Scaffolding Simulations with Deep Learning for High-Dimensional Deconvolution

Anders Andreassen (Google), Patrick T. Komiske (MIT, IAIFI),
Eric M. Metodiev (MIT, IAIFI), Benjamin Nachman (Berkeley Lab, BIDS),
Adi Suresh (UC Berkeley), Jesse Thaler (MIT, IAIFI)

ICLR 2021 Workshop
Deep Learning for Simulation (simDL)

We introduce and extend the **OmniFold** method: an EM-style, likelihood-free approach to deconvolution that is unbinned and can process variable- and high-dimensional data.

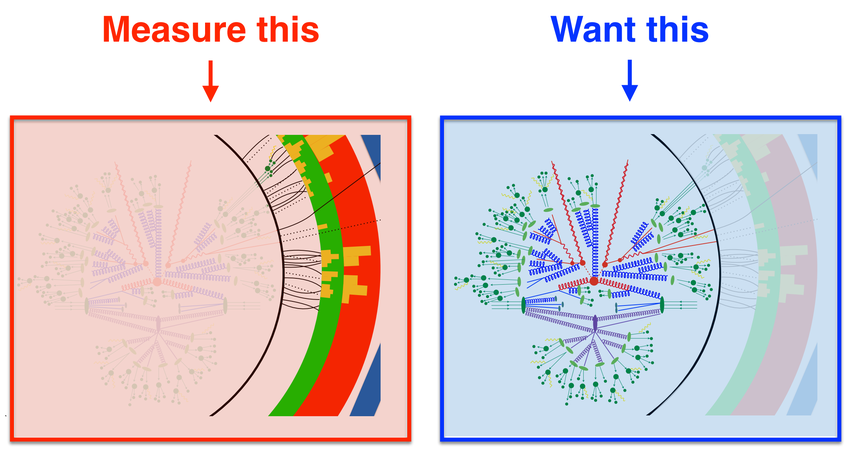
A complete deconvolution algorithm must account for four effects:

- **Noise processes.** *In many cases, the data is a coherent superposition of a signal process and a background process.*
- **Detector acceptance.** *The detector elements may not capture all signal process examples due to finite thresholds and other acceptance effects.*
- **Detector distortions.** *This is the classical convolution of the data with a noise function that must be statistically removed.*
- **Detector efficiency.** *The definition of a "signal event" may include a restricted phase space.*

The Unfolding Challenge

Deconvolution (also known as Unfolding) is particularly challenging when the data have a complex structure.

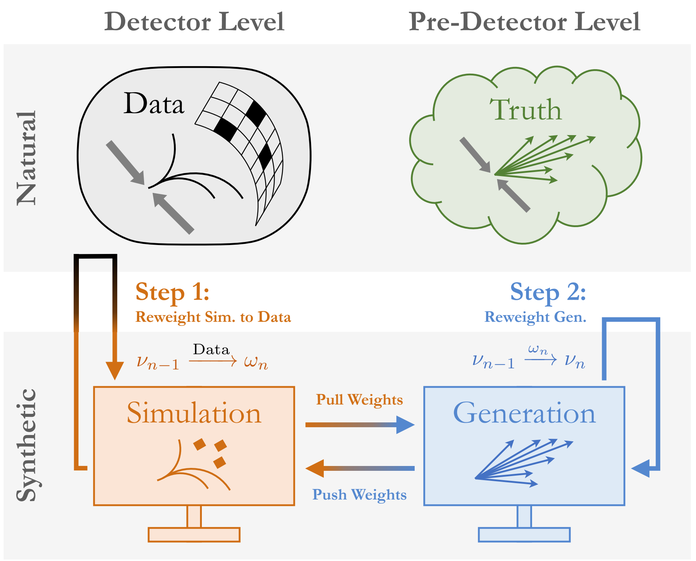
For example, in collider physics, the data are naturally represented as an unordered and variable-length set of particles. Each particle has a momentum and possibly other attributes (e.g. electric charge).



A schematic diagram of a proton-proton collision at the Large Hadron Collider. The left part of each diagram represents the sub-nuclear physics of particle production and decay that we want to infer from the detector measurements represented in the right part of each diagram.

Correcting Detector Distortions

We use a synthetic dataset that has pre- and post-detector examples which are matched to each other. A series of weights are constructed using neural networks as likelihood-ratio approximators.



