An end-to-end pipeline for uncertainty-aware validation of generative AI

Giada Badaracco
ETH Zürich, MIT, IAIFI
gbadaracco@student.ethz.ch

Sean Benevedes
MIT, IAIFI
seanmb@mit.edu

Gaia Grosso
MIT, IAIFI
gaiag795@mit.edu

Christina Reissel
MIT, IAIFI
creissel@mit.edu

Thea Aarrestad
ETH Zürich
thea.aarrestad@cern.ch

Philip Harris
MIT, IAIFI
pcharris@mit.edu

Abstract

Density estimation with generative AI is a common task in the physical sciences, with applications ranging from particle physics to gravitational-wave parameter estimation. Many of the existing methods, however, do not provide a way to estimate epistemic uncertainties, which is essential for reliable hypothesis testing. We propose an end-to-end framework combining generative modeling with principled uncertainty quantification. A normalizing-flow ensemble is trained to synthesize events; ensemble-based epistemic uncertainties are computed and propagated into a learned likelihood–ratio goodness-of-fit test. This yields robust distributional estimates that allow to synthesize significantly more events than those in the original training dataset and enable uncertainty-aware scientific discovery.

1 Introduction

Reliable density estimation is challenging in data-scarce regions of high-dimensional distributions. This regimes arise, for example, in searches for new physics in Large Hadron Collider (LHC) physics data: the signal often lies in regions of the phase space that can be computationally expensive to populate with background simulations, making downstream statistical inference hard. Generative models can interpolate complex distributions and oversample low-statistics regions. Normalizing flows (NFs) [1] are natural candidates because they represent flexible, high-dimensional densities with tractable likelihoods. However, their epistemic reliability in data-scarce regimes is uncertain [2]: when trained on limited statistics, NFs struggle to reproduce distribution tails, an issue for reliable hypothesis testing.

We introduce a pipeline that integrates generative modeling, epistemic uncertainty quantification, and statistical testing. We illustrate the pipeline with a controlled toy study where the true underlying distribution is known. Our key novelty is to propagate epistemic uncertainty from an ensemble density estimator [3] into a ML-based likelihood–ratio test statistic [4, 5], yielding calibrated p-values for uncertainty-aware decisions. While density estimation [6, 7] and ML-based likelihood-ratio tests (LRTs) have been extensively applied separately in fundamental physics problems, this joint treatment closes the loop from generative AI modeling to a robust statistical model for decision-making.

Machine Learning and the Physical Sciences Workshop, NeurIPS 2025.

2 End-to-end Method

Step 1: Distributional modeling with normalizing flows Let $p(x)$ denote the target density on $x \in \mathbb{R}^d$. We train an ensemble of M flows $\{f_{ij}\}$ by combining bootstrap resampling of the training data (index j) with independent random initializations (index i), so M counts all (i, j) pairs. This captures both data and optimization variability, increasing functional diversity and reducing correlations across members. Following Benevedes and Thaler [3], the frequentist uncertainty construction assumes that the true density is well approximated by the linear span of the ensemble.

Step 2: Frequentist uncertainty with $w_i f_i$ ensembles We form a weighted mixture $\hat{f}(x) = \sum_{i,j} \hat{w}_{ij}\, f_{ij}(x)$ with weights estimated by (penalized) maximum likelihood:

$$\mathcal{L}(w) = -\frac{1}{N}\sum_{n=1}^{N} \log \hat{f}(x_n) \;+\; \lambda\left(\sum_{i,j} w_{ij} - 1\right), \qquad \hat{w} = \arg\min_{w} \mathcal{L}(w). \tag{1}$$

λ acts as a Lagrange multiplier enforcing $\sum_{i,j} w_{ij} = 1$, so the ensemble is a proper probability density, with weights tuned to best match the true distribution. From analytic calculations, we find the optimal $\lambda = 1$. Treating $\hat{w}$ as an M–estimator [8], we obtain the covariance via the sandwich estimator:

$$\mathrm{Cov}^{kl}(\hat{w}) = V^{km}\, U_{mn}\, V^{nl}, \qquad V_{km} = \left.\frac{\partial^2 \mathcal{L}}{\partial w_k \partial w_m}\right|_{\hat{w}}, \;\; U_{mn} = \mathrm{Cov}\left[\frac{\partial \mathcal{L}}{\partial w_m}, \frac{\partial \mathcal{L}}{\partial w_n}\right]\Bigg|_{\hat{w}}. \tag{2}$$

Via the predictive variance

$$\sigma^2_{\hat{f}}(x) \;=\; \hat{f}_k(x)\, \mathrm{Cov}_{kl}(\hat{w})\, \hat{f}_l(x). \tag{3}$$

we can propagate the uncertainty of the weights into pointwise variance of the ensemble density.

