	0.035 ± 0.015	0.039 ± 0.033
	MB					1.000 ± 0.000	1.003 ± 0.012	0.006 ± 0.003	0.000 ± 0.001
$[3, 6)$	MB^0	688	445	65	4.28	0.991 ± 0.005	0.921 ± 0.064	0.032 ± 0.011	0.258 ± 0.186
	MB					1.000 ± 0.000	0.976 ± 0.027	0.010 ± 0.004	0.000 ± 0.002
$[6, 10)$	MB^0	367	284	77	8.21	0.989 ± 0.006	0.872 ± 0.064	0.036 ± 0.014	3.011 ± 2.840
	MB					0.999 ± 0.001	0.951 ± 0.028	0.015 ± 0.006	0.000 ± 0.002
$[10, 10^6)$	MB^0	2	1	50	11.43	0.994 ± 0.000	0.867 ± 0.000	0.032 ± 0.000	42.806 ± 0.000
	MB					0.998 ± 0.000	0.962 ± 0.000	0.009 ± 0.000	0.036 ± 0.000

D.5 Data-Poor Groups

Setup. Hierarchical models are routinely fit to groups with fewer observations than parameters, so we quantify posterior quality as a function of the per-group information ratio $\rho = n_i/(d+q)$. For GLMMs, $\rho < 1$ is the local $n < p$ regime. The oracle test sets cover this regime densely: 37–47% of all groups fall below $\rho = 1$ across the three likelihood families (minima 0.26–0.33). We complement ρ with the most adversarial global count, $\gamma = n/p_{\text{all}}$ with $p_{\text{all}} = d + mq + q(q+1)/2 + 1$, i.e. every random effect counted as a free parameter. Even so, γ almost never drops below 1 (minimum 1.1), as identifying the $q \times q$ random-effect covariance requires $m \gtrsim q(q+1)/2$ groups. Metrics aggregate over the parameter entries pooled within each bin, size regimes pooled and NUTS restricted to its converged subset, so entries are single values rather than median $\pm$ MAD.

Results. Table D13–D15 score the local (random-effect) posteriors by ρ bin and the global posteriors by γ bin against the ground truth. `metabeta` matches NUTS in local recovery and calibration in the data-poorest bins ($\rho < 1$) of every family and is never meaningfully worse elsewhere (local NRMSE within 0.02 of NUTS where it trails). For the global parameters, `metabeta`'s coverage carries the IMH head's mild finite-pool overconfidence, while recovery matches NUTS throughout; on Poisson data the head is also what makes the predictive fit usable in the smallest bin (mean LOO-NLL 58.3 raw $\rightarrow$ 1.46, NUTS 1.49). Exactly where amortization is often expected to fail, in groups carrying less than one observation per parameter, `metabeta` delivers honest uncertainty and NUTS-level recovery.

Table D13: Posterior quality in the data-poor regime on the Gaussian oracle benchmarks, size regimes pooled. Top: local (random-effect) posteriors by per-group $\rho = n_i/(d+q)$; #grp and #conv count all groups and those in NUTS-converged datasets. Bottom: global parameters by per-dataset $\gamma = n/p_{\text{all}}$ (every random effect counted as a free parameter); #ds and #conv count datasets. Entries pool the parameter entries within each bin; c_{90} is the coverage of the 90% credible interval (nominally 0.900) and LOO the mean per-dataset LOO-NLL.

			MB^0			MB			NUTS		
$\rho = n_i/(d+q)$	# grp	# conv	EACE	NRMSE	c_{90}	EACE	NRMSE	c_{90}	EACE	NRMSE	c_{90}
$[0, 0.5)$	17029	11114	0.003	0.477	0.898	0.011	0.475	0.891	0.008	0.475	0.895
$[0.5, 1)$	23262	15936	0.001	0.418	0.901	0.005	0.417	0.894	0.003	0.437	0.896
$[1, 2)$	24180	16479	0.001	0.348	0.900	0.004	0.347	0.894	0.002	0.419	0.899
$[2, 5)$	16514	11429	0.013	0.363	0.909	0.008	0.362	0.903	0.012	0.467	0.906
$[5, \infty)$	7901	5753	0.006	0.271	0.904	0.004	0.270	0.899	0.002	0.303	0.899

			MB^0				MB				NUTS			
$\gamma = n/p$	# ds	# conv	EACE	NRMSE	c_{90}	LOO	EACE	NRMSE	c_{90}	LOO	EACE	NRMSE	c_{90}	LOO
$[0, 2)$	197	111	0.011	0.265	0.911	2.449	0.023	0.248	0.866	2.369	0.011	0.246	0.912	2.354
$[2, 4)$	447	315	0.002	0.219	0.901	2.449	0.026	0.220	0.870	2.332	0.004	0.224	0.899	2.349
$[4, 8)$	558	431	0.004	0.186	0.904	2.274	0.023	0.183	0.874	2.216	0.002	0.200	0.901	2.191
$[8, \infty)$	846	653	0.007	0.168	0.906	2.241	0.021	0.160	0.871	2.195	0.003	0.172	0.899	2.252

Table D14: Posterior quality in the data-poor regime on the Bernoulli oracle benchmarks. Layout as in Table D13.

			MB^0			MB			NUTS		
$\rho = n_i/(d+q)$	# grp	# conv	EACE	NRMSE	c_{90}	EACE	NRMSE	c_{90}	EACE	NRMSE	c_{90}
$[0, 0.5)$	17854	12479	0.008	0.826	0.892	0.002	0.823	0.896	0.002	0.836	0.897
$[0.5, 1)$	23797	16903	0.004	0.735	0.902	0.004	0.732	0.902	0.003	0.727	0.903
$[1, 2)$	21924	16822	0.009	0.644	0.898	0.008	0.639	0.896	0.009	0.638	0.898
$[2, 5)$	16630	13615	0.004	0.555	0.898	0.003	0.553	0.900	0.004	0.557	0.900
$[5, \infty)$	8695	7768	0.014	0.484	0.895	0.006	0.476	0.900	0.005	0.473	0.901

			MB^0				MB				NUTS			
$\gamma = n/p$	# ds	# conv	EACE	NRMSE	c_{90}	LOO	EACE	NRMSE	c_{90}	LOO	EACE	NRMSE	c_{90}	LOO
$[0, 2)$	184	92	0.004	0.562	0.904	0.582	0.007	0.563	0.894	0.464	0.006	0.597	0.905	0.463
$[2, 4)$	406	281	0.005	0.453	0.902	0.508	0.012	0.447	0.888	0.450	0.002	0.457	0.898	0.454
$[4, 8)$	550	452	0.006	0.404	0.907	0.477	0.008	0.406	0.887	0.449	0.003	0.417	0.897	0.453
$[8, \infty)$	908	794	0.007	0.308	0.906	0.476	0.004	0.303	0.891	0.459	0.006	0.310	0.902	0.464

Table D15: Posterior quality in the data-poor regime on the Poisson oracle benchmarks. Layout as in Table D13.

