A Statistician's View on Current Unfolding Software

Mikael Kuusela

SAMSI and UNC Chapel Hill

DIANA / HEP meeting

November 13, 2017

Current unfolding methodology

- Let $\boldsymbol{y}$ denote the smeared histogram
- We have $\boldsymbol{y} \sim \mathrm{Poisson}(\boldsymbol{K\lambda})$, where $\boldsymbol{K}$ is the response matrix and $\boldsymbol{\lambda}$ the true histogram
- Goal: Make inferences about $\boldsymbol{\lambda}$
- Two main approaches (more details available, e.g., in ‣ Unfolding lecture):
  1. Tikhonov regularization (Höcker and Kartvelishvili, 1996; Schmitt, 2012):
  $$\min_{\boldsymbol{\lambda} \in \mathbb{R}^p} (\boldsymbol{y} - \boldsymbol{K\lambda})^{\mathrm{T}} \hat{\boldsymbol{C}}^{-1} (\boldsymbol{y} - \boldsymbol{K\lambda}) + \delta P(\boldsymbol{\lambda})$$
  with
  $$P(\boldsymbol{\lambda}) = \left\| \boldsymbol{L} \begin{bmatrix} \lambda_1/\lambda_1^{\mathrm{MC}} \\ \lambda_2/\lambda_2^{\mathrm{MC}} \\ \vdots \\ \lambda_p/\lambda_p^{\mathrm{MC}} \end{bmatrix} \right\|^2 \quad \text{or} \quad P(\boldsymbol{\lambda}) = \|\boldsymbol{L}(\boldsymbol{\lambda} - \boldsymbol{\lambda}^{\mathrm{MC}})\|^2,$$
  where $\boldsymbol{L}$ is usually the discretized second derivative (also other choices possible)
  2. Expectation-maximization iteration with early stopping (D'Agostini, 1995):
  $$\lambda_j^{(t+1)} = \frac{\lambda_j^{(t)}}{\sum_{i=1}^n K_{i,j}} \sum_{i=1}^n \frac{K_{i,j} y_i}{\sum_{k=1}^p K_{i,k} \lambda_k^{(t)}}, \quad \text{with } \boldsymbol{\lambda}^{(0)} = \boldsymbol{\lambda}^{\mathrm{MC}}$$
- Regularization strength controlled by the choice of δ in Tikhonov or by the number of iterations in D'Agostini
- Uncertainty quantification by error propagation

Wish list for future unfolding software

1. We should not call D'Agostini iteration "Bayesian unfolding"
   - This method has nothing to do with Bayesian inference (in fact, it converges to the frequentist MLE)
   - Calling it Bayesian gives a misleading picture of what the method does
   - It is fine to call it the D'Agostini iteration, but even better would be to call it the expectation-maximization (EM) iteration
2. I do not think we should provide software defaults for the regularization strength
   - There are countless of LHC papers that use 4 iterations of D'Agostini simply because it is the default in `RooUnfold`
   - The optimal regularization strength is highly problem-dependent
   - There is no way to give a single universal value that always works well
   - Instead, the choice should always be based on an objective data or Monte Carlo based criterion
3. The software should provide built-in implementations of standard techniques for data-driven choice of the regularization strength
   - More on this in a couple of slides

Wish list for future unfolding software

4. The software should provide the user the full freedom to choose the histogram $\boldsymbol{\lambda}^{\mathrm{MC}}$ defining the regularization
   - `RooUnfold` currently defines $\boldsymbol{\lambda}^{\mathrm{MC}}$ using the same MC that was used to construct $\boldsymbol{K}$
   - It is important to be able to define $\boldsymbol{\lambda}^{\mathrm{MC}}$ irrespective of how $\boldsymbol{K}$ is chosen (enables studying separately the effect of the MC on $\boldsymbol{K}$ and on the regularization; also MC independent regularization by setting $\boldsymbol{\lambda}^{\mathrm{MC}} = \text{const} \cdot \mathbf{1}$)
5. The software should provide automated tools for checking the coverage of the unfolded uncertainties against an alternative MC model
   - It is much too easy to accidentally perform circular coverage checks using the current software
   - Let $\boldsymbol{\lambda}^{\mathrm{MC1}}$ and $\boldsymbol{\lambda}^{\mathrm{MC2}}$ be MC truths from two different generators and let $\boldsymbol{\lambda}^{\mathrm{MC1}}$ define the regularization
   - If you now check the coverage against $\boldsymbol{\lambda}^{\mathrm{MC1}}$, you will always find perfect coverage, irrespective of the regularization strength
   - But that is correct only if $\boldsymbol{\lambda}^{\mathrm{MC1}}$ is indeed the physical truth
   - A more realistic check verifies the coverage against $\boldsymbol{\lambda}^{\mathrm{MC2}}$, simulating what the coverage would be when the regularization is defined using a MC model that is a slight deviation of the physical truth

