Introduction to Unfolding in High Energy Physics

Mikael Kuusela

Institute of Mathematics,
EPFL

Advanced Scientific Computing Workshop,
ETH Zurich

July 15, 2014

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Outline

1. Introduction
2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding
3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. Conclusions

Outline

1. Introduction
2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding
3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. Conclusions

The unfolding problem

- Unfolding refers to the problem of estimating the particle-level distribution of some physical quantity of interest on the basis of observations smeared by an imperfect measurement device
- What would the distribution look like when measured with a device having a perfect experimental resolution?
    - Cf. deconvolution in optics, image reconstruction in medical imaging

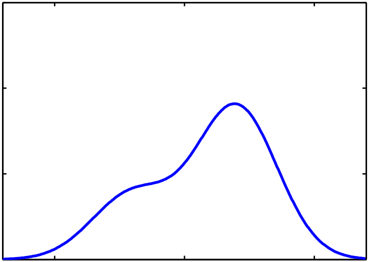
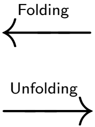


Figure : Smeared density



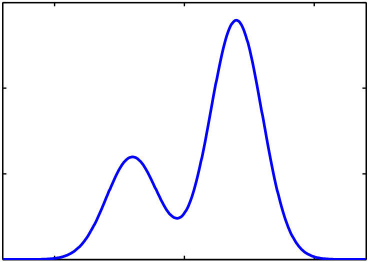


Figure : True density

Why unfold?

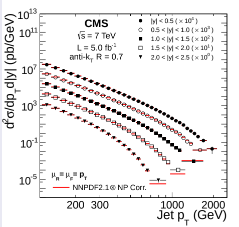
Unfolding is usually done to achieve one or more of the following goals:

1. **Comparison of the measurement with future theories**
2. **Comparison of experiments with different responses**
3. **Input to a subsequent analysis**
4. **Exploratory data analysis**

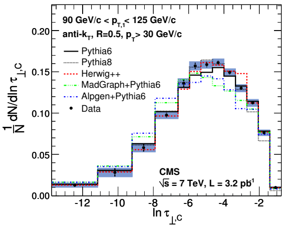
Unfolding is most often used in *measurement* analyses (as opposed to *discovery* analyses): QCD, electroweak, top, forward physics,...

Examples of unfolding in LHC data analysis

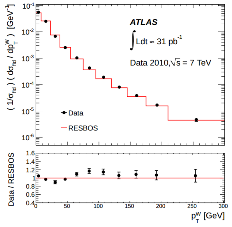
Inclusive jet cross section



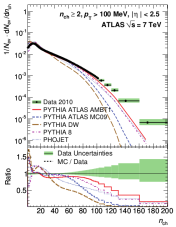
Hadronic event shape



W boson cross section



Charged particle multiplicity



Problem formulation

- Notation:
  - $\boldsymbol{\lambda} \in \mathbb{R}_+^p$ bin means of the true histogram
  - $\mathbf{x} \in \mathbb{N}_0^p$ bin counts of the true histogram
  - $\boldsymbol{\mu} \in \mathbb{R}_+^n$ bin means of the smeared histogram
  - $\mathbf{y} \in \mathbb{N}_0^n$ bin counts of the smeared histogram
- Assume that:
  1. The true counts are independent and Poisson distributed

     $$\mathbf{x}|\boldsymbol{\lambda} \sim \mathrm{Poisson}(\boldsymbol{\lambda}), \quad \perp\!\!\!\perp x_i|\boldsymbol{\lambda}$$

  2. The propagation of events to neighboring bins is multinomial conditional on x_i and independent for each true bin
- It follows that the smeared counts are also independent and Poisson distributed

$$\mathbf{y}|\boldsymbol{\lambda} \sim \mathrm{Poisson}(\mathbf{K}\boldsymbol{\lambda}), \quad \perp\!\!\!\perp y_i|\boldsymbol{\lambda}$$

Problem formulation

- Here the elements of the smearing matrix $\mathbf{K} \in \mathbb{R}^{n \times p}$ are given by

$$K_{ij} = P(\text{smeared event in bin } i \,|\, \text{true event in bin } j)$$

  and assumed to be known
- The unfolding problem:

Problem statement

Given the smeared observations $\mathbf{y}$ and the Poisson regression model

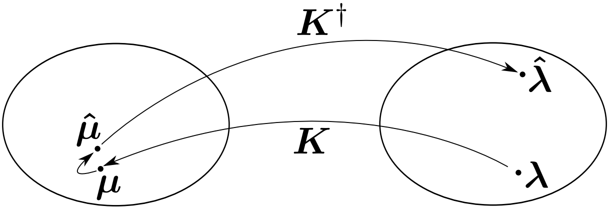
$$\mathbf{y}|\boldsymbol{\lambda} \sim \mathrm{Poisson}(\mathbf{K}\boldsymbol{\lambda}),$$

what can be said about the means $\boldsymbol{\lambda}$ of the true histogram?

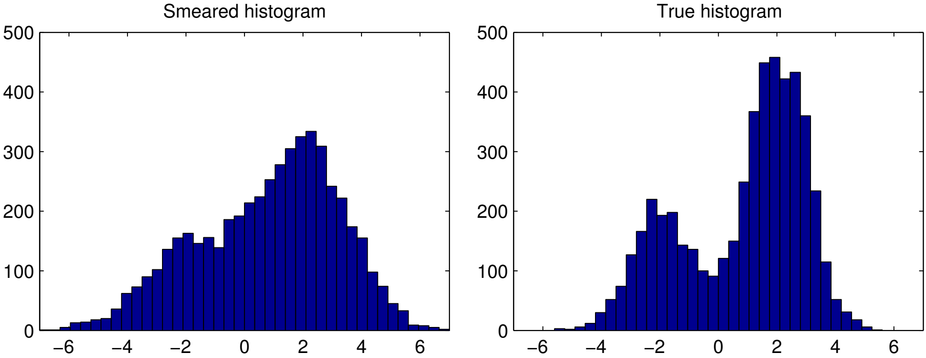
- The problem here is that typically $\mathbf{K}$ is an ill-conditioned matrix

Unfolding is an ill-posed inverse problem

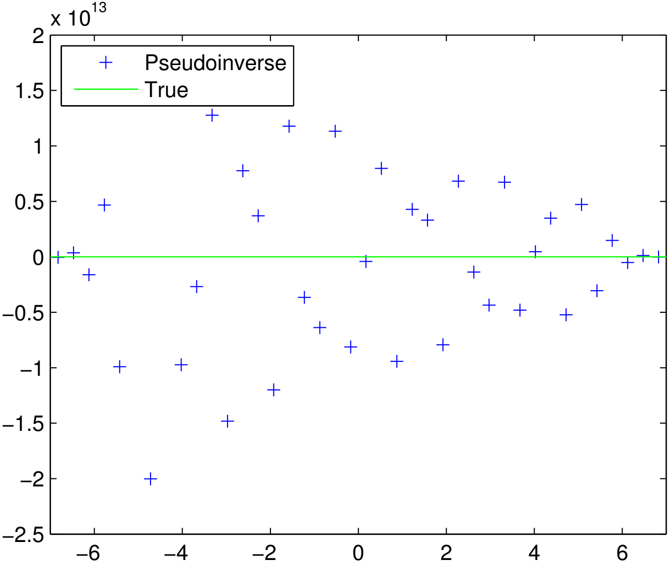
- The unfolding problem is typically ill-posed in the sense that the (pseudo)inverse of $\mathbf{K}$ is very sensitive to small perturbations in the data
- From $\mathbf{y}|\boldsymbol{\lambda} \sim \mathrm{Poisson}(\mathbf{K}\boldsymbol{\lambda})$ we have that $\boldsymbol{\mu} = \mathbf{K}\boldsymbol{\lambda}$
- We could naïvely estimate $\hat{\boldsymbol{\lambda}} = \mathbf{K}^\dagger \hat{\boldsymbol{\mu}} = \mathbf{K}^\dagger \mathbf{y}$
- But this can lead to catastrophic results!



