A Survey of Unfolding Methods for Particle Physics

Glen Cowan

Royal Holloway, University of London

g.cowan@rhul.ac.uk

Advanced Statistical Techniques in Particle Physics

Grey College, University of Durham

18–22 March, 2002

Outline

(*i*) The mathematical problem

notation, response function (matrix)

(*ii*) Inverting the response matrix

and why this doesn't (and does) work

(*iii*) Correction factors

and why we use them anyway

(*iv*) Regularized unfolding

Tikhonov, MaxEnt

Variance and bias

Choice of the regularization parameter

(*v*) Bayesian (iterative) method

(*vi*) Further examples

HEP, astronomy, medical imaging, art

(*vii*) Can we do better?

How we perhaps could

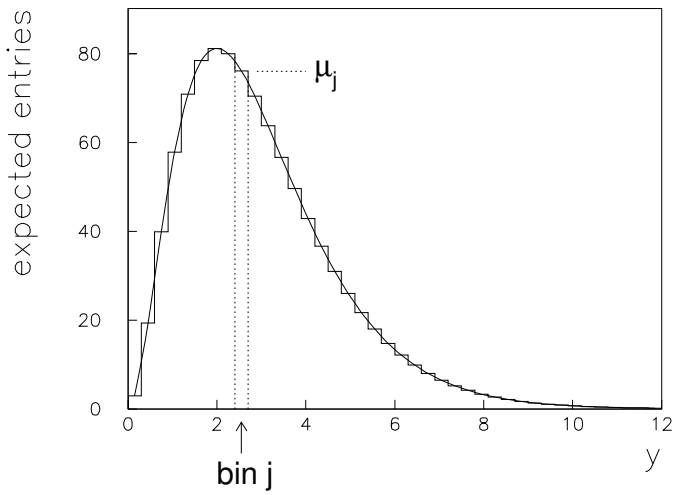
Formulation of the unfolding problem

Consider random variable y. Goal: determine pdf $f(y)$.

If parametrization $f(y;\vec{\theta})$ known,

maximum likelihood $\rightarrow \hat{\vec{\theta}}$

If no parametrization available, construct histogram:



$$p_j = \int_{\text{bin } j} f(y)\, dy \quad j = 1, \ldots, M$$

$$\mu_j = \mu_{\text{tot}}\, p_j \leftarrow \text{the 'true histogram'}$$

The goal: construct estimators for the μ_j (or p_j).

$\rightarrow$ number of parameters = number of bins, M

The problem: y cannot be measured without error.

$\rightarrow$ migration of entries between bins

$\rightarrow$ $f(y)$ is 'smeared out', peaks broadened

Response matrix

Effect of measurement errors: y = true value

x = observed value

$$f_{\text{meas}}(x) = \int R(x|y)\, f_{\text{true}}(y)\, dy$$

↓ discretize

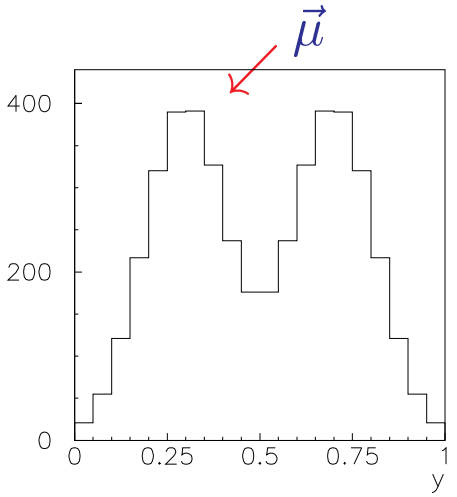
$$\nu_i = \sum_{j=1}^{M} R_{ij}\, \mu_j\,, \quad i = 1, \ldots, N$$

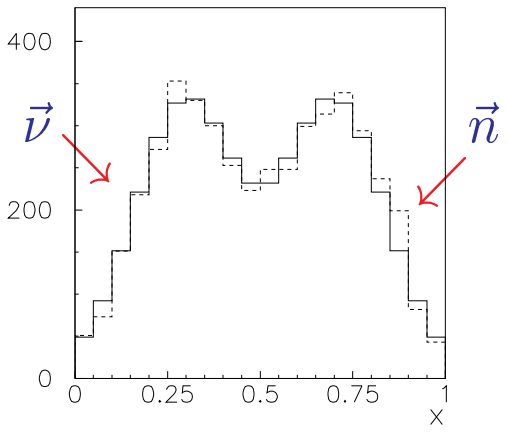
observed histogram (expec. val.) | response matrix | true histogram

$$R_{ij} = P(\text{observed in bin } i \,|\, \text{true value in bin } j)$$

The data: $\vec{n} = (n_1, \ldots, n_N)$, where $\nu_i = E[n_i]$.

N.B. $\vec{\mu}$, $\vec{\nu}$ constants, $\vec{n}$ subject to statistical fluctuations.





Efficiency, background

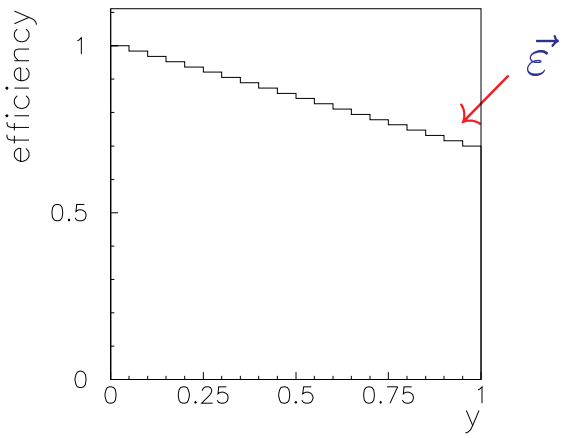
Sometimes an event goes undetected,

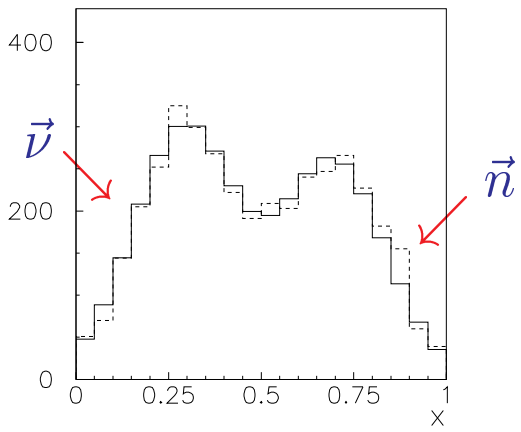
$$\sum_{i=1}^{N} R_{ij} = \sum_{i=1}^{N} P(\text{observed in bin } i \,|\, \text{true value in bin } j)$$

$$= P(\text{observed anywhere} \,|\, \text{true value in bin } j)$$

$$= \varepsilon_j \quad \text{(efficiency)}$$

N.B. ε_j depends on bin j of *true* histogram.





Sometimes we observe something when no true event occurred,

$$\rightarrow \nu_i = \sum_{j=1}^{M} R_{ij}\,\mu_j + \beta_i$$

β_i = expected number of background events in *observed* histogram.

For now, assume $\vec{\beta}$ is known.

Summary of ingredients

'true' histogram: $\vec{\mu} = (\mu_1, \ldots, \mu_M)\,, \quad \mu_{\rm tot} = \sum_{j=1}^{M} \mu_j$

probabilities: $\vec{p} = (p_1, \ldots, p_M) = \vec{\mu}/\mu_{\rm tot}$

expectation values for observed histogram: $\vec{\nu} = (\nu_1, \ldots, \nu_N)$

observed histogram: $\vec{n} = (n_1, \ldots, n_N)$

response matrix: $R_{ij} = P(\text{observed in bin } i \,|\, \text{true value in bin } j)$

efficiencies: $\varepsilon_j = \sum_{i=1}^{N} R_{ij}$

expected background: $\vec{\beta} = (\beta_1, \ldots, \beta_N)$

These are related by:

$$E[\vec{n}] = \vec{\nu} = R\vec{\mu} + \vec{\beta}$$

To find estimators for $\vec{\mu}$, we need probability law, e.g.

$$P(n_i; \nu_i) = \frac{\nu_i^{n_i}}{n_i!} e^{-\nu_i} \quad \text{(Poisson)},$$

or covariance matrix,

$$V_{ij} = \text{cov}[n_i, n_j],$$

in order to construct likelihood function or χ^2.

