End-to-End Optimization of Generative AI for Robust Background Estimation

Giada Badaracco [1][2], Sean Benevedes [2][3], Christina Reissel [2], Gaia Grosso [2][3], Thea Aarrestad [1], Philip Harris [2][3]

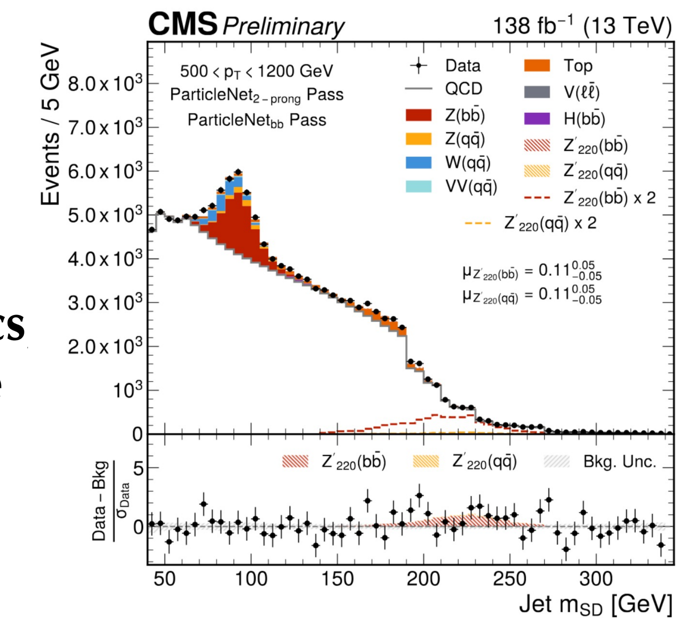
[1] ETH Zurich, [2] MIT , [3] NSF Institute for Artificial Intelligence and Fundamental Interaction (IAIFI)

MOTIVATION

Why do we need **better background models?**

LHC searches for BSM physics often target **rare-event, tails of distributions** where tight selection cuts **reduce background statistics** making **MC simulations too computational expensive for reliable background modeling, necessary for anomaly detection.**

→ **Generative data-driven models**



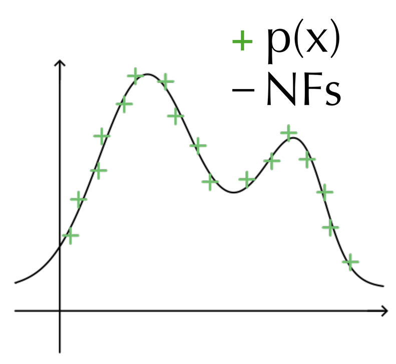
PROBLEM

To what extent can we trust **Generative AI?**

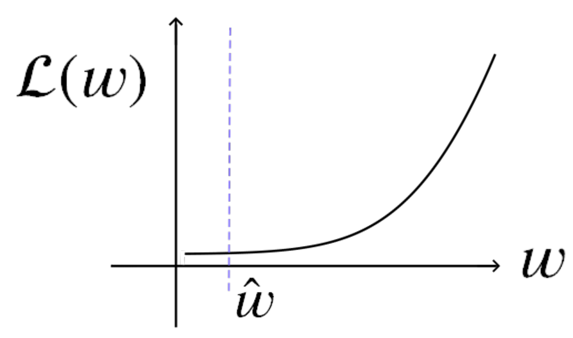
Generative models can interpolate complex distributions, **but in low-statistics regions** they may be **less precise**, so **estimating shape uncertainty** is **essential for robust use**.

→ **Model uncertainties** to ensure **robust anomaly detection in data-limited scenarios**

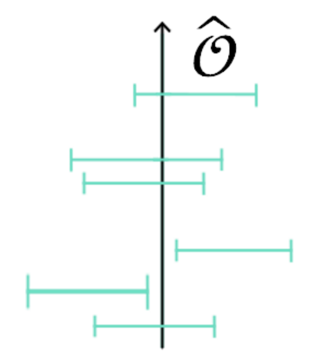
END-TO-END METHOD



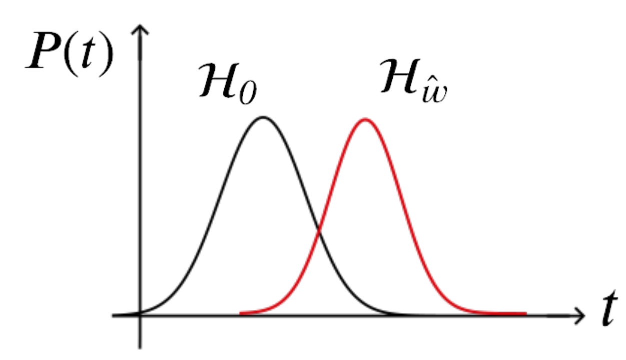
BACKGROUND MODELING WITH NORMALIZING FLOWS



