Terascale Statistics Tools School – Spring 2010

Tuesday 23 March – Friday 26 March 2010

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. Cross section determination

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

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

Measurement/MC: **true distribution** $\Longrightarrow \boldsymbol{A} \Longrightarrow$ **measured distribution**

direct $\boldsymbol{A}$ = Kernel, response matrix

Unfolding: **true distribution** $\Longleftarrow \boldsymbol{A}^{\#} \Longleftarrow$ **measured distribution**

inverse $\boldsymbol{A}^{\#}$ = generalized inverse

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

The generalized inverse $\boldsymbol{A}^{\#}$ should depend only on the detector properties, it should **not** depend on the expected result; it allows to propagate the input errors to the result. The product $\Xi = \boldsymbol{A}^{\#}\boldsymbol{A}$ is called resolution matrix.

Cross sections are measured by counting events for the specific process:

$$\text{cross section} = \frac{\text{number of events}}{\text{integrated luminosity}} = \frac{n}{\int \mathcal{L}\,\mathrm{d}t}$$

Detector acceptance probabilities and [efficiencies], ϵ_1, ϵ_2 . . . have to be taken into account:

$$\text{cross section} = \frac{n}{\epsilon_1 \cdot \epsilon_2 \cdots \epsilon_k \int \mathcal{L}\,\mathrm{d}t} = \frac{n}{A \int \mathcal{L}\,\mathrm{d}t}$$

where A = total acceptance and $\int \mathcal{L}\,\mathrm{d}t$ = integrated luminosity.

The candidate event sample may contain background, which has to be subtracted:

$$\text{cross section} = \frac{n_{\text{cand}} - \tau \cdot n_{\text{bg}}}{A \int \mathcal{L}\,\mathrm{d}t}$$

For a small number of events this represents the classical problem of observing a signal in the presence of background (limit calculation).

Statistics:

- the number(s) n of events follows the Poisson distribution;
- the total acceptance factor A follows the log-normal distribution (i.e. $\log A$ is Gaussian);
- the factor τ and the integrated luminosity $\left[\int \mathcal{L}\,\mathrm{d}t\right]$ follow a Gaussian distribution.

Cross section in bins

> 1 **channel/bin**

Cross section x_i as a function of a variable is measured in bins (bin index i), in ideal case of perfect resolution:

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

In practice there are migration effects between bins: an event originating from bin j is measured in another bin i, due to limited detector resolution. It becomes impossible to consider one bin without the other bins: $A \to \boldsymbol{A}$ = matrix with elements A_{ij}:

$$\left[\int \mathcal{L}\, \mathrm{d}t\right] \cdot \sum_j A_{ij}\, x_j + \tau \cdot n_{i,\text{bg}} = n_{i,\text{cand}}$$

using matrix formalism:

$$\left[\int \mathcal{L}\, \mathrm{d}t\right] \cdot \boldsymbol{A}\,\boldsymbol{x} + \tau \boldsymbol{n}_{\text{bg}} = \boldsymbol{n}_{\text{cand}}$$

In the following the equations are written without the factor $\left[\int \mathcal{L}\, \mathrm{d}t\right]$ and without background $\tau \boldsymbol{n}_{\text{bg}}$, and the measured histogram is the vector $\boldsymbol{y}$:

$$\boldsymbol{A}\,\boldsymbol{x} = \boldsymbol{y}$$

to be solved for the cross section $\boldsymbol{x}$, given $\boldsymbol{A}$ and $\boldsymbol{y}$.

Basis is Fredholm integral equation of first kind:

$$\int \boldsymbol{A}(x,y)\, f(x)\, \mathrm{d}x = g(y)$$

which is written in discrete form above.

Inverse problems in other fields

Linear inversion: (measurement) $\boldsymbol{y} \Rightarrow \boldsymbol{x}$ (unfolded distribution)

There is an extensive literature about unfolding, which is of great importance in many different fields:

- tomography in medicine, archaeology, biology, geophysics, oceanography, materials science, astrophysics ... using several different physical phenomena including
  X-rays, gamma rays, positron-electron annihilation reactions, nuclear magnetic resonance, ultrasound, electrons, ions ...;
- image restauration, picture deblurring;

Special conditions in HEP:

Statistical precision of measurement $\boldsymbol{y}$ often limited;

Response matrix $\boldsymbol{A}$ defined by MC sample of events;

Covariance matrix $\boldsymbol{V}_x$ of unfolding result $\boldsymbol{x}$ required!

Literature:
Per Christian Hansen, **Rank-Deficient and Discrete Ill-posed Problems**, Siam (1998)
Jari Kaipio and Erkki Somersalo, Statistical and Computational Inverse Problems, Springer (2005)
Curtis R. Vogel, Computational Methods for Inverse Problems, Siam (2002)
Andreas Rieder, Keine Probleme mit Inversen Problemen, Vieweg (2003)

Response/acceptance matrix determination ... by MC simulation

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.

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

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

$$
\begin{aligned}
A_{11}x_1 + A_{12}x_2 + A_{13}x_3 + \ldots A_{1n}x_n &\cong y_1 \\
A_{21}x_1 + A_{22}x_2 + A_{23}x_3 + \ldots A_{2n}x_n &\cong y_2 \\
A_{31}x_1 + A_{32}x_2 + A_{33}x_3 + \ldots A_{3n}x_n &\cong y_3 \\
&\ldots \\
A_{m1}x_1 + A_{m2}x_2 + A_{m3}x_3 + \ldots A_{mn}x_n &\cong y_m
\end{aligned}
$$

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

If there is no migration between bins:

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

using identical bins for the true and measured variable. The correction factor $n^{MC}_{\text{gen}}/n^{MC}_{\text{rec}}$ is the inverse acceptance probability of the bin, if there is no migration between bins.

The same method is used in many experiments even if there are migration effects between bins, i.e. using no matrix, and fixing migration to the input assumption.
Purity P and stability S determined from an adjusted/optimized Monte Carlo event sample

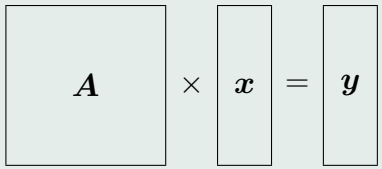
$$\text{Purity} \quad P = \frac{n^{MC}_{\text{rec,gen}}}{n^{MC}_{\text{rec}}} \qquad \text{Stability} \quad S = \frac{n^{MC}_{\text{rec,gen}}}{n^{MC}_{\text{gen}}} \qquad \left(\frac{n^{MC}_{\text{rec}}}{n^{MC}_{\text{gen}}} = \frac{S}{P}\right)$$

- An method without any matrix (operation), without the use of data covariance matrix $\boldsymbol{V}_y$;
- migration out of bin and into bin **fixed** by Monte Carlo input assumption;
- calculation of covariance matrix $\boldsymbol{V}_x$ undefined;
- very **popular** in HEP.
- – for the estimate by CFM it is hard to calculate noise characteristics and bias in the estimate because of nonlinearity of the procedure (V.B.Anykeyev et al.,NIM A 322 (1992) 280-285
- 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

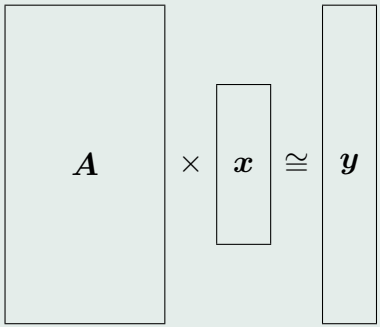
2. Unfolding by matrix inversion

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

Case $n = m$, matrix $\boldsymbol{A}$ is square, non-symmetric



Case $n < m$, matrix $\boldsymbol{A}$ is rectangular



Case of $m = n$ can be solved by matrix inversion, without least squares:

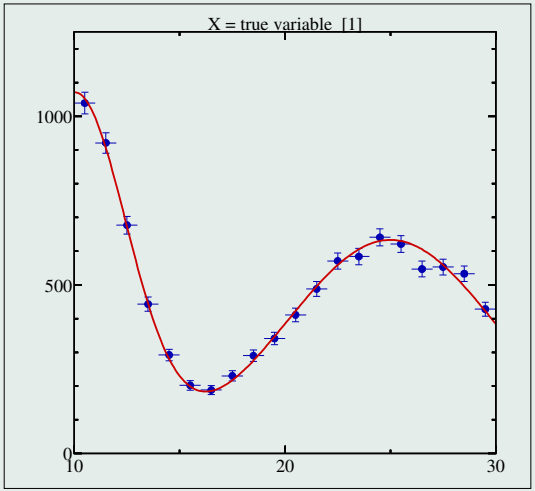
$$\boldsymbol{Ax} = \boldsymbol{y}$$

Direct solution: $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{y}$ $\qquad \boldsymbol{V}_x = \boldsymbol{A}^{-1}\boldsymbol{V}_y \left(\boldsymbol{A}^{-1}\right)^{\mathrm{T}}$

Note: the covariance matrix $\boldsymbol{V}_y$ has no influence on the result $\boldsymbol{x}$; residuals are zero.