			MB^0			MB			NUTS		
$\rho = n_i/(d+q)$	# grp	# conv	EACE	NRMSE	c_{90}	EACE	NRMSE	c_{90}	EACE	NRMSE	c_{90}
$[0, 0.5)$	6938	4907	0.022	0.675	0.890	0.018	0.671	0.890	0.013	0.678	0.894
$[0.5, 1)$	15655	10907	0.005	0.622	0.895	0.004	0.619	0.893	0.003	0.604	0.895
$[1, 2)$	16154	11538	0.012	0.509	0.895	0.014	0.507	0.892	0.016	0.555	0.891
$[2, 5)$	14322	10633	0.025	0.466	0.882	0.021	0.462	0.885	0.019	0.500	0.885
$[5, \infty)$	8487	7018	0.015	0.418	0.895	0.011	0.415	0.896	0.010	0.430	0.896

			MB^0				MB				NUTS			
$\gamma = n/p$	# ds	# conv	EACE	NRMSE	c_{90}	LOO	EACE	NRMSE	c_{90}	LOO	EACE	NRMSE	c_{90}	LOO
$[0, 2)$	95	43	0.010	0.390	0.897	58.288	0.009	0.384	0.877	1.463	0.033	0.355	0.881	1.488
$[2, 4)$	273	181	0.005	0.309	0.899	7.102	0.010	0.312	0.883	1.449	0.005	0.313	0.904	1.457
$[4, 8)$	423	338	0.004	0.248	0.895	10.352	0.015	0.248	0.882	1.368	0.010	0.251	0.890	1.371
$[8, \infty)$	745	608	0.003	0.249	0.902	4.837	0.009	0.244	0.889	1.347	0.007	0.264	0.895	1.330

D.6 Deployment Safeguards

`metabeta`'s inference pipeline ships three per-dataset safeguards that detect (and where possible repair) unreliable posteriors at deployment.

Support validation. Datasets whose dimensions fall outside the training ranges (Appendix F.2) receive an explicit warning.

Acceptance gate. The default IMH head carries its own reliability signal: the chain's mean acceptance rate $\bar{a}$ measures the overlap between the flow proposal and the exact posterior. The suggested pool size $S_{\text{sugg}} = \lceil 700/\bar{a} \rceil$ derived from it (Appendix C.1) is the budget at which the finite-pool calibration error reaches the level of the easy regimes. A mean acceptance below 0.1 flags a large flow–posterior gap; the refined posterior is returned together with a warning that recommends re-running at S_{sugg} or falling back to MCMC (e.g. `MB+NUTS`, Appendix C.2).

PSIS-$\hat{k}$. The pareto-smoothed shape diagnostic $\hat{k}$ (Yao et al., 2018) is an analogous gate (Appendix B.6). Since IMH computes importance weights, their empirical distribution can be approximated with a Pareto distribution with the shape parameter k (with low k indicating relatively even spread): $\hat{k} \leq 0.7$ certifies the flow posterior as a usable proposal, $\hat{k} > 0.7$ returns the posterior with a warning.

MAP-consistency check. Posterior means are compared against the (flow-independent) analytical MAP estimates that already condition the summarizer (Appendix E.3), via the per-dataset score $z = \max_j |\text{mean}_j - \text{MAP}_j| / \text{SD}_j$. A large z means the flow posterior drifted away from its analytical anchor (a symptom of data outside the training distribution). The warning threshold ($z^* = 5$) is calibrated on held-out simulated data (99% quantile). In distribution it fires rarely ($\leq 0.7\%$ of test datasets). Under the design shifts of Appendix D.3 it fires up to $\sim 7\times$ more often (validating the check).

Together with the refinements of Appendix C, these checks make silent failure the exception.

E Architecture and Training Details

E.1 Data Representation and Summary Networks

Observations within each group are concatenated to $\mathbf{D}_i = [\mathbf{y}_i, \mathbf{X}_i, \mathbf{Z}_i]$ (intercept columns excluded) and processed as an unordered set; group structure is a separate tensor dimension, preserving permutation invariance over groups and over observations within groups. Inputs shorter than the model's maximum dimensions are zero-padded, and boolean masks $\boldsymbol{\mu}$ propagate through every layer to exclude padded entries from attention, summaries, flow densities, and losses. To spread the learning signal evenly across predictor positions, a shared random permutation is applied per dataset to the columns of $\mathbf{X}$, $\mathbf{Z}$, $\boldsymbol{\beta}$, $\boldsymbol{\alpha}_{1:m}$, and $\mathbf{S}$.

$\boldsymbol{\Sigma}_l$ and $\boldsymbol{\Sigma}_g$ are Set Transformer encoders (Lee et al., 2019) with identical layout: a linear projection to $d_{\text{model}} = 128$ (GELU), three attention blocks (4 heads, feed-forward width 512), and a learned pool token whose final state is projected to the summary, $h_l = h_g = 32$. The global encoder replaces its first block by an ISAB block with 32 inducing points, which runs on the group summaries before the pool token is inserted; the local encoder uses three MAB blocks. Both posterior networks receive analytical features (Appendix E.3) in addition to the learned summary and the prior embedding.

E.2 Normalizing Flows

Both $\boldsymbol{\Pi}_g$ and $\boldsymbol{\Pi}_l$ are conditional coupling flows (Dinh et al., 2017; Durkan et al., 2019; Papamakarios et al., 2021). Let $\boldsymbol{v}$ denote the vector being modeled and $\mathbf{c}$ the full conditioning vector: for the global flow, $\boldsymbol{v}$ is the unconstrained representation of $\boldsymbol{\vartheta}$ (see below) and $\mathbf{c}$ is the augmented global summary $\mathbf{s}$. For the local flow, $\boldsymbol{v} = \boldsymbol{\alpha}_i$ and $\mathbf{c} = (\boldsymbol{\vartheta}, \mathbf{s}_i)$ plus the per-group features of Appendix E.3. A normalizing flow is a learned invertible map $T(\cdot\,; \mathbf{c}) = T_B \circ \cdots \circ T_1$ from parameter space to a base space. Intermediate states are $\boldsymbol{v}^{(0)} = \boldsymbol{v}$ and $\boldsymbol{v}^{(b)} = T_b(\boldsymbol{v}^{(b-1)}; \mathbf{c})$. The base space is a diagonal Normal density $p_0(\cdot \mid \mathbf{c})$ whose per-coordinate location and scale are predicted from $\mathbf{c}$:

$$\log p_{\boldsymbol{\Pi}}(\boldsymbol{v} \mid \mathbf{c}) = \log p_0\Big(\boldsymbol{v}^{(B)} \mid \mathbf{c}\Big) + \sum_{b=1}^{B} \log \Big|\det J_{T_b}\Big(\boldsymbol{v}^{(b-1)}\Big)\Big| .$$

Density evaluation is one forward pass; sampling draws from p_0 and applies the blocks in reverse. The global flow uses $B = 4$ blocks, the local flow $B = 3$.

