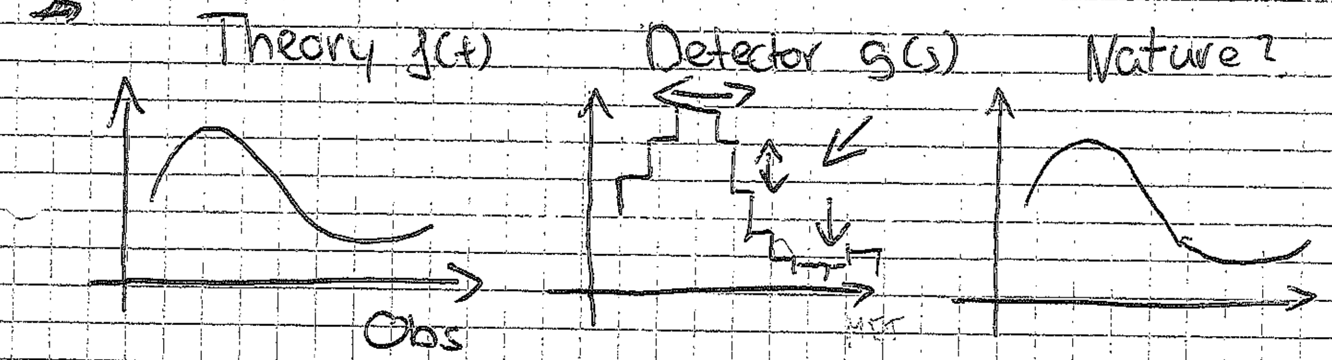
Unfolding

Lecture 1

Introduction

→ Unfolding is at interface of theory & experiment



Possible effects

↓ Limited acceptance & efficiency

⇕ Statistical fluctuations

⟷ Resolution effects

↙ Non-linear response

→ Have to take detector effects into account!

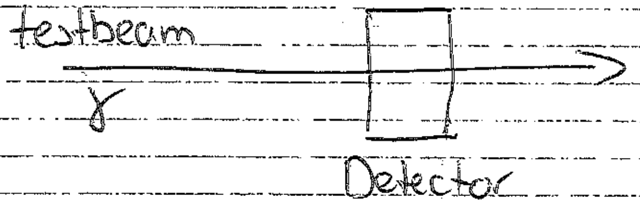
Friedholm-Equation

$$(I)\quad g(s) = \int_{\Omega} A(s,t)\, f(t)\, dt + \underbrace{b(s)}_{\text{background}}$$

→ Kernel $A(s,t)$ contains detector response

How to determine A(s,t) ?

→ "Simple" case: determine experimentally



→ testbeam composition well known

→ Classify detector (e.g. ε)

→ "Complex" case: Simulation

a) Full Geant 4 model

Pros: → Includes dead material, electronics, broken modules, etc

→ Tuned in testbeam measurements

→ Simulates intermediate steps (e.g. showers)

→ Very precise

Cons: → very slow

→ not publicly available

b) Parametric simulation (e.g. DELPHES)

Pros: → Publicly available

→ fast

Cons: → No tuning to experimental data

→ Less precise

→ Higher uncertainties

Ways to account for detector effects

„Applying" the detector to theory (Folding)

→ Process particle level theory sample with full simulation

→ Typically done in ATLAS searches

↳ restricted to two or three benchmark models

→ „Outside" ATLAS : DELPHES

↳ leads to additional uncertainties

„Removing" detector effects from data (Unfolding)

→ Solve I for f(t)

→ Run full detector simulation with high statistics

→ Allows for easier model comparison

→ Improved longevity of data

The „naive" but instructive approach

→ Discretize I

$f(t) \longrightarrow f_j, \; j \in \{1..m\}$

$g(s) \longrightarrow g_i, \; i \in \{1..n\}$

$b(s) \longrightarrow b_i$

$A(s,t) \longrightarrow A_{ij} \in \mathbb{R}^{n \times m}$

(I) $g_i = \sum_{j=1}^{m} A_{ij} f_j + b_i \iff \underline{g} = \underline{\underline{A}}\,\underline{f} + \underline{b}$