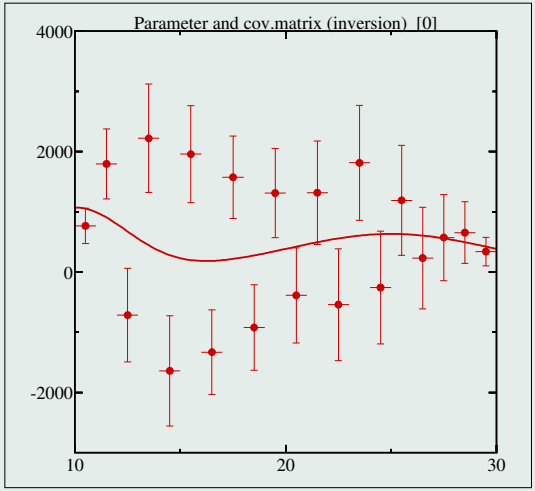
Example of unfolding problem with σ/binwidth $= 3$

Perfect resolution (no unfolding)



Histogram
for sample with 10 000 entries.

Unfolding by inversion



Huge fluctuations, due to large negative correlations: neighbour bin -95%, second $+85\%$.

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

The requirements for the solution of the unfolding equation

$$\boxed{\boldsymbol{A}\boldsymbol{x} \cong \boldsymbol{y}}$$

- Complete usage of all available measurement information, including the (usually diagonal) covariance matrix $\boldsymbol{V}_y$ of measured vector $\boldsymbol{y}$.
- Unfolding should introduce **no bias**, with respect to a particular model of the physical process and Monte Carlo simulation.
- Unfolding should introduce **no or only a small bias**, with respect to general requirements of the solution (a general requirement is e.g. *smoothness* of the solution).
- Complete covariance matrix $\boldsymbol{V}_x$ of solution (and weight matrix $\boldsymbol{W}_x$ = inverse covariance matrix) to be used in subsequent fits of parametrizations to the data.
  For a linear solution $\boldsymbol{x} = \boldsymbol{A}^{\#}\boldsymbol{y}$ by standard error propagation:

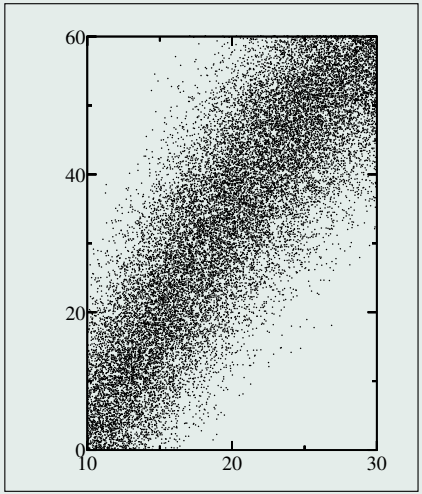
$$\boldsymbol{V}_x = \boldsymbol{A}^{\#}\,\boldsymbol{V}_y\boldsymbol{A}^{\#\mathrm{T}} \ .$$

- Solution should have small correlations between different $\boldsymbol{x}$-bins. Correlation coefficients up to ± 0.5 are acceptable, values $|\pm\rho| > 0.9$ should be avoided, i.e. the matrix $\boldsymbol{V}_x$ should be *almost* diagonal.

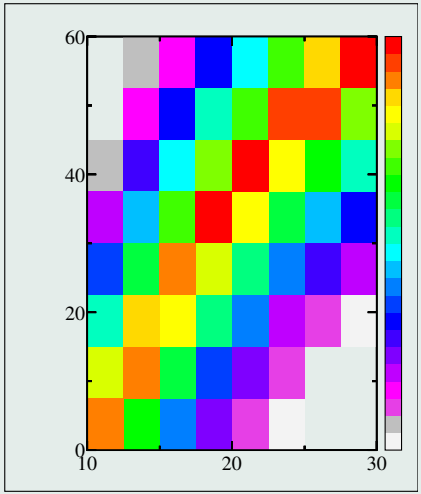
A correct determination of the response matrix $\boldsymbol{A}$ (e.g. by Monte Carlo) is essential.

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

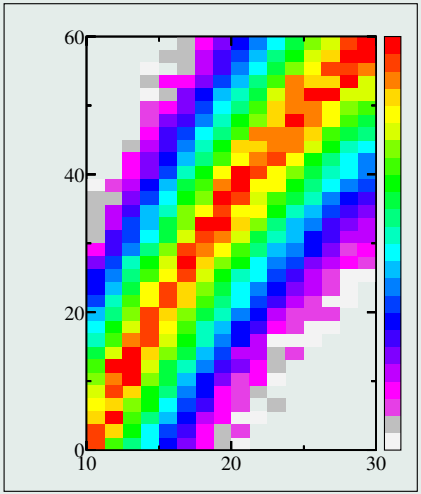
The resolution is deteriorated, if too few bins are used (m too small).



scatter plot x



raw 8×8 bins x



fine 16×32 bins x

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

use number of y-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.)

Problem: Unfolding with 2 bins

Exercise in unfolding and propagation of uncertainties

$$\text{probability matrix} \quad \boldsymbol{A} = \left(\begin{array}{cc} A_{11} & A_{12} \\ A_{21} & A_{22} \\ \hline A_{01} & A_{02} \end{array} \right) = \left(\begin{array}{cc} 0.7 & 0.4 \\ 0.3 & 0.6 \\ 0 & 0 \end{array} \right)$$

Example: A_{12} is probability to observe in bin 1, originating from bin 2.

$$\text{inverse matrix} \quad \boldsymbol{A}^{-1} = \frac{1}{A_{11} \cdot A_{22} - A_{12} \cdot A_{21}} \left(\begin{array}{cc} A_{22} & -A_{12} \\ -A_{21} & A_{11} \end{array} \right) = \left(\begin{array}{cc} ? & ? \\ ? & ? \end{array} \right)$$

$$\text{observed values} \quad \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} = \left(\begin{array}{c} y_1 \\ y_2 \end{array} \right) = \left(\begin{array}{c} 64 \\ 36 \end{array} \right) \qquad \boldsymbol{V} \left(\begin{array}{c} y_1 \\ y_2 \end{array} \right) = \left(\begin{array}{cc} 64 & 0 \\ 0 & 36 \end{array} \right)$$

$$\textbf{calculate} \text{ “true” values} \quad \boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{y} = \left(\begin{array}{c} x_1 \\ x_2 \end{array} \right) = \textbf{?} \qquad \boldsymbol{V} \left(\begin{array}{c} x_1 \\ x_2 \end{array} \right) = \left(\begin{array}{cc} ? & ? \\ ? & ? \end{array} \right)$$

$$\textbf{calculate} \text{ sum and diff} \quad = \left(\begin{array}{c} x_1 + x_2 \\ x_1 - x_2 \end{array} \right) = \textbf{?} \qquad \boldsymbol{V} \left(\begin{array}{c} x_1 + x_2 \\ x_1 - x_2 \end{array} \right) = \left(\begin{array}{cc} ? & ? \\ ? & ? \end{array} \right)$$

Replace ? by numbers!