Demonstration of the ill-posedness



Demonstration of the ill-posedness



Outline

1. Introduction
2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding
3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. Conclusions

Outline

1. Introduction
2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding
3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. Conclusions

The likelihood function

- The *likelihood function* in unfolding is:

$$L(\boldsymbol{\lambda}) = p(\mathbf{y}|\boldsymbol{\lambda}) = \prod_{i=1}^{n} p(y_i|\boldsymbol{\lambda}) = \prod_{i=1}^{n} \frac{\left(\sum_{j=1}^{p} K_{ij}\lambda_j\right)^{y_i}}{y_i!} e^{-\sum_{j=1}^{p} K_{ij}\lambda_j}, \quad \boldsymbol{\lambda} \in \mathbb{R}_+^p$$

- This function uses our Poisson regression model to link the observations $\mathbf{y}$ with the unknown $\boldsymbol{\lambda}$
  - The likelihood function plays a key role in all sensible unfolding methods
- In most statistical problems, the maximum of the likelihood, or equivalently the maximum of the log-likelihood, provides a good estimate of the unknown
  - In ill-posed problems, *this is usually not the case*, but the maximum likelihood solution still provides a good starting point

Maximum likelihood estimation

- Any histogram that maximizes the log-likelihood of the unfolding problem is called a *maximum likelihood estimator* $\hat{\boldsymbol{\lambda}}_{\mathrm{MLE}}$ of $\boldsymbol{\lambda}$
- Hence, we want to solve:

$$\max_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} \log p(\mathbf{y}|\boldsymbol{\lambda}) = \sum_{i=1}^{n} \left[y_i \log \left(\sum_{j=1}^{p} K_{ij} \lambda_j \right) - \sum_{j=1}^{p} K_{ij} \lambda_j \right] + \mathrm{const}$$

Maximum likelihood estimation

Theorem (Vardi et al. (1985))

Assume $K_{ij} > 0$ *and* $\mathbf{y} \neq \mathbf{0}$*. Then the following hold for the log-likelihood* $\log p(\mathbf{y}|\boldsymbol{\lambda})$ *of the unfolding problem:*

1. *The log-likelihood has a maximum.*
2. *The log-likelihood is concave and hence all the maxima are global maxima.*
3. *The maximum is unique if and only if the columns of* $\mathbf{K}$ *are linearly independent*

- So a unique MLE exists when the columns of $\mathbf{K}$ are linearly independent but how do we find it?

Maximum likelihood estimation

Proposition

Let $\mathbf{K}$ *be an invertible square matrix and assume that* $\hat{\boldsymbol{\lambda}} = \mathbf{K}^{-1}\mathbf{y} \geq \mathbf{0}$*. Then* $\hat{\boldsymbol{\lambda}}$ *is the MLE of* $\boldsymbol{\lambda}$*.*

- That is, matrix inversion gives us the MLE if $\mathbf{K}$ is invertible and the resulting estimate is positive
- Note that this result is more restrictive than it may seem
  - $\mathbf{K}$ is often non-square
  - Even if $\mathbf{K}$ was square, it is often not invertible
  - And even if $\mathbf{K}$ was invertible, $\mathbf{K}^{-1}\mathbf{y}$ often contains negative values
- Is there a general recipe for finding the MLE?

Maximum likelihood estimation

- The MLE can always be found computationally by using the *expectation-maximization (EM) algorithm* (Dempster et al. (1977))
  - This is a widely used iterative algorithm for finding maximum likelihood solutions in problems that can be seen as containing incomplete observations
- Starting from some initial value $\boldsymbol{\lambda}^{(0)} > \mathbf{0}$, the EM iteration for unfolding is given by:

$$\lambda_j^{(k+1)} = \frac{\lambda_j^{(k)}}{\sum_{i=1}^{n} K_{ij}} \sum_{i=1}^{n} \frac{K_{ij} y_i}{\sum_{l=1}^{p} K_{il} \lambda_l^{(k)}}, \quad j = 1, \ldots, p$$

- The convergence of this iteration to an MLE (i.e. $\boldsymbol{\lambda}^{(k)} \overset{k\to\infty}{\longrightarrow} \hat{\boldsymbol{\lambda}}_{\mathrm{MLE}}$) was proved by Vardi et al. (1985)

Maximum likelihood estimation

- The EM iteration for finding the MLE in Poisson regression problems has been rediscovered many times in different fields:
  - Optics: Richardson (1972)
  - Astronomy: Lucy (1974)
  - Tomography: Shepp and Vardi (1982); Lange and Carson (1984); Vardi et al. (1985)
  - HEP: Kondor (1983); Mülthei and Schorr (1987); Mülthei et al. (1987, 1989); D'Agostini (1995)
- In modern use, the algorithm is most often called *D'Agostini iteration* in HEP and *Lucy–Richardson deconvolution* in astronomy and optics
- In HEP, also the name "Bayesian unfolding" is used but this is an unfortunate misnomer
  - D'Agostini iteration is a fully frequentist technique for finding the MLE
  - *There is nothing Bayesian about it!*

