Neural Posterior Unfolding

Fernando Torales Acosta, Jay Chan, Krish Desai, Vinicius Mikuni,
Benjamin Nachman, Jingjing Pan (jingjing.pan@yale.edu)

Introduction

Differential cross section measurements are the currency of scientific exchange in particle and nuclear physics. The key challenge for these analyses is the correction for detector distortions known as deconvolution or *unfolding*. In this work, we show how normalizing flows and neural posterior estimation can be used for unfolding, which we call Neural Posterior Unfolding (NPU). This approach has many potential advantages, including implicit regularization from the neural networks and fast inference from amortized training. We demonstrate this approach using simple Gaussian examples as well as a simulated jet substructure measurement at the Large Hadron Collider.

Statistics of Unfolding

For binned unfolding, we approximate the detector response with the response matrix $R_{ij} = \Pr(m_i\,|\,t_j)$ where m_i indicates that the observable is measured in bin i at detector-level and t_j indicates that the observable is in bin j at particle-level. The response matrix for collider experiments can be computed extremely precisely using simulations. Iterative Bayesian Unfolding (IBU) [1] will serve as our baseline. One way of describing it is through the following iterative protocol:

$$t_j^{(n)} = \sum_i \Pr_{n-1}(t_j\,|\,m_i)\Pr(m_i) = \sum_i \frac{R_{ij}t_j^{(n-1)}}{\sum_k R_{ik}t_k^{(n-1)}} \times m_i$$

where n is the iteration and stopping $n < \infty$ is a form of regularization.
IBU only provides a point estimate.

This already highlights potential advantages of NPU: the statistical uncertainty is part of the result and the regularization is implicit in the training and automated through model selection (via the validation loss). Another difference between NPU and IBU is their behavior in unconstrained regions of phase space. Cross sections in the particle-level phase space that are un- or poorly constrained by detector-level measurements should be highly uncertain. However, IBU could only return a point estimate with an uncertainty of zero, as demonstrated in Fig.1, whereas NPU can circumvent this issue by returning a wide posterior.