Block structure. Each block is *ActNorm* → *Permute* → *DualCoupling*. *ActNorm* (Kingma & Dhariwal, 2018) standardizes each coordinate with a learned shift and scale, $v_j \mapsto (v_j - t_j)/a_j$, initialized from the first training batch; its log-determinant is $-\sum_j \log a_j$. *Permute* applies a fixed random permutation (log-determinant zero), so that every coordinate eventually conditions on every other. *DualCoupling* splits $\boldsymbol{v} = (\boldsymbol{v}_1, \boldsymbol{v}_2)$ at the midpoint and applies two coupling steps, first transforming $\boldsymbol{v}_2$ given $\boldsymbol{v}_1$ and then $\boldsymbol{v}_1$ given the updated $\boldsymbol{v}_2$, so that every coordinate is modified in every block. A single coupling step maps

$$\boldsymbol{v}_2 \mapsto f(\boldsymbol{v}_2;\ g_\psi(\boldsymbol{v}_1, \mathbf{c})) , \qquad \boldsymbol{v}_1 \mapsto \boldsymbol{v}_1,$$

where f acts elementwise and the conditioner g_ψ is a depth-3 residual network (ReLU, width 256, GLU gating by the context, no biases, zero-initialized output layer, so each step starts as the identity). The Jacobian is block-triangular, $\log|\det J| = \sum_j \log|\partial f_j / \partial v_{2,j}|$.

Rational-quadratic splines. Each f_j is a monotone rational-quadratic spline (Durkan et al., 2019) with $K = 8$ bins on a domain $[l, r]$, whose knots and derivatives are predicted by g_ψ. Inside bin k, with input knots c_k and widths $w_k = c_{k+1} - c_k$, output knots c'_k and heights $h_k = c'_{k+1} - c'_k$, secant slopes $s_k = h_k / w_k$, and knot derivatives δ_k (interior ones made positive by a softplus),

$$f_j(v) = c'_k + h_k \cdot \frac{s_k\, \xi^2 + \delta_k\, \xi(1-\xi)}{s_k + (\delta_{k+1} + \delta_k - 2s_k)\, \xi(1-\xi)}, \qquad \xi = \frac{v - c_k}{w_k} \in [0, 1), \tag{4}$$

which is invertible by solving a quadratic in ξ. Outside $[l, r]$ the map is affine with slope and offset predicted per coordinate; the output domain $[c'_1, c'_{K+1}]$ is the affine image of $[l, r]$ and the boundary derivatives equal the tail slope, so the map is C^1 (*adaptive tails*). g_ψ also predicts l and $r - l$ per coordinate, so the bins concentrate where each posterior has its mass (*adaptive domain*).

Masking. Padded coordinates (mask $\boldsymbol{\mu}$) are skipped by *ActNorm*, suppressed in the inputs to g_ψ, passed through the couplings as identity with zero log-determinant, and excluded from the base log-density, so one trained flow handles every (d, q) within its size class.

Unconstrained parameterization. The global flow models $\boldsymbol{\vartheta}$ through a bijection $\boldsymbol{\vartheta} \mapsto \boldsymbol{v} \in \mathbb{R}^{\dim(\boldsymbol{\vartheta})}$. Scales ($\boldsymbol{\sigma}_\alpha$, σ_ε) are mapped by the inverse softplus, $\sigma \mapsto \log(e^\sigma - 1)$, which is close to the identity away from zero and stretches only the region near it. The correlation matrix $\mathbf{R}$ ($q \geq 2$) is represented by its $q(q-1)/2$ canonical partial correlations (Lewandowski et al., 2009): writing $\mathbf{R} = \mathbf{L}\mathbf{L}^\top$ with lower-triangular $\mathbf{L}$, each strictly lower entry is normalized by the norm its row has left, $\tilde{\rho}_{kj} = L_{kj}/\sqrt{1 - \sum_{j'<j} L_{kj'}^2} \in (-1, 1)$ for $1 \leq j < k \leq q$, and mapped to $\gamma_{kj} = \operatorname{artanh}(\tilde{\rho}_{kj}) \in \mathbb{R}$. The inverse rebuilds $\mathbf{L}$ row by row from $\tilde{\rho}_{kj} = \tanh(\gamma_{kj})$ with unit-norm rows, so every $\boldsymbol{\gamma} \in \mathbb{R}^{q(q-1)/2}$ yields a valid correlation matrix and $\boldsymbol{\gamma} = \mathbf{0}$ gives the identity. Samples are mapped back to the constrained scale before any evaluation, and so is the density: with $\mathbf{r}$ the lower triangle of $\mathbf{R}$, the reported global density is

$$\log q(\boldsymbol{\vartheta}) = \log p_{\boldsymbol{\Pi}_g}(\boldsymbol{v} \mid \mathbf{s}) - \sum_\sigma \log \operatorname{softplus}'(v_\sigma) - \log \left| \det \frac{\partial \mathbf{r}}{\partial \boldsymbol{\gamma}} \right|,$$

$$\log \left| \det \frac{\partial \mathbf{r}}{\partial \boldsymbol{\gamma}} \right| = \tfrac{1}{2} \sum_{k>j} (q - j + 1) \log\bigl(1 - \tanh^2 \gamma_{kj}\bigr),$$

where the sum over σ runs over the scale coordinates. This $q(\boldsymbol{\vartheta})$ lives in the same coordinates as prior and likelihood and is the density that enters the IMH acceptance ratio (Appendix C.1) and the evidence weights (Appendix B.6).

E.3 Analytical Context Features

Before any data enter the networks, we compute classical point estimates of every model parameter: $\hat{\boldsymbol{\beta}}$, $\hat{\mathbf{S}}$, $\hat{\phi}$, and per-group BLUPs $\hat{\boldsymbol{\alpha}}_i$ (Laird & Ware, 1982; Searle et al., 1992). These are fed to the normalizing flows as conditioning inputs. All estimates are closed-form expressions or derived from short fixed-length iterations (vectorized over datasets and groups). They depend only on the data and its stored prior, so they are computed once per dataset and cached.

Gaussian outcomes. Variance components have (near) closed-form estimators (Henderson, 1953; Searle et al., 1992), and we use a four-stage pipeline.

1. *Residual variance.* Outcomes and predictors are projected onto the within-group complement of $\mathbf{Z}_i$ (group-demeaning when $q = 1$), such that the random effects vanish; pooled OLS on the projected data yields $\hat{\phi} = \hat{\sigma}_\varepsilon$.
2. *Random-effect covariance.* Per-group OLS estimates of the random effects are combined into a method-of-moments estimate $\hat{\mathbf{S}}$ that subtracts the share of their spread (Henderson, 1953; Cochran, 1954).
3. *Fixed effects and BLUPs.* Given $(\hat{\sigma}_\varepsilon, \hat{\mathbf{S}})$, generalized least squares yields $\hat{\boldsymbol{\beta}}$, and each group's random effect is the shrinkage compromise between its own data and the population (Robinson, 1991),
$$\hat{\boldsymbol{\alpha}}_i = \bigl(\hat{\sigma}_\varepsilon^2 \hat{\mathbf{S}}^{-1} + \mathbf{Z}_i^\top \mathbf{Z}_i\bigr)^{-1} \mathbf{Z}_i^\top \bigl(\mathbf{y}_i - \mathbf{X}_i \hat{\boldsymbol{\beta}}\bigr).$$
This pulls sparsely observed groups toward zero and densely observed ones toward their own least-squares fit.
4. *EM polish.* A few EM iterations (Dempster et al., 1977) refine $\hat{\mathbf{S}}$ and $\hat{\sigma}_\varepsilon$, since the moment estimate truncates at zero exactly where between-group variation is weakest.