Why unfold?

Often unfolding not needed, e.g. when

comparing to prediction of existing theory, better to

'fold' theory with detector response,

i.e. include detector effects in its prediction,

compare this with uncorrected ('raw') data $\vec{n}$.

$\rightarrow$ simpler, more robust.

But, 'folding' theory with detector effects requires response matrix,

usually this knowledge not retained after publication of result.

Unfolded distribution can be compared directly to:

predictions of theories,

unfolded results from other experiments.

Usually unfolded result is more useful, since new theories

may be invented when response matrix is long gone.

In HEP often unfold:

structure functions $F_2(x, Q^2)$,

τ spectral functions (hadronic mass distributions),

hadronic event-shape distributions,

particle multiplicity distributions.

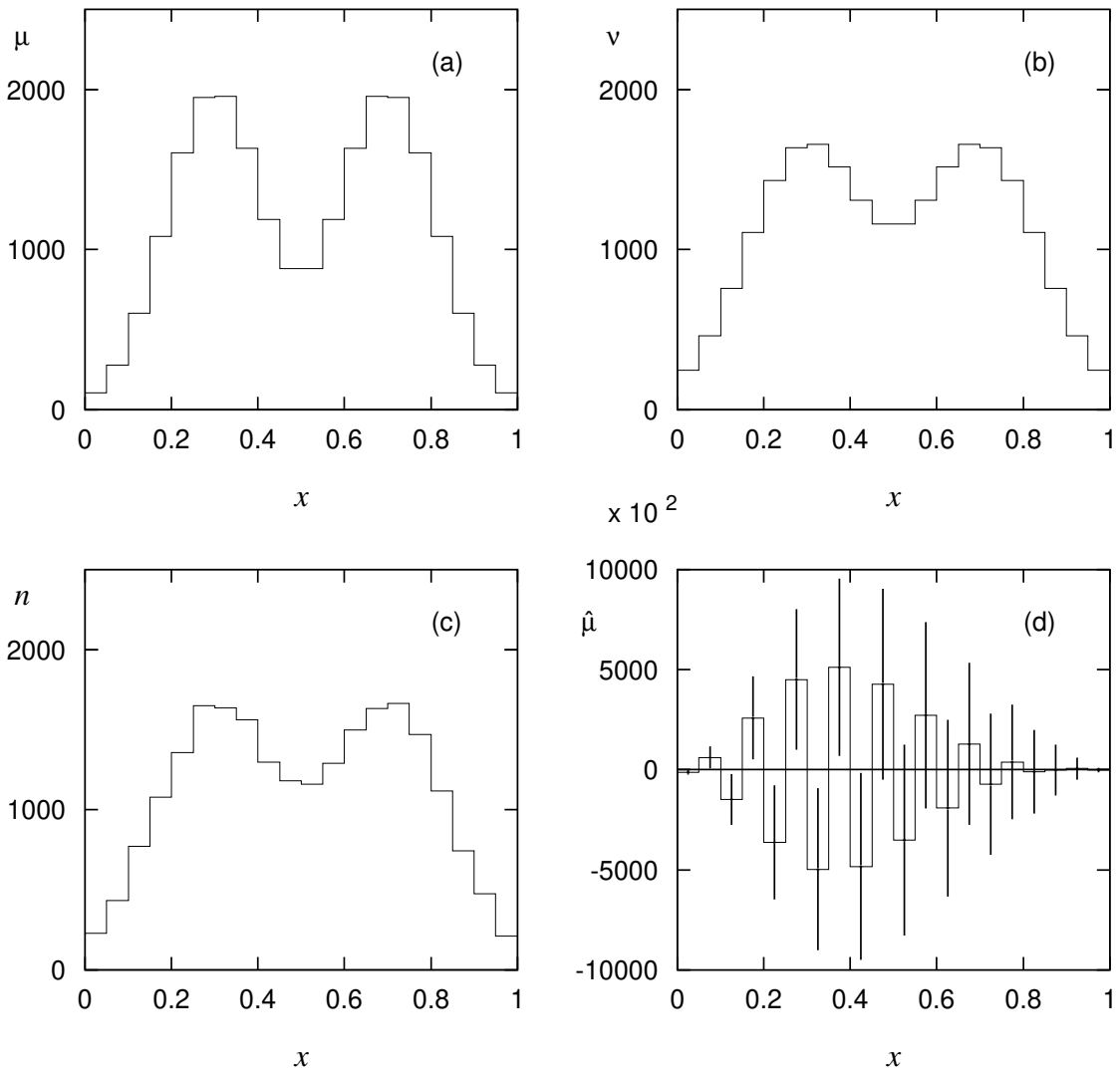
Inverting the response matrix

Assume $\vec{\nu} = R\vec{\mu} + \vec{\beta}$ can be inverted: $\vec{\mu} = R^{-1}(\vec{\nu} - \vec{\beta})$

Suppose data are Poisson: $P(n_i; \nu_i) = \dfrac{\nu_i^{n_i}}{n_i!} e^{-\nu_i}$

$$\rightarrow \log L(\vec{\mu}) = \sum_{i=1}^{N} (n_i \log \nu_i - \nu_i)$$

ML estimator is $\hat{\vec{\nu}} = \vec{n} \;\rightarrow\; \hat{\vec{\mu}} = R^{-1}(\vec{n} - \vec{\beta})$.

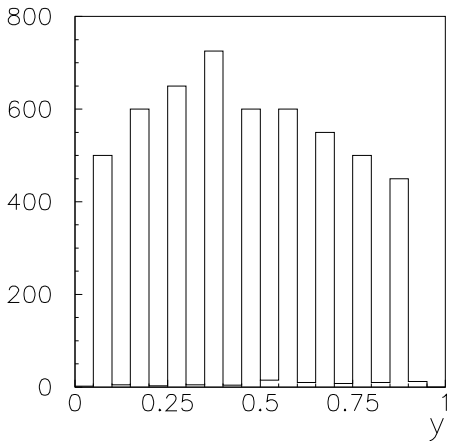


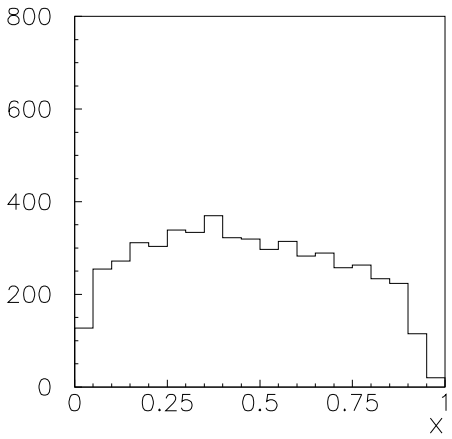
Catastrophic failure (?!)

What went wrong?

Suppose $\vec{\mu}$ really had fine structure:

$\vec{\mu} \rightarrow$





Applying R washes this out, but leaves residual structure.

$\leftarrow \vec{\nu} = R\vec{\mu}$

Applying R^{-1} to $\vec{\nu}$ puts the fine structure back: $\vec{\mu} = R^{-1}\vec{\nu}$.

But we don't have $\vec{\nu}$, only $\vec{n}$.

$\vec{n}$ *has* small bumps due to statistical fluctuations.

$\rightarrow R^{-1}$ 'thinks' this is residual of original fine structure,

$\hat{\vec{\mu}} = R^{-1}\vec{n}$ winds up getting huge 'fine structure'.