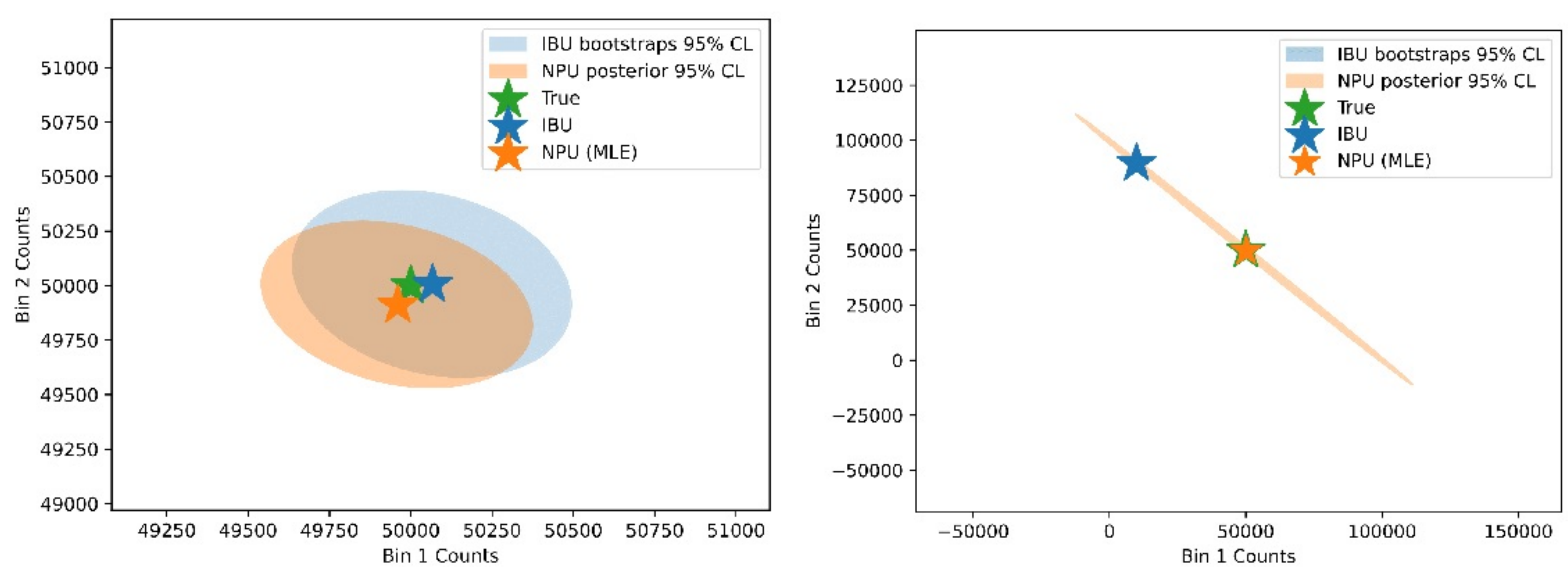
Numerical Results

2-bin Degenerate Response Example

We first start with a simple two-bin example. In this case, $t, m \in \mathbb{R}^2$ and R is a 2×2 matrix. Let the response matrix $R = [\sigma\ \rho,\ \rho\ \sigma]$, where in this example, $\sigma = \sigma_1 = \sigma_2 = 0.8$ and $\rho = \mathrm{Cov}_{12}$. We fix $t_0 = t_1 = 5 \times 10^4$. First, Fig.1(a) shows the case where $\rho = 1$. With small migrations, all methods perform about the same, and the true answer is well-contained within the confidence regions of both IBU (determined via boostrapping) and NPU. However, once we set $\rho \approx 0$, Fig.1(b) shows the challenges with IBU highlighted in the previous section. In particular, IBU returns a single value while NPU returns a broad uncertainty region (all values consistent with the total counts).

Machine Learning Implementation

We implement neural posterior unfolding using Tensorflow [2] and Tensorflow Probability packages [3]. The normalizing flow implementation is based on the MADE [4,5] implementation consisting of an invertible transformation using two fully connected layers with 16 nodes and SWISH [6] activation functions. The conditional inputs, based on pre-detector observables, are included with an additional fully connected layer. The model is then trained for between 1000 and 1500 epochs. We used the ADAM [7] optimizer with a learning rate of 0.001. After training, we determine the unfolded response by performing the MLE. This is carried out by minimizing the negative log-likelihood of the data based on the conditional inputs, also with ADAM.