A classifier trained to distinguish two datasets with e.g. cross entropy will asymptotically learn the likelihood ratio between the underlying generative models. This ratio can reweight one dataset to statistically match another. **Step 1** reweights the synthetic detector-level (Simulation) to match Data. The weights from Simulation are then assigned to the synthetic pre-detector examples (Generation). **Step 2** builds a proper function of the Generation phase space. This process is then repeated.

Extending OmniFold

Phys. Rev. Lett. 124 (2020) 182001 showed how OmniFold can correct for detector distortions. In this work, we show that it can also be used to statistically subtract noise and detector acceptance and efficiency effects.

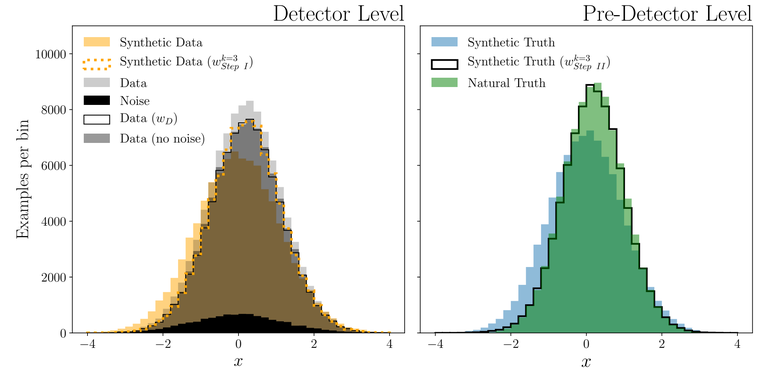
Noise processes are statistically subtracted using Neural Positive Reweighting (Nachman and Thaler, Phys. Rev. D 102 (2020) 076004): a classifier is trained to distinguish {data} from {data, noise} where the noise in the latter set are given a weight of -1.

Acceptance and **Efficiency effects** are corrected for by including a special symbol for examples that lack a pre- or post-detector component. These are then propagated through the entire analysis.

Numerical Results

OmniFold has been demonstrated on variable- and high-dimensional data, but for illustrating the extended version, here are Gaussian examples.

First, a one-dimensional Gaussian example:



The fact that the black outline and green filled histogram agree in the right plot shows that the method works. Note that the data are binned for illustration, but the actual result is **unbinned**.

Next is a multidimensional example. The target is a 1-dimensional Gaussian to which 4 additional Gaussians have been added. The unfolding is done using either the full sum *(N=1)*, the sum plus the first Gaussian noise dimension *(N=2)*, etc.

	$\bar{x}$ mean ($\times 10^2$)				$\bar{x}$ standard deviation ($\times 10^3$)			
iterations $\rightarrow$	1	2	4	8	1	2	4	8
N								
1	21.62(8)	25.13(8)	28.12(8)	29.67(8)	8.4(5)	8.3(6)	8.0(7)	7.9(7)
2	28.54(5)	29.24(6)	29.88(6)	**30.06(5)**	5.3(4)	5.6(4)	5.2(4)	4.8(4)
3	29.54(4)	**29.91(4)**	**30.02(4)**	**30.00(4)**	3.6(3)	4.4(4)	3.6(3)	4.0(3)
4	29.89(3)	**30.01(3)**	**30.01(3)**	**30.01(3)**	**3.2(2)**	**2.8(2)**	**3.1(2)**	**3.1(2)**
5	**30.04(3)**	**30.00(3)**	**29.99(4)**	**30.06(3)**	3.5(3)	**3.1(2)**	3.8(3)	**3.1(2)**

The fact that the mean (left) and standard error (right) are smaller when more dimensions are included in the deconvolution shows the power of adding more information. In contrast to other algorithms, adding additional dimensions in OmniFold is as easy as changing an *n* to an *n+1* in one line of python!

Acknowledgements

BN and AS are supported by the U.S. Department of Energy (DOE), Office of Science under contract DE-AC02-05CH11231. JT and PK are supported by the National Science Foundation under Cooperative Agreement PHY-2019786 (The NSF AI Institute for Artificial Intelligence and Fundamental Interactions, http://iaifi.org/), and by the U.S. DOE Office of High Energy Physics under grant number DE-SC0012567.

https://github.com/hep-lbdl/OmniFold