Step 3: Coverage test A coverage test checks whether the true value of an observable lies within the ensemble's predicted error band at the chosen confidence level. Let $\hat{\mathcal{O}}$ be the estimator of the observable $\mathcal{O}$ obtained using the mixture model $\hat{f}$, and be $\mathcal{O}^*$ the observable's true value. Coverage at confidence level α is declared when

$$|\hat{\mathcal{O}} - \mathcal{O}^\star| \;<\; z_\alpha\, \sigma_{\hat{\mathcal{O}}}, \tag{4}$$

where $\sigma_{\hat{\mathcal{O}}}$ is propagated from $\mathrm{Cov}(\hat{w})$ and z_α is a standard-normal quantile.

Because $\mathcal{O}^\star$ is rarely known for real data, we use coverage as an optional diagnostic, not a required step. Checking coverage for a specific observable is only possible when the true value is known, and its assessments are limited to the specific features of the model that are relevant to estimate the observable. We perform coverage checks of the distribution's first moment as a coarse sanity check of uncertainty propagation. In applications without a known truth, Step 4 serves as the primary evaluation.

Step 4: Goodness of fit via learned likelihood–ratio tests A more exhaustive way to assess the accuracy and precision of the generative model is to perform a goodness-of-fit (GoF) test based on a likelihood ratio.

Accordingly, both hypotheses are parameterized densities that are re-fit on the test sample $\mathcal{D}$ and compared via a profile likelihood ratio [5, 9–11].

For this application, the null hypothesis H_ϕ is the composite hypothesis given by $f(x, \phi)$ and endowed with epistemic uncertainties, where ϕ collects the trainable parameters re-fit on $\mathcal{D}$. In our experiments, $\phi \equiv w$ are the ensemble weights, constrained by $\mathcal{N}(\hat{w}, \mathrm{Cov}(\hat{w}))$. The alternative $H_{\phi,a}$ augments the null with a Gaussian-kernel correction $g(x, a)$, so the density becomes $f(x, \phi) + g(x, a)$, with coefficients $a = (a_1, \ldots, a_K)$. Similar to [5], the likelihood–ratio test is built according to the standard LHC prescription for hypothesis testing in presence of nuisance parameters [11]

$$t(\mathcal{D}) \;=\; 2\log\left[\frac{\max\limits_{\phi,a} \mathcal{L}(\mathcal{D} \mid H_{\phi,a})}{\max\limits_{\phi} \mathcal{L}(\mathcal{D} \mid H_\phi)}\right]. \tag{5}$$

Here each maximization denotes a profiling step on the test data $\mathcal{D}$: we optimize two parameterized neural density models—the denominator re-fits ϕ (no kernel) and the numerator re-fits ϕ and a (with kernel)—with ϕ constrained by a Gaussian term centered at $\hat{w}$ with covariance $\mathrm{Cov}(\hat{w})$. This profiling is required for composite hypotheses so that the LRT compares the best-fitting members of each class on $\mathcal{D}$.

In practice we minimize the penalized negative log–likelihood, which is equivalent to maximizing the likelihood. The penalty includes (i) a Gaussian constraint on ϕ centered at $\hat{w}$ with covariance $\mathrm{Cov}(\hat{w})$ (importing Step 2 information) to encode the systematic uncertainties and (ii) a normalization term as in Eq. 1.

The test statistic in Eq. (5) quantifies whether adding the kernel degrees of freedom significantly improves the fit to the test data $\mathcal{D}$. A significant outcome of the test indicates that the original ensemble, together with its uncertainty estimate, does not provide an adequate description of the underlying distribution. To avoid double-dipping, data used to train/tune the NFs and estimate the ensemble weights $\hat{w}$ (see Steps 1-2) are kept strictly separate from the data $\mathcal{D}$ used to fit H_ϕ and to evaluate $t(\mathcal{D})$.