Wish list for future unfolding software

6. Replace the Gaussian least-squares approximation in Tikhonov with the actual Poisson likelihood, i.e., $\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda} \in \mathbb{R}^p} F(\boldsymbol{\lambda})$, where

$$F(\boldsymbol{\lambda}) = \underbrace{\sum_{i=1}^{n} \left[y_i \log \left(\sum_{j=1}^{p} K_{i,j} \lambda_j \right) - \sum_{j=1}^{p} K_{i,j} \lambda_j \right]}_{= \log p(\mathbf{y}|\boldsymbol{\lambda}) + \mathrm{const}} - \delta P(\boldsymbol{\lambda})$$

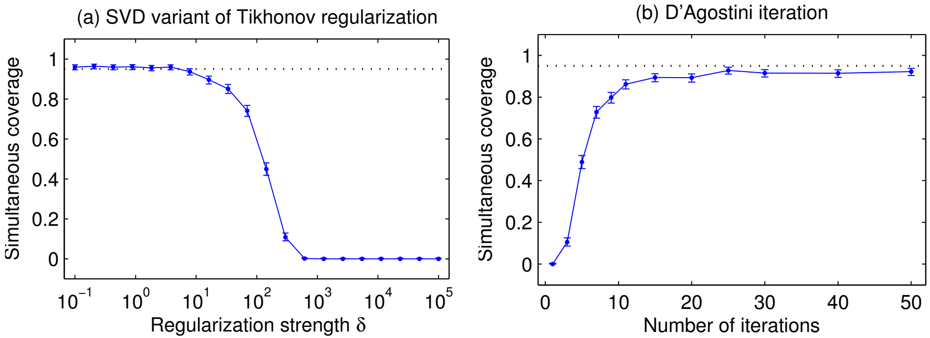
   - Should be helpful for bins where there is little data
   - Any modern numerical optimization package should be able to maximize $F(\boldsymbol{\lambda})$ without any trouble
7. Enforce the positivity constraint by optimizing over $\mathbb{R}^p_+$ instead of $\mathbb{R}^p$
   - Can optimize in log-space or use constrained numerical optimization
   - Gives a big physics-driven boost for regularization

[Note: D'Agostini by construction is based on the full Poisson likelihood and enforces the positivity constraint. But I find the regularization in Tikhonov to be much more transparent.]

Choice of the regularization strength

- A key issue in unfolding is the choice of the regularization strength (δ in Tikhonov, # of iterations in D'Agostini)
- To avoid MC dependence, this choice should ideally be done using a data-driven technique instead of MC studies
- Many data-driven methods have been proposed:
  1. (Weighted/generalized) cross-validation (e.g., Green and Silverman, 1994)
  2. L-curve (Hansen, 1992)
  3. Empirical Bayes estimation (Kuusela and Panaretos, 2015)
  4. Goodness-of-fit test in the smeared space (Veklerov and Llacer, 1987)
  5. Akaike information criterion (Volobouev, 2015)
  6. Minimization of a global correlation coefficient (Schmitt, 2012)
  7. ...
- `RooUnfold` does not implement any of these; `TUnfold` implements (2) and (6); I think all deserve to be implemented (would facilitate comparing the methods in real-life unfolding situations)
- Important note: All of these are designed for point estimation
  - Not necessarily optimal for uncertainty quantification!

Coverage as a function of regularization strength (Kuusela, 2016)

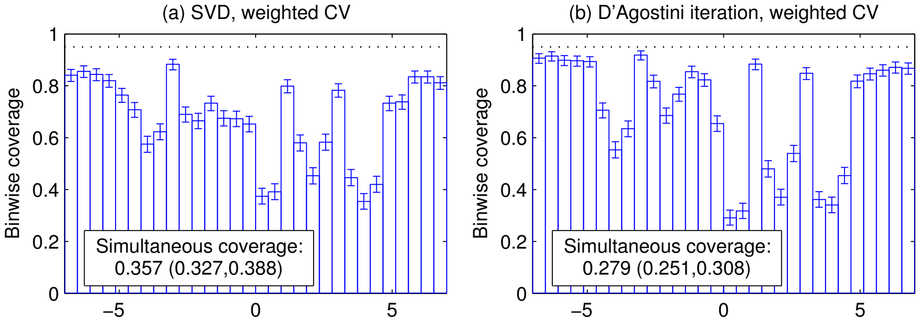


$$f(t) = \lambda_{\mathrm{tot}} \left\{ \pi_1 \mathcal{N}(t|-2,1) + \pi_2 \mathcal{N}(t|2,1) + \pi_3 \frac{1}{|E|} \right\}$$