Bernoulli and Poisson outcomes. A pooled GLM fit (McCullagh & Nelder, 1989) that ignores the group structure gives initial fixed effects and a Pearson dispersion statistic $\hat{\phi}$. The latter is kept as a global feature that signals excess variability that the random effects must absorb. Penalized quasi-likelihood (PQL; Breslow & Clayton, 1993) then rewrites the model locally as a weighted

LMM, and the Gaussian pipeline supplies a first estimate $\hat{\mathbf{S}}$. An alternating Newton–Laplace–GLS loop refines all estimates jointly: damped Newton steps move each group to its posterior mode $\hat{\boldsymbol{\alpha}}_i$, a Laplace approximation step re-estimates the covariance from the modes and their curvature, $\hat{\mathbf{S}} = \frac{1}{m}\sum_i(\hat{\boldsymbol{\alpha}}_i\hat{\boldsymbol{\alpha}}_i^\top + \mathbf{H}_i^{-1})$, and a GLS step updates $\hat{\boldsymbol{\beta}}$. Each pass is one EM step on the Laplace-approximated marginal likelihood, targeting the same fixed point as `lme4`'s outer optimizer (Bates et al., 2015). The loop is stopped by a batch-wide convergence test (and has at most 6 iterations).

Prior calibration (empirical-Bayes MAP). The estimates so far ignore the prior, but the flows approximate a Bayesian posterior: with few groups or weakly identified parameters the posterior sits between likelihood and prior, and an unpenalized anchor would point the flows to the wrong place exactly where prior guidance matters most. A final stage therefore moves the estimates to the mode of the Laplace-approximated marginal posterior under the dataset's stored prior. For Gaussian outcomes this is a cheap low-dimensional optimization over the variance components (which is why this family's features are the fastest to compute, Appendix B.5). For Bernoulli and Poisson, $(\hat{\boldsymbol{\beta}}, \log\hat{\boldsymbol{\sigma}}_\alpha)$ are optimized jointly for 24 Adam steps, re-solving the group modes with a few damped Newton steps inside every iteration; a continuation schedule releases a cap on $\hat{\boldsymbol{\sigma}}_\alpha$ gradually so the scales cannot inflate before $\hat{\boldsymbol{\beta}}$ settles. The result is accepted only where it improves the Laplace objective.

Context features. The global flow receives $\hat{\boldsymbol{\beta}}$, $\hat{\boldsymbol{\sigma}}_\alpha$, $\hat{\phi}$, and the lower triangle of the estimated correlation matrix $\hat{\mathbf{R}}$. The local flow receives BLUPs $\hat{\boldsymbol{\alpha}}_i$, the BLUP's standard deviation, and the shrinkage factor $\hat{\lambda}_{ik} = 1 - \mathrm{Var}(\hat{\alpha}_{ik})/\hat{\sigma}^2_{\alpha,k}$ for random effect k. The latter measures how firmly the group's own data pin down the random effect ($\hat{\lambda} \to 1$ group-dominated, $\hat{\lambda} \to 0$ population-dominated). For Gaussian outcomes, BLUPs are computed in closed form, for discrete outcomes they are the Laplace modes from the pipeline above. All analytical features are conditioning inputs the flows are free to overrule, not the final inference.

E.4 Optimization and Hyperparameters

All models are trained with Schedule-Free AdamW (Defazio et al., 2024) ($\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay 0, learning rate 3×10^{-4}, gradient clipping at norm 1.0, batch size 32). The global and local losses are summed with equal weight; local losses are averaged over active groups within each dataset. Each model is trained for $8{,}000$–$16{,}000$ epochs of $4{,}096$ datasets precomputed by the simulator. This totals approximately 33–66M unique training datasets and 1–2M gradient steps, on a single NVIDIA H100 80 GB GPU (4 CPU cores, 64 GB RAM) in 3–5 days.

Efficient mini-batch construction. Because datasets vary in group count m and maximum per-group observation count $n_{i,\max}$, naive random batching wastes memory on zero-padding. We use a *sortish* batch sampler: within consecutive windows of 50 mini-batches, indices are sorted by m (primary key) and $n_{i,\max}$ (secondary key). The resulting batches are then globally reshuffled. Sorting by m first matters because m spans a wider relative range than $n_{i,\max}$ and dominates cross-axis padding waste.

F Simulator and Datasets

This appendix details the simulation procedure sketched in Section 2.4: the literature survey that grounds the simulation ranges (Appendix F.1), the size classes and their sampling distributions (Appendix F.2), the prior hyperparameters (Appendix F.3), the two predictor sources (Appendix F.4, Appendix F.5), the analytical standardization that keeps parameters and hyperparameters coherent under z-scoring (Appendix F.6), and the held-out test collection (Appendix F.7).

F.1 A Survey of Published GLMM Analyses

To quantify how much of published GLMM practice `metabeta`'s model class covers, we surveyed 1,133 randomly sampled papers from the cognitive and behavioral sciences[2]. An LLM pipeline (Claude Haiku 4.5 for screening, Claude Sonnet 5 for extraction) identified 576 papers (51%) reporting at least one GLMM analysis and extracted the model equation for 500 of them. Each extraction stores the verbatim equation together with its text or code reference for auditability, and a manual check of 20 randomly drawn extractions against their source papers found no extraction errors; the full dataset is released alongside our code (`model_equations.csv`).

The extracted models contain on average $d = 5.14$ fixed effects including the intercept (95% CI $[4.74, 5.55]$; median 4) and $q = 2.91$ random effects ($[2.69, 3.13]$; median 2); Gaussian (53%), Bernoulli (41%), and Poisson (0.4%) likelihoods together account for 94% of all models (Figure F1). The remaining 6% are predominantly ordinal (cumulative-link) models for rating scales, alongside a few heavy-tailed response-time (ex-Gaussian, log-normal) and bounded-proportion (beta) likelihoods. Taken together, 83% of the surveyed analyses fall within `metabeta`'s supported model class ($d \leq 16$, $q \leq 5$, Gaussian/Bernoulli/Poisson).

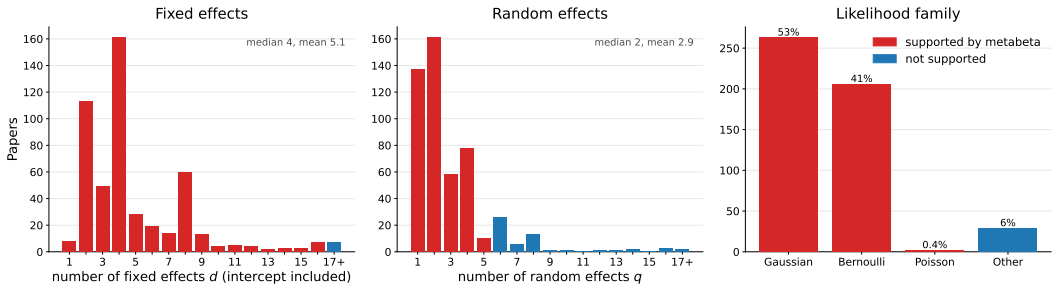


Figure F1: Survey of published GLMM analyses: distributions of the fixed-effect count d (intercept included, as throughout the paper), the random-effect count q, and the likelihood family over the 500 extracted models. Red marks analyses within `metabeta`'s supported ranges ($d \leq 16$, $q \leq 5$, Gaussian/Bernoulli/Poisson), blue those outside.