ML solution revisited

$$E[\hat{\vec{\mu}}] = R^{-1}(E[\vec{n}] - \vec{\beta}) = \vec{\mu} \;\rightarrow\; \text{unbiased!}$$

Compute variance of estimators,

$$U_{ij} = \mathrm{cov}[\hat{\mu}_i, \hat{\mu}_j] = \sum_{k,l=1}^{N} (R^{-1})_{ik}\,(R^{-1})_{jl}\,\mathrm{cov}[n_k, n_l]$$

$$= \sum_{k=1}^{N} (R^{-1})_{ik}\,(R^{-1})_{jk}\,\nu_k$$

Recall RCF bound for unbiased estimators,

$$(U^{-1})_{kl} \;=\; -E\left[\frac{\partial^2 \log L}{\partial\mu_k\,\partial\mu_l}\right] \;=\; \sum_{i=1}^{N} \frac{R_{ik}\,R_{il}}{\nu_i}$$

Inverting gives

$$U_{ij} = \sum_{k=1}^{N} (R^{-1})_{ik}\,(R^{-1})_{jk}\,\nu_k$$

$\rightarrow$ ML estimator has minimum variance among unbiased estimators.

But this variance was huge!

$\rightarrow$ to reduce variance, we must introduce some bias.

Strategy: accept small bias (systematic error) in exchange for large reduction in variance (statistical error).

Correction factor method

Use equal binning for $\vec{\mu}$, $\vec{\nu}$ and take $\hat{\mu}_i = C_i(n_i - \beta_i)$, where

$$C_i = \frac{\mu_i^{\rm MC}}{\nu_i^{\rm MC}} \quad \text{(correction factor)}$$

$\nu_i^{\rm MC}$ and $\mu_i^{\rm MC}$ from Monte Carlo simulation (no background).

$$U_{ij} = {\rm cov}[\hat{\mu}_i, \hat{\mu}_j] = C_i^2 \, {\rm cov}[n_i, n_j]$$

Usually $C_i \approx O(1)$, so variances don't blow up.

But the bias $b_i = E[\hat{\mu}_i] - \mu_i$ is

$$b_i = \left(\frac{\mu_i^{\rm MC}}{\nu_i^{\rm MC}} - \frac{\mu_i}{\nu_i^{\rm sig}} \right) \nu_i^{\rm sig} \quad , \text{ where } \nu_i^{\rm sig} = \nu_i - \beta_i$$

Need to include systematic error due to MC dependence.

N.B. bias tends to pull $\hat{\vec{\mu}}$ towards $\vec{\mu}^{\rm MC}$

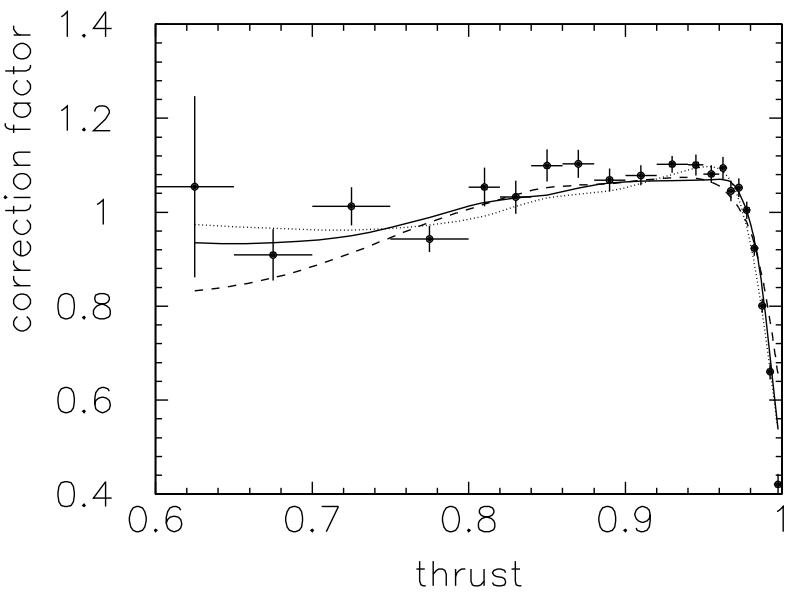
$\rightarrow$ hard to test models.

Not too bad if bin width $\geq$ several times resolution.

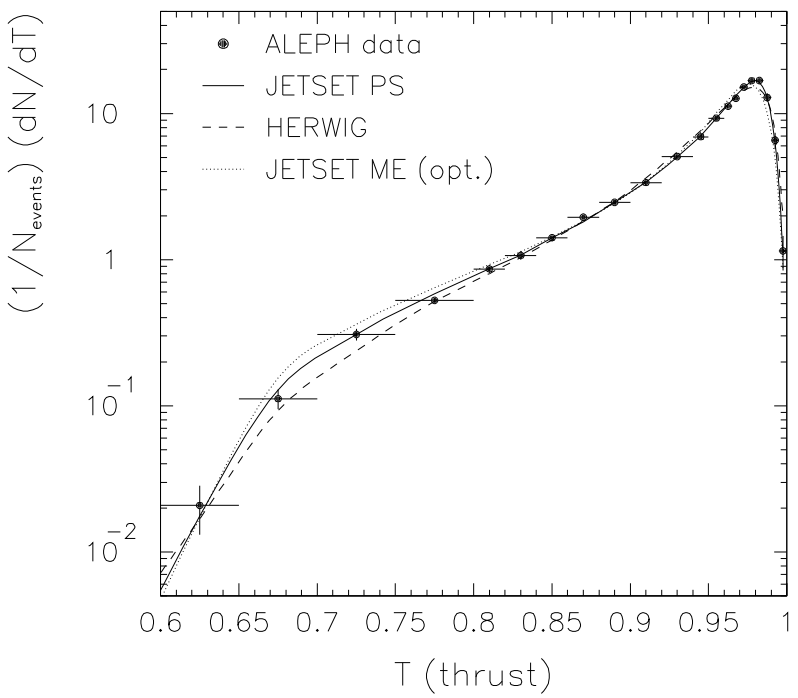
Often used for distributions of event-shape variables.

Example of correction factors in $e^+e^- \to$ hadrons

The correction factors:



The unfolded distribution compared to model predictions:



Regularized unfolding

Consider 'reasonable' estimators such that for some $\Delta \log L$,

$$\log L(\vec{\mu}) \geq \log L_{\max} - \Delta \log L$$

Out of these estimators, choose the 'smoothest', by maximizing

$$\Phi(\vec{\mu}) = \alpha \, \log L(\vec{\mu}) \, + \, S(\vec{\mu}),$$

$S(\vec{\mu})$ = regularization function (measure of smoothness),

α = regularization parameter (choose to give desired $\Delta \log L$)

In addition require $\sum_{i=1}^{N} \nu_i = \sum_{i,j} R_{ij}\mu_j = n_{\text{tot}}$, i.e. maximize

$$\varphi(\vec{\mu}, \lambda) = \alpha \, \log L(\vec{\mu}) \, + \, S(\vec{\mu}) \, + \, \lambda \left[n_{\text{tot}} \, - \, \sum_{i=1}^{N} \nu_i \right]$$

where λ is a Lagrange multiplier,

$$\partial\varphi/\partial\lambda = 0 \rightarrow \sum_{i=1}^{N} \nu_i = n_{\text{tot}}.$$

$\alpha = 0$ gives smoothest solution (ignores data!),

$\alpha \rightarrow \infty$ gives ML solution (variance too large).

We need: regularization function $S(\vec{\mu})$,

a prescription for setting α.

Goodness of resulting estimators judged by their bias and variance.

Variance and bias of $\hat{\vec{\mu}}$

In general, the equations determining $\hat{\vec{\mu}}(\vec{n})$ are nonlinear.