$$g(s) = \int_T \mathcal{N}(s-t|0,1) f(t)\,\mathrm{d}t$$

$$f^{\mathrm{MC}}(t) = \lambda_{\mathrm{tot}} \left\{ \pi_1 \mathcal{N}(t|-2,1.1^2) + \pi_2 \mathcal{N}(t|2,0.9^2) + \pi_3 \frac{1}{|E|} \right\}$$

Undercoverage of existing methods (Kuusela, 2016)



[The uncertainties tend to be especially badly estimated when $\boldsymbol{\lambda}^{MC}$ is close to the physical truth; some preliminary studies indicate that $\boldsymbol{\lambda}^{MC} = \text{const} \cdot \mathbf{1}$ performs better, but it is definitely not perfect either.]

Undersmoothed unfolding

- A simple way to improve the coverage is to reduce the regularization strength δ from the value that is optimal for point estimation
- This idea of undersmoothed confidence intervals goes back to at least Hall (1992), but previous work has mostly focused on theory and not on how to do this in practice
- Kuusela (2016) introduced a technique for choosing *how much* one should undersmooth

Outline of undersmoothed UQ for unfolding

1. Choose a pilot estimate of δ using one of the standard data-driven methods (CV, empirical Bayes, L-curve,...)
2. Reduce δ until intervals in all bins have estimated coverage greater than $1 - \alpha - \varepsilon$, for some small tolerance ε

TUnfold with undersmoothing

- I have been working with Lyle Kim (a statistics student at UChicago) to implement undersmoothing as an extension of `TUnfold` V17.6
- The code is available at

  `https://github.com/lylejkim/UndersmoothedUnfolding`
- It includes two new functions:
  1. `TVectorD ComputeCoverage(TMatrixD *lambda, Double_t tau)`:

     Computes binwise coverages given an estimate of the true histogram $\boldsymbol{\lambda}$ and the regularization strength $\tau = \sqrt{\delta}$
  2. `Double_t UndersmoothTau(Double_t tau, Double_t epsilon, Int_t max_iter)`:

     Reduces a pilot estimate of τ until the estimated coverage in all bins is greater than $0.68 - \epsilon$; the output is the undersmoothed value of τ
- If you are working on an analysis that uses `TUnfold`, then trying this approach requires only one extra call to `UndersmoothTau`
  - See `UndersmoothDemo.C` in the repository for example usage

Unfolded histograms, $\boldsymbol{\lambda}^{\mathrm{MC}} = 0$

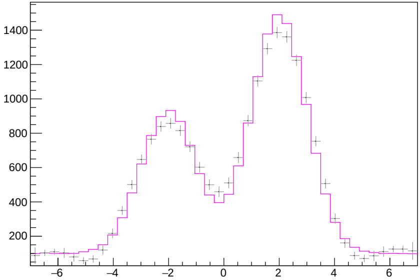


Figure: L-curve, $\tau = \sqrt{\delta}$=0.01186

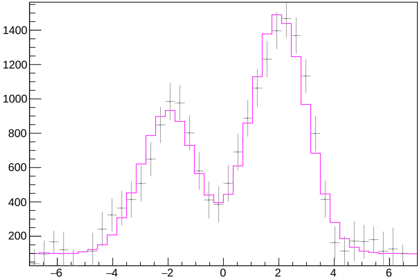


Figure: Undersmoothing, $\tau = \sqrt{\delta}$=0.00177

Binwise coverage, $\lambda^{\mathrm{MC}} = 0$

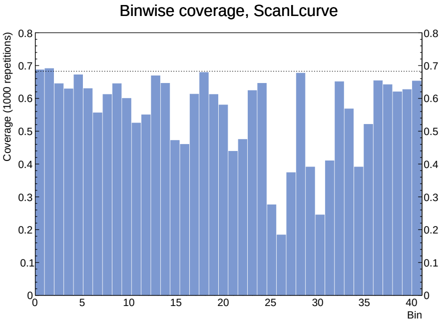


Figure: L-curve

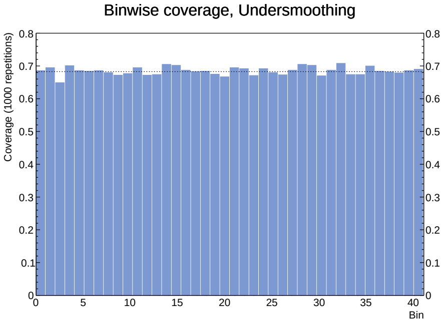


Figure: Undersmoothing

Conclusions and outlook

- Existing unfolding software should be improved especially in terms of enabling user-defined regularization and data-driven choice of the regularization strength
  - Some of the items from my wish list are already implemented in `TUnfold`
- Obtaining realistic uncertainties requires using less regularization than what is optimal for point estimation
  - Can be done in a fully data-driven manner using techniques introduced in Kuusela (2016)
  - `ROOT` implementation now available at:

    `https://github.com/lylejkim/UndersmoothedUnfolding`
- What is the best way forward?
  - Extension of existing software? If so, then `RooUnfold` or `TUnfold`?
  - A new project from scratch? In `ROOT`? In Python? In some other environment?

References I

G. D'Agostini, A multidimensional unfolding method based on Bayes' theorem, *Nuclear Instruments and Methods A*, 362:487–498, 1995.

P. J. Green and B. W. Silverman. *Nonparametric Regression and Generalized Linear Models: A Roughness Penalty Approach*. Chapman & Hall, 1994.

P. Hall, Effect of bias estimation on coverage accuracy of bootstrap confidence intervals for a probability density, *The Annals of Statistics*, 20(2):675–694, 1992.

P. C. Hansen, Analysis of discrete ill-posed problems by means of the L-curve, *SIAM Review*, 34(4):561–580, 1992.

A. Höcker and V. Kartvelishvili, SVD approach to data unfolding, *Nuclear Instruments and Methods in Physics Research A*, 372:469–481, 1996.

M. Kuusela. *Uncertainty quantification in unfolding elementary particle spectra at the Large Hadron Collider*. PhD thesis, EPFL, 2016. Available online at: `https://infoscience.epfl.ch/record/220015`.

M. Kuusela and V. M. Panaretos, Statistical unfolding of elementary particle spectra: Empirical Bayes estimation and bias-corrected uncertainty quantification, *The Annals of Applied Statistics*, 9(3):1671–1705, 2015.

References II

S. Schmitt, TUnfold, an algorithm for correcting migration effects in high energy physics, *Journal of Instrumentation*, 7:T10003, 2012.

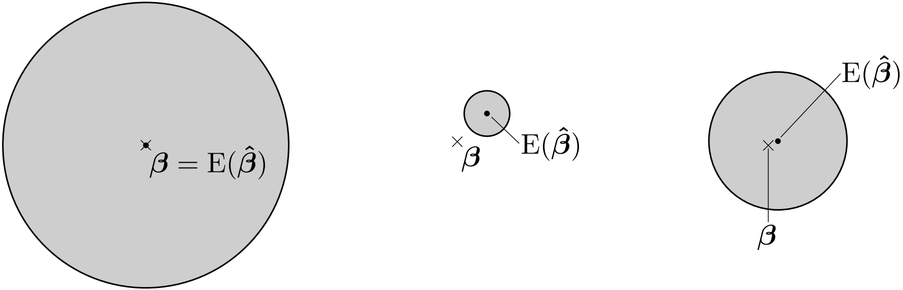
M. Stone, Cross-validatory choice and assessment of statistical predictions, *Journal of the Royal Statistical Society. Series B (Methodological)*, 36:111–147, 1974.

E. Veklerov and J. Llacer, Stopping rule for the MLE algorithm based on statistical hypothesis testing, *IEEE Transactions on Medical Imaging*, 6(4):313–319, 1987.

I. Volobouev. On the expectation-maximization unfolding with smoothing. arXiv:1408.6500v2 [physics.data-an], 2015.

Backup

Bias-variance trade-off and uncertainty quantification



Unbiased,
coverage $= 1 - \alpha$

Optimal point estimation,
coverage $\ll 1 - \alpha$

Optimal UQ?,
coverage $= 1 - \alpha - \varepsilon$

Obtaining good coverage performance requires adjusting the bias-variance trade-off to the direction of less bias and more variance!