FREQUENTIST UNCERTAINTY ESTIMATION WITH W_iF_i ENSEMBLES



COVERAGE CHECK



GOODNESS OF FIT WITH NPLM

target distribution $p(x)$

Train different f_{ij} varying **model initializations** i and **data bootstraps** j

Average over B bootstraps $f_i(x) = \frac{1}{B}\Sigma_j f_{ij}(x)$

Optimize weights

$\mathcal{L}(w) = NLL(w) + (\Sigma_i w_i - 1)$

$\hat{w}_i = argmin_{w_i}(\mathcal{L}(w))$

Estimate covariance

$Cov(w) = V^{-1}UV^{-1}$

$V_{ik} = \frac{d^2\mathcal{L}(w)}{d_{w_i}d_{w_k}}\Big|_{w=\hat{w}}$

$U_{kl} = Cov\left[\frac{d\mathcal{L}(w)}{dw_k}, \frac{d\mathcal{L}(w)}{dw_l}\right]\Big|_{w=\hat{w}}$

Predict $f(x) \pm \sigma(x)$

$\hat{f}(x) = \Sigma_i \hat{w}_i f_i(x)$

$\sigma^2_{\hat{f}}(x) = \hat{f}_i(x)\, Cov_{ij}(w)\, \hat{f}_j(x)$

Choose **observable** $\mathcal{O}$ whose **estimate** $\hat{\mathcal{O}}$ is compared to **expected value** $\mathcal{O}^*$

$\hat{\mathcal{O}} = \mathcal{O}(\hat{f}(x)) \quad \mathcal{O}^* = \mathcal{O}(p(x))$

Propagate error and check coverage

$|\hat{\mathcal{O}} - \mathcal{O}^*| < \sigma_{\hat{\mathcal{O}}}$

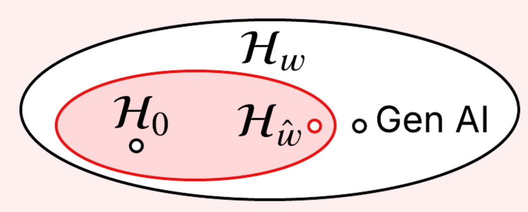
if covered

if not covered

New **P**hysics **L**earning **M**achine is a **semi-supervised anomaly detection** method that:

- learns **likelihood ratio** between generated and reference distributions
- models **alternative hypothesis** $\mathcal{H}_w$ using a kernel-based ML model
- computes a **test statistic** $t(\mathcal{D})$ to derive a p-value

$$t(\mathcal{D}) = 2\max_w \log \frac{\mathcal{L}(\mathcal{D}|\mathcal{H}_w)}{\mathcal{L}(\mathcal{D}|\mathcal{H}_0)}$$



NUMERICAL EXPERIMENT

Training:

- target distribution $p(x) = \mathcal{N}(\vec{\mu}, \vec{\sigma})$ $\begin{cases} \vec{\mu} = (-0.5, 0.6) \\ \vec{\sigma} = (0.25, 0.4) \end{cases}$
- 100k target events
- each NF uses masked autoregressive splines and permutations
- 30 models on different bootstrapped datasets

For checking the coverage:

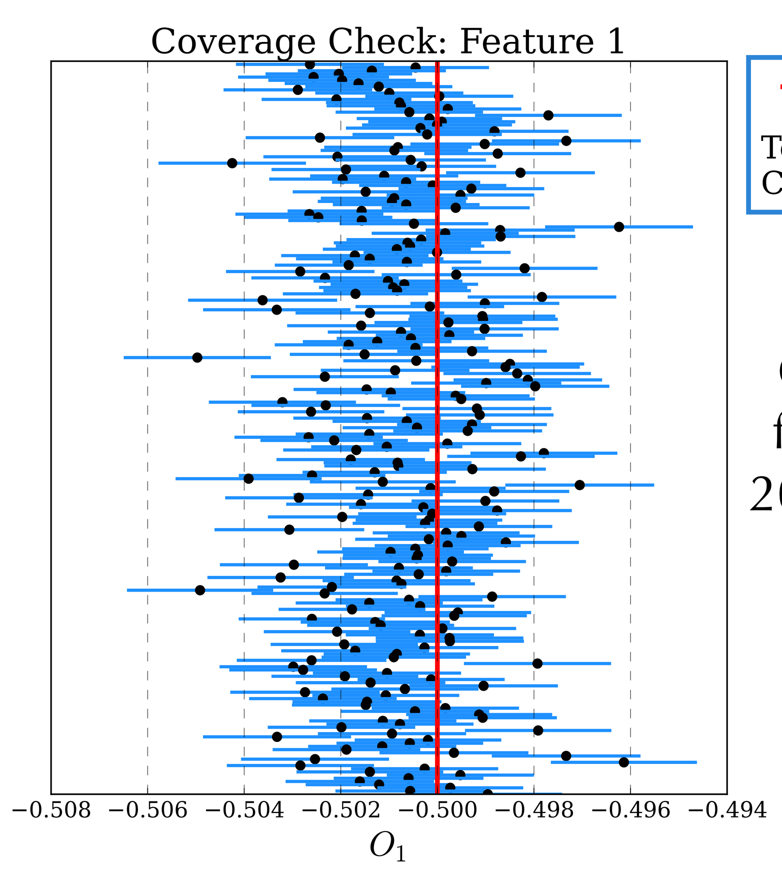
we choose observable $\hat{\mathcal{O}} = \int dx\, x\, \hat{f}(x)$

with expected value $\mathcal{O}^* = \int dx\, x\, p(x)$

best architecture found for coverage:
#bins = 15, #layers = 4, #blocks = 16, #features = 128

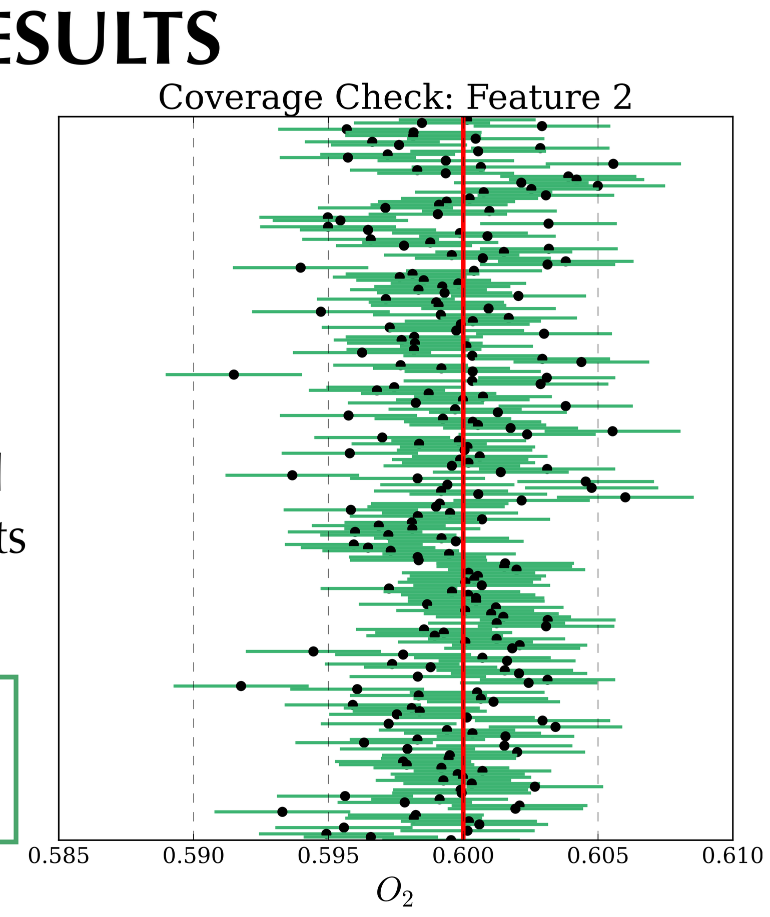
We use NPLM to test whether generated distribution $f(x)$ differs significantly from target distribution $p(x)$ corresponding to null hypothesis $\mathcal{H}_0$