D'Agostini demo, $k = 0$

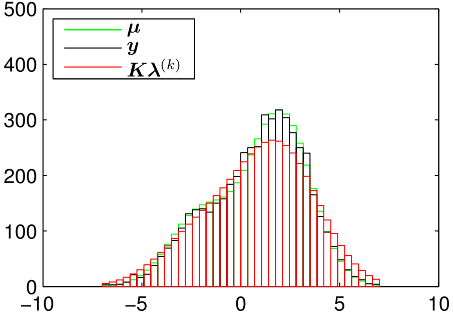


Figure : Smeared histogram

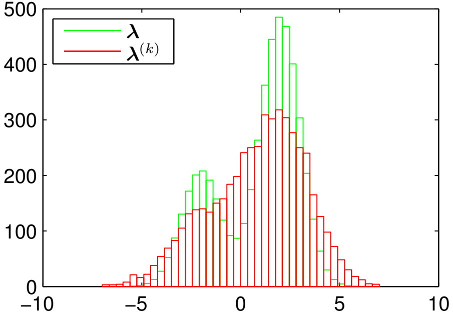


Figure : True histogram

D'Agostini demo, $k = 100$

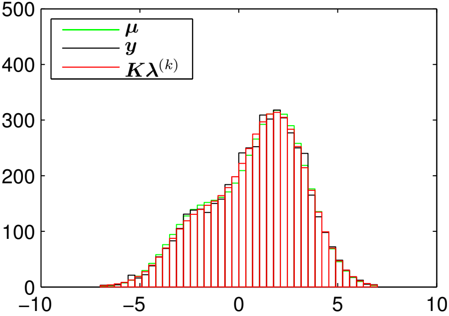


Figure : Smeared histogram

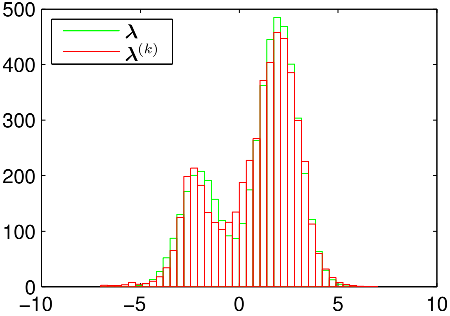


Figure : True histogram

D'Agostini demo, $k = 10000$

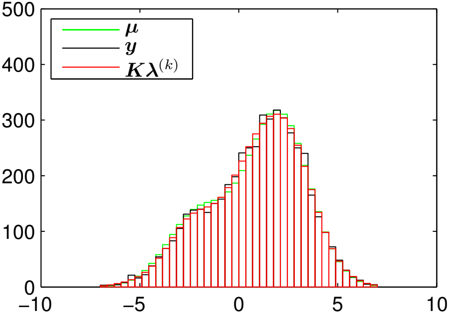


Figure : Smeared histogram

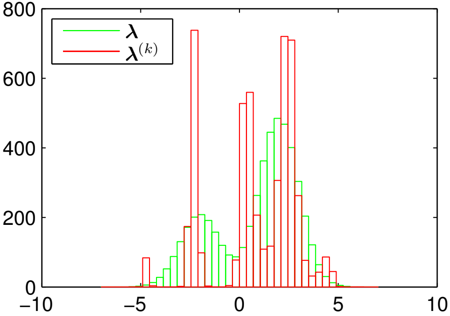


Figure : True histogram

D'Agostini demo, $k = 100000$

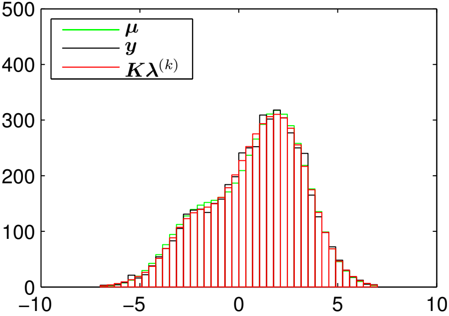


Figure : Smeared histogram

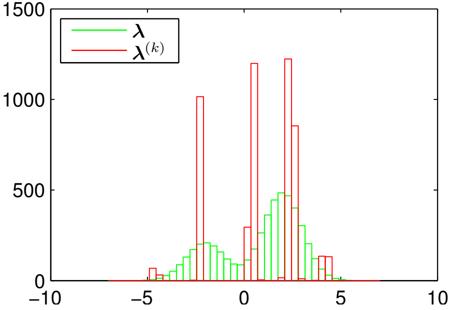


Figure : True histogram

Outline

1. Introduction

2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding

3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix

4. Unfolding with RooUnfold

5. Conclusions

Regularization by early stopping of the EM iteration

- We have seen that unfortunately the MLE itself is often useless
  - Due to the ill-posedness of the problem, it exhibits large, unphysical fluctuations
  - In other words, the likelihood function alone does not contain enough information to constrain the solution
- As the EM iteration proceeds, the solutions will typically first improve but will start to degrade at some point
  - This is because the algorithm will start overfitting to the Poisson fluctuations in $\mathbf{y}$
- This behavior can be exploited by stopping the iteration before unphysical features start to appear
  - The number of iterations k now becomes a *regularization parameter* that controls the trade-off between fitting the data and taming unphysical features

D’Agostini demo, $k = 100$

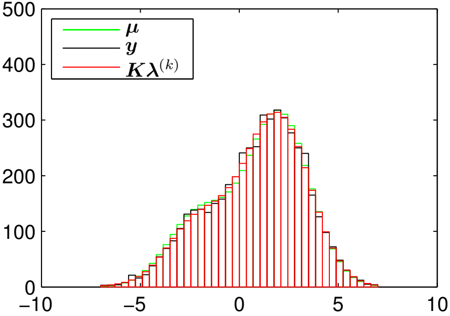


Figure : Smeared histogram

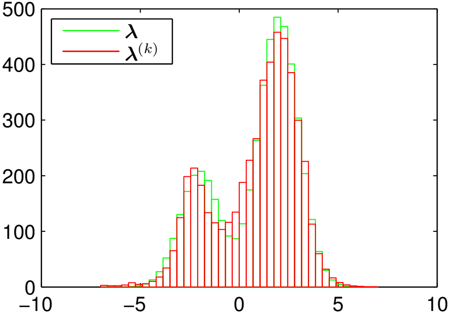


Figure : True histogram

Penalized maximum likelihood estimation

- Early stopping of the EM iteration seems a bit ad-hoc
  - Is there a more principled way of finding good solutions?
- Ideally we would like to find a solution that fits the data but at the same time seems physically plausible
- Let's consider a *penalized maximum likelihood* problem:

$$\max_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} \; F(\boldsymbol{\lambda}) = \log p(\mathbf{y}|\boldsymbol{\lambda}) - \delta P(\boldsymbol{\lambda}),$$

- Here:
  - $P(\boldsymbol{\lambda})$ is a *penalty function* which obtains large values for physically implausible solutions
  - $\delta > 0$ is a *regularization parameter* which controls the balance between maximizing the likelihood and minimizing the penalty
- Typically $P(\boldsymbol{\lambda})$ is a measure of the curvature of the solution
  - I.e., it penalizes for large oscillations

From penalized likelihood to Tikhonov regularization

- To simplify this optimization problem, we use a Gaussian approximation of the Poisson likelihood

$$\mathbf{y}|\boldsymbol{\lambda} \sim \mathrm{Poisson}(\mathbf{K}\boldsymbol{\lambda}) \approx N(\mathbf{K}\boldsymbol{\lambda}, \hat{\mathbf{C}}),$$

  where $\hat{\mathbf{C}} = \mathrm{diag}(\mathbf{y})$

- Hence the objective function becomes:

$$\begin{aligned} F(\boldsymbol{\lambda}) &= \log p(\mathbf{y}|\boldsymbol{\lambda}) - \delta P(\boldsymbol{\lambda}) \\ &= \sum_{i=1}^{n} \left[y_i \log \left(\sum_{j=1}^{p} K_{ij}\lambda_j \right) - \sum_{j=1}^{p} K_{ij}\lambda_j \right] - \delta P(\boldsymbol{\lambda}) + \mathrm{const} \\ &\approx -\frac{1}{2}(\mathbf{y} - \mathbf{K}\boldsymbol{\lambda})^{\mathsf{T}} \hat{\mathbf{C}}^{-1} (\mathbf{y} - \mathbf{K}\boldsymbol{\lambda}) - \delta P(\boldsymbol{\lambda}) + \mathrm{const} \end{aligned}$$

From penalized likelihood to Tikhonov regularization

- We furthermore drop the positivity constraint and absorb the factor $1/2$ into the penalty to obtain

$$\begin{aligned}\hat{\boldsymbol{\lambda}} &= \operatorname*{arg\,max}_{\boldsymbol{\lambda}\in\mathbb{R}^p} -(\mathbf{y}-\mathbf{K}\boldsymbol{\lambda})^{\mathsf{T}}\hat{\mathbf{C}}^{-1}(\mathbf{y}-\mathbf{K}\boldsymbol{\lambda}) - \delta P(\boldsymbol{\lambda}) \\ &= \operatorname*{arg\,min}_{\boldsymbol{\lambda}\in\mathbb{R}^p} (\mathbf{y}-\mathbf{K}\boldsymbol{\lambda})^{\mathsf{T}}\hat{\mathbf{C}}^{-1}(\mathbf{y}-\mathbf{K}\boldsymbol{\lambda}) + \delta P(\boldsymbol{\lambda})\end{aligned}$$

- We see that we have ended up with a penalized χ^2 problem
- This is typically called *(generalized) Tikhonov regularization*

How to choose the penalty?

- The penalty term should reflect the analyst's a priori understanding of the desired solution
- Common choices include:
  - Norm of the solution: $P(\boldsymbol{\lambda}) = \|\boldsymbol{\lambda}\|^2$
  - Curvature of the solution: $P(\boldsymbol{\lambda}) = \|\mathbf{L}\boldsymbol{\lambda}\|^2$, where $\mathbf{L}$ is a discretized 2nd derivative operator
  - SVD unfolding (Höcker and Kartvelishvili, 1996):

$$P(\boldsymbol{\lambda}) = \left\| \mathbf{L} \begin{bmatrix} \lambda_1/\lambda_1^{\mathrm{MC}} \\ \lambda_2/\lambda_2^{\mathrm{MC}} \\ \vdots \\ \lambda_p/\lambda_p^{\mathrm{MC}} \end{bmatrix} \right\|^2 ,$$

    where $\boldsymbol{\lambda}^{\mathrm{MC}}$ is a MC prediction for $\boldsymbol{\lambda}$
  - `TUnfold`[1] (Schmitt, 2012): $P(\boldsymbol{\lambda}) = \|\mathbf{L}(\boldsymbol{\lambda} - \boldsymbol{\lambda}^{\mathrm{MC}})\|^2$

[1] Also more general penalty terms are allowed in `TUnfold`

Least squares estimation with the pseudoinverse

- Consider the least squares problem

$$\min_{\mathbf{x}\in\mathbb{R}^p} \|\mathbf{A}\mathbf{x}-\mathbf{y}\|^2,$$

  where $\mathbf{A} \in \mathbb{R}^{n\times p}$, $\mathbf{x} \in \mathbb{R}^p$ and $\mathbf{y} \in \mathbb{R}^n$