Solution: Unfolding with 2 bins

Solution for the exercise in unfolding

$$\text{probability matrix} \quad \boldsymbol{A} = \left(\begin{array}{cc} A_{11} & A_{12} \\ A_{21} & A_{22} \\ \hline A_{01} & A_{02} \end{array}\right) = \left(\begin{array}{cc} 0.7 & 0.4 \\ 0.3 & 0.6 \\ \hline 0 & 0 \end{array}\right)$$

$$\text{inverse matrix} \quad \boldsymbol{A}^{-1} = \frac{1}{A_{11} \cdot A_{22} - A_{12} \cdot A_{21}} \begin{pmatrix} A_{22} & -A_{12} \\ -A_{21} & A_{11} \end{pmatrix} = \begin{pmatrix} 2 & -4/3 \\ -1 & 7/3 \end{pmatrix}$$

$$\text{observed values} \quad \boldsymbol{y} = \boldsymbol{A}\boldsymbol{x} = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 64 \pm 8 \\ 36 \pm 6 \end{pmatrix} \qquad \boldsymbol{V} \begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \begin{pmatrix} 64 & 0 \\ 0 & 36 \end{pmatrix}$$

$$\text{“true” values} \quad \boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{y} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 80 \pm 17.9 \\ 20 \pm 16.1 \end{pmatrix} \qquad \boldsymbol{V} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 320 & -240 \\ -240 & 260 \end{pmatrix}$$

$$\text{sum and diff} \quad = \begin{pmatrix} x_1 + x_2 \\ x_1 - x_2 \end{pmatrix} = \begin{pmatrix} 100 \pm 10 \\ 60 \pm 32.6 \end{pmatrix} \qquad \boldsymbol{V} \begin{pmatrix} x_1 + x_2 \\ x_1 - x_2 \end{pmatrix} = \begin{pmatrix} 100 & 60 \\ 60 & 1060 \end{pmatrix}$$

The correlation coefficient between x_1 and x_2 is $\rho = -0.83$,
and between $x_1 + x_2$ and $x_1 - x_2$ it is $\rho = +0.18$.

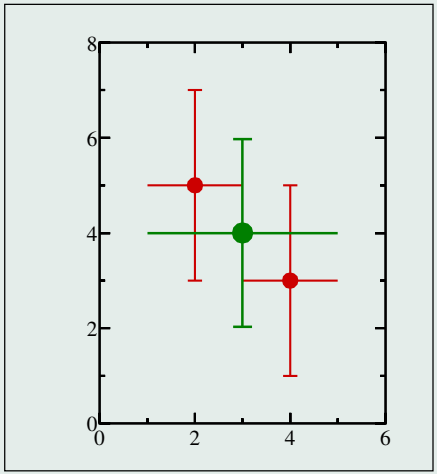
The wonderful world of correlations

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

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

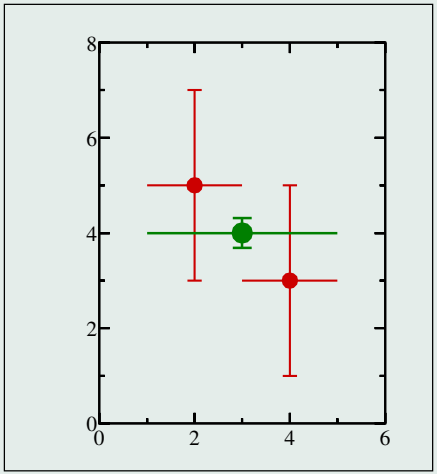
$$\boldsymbol{V} = \begin{pmatrix} \sigma_1^2 & \rho_{12}\sigma_1\sigma_2 \\ \rho_{12}\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

$\rho_{12} = +0.95$



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

$\rho_{12} = -0.95$



Averaged value has much smaller error than each single data value (0.158σ).

3. Least Squares methods

Optimal solution in the least squares sense of the equation $\boldsymbol{A}\boldsymbol{x} \cong \boldsymbol{y}$ is defined by the requirement

$$F(\boldsymbol{x}) = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|^2 = \text{minimum}$$

In the following the covariance matrix $\boldsymbol{V}_y$ is inserted:

$$F(\boldsymbol{x}) = (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y})^{\mathrm{T}}\, \boldsymbol{V}_y^{-1}\, (\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}) = \text{minimum}$$

In the normal-equation formalism for the solution the matrix equation

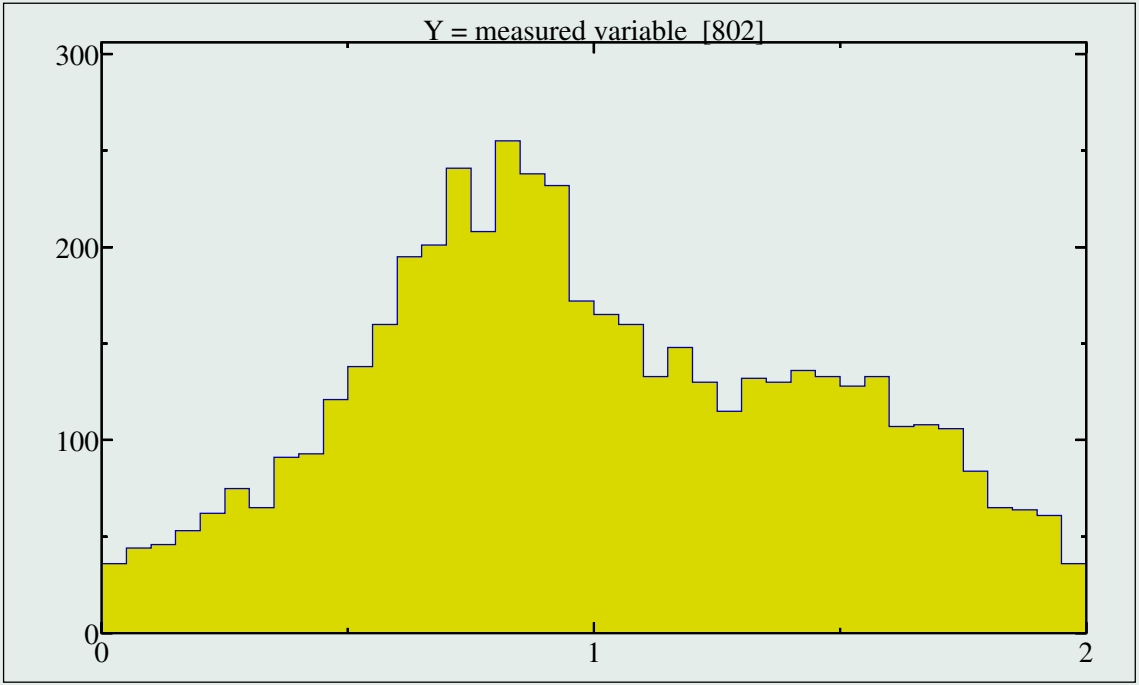
$$\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{A}\right)\boldsymbol{x} = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{y}\right)$$
$$\boldsymbol{C}\boldsymbol{x} = \boldsymbol{b} \qquad \boldsymbol{C} = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{A}\right) \qquad \boldsymbol{b} = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\right)\boldsymbol{y}$$

has to be solved. The solution vector $\boldsymbol{x}$ is a linear transformation of the measurement vector $\boldsymbol{y}$, which allows standard error propagation:

$$\boldsymbol{x} = \boldsymbol{A}^{\#}\boldsymbol{y} \qquad \boldsymbol{A}^{\#} = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{A}\right)^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1} \qquad \boldsymbol{V}_x = \boldsymbol{A}^{\#}\,\boldsymbol{V}_y\boldsymbol{A}^{\#\,\mathrm{T}} = \boldsymbol{C}^{-1}$$

The measured distribution

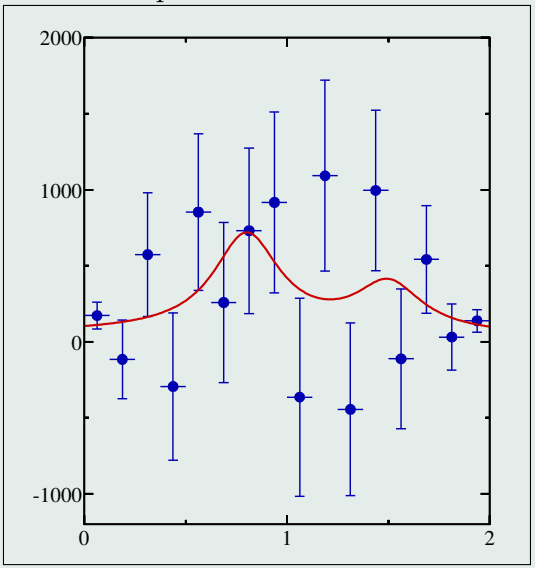
Measured histogram in 40 bins ...



