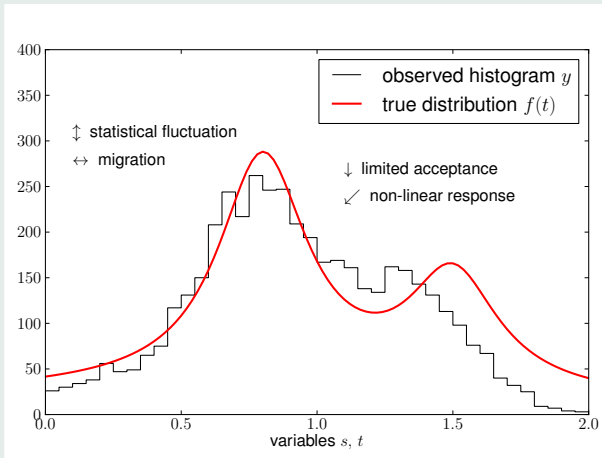# Data Unfolding

Volker Blobel   –   Universität Hamburg

Unfolding is required, due to migration effects, for the measurement of distributions in HEP, where the quantity of interest is often measured indirectly. Unfolding is an linear inverse problem with a coefficient matrix, which is usually ill-conditioned. Techniques of standard linear algebra no longer apply and the numerical treatment becomes more difficult. Available additional information can be used to stabilize the solution, without introducing a significant bias. The use of these regularization methods requires some insight into statistical behavior and mathematical operations.

1. **Unfolding and the inverse problem**

2. **Why are inverse problems ill-posed?**

3. **Regularization methods**

4. **Unfolding in physics**

5. **Results and correlations**

6. **Unfolding software**

7. **Future developments**

1. Unfolding and the inverse problem

Cross section x_i as a function of a variable is measured in bins (bin index i), in ideal case of perfect resolution and for acceptance A_{ii}:

$$(\text{cross section})_i \ \equiv x_i = \frac{n_{i,\text{cand}} - \tau \cdot n_{i,\text{bg}}}{A_{ii} \ \left[\int \mathcal{L} \, dt\right]}$$

The measurement of distributions, cross sections in HEP ... is complicated by

- **migration effects, bias, limited acceptance**, and **limited statistical precision**

An event originating from bin j is measured in another bin i, due to limited detector resolution. It is impossible to consider one bin without other bins: $A \to \boldsymbol{A} =$ matrix with elements A_{ij}:

$$\text{expected:} \qquad \left[\int \mathcal{L} \, dt\right] \cdot \sum_j A_{ij} \, x_j + \tau \cdot n_{i,\text{bg}} = n_{i,\text{cand}} \qquad \text{measured}$$

$$\text{using matrix formalism:} \qquad \left[\int \mathcal{L} \, dt\right] \cdot \boldsymbol{A}\,\boldsymbol{x} + \tau \boldsymbol{n}_{\text{bg}} = \boldsymbol{n}_{\text{cand}}$$

In the following the equations are written without the factor $\left[\int \mathcal{L} \, dt\right]$ and without background, and the measured histogram-vector $\boldsymbol{y}$:

$$\boxed{\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{y} + \boldsymbol{e}}$$

(vector $\boldsymbol{e} =$ error) to be solved for the cross section $\boldsymbol{x}$, given $\boldsymbol{A}$ and the measurement $\boldsymbol{y}$.

Direct and inverse processes

The process of the transition between the **true distribution** $f(t)$ and the **measured distribution** $g(s)$ for linear inverse problems is described by the Fredholm integral equation of the first kind:

$$\int_\Omega K(s,t)\, f(t)\, \mathrm{d}t + b(s) = g(s)$$

($b(s)$ = background contribution). Two types of processes are based on the integral equation:

> **direct process (MC)**   true/MC dist. $f(t) \Longrightarrow g(s)$ measured dist.
> **inverse process (unfolding)**   measured dist. $g(s) \Longrightarrow f(t)$ true dist.

Discretization: the integral equation becomes an (usually ill-posed) linear system of equations:

$$\boldsymbol{A}\boldsymbol{x} = \boldsymbol{y} + \boldsymbol{e} \qquad\qquad y_i = \int_{s_{i-1}}^{s_i} g(s)\, \mathrm{d}s \quad i = 1,\, 2,\, \ldots,\, m$$

(assuming a case without background contribution) with the representation

$$\text{true distribution } f(t) \Rightarrow \boldsymbol{x} \qquad n\text{-vector of unknowns}$$
$$\text{measured distibution } g(s) \Rightarrow \boldsymbol{y} \qquad m\text{-vector of measured data}$$
$$\text{Kernel } K(s,t) \Rightarrow \boldsymbol{A} \qquad \text{rectangular } m\text{-by-}n \text{ response matrix .}$$

The variables s, t and vectors $\boldsymbol{x}$, $\boldsymbol{y}$ can be multi-dimensional. Elements of the response matrix $\boldsymbol{A}$ are positive, usually probabilities, and include efficiency.

Discretization

- Several different methods can be applied in the discretization: <u>simple binning</u>, basis functions: orthogonal or B-splines (allows re-weighting of MC) . . . ;
- unfolding is independent of the assumed $f(t)^{\mathrm{MC}}$, if the Fredholm equation above is correct.

Problems in <u>Particle Physics</u> differ from problems in other fields:

- input errors are well-known (Poisson data, . . . covariance matrix $\boldsymbol{V}_y$);
- covariance matrix of result is required, no bias, small correlations;
- dimension parameters are small compared to other fields;
- the response matrix $\boldsymbol{A}$ has to be determined in particle physics from MC sample (statistical errors).

Unfolding with the inverse transition is a complex mathematical and statistical problem (ill-posed problem, instability of solution) and requires a good understanding of the detector and the measurement process, decribed in the MC.
Straightforward methods can result in solutions which look chaotic. Alternative home-made methods usually produce biased results.

The response $\boldsymbol{A}$ should depend only on the detector properties, it should **not** depend on the expected result.
The use of histograms for the data and the response matrix (simple binning) may not be optimal.

y measured
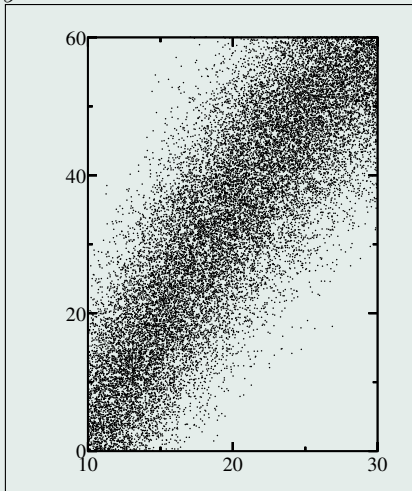
x true

The response matrix $\boldsymbol{A}$ is generated from Monte Carlo $x - y$-pairs.

Example: plots of x versus y, with small non-linearity in $y = y(x)$ . . .

How many bins should be used for x and y?

The resolution is deteriorated, if too few bins are used (m too small).

$$\boxed{\text{use number of } y\text{-bins } m \gtrsim 2n}$$

Never use $n = m$ with identical bins! –

"inverse crime": . . . the numerical methods contain features that effectively render the inverse problem less ill-posed than it actually is, thus yielding unrealistically optimistic results.

Literature

PHYSTAT 2011 Workshop, **Statistical Issues Related to Discovery Claims in Search Experiments and Unfolding**, Ed. Harrison Prosper, Louis Lyons, CERN-2011-006 (Yellow Report)

Per Christian Hansen: **Rank-deficient and Discrete Ill-Posed Problems: Numerical Aspects of Linear Inversion**,
SIAM monographs on mathematical modeling and computation. Society for Industrial and Applied Mathematics, Philadelphia, 1997.

...: **Discrete Inverse Problems – Insight and Algorithms**,
Fundamentals of Algorithms. Society for Industrial and Applied Mathematics, Philadelphia, 2010.

Jari Kaipio and Erkki Somersalo: **Statistical and Computational Inverse Problems**,
Applied Mathematical Science. Springer, 2004.

...: **Statistical Inverse problems: Discretization, Model Reduction and Inverse Crimes**,
Journal of Computational and Applied Mathematics, 198:493 – 504, 2007.

Curtis R. Vogel: **Computational Methods for Inverse Problems**,
SIAM Frontiers in Applied mathematics. Society for Industrial and Applied Mathematics, Philadelphia, 2002.

Andreas Rieder: **Keine Probleme mit Inversen Problemen**.
Vieweg, 2003.

Ake Bjork: **Numerical Methods for Least Squares Problems**.
Society for Industrial and Applied Mathematics, Philadelphia, 1996.

2. Why are inverse problems ill-posed?

Inverse problems, described by the Fredhold equation of first kind . . .

$$\int_{\Omega} K(s,t)\ f(t)\,\mathrm{d}t + b(s) = g(s)$$

$$\boldsymbol{A} \quad \boldsymbol{x} \quad + \boldsymbol{b} = \boldsymbol{y} \ + \boldsymbol{e}$$

arise quite naturally.

$$\text{measured values} \quad \boldsymbol{y} = \text{Poisson-distributed} \ \Rightarrow \ \text{ML}$$
$$= \text{normal-distributed} \ \Rightarrow \ \text{LS with normalization: } e \in N(0,1) \quad \boldsymbol{V}_y = \boldsymbol{I}$$

A problem is called ill-posed, if "a small perturbation of the data can cause a large perturbation" of the solution. The system of linear equations from the discretization has a ill-conditioned matrix with a large condition number κ, even if there are no measurement errors.