F.2 Size Classes

Every dataset first draws its size class (Table F1) and then its structural dimensions: the fixed-effect count d, the random-effect count q, the group count m, and the group sizes $n_{1:m}$. The classes differ only in the band for d and the cap $q_{\max}$; all share one data budget of $m \leq 200$ groups, $n_i \leq 150$ observations per group, and $N_{\max} = 3{,}000$ observations in total.

Identifiability. Estimating d fixed effects requires $m > d$ groups, and estimating a $q \times q$ random-effect covariance requires $m > q(q+1)/2$. The group count is therefore floored at

$$m_{\text{low}} = \max\big(m_{\min},\ \max(d,\ \tfrac{q(q+1)}{2}) + 4\big),$$

which leaves at least four between-group degrees of freedom above the tighter constraint. Emulated designs (Appendix F.5) can deliver fewer groups than requested, so a post-hoc clamp trims q to the

[2] *J. Exp. Psychol.* (General & LMC; 121), *Cognition* (67), *Cogn. Sci.* (54), *Psychol. Sci.* (47), *Behav. Res. Methods* (36), *Psychon. Bull. & Rev.* (31), *Cogn. Psychol.* (30), *Judgm. Decis. Mak.* (21), *J. Mem. Lang.* (20), *Nat. Hum. Behav.* (19), and 27 other venues (54).

largest value identifiable from the *realized* m, $q^* = \max\big(1,\ \lfloor(\sqrt{1+8(m-4)}-1)/2\rfloor\big)$. Within groups, the minimum size is $\max(n_{\min},\ q+2)$, so every group keeps at least two residual degrees of freedom after fitting its own random effects. The floor rises with d and q: in the realized test sets, the median group count grows from 21 in the *small* class to 42 in *huge*. The cap $N_{\max}$ only bounds the largest datasets; it binds for well under 1% of oracle test datasets, whose median total is 406–512 observations.

Sampling distributions.

- *Fixed effects* d: uniform integer over the class band, giving equal coverage across complexity levels.
- *Random effects* q: log-uniform over $[1, \min(q_{\max}, d)]$. This concentrates coverage on the small random-effect dimensions that dominate applied practice (Appendix F.1: median 2; 71% of surveyed models have $q \le 3$ and 89% have $q \le 5$), and the cap at d keeps the random-effect design a sub-design of the fixed-effect one.
- *Groups* m: a two-component mixture. With probability 0.8, a mode-preserving Beta (defined in Appendix F.3) on $[m_{\text{low}}, m_{\max}]$ with concentration $\kappa = 8$ and mode $m_{\text{low}} + \min\big(0.08\,(m_{\max} - m_{\text{low}}),\ 20\big)$, floored to an integer; this places most mass near the identifiability floor, in line with the test collection (61% of its datasets have $m \le 20$). With probability 0.2, m is uniform over $[\max(m_{\text{low}}, 60),\ m_{\max}]$, which keeps many-group designs represented.
- *Group sizes* n_i *for synthetic designs*: a mean group size $\bar{n}$ is drawn on a log scale over $[n_{\min},\ \min(n_{i\,\max},\ N_{\max}/m)]$, at position $0.15 + 0.65\,\tilde{m}$ of the log-span plus Gaussian noise with scale 0.25 of the span, where $\tilde{m} \in [0,1]$ is the log-scaled position of m in its range. With probability 0.65 every group receives $\bar{n}$ observations; otherwise the budget $m\bar{n}$ is split by $\text{Dirichlet}(c\,\mathbf{1}_m)$ proportions with $c \sim \text{LogUniform}(8, 40)$.
- *Group sizes* n_i *for emulated designs* (the source of all oracle test sets): only m and the total budget are requested; the hierarchical bootstrap of Appendix F.7 allocates group sizes, clipped to what each source group actually contains. Group sizes are bounded by real data rather than by a parametric family, which reproduces the heavy right tail and imbalance of real hierarchical data.

F.3 Prior Hyperparameters

Each dataset draws its prior families independently for the fixed-effect coefficients (Normal 0.80, Student-t 0.20), the random-effect scales (half-Normal 0.60, half-Student-t 0.30, Exponential 0.10). Under the Gaussian likelihood, the residual scale is added (half-Normal 0.50, half-Student-t 0.40, Exponential 0.10). All Student-t families have 5 degrees of freedom. The hyperparameters are then drawn as follows.

Locations. Locations are sampled from a spike-and-slab distribution: each component of $\boldsymbol{\nu}_\beta \in \mathbb{R}^d$ is zero with probability 0.90 and otherwise drawn from $\text{Laplace}(0, s)$, with $s = 0.7$ for the Bernoulli likelihood and $s = 0.5$ otherwise. The network thus sees mostly zero-centered priors with occasional strongly shifted ones.

Table F1: Size-class parameter ranges. d is the fixed-effect dimension band (intercept included) and $q_{\max}$ the cap on the random-effect dimension, drawn log-uniformly from $[1, \min(q_{\max}, d)]$. All regimes share the same data budget: $m \in [5, 200]$ groups, $n_i \in [5, 150]$ observations per group, and a total-observation cap of $N_{\max} = 3{,}000$.

regime	d	$q_{\max}$
small	1–4	2
medium	5–8	3
large	9–12	4
huge	13–16	5

Scales. Scale hyperparameters follow a *mode-preserving Beta*: a Beta distribution rescaled to $[a, b]$ with mode ω and concentration κ. Writing $t = (\omega - a)/(b - a)$ for the mode's relative position,

$$\tau \sim a + (b - a) \cdot \mathrm{Beta}\big(1 + (\kappa - 1)\, t,\ 1 + (\kappa - 1)(1 - t)\big),$$

so $\kappa = 1$ gives the uniform distribution on $[a, b]$ and larger κ concentrates mass around ω without displacing it. Ranges are tighter for Bernoulli and Poisson likelihoods to prevent extreme linear predictors:

- $\boldsymbol{\tau}_\beta$: $[0.01, 4.0]$, mode 0.5, $\kappa = 8$ (Gaussian); $[0.01, 3.0]$, mode 0.8, $\kappa = 6$ (Bernoulli); $[0.01, 1.5]$, mode 0.5, $\kappa = 6$ (Poisson).
- $\boldsymbol{\tau}_\sigma$: $[0.01, 5.0]$, mode 1.0 (Gaussian); $[0.01, 2.0]$, mode 0.7 (Bernoulli); $[0.01, 1.0]$, mode 0.3 (Poisson); $\kappa = 5$ throughout.
- τ_ε: $[0.05, 8.0]$, mode 2.0, $\kappa = 5$ (Gaussian likelihood only).