... reconstruction by unfolding will be done in 16 bins, with σ = bin-width.

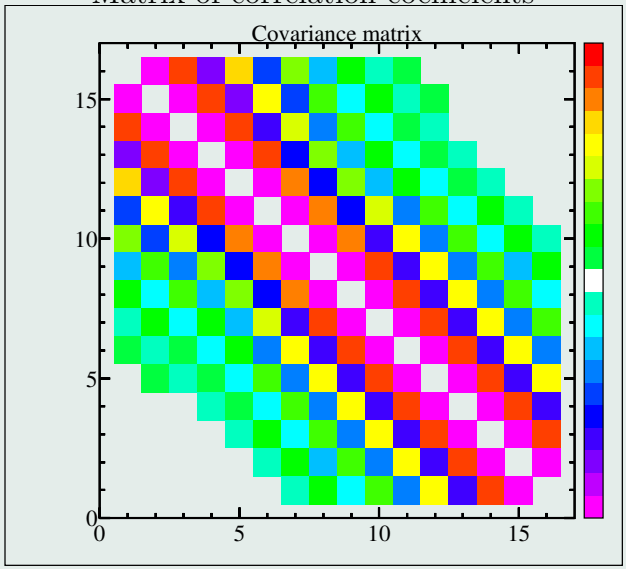
Example of unfolding problem with σ = bin size (Gaussian resolution)

Data points with error bars



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

Matrix of correlation coefficients



$-1 \leq \rho_{ij} \leq +1$

Properties of the LS solution

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

(1) Check of a potential bias in the solution:

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

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

(2) Variance: 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.

Matrix diagonalization

Study transformation properties of symmetric n-by-n matrix $\boldsymbol{C} \quad = (\boldsymbol{A}\boldsymbol{V}_y^{-1}\boldsymbol{A}^{\mathrm{T}})$

by decomposition in the form

$$\boldsymbol{C} = \boldsymbol{U}\boldsymbol{\Lambda}\boldsymbol{U}^{\mathrm{T}} \qquad \boldsymbol{\Lambda} = \boldsymbol{U}^{\mathrm{T}}\boldsymbol{C}\boldsymbol{U} \qquad \boldsymbol{U}^{\mathrm{T}}\boldsymbol{U} = \boldsymbol{U}\boldsymbol{U}^{\mathrm{T}} = \boldsymbol{I} \qquad \boldsymbol{C}^{-1} = \boldsymbol{U}\boldsymbol{\Lambda}^{-1}\boldsymbol{U}^{\mathrm{T}}$$

with a diagonal matrix $\boldsymbol{\Lambda}$.

All matrices in the decomposition are n-by-n matrices:

$$\begin{pmatrix} & \\ & \boldsymbol{C} \\ & \end{pmatrix} = \begin{pmatrix} & \\ & \boldsymbol{U} \\ & \end{pmatrix} \cdot \begin{pmatrix} \lambda_1 & & & \\ & \lambda_2 & & \\ & & \ddots & \\ & & & \lambda_n \end{pmatrix} \cdot \begin{pmatrix} & \\ & \boldsymbol{U}^{\mathrm{T}} \\ & \end{pmatrix}$$

Eigenvalues $\lambda_j \geq 0$ are in decreasing order, with $\lambda_1 \geq \lambda_2 \geq \ldots \lambda_n \geq 0$.

The orthogonal matrix $\boldsymbol{U} = [\boldsymbol{u}_1, \boldsymbol{u}_1, \ldots, \boldsymbol{u}_n]$ is an array of column vectors = eigenvectors $\boldsymbol{u}_j$ of the matrix $\boldsymbol{C}$.

If the inverse $\boldsymbol{C}^{-1}$ exists (i.e. all $\lambda_j > 0$):

$$\boldsymbol{x} = \boldsymbol{C}^{-1}\boldsymbol{b} = \boldsymbol{U}\boldsymbol{\Lambda}^{-1/2}\left(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{U}^{\mathrm{T}}\right)\boldsymbol{b} = \boldsymbol{U}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{c} \qquad \text{with coefficients} \quad \boldsymbol{c} = \boldsymbol{\Lambda}^{-1/2}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{b}\,,$$

i.e. the solution is expressed by a vector $\boldsymbol{c}$ of Fourier coefficients.

The covariance matrix of the vector of Fourier coefficients $\boldsymbol{c}$ is simply the unit matrix, because

$$\boldsymbol{V}_c = \left(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{U}^{\mathrm{T}}\right)\boldsymbol{V}_b\left(\boldsymbol{U}\boldsymbol{\Lambda}^{-1/2}\right) = \left(\boldsymbol{\Lambda}^{-1/2}\boldsymbol{U}^{\mathrm{T}}\right)\boldsymbol{C}\left(\boldsymbol{U}\boldsymbol{\Lambda}^{-1/2}\right) = \boldsymbol{\Lambda}^{-1/2}\boldsymbol{\Lambda}\boldsymbol{\Lambda}^{-1/2} = \boldsymbol{1}\,.$$

The solution can be written with a sum in the form

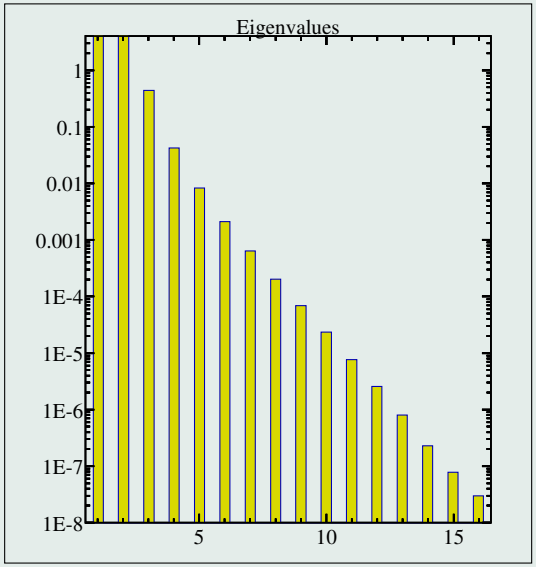
Solution: $\displaystyle \boldsymbol{x} = \sum_{j=1}^{n} \frac{1}{\sqrt{\lambda_j}} c_j \boldsymbol{u}_j$ with $\displaystyle c_j = \frac{1}{\sqrt{\lambda_j}}\left(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{u}_j\right)$ $\displaystyle \boldsymbol{V}_x = \sum_{j=1}^{n} \frac{1}{\lambda_j}\boldsymbol{u}_j\boldsymbol{u}_j^{\mathrm{T}}$

Fourier coefficients c_j, which are insignificant (compatible with zero), should follow a normal distribution $N(0,1)$; this allows simple statistical tests for significance.

The insignificant coefficients can make a huge and dominating contribution to the solution $\boldsymbol{x}$, if the eigenvalues λ_j are small. Diagonalization allows to determine the degree of freedom = number of significant coefficients

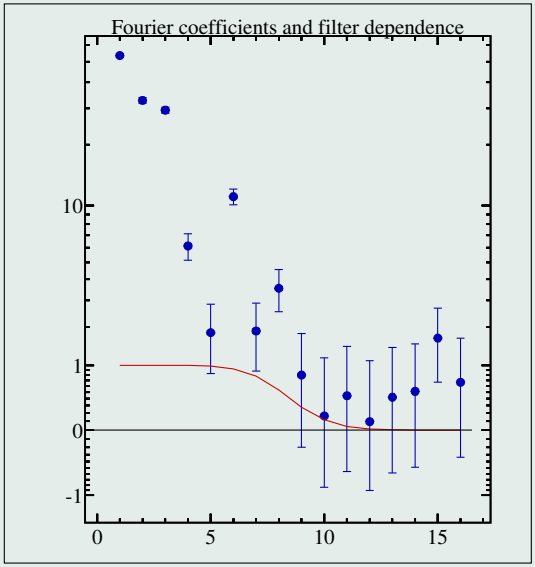
Eigenvalue spectrum and Fourier coefficients

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 error $1 \Rightarrow$ only ten coefficients are significant.