Solution: Invert A $\quad f = A^{-1}(g - b)$

a) Simple case

→ Only efficiency effects ε
→ Resolution negligible
→ $n = m$, $b = 0$

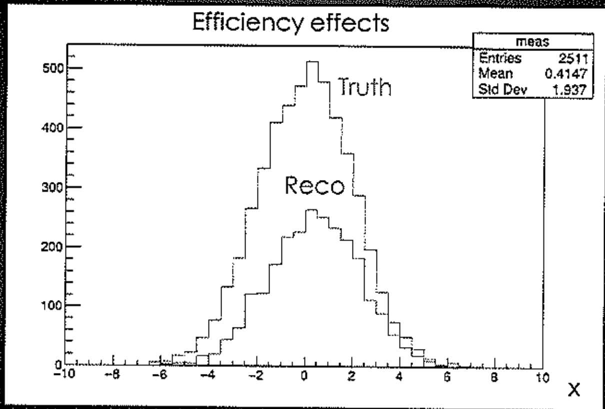
$\Rightarrow \quad A_{ij} = 0$ for $i \neq j$, $\quad A_{ii} = \varepsilon_i$

ε_i efficiency in bin i

$\Rightarrow f_i = \frac{g_i}{\varepsilon_i}$ → Valid result if bins large enough & ε (observable) reasonable

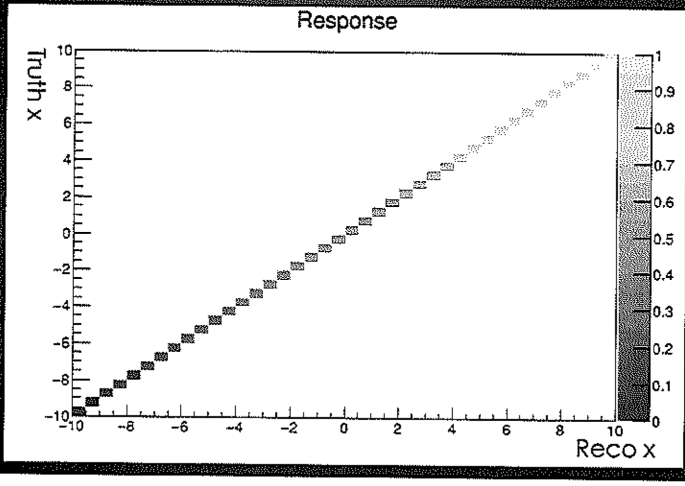
Bin-by-Bin Unfolding

- Truth: Gaussian with $\mu = 0, \sigma = 2$
- Reco: Apply detector efficiency, $\epsilon = (x + 10)/20$
- Assume perfect resolution

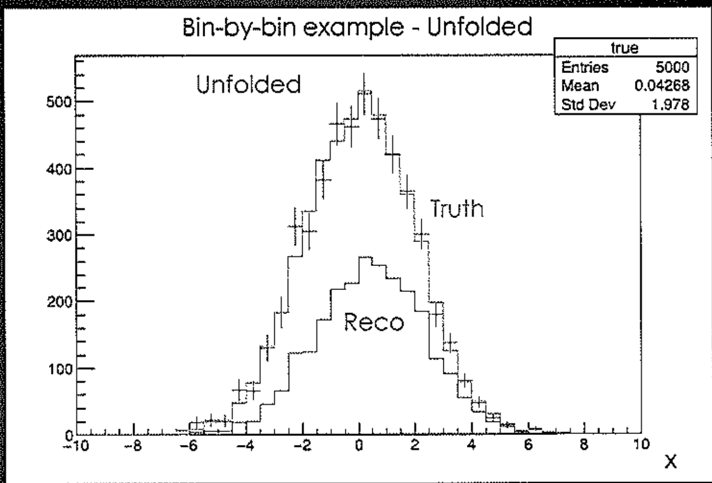


Unfolding Matrix

- Train Unfolding Matrix A with 100000 evts
- Matrix is diagonal, entries correspond to efficiency