In particle physics there are statistical errors in the measurement by the detector and in the MC simulation of the detector response, plus systematic errors – but we want a numerical result with information on uncertainties (i.e. the covariance matrix).
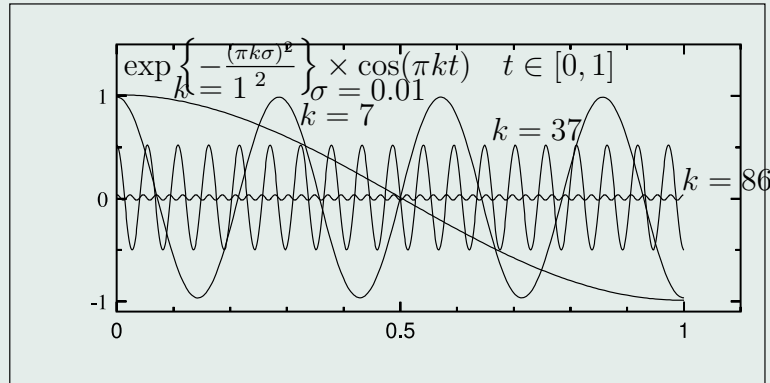
Effect of a Gaussian resolution

An even function $f(t)$ with period 1 can be approximated by a sum with n terms

$$f(t) \approx a_0 + \sum_{k=1}^{n-1} a_k \cos(\pi k t) \qquad g(s) \approx \alpha_0 + \sum_{k=1}^{n-1} \alpha_k \cos(\pi k s)$$

Convolution of functions $\cos(\pi k t)$ by a kernel function $K(s,t) \equiv K(s-t)$ given by a Gaussian resolution function (standard deviation σ):

$$\int_{-\infty}^{+\infty} \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(s-t)^2}{2\sigma^2}\right) \times \cos(\pi k t)\, dt = \exp\left(-\frac{(\pi k \sigma)^2}{2}\right) \times \cos(\pi k s)\,,$$

The amplitude is attenuated by an **exponential factor**, which will become $\ll 1$ for larger values of k.

$$\exp\left\{-\frac{(\pi k \sigma)^2}{2}\right\} \times \cos(\pi k t) \quad t \in [0,1]$$
$$\sum_{k=1}^{} \quad \sigma = 0.01$$
$k = 7$ $\quad k = 37$ $\quad k = 86$

. . . recommended by mathematicians.

Decomposition of the <u>general</u> m-by-n matrix matrix $\boldsymbol{A}$ (assuming $m \geq n$):

$$\boldsymbol{A} = \boldsymbol{U\Sigma V}^T = \sum_{i=1}^{n} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^{\mathrm{T}}$$

$$\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A} = \left(\boldsymbol{U\Sigma V}^{\mathrm{T}}\right)^{\mathrm{T}} \boldsymbol{U\Sigma V}^{\mathrm{T}} = \boldsymbol{V\Sigma}^2 \boldsymbol{V}^{\mathrm{T}} \qquad \text{(diagonalization of \underline{symmetric} matrix)}$$

$$\boldsymbol{U} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_n) \in \mathbb{R}^{m \times n} \qquad \text{orthonormal columns}$$

$$\boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_n) \in \mathbb{R}^{n \times n} \qquad \text{orthonormal columns}$$

$$\boldsymbol{\Sigma} = \mathbf{diag}\left\{\sigma_1, \ldots, \sigma_n\right\} = \boldsymbol{U}^{\mathrm{T}}\boldsymbol{A}\boldsymbol{V} \qquad \text{diagonal}$$

with non-negative diagonal elements σ_i, called singular values, in non-increasing order.

The SVD allows the least squares solution of the linear system $\boldsymbol{Ax} = \boldsymbol{y} + \boldsymbol{e}$ for $\boldsymbol{x}$, by the product with the <u>generalized inverse $\boldsymbol{A}^{\dagger}$</u>:

$$\boldsymbol{A}^{\dagger} = \boldsymbol{V\Sigma}^{-1}\boldsymbol{U}^{\mathrm{T}}$$

$$\boldsymbol{A}^{\dagger}\boldsymbol{A} = \boldsymbol{V\Sigma}^{-1}\boldsymbol{U}^{\mathrm{T}}\,\boldsymbol{U\Sigma V}^T = \boldsymbol{I}$$

Left- and right-singular vectors

SVD ... a new way to see into the heart of a matrix.
Gilbert Strang

$$\text{Singular value decomposition} \quad \boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T = \sum_{i=1}^{n} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^{\mathrm{T}}$$

$$
\overset{(n)}{\underset{(m)}{\begin{pmatrix} \boldsymbol{A} \end{pmatrix}}}
=
\begin{pmatrix} \boldsymbol{u}_1 & \boldsymbol{u}_2 & \cdots & \boldsymbol{u}_{n-1} & \boldsymbol{u}_n \end{pmatrix}
\cdot
\begin{pmatrix} \sigma_1 & & & & \boldsymbol{0} \\ & \sigma_2 & & & \\ & & \ddots & & \\ & & & \sigma_{n-1} & \\ \boldsymbol{0} & & & & \sigma_n \end{pmatrix}
\begin{pmatrix} \boldsymbol{v}_1^{\mathrm{T}} \\ \boldsymbol{v}_2^{\mathrm{T}} \\ \cdots \\ \boldsymbol{v}_{n-1}^{\mathrm{T}} \\ \boldsymbol{v}_n^{\mathrm{T}} \end{pmatrix}
$$

Matrix $\boldsymbol{\Sigma} = \mathrm{diag}(\sigma_1, \ldots, \sigma_n)$ with ordered singular values $\sigma_1 \geq \sigma_2 \geq \ldots \sigma_n \geq 0$.

$$\boldsymbol{U}^{\mathrm{T}}\boldsymbol{U} = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{V} = \boldsymbol{V}\boldsymbol{V}^{\mathrm{T}} = \boldsymbol{I}$$

Right-singular vectors are equal to the eigenvectors of $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ (least squares normal eauations).

Spectrum of singular values

Spectrum of singular values of the Gaussian response matrix $\boldsymbol{A}$. The bars are for a Gaussian resolution of σ_{left} equal to the bin width (left bar) and $\sigma_{\text{right}} = \sigma_{\text{left}}/2$ (right bar).

spectrum of singular values
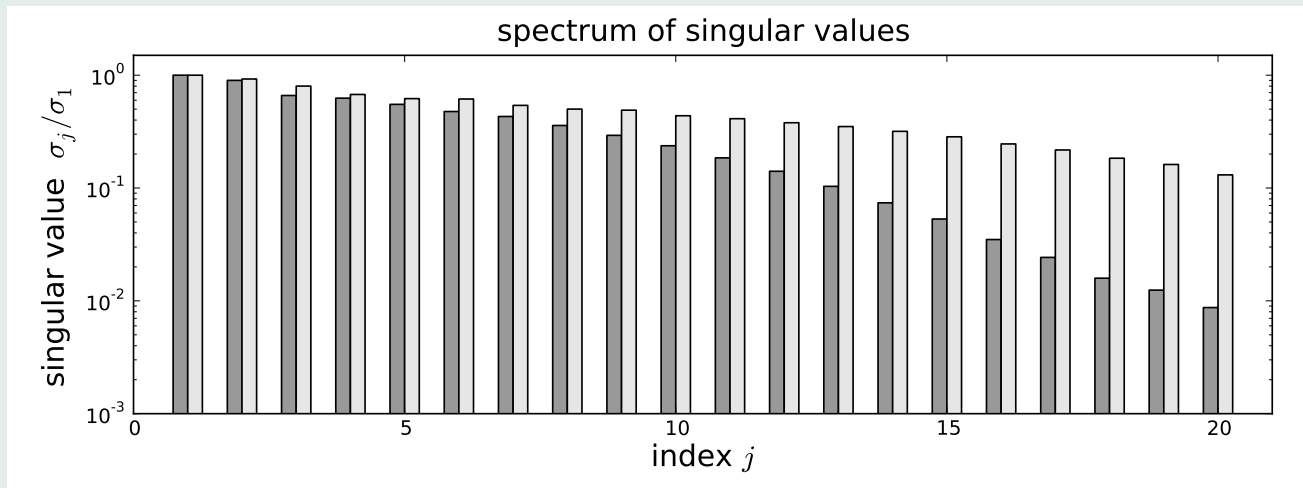
The SVD matrices represent solely the properties of the response matrix of the measurement process.

Solutions of matrix equations are <u>inaccurate</u> in case of a <u>large condition number</u> $\kappa = \sigma_1/\sigma_n$.

The direct and the inverse problem

(1) The direct problem:

$$\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} = \boldsymbol{U}\,\boldsymbol{\Sigma}\boldsymbol{V}^T\boldsymbol{x} = \sum_{j=1}^{n} \sigma_j \left(\boldsymbol{v}_j^{\mathrm{T}}\boldsymbol{x}\right)\boldsymbol{u}_j$$

The "true" vector $\boldsymbol{x}$ is decomposed into components $\left(\boldsymbol{v}_j^{\mathrm{T}}\boldsymbol{x}\right)$, and the expected "measured" vector $\boldsymbol{y}$ is a superposition of the vectors $\boldsymbol{u}_j$, weighted with singular values σ_j. Components with $\sigma_j = 0$ or $\sigma_j \ll 1$ *disappear* in the real measured vector $\boldsymbol{y} + \boldsymbol{e}$.