Truncation: The truncated solution can be written with a **sum up to** $k = 10$ only (instead of n) ... the noisy contributions are removed from the result, without the introduction of a bias, ... but the rank of $\boldsymbol{V}_x$ is only k $(< n)$ ($\Rightarrow \boldsymbol{V}_x$ is singular).

4. Regularization methods

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

Thikhonov-Phillips: $F_\tau(\boldsymbol{x}) = \|\boldsymbol{A}\boldsymbol{x} - \boldsymbol{y}\|^2 + \tau \|\boldsymbol{L}\boldsymbol{x}\|^2 = \text{minimum} \qquad \tau > 0 \qquad \boldsymbol{L} = \boldsymbol{1}$

Matrix equation: $\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{A} + \tau \cdot \boldsymbol{1}\right) \quad \boldsymbol{x} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{y}$ to be solved

$(\boldsymbol{C} + \tau \cdot \boldsymbol{1}) \quad \boldsymbol{x} = \boldsymbol{b}$

(diagonalization of $\boldsymbol{C}$) $\quad \boldsymbol{U}\,(\boldsymbol{\Lambda} + \tau \cdot \boldsymbol{1})\,\boldsymbol{U}^{\mathrm{T}}\boldsymbol{x} = \boldsymbol{b}$

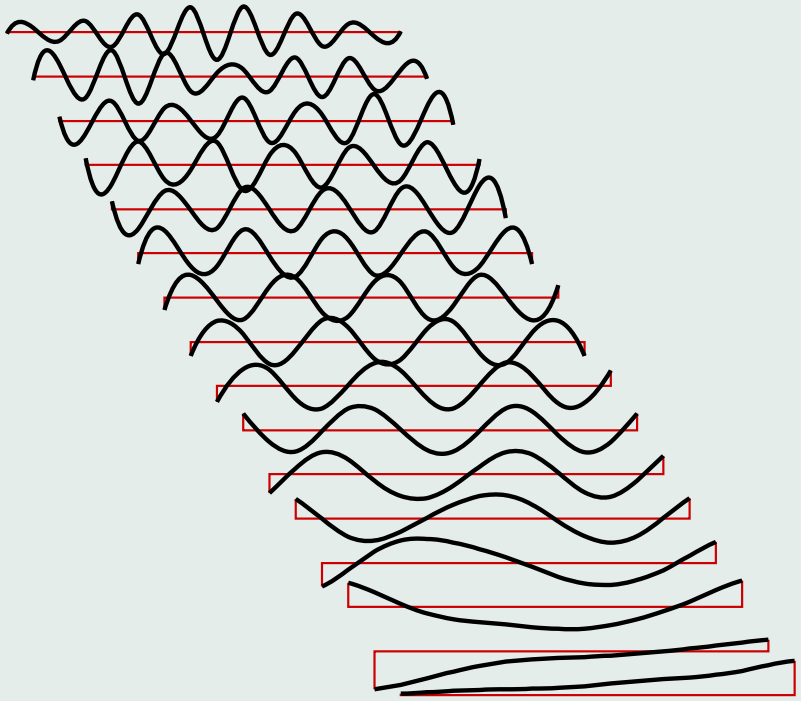
Fourier coefficients $\boldsymbol{c} = \boldsymbol{\Lambda}^{-1/2}\boldsymbol{U}^{\mathrm{T}}\boldsymbol{b}$ filter factor $f_j = \dfrac{\lambda_j}{\lambda_j + \tau} = 1 \ \ldots \ 0.5 \ \ldots \ 0$

Solution: $\boldsymbol{x} = \displaystyle\sum_{j=1}^{n} \frac{f_j}{\sqrt{\lambda_j}} c_j \boldsymbol{u}_j$ with $c_j = \dfrac{1}{\sqrt{\lambda_j}} \left(\boldsymbol{b}^{\mathrm{T}}\boldsymbol{u}_j\right) \quad (\pm 1)$

The result $\boldsymbol{x}$ is expressed as a superposition of eigenvectors $\boldsymbol{u}_j$, each weighted with the Fourier coefficient c_j and $1/\sqrt{\lambda_j}$ (!); the filter factor f_j reduces the effect of insignificant contributions, without introducing a bias.

Plot of the eigenvectors $\boldsymbol{u}_j$

solution $\boldsymbol{x} = \sum_{j=1}^{n} \frac{f_j}{\sqrt{\lambda_j}}\, c_j\, \boldsymbol{u}_j$



The solution $\boldsymbol{x}$ is expressed as a superposition of normalized eigenvectors $\boldsymbol{u}_j$, each

- proportional to the Fourier coefficient c_j (± 1), and
- weigthted by $1/\sqrt{\lambda_j}$ (!);
- the filter factor f_j reduces the effect of insignificant contributions.

Second derivative

Key idea: incorporate certain a-priori assumptions about the size and/or smoothness of the solution!
$\Rightarrow$ control the *norm of the residuals* and, simultaneously, the *norm of second derivative* $\boldsymbol{Lx}$.

$$F_\tau(\boldsymbol{x}) = \|\boldsymbol{Ax} - \boldsymbol{y}\|^2 + \tau \|\boldsymbol{Lx}\|^2 = \text{minimum} \qquad \tau > 0$$

$$x_i'' \;\propto\; x_{i-1} - 2x_i + x_{i+1} \qquad \boldsymbol{L} = \begin{pmatrix} 1 & -2 & 1 & 0 & \cdots & 0 & 0 & 0 \\ 0 & 1 & -2 & 1 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & 1 & -2 & 1 \end{pmatrix}$$

Matrix equation:
$$\left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{A} + \tau \cdot \boldsymbol{L}^{\mathrm{T}}\boldsymbol{L}\right) \quad \boldsymbol{x} = \boldsymbol{A}^{\mathrm{T}}\boldsymbol{V}_y^{-1}\boldsymbol{y} \qquad \text{to be solved}$$
$$\left(\boldsymbol{C} + \tau \cdot \boldsymbol{L}^{\mathrm{T}}\boldsymbol{L}\right) \quad \boldsymbol{x} = \boldsymbol{b}$$
$$\boldsymbol{R}\left(\boldsymbol{1} + \tau \cdot \quad \boldsymbol{S}\right)\boldsymbol{R}^{\mathrm{T}}\,\boldsymbol{x} = \boldsymbol{b} \qquad \text{with} \quad \boldsymbol{R} = \boldsymbol{U}_1\boldsymbol{\Lambda}^{1/2}\boldsymbol{U}_2$$

requires simultaneous diagonalization of two symmetric matrices $\boldsymbol{C}$ and $\boldsymbol{L}^{\mathrm{T}}\boldsymbol{L}$ ($\boldsymbol{S}$ is diagonal).

Fourier coefficients $\boldsymbol{c} = \boldsymbol{U}_2^{\mathrm{T}}\boldsymbol{\Lambda}^{-1/2}\boldsymbol{U}_1^{\mathrm{T}}\boldsymbol{b}$ filter factor $f_i = \dfrac{\lambda_j}{\lambda_j + \tau}$ with $\lambda = S_{jj}^{-1}$

Often better than standard regularization ($\boldsymbol{L} = \boldsymbol{1}$), but depends on order of bins (2-dim solution?)! $\Rightarrow$**more**

Regularization parameter I

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

L-curve method: plot $\|\boldsymbol{L}\boldsymbol{x}_{\text{reg}}\|_2$ versus $\|\boldsymbol{A}\boldsymbol{x}_{\text{reg}} - \boldsymbol{y}\|_2$ 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

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

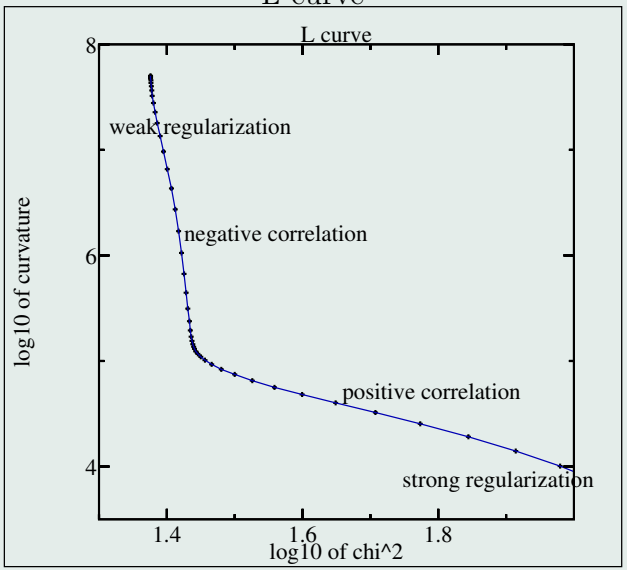
The global correlation coefficient is a measure of the total amount of correlation between element j of $\boldsymbol{x}$ and all other elements.