For calibration, we adopt a plug-in (parametric bootstrap) procedure for the composite null. We fix the null to $H_{\hat{w}}$ (the ensemble density $f(x;\hat{w})$ from Step 2) and generate synthetic data[1] $\mathcal{D}^{(b)} \sim f(x;\hat{w})$ via hit-or-miss (accept–reject) Monte Carlo. For each $\mathcal{D}^{(b)}$, we re-profile the denominator over ϕ and the numerator over (ϕ, a), keeping the Gaussian constraint on ϕ centered at $\hat{w}$ with covariance $\mathrm{Cov}(\hat{w})$, to compute $t^{(b)}$. The empirical distribution of $\{t^{(b)}\}$ defines the null distribution, from which we obtain the p-value of the observed $t(\mathcal{D})$. If more than one test dataset is available, the pipeline can be run recursively to optimize $\hat{f}$ until the test is passed successfully.

3 Numerical experiment

We validate the pipeline on a controlled two-dimensional toy problem with known ground truth. The target distribution is a multivariate gaussian with diagonal covariance matrix

$$p(x) = \mathcal{N}(x \mid \mu, \Sigma), \qquad \mu = (-0.5,\, 0.6), \;\; \Sigma = \mathrm{diag}(0.25^2,\, 0.4^2). \tag{6}$$

We generate 10^5 events, using 8×10^4 for training and 2×10^4 for validation. NFs are implemented with `nflows` and use masked piecewise rational–quadratic autoregressive transformations with permutation layers; the prior is standard normal. Training minimizes negative log-likelihood with learning rate 5×10^{-6}, batch size 512, and early stopping after 10 epochs without validation improvement. An ensemble of $M = 60$ models is trained on bootstrap replicas with independent initializations. The NF hyperparameters are: 4 layers, 16 blocks, 128 hidden features, 15 bins. Ensemble weights $\hat{w}$ are obtained by minimizing Eq. (1) utilizing the Newton's method as implemented in `pytorch-minimize` with a maximum of 300 iterations, initialized with small random perturbations and up to 50 restarts. Predictive variances for downstream propagation are computed via Eq. (3), using $\mathrm{Cov}(\hat{w})$ from Eq. (2). As an illustrative coverage test, we take the first moment as the downstream observable, and evaluate coverage according to Eq. (4). For GoF, we evaluate the learned likelihood–ratio test statistic in Eq. (5). We profile the parameters (ϕ, a) on the dataset $\mathcal{D}$. We place a Gaussian prior on ϕ with mean $\hat{w}$ and covariance $\mathrm{Cov}(\hat{w})$ from the $w_i f_i$ fit, and constrain the ensemble weights to sum to one via a quadratic penalty. All data used for training and tuning are disjoint from the GoF test data. For $H_{\phi,a}$, we use a Gaussian–kernel expansion with $K = 100$ isotropic kernels ($\sigma = 0.05$); centers are drawn from the held-out split and coefficients are renormalized to sum to one. We optimize with Adam (learning rate 10^{-4}) with early stopping. We add an ℓ_2-penalty on the kernel coefficients a, $\lambda_{\text{net}}\|a\|_2^2$ with $\lambda_{\text{net}} = 10^{-6}$. At convergence this term contributes at most 0.1% of the data term and acts only as a weak stabilizer to discourage overfitting.

4 Results

Coverage is first assessed on the first-moment observable (Appendix A). The ensemble uncertainties propagated from Eq. (2) yield coverage fractions consistent with the nominal 68% level, even when oversampling by a factor of two. This demonstrates that the ensemble provides reliable confidence

[1] In HEP terminology, synthetic datasets are often referred to as "pseudo-experiments".

intervals and that uncertainties are correctly estimated even well beyond the statistical power of the training data.

Fig. 1 shows the marginal densities of the two features. Residual differences between the ensemble reference (REF) and the target distribution (DATA) are concentrated in the tails. The LRT reconstruction (LRT RECO) denotes the profiled LRT alternative obtained by fitting a kernel correction to the reference. In the ratio panels, the density ratio LRT RECO/REF follows the empirical DATA/REF points and remains within the propagated predictive 1σ and 2σ bands from Eq. (3). Therefore, the ensemble reference reproduces the target distribution within its calculated epistemic uncertainties. This indicates that the epistemic uncertainty is well calibrated: it covers distortions from generative modeling in low-statistics regions and improves robustness.