Expand $\hat{\vec{\mu}}(\vec{n})$ about $\vec{n}_{\rm obs}$ (observed data set),

$$\hat{\vec{\mu}}(\vec{n}) \approx \hat{\vec{\mu}}_{\rm obs} \; - \; A^{-1}B(\vec{n} - \vec{n}_{\rm obs}) \; ,$$

$$A_{ij} = \begin{cases} \dfrac{\partial^2\varphi}{\partial\mu_i\partial\mu_j}, & i,j = 1,\ldots,M, \\ \dfrac{\partial^2\varphi}{\partial\mu_i\partial\lambda} \; = \; -1, & i = 1,\ldots,M, j = M+1, \\ \dfrac{\partial^2\varphi}{\partial\lambda^2} \; = \; 0, & i = M+1, j = M+1, \end{cases}$$

$$B_{ij} = \begin{cases} \dfrac{\partial^2\varphi}{\partial\mu_i\partial n_j}, & i = 1,\ldots,M, j = 1,\ldots,N, \\ \dfrac{\partial^2\varphi}{\partial\lambda\partial n_j} \; = \; 1, & i = M+1, j = 1,\ldots,N. \end{cases}$$

Use error propagation to get covariance $U_{ij} = {\rm cov}[\hat{\mu}_i, \hat{\mu}_j]$,

$$U = CVC^T \;\; \text{where } C = A^{-1}B,$$

and estimators for the bias, $b_i = E[\hat{\mu}_i] - \mu_i$,

$$\hat{b}_i = \sum_{j=1}^{N} C_{ij}(\hat{\nu}_j - n_j) \; = \; \sum_{j=1}^{N} \frac{\partial\hat{\mu}_i}{\partial n_j}(\hat{\nu}_j - n_j),$$

where $\hat{\vec{\nu}} = R\hat{\vec{\mu}} + \vec{\beta}\,.$ (N.B. $\hat{\vec{\nu}} \neq \vec{n}.$)

Choosing the regularization parameter α

$\alpha = 0 \rightarrow \hat{\vec{\mu}}$ maximally smooth (ignores data).

$\alpha \rightarrow \infty \rightarrow$ ML solution (no bias, very large variance).

Possible criteria for best trade-off between bias and variance:

Minimize mean squared error,

$$\text{MSE} = \frac{1}{M} \sum_{i=1}^{M} (U_{ii} + \hat{b}_i^2) \,, \quad \text{or}$$

$$\text{MSE}' = \frac{1}{M} \sum_{i=1}^{M} \frac{U_{ii} + \hat{b}_i^2}{\hat{\mu}_i} .$$

Or look at changes in χ^2 from unregularized (ML) solution,

$$\Delta\chi^2 = 2\Delta \log L = N \,, \quad \text{or}$$

$$\Delta\chi^2_{\text{eff}} = (\hat{\vec{\nu}} - \vec{n})^T \, R \, C \, V^{-1} (R \, C)^T \, (\hat{\vec{\nu}} - \vec{n}) \; = \; 1 \; .$$

Or require that bias be consistent with zero to within its own error,

$$\chi^2_b = \sum_{i=1}^{M} \frac{\hat{b}_i^2}{W_{ii}} = M \quad \text{where } W_{ij} = \text{cov}[\hat{b}_i, \hat{b}_j].$$

i.e. if bias significantly different from zero, we would subtract it;

$\rightarrow$ equivalent to going to smaller $\Delta \log L$ or larger α (less bias).

Tikhonov regularization

Take measure of smoothness = mean square of kth derivative,

$$S[f_{\text{true}}(y)] = -\int \left(\frac{d^k f_{\text{true}}(y)}{dy^k} \right)^2 dy \ , \text{ where } k = 1, 2, \ldots$$

Often take $k = 2$, $\rightarrow$ $S \approx$ mean squared curvature.

For histogram this becomes (e.g. for $k = 2$),

$$S(\vec{\mu}) = -\sum_{i=1}^{M-2} (-\mu_i + 2\mu_{i+1} - \mu_{i+2})^2$$

N.B. 2nd derivative not well defined for first and last bins.

If we use Tikhonov ($k = 2$) with $\log L = -\frac{1}{2}\chi^2$,

$$\varphi(\vec{\mu}, \lambda) = -\frac{\alpha}{2}\chi^2(\vec{\mu}) + S(\vec{\mu}) \quad \text{quadratic in } \mu_i,$$

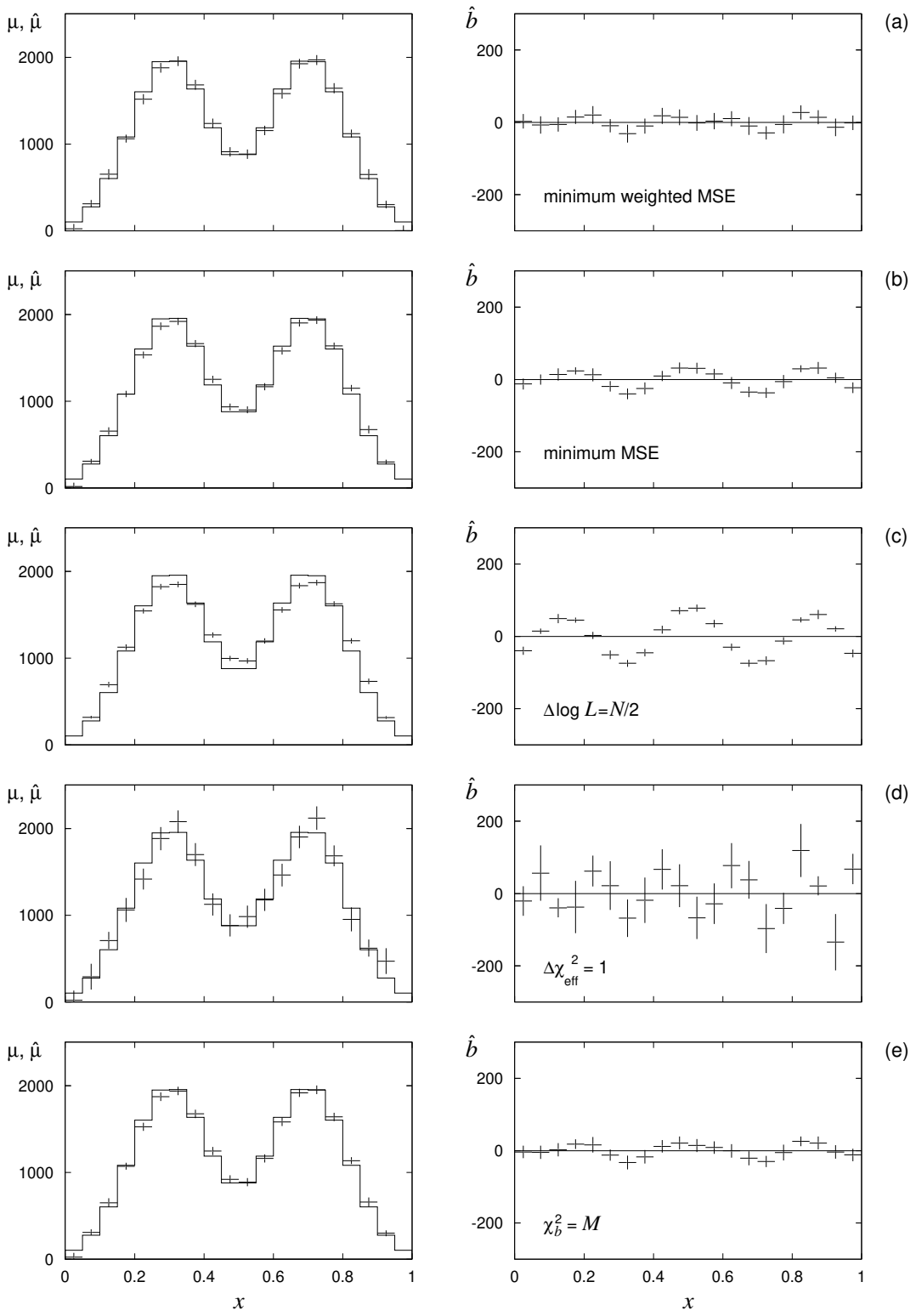
$\rightarrow$ setting derivatives of φ equal to zero gives linear equations.

Several programs available for use in HEP:

`RUN`, Blobel

`SVD`, Höcker and Kartvelishvili

Examples with Tikhonov regularization ($k = 2$)



N.B. solution not always positive.