- This problem always has a solution, but it may not be unique
- A solution is always given by the Moore–Penrose pseudoinverse of $\mathbf{A}$:

$$\hat{\mathbf{x}}_{\mathrm{LS}} = \mathbf{A}^{\dagger}\mathbf{y}$$

- When there are multiple solutions, the pseudoinverse gives the one with the smallest norm
- When $\mathbf{A}$ has full column rank, the solution is unique
    - In this special case, the pseudoinverse is given by $\mathbf{A}^{\dagger} = (\mathbf{A}^{\mathsf{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathsf{T}}$
    - Hence, the least squares solution is: $\hat{\mathbf{x}}_{\mathrm{LS}} = (\mathbf{A}^{\mathsf{T}}\mathbf{A})^{-1}\mathbf{A}^{\mathsf{T}}\mathbf{y}$

Finding the Tikhonov regularized solution

- We will now find an explicit form of the Tikhonov regularized estimator

$$\hat{\boldsymbol{\lambda}} = \operatorname*{arg\,min}_{\boldsymbol{\lambda} \in \mathbb{R}^p} \, (\mathbf{y} - \mathbf{K}\boldsymbol{\lambda})^{\mathsf{T}} \hat{\mathbf{C}}^{-1} (\mathbf{y} - \mathbf{K}\boldsymbol{\lambda}) + \delta \|\mathbf{L}\boldsymbol{\lambda}\|^2$$

  by rewriting this as a least squares problem
- This approach also easily generalizes to penalty terms involving $\boldsymbol{\lambda}^{\mathrm{MC}}$
- Let us rewrite:

$$\begin{aligned} \hat{\mathbf{C}}^{-1} &= \mathrm{diag}\left(\frac{1}{y_1}, \ldots, \frac{1}{y_n}\right) \\ &= \underbrace{\mathrm{diag}\left(\frac{1}{\sqrt{y_1}}, \ldots, \frac{1}{\sqrt{y_n}}\right)}_{:=\mathbf{A}} \underbrace{\mathrm{diag}\left(\frac{1}{\sqrt{y_1}}, \ldots, \frac{1}{\sqrt{y_n}}\right)}_{:=\mathbf{A}} \\ &= \mathbf{A}\mathbf{A} = \mathbf{A}^{\mathsf{T}}\mathbf{A} \end{aligned}$$

- Defining $\tilde{\mathbf{y}} := \mathbf{A}\mathbf{y}$ and $\tilde{\mathbf{K}} := \mathbf{A}\mathbf{K}$, our optimization problem becomes

$$\hat{\boldsymbol{\lambda}} = \operatorname*{arg\,min}_{\boldsymbol{\lambda} \in \mathbb{R}^p} \, (\tilde{\mathbf{y}} - \tilde{\mathbf{K}}\boldsymbol{\lambda})^{\mathsf{T}} (\tilde{\mathbf{y}} - \tilde{\mathbf{K}}\boldsymbol{\lambda}) + \delta \|\mathbf{L}\boldsymbol{\lambda}\|^2$$

Finding the Tikhonov regularized solution

- We can rewrite the objective function as follows:

$$\begin{aligned}
&(\tilde{\mathbf{y}} - \tilde{\mathbf{K}}\boldsymbol{\lambda})^{\mathsf{T}}(\tilde{\mathbf{y}} - \tilde{\mathbf{K}}\boldsymbol{\lambda}) + \delta\|\mathbf{L}\boldsymbol{\lambda}\|^2 \\
&= \|\tilde{\mathbf{K}}\boldsymbol{\lambda} - \tilde{\mathbf{y}}\|^2 + \|\sqrt{\delta}\mathbf{L}\boldsymbol{\lambda}\|^2 \\
&= \left\| \begin{bmatrix} \tilde{\mathbf{K}}\boldsymbol{\lambda} - \tilde{\mathbf{y}} \\ \sqrt{\delta}\mathbf{L}\boldsymbol{\lambda} \end{bmatrix} \right\|^2 \\
&= \left\| \begin{bmatrix} \tilde{\mathbf{K}} \\ \sqrt{\delta}\mathbf{L} \end{bmatrix} \boldsymbol{\lambda} - \begin{bmatrix} \tilde{\mathbf{y}} \\ \mathbf{0} \end{bmatrix} \right\|^2
\end{aligned}$$

- Here we recognize a least squares problem, so a minimizer is given by

$$\hat{\boldsymbol{\lambda}} = \begin{bmatrix} \tilde{\mathbf{K}} \\ \sqrt{\delta}\mathbf{L} \end{bmatrix}^{\dagger} \begin{bmatrix} \tilde{\mathbf{y}} \\ \mathbf{0} \end{bmatrix}$$

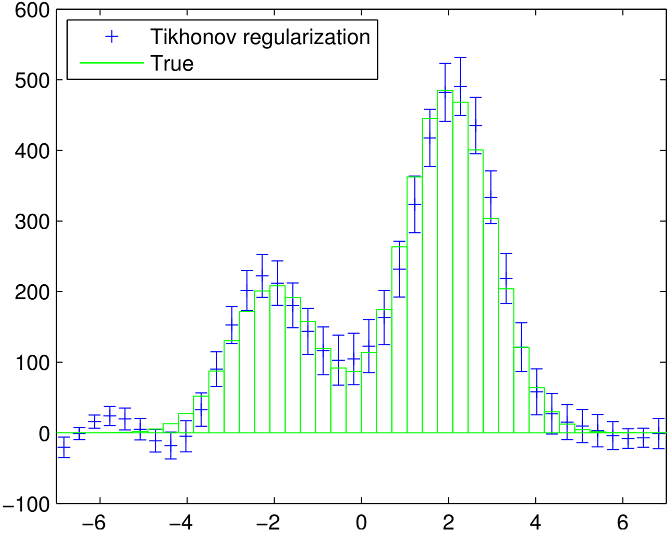
Finding the Tikhonov regularized solution

- Assuming that $\ker(\tilde{\mathbf{K}}) \cap \ker(\mathbf{L}) = \{\mathbf{0}\}$, the minimizer is unique and can be simplified as follows:

$$\begin{aligned}
\hat{\boldsymbol{\lambda}} &= \begin{bmatrix} \tilde{\mathbf{K}} \\ \sqrt{\delta}\mathbf{L} \end{bmatrix}^{\dagger} \begin{bmatrix} \tilde{\mathbf{y}} \\ \mathbf{0} \end{bmatrix} \\
&= \left(\begin{bmatrix} \tilde{\mathbf{K}} \\ \sqrt{\delta}\mathbf{L} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \tilde{\mathbf{K}} \\ \sqrt{\delta}\mathbf{L} \end{bmatrix} \right)^{-1} \begin{bmatrix} \tilde{\mathbf{K}} \\ \sqrt{\delta}\mathbf{L} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} \tilde{\mathbf{y}} \\ \mathbf{0} \end{bmatrix} \\
&= \left(\begin{bmatrix} \tilde{\mathbf{K}}^{\mathsf{T}} & \sqrt{\delta}\mathbf{L}^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{K}} \\ \sqrt{\delta}\mathbf{L} \end{bmatrix} \right)^{-1} \begin{bmatrix} \tilde{\mathbf{K}}^{\mathsf{T}} & \sqrt{\delta}\mathbf{L}^{\mathsf{T}} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{y}} \\ \mathbf{0} \end{bmatrix} \\
&= \left(\tilde{\mathbf{K}}^{\mathsf{T}} \tilde{\mathbf{K}} + \delta \mathbf{L}^{\mathsf{T}} \mathbf{L} \right)^{-1} \tilde{\mathbf{K}}^{\mathsf{T}} \tilde{\mathbf{y}} \\
&= \left(\mathbf{K}^{\mathsf{T}} \hat{\mathbf{C}}^{-1} \mathbf{K} + \delta \mathbf{L}^{\mathsf{T}} \mathbf{L} \right)^{-1} \mathbf{K}^{\mathsf{T}} \hat{\mathbf{C}}^{-1} \mathbf{y}
\end{aligned}$$