(2) The inverse problem: least squares solution using generalized inverse $\boldsymbol{A}^\dagger$

$$\widehat{\boldsymbol{x}} = \boldsymbol{A}^\dagger\boldsymbol{y} = \boldsymbol{V}\boldsymbol{\Sigma}^{-1}\left(\boldsymbol{U}^{\mathrm{T}}\boldsymbol{y}\right) = \sum_{j=1}^{n} \frac{1}{\sigma_j}\left(\boldsymbol{u}_j^{\mathrm{T}}\boldsymbol{y}\right)\boldsymbol{v}_j = \sum_{j=1}^{n} \frac{1}{\sigma_j}c_j\boldsymbol{v}_j$$

The estimated "true" vector $\widehat{\boldsymbol{x}}$ is a superposition of the vectors $\boldsymbol{v}_j$, with "measured" Fourier coefficients $c_j = \left(\boldsymbol{u}_j^{\mathrm{T}}\boldsymbol{y}\right)$, and weighted with the inverse singular values $1/\sigma_j$.

$$\boldsymbol{V}_x = \boldsymbol{A}^\dagger\boldsymbol{V}_y\boldsymbol{A}^{\dagger\mathrm{T}} = \boldsymbol{V}\boldsymbol{\Sigma}^{-2}\boldsymbol{V}^{\mathrm{T}} = \sum_{j=1}^{n}\left(\frac{1}{\sigma_j^2}\right)\boldsymbol{v}_j\boldsymbol{v}_j^{\mathrm{T}} \qquad \text{(assuming } \boldsymbol{V}_y = \boldsymbol{I}\text{)}$$

The estimated "true" vector $\widehat{\boldsymbol{x}}$ and its covariance matrix $\boldsymbol{V}_x$ will be dominated by components with small singular values σ_j.

Properties of the LS solution

The potential bias and the accuracy of a method should be checked.

(1) Check of a potential <u>bias</u> in the solution:

$$\text{estimator} \quad \widehat{\boldsymbol{x}} = \boldsymbol{A}^{\dagger}\boldsymbol{y} \quad \text{with} \quad E\left[\boldsymbol{y}\right] = \boldsymbol{A}\boldsymbol{x}_{\text{exact}}$$
$$E\left[\widehat{\boldsymbol{x}}\right] = \boldsymbol{A}^{\dagger}E\left[\boldsymbol{y}\right] = \left(\boldsymbol{A}^{\dagger}\boldsymbol{A}\right)\boldsymbol{x}_{\text{exact}} = \boldsymbol{x}_{\text{exact}}$$

$\Rightarrow$ the estimator $\widehat{\boldsymbol{x}}$ is unbiased, because: resolution matrix $\boldsymbol{\Xi} \equiv \boldsymbol{A}^{\dagger}\boldsymbol{A} = \boldsymbol{1}$.

(2) <u>Variance:</u> Lower bound of the variance is given by the Rao-Cramér-Frechet (RCF) inequality. The covariance matrix $\boldsymbol{V}_x$ is equal to the lower bound:
$\Rightarrow$ the estimator $\widehat{\boldsymbol{x}}$ has the smallest possible variance for an estimator *with zero bias*.

General statement by the Gauss-Markov theorem: the least square estimate is unbiased and efficient.

But: the result $\widehat{\boldsymbol{x}}$ will often show large, unacceptable fluctuations!

The fluctuations are not caused by inaccurate matrix elements (from Monte Carlo), but are inherent in the problem, i.e. the response matrix $\boldsymbol{A}$ and its "smoothing" properties.

Plot of the singular vectors

Note: The measurement is represented by coefficients c_j, each representing a whole distribution $\boldsymbol{u}_j$, by $c_j = \left(\boldsymbol{u}_j^{\mathrm{T}} \boldsymbol{y}\right)$, with **white noise** (same uncertainty in all frequencies).

The solution

$$\widehat{\boldsymbol{x}} = \sum_{j=1}^{n} \frac{1}{\sigma_j}\, c_j\, \boldsymbol{v}_j$$

is expressed as a superposition of normalized eigenvectors $\boldsymbol{v}_j$, each

- proportional to the Fourier coefficient c_j, and

- weigthted by $1/\sigma_j$, resulting in **blue noise** (uncertainty increasing with frequency).
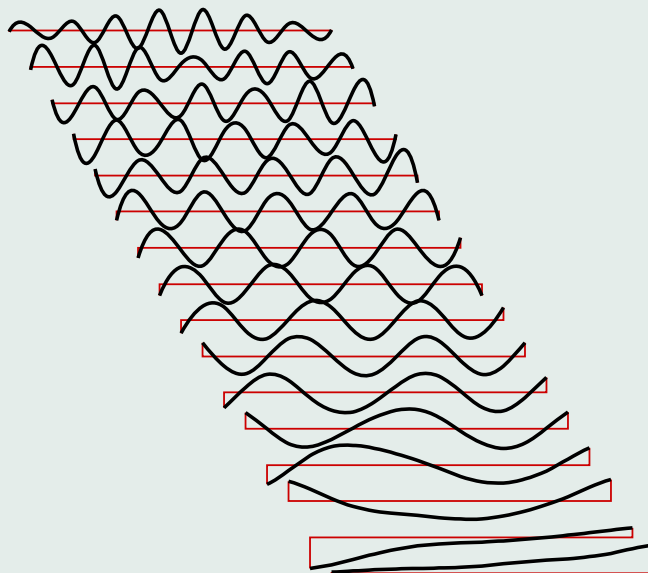
Note: building blocks of the solution are not single bins, but whole distributions $\boldsymbol{v}_j$

Naive result with narrow bins

Example of unfolding problem with narrow bins

Perfect resolution (no smearing)

X = true variable [1]
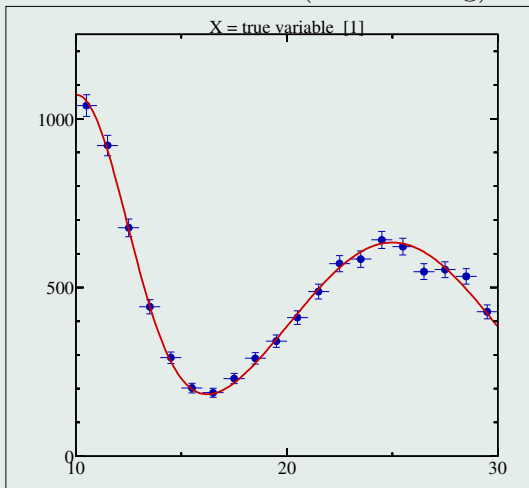
Naive unfolding by inversion

Parameter and cov.matrix (inversion) [0]

Histogram
for sample with 10 000 entries.
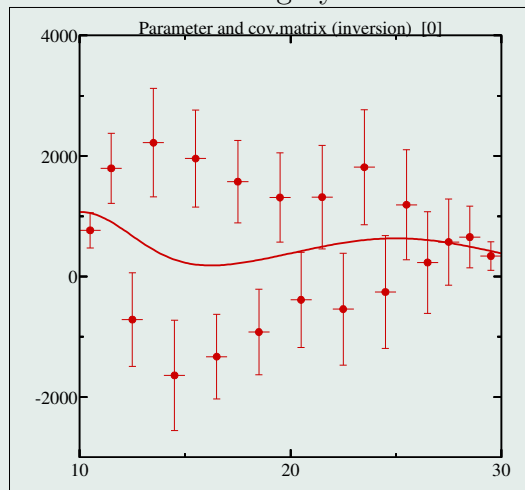
Huge fluctuations, due to large negative correlations: neighbour bin -95% (second $+85\%$).

True curve $f(x)$ is shown in red.

Example with truncation

Example of unfolding problem with narrow bins
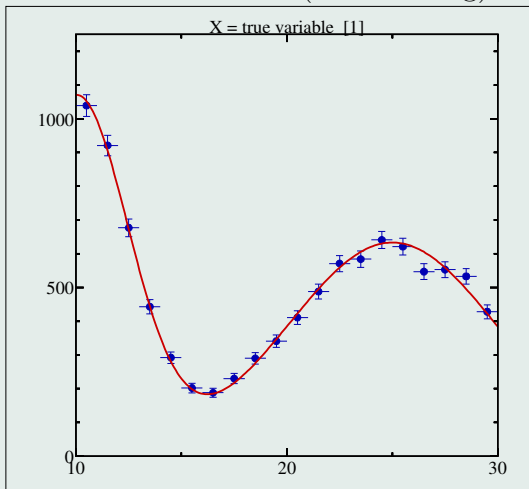
Perfect resolution (no smearing)

X = true variable [1]

Truncation method (15 terms kept)

Parameter Cov.matrix [0]

Histogram
for sample with 10 000 entries.
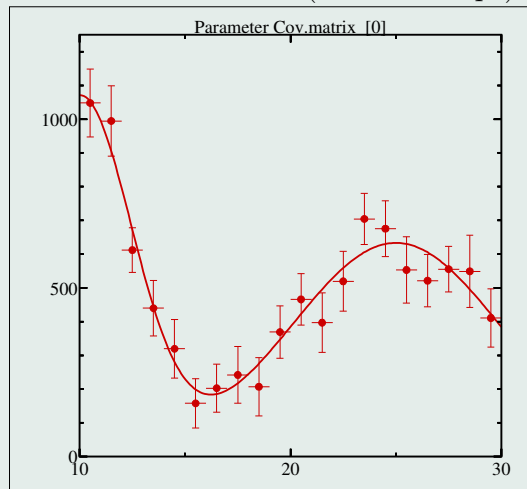
Reduced fluctuations

True curve $f(x)$ is shown in red.