SVD implementation of Tikhonov regularization

A. Höcker, V. Kartvelishvili, NIM A**372** (1996) 469.

→ program **GURU**, see **www.hep.man.ac.uk/~vato**

Minimizes $\chi^2 + \tau \sum_i [(\mu_{i+1} - \mu_i) - (\mu_i - \mu_{i-1})]^2$

Numerical implementation → Singular Value Decomposition

Recommendations for setting regularization parameter τ:

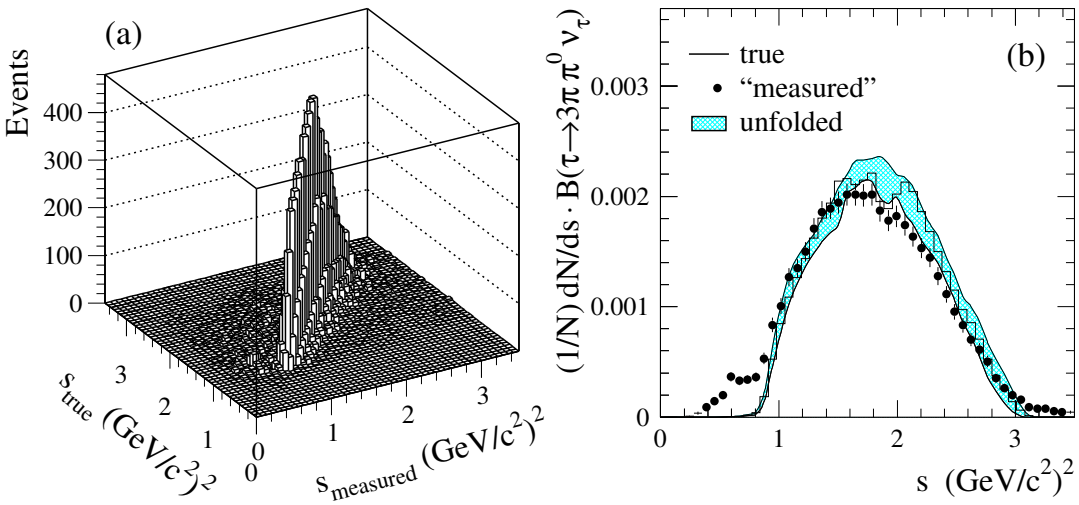
Transform variables so estimators are $N(0,1)$,

number significantly different from zero → prescription for τ;

or base choice on unfolding of test distributions.

