PyUnfold: A Python package for iterative unfolding

James Bourbeau[1] and Zigfried Hampel-Arias[1, 2]

1 University of Wisconsin - Madison, Wisconsin, USA **2** IIHE, Université Libre de Bruxelles, Bruxelles, Belgium

Summary

In an ideal world, experimentalists would have access to the perfect detector: an apparatus that makes no error in measuring a desired quantity. However, in real life scenarios, detectors have:

- Finite resolutions
- Characteristic biases that cannot be eliminated
- Less than 100% detection efficiencies
- Statistical uncertainties

By building a matrix that encodes a detector's smearing of the desired true quantity into the measured observable(s), a deconvolution can be performed that provides an estimate of the true variable. This deconvolution process is known as unfolding.

PyUnfold is an extensible framework for the unfolding of discrete probability distributions via the iterative unfolding method described in (D'Agostini 1995). This method falls into the general class of inverse problems, and is especially powerful when no analytic form of the inverse function is explicitly available, instead requiring an estimate (e.g. via a finite amount of simulation) of the response function. Given that measured data comprise a finite sample, PyUnfold also implements the uncertainty contributions stated in (Adye 2011).

The unfolding method itself is data-agnostic, referring to the measurement process as the smearing of a set of true causes into a set of detectable effects. For example one could define as causes the true energy of a particle and the effects the measured energy of that particle in a detector. Another example might be a set of diseases (causes) and possible clinical symptoms (effects). So long as it is possible to encode estimable resolutions and biases connecting causes to effects in a binned response matrix, one can perform a deconvolution with PyUnfold.

The primary purpose of PyUnfold is to provide an unfolding toolkit for members of all scientific disciplines in an easy-to-use package. For example, unfolding methods are commonly used in the analysis pipeline of the high-energy physics (HEP) community. However, the deconvolution packages used in HEP maintain a strong dependence on the ROOT data analysis framework (ROOT 1997), which is almost exclusively used in HEP. Instead, PyUnfold is built on top of the Python scientific computing stack (i.e. NumPy, SciPy, and pandas), thus broadening its scope to a general scientific audience.

PyUnfold has been designed to be both easy to use for first-time users as well as flexible enough for fine-tuning an analysis and seamlessly testing the robustness of results. It provides support for the following:

- Custom, user defined initial prior probability distributions, the default being the uniform prior. The non-informative Jeffreys' prior (Jeffreys 1946) is accessible as a utility function.

- Unfolding stopping criteria based on test statistic calculations comparing unfolded distributions from one iteration to the next. These include Kolmogorov-Smirnov (Kolmogorov 1933)(Smirnov 1948), χ^2, relative difference, and Bayes factor (BenZvi et al. 2011) tests.
- Tunable spline regularization as a means of ensuring that unfolded distributions do not suffer from growing fluctuations potentially arising from the finite binning of the response matrix.
- Option to choose between Poisson or multinomial forms of the covariance matrices for both the data and response contributions to the uncertainty calculation.
- Multivariate unfolding via definitions of subsets of causes, which are regularized in their respective blocks or groups.

Further mathematical details regarding the iterative unfolding procedure, including complete derivations of the statistical and systematic uncertainty propagation can be found in the online documentation.

PyUnfold has been applied successfully for the measurement of the cosmic-ray energy spectrum by the HAWC Observatory (Alfaro 2017) and is currently being used in an analysis by members of the IceCube Neutrino Observatory.

Acknowledgements

References

Adye, T. 2011. "Corrected Error Calculation for Iterative Bayesian Unfolding." http://hepunx.rl.ac.uk/~adye/software/unfold/bayes_errors.pdf.

Alfaro, R. et al. 2017. "All-Particle Cosmic Ray Energy Spectrum Measured by the HAWC Experiment from 10 to 500 Tev." *Phys. Rev. D* 96 (12):122001. https://doi.org/10.1103/PhysRevD.96.122001.

BenZvi, S. Y., B. M. Connolly, C. G. Pfendner, and S. Westerhoff. 2011. "A Bayesian Approach to Comparing Cosmic Ray Energy Spectra." *The Astrophysical Journal* 738 (1):82. https://doi.org/10.1088/0004-637X/738/1/82.

D'Agostini, G. 1995. "A multidimensional unfolding method based on Bayes' theorem." *Nucl. Instrum. Meth. A* 362:487–98. https://doi.org/10.1016/0168-9002(95)00274-X.

Jeffreys, H. 1946. "An Invariant Form for the Prior Probability in Estimation Problems." *Proc. Of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* 186 (1007). London, England:453–61. https://doi.org/10.1098/rspa.1946.0056.

Kolmogorov, A. N. 1933. "Sulla Determinazione Empirica Di Una Legge Di Distribuzione." *Giornale Dell'Istituto Italiano Degli Attuari* 4:83–91.

ROOT. 1997. "An Object Oriented Data Analysis Framework." *Nucl. Instr. & Meth. In Phys. Rev. A* 389:81–86. https://doi.org/10.1016/S0168-9002(97)00048-X.

Smirnov, N. 1948. "Table for Estimating the Goodness of Fit of Empirical Distributions." *Ann. Math. Statist.* 19 (2). The Institute of Mathematical Statistics:279–81. https://doi.org/10.1214/aoms/1177730256.