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.

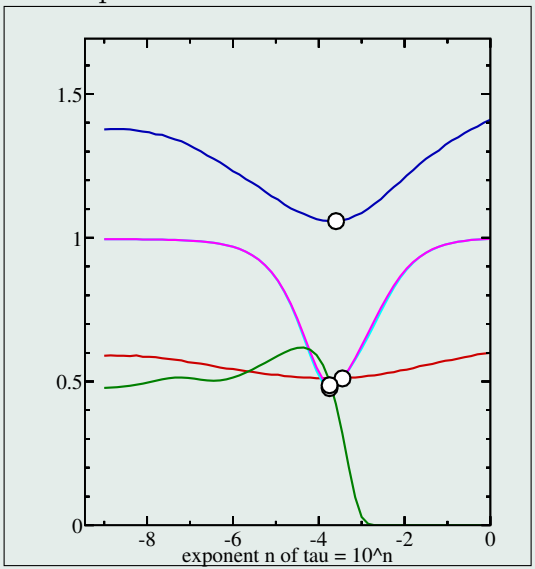
. . . seems to be the best method!

Regularization parameter II

L-curve



Dependence as a function of τ



optimal regularization in region of largest curvature

circle=minimum

magenta/cyan=mean global correlation

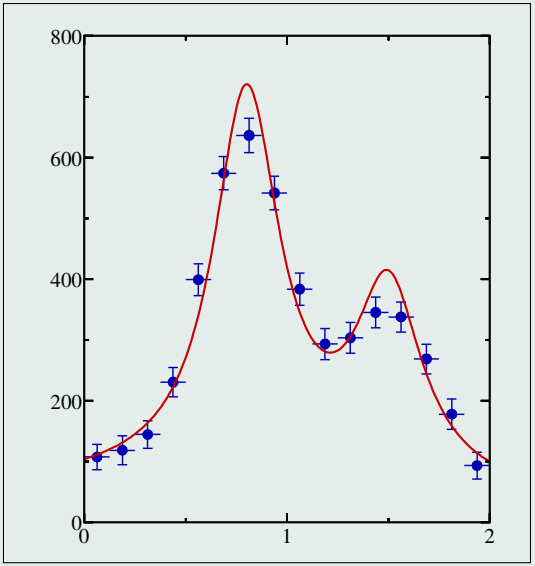
green=χ^2 probability

red=average probability

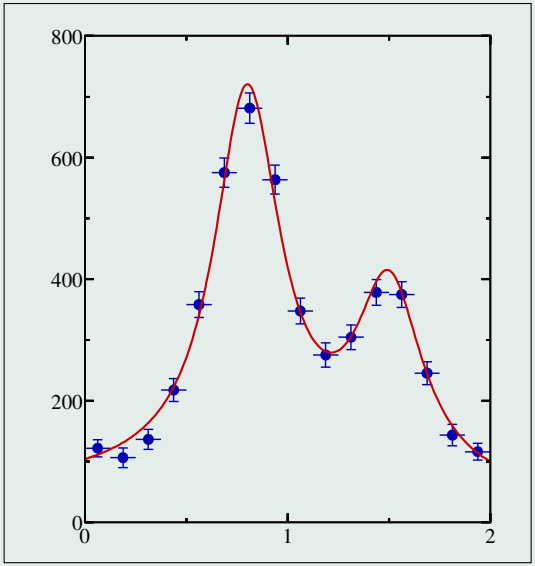
blue=relative sigma

Regularization parameter τ taken from minimum of mean of global correlation coefficients:

$\sigma =$ bin-width $\qquad\qquad$ $\sigma = \frac{1}{2} \times$ bin-width



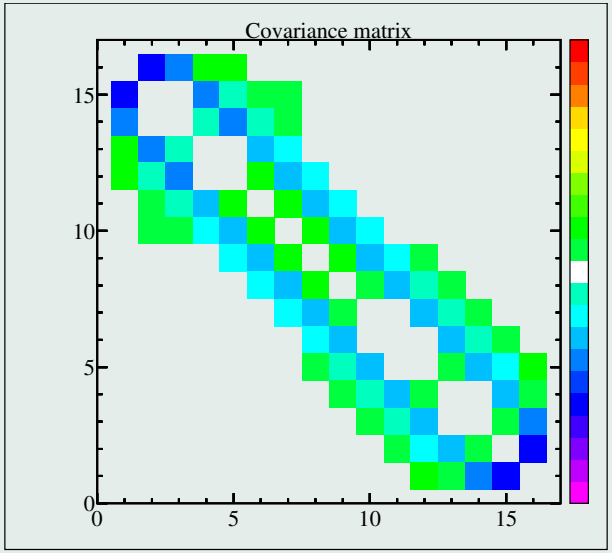
Small bias at the peaks.



Reduced bias at the peaks.

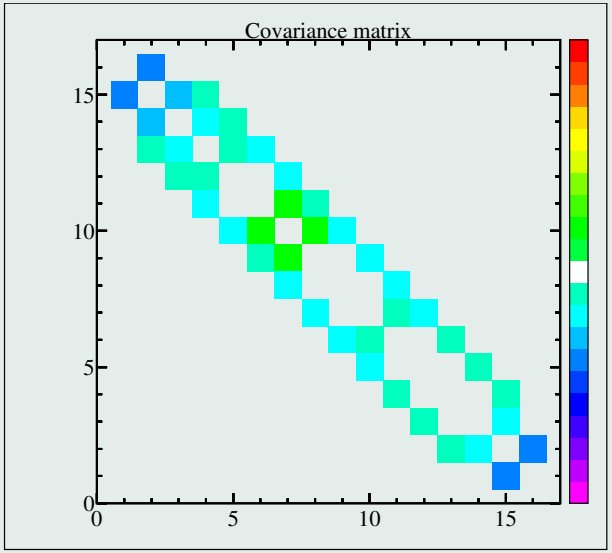
Correlation matrices

$\sigma =$ bin-width



Correlations $\rho_{j,j+1} \approx +0.1$ and $\rho_{j,j+2} \approx -0.25$

$\sigma = \frac{1}{2} \times$ bin-width



Correlations $\rho_{j,j+1} \approx \pm 0.08$ and $\rho_{j,j+2} \approx -0.1$

Summary

Pragmatic decisions about the selection of unfolding methods alone are suboptimal:

- Tests of black-box unfolding algorithms on a few selected examples are not sufficient;
- it is essential, to *understand* the statistical and mathematical properties of the algorithms!

Experimental data, measured with limited resolution and acceptance, require unfolding to allow a correct interpretation:

- Statistical errors are increased, if there are migration effects.
- Number of bins of unfolded distribution is in general small, and has to be adjusted to the size of migration effects – otherwise correlations between bins will be large.
- Methods based on **Least squares (for Gaussian errors)** and **Poisson ML (for counting data)**, supplemented by **regularization** terms, using orthogonalization methods (singular values or diagonalization), exist and allow to control a potential bias and to propagate the measurement errors.