- Hence we have obtained an explicit, closed-form solution for the Tikhonov regularization problem

Demonstration of Tikhonov regularization, $P(\boldsymbol{\lambda}) = \|\boldsymbol{\lambda}\|^2$



Outline

1. Introduction
2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding
3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. Conclusions

Bayesian unfolding

- In Bayesian unfolding, the inferences about $\boldsymbol{\lambda}$ are based on the posterior distribution $p(\boldsymbol{\lambda}|\mathbf{y})$
- This is obtained using Bayes' rule:

$$p(\boldsymbol{\lambda}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\lambda})p(\boldsymbol{\lambda})}{p(\mathbf{y})} = \frac{p(\mathbf{y}|\boldsymbol{\lambda})p(\boldsymbol{\lambda})}{\int_{\mathbb{R}_+^p} p(\mathbf{y}|\boldsymbol{\lambda}')p(\boldsymbol{\lambda}')\,\mathrm{d}\boldsymbol{\lambda}'}, \quad \boldsymbol{\lambda} \in \mathbb{R}_+^p$$

  where the likelihood $p(\mathbf{y}|\boldsymbol{\lambda})$ is the same as earlier and $p(\boldsymbol{\lambda})$ is a prior distribution for $\boldsymbol{\lambda}$
- The most common choices as a point estimator of $\boldsymbol{\lambda}$ are:
  - The *posterior mean*: $\hat{\boldsymbol{\lambda}} = \mathrm{E}[\boldsymbol{\lambda}|\mathbf{y}] = \int_{\mathbb{R}_+^p} \boldsymbol{\lambda} p(\boldsymbol{\lambda}|\mathbf{y})\,\mathrm{d}\boldsymbol{\lambda}$
  - The *maximum a posteriori* (MAP) *estimator*: $\hat{\boldsymbol{\lambda}} = \arg\max_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} p(\boldsymbol{\lambda}|\mathbf{y})$
- The width of the posterior distribution $p(\boldsymbol{\lambda}|\mathbf{y})$ can be used to quantify uncertainty regarding $\boldsymbol{\lambda}$
  - But note that the interpretation of the resulting Bayesian *credible intervals* is different from frequentist confidence intervals

Regularization using the prior

- In the Bayesian approach, the prior density $p(\boldsymbol{\lambda})$ regularizes the otherwise ill-posed problem
  - It concentrates the probability mass of the posterior on physically plausible solutions
- The prior is typically of the form

  $$p(\boldsymbol{\lambda}) \propto \exp\left(-\delta P(\boldsymbol{\lambda})\right), \quad \boldsymbol{\lambda} \in \mathbb{R}_+^p,$$

  where $P(\boldsymbol{\lambda})$ is a function characterizing a priori plausible solutions and $\delta > 0$ is a *hyperparameter* controlling the scale of the prior density
- For example, choosing $P(\boldsymbol{\lambda}) = \|\mathbf{L}\boldsymbol{\lambda}\|^2$, where $\mathbf{L}$ a discretized 2nd derivative operator, leads to the positivity-constrained Gaussian smoothness prior

  $$p(\boldsymbol{\lambda}) \propto \exp\left(-\delta\|\mathbf{L}\boldsymbol{\lambda}\|^2\right), \quad \boldsymbol{\lambda} \in \mathbb{R}_+^p$$

Connection between Bayesian unfolding and penalized MLE

- Notice that when $p(\boldsymbol{\lambda}) \propto \exp\left(-\delta P(\boldsymbol{\lambda})\right)$, the Bayesian MAP solution coincides with the penalized maximum likelihood estimator:

$$
\begin{aligned}
\hat{\boldsymbol{\lambda}}_{\mathrm{MAP}} &= \operatorname*{arg\,max}_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} p(\boldsymbol{\lambda}|\mathbf{y}) \\
&= \operatorname*{arg\,max}_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} \log p(\boldsymbol{\lambda}|\mathbf{y}) \\
&= \operatorname*{arg\,max}_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} \log p(\mathbf{y}|\boldsymbol{\lambda}) + \log p(\boldsymbol{\lambda}) \\
&= \operatorname*{arg\,max}_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} \log p(\mathbf{y}|\boldsymbol{\lambda}) - \delta P(\boldsymbol{\lambda}) \\
&= \hat{\boldsymbol{\lambda}}_{\mathrm{PMLE}}
\end{aligned}
$$

- So the penalty term $\delta P(\boldsymbol{\lambda})$ can either be interpreted as a Bayesian prior or as a frequentist regularization term
- The Bayesian interpretation has the advantage that we can visualize the prior $p(\boldsymbol{\lambda})$ by, e.g., drawing samples from it

A note about Bayesian computations

- To be able to compute the posterior mean $\mathrm{E}[\boldsymbol{\lambda}|\mathbf{y}]$ or form the Bayesian credible intervals, we need to be able to evaluate the posterior

$$p(\boldsymbol{\lambda}|\mathbf{y}) = \frac{p(\mathbf{y}|\boldsymbol{\lambda})p(\boldsymbol{\lambda})}{\int_{\mathbb{R}_+^p} p(\mathbf{y}|\boldsymbol{\lambda}')p(\boldsymbol{\lambda}')\,\mathrm{d}\boldsymbol{\lambda}'}$$

- But the denominator is an intractable high-dimensional integral...
- Luckily, it turns out that it is possible to *sample* from the posterior without evaluating the denominator
    - The sample mean and sample quantiles can then be used to compute the posterior mean and the credible intervals
- The class of algorithms that enable this are called Markov chain Monte Carlo (MCMC) samplers and are based on a Markov chain whose equilibrium distribution is the posterior $p(\boldsymbol{\lambda}|\mathbf{y})$
- The single-component Metropolis–Hastings sampler of Saquib et al. (1998) is particularly well-suited for the unfolding problem and seems to also work well in practice

Outline

1. Introduction

2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding

3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix

4. Unfolding with RooUnfold

5. Conclusions

Outline

1. Introduction

2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding

3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix

4. Unfolding with RooUnfold

5. Conclusions

Choice of the regularization strength

- All unfolding methods involve a free parameter controlling the strength of the regularization
  - The parameter δ in Tikhonov regularization and Bayesian unfolding, the number of iterations in D'Agostini
- This parameter is typically difficult to choose using only a priori information
  - But its value usually has a major impact on the unfolded spectrum
- Most LHC analyses choose the regularization parameter using MC studies
  - But this may create an undesired MC bias
- It would be better to choose the regularization parameter based on the observed data $\mathbf{y}$

Data-dependent choice of the regularization strength

- Many methods for using the observed data $\mathbf{y}$ to choose the regularization strength have been proposed in the literature:
  - Goodness-of-fit test in the smeared space (Veklerov and Llacer, 1987)
  - Empirical Bayes estimation (Kuusela and Panaretos, 2014)
  - L-curve (Hansen, 1992)
  - (Generalized) cross validation (Wahba, 1990)
  - ...
- At the moment, we have very limited experience about the relative merits of these methods in HEP unfolding

Goodness-of-fit for choosing the regularization strength

- We present here a simplified version of the procedure proposed by Veklerov and Llacer (1987)
- Let $\hat{\boldsymbol{\mu}} = \mathbf{K}\hat{\boldsymbol{\lambda}}$ be the estimated smeared mean
- Consider the χ^2 statistic

$$T = (\hat{\boldsymbol{\mu}} - \mathbf{y})^{\mathsf{T}} \mathbf{C}^{-1} (\hat{\boldsymbol{\mu}} - \mathbf{y}),$$

  where $\mathbf{C} = \mathrm{diag}(\hat{\boldsymbol{\mu}})$
- If $\mathbf{y} \sim \mathrm{Poisson}(\hat{\boldsymbol{\mu}})$, then asymptotically $T \overset{a}{\sim} \chi^2_n$, where n is the number of bins in $\mathbf{y}$
- Hence, $\mathrm{E}[T] \approx n$
- This suggests that we should choose the regularization strength so that T is as close as possible to n
- Note that this provides a balance between overfitting ($T < n$) and underfitting ($T > n$) the data