Correlation matrices – Examples

$$\text{correlation coefficient } \rho_{ij} = \frac{V_{ij}}{\sqrt{V_{ii}V_{jj}}} = \frac{V_{ij}}{\sigma_i \times \sigma_j}$$

Colour graph of correlation coefficients ρ with range $-1 \ldots +1$:

Direct inversion

Parameter and cov.matrix (inversion) [7]
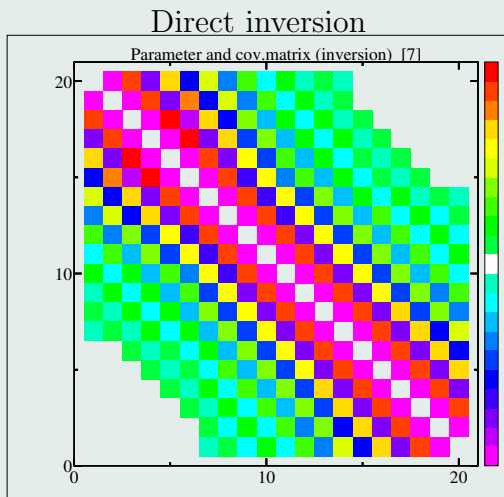
Truncation method (15 terms of 20 kept)

Parameter Cov.matrix [15]
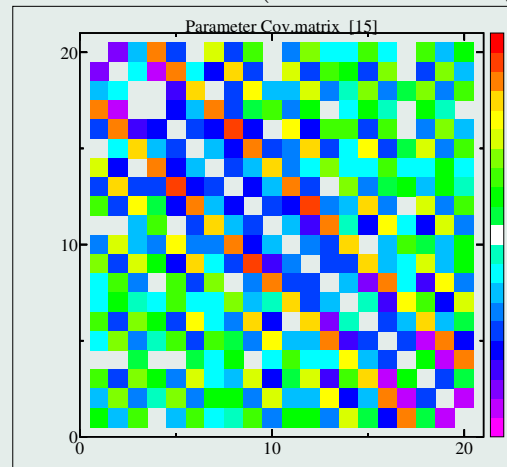
Large negative and positive correlations: neighbour bin -95%, second neighbour $+85\%$.

Correlations reduced, negative for neighbour bin -40% and second neighbour -30%.

Strategies

Naive unfolding results will be wildly fluctuating and are not acceptable.

Wider bins – reduce number n of bins of estimated "true" vector $\widehat{\boldsymbol{x}}$; this will avoid very small singular values σ_j.

Cut/truncation – ignore insignificant Fourier coefficients c_j (value $\approx$ uncertainty) with small values σ_j:

$$\widehat{\boldsymbol{x}} = \boldsymbol{A}^{\dagger}\boldsymbol{y} = \sum_{j=1}^{n} \frac{1}{\sigma_j} c_j \boldsymbol{v}_j \qquad \Rightarrow \qquad \sum_{j=1}^{n_0} \frac{1}{\sigma_j} c_j \boldsymbol{v}_j \quad \text{with} \quad n_0 < n$$

A sharp cut-off may result in Gibbs oscillations.

Regularization – add e.g. "smoothing" term $\|\boldsymbol{Lx}\|^2$ to least squares condition:

$$\underset{\boldsymbol{a}}{\text{minimize}}\, F_\tau(\boldsymbol{x}) = \|\boldsymbol{Ax} - \boldsymbol{y}\|^2 + \tau \|\boldsymbol{Lx}\|^2 = \text{minimum}$$

Result is a smooth cut-off, that avoids Gibbs oscillations.

Parametrized unfolding – if a well-known parametrization $f(t; \boldsymbol{a})$ has to be tested, this parametrization can be used directly, without the need for regularization.

Parametrized unfolding

The parametrization $f(t) \equiv f(t; \boldsymbol{a})$ can be directly used in unfolding, using the response matrix $\boldsymbol{A}$, without the need to introduce a regularization.

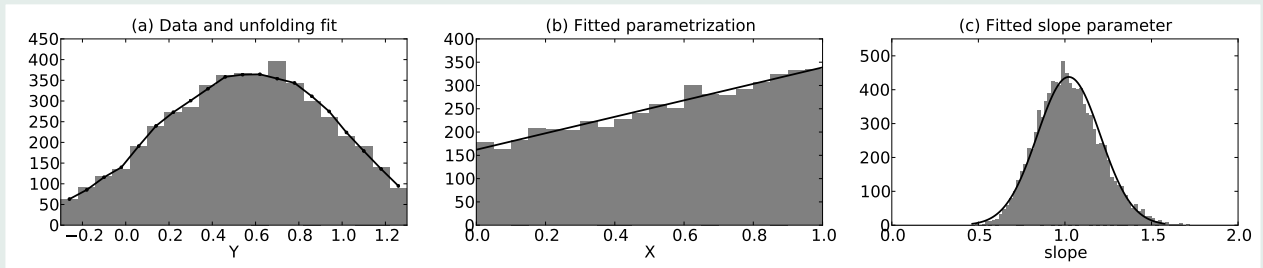Example: Gaussian resolution function with a std. dev. of 0.3 assumed, and

$$f(t; a) = \frac{(1 + a\,t)}{(1 + a/2)} \quad \text{with } a = 1 \qquad t \in [0, 1] \qquad s \in [-0.3, +1.3]$$

The bin content y_i is approximated using the elements of an auxiliary vector $\boldsymbol{x}$:

$$y_i = \int_{s_{i-1}}^{s_i} \mathrm{d}s\; g(s) = \boldsymbol{A}_i^{\mathrm{T}} \boldsymbol{x} \quad \text{with} \quad x_j(\boldsymbol{a}) = \int_{t_{j-1}}^{t_j} \mathrm{d}t\; f(t; \boldsymbol{a}) \qquad j = 1, 2, \ldots, n$$

$$\text{Unfolding:} \qquad \underset{\boldsymbol{a}}{\text{minimize }} F(\boldsymbol{a}) = (\boldsymbol{A}\boldsymbol{x}(\boldsymbol{a}) - \boldsymbol{y})^{\mathrm{T}} \boldsymbol{V}_y^{-1} (\boldsymbol{A}\boldsymbol{x}(\boldsymbol{a}) - \boldsymbol{y})$$

The result of the parameter fit in a single example (left, middle) is $\hat{a} = 1.09 \pm 0.18$.

(a) Data and unfolding fit   (b) Fitted parametrization   (c) Fitted slope parameter

From 10^5 simulated reconstructions (right): fitted slope a has uncertainty of ≈ 0.18.

3. Regularization methods

Key idea: incorporate certain a-priori assumptions about the solution, e.g. smoothness
$\Rightarrow$ control the *norm of the residuals* and, simultaneously, the *norm of the solution* $\boldsymbol{x}$.

$$\textit{Thikhonov-Phillips:} \quad F_\tau(\boldsymbol{x}) = \left\| \boldsymbol{Ax} - \boldsymbol{y} \right\|^2 + \tau \left\| \boldsymbol{Lx} \right\|^2 = \text{minimum} \qquad \tau > 0$$

$$\underset{\boldsymbol{x}}{\text{minimize}} \left\{ \left\| \boldsymbol{Ax} - \boldsymbol{y} \right\|^2 + \tau \left\| \boldsymbol{Lx} \right\|^2 \right\} \implies \boldsymbol{A}^\dagger = \left[\left(\boldsymbol{A}^{\mathrm{T}} \boldsymbol{A} + \tau \boldsymbol{L}^{\mathrm{T}} \boldsymbol{L} \right)^{-1} \boldsymbol{A}^{\mathrm{T}} \right]$$

Norm-regularization with $\boldsymbol{L} = \boldsymbol{1}$, or (better) based on derivatives, e.g.

$$\text{most popular: sec.der.} \quad \boldsymbol{L}_2^r = \begin{pmatrix} 1 & -1 & 0 & 0 & \cdots & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & \cdots & 0 & -1 & 1 \end{pmatrix} = \boldsymbol{U}_{\mathrm{DCT}} \, \boldsymbol{\Lambda} \, \boldsymbol{U}_{\mathrm{DCT}}^{\mathrm{T}}$$

(second derivatives $\approx -x_{i-1} + 2x_i - x_{i+1}$, and first derivative for the first and the last bin.)

- Regularization will not introduce unwanted bias, but allows to reduce or suppress insignificant contributions (noise), that would destroy the unfolding result.
- Regularization strength depends on the regularisation parameter τ.

Solution with regularization

Orthogonalization is mathematically more complicated because there are two terms, the (1) squared residuals sum, and (2) the regularization term with the regularisation parameter τ. Math: generalized singular value decomposition GSVD or double diagonalization.

$$\text{Solution:} \qquad \widehat{\boldsymbol{x}} = \sum_{j=1}^{n} \frac{\varphi_j}{\sigma_j} \, c_j \, \boldsymbol{v}_j \qquad \text{with} \quad \text{Fourier coefficients } c_j$$

filter factor $\quad \varphi_j = \dfrac{\sigma_j^2}{\sigma_j^2 + \tau} \quad \text{or} \quad = \dfrac{\lambda}{\lambda + \tau}$

$$= 1 \; \ldots \; 0.5 \; \ldots \; 0$$

$$\varphi_k = \quad 0.5 \qquad \text{for} \quad \tau = \sigma_k^2 = \lambda_k$$

filter factor dependence

- truncation
- regularization
- $\alpha = 2$
- low-pass filter

filter factor φ_j — index j
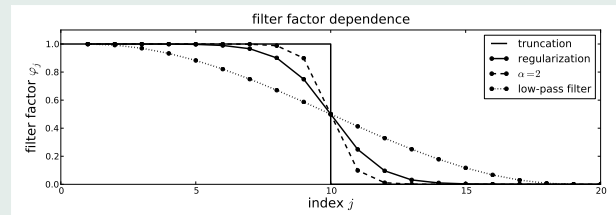
The value of the regularization parameter τ can be determined according to the significance of the Fourier coefficients.