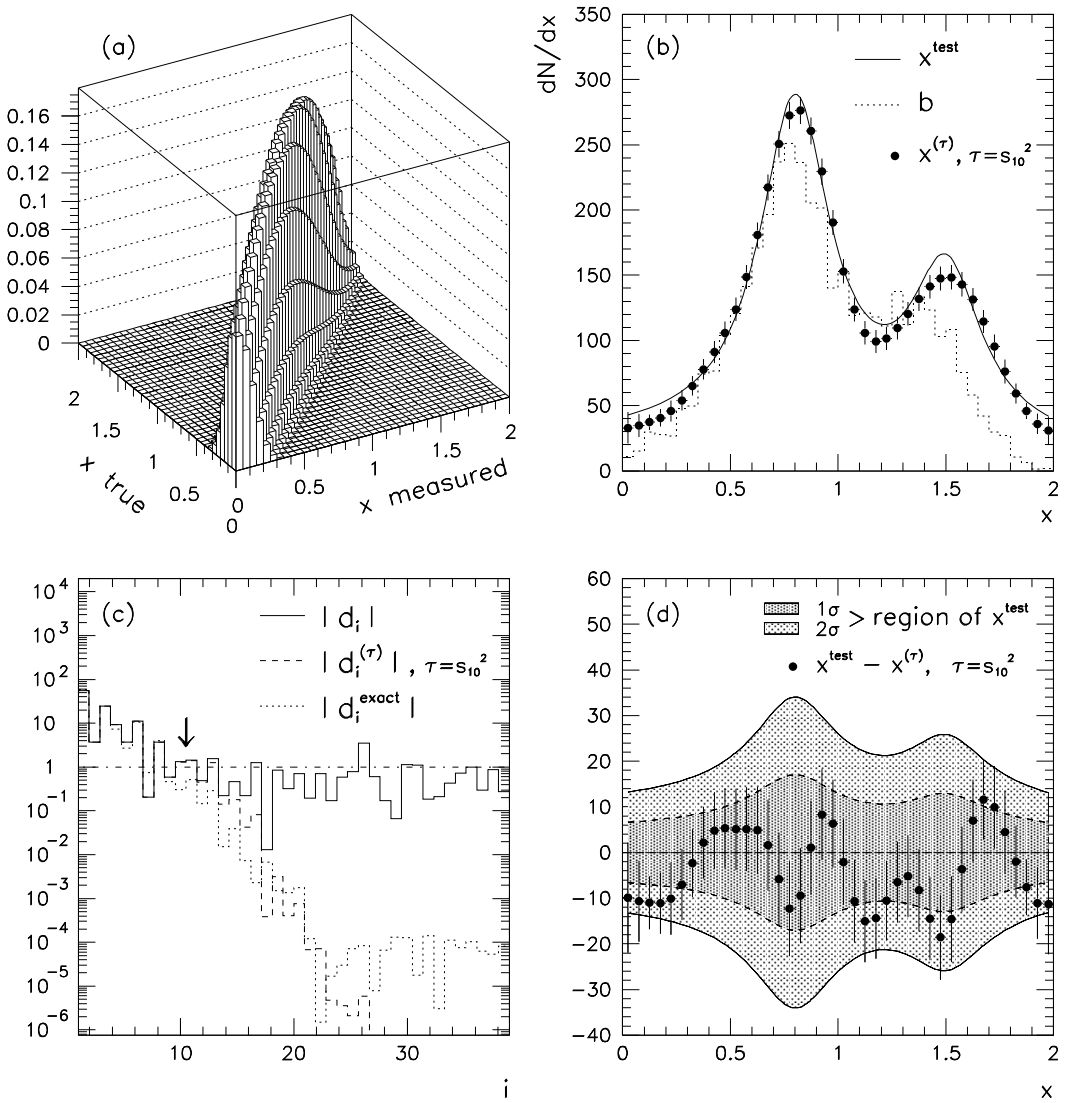
Widely used in HEP

e.g. τ spectral functions, ALEPH, Z.f.Ph. **C76** (1997) 15



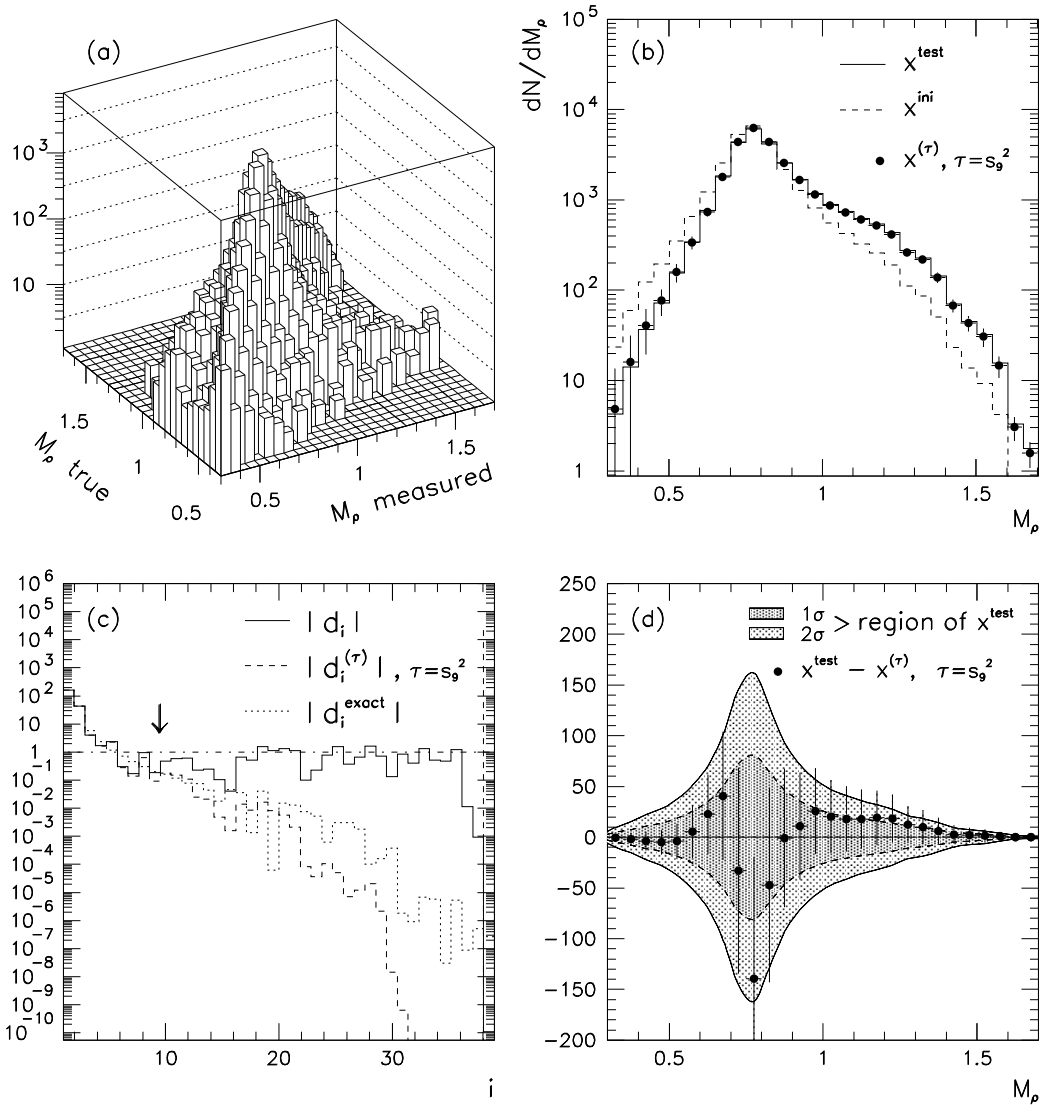
SVD example

A. Höcker, V. Kartvelishvili, NIM A**372** (1996) 469.



SVD example

A. Höcker, V. Kartvelishvili, NIM A**372** (1996) 469.



Regularization function based on entropy (MaxEnt)

Shannon entropy of a set of probabilities is

$$H = -\sum_{i=1}^{M} p_i \log p_i$$

All p_i equal $\rightarrow$ maximum entropy (maximum smoothness)

One $p_i = 1$, all others $= 0$ $\rightarrow$ minimum entropy

Use entropy as regularization function,

$$S(\vec{\mu}) = H(\vec{\mu}) = -\sum_{i=1}^{M} \frac{\mu_i}{\mu_{\rm tot}} \log \frac{\mu_i}{\mu_{\rm tot}}$$

$$\propto \log(\text{number of ways to arrange } \mu_{\rm tot} \text{ entries in } M \text{ bins})$$

Sometimes motivated by Bayesian statistics,

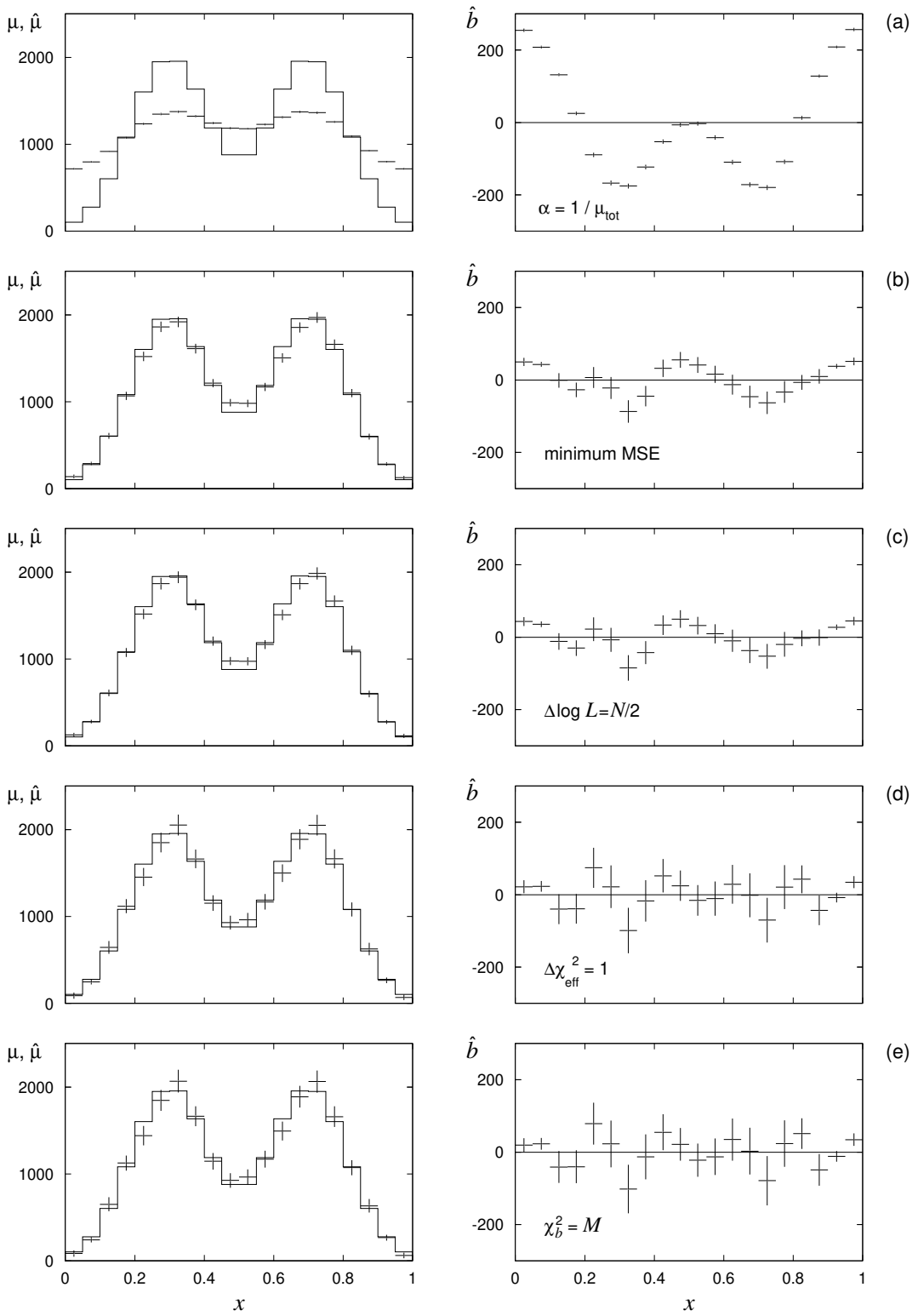
$$S(\vec{\mu}) \rightarrow \text{prior pdf for } \vec{\mu} \quad (?)$$

Here stay with classical approach:

goodness of estimator judged by bias, variance.

N.B. Entropy does not depend on order of bins.