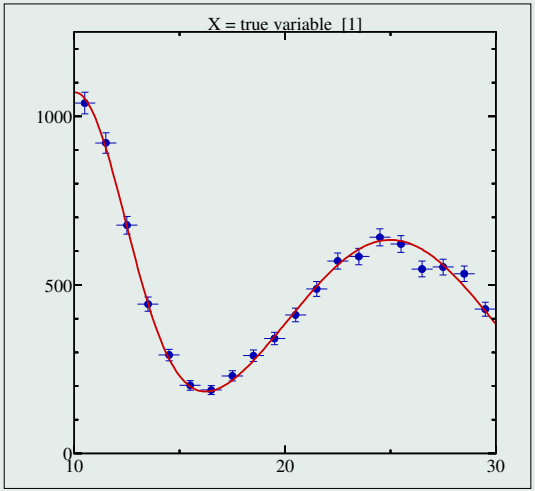
Unfolding example

- Unfolding by inversion
- Eigenvalues and coefficients
- Truncation
- Correlation matrices
- Regularization method
- Bin combination: $2 \rightarrow 1$
- Low-pass regularization with $3 \rightarrow 1$ averaging
- Low-pass regularization
-
- Regularized unfolding by RUN

Unfolding by inversion

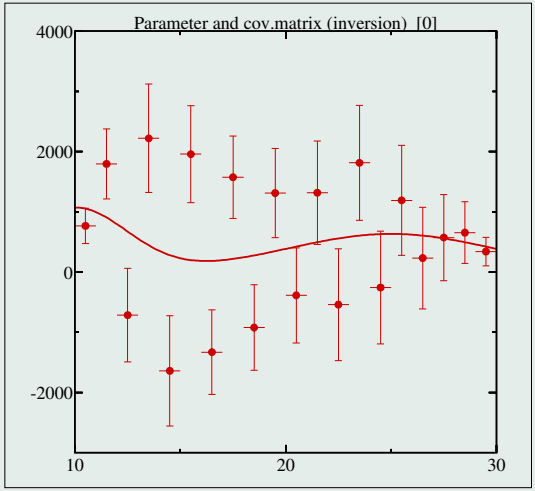
Example of unfolding problem with σ/binwidth $= 3$

Perfect resolution (no unfolding)



Histogram
for sample with 10 000 entries.

Unfolding by inversion

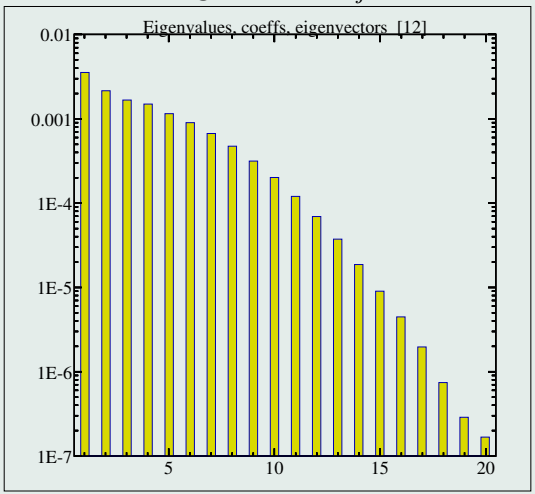


Huge fluctuations, due to large negative correlations: neighbour bin -95%, second $+85\%$.

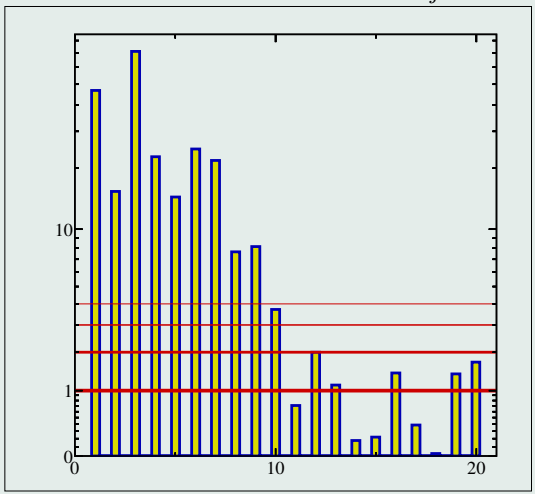
True curve $f(x)$ is shown in red.

Example of unfolding problem with σ/binwidth = 3

Eigenvalues $\boldsymbol{u}_j$



Normalized coefficients c_j



Eigenvalues decrease by 4 orders of magnitude.

Only 10 of the 20 coefficients are significant.

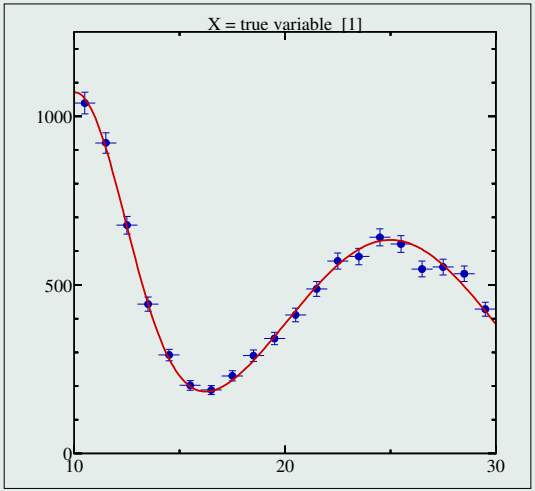
Red lines are for 1, 2, 3 and 4 standard deviations.
Statistical errors are 1 for all coefficients.

Diagonalization allows to determine the degree of freedom = number of significant coefficients

Truncation

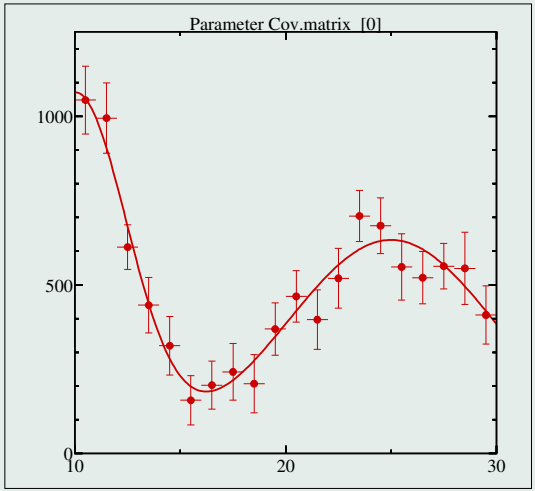
Example of unfolding problem with σ/binwidth $= 3$

Perfect resolution (no unfolding)



Histogram
for sample with 10 000 entries.

Truncation method (15 terms kept)



Reduced fluctuations, but large errors due to large correlations.

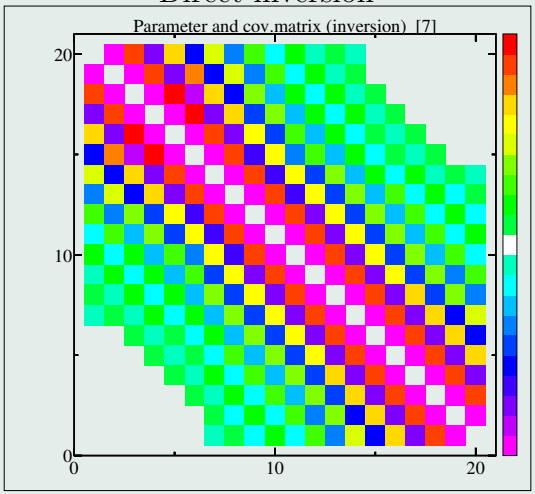
True curve $f(x)$ is shown in red.

Correlation matrices

Example of unfolding problem with σ/binwidth $= 3$

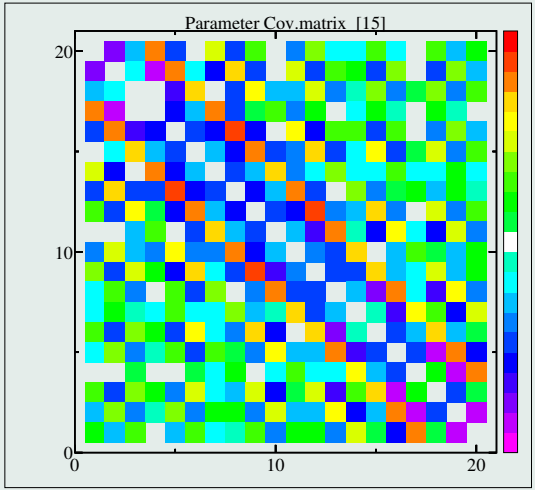
Colour graph of correlation coefficients ρ with range $-1 \ldots +1$:

Direct inversion



Large negative and positive correlations: neighbour bin -95%, second neighbour $+85\%$.