The result $\widehat{\boldsymbol{x}}$ is expressed as a superposition of eigenvectors $\boldsymbol{v}_j$, each weighted with the Fourier coefficient c_j and $1/\sigma_j$; the filter factor φ_j reduces the effect of insignificant contributions, without introducing a bias.

Eigenvalue spectrum and Fourier coefficients

Eigenvalues $= \lambda_j$
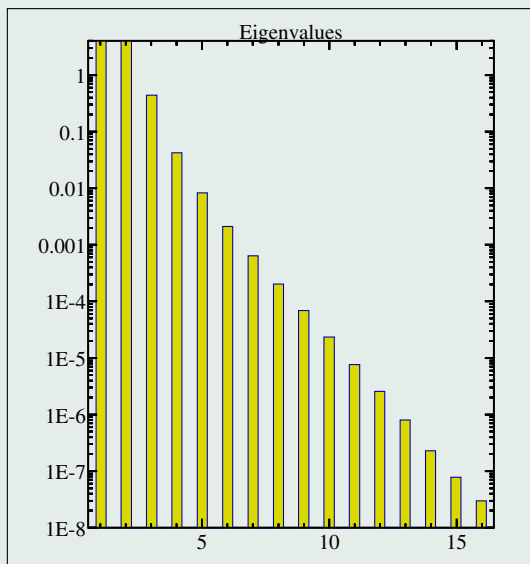
The eigenvalues decrease by ≈ 8 orders of magnitude, due to limited resolution.

Fourier coefficients $= c_j$
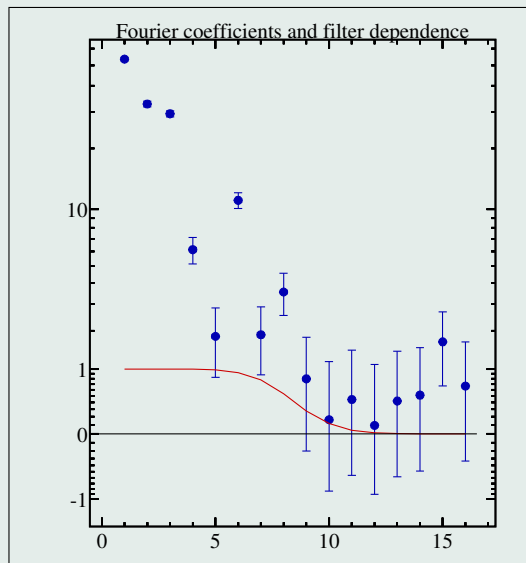
Note: the c_j are independent, and all have uncertainty $1 \Rightarrow$ only ten coefficients are significant.

Regularization parameter I

Discrepancy principle (Morozov): choose τ such that $\|\boldsymbol{Ax}_{\text{reg}} - \boldsymbol{y}\|_2^2 = n_{df}$

L-curve method: plot $\|\boldsymbol{Lx}_{\text{reg}}\|_2$ versus $\|\boldsymbol{Ax}_{\text{reg}} - \boldsymbol{y}\|_2$ for a set of τ-values $\Rightarrow$ dependence has shape of an L with a distinct corner: optimal value of τ.

Effective number of degrees of freedom: (RUN) determine n_{df} from spectrum of Fourier coefficients c_j, and determine τ such that sum of filter factors

$$\sum_{j=1}^{n} \frac{\lambda_j}{\lambda_j + \tau} = n_{df}$$

Minimum of global correlation: minimum mean value of global correlation coefficients

$$\text{Definition:} \quad \rho_j = \sqrt{1 - \left[(\boldsymbol{V}_x)_{jj} \cdot (\boldsymbol{V}_x^{-1})_{jj} \right]^{-1}} \quad \text{with} \quad 0 \le \rho_j \le 1$$

The global correlation coefficient is a measure of the total amount of correlation between element j of $\boldsymbol{x}$ and all other elements.
The arithmetic and the geometric mean of all n global correlation coefficients is determined for a large range of τ-values: the τ-value with the smallest mean value is accepted.

... seems to be a good method, result similar to L-curve method!

Regularization parameter II

L-curve

L curve

weak regularization

negative correlation

positive correlation

strong regularization

log10 of curvature

log10 of chi^2

Dependence as a function of τ

exponent n of tau = 10^n
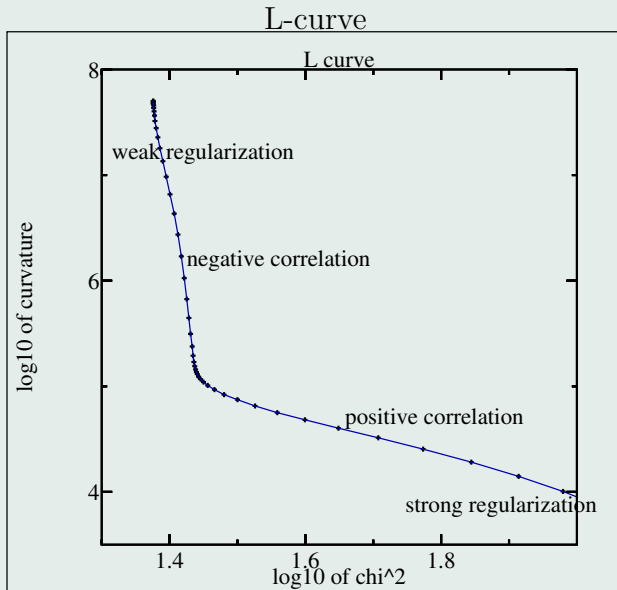
optimal regularization in region of largest curvature
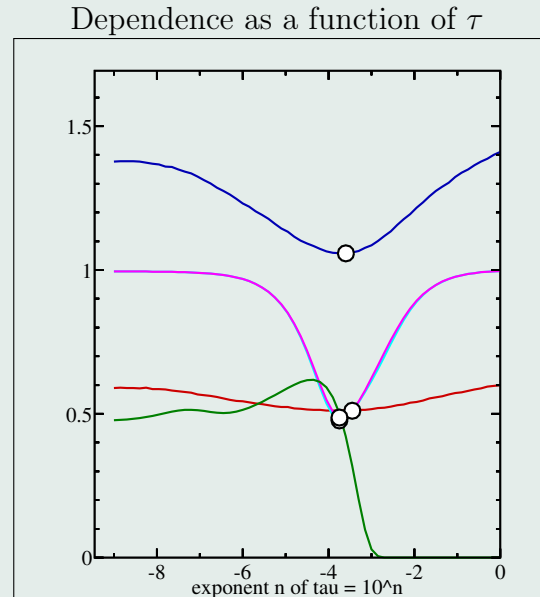
circle=minimum

magenta/cyan=mean global correlation
green=χ^2 probability
red=average probability
blue=relative sigma

Unfolding of inclusive jet cross section as a function of p_T:

- p_T, measured by calorimeter, systematically underestimated (bias);

- steeply falling: transformation to $\sqrt{p_T} \Rightarrow$ variable bin size and constant std. deviation;

- region of small p_T unmeasurable: uncertainties increase for region $p_T \to 0$

- unfolding in one step allows the consistent determination of the covariance matrix.

Fourier coefficients c_j,
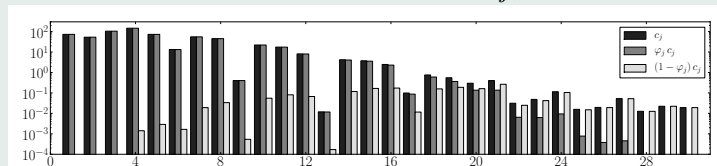
filtered coefficients, and difference

jet transverse momentum p_T distribution
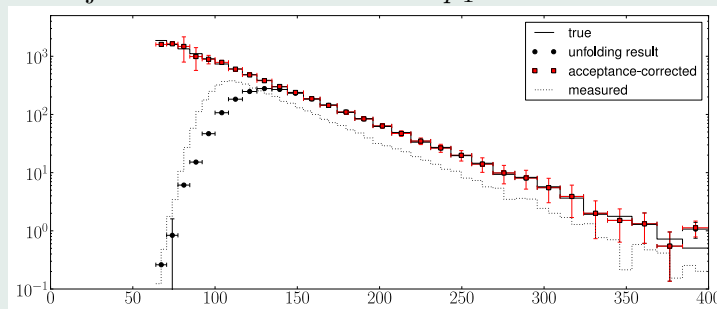
MC simulation

Correction for (1) bias, and for (2) limited resolution (smearing) using bin-by-bin CF method, done in two separate steps in a FNAL-publication.