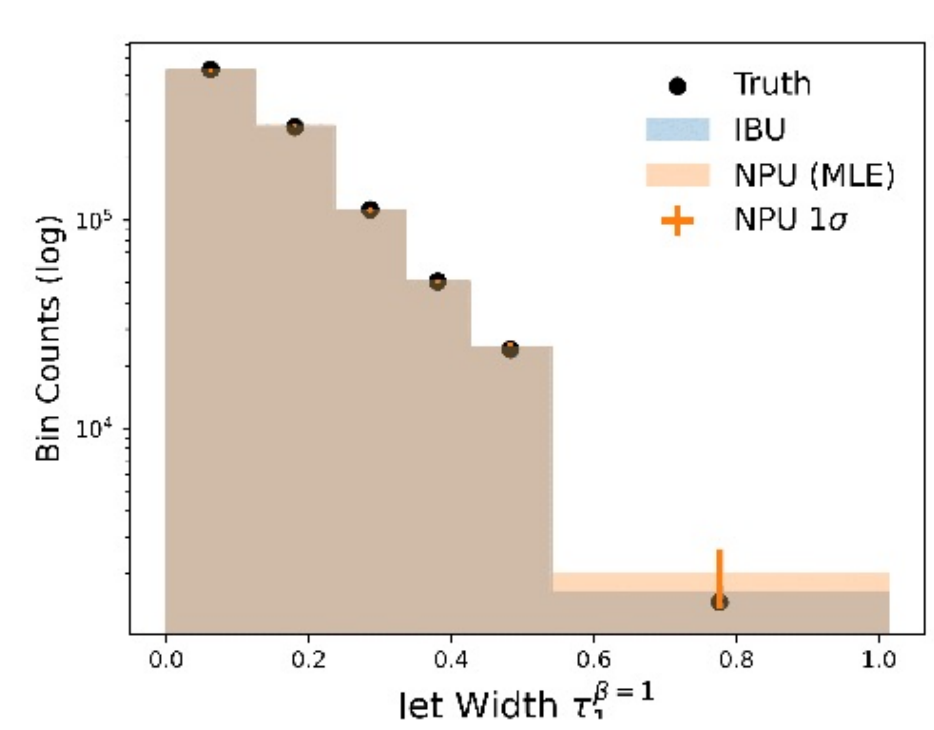
(a) 2-bin example

(b) 2-bin example with degenerate response

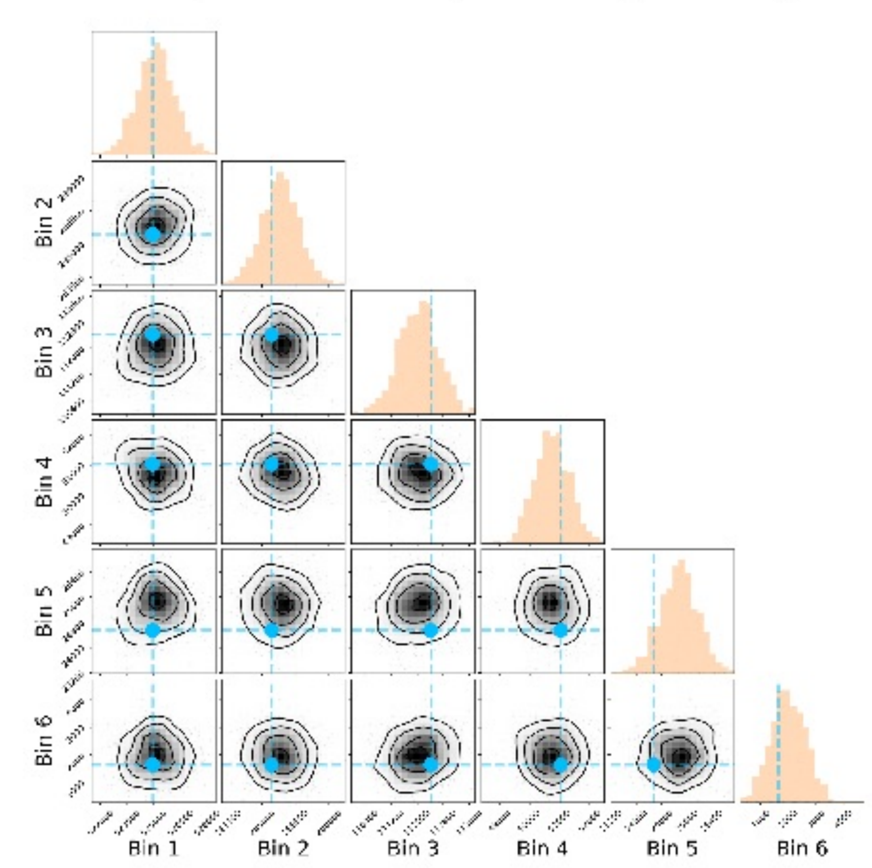
Figure 1: Demonstration of NPU using a simple two-bin example with (a) a non-degenerate (*i.e.* nearly diagonal) response matrix *v.s.* (b) a response with degeneracy (as in the example from the text). The maximum likelihood estimation (MLE) of NPU shows good agreement with the truth values. The posterior from NPU is also compatible with the confidence region evaluated by IBU using bootstrap datasets.

Particle Physics Example

Our study is based on proton-proton collisions generated at $\sqrt{s} = 14$ TeV; the dataset is the same as used in [8]. To investigate the unfolding performance, we investigated a number of jet substructure observables. For brevity, we show the representative results from the jet width ($\tau_1^{\beta=1}$). The jet width is the transverse-momentum-weighted first radial moment of the radiation within a jet. Gluon-initiated jets tend to be wider than quark jets. The unfolding performance of NPU is shown in Fig.2(a) and compared to IBU with 10 iterations. The corresponding corner plot is in Fig.2(b). NPU continues to succeed in recovering the truth distribution inside the high-energy jets produced in the complicated real-world LHC environment, despite the challenging long tail of this widely used observable.



(a) Gaussian result



(b) Gaussian result in corner plot

References

[1] G. D'Agostini. A Multidimensional unfolding method based on Bayes' theorem. Nucl. Instrum. Meth., A362:487–498, 1995. doi: 10.1016/0168-9002(95)00274-X.

[2] Martín Abadi et al. TensorFlow: Large- scale machine learning on heterogeneous systems, 2015. URL https:// www.tensorflow. org/. Software available from tensorflow.org.

[3] Joshua V.Dillon et al. Tensorflow distributions. CoRR, abs/ 1711.10604, 2017. URL http://arxiv.org/abs/1711.10604.

[4] Mathieu Germain et al. MADE: Masked Autoen- coder for Distribution Estimation. 2015.

[5] George Papamakarios et al. Masked Autoregressive Flow for Density Estimation. 2018.

[6] P. Ramachandran et al. Searching for activation functions. arXiv preprint arXiv:1710.05941, 2017.

[7] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. 2015.

[8] Anders Andreassen et al. OmniFold: A Method to Simultaneously Unfold All Observables. Phys. Rev. Lett., 124 (18):182001, 2020. doi: 10.1103/PhysRevLett.124.182001.