Examples with MaxEnt regularization



Bayesian (iterative) example

G. D'Agostini, NIM **A362** (1995) 487.

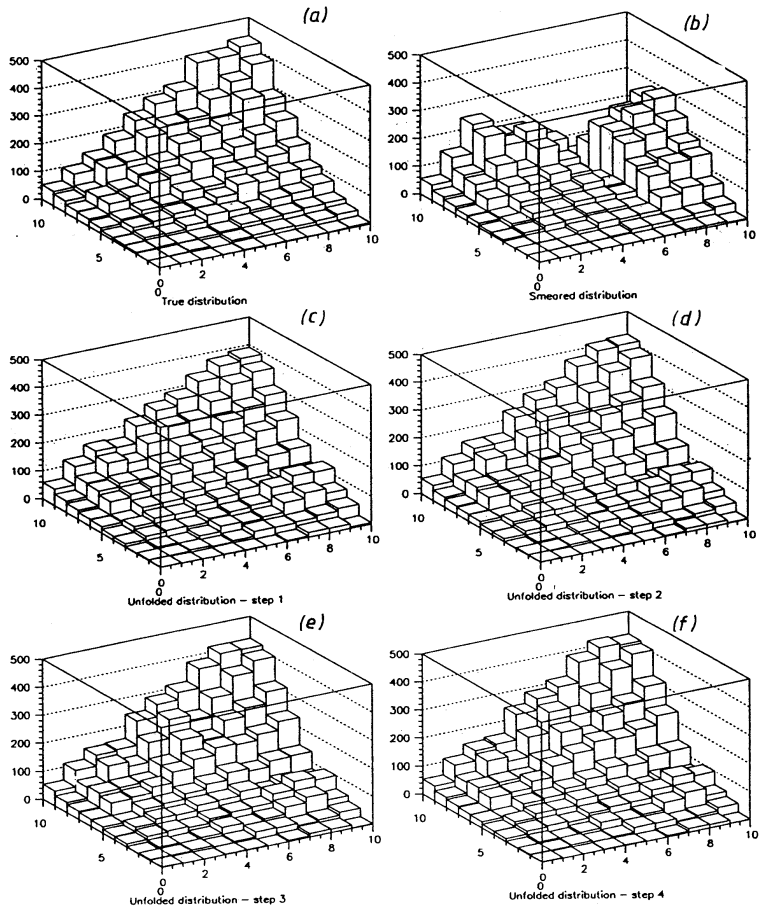


Figure 13: Example of a two dimensional unfolding: true distribution (a), smeared distribution (b) and results after the first 4 steps ((c) to (f)).

Comparison of methods for unfolding $F_2^\gamma(x, Q^2)$

G. Abbiendi *et al.* (OPAL) Eur. Phys. J. C18 (2000) 15-39

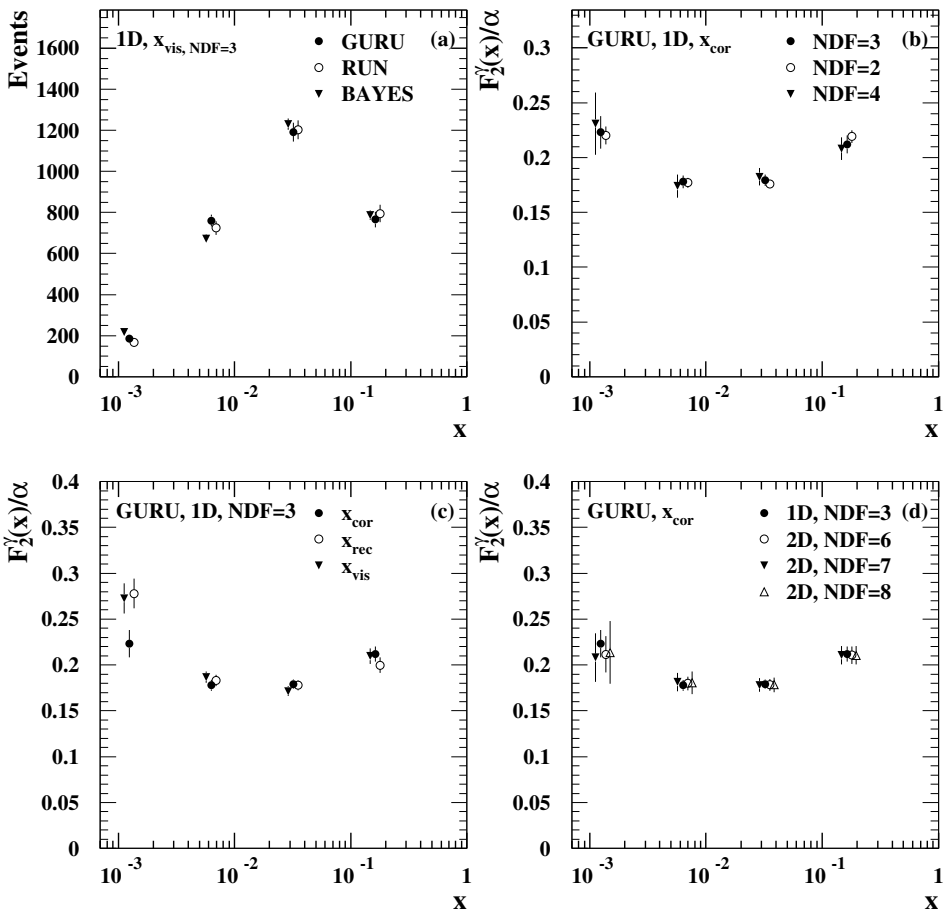
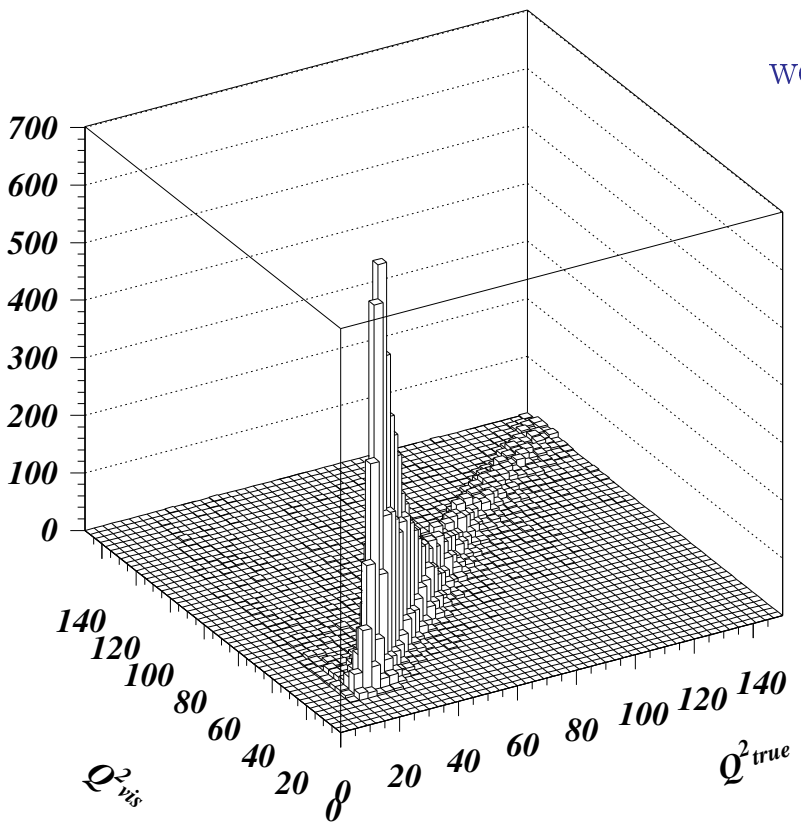


Figure 12: Tests of the unfolding procedure using the LEP1 SW low Q^2 data unfolded with the HERWIG 5.9+k_t(dyn) Monte Carlo sample. (a) Comparison of the results using GURU, RUN and BAYES. (b) Different degrees of freedom for one dimensional unfolding. (c) Different measured x variables for one dimensional unfolding. (d) Different degrees of freedom for two dimensional unfolding using a random number as the second variable. The errors are statistical only.