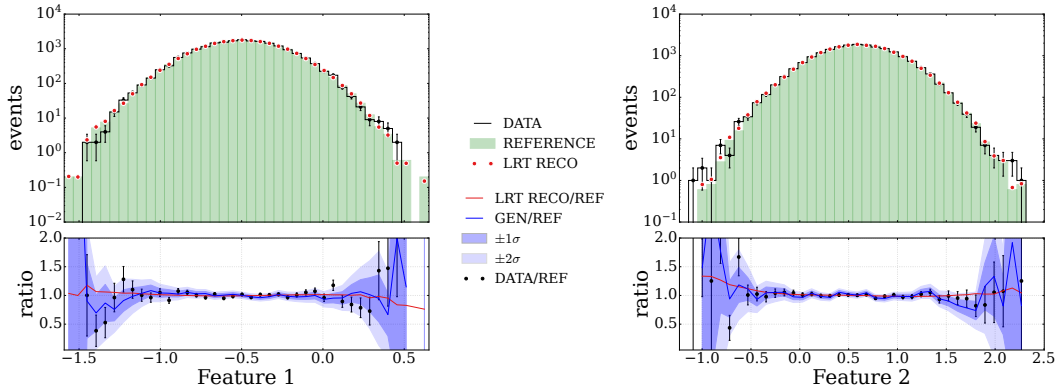

Figure 1: Marginal distributions of the two features. In the top panels the reference ensemble distribution (REF) is compared to a sample from the target distribution (DATA), and to the LRT reconstruction (LRT RECO). In the bottom panels, the ratios to the reference are shown with propagated $\pm1\sigma$ (blue) and $\pm2\sigma$ (light blue) predictive uncertainty bands from Eq. (3). Tail distortions remain within the predicted uncertainties.

The learned likelihood–ratio test of Eq. (5) provides a global, observable-independent validation. As shown in Fig. 2, including ensemble uncertainties in the GoF test reduces the discrepancy to less than 1σ ($Z = 0.92^{+0.20}_{-0.14}$). Even without uncertainty propagation, the ensemble already improves agreement between the generative model and the ground truth ($Z = 4.13^{+0.12}_{-0.13}$), as opposed to the use of a single NF ($Z = 20.29 \pm 0.10$). The ensembling approach therefore provides two key benefits: it improves model accuracy, and, once uncertainties are incorporated, it yields compatibility with the ground truth under GoF testing.

5 Conclusion and Future Work

We present an end-to-end framework for reliable distributional estimation in data-limited scenarios that propagates epistemic uncertainty from a normalizing flow density estimator ensemble into a learned likelihood–ratio goodness-of-fit statistic. The framework yields calibrated p-values and paves the way for uncertainty-aware validation enabling trustworthy use of generative AI in scientific discovery pipelines.

In a controlled toy study, the ensemble with $w_i f_i$ weighting not only produces well-calibrated intervals but also improves accuracy over single models. Propagating these uncertainties through the learned likelihood–ratio further shows statistical consistency between generated and reference samples within the expected bands. This particularly holds even in scenarios when the sample obtained from the generated distribution has significantly higher statistics than the training sample of the generative method (in our study, we are able to obtain at least twice as many samples as previously available). Our method paves the way to significantly increase statistics in low-populated phase spaces with statistical rigor, making it possible to use the obtained samples from generated distributions in various applications, e.g. anomaly detection in high-energy physics.

Although we explore NFs in our work, the framework is agnostic to the density estimator and could equally use matched flows, diffusion models, or kernel density estimators. A natural next step is to