Sampled scales σ_ε and σ_{α_j} are floored at 10^{-3} to keep the likelihood well defined. The floor is not mirrored in the `PyMC` priors of the baselines, since raising it would remove high-R^2 datasets from the benchmark and change the training distribution seen by `metabeta`.

Correlations. Random effects are correlated in only part of the draws: never for $q = 1$, with probability 0.8 for $q = 2$, and with probability 0.5 for $q \geq 3$. When they are, the correlation matrix is drawn from $\mathrm{LKJ}(\eta)$ with $\eta \sim \mathcal{U}(1, 2)$; otherwise it is the identity.

Outcome calibration. Prior draws that would produce degenerate outcomes are corrected before $\mathbf{y}$ is sampled. Under the *Gaussian* likelihood, an explanatory-power ceiling $R^2_{\max} \sim \mathcal{N}(0.68 + 0.015\,(d - 1) + 0.010\,(q - 1),\ 0.10^2)$, clipped to $[0.45, 0.95]$, is drawn per dataset; whenever $\mathbb{V}(\boldsymbol{\eta})/(\mathbb{V}(\boldsymbol{\eta}) + \sigma^2_\varepsilon)$ exceeds it, σ_ε is inflated to meet the ceiling. This bounds the signal-to-noise ratio of the Gaussian benchmarks, and it loosens with design size because more predictors legitimately explain more variance. Under the *Bernoulli* and *Poisson* likelihoods, the parameters are instead redrawn (up to 20 times) while too large a share of the linear predictor is extreme: $|\eta| > 10$ for more than 5% of observations, or $\eta > 10$ for more than 1%, respectively. Any remaining excess is removed by scaling all regression parameters so that $\mathrm{sd}(\boldsymbol{\eta})$ meets a per-dataset cap drawn from $\mathcal{U}(2, 4)$ (Bernoulli) or $\mathcal{U}(1, 2)$ (Poisson); Poisson predictors are additionally clipped to ± 5 after standardization.

F.4 SCM Generator

The SCM Generator samples predictor matrices from structural causal models whose graph, weights, activations and marginal transforms are re-drawn on every call, so each dataset carries a distinct dependency structure. Per dataset it picks one of two variants, a layered MLP-SCM (probability 0.6) or a sparse DAG-SCM (0.4), and one of three equiprobable presets that jointly set the activation pool and the root-cause distribution, trading smoothness against variability.

Root causes. In the MLP-SCM, each exogenous column is assigned independently to one of nine marginal families (Gaussian, uniform, multinomial, Zipf, gamma, log-normal, beta, Student-t, Gaussian mixture), so right-skewed, heavy-tailed, bounded and multimodal marginals coexist within a dataset; the smoothest preset uses uniform causes only. The DAG-SCM has no separate cause stage: its root nodes are standard normal, and all structure arises during propagation.

Propagation. The MLP-SCM passes the root causes through ℓ blocks of `Linear` $\to$ `NoiseLayer` $\to$ `Activation`. Block-diagonal weight matrices cluster features into correlated groups, and a noiseless pilot pass scales the additive noise at each depth to a fixed fraction of the signal's interquartile range. Three optional stages add structure the weights alone do not produce: shared latent noise over random feature subsets, a low-rank factor injection, and per-feature monotone maps that diversify marginal shapes while leaving the rank structure, and hence the dependency graph, intact. The DAG-SCM instead samples a scale-free (Barabási–Albert) or sparse random (Erdős–Rényi) graph over the observed features plus a set of latent nodes, and evaluates $x_i = \phi(\mathbf{W}\mathbf{x}_{\mathrm{pa}(i)}) + \varepsilon_i$ in topological order, reading observed features off leaf nodes wherever the graph supplies enough of them. It shares the noise calibration but none of the optional stages, and contributes sparse, explicitly causal dependencies alongside the dense ones of the layered variant.

Activation pool. Each layer draws from 25 fixed nonlinearities, both standard (ReLU, Tanh, SiLU) and irregular (Abs, Sine, Mod, Ceil, Sign), and from random Fourier feature approximations of Gaussian process draws under squared-exponential, Matérn and fractional kernels. `RandomChoice` layers apply different activations to different feature subsets within one layer. The preset decides how much of the pool is available, from a small, smooth, GP-dominated selection to the full set.

Post-hoc transforms. Finally, a random subset of continuous outputs is replaced by discrete or bounded variants (Table F2). Each transform first mixes several SCM features with Dirichlet weights and then applies its mapping, either per column or as a block that emits several dummy-coded or ordinal columns at once. The ordinal block thresholds one shared latent direction, so its columns are strongly rank-correlated, as multiple ordinal predictors typically are in real data. The result is mixed-type feature matrices with nonlinear inter-predictor dependencies that parametric families cannot reproduce.

Table F2: Post-hoc feature transforms of the SCM Generator and the predictor type each one emits.

Transform	Output
`Threshold`	Binary (thresholded at zero)
`MultiThreshold`	Ordinal integer (Gaussian-sampled thresholds)
`QuantileBins`	Discrete bins (data-driven quantile cut points)
`Clamp`	Continuous with random quantile floor/ceiling
`CensoredFloor`	Left-censored continuous (detection-limit effect)
`Categorical` / `CategoricalBlock`	Dummy-coded factor(s), mutually exclusive within group
`OrdinalBlock`	Correlated ordinal integers from a shared latent direction
`Poisson` / `NegativeBinomial`	Integer counts

F.5 EMULATOR

The Emulator draws predictors from a bank of 596 real datasets from PMLB (Romano et al., 2022), SRM (Lichtenberg & Şimşek, 2017) and UCI (Dua & Graff, 2019), each preprocessed by the pipeline of Appendix G and stored as a compressed array; the test collection of Appendix F.7 is excluded from the bank. Existing hierarchical grouping is preserved where present, otherwise groups are assigned at random. For a requested predictor count d, a source dataset with d_{src} predictors is selected with probability proportional to $\min(d_{\mathrm{src}}, d)/d$: sources with $d_{\mathrm{src}} \geq d$ receive full weight and fill all $d-1$ predictor slots with real columns, while smaller sources receive proportionally less weight and have the shortfall padded with SCM-generated columns. This soft weighting avoids a hard eligibility cutoff, which would shrink the source pool sharply at values of d that few datasets reach and thereby make source diversity uneven across d.