Outline

1. Introduction
2. Basic unfolding methodology
    - Maximum likelihood estimation
    - Regularized frequentist techniques
    - Bayesian unfolding
3. Challenges in unfolding
    - Choice of the regularization strength
    - Uncertainty quantification
    - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. Conclusions

Uncertainty quantification

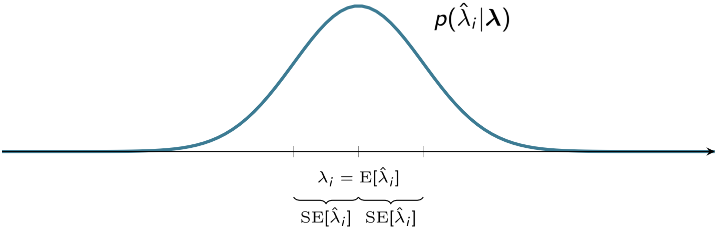
- Proper uncertainty quantification is one of the main challenges in unfolding
- By uncertainty quantification, we mean computing bin-wise frequentist confidence intervals at $1 - \alpha$ confidence level:

$$\inf_{\boldsymbol{\lambda} \in \mathbb{R}_+^p} P_{\boldsymbol{\lambda}}[\hat{\lambda}_{i,L}(\mathbf{y}) \leq \lambda_i \leq \hat{\lambda}_{i,U}(\mathbf{y})] = 1 - \alpha$$

- In practice, we can only hope to satisfy this approximately for finite sample sizes

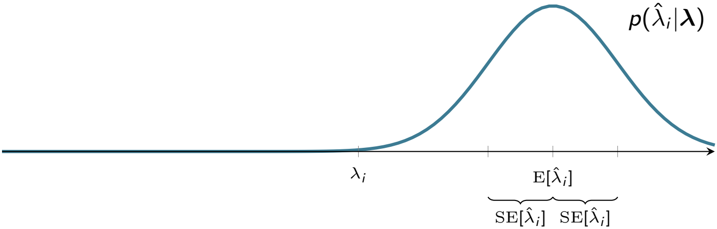
Uncertainty quantification

- Let $\mathrm{SE}[\hat{\lambda}_i]$ be the standard error of $\hat{\lambda}_i$ (i.e., the standard deviation of the sampling distribution of $\hat{\lambda}_i$)
- In many situations, $\hat{\lambda}_i \pm \widehat{\mathrm{SE}}[\hat{\lambda}_i]$ provides a reasonable 68% confidence interval
  - But this is only true when $\hat{\lambda}_i$ is unbiased and has a symmetric sampling distribution
- But in regularized unfolding the estimators are always biased!
  - Regularization reduces variance by increasing the bias (*bias-variance trade-off*)
  - Hence the SE confidence intervals may have lousy coverage

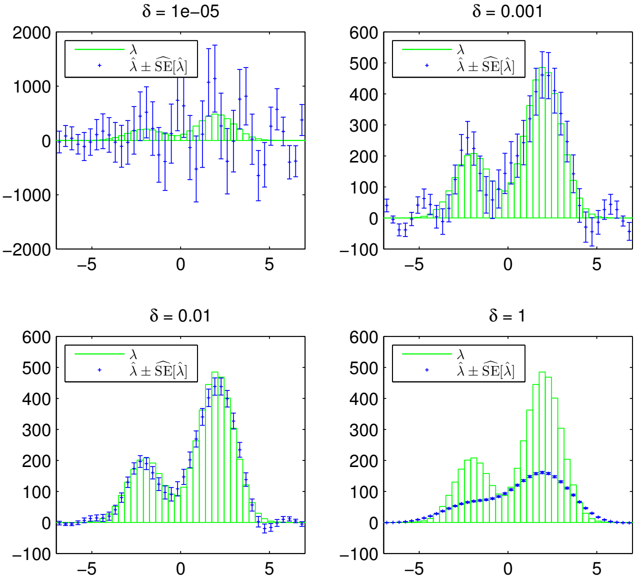

$p(\hat{\lambda}_i|\boldsymbol{\lambda})$
$\lambda_i = \mathrm{E}[\hat{\lambda}_i]$
$\mathrm{SE}[\hat{\lambda}_i]$
$\mathrm{SE}[\hat{\lambda}_i]$

Uncertainty quantification

- Let $\mathrm{SE}[\hat{\lambda}_i]$ be the standard error of $\hat{\lambda}_i$ (i.e., the standard deviation of the sampling distribution of $\hat{\lambda}_i$)
- In many situations, $\hat{\lambda}_i \pm \widehat{\mathrm{SE}}[\hat{\lambda}_i]$ provides a reasonable 68% confidence interval
    - But this is only true when $\hat{\lambda}_i$ is unbiased and has a symmetric sampling distribution
- But in regularized unfolding the estimators are always biased!
    - Regularization reduces variance by increasing the bias (*bias-variance trade-off*)
    - Hence the SE confidence intervals may have lousy coverage


$p(\hat{\lambda}_i|\boldsymbol{\lambda})$
λ_i
$\mathrm{E}[\hat{\lambda}_i]$
$\mathrm{SE}[\hat{\lambda}_i]$
$\mathrm{SE}[\hat{\lambda}_i]$

Demonstration with Tikhonov regularization, $P(\boldsymbol{\lambda}) = \|\boldsymbol{\lambda}\|^2$



Uncertainty quantification

- The uncertainties returned by RooUnfold are estimates of the standard errors computed either using error propagation or resampling
    - Hence these uncertainties should be understood as estimates of the spread of the sampling distribution of $\hat{\boldsymbol{\lambda}}$
    - These should only be understood as approximate confidence intervals if it can be shown that the bias is negligible
- Bootstrap resampling provides an attractive way of forming approximate confidence intervals that take into account the bias and the potential skewness of $p(\hat{\lambda}_i|\boldsymbol{\lambda})$ (Kuusela and Panaretos, 2014)

Outline

1. Introduction
2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding
3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. Conclusions

MC dependence in the smearing matrix

- The smearing matrix $\mathbf{K}$ is typically estimated using Monte Carlo
- In addition to a statistical error due to the finite sample size, there are two sources of systematics in $\mathbf{K}$:
  1. The matrix depends on the shape of the spectrum within each true bin
  $$K_{ij} = \frac{\int_{F_i} \int_{E_j} k(y,x) f(x) \, \mathrm{d}x \, \mathrm{d}y}{\int_{E_j} f(x) \, \mathrm{d}x}, \quad i = 1, \ldots, n, \quad j = 1, \ldots, p$$
  2. The smearing of the variable of interest may depend on the MC distribution of some auxiliary variables
     - For example, the energy resolution of jets depends on the pseudorapidity distribution of the jets
- The first problem can be alleviated by making the true bins smaller at the cost of increased ill-posedness of the problem

Outline

1. Introduction

2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding

3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix

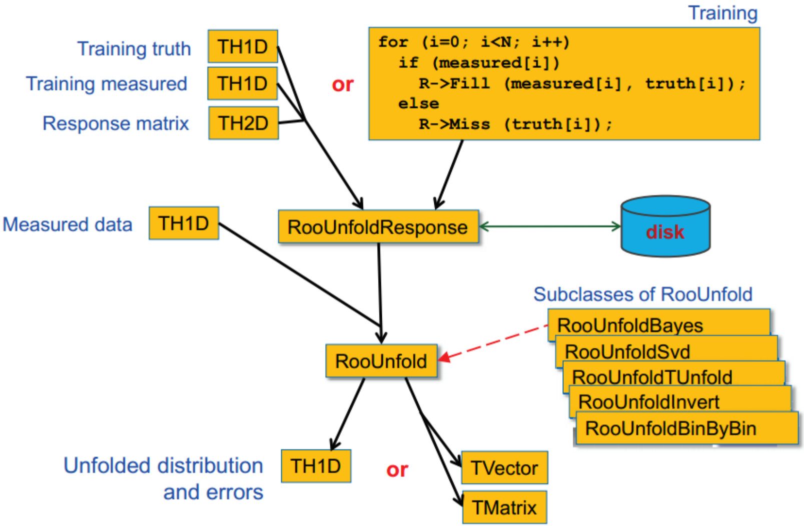
4. Unfolding with RooUnfold

5. Conclusions

Introduction to RooUnfold

- RooUnfold (Adye, 2011) an unfolding framework for ROOT that provides an interface for many standard unfolding methods
- Written by Tim Adye, Richard Claridge, Kerstin Tackmann and Fergus Wilson
- RooUnfold is currently the most commonly used unfolding framework among the LHC experiments although other implementations are also occasionally used
- RooUnfold includes the following unfolding techniques:
    1. Matrix inversion
    2. D'Agostini iteration
    3. The SVD flavor of Tikhonov regularization
    4. The `TUnfold` flavor of Tikhonov regularization
- There is also an implementation for the so-called bin-by-bin unfolding technique
    - This is an obsolete method that replaces the full response matrix $\mathbf{K}$ by a diagonal approximation and while doing so introduces a huge MC bias
    - This method should not be used!