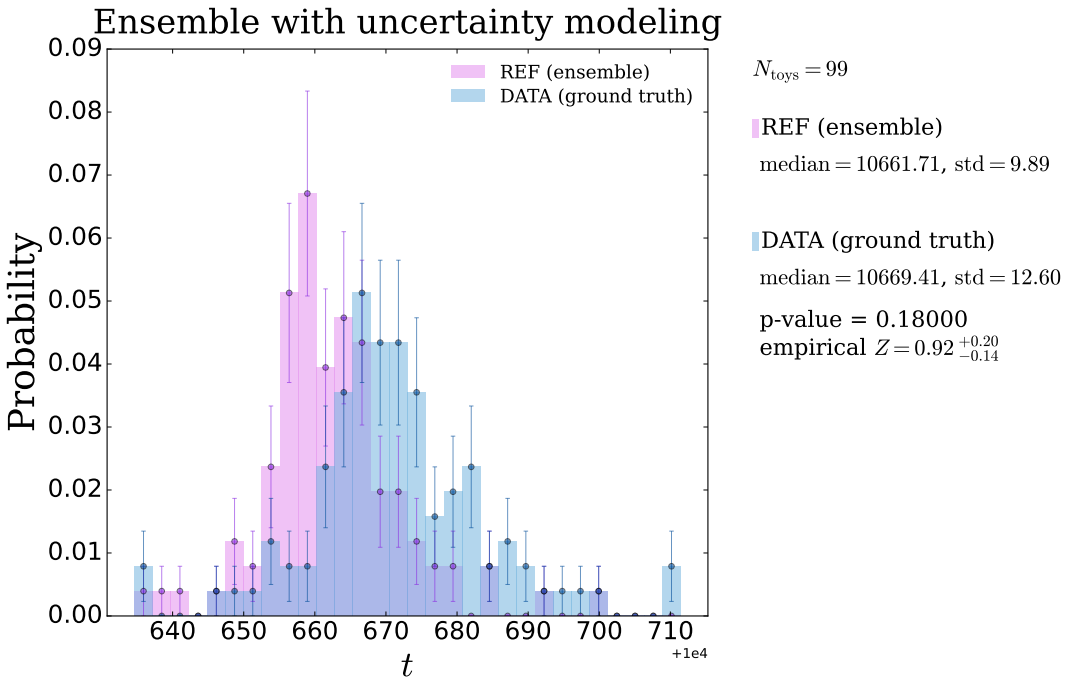

Figure 2: Learned likelihood–ratio test statistic. Null distribution of $t(\mathcal{D})$ (ensemble density) compared with the alternative with additional degrees of freedom. The ensemble is almost consistent with the ground truth ($Z = 0.92^{+0.20}_{-0.14}$).

adopt conditional generative models that take observed context as input. This would let us run the same profile-likelihood test within each context and then combine the evidence, improving power and enabling targeted, uncertainty-aware oversampling.

Further studies are underway to compare our approach to existing uncertainty estimation methods, such as Bayesian flows [12] and dropout [13] variants. While the methodology itself is independent of the exact use case and applications in various physical domains are possible, we plan to apply the method for background estimation, looking for anomalous signals in LHC physics data. Specifically, we plan to estimate the background distribution resulting from multiple jet production as predicted by the Standard Model of particle physics to facilitate new physics searches with jet final states.

References

[1] Ivan Kobyzev, Simon J. D. Prince, and Marcus A. Brubaker. Normalizing flows: An introduction and review of current methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11), 2020. doi: 10.1109/TPAMI.2020.2992934.

[2] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48), 2020. doi: 10.1073/pnas.1912789117.

[3] Sean Benevedes and Jesse Thaler. Frequentist uncertainties on neural density ratios with $w_i f_i$ ensembles. 2025. doi: 10.48550/arXiv.2506.00113.

[4] Johann Brehmer, Gilles Louppe, Juan Pavez, and Kyle Cranmer. Mining gold from implicit models to improve likelihood-free inference. *Proceedings of the National Academy of Sciences*, 117(10), 2020. doi: 10.1073/pnas.1915980117.

[5] Raffaele Tito d'Agnolo, Gaia Grosso, Maurizio Pierini, Andrea Wulzer, and Marco Zanetti. Learning new physics from an imperfect machine. *Eur. Phys. J. C*, 82(3):275, 2022. doi: 10.1140/epjc/s10052-022-10226-y.

[6] Anja Butter, Sascha Diefenbacher, Gregor Kasieczka, Benjamin Nachman, and Tilman Plehn. GANplifying event samples. *SciPost Phys.*, 10(6):139, 2021. doi: 10.21468/SciPostPhys.10.6.139.

[7] Riccardo Di Sipio, Michele Faucci Giannelli, Sana Ketabchi Haghighat, and Serena Palazzo. DijetGAN: A Generative-Adversarial Network Approach for the Simulation of QCD Dijet Events at the LHC. *JHEP*, 08:110, 2019. doi: 10.1007/JHEP08(2019)110.