F.6 STANDARDIZATION

All continuous predictors are z-standardized over groups and observations, as is standard for GLMMs, and for training stability so are continuous outcomes y. To keep the dependence structure intact, parameters and hyperparameters are standardized analytically during training and un-standardized after sampling:

$$\beta_k^* = \beta_k \frac{\sigma_{x_k}}{\sigma_y}, \qquad \alpha_{ik}^* = \alpha_{ik} \frac{\sigma_{z_k}}{\sigma_y} \sim \mathcal{N}\left(0, \sigma_k^{*2}\right),$$
$$\sigma_k^{*2} = \sigma_k^2 \frac{\sigma_{z_k}^2}{\sigma_y^2}, \quad \sigma_\varepsilon^{*2} = \frac{\sigma_\varepsilon^2}{\sigma_y^2},$$

where σ_{x_k} and σ_y are the standard deviations of the kth predictor and the outcome, and β_k^* is the standardized kth slope. The intercepts require special care:

$$\beta_0^* = \frac{\beta_0 + \sum_{k=1}^{d} \mu_{x_k} \beta_k - \mu_y}{\sigma_y},$$
$$\alpha_{i0}^* = \frac{\alpha_{i0} + \sum_{k=1}^{q} \mu_{z_k} \alpha_{ik}}{\sigma_y} = \frac{\sum_{k=0}^{q} \mu_{z_k} \alpha_{ik}}{\sigma_y} \sim \mathcal{N}\left(0, \sigma_0^{*2}\right),$$

where μ_{x_k} is the mean of the kth predictor. Due to the sum term in the latter,

$$\sigma_0^{*2} = \mathbb{V}\left(\alpha_{i0}\right) + \mathbb{V}\left(\sum_{k=1}^{q} \mu_{z_k} \alpha_{ik}\right) + 2 \cdot \mathrm{Cov}\left(\alpha_{i0}, \sum_{k=1}^{q} \mu_{z_k} \alpha_{ik}\right),$$

which is equivalent to summing up the covariance matrix of the random vector $\boldsymbol{\mu}_z \odot \boldsymbol{\alpha}_i$.

Proof:

$$\begin{aligned}
y_{ij}^* &= \frac{y_{ij} - \mu_y}{\sigma_y} \\
&= \frac{1}{\sigma_y}\left(\beta_0 + \sum_{k=1}^{d} x_{ijk}\beta_k + \alpha_{i0} + \sum_{k=1}^{q} z_{ijk}\alpha_{ik} + \varepsilon_{ij} - \mu_y\right) \\
&\overset{!}{=} \beta_0^* + \sum_{k=1}^{d} x_{ijk}^*\beta_k^* + \alpha_{i0}^* + \sum_{k=1}^{q} z_{ijk}^*\alpha_{ik}^* + \varepsilon_{ij}^* \\
&= \beta_0^* + \sum_{k=1}^{d}\left(\frac{x_{ijk} - \mu_{x_k}}{\sigma_{x_k}}\right)\beta_k^* + \alpha_{i0}^* + \sum_{k=1}^{q}\left(\frac{z_{ijk} - \mu_{z_k}}{\sigma_{z_k}}\right)\alpha_{ik}^* + \varepsilon_{ij}^* \\
&= \beta_0^* + \sum_{k=1}^{d}\left(\frac{x_{ijk} - \mu_{x_k}}{\sigma_{x_k}}\right)\left(\beta_k\frac{\sigma_{x_k}}{\sigma_y}\right) + \alpha_{i0}^* + \sum_{k=1}^{q}\left(\frac{z_{ijk} - \mu_{z_k}}{\sigma_{z_k}}\right)\left(\alpha_{ik}\frac{\sigma_{z_k}}{\sigma_y}\right) + \frac{\varepsilon_{ij}}{\sigma_y} \\
&= \beta_0^* - \sum_{k=1}^{d}\frac{\mu_{x_k}\beta_k}{\sigma_y} + \sum_{k=1}^{d}\frac{x_{ijk}\beta_k}{\sigma_y} + \alpha_{i0}^* - \sum_{k=1}^{q}\frac{\mu_{z_k}\alpha_{ik}}{\sigma_y} + \sum_{k=1}^{q}\frac{z_{ijk}\alpha_{ik}}{\sigma_y} + \frac{\varepsilon_{ij}}{\sigma_y} \\
&= \frac{\beta_0 - \mu_y}{\sigma_y} + \sum_{k=1}^{d}\frac{x_{ijk}\beta_k}{\sigma_y} + \frac{\alpha_{i0}}{\sigma_y} + \sum_{k=1}^{q}\frac{z_{ijk}\alpha_{ik}}{\sigma_y} + \frac{\varepsilon_{ij}}{\sigma_y} \qquad \square
\end{aligned}$$

The distributions of the standardized random effects and noise follow from scaling normal variables, and the random-intercept variance from the variance of a sum (Wasserman, 2010).

Table F3: Real-world test datasets: source package, likelihood, fixed-effect count d (intercept included), groups m, observations n, range of observations per group, and group-size entropy ratio H (1 = balanced).

Dataset	Package	Likelihood	d	m	n	n/group	H
Orange	datasets	Gaussian	2	5	35	7–7	1.000
Machines	nlme	Gaussian	3	6	54	9–9	1.000
Indometh	datasets	Gaussian	2	6	66	11–11	1.000
Oats	nlme	Gaussian	4	6	72	12–12	1.000
ergoStool	nlme	Gaussian	4	9	36	4–4	1.000
Pastes	lme4	Gaussian	3	10	60	6–6	1.000
Pixel	nlme	Gaussian	3	10	102	4–14	0.972
Theoph	datasets	Gaussian	3	12	132	11–11	1.000
sleepstudy	lme4	Gaussian	2	18	180	10–10	1.000
Penicillin	lme4	Gaussian	6	24	144	6–6	1.000
Oxboys	nlme	Gaussian	10	26	234	9–9	1.000
Orthodont	nlme	Gaussian	3	27	108	4–4	1.000
Exam	mlmRev	Gaussian	11	65	4059	2–198	0.974
Gcsemv	mlmRev	Gaussian	3	73	1703	2–93	0.934
Hsb82	mlmRev	Gaussian	5	160	7185	14–67	0.993
MathAchieve	MEMSS	Gaussian	5	160	7185	14–67	0.993
InstEval	lme4	Gaussian	12	2972	73421	1–92	0.978
analcatdata_boxing1	PMLB	Bernoulli	3	10	120	12–12	1.000
analcatdata_boxing2	PMLB	Bernoulli	3	11	132	12–12	1.000
cbpp	lme4	Bernoulli	4	15	842	26–96	0.968
cane	boot	Bernoulli	5	45	21266	232–826	0.988
bacteria	MASS	Bernoulli	6	50	220	2–5	0.994
respiratory	geepack	Bernoulli	7	56	444	4–8	0.999
Contraception	mlmRev	Bernoulli	6	60	1934	2–118	0.948
OME	MASS	Bernoulli	6	77	4196	19–134	0.970
guImmun	mlmRev	Bernoulli	16	161	2159	1–55	0.961
guPrenat	mlmRev	Bernoulli	22	161	2449	1–50	0.965
respInf	gamlss.data	Bernoulli	9	275	1200	1–6	0.984
VerbAgg	lme4	Bernoulli	7	316	7584	24–24	1.000
ohio	geepack	Bernoulli	3	537	2148	4–4	1.000
muscatine	geepack	Bernoulli	5	4856	9856	1–3	0.989
Mmmec	mlmRev	Poisson	3	9	354	3–95	0.824
Arabidopsis	lme4	Poisson	5	9	625	32–137	0.961
analcatdata_apnea1	PMLB	Poisson	3	19	475	25–25	1.000
analcatdata_apnea2	PMLB	Poisson	3	19	475	25–25	1.000
Salamanders	glmmTMB	Poisson	12	23	644	28–28	1.000
Owls	glmmTMB	Poisson	5	27	599	4–52	0.964
epil	MASS	Poisson	5	59	236	4–4	1.000
grouseticks	lme4	Poisson	4	118	403	1–10	0.969
Chem97	mlmRev	Poisson	4	2410	31022	1–188	0.946