RooUnfold classes



```
for (i=0; i<N; i++)
  if (measured[i])
    R->Fill (measured[i], truth[i]);
  else
    R->Miss (truth[i]);
```

Figure from Adye (2011)

RooUnfoldInvert

```
RooUnfoldInvert(const RooUnfoldResponse* res, const TH1*
meas, const char* name = 0, const char* title = 0)
```

- This is the most basic method: it estimates $\boldsymbol{\lambda}$ using $\hat{\boldsymbol{\lambda}} = \mathbf{K}^{-1}\mathbf{y}$
- Remember that when $\hat{\boldsymbol{\lambda}}$ is positive, this is the MLE
- `res` contains the response matrix $\mathbf{K}$
- `meas` contains the smeared data $\mathbf{y}$
- The standard error of $\hat{\boldsymbol{\lambda}}$ is estimated using error propagation

RooUnfoldBayes

`RooUnfoldBayes(const RooUnfoldResponse* res, const TH1* meas, Int_t niter = 4, Bool_t smoothit = false, const char* name = 0, const char* title = 0)`

- This implements the D'Agostini/Lucy-Richardson/EM iteration for finding the MLE
- Remember that despite the name this is not a Bayesian technique
- The iteration is started from the MC spectrum, i.e., $\boldsymbol{\lambda}^{(0)} = \boldsymbol{\lambda}^{\mathrm{MC}}$ contained in `res`
- `niter` is the number of iterations
  - For small `niter`, the solution is biased towards $\boldsymbol{\lambda}^{\mathrm{MC}}$; for large `niter`, we get a solution close to the MLE
  - Note that the default `niter = 4` is completely arbitrary and with no optimality guarantees
- `smoothit` can be used to enable a smoothed version of the EM iteration (outside the scope of this course)
- By default, the standard error of $\hat{\boldsymbol{\lambda}}$ is estimated using error propagation at each iteration of the algorithm

RooUnfoldSvd

`RooUnfoldSvd(const RooUnfoldResponse* res, const TH1* meas, Int_t kreg = 0, Int_t ntoyssvd = 1000, const char* name = 0, const char* title = 0)`

- This implements the SVD flavor of Tikhonov regularization, i.e.,

$$\hat{\boldsymbol{\lambda}} = \operatorname*{arg\,min}_{\boldsymbol{\lambda} \in \mathbb{R}^p} (\mathbf{y} - \mathbf{K}\boldsymbol{\lambda})^{\mathsf{T}} \hat{\mathbf{C}}^{-1} (\mathbf{y} - \mathbf{K}\boldsymbol{\lambda}) + \delta \left\| \mathbf{L} \begin{bmatrix} \lambda_1/\lambda_1^{\mathrm{MC}} \\ \lambda_2/\lambda_2^{\mathrm{MC}} \\ \vdots \\ \lambda_p/\lambda_p^{\mathrm{MC}} \end{bmatrix} \right\|^2 ,$$

  where $\boldsymbol{\lambda}^{\mathrm{MC}}$ is again contained in `res`
- This is a wrapper for the `TSVDUnfold` class by K. Tackmann
- `kreg` chooses the number of significant singular values in a certain transformation of the smearing matrix $\mathbf{K}$
  - Small `kreg` corresponds to a large δ and a large `kreg` to a small δ
- The standard error of $\hat{\boldsymbol{\lambda}}$ is estimated by resampling `ntoyssvd` observations
  - Also includes a contribution from the uncertainty of $\mathbf{K}$

RooUnfoldTUnfold

`RooUnfoldTUnfold(const RooUnfoldResponse* res, const TH1* meas, TUnfold::ERegMode reg = TUnfold::kRegModeDerivative, const char* name = 0, const char* title = 0)`

- This implements the `TUnfold` flavor of Tikhonov regularization, i.e.,

  $$\hat{\boldsymbol{\lambda}} = \underset{\boldsymbol{\lambda}\in\mathbb{R}^p}{\arg\min}\ (\mathbf{y}-\mathbf{K}\boldsymbol{\lambda})^{\mathsf{T}}\hat{\mathbf{C}}^{-1}(\mathbf{y}-\mathbf{K}\boldsymbol{\lambda}) + \delta\|\mathbf{L}(\boldsymbol{\lambda}-\boldsymbol{\lambda}^{\mathrm{MC}})\|^2,$$

  where the minimizer is found subject to an additional area constraint[2]
- This is a wrapper for the `TUnfold` class by S. Schmitt
  - `TUnfold` actually provides a lot of extra functionality which cannot be accessed through RooUnfold
- The form of the matrix **L** is chosen using `reg`
  - The supported choices are identity, 1st derivative and 2nd derivative
- The regularization parameter δ is chosen using the `SetRegParm(Double_t parm)` method
  - If δ is not chosen manually, it is found automatically using the L-curve technique, but this only seems to work when $n \gg p$

[2] In the case of the `TUnfold` wrapper, the RooUnfold documentation is not explicit about the choice of $\boldsymbol{\lambda}^{\mathrm{MC}}$ (it does not seem to come from `res` in this case)

RooUnfold practical

- Start by downloading the code template at:

  `www.cern.ch/mkuusela/ETH_workshop_July_2014/RooUnfoldExercise.cxx`
- A set of exercises based on this code can be found at:

  `www.cern.ch/mkuusela/ETH_workshop_July_2014/practical.pdf`
- Useful supplementary material
  - These slides:

    `www.cern.ch/mkuusela/ETH_workshop_July_2014/slides.pdf`
  - RooUnfold website:

    `http://hepunx.rl.ac.uk/~adye/software/unfold/RooUnfold.html`
  - RooUnfold class documentation:

    `http://hepunx.rl.ac.uk/~adye/software/unfold/htmldoc/RooUnfold.html`

Outline

1. Introduction
2. Basic unfolding methodology
   - Maximum likelihood estimation
   - Regularized frequentist techniques
   - Bayesian unfolding
3. Challenges in unfolding
   - Choice of the regularization strength
   - Uncertainty quantification
   - MC dependence in the smearing matrix
4. Unfolding with RooUnfold
5. **Conclusions**

Conclusions

- Unfolding is a complex data analysis task that involves several assumptions and approximations
    - It is crucial to understand the ingredients that go into an unfolding procedure
    - Unfolding algorithms should never be used as black boxes!
- All unfolding methods are based on complementing the likelihood by additional information about physically plausible solutions
- The most popular techniques are the D'Agostini iteration and various flavors of Tikhonov regularization
- Beware when using RooUnfold that:
    - There is a MC dependence in both the smearing matrix and the regularization
    - The uncertainties should be understood as standard errors and do not necessarily provide good coverage properties
    - The regularization parameter has a major impact on the solution and should be chosen in a data-dependent way
- There is plenty room for further improvements in both unfolding methodology and software

References I

T. Adye. Unfolding algorithms and tests using RooUnfold. In H. B. Prosper and L. Lyons, editors, *Proceedings of the PHYSTAT 2011 Workshop on Statistical Issues Related to Discovery Claims in Search Experiments and Unfolding*, CERN-2011-006, pages 313–318, CERN, Geneva, Switzerland, 17–20 January 2011.

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

A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1–38, 1977.

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.

References II

A. Kondor. Method of convergent weights – An iterative procedure for solving Fredholm's integral equations of the first kind. *Nuclear Instruments and Methods*, 216:177–181, 1983.