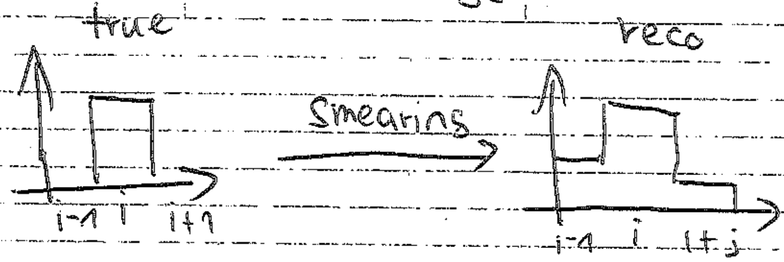
Bin-By-Bin Unfolding



→ Unfolding performs satisfactorily

b) Realistic case

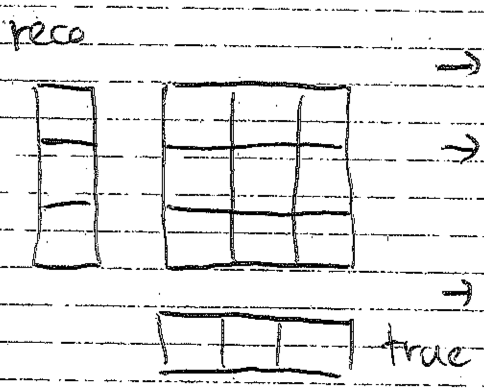
→ Add resolution effect



→ resolution introduces bin migrations ⇒ $A_{ij} \neq 0$ for $i \neq j$

→ $A_{ij} = P_{ij}$ (obs. in bin i | true in bin j)

How to determine A_{ij} from simulation?



→ Process MC events

→ if true in bin i and reco in bin j :

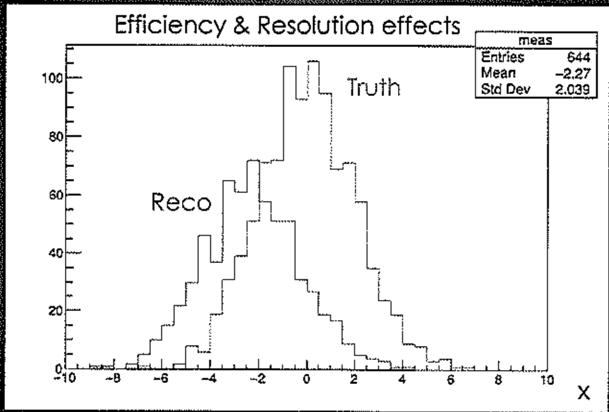
↳ Fill A_{ij}

→ In the end, normalize each column A_i by true value in bin i

Inversion Unfolding

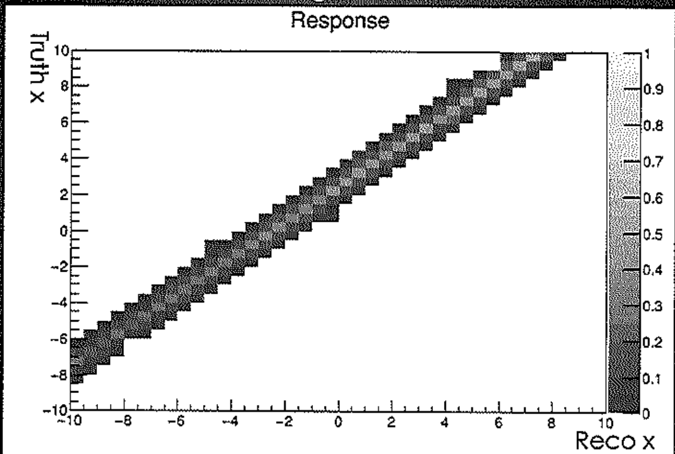
- Truth: Gaussian with $\mu = 0, \sigma = 2$
- Reco: Apply detector efficiency, $\epsilon = 0.3 + \frac{0.7}{20}(x + 10)$