RESULTS



O_1* = -0.50
Total intervals: 230
Coverage: 65.22%

Coverage at ~68% for 100k target and 20k generated events

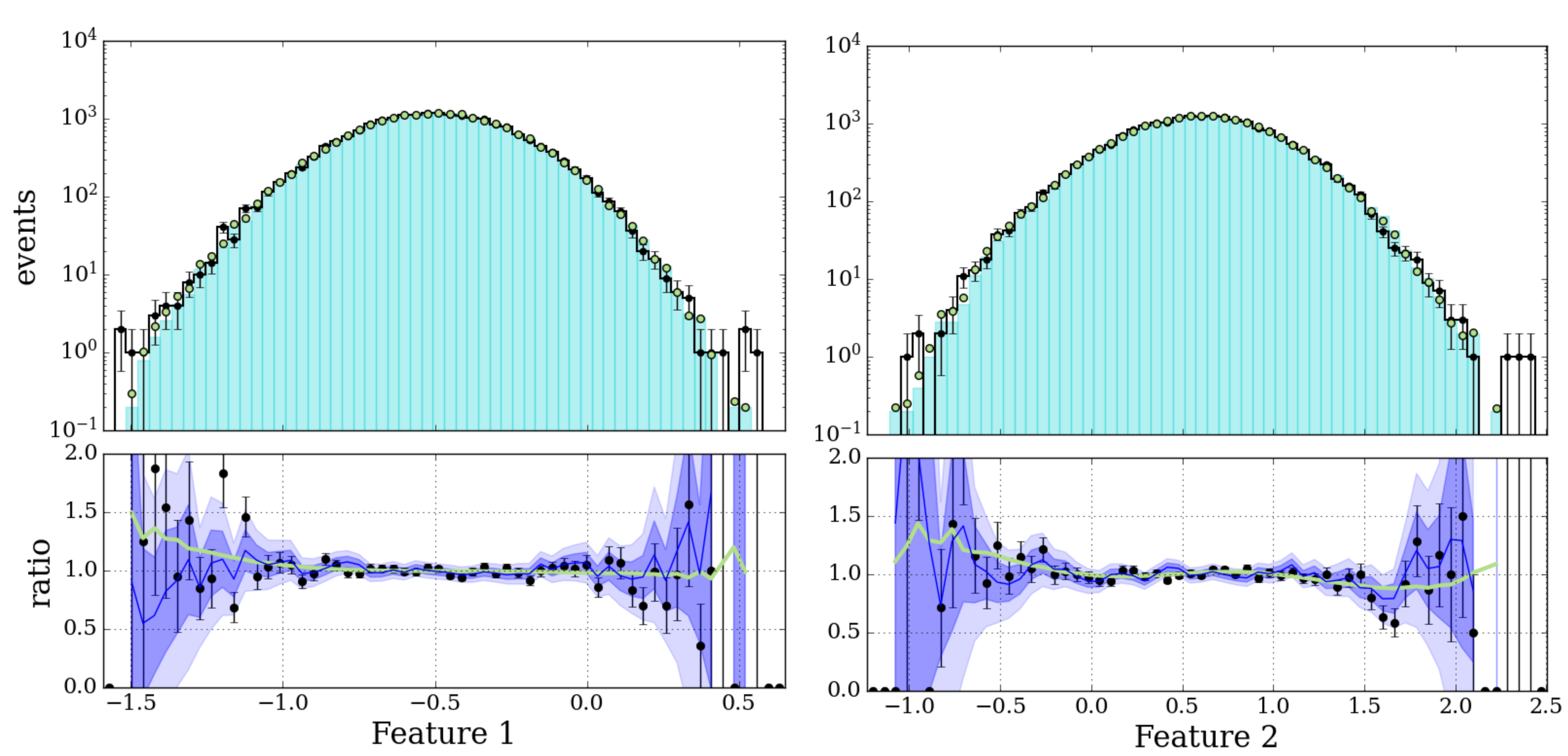


O_2* = 0.60
Total intervals: 230
Coverage: 68.70%

Asymptotic Z-scores:

- Ensemble: Z = 1.9 ± 0.2
- f_i : Z = 7.8 ± 0.1

→ ensemble improves accuracy



DATA
REFERENCE
NPLM RECO

NPLM RECO/REF
MODEL/REF
±1σ
±2σ
DATA/REF

NPLM fit over marginals → detected distortion within 1σ uncertainty bands

FUTURE WORK

- Smoothness and flexibility impact on ensemble behavior
- Compare with other density estimators, e.g. Mixture of Gaussians
- Compare with alternative uncertainty modeling, e.g. Dropout, Bayesian Flows
- Integrate uncertainty into the GoF test

1. "Learning multivariate new physics" Eur. Phys. J. C 81, 89 (2021)
2. "Learning new physics efficiently with nonparametric methods" Eur. Phys. J. C, 82(10)
3. "Goodness of fit by Neyman-Pearson testing" arXiv:2305.14137
4. "Frequentist Uncertainties on Neural Density Ratios with w_if_i Ensembles" arXiv:2506.00113