Particle physics

Event data multi-dimensional

$$n \approx 20, \ m \approx 50, \quad n \times m \approx 1000$$

- Small matrix dimensions allow analytical solution with complex matrix operations: $\widehat{\boldsymbol{x}} = \boldsymbol{A}^{\dagger} \boldsymbol{y}$;
- orthogonalization by e.g. SVD or diagonalization;
- regularization with (e.g. derivative) matrix $\boldsymbol{L}$;
- covariance matrix by propagation of uncertainties: $\boldsymbol{V}_x = \boldsymbol{A}^{\dagger} \boldsymbol{V}_y \boldsymbol{A}^{\dagger \mathrm{T}}$

Picture deblurring (HST)

Picture data 2-dimensional

$$n = m = 1000 \times 1000 = 10^6 \quad n^2 = 10^{12}$$

- Only <u>iterative solutions</u> possible;
- regularization implicit (stop early);
- e.g. Richardson-Lucy (1972, 1974);
- matrix often sparse (point spread function PSF);
- only matrix-vector products;
- no covariance matrix meaningful and technically impossible.

complex matrix operations: space requirement $\propto n^2$ and cpu time requirement $\propto n^3$

Richardson-Lucy algorithm

The Richardson-Lucy (RL) algorithm has been used for the deconvolution of images and spectra from the Hubble space telescope (HST). The derivation of the formula for one iteration step

$$x_j^{[k+1]} := \frac{x_j^{[k]}}{\epsilon_j} \sum_{i=1}^{m} \frac{y_i}{c_i} A_{ij} \qquad \text{with } c_i = \sum_{j=1}^{n} A_{ij} x_j^{[k]} \quad \text{and } \epsilon_j = \sum_{i=1}^{m} A_{ij}$$

make use of *Bayes theorem*.

Richardson, W.H., Bayesian-based iterative method for image restoration", Journal of the Optical Society of America, vol 62, 55-59 (1972)   Lucy, L.B., An iterative technique for the rectification of observed distribution, Astronomical Journal, vol 79, 745-754 (1974)

Empirically the method converges to the maximum-likelihood solution for Poisson distributed data.

Available in PC-Photo-software (Raw Therapy).

Another iterative method is the Landweber iteration, which converges to the least squares solution.

A method identical to Lucy–Richardson convolution is used in particle physics:

G. DAgostini, A multidimensional unfolding method based on Bayes theorem, NIM A362 (1995) 487 (used in several large collaborations).

Soft migration effects

Indirect measurement

Experiment with
- accurate measurement;
- standard deviation $<$ bin width;
- no bias, only migration (and limited acceptance);
- "high" purity and stability in the bins.

- No direct measurement possible,
- but some measured distribution(s) in a certain sense "correlated" with physics variable of interest;
- i.e. almost *no* measurement.

Question: Is unfolding necessary?

Question: Is unfolding possible?
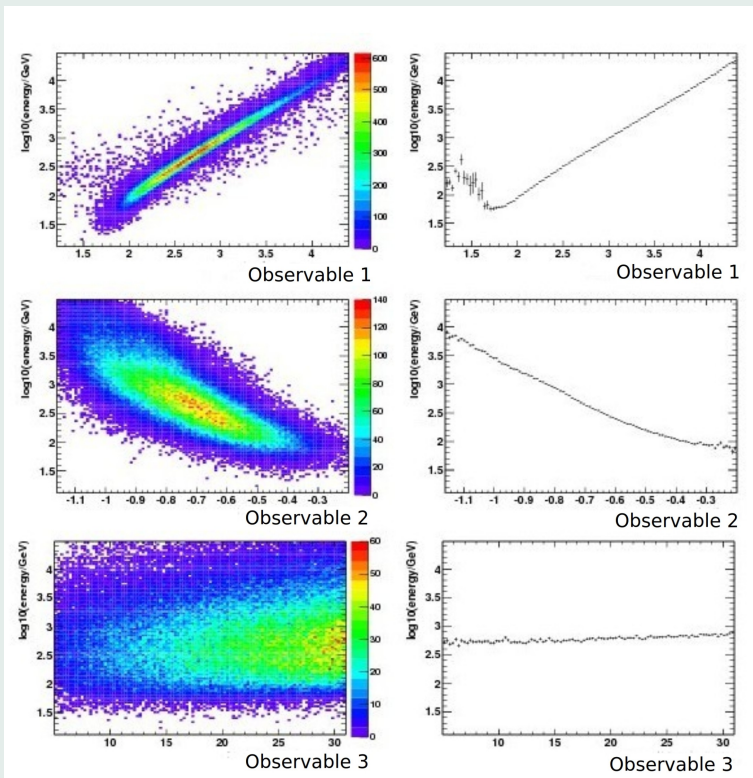
Measurement: flux of cosmic gamma-rays, up to tens of TeV, by system of two Cherenkov telescopes:

- Observable 1: energy estimate – good correlation with energy;

- Observable 2: light distribution parameter, has some correlation with energy;

- Observable 3: angle, no direct correlation with energy, but energy estimates differ for different angles.

3-dim distribution of observables used to unfold (reconstruct) (1-dim) energy flux. (RUN/TRUEE)

Plots from a MC study.

If there is no migration between bins:

$$(\text{cross section})_i \equiv x_i = \frac{n_{i,\text{cand}} - \tau \cdot n_{i,\text{bg}}}{A_{ii} \left[\int \mathcal{L} \, dt \right]} \implies \frac{n_{i,\text{cand}} - \tau \cdot n_{i,\text{bg}}}{n_{\text{rec}}^{MC}/n_{\text{gen}}^{MC} \left[\int \mathcal{L} \, dt \right]}$$

using *identical bins* for the true and measured variable. The correction factor $n_{\text{gen}}^{MC}/n_{\text{rec}}^{MC} = (S/P)^{-1}$ is the inverse acceptance probability of the bin.

This method, also called bin-by-bin correction, is used in many experiments
<u>even if there are migration effects</u> between bins, fixing the migration to the input assumption (MC).
Purity P and stability S determined from an adjusted/optimized Monte Carlo event sample

$$\text{Purity} \quad P = \frac{n_{\text{rec,gen}}^{MC}}{n_{\text{rec}}^{MC}} \qquad \text{Stability} \quad S = \frac{n_{\text{rec,gen}}^{MC}}{n_{\text{gen}}^{MC}} \qquad \left(\frac{n_{\text{rec}}^{MC}}{n_{\text{gen}}^{MC}} = \frac{S}{P} \right)$$

- Very **popular** in HEP as "approximate" method;
- a method without any matrix (operation), without the use of data covariance matrix $\boldsymbol{V}_y$; <u>calculation of covariance matrix $\boldsymbol{V}_x$ undefined</u>;
- migration out of bin and into bin **fixed** by Monte Carlo input assumption;
- Correction factors – a disaster. ... The data will tend to follow the MC that gave you the correction factors ... (Roger Barlow, SLUO Lecture 9 (2000) SLAC

...means: only migration (and acceptance loss), but no shift (bias); <u>high stability</u> = large fraction of events remaining in same bin is required.

Uncertainties of corrected bin content often determined by $\sqrt{\text{bin content}}$, ignoring migration/correlation effects.
Question: Is the CF method and its uncertainty determination statistically acceptable?

Example: for stability of 90 %, i.e. 90 % of events remain in same bin, the uncertainties increase by 12 % on average; the correlation coefficient to the next bin is only $\rho = -11\%$. This corresponds to a measurement with a standard deviation of $0.3 \times$ binwidth.

| standard deviation | bin contents | | | | | error magnification | | corr. coeff. |
	$\Leftarrow\Leftarrow$	$\Leftarrow$	central	$\Rightarrow$	$\Rightarrow\Rightarrow$	largest	average	to next bin
$0.30\times$ binwidth	0 %	5 %	90 %	5%	0 %	$\times 1.24$	$\times 1.12$	$\rho = -11\%$
$0.39\times$ binwidth	0 %	10 %	80 %	10%	0 %	$\times 1.64$	$\times 1.31$	$\rho = -25\%$
$0.74\times$ binwidth	5 %	20 %	**50 %**	20%	5 %	$\times\mathbf{5.00}$	$\times\mathbf{3.24}$	$\rho = \mathbf{-61\%}$
$1.00\times$ binwidth			38 %				$\times 7.71$	

"The purity and stability typically exceed 50 %. If either the purity or the stability is below 25 % in a bin for the chosen reconstruction method, the bin is combined with an adjacent bin."(from a Publication)

Iterative methods in particle physics

In general there is the attempt (by iterative tuning) to use the "correct" distribution in the MC simulation, i.e. that distribution that should be extracted from the measured data $\boldsymbol{y} \simeq \boldsymbol{A}\boldsymbol{x}$. Iterative improvement of a <u>matrix $\boldsymbol{M}_x^{[k]}$</u>

$$\boldsymbol{x}^{[k+1]} = \boldsymbol{M}_x^{[k]}\,\boldsymbol{y} \quad \Longrightarrow \quad \boldsymbol{x}^{[k+1]} = \left(\boldsymbol{M}_x^{[k]}\,\boldsymbol{A}\right)\boldsymbol{x} \qquad \text{but} \quad \left(\boldsymbol{M}_x^{[k]}\,\boldsymbol{A}\right) \neq \boldsymbol{I}$$

"equation" valid only for one solution $\boldsymbol{x}$

bin-by-bin correction factor:	$\boldsymbol{M}_x =$ diagonal, positive elements
other iterative methods:	$\boldsymbol{M}_x =$ matrix with non-negative elements

Disadvantage: <u>convergence speed depends on spectrum of singular values</u> (extremely slow for small singular values) $\Rightarrow$ unknown regularization strength; no direct error propagation with matrix $\boldsymbol{M}_x$ possible; correlations unknown, and ignored in bin-by-bin correction method; questionable statistically and mathematically; applicable only to "correct" for migration effects, no general unfolding with e.g bias correction in CF method.