M. Kuusela and V. M. Panaretos. Empirical Bayes unfolding of elementary particle spectra at the Large Hadron Collider. arXiv:1401.8274 [stat.AP], 2014.

K. Lange and R. Carson. EM reconstruction algorithms for emission and transmission tomography. *Journal of Computer Assisted Tomography*, 8(2): 306–316, 1984.

L. B. Lucy. An iterative technique for the rectification of observed distributions. *Astronomical Journal*, 79(6):745–754, 1974.

H. N. Mülthei and B. Schorr. On an iterative method for the unfolding of spectra. *Nuclear Instruments and Methods in Physics Research A*, 257:371–377, 1987.

H. N. Mülthei, B. Schorr, and W. Törnig. On an iterative method for a class of integral equations of the first kind. *Mathematical Methods in the Applied Sciences*, 9:137–168, 1987.

References III

H. N. Mülthei, B. Schorr, and W. Törnig. On properties of the iterative maximum likelihood reconstruction method. *Mathematical Methods in the Applied Sciences*, 11:331–342, 1989.

W. H. Richardson. Bayesian-based iterative method of image restoration. *Journal of the Optical Society of America*, 62(1):55–59, 1972.

S. S. Saquib, C. A. Bouman, and K. Sauer. ML parameter estimation for Markov random fields with applications to Bayesian tomography. *IEEE Transactions on Image Processing*, 7(7):1029–1044, 1998.

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

L. A. Shepp and Y. Vardi. Maximum likelihood reconstruction for emission tomography. *IEEE Transactions on Medical Imaging*, 1(2):113–122, 1982.

Y. Vardi, L. Shepp, and L. Kaufman. A statistical model for positron emission tomography. *Journal of the American Statistical Association*, 80(389):8–20, 1985.

References IV

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.

G. Wahba. *Spline Models for Observational Data*. SIAM, 1990.

Backup

Uncertainty quantification with the bootstrap

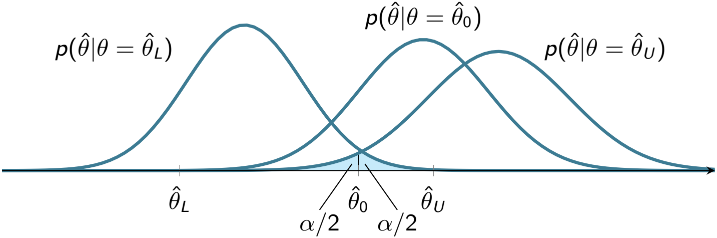
- The bootstrap sample can be obtained as follows:
  1. Unfold $\mathbf{y}$ to obtain $\hat{\boldsymbol{\lambda}}$
  2. Fold $\hat{\boldsymbol{\lambda}}$ to obtain $\hat{\boldsymbol{\mu}} = \mathbf{K}\hat{\boldsymbol{\lambda}}$
  3. Obtain a resampled observation $\mathbf{y}^* \sim \mathrm{Poisson}(\hat{\boldsymbol{\mu}})$
  4. Unfold $\mathbf{y}^*$ to obtain $\hat{\boldsymbol{\lambda}}^*$
  5. Repeat R times from 3
- The bootstrap sample $\{\hat{\boldsymbol{\lambda}}^{*(r)}\}_{r=1}^{R}$ follows the sampling distribution of $\hat{\boldsymbol{\lambda}}$ if the true value of $\boldsymbol{\lambda}$ was the observed value of our estimator
  - I.e., it is our best understanding of the sampling distribution of $\hat{\boldsymbol{\lambda}}$ for the data at hand
- This procedure also enables us to take into account the data-dependent choice of the regularization strength
  - This is very difficult to do using competing methods

Uncertainty quantification with the bootstrap

- The bootstrap sample can be used to compute $1-\alpha$ *basic bootstrap intervals* to serve as approximate $1-\alpha$ confidence intervals for λ_i:

$$[\hat{\lambda}_{i,L}, \hat{\lambda}_{i,U}] = [2\hat{\lambda}_i - \hat{\lambda}^*_{i,1-\alpha/2}, 2\hat{\lambda}_i - \hat{\lambda}^*_{i,\alpha/2}],$$

  where $\hat{\lambda}^*_{i,\alpha}$ denotes the α-quantile of the bootstrap sample $\{\hat{\lambda}_i^{*(r)}\}_{r=1}^R$
- This can be understood as the bootstrap analogue of the Neyman construction of confidence intervals

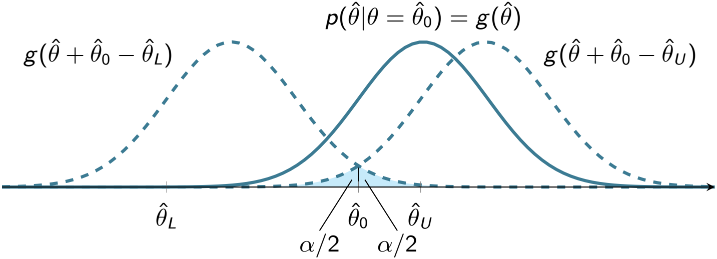

$p(\hat{\theta}|\theta = \hat{\theta}_0)$
$p(\hat{\theta}|\theta = \hat{\theta}_L)$
$p(\hat{\theta}|\theta = \hat{\theta}_U)$
$\hat{\theta}_L$
$\hat{\theta}_0$
$\hat{\theta}_U$
$\alpha/2$
$\alpha/2$

Uncertainty quantification with the bootstrap

- The bootstrap sample can be used to compute $1-\alpha$ *basic bootstrap intervals* to serve as approximate $1-\alpha$ confidence intervals for λ_i:

$$[\hat{\lambda}_{i,L}, \hat{\lambda}_{i,U}] = [2\hat{\lambda}_i - \hat{\lambda}^*_{i,1-\alpha/2}, 2\hat{\lambda}_i - \hat{\lambda}^*_{i,\alpha/2}],$$

  where $\hat{\lambda}^*_{i,\alpha}$ denotes the α-quantile of the bootstrap sample $\{\hat{\lambda}_i^{*(r)}\}_{r=1}^R$
- This can be understood as the bootstrap analogue of the Neyman construction of confidence intervals


$p(\hat{\theta}|\theta = \hat{\theta}_0) = g(\hat{\theta})$
$g(\hat{\theta} + \hat{\theta}_0 - \hat{\theta}_L)$
$g(\hat{\theta} + \hat{\theta}_0 - \hat{\theta}_U)$
$\hat{\theta}_L$
$\hat{\theta}_0$
$\hat{\theta}_U$
$\alpha/2$
$\alpha/2$

Demonstration

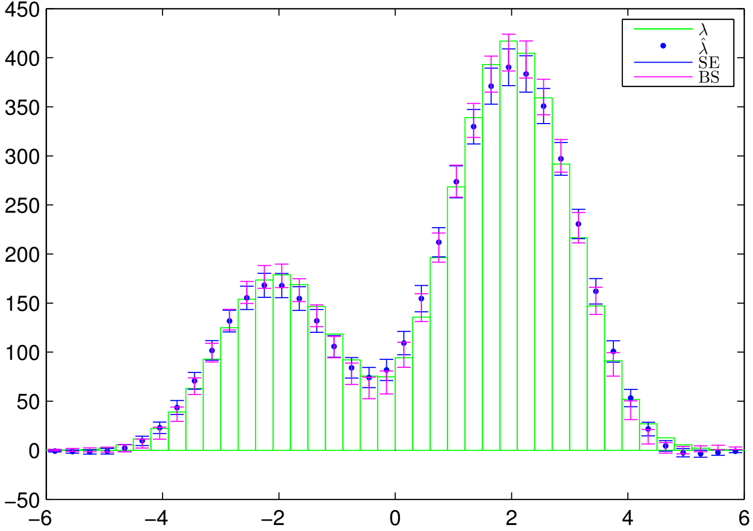


Figure : Tikhonov regularization with 95% bin-wise confidence intervals. The SE intervals cover in 23 bins out of 40, while the bootstrap intervals cover in 32 bins.