Coverage as a function of $\tau = \sqrt{\delta}$

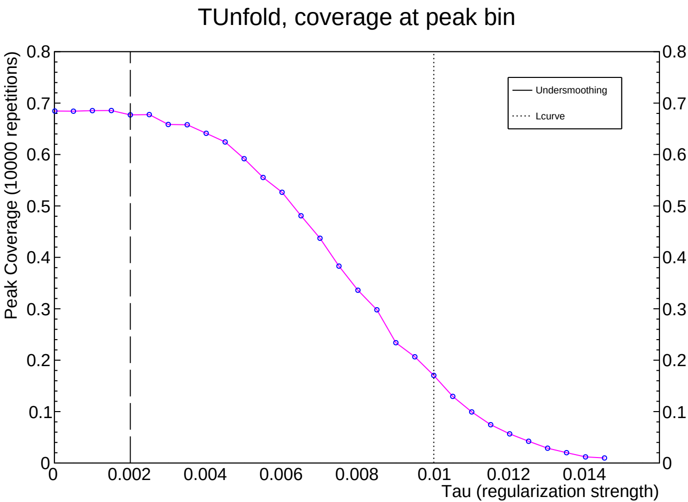
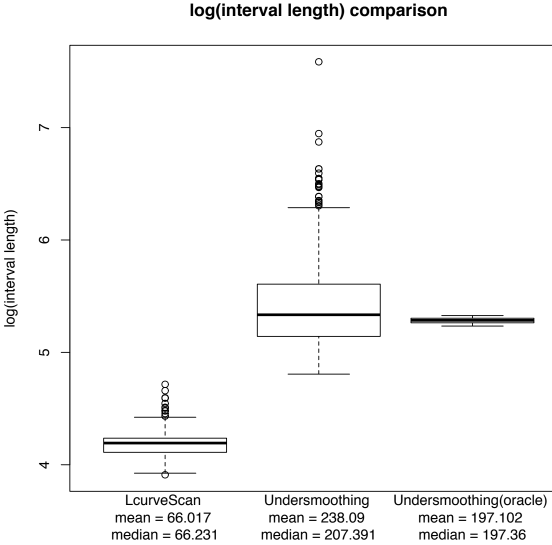
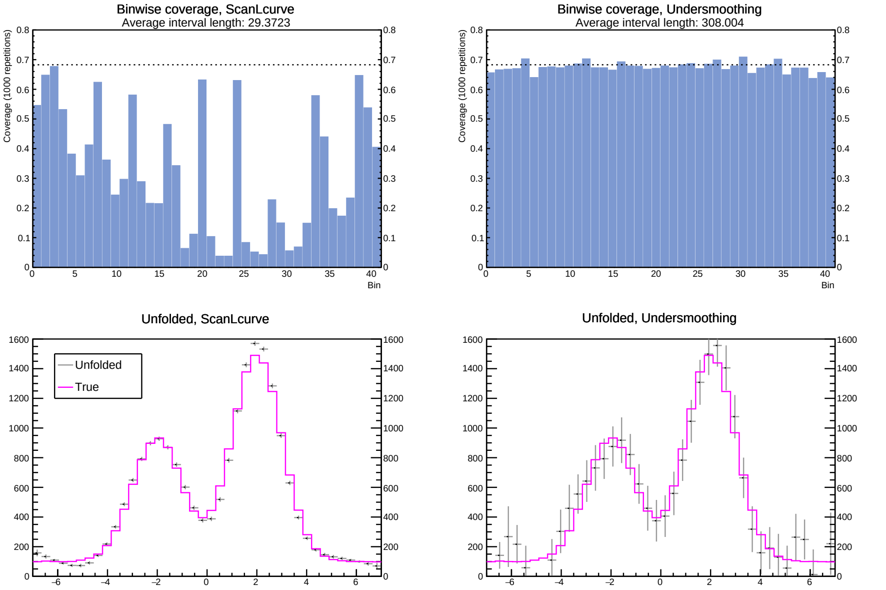


Figure: Coverage at the right peak of a bimodal density

Interval lengths, $\boldsymbol{\lambda}^{\mathrm{MC}} = 0$



Histograms, coverage and interval lengths when $\boldsymbol{\lambda}^{\mathrm{MC}} \neq 0$



Coverage study from Kuusela (2016)

Method	Coverage at $t = 0$	Mean length
BC (data)	0.932 *(0.915, 0.947)*	0.079 *(0.077, 0.081)*
BC (oracle)	0.937 *(0.920, 0.951)*	0.064 *(0.064, 0.064)*
US (data)	0.933 *(0.916, 0.948)*	0.091 *(0.087, 0.095)*
US (oracle)	0.949 *(0.933, 0.962)*	0.070 *(0.070, 0.070)*
MMLE	0.478 *(0.447, 0.509)*	0.030 *(0.030, 0.030)*
MISE	0.359 *(0.329, 0.390)*	0.028
Unregularized	0.952 *(0.937, 0.964)*	40316

BC = iterative bias-correction

US = undersmoothing

MMLE = choose δ to maximize the marginal likelihood

MISE = choose δ to minimize the mean integrated squared error