Advantage: popular and accepted by collaborations; simple mathematics: no "complicated" operations like SVD; easy to use.

5. Results and correlations

(from A. Hoecker and V. Kartvelishvili: NIM A 372)

- Result with 40 data points, almost like a "band" representing result

- constructed from 10 significant parameters, with 40-by-40 covariance matrix, singular with rank 10;

- large positive correlations, therefore few sign-changes of the residuals to MC input distribution.

- (Same) result with 10 data points, each point represents a bin average of the result

- 10-by-10 covariance matrix non-singular, inverse is weight matrix.

- small and negligible correlations.
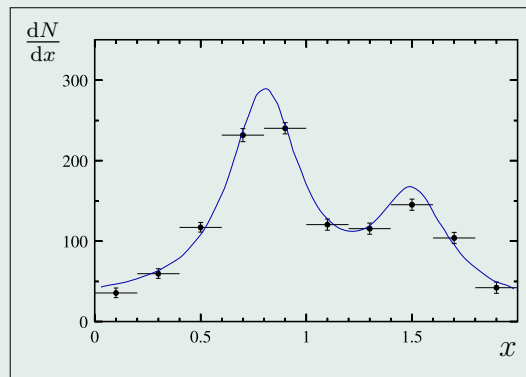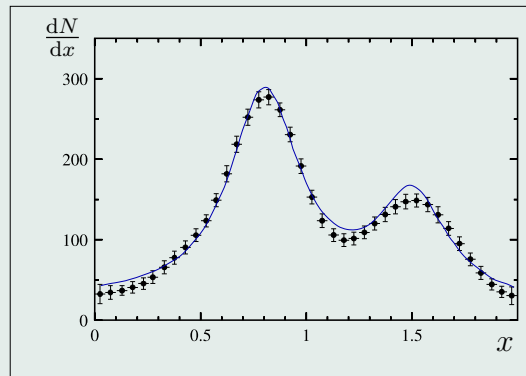
...the world of correlations

Average of two *correlated* numbers d_1 and d_2 (assuming $\sigma_1 = \sigma_2$) with positive/negative correlation:

$$\text{average} \quad \overline{\boldsymbol{d}} = \tfrac{1}{2}(d_1 + d_2)$$
$$\boldsymbol{V}_{\overline{d}} = \tfrac{1}{2}(1 + \rho_{12})\,\sigma^2$$

$$\boldsymbol{V} = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

$$\boxed{\rho_{12} = +0.95}$$
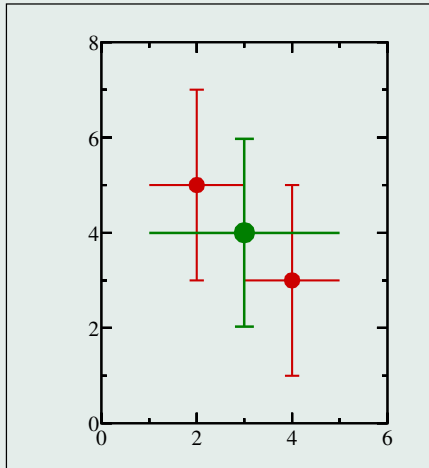
$$\boxed{\rho_{12} = -0.95}$$

Averaged value has almost the same error as each single data value (0.987σ).
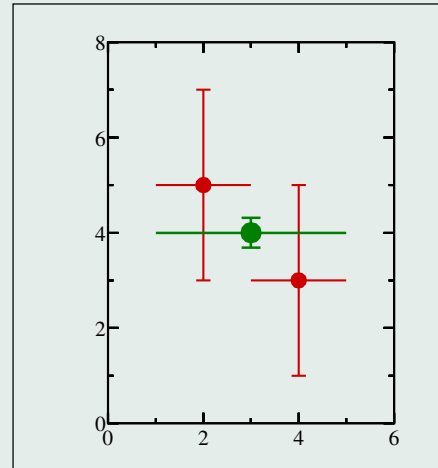
Averaged value has much smaller error than each single data value (0.158σ).

Example for positive correlation

Unfolding of charged multiplicity distribution in ALICE:

Hump observed around $N_{ch} = 30 \pm 5$ $\Rightarrow$ new physics?

Residuals of measured distribution

Charged Multiplicity distribution
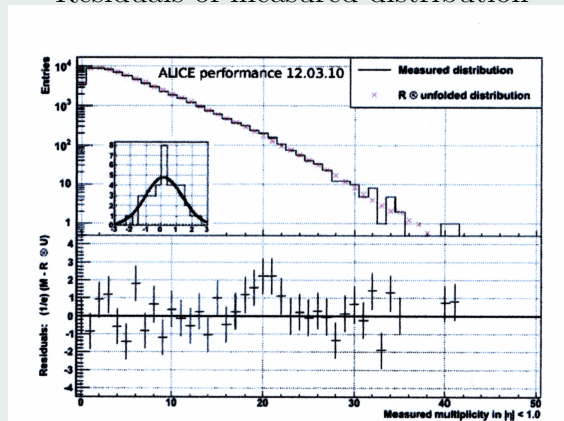
Small upward fluctuation observed
in ≈ 4 bins around $N_{ch} = 20$.

Bins of the unfolded distribution are (positively) correlated over a large range. The origin of the 10-bins hump at $N_{ch} = 30$ is a 4-bin fluctuation at observed $N_{ch} = 20$.

Note: $n = 40$ bins are unfolded here from $m = 30$ measured bins – covariance matrix must have rank defect > 10.

From: Jan Fiete Grosse-Oetringhaus: Comments on Unfolding Methods in ALICE, PHYSTAT 2011

RUN/TRUEE: RUN (Fortran $\approx$ 1980) by V.B., converted to TRUEE in C++ by Natalie Milke (Uni Dortmund). Diagonalization. TRUEE manual and code available from `http://app.tu-dortmund.de/TRUEE/`

GURU Fortran ($\approx$ 1995), by Andreas Hoecker and Vato Kartvelishveli, rewritten in C++. SVD

TUnfold C++ by Stefan Schmitt, L-curve scan, TUnfoldSys (systematic error propagation, unfolding with background subtraction)

`https://www.wiki.terascale.de/index.php/Statistics_Projects`
RooUnfold = ROOT Unfolding Framework, by Tim Adye et al. (includes bin-by-bin, inversion, iterative methods); partly in Root distribution.

Method	RUN/TRUEE	GURU	Tunfold	Iterative
Input: matrix		✔	✔	✔
n-tuple	✔			
discretization	B-Splines	hist	hist	hist
Least squares		✔	✔	Landweber
MaxLik (Poisson)	✔			R-L (D'Ago.)
Orthogonalization	✔	✔		
Cov.mat. by prop.	✔	✔	✔	
Dimension in/out	3/1	1/1	multi	multi
Regularization	✔	✔	✔	**implicit**
automatic binning	✔			
MC re-weighting	✔			

7. Future developments

Optimal unfolding by mouse click?   Not yet!

Unfolding software requires several "unfolding parameters":
      e.g. bin sizes, regularization parameter, ...

- Develop algorithms to determine optimal "unfolding parameters" automatically;
- repeat unfolding for whole matrix of "unfolding parameters", and select set optimal parameter set. Which optimality criterium?   (option included in TRUEE)
- develop algorithms which avoid "unfolding parameters";
- introduce a common interface for several programs
  (Unfolding Framework Project in Terascale).

Unfolding without histograms

Discretization of noisy measurement $\boldsymbol{y}$ and of noisy response matrix $\boldsymbol{A}$ is necessary.

Histograms are not optimal for numerical applications; they
- are discontinuous, and require fixed binning,

- have too much local variability (bins too small), or

- have low resolution (bins too large).

Remember fundamental formula for SVD solution:

$$\text{solution} \qquad \widehat{\boldsymbol{x}} = \sum_{j=1}^{n} \frac{\varphi_j}{\sigma_j} \, c_j \, \boldsymbol{v}_j$$
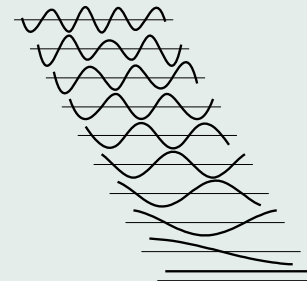
$$u_j(s) = \cos\left(\pi j s\right) \quad j = 0, \, 1, \ldots$$

The measurement $\boldsymbol{y}$ is represented by coefficients $c_j = \boldsymbol{u}_j^{\mathrm{T}} \boldsymbol{y}$ or $\boldsymbol{c} = \boldsymbol{U}^{\mathrm{T}} \boldsymbol{y}$ $\qquad \boldsymbol{y}$ = histogram.
Alternative estimate of pdf: orthogonal series estimator from sample $s_1, \ldots s_N$:

$$g(s) = \sum_{j=0}^{\infty} c_j u_j(s) \qquad c_j = \int_0^1 g(s) u_j(s)\,\mathrm{d}s \quad \Rightarrow \quad \widehat{c}_j = \frac{1}{N} \sum_{i=1}^{N} u_j(s_i)$$

$$\widehat{\boldsymbol{c}} = \boldsymbol{U}_{\mathrm{DCT}}^{\mathrm{T}} \, \boldsymbol{y} \qquad\qquad \boldsymbol{y} = \text{1024-bin histogram}$$

DCT = Discrete cosine transformation, used in modern coding standards like JPEG, MPEG.