Truncation method (15 terms kept)

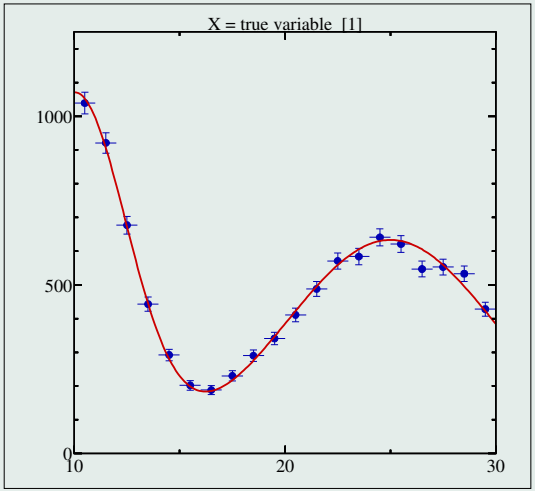


Correlations reduced, negative for neighbour bin -40% and second neighbour -30%.

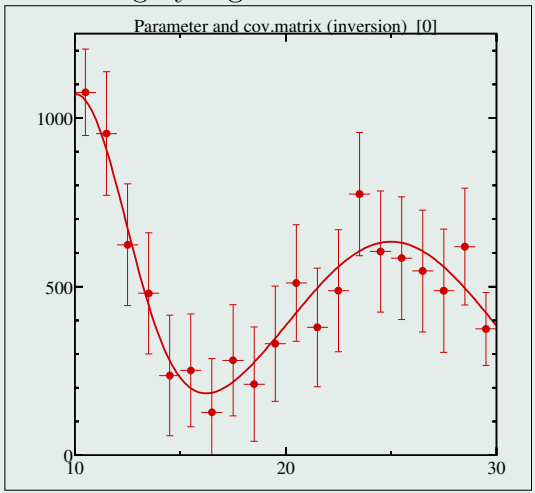
Regularization

Example of unfolding problem with σ/binwidth $= 3$

Perfect resolution (no unfolding)



Unfolding by regularization $\tau = 10^{-5}$



Histogram
for sample with 10 000 entries.

Reduced fluctuations: correlations neighbour bin -80% and second neighbour $+40\%$.

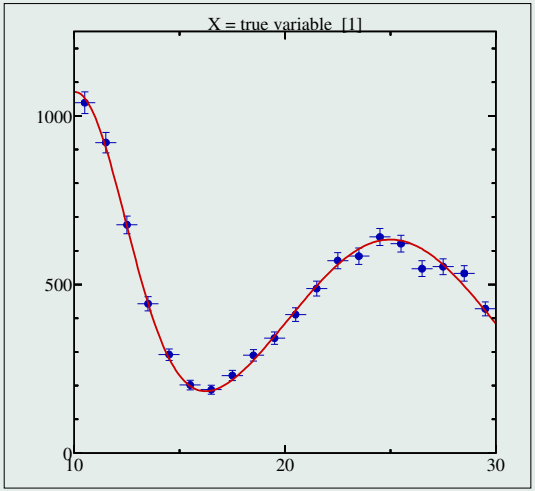
True curve $f(x)$ is shown in red.

Bin combination $2 \to 1$

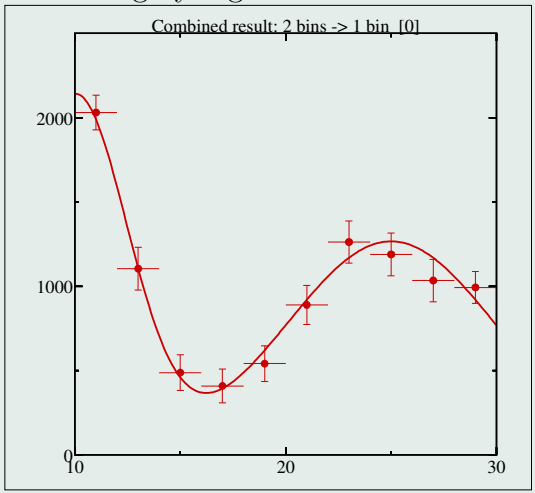
Reduce correlations and errors by combination of two bins to one:

Example of unfolding problem with σ/binwidth $= 3$ $\Rightarrow 1.5$

Perfect resolution (no unfolding)



Unfolding by regularization $\tau = 10^{-5}$



Histogram
for sample with 10 000 entries.

Reduced fluctuations: correlations neighbour bin -20% and second neighbour -30%.

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

Low-pass regularization with $3 \to 1$ averaging

Without regularization there are large bin-to-bin fluctuations due to negative correlations between neighbour bins. These fluctuations can be suppressed in a low-pass filter by averaging 3-to-1 bins:

$$\overline{x}_j = \tfrac{1}{4}x_{j-1} + \tfrac{1}{2}x_j + \tfrac{1}{4}x_{j+1} \qquad \text{or general} \qquad \overline{x}_j = a_j x_{j-1} + (1-2a_j)\, x_j + a_j x_{j+1}$$

The factor a_j can be chosen to minimize[1)] the variance of $\overline{x}_j$, using the known matrix $\boldsymbol{V}_x$.

Pro: Fluctuations are really suppressed and the true dependence is clearer visible. ⇒ **more**

No bias, if number of bins large and no strong structure.

Con: In regions of larger second-derivatives a bias is introduced, because the above filter assumes an almost linear dependence over 3-point regions.
First and last bins disappear.

The general averaging algorithm for this "local"-regularization method:

$$\overline{\boldsymbol{x}} = \boldsymbol{T}\boldsymbol{x}$$
$$\boldsymbol{V}_{\overline{x}} = \boldsymbol{T}\boldsymbol{V}_x\boldsymbol{T}^{\mathrm{T}}$$

with $\boldsymbol{T} =$

$$\begin{pmatrix} a_2 & 1-2a_2 & a_2 & 0 & \cdots & 0 & 0 & 0 \\ 0 & a_3 & 1-2a3 & a_3 & \cdots & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & a_{n-1} & 1-2a_{n-1} & a_{n-1} \end{pmatrix}$$

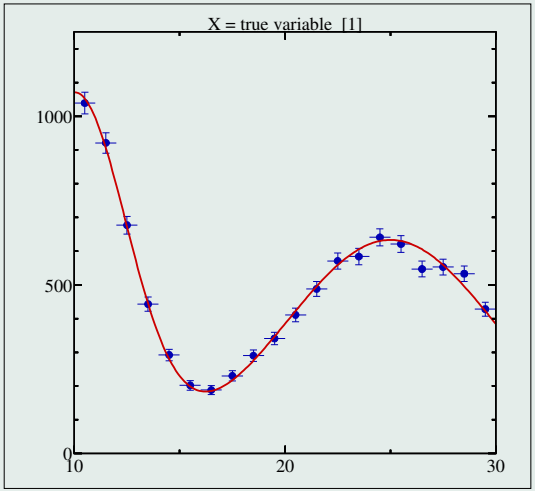
Question: Can the bias in curved regions be avoided in an improved averaging algorithm?

1) O. Helene et al., NIM A 523 (2004) 186; NIM A 580 (2007) 1466 - 1473

Low-Pass regularization

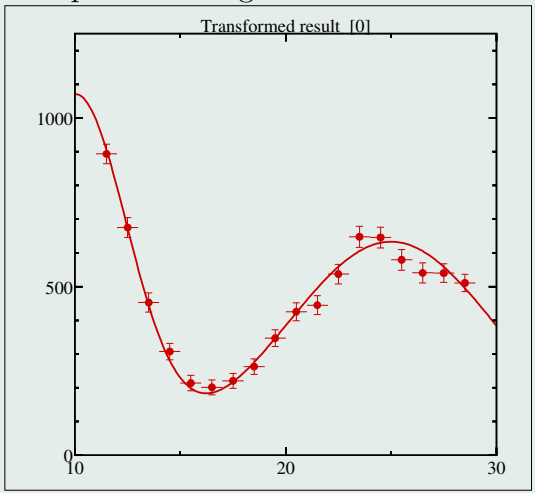
Example of unfolding problem with σ/binwidth $= 3$

Perfect resolution (no unfolding)



Histogram
for sample with 10 000 entries.

Low pass after regularization $\tau = 10^{-5}$



Reduced correlations: neighbour bin -10% and second neighbour -30%.

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

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