Measuring x and Q^2 in single-tag $\gamma\gamma$ collisions

$$Q^2 = 2E_{\rm beam}E_{\rm tag}(1 - \cos\theta_{\rm tag})$$

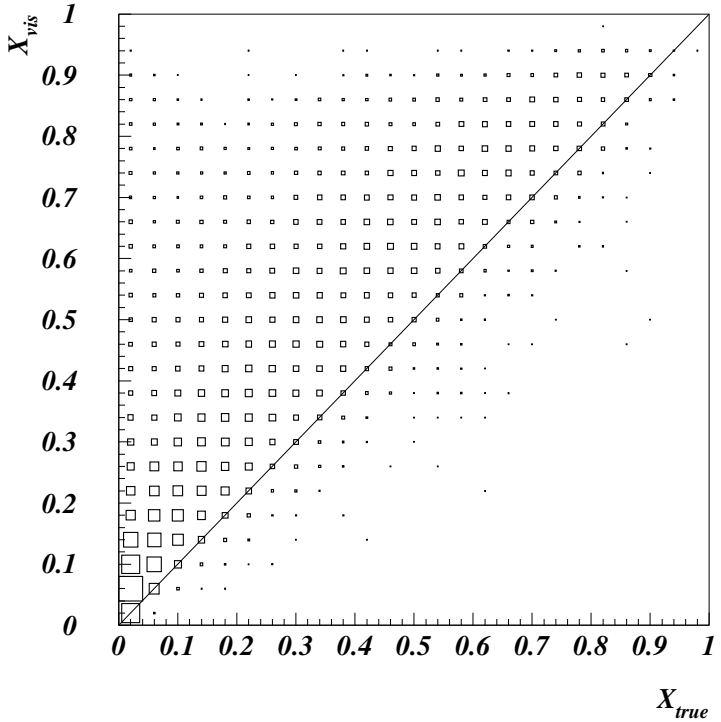
well measured



$$x = \frac{Q^2}{Q^2 + W^2}$$

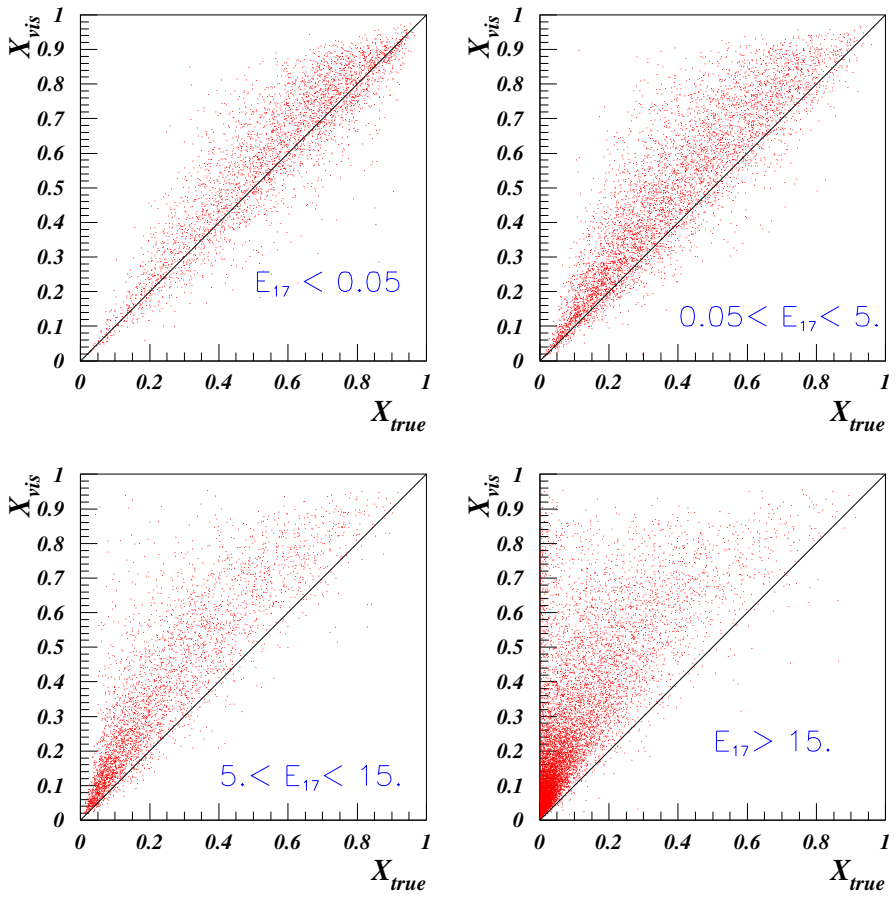
(often) poorly measured

→ unfolding



Model dependence of response matrix

x resolution depends on whether hadrons emitted at low angles:



E_{17} = energy with $\theta < 17°$ wrt beam

Solution: measure x and another variable related to x resolution

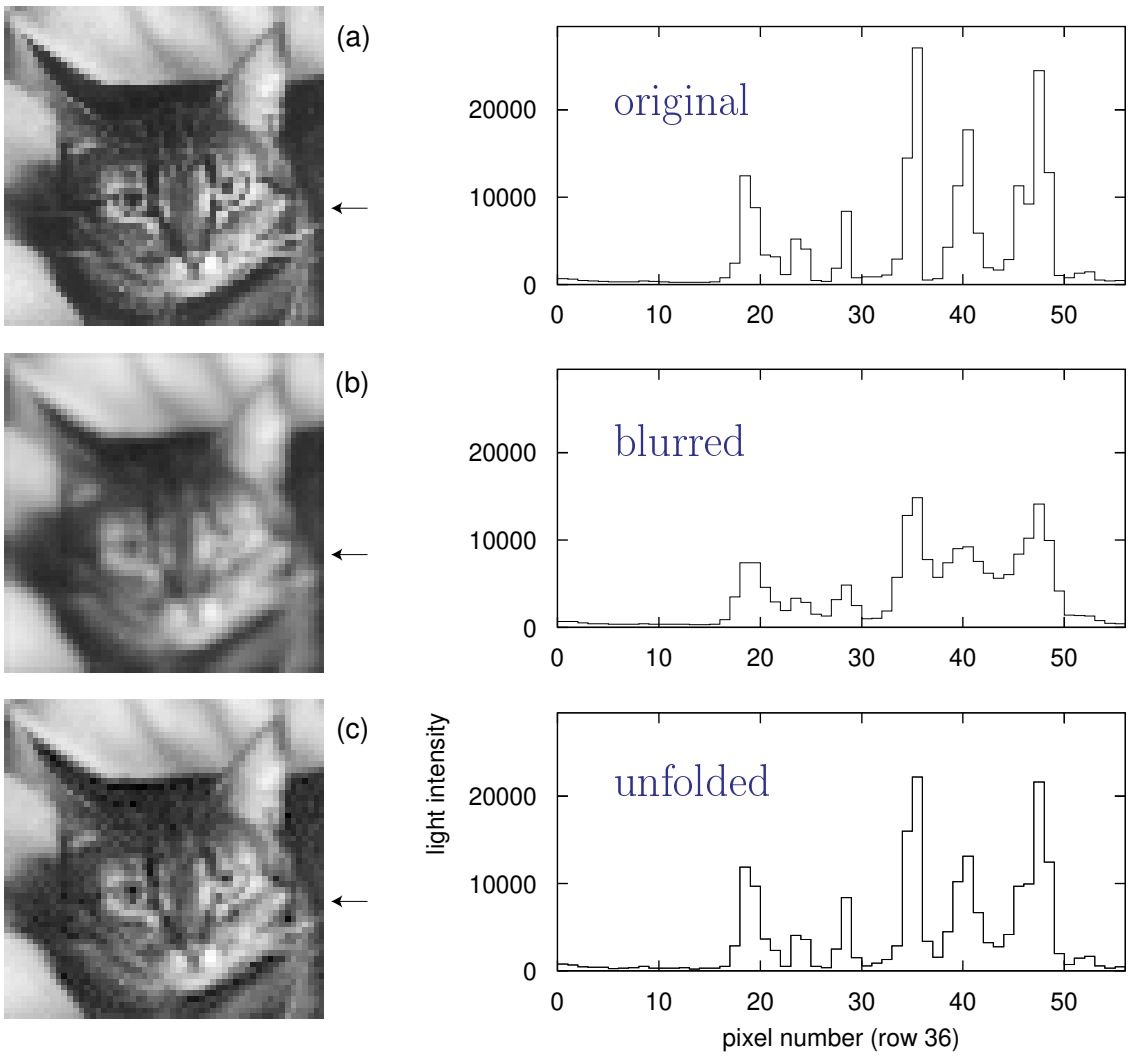
→ two-dimensional unfolding (ALEPH, OPAL)

→ reduces model dependence

→ reduces statistical errors

cf. Affholderbach, Cowan et al., J. Phys. G24 (1998) 457;
G. Abbiendi et al., Eur. Phys. J. C18 (2000) 15-39.

A MaxEnt example with image reconstruction (Newton)



MaxEnt often used in astronomical image reconstruction,

only small bias against point sources (peaks),

easy to generalize to two (or more) dimensions.

MaxEnt for radar traps

S.F. Gull and J. Skilling, IEE Proc., 131F, 646-59 (1984)

Maximum entropy image reconstruction