Transformation to Fourier coefficients c

Probability estimate of coefficients from 1024-bin histogram, using special orthogonal series (similar, not identical to DCT), with automatic recognition of necessary number of coefficients, and filtering of high-frequency contributions.
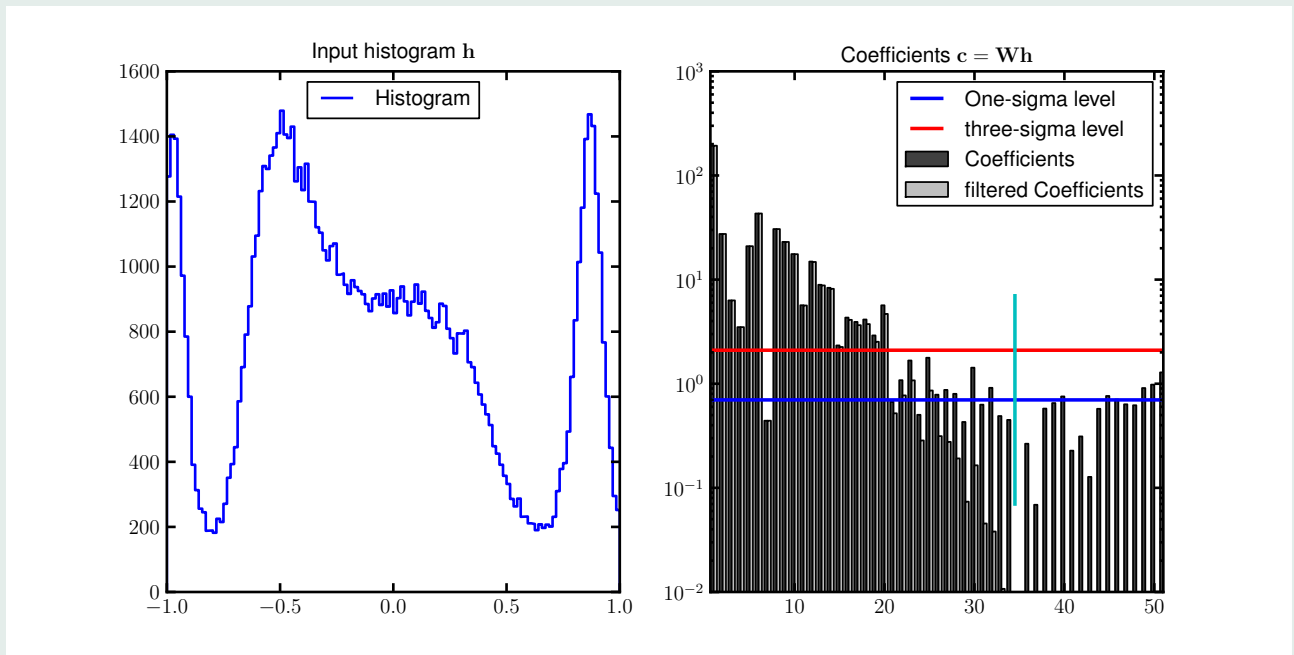
Smoothing by back-transformation

Probability density (red curve), underlying the noisy histogram, is reconstructed from the first 34 terms of an orthogonal series estimation.
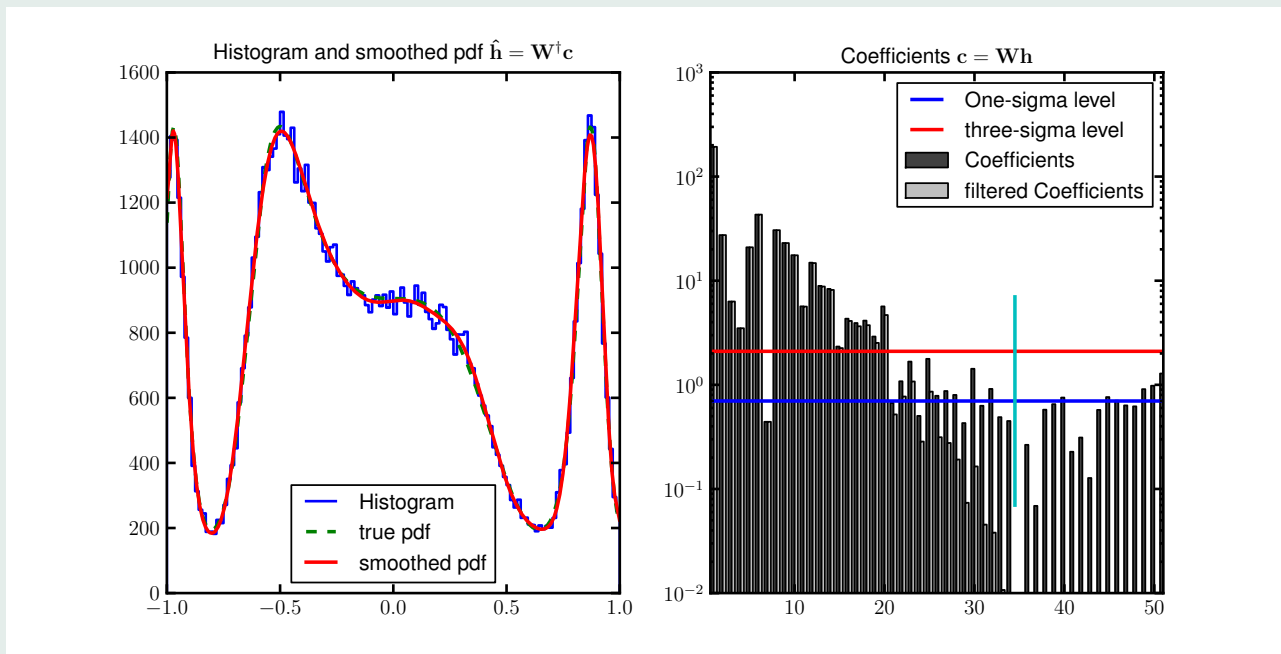
Properties: no bias, no discontinuities, noise and high-frequency components strongly reduced – no user parameters required.

Summary

- Simple bin-by-bin correction factor method is accepted by large collaborations for "soft" migration cases – ignores correlations and tends to follow the MC assumption.
- Regularization methods allow unfolding of difficult data with "indirect" measurements, with orthogonalization – requires knowledge of statistics and mathematics.
- Use of unfolding software is being simplified by attempts to introduce a common interface (Unfolding Framework Project) for existing software.

- Work on improved methods, easy to use, with increased stability and robustness, would be promising.

Appendix

Standard method for the determination of the response matrix $\boldsymbol{A}$ in HEP is Monte Carlo simulation. Input to the simulation is a certain "true" distribution $f(x)$, in the form of a histogram of event numbers $\{\mathcal{N}_1, \mathcal{N}_2 \ldots \mathcal{N}_n\}$. In the simulation an event from bin j is generated, simulated in the detector and reconstructed/observed in bin i.

$$\mathcal{N}_j = \text{number of events, generated in bin } j = 1 \ldots n$$
$$N_{ij} = \text{number of events, observed in bin } i = 1 \ldots m, \text{ generated in bin } j$$
$$N_{0j} = \text{number of events, not observed, generated in bin } j$$
$$A_{ij} = \frac{N_{ij}}{\mathcal{N}_j} = \underline{\text{probability}} \text{ to observe in bin } i, \text{ if generated in bin } j$$

Equation $\boldsymbol{Ax = y + e}$ (Measured histogram $\boldsymbol{y}$ is m-vector, result histogram $\boldsymbol{x}$ is n-vector, $m \geq n$)

$$A_{11}x_1 + A_{12}x_2 + A_{13}x_3 + \ldots A_{1n}x_n = y_1 + e_1$$
$$A_{21}x_1 + A_{22}x_2 + A_{23}x_3 + \ldots A_{2n}x_n = y_2 + e_2$$
$$A_{31}x_1 + A_{32}x_2 + A_{33}x_3 + \ldots A_{3n}x_n = y_3 + e_3$$
$$\ldots$$
$$A_{m1}x_1 + A_{m2}x_2 + A_{m3}x_3 + \ldots A_{mn}x_n = y_m + e_m$$

Note that the probabilities A_{ij} do not depend on the MC distribution $\{\mathcal{N}_1, \mathcal{N}_2 \ldots \mathcal{N}_n\}$

Regularized UNfolding by RUN

The program for regularized unfolding:

- Developed in the 1980's and under conditions of the 1980's (with punched cards?);

- input are n-tuple files, with additional quantities for detailed checks after unfolding;

- measured distribution can be > 1-dimensional;

- curvature (square of second derivates) is used for regularization;

- use of cubic splines to represent intermediate result without discontinuities; allows to calculate accurate second derivatives for regularization and finding of optimized bins;

- maximum likelihood fit based on Poisson distribution;

- final result converted to bins (optional with different optimized bin size)

- Test of covariance matrix:

    - Generate large number of sets of random measurements from the n-dimensional normal distribution, using full matrix $\boldsymbol{V}_x$.
    - Calculate χ^2 for each set, ignoring all off-diagonal elements of $\boldsymbol{V}_x$.
    - Convert each χ^2 with n_{df} into the p-value and make histogram of p-values.

Off-diagonal elements can be neglected, if the histogram of p-values is flat.

Now converted to C++.

Contents