F.7 Test Datasets

The held-out collection comprises 40 real hierarchical datasets (Table F3): 36 from standard R mixed-effects packages (`boot`, `datasets`, `gamlss.data`, `geepack`, `glmmTMB`, `lme4`, `MASS`, `MEMSS`, `mlmRev`, and `nlme`) and four (the `analcatdata` apnea and boxing sets) from PMLB (Romano et al., 2022), all preprocessed as in Appendix G. In the table, d counts the fixed effects including the intercept (after dummy-coding categoricals), m the groups, n the observations, and $H = -\sum_i p_i \log p_i / \log m$ with $p_i = n_i/n$ is the group-size entropy ratio, where $H = 1$ indicates perfectly balanced groups and $H \to 0$ extreme imbalance. Aggregate binomial datasets are expanded to Bernoulli observations before these summaries are computed. `InstEval`, `Chem97` and `muscatine` exceed the group budget of Table F1 and enter the test sets only through group subsampling.

Hierarchical bootstrap. Both test set types draw their designs from this collection with the hierarchical bootstrap of Algorithm 1. Oracle test sets then replace $\mathbf{y}$ with simulated outcomes; real-world test sets keep the original outcome and sample no regression parameters. Their model specification follows `Bambi`'s weakly informative defaults (Capretto et al., 2022), e.g. $\mathcal{N}(0, 2.5)$ fixed effects and half-Normal(2.5) random-effect scales, with correlated random effects and an LKJ(1) prior on $\mathbf{R}$ whenever $q \geq 2$; all inference methods receive the identical specification. Gaussian outcomes are rescaled to unit standard deviation, which leaves posterior shapes unchanged.

Algorithm 1 Hierarchical bootstrap (`Subsampler`)

Require: Source dataset with groups $\{G_j\}$, target counts (d, m, n)
Ensure: Subsampled $(\mathbf{X}, \mathbf{y}, \mathbf{ns})$ with original $\mathbf{y}$

1: Randomly select m eligible groups (groups with $\geq n_{\min}$ observations)
2: Sample per-group observation budgets: seed every group at $n_{\min}$, then allocate the remainder by repeated Dirichlet$(c\,\mathbf{1}_m)$–multinomial draws with concentration $c \sim \mathcal{U}(2, 20)$, clipped to each group's available count
3: Randomly subsample ns_i observations from group G_i without replacement
4: Randomly select $d - 1$ predictor columns; append intercept column
5: Standardize predictors; return original $\mathbf{y}$ unchanged

G Preprocessing Pipeline

G.1 Design Philosophy

The preprocessing pipeline is a *canonicalization map*, not a statistical cleaning step: it projects raw inputs onto a common representation that is numerically stable for the Set Transformer, consistent between simulation and deployment, and interpretable at the level of regression coefficients. All transforms are monotone (log1p, clipping) or scale-equivariant (z-standardization), so the qualitative structure of the likelihood — in particular the monotonicity of the predictor–outcome mapping — is preserved; mild biases (e.g. a small slope bias from winsorization) are accepted in exchange for reduced representational heterogeneity. The pipeline is *stateful*: all statistics used to transform data (column medians, winsorization bounds, encoder vocabularies) are fitted once and replayed identically on new data via `fit`/`transform` methods analogous to `sklearn` estimators, eliminating covariate shift between training and deployment.

G.2 Pipeline Steps

Table G1 lists every step in order with its motivation; the first six define the structure of the dataset and contain the only row or column removals, the remaining seven are pure transformations. Three steps carry decisions that the table cannot convey in a line.

Outcome type detection. The outcome is coerced to a numeric array (two-class strings are mapped to $\{0, 1\}$; rows with non-finite `y` are removed) and its type determines the likelihood family: binary outcomes stay integer indicators, counts (non-negative integers with zeros) stay raw integers for the Poisson likelihood, continuous outcomes are z-standardized in the final step, and multiclass strings are treated as pseudo-continuous under the Gaussian likelihood. Datasets without a designated target are kept as design-matrix sources for the Emulator only.

Grouping variable detection. The grouping variable is (i) the explicitly specified column if given; otherwise (ii) the top-scoring integer/categorical column under a heuristic rewarding intermediate cardinality (5–200 levels), average group size near 20, low imbalance, and clustering-typical names (`subject`, `school`, `site`, ...); otherwise (iii) a categorical predictor with more than `max_categories` levels and $\geq$2 observations per level, since high-cardinality categoricals are misspecified as fixed effects (near-rank-deficient design) and incompatible with the network's input contract. The dataset is sorted by the resulting integer group codes; crossed or nested structures are approximated by the single dominant factor. Categorical predictors with more than `max_categories` levels keep their K most frequent levels; the rest are pooled into `other` if their combined prevalence is at least 5% and are otherwise removed (matching `forcats::fct_lump_n` in R).

Numeric transforms. Numeric predictors are winsorized to $[\mu \pm 4\sigma]$ (training-data statistics): extreme values push the summarizer outside its training distribution, and the conservative threshold limits the impact on skewed predictors. Binary predictors pass through unchanged, since standardization would destroy the reference-level interpretation. Count-like predictors with $|\text{skewness}| > 1$ are $\log(x{+}1)$-transformed then z-standardized (their structural $\mathbb{N}_0$ support produces systematic right-skew); low-skew counts and continuous predictors are z-standardized only, because their skew has no consistent direction and direction-specific transforms would heterogenize the representation. Categoricals are encoded as treatment contrasts with the training-modal category as reference level (matching R's `contr.treatment`); unseen categories at deployment map to the zero vector, i.e. the reference effect.

Table G1: Preprocessing steps in order of application, with the decision taken and its primary motivation. Steps 1–6 fix the structure of a dataset and contain the only row or column removals; steps 7–13 are pure transformations.

	Step	Decision	Primary motivation
1	Sentinel replacement	$-999 \to$ `NaN`	avoid contaminating distribution estimates
2	Heavy-column removal	drop if $> 25\%$ missing	imputation unreliable (Sterne et al., 2009)
3	Outcome coercion	detect binary / count / continuous	selects the likelihood family
4	Grouping detection	explicit, heuristic, or high-cardinality	high-cardinality fixed effects are misspecified
5	Rare-category rows	drop pools with prevalence $< 5\%$	cannot be assigned to a reference group
6	Imputation	group-aware median/mode	train/deploy consistency
7	Winsorization	clip to $[\mu \pm 4\sigma]$	input regularization for the summarizer
8	Near-constant removal	drop if $> 95\%$ identical	collinear with intercept
9	Correlation filter	drop one of each $\lvert r\rvert > 0.95$ pair	well-conditioned posterior geometry
10	Categorical lumping	pool rare levels as `other`	avoid wide sparse dummies
11	Numeric transforms	binary: pass; skewed counts: log1p; z-score	keep interpretation, remove structural skew
12	Categorical encoding	modal reference level	matches `contr.treatment`
13	Outcome standardization	continuous: z-score (Appendix F.6); else raw	common scale for loss and flow