[8] Peter J. Huber. Robust estimation of a location parameter. *Annals of Mathematical Statistics*, 35(1), 1964. doi: 10.1214/aoms/1177703732.

[9] Marco Letizia, Gianvito Losapio, Marco Rando, Gaia Grosso, Andrea Wulzer, Maurizio Pierini, Marco Zanetti, and Lorenzo Rosasco. Learning new physics efficiently with nonparametric methods. *Eur. Phys. J. C*, 82(879), 2022. doi: 10.1140/epjc/s10052-022-10830-y.

[10] Gaia Grosso, Marco Letizia, Maurizio Pierini, and Andrea Wulzer. Goodness of fit by neyman–pearson testing. *SciPost Phys.*, 16(123), 2024. doi: 10.21468/SciPostPhys.16.5.123.

[11] Glen Cowan, Kyle Cranmer, Eilam Gross, and Ofer Vitells. Asymptotic formulae for likelihood-based tests of new physics. *Eur. Phys. J. C*, 71:1554, 2011. doi: 10.1140/epjc/s10052-011-1554-0. [Erratum: Eur.Phys.J.C 73, 2501 (2013)].

[12] Marco Bellagente, Manuel Haußmann, Michel Luchmann, and Tilman Plehn. Understanding event-generation networks via uncertainties. *SciPost Physics*, 12(5):159, 2022. doi: 10.21468/SciPostPhys.12.5.159.

[13] Mehedi Hasan, Abbas Khosravi, Ibrahim Hossain, Ashikur Rahman, and Saeid Nahavandi. Controlled dropout for uncertainty estimation. *IEEE Transactions on Neural Networks and Learning Systems*, 32(7), 2021. doi: 10.1109/TNNLS.2020.3008945.

A Intermediate coverage test

Fig. 3 reports the empirical coverage for the first-moment observable (Sec. 3):

$$\hat{\mathcal{O}} = \int x\, \hat{f}(x)\, dx, \qquad \mathcal{O}^\star = \int x\, p(x)\, dx.$$

We evaluate coverage according to Eq. (4). Out of 284 synthetic datasets, Feature 1 achieved 61.3% coverage (target $O_1^\star = -0.50$) and Feature 2 achieved 68.7% (target $O_2^\star = 0.60$), both close to the nominal 68% level. The test used 100000 target events and 200000 generated samples, confirming that ensemble uncertainties remain trustworthy even under a factor-of-2 oversampling.

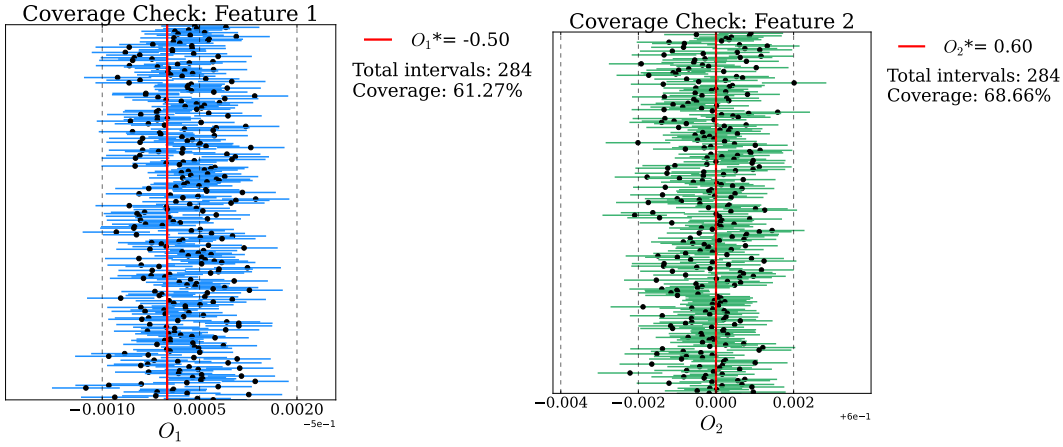

Figure 3: Coverage validation for first-moment observables. Left: Feature 1 ($O_1^\star = -0.50$), 61.3% coverage. Right: Feature 2 ($O_2^\star = 0.60$), 68.7% coverage.