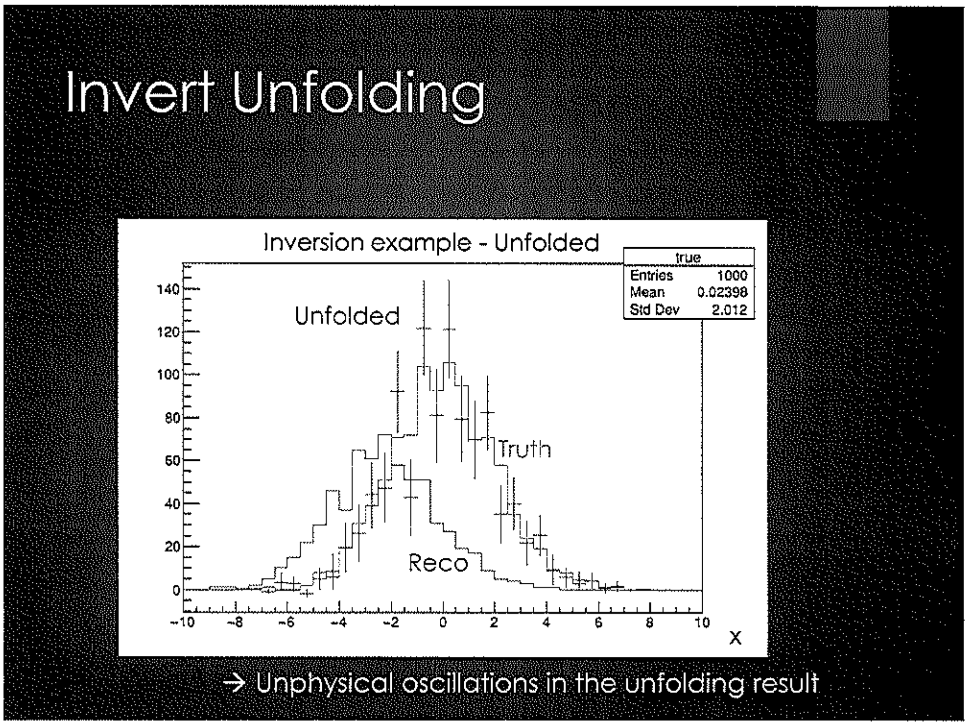
  Gaussian smearing $\mu = -2.5, \sigma = 0.2$



Unfolding Matrix

- Train Unfolding Matrix A with 100000 evts
- Matrix includes bin migrations




Invert Unfolding
Inversion example - Unfolded
true
Entries 1000
Mean 0.02398
Std Dev 2.012
Unfolded
Truth
Reco
x
→ Unphysical oscillations in the unfolding result

Fluctuations due to unfolding by inversion

→ Consider periodic function $f(x)$

(II) $$f(x) = a_0 + \sum_{r=1}^{\infty} a_r \cos rx + b_r \sin rx$$

$a_r, b_r \to 0$ for $r \to \infty$

→ Apply resolution effect (folding)

(III) $$g(y) = \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(y-x)^2}{2\sigma^2}\right) f(x)\, dx$$

→ Plug in single term from II:

$$\to a_r \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi}\,\sigma} \exp\left(-\frac{(y-x)^2}{2\sigma^2}\right) \cos rx \, dx$$

$$= \exp\left(-\frac{r^2\sigma^2}{2}\right) a_r \cos ry \quad \text{(IV)}$$

⇒ Structure is preserved, detector effect results in attenuation of the term by $\exp\left(-\frac{r^2\sigma^2}{2}\right)$

Now assume function $g(y)$ with detector effects as in III

$$g(y) = \alpha_0 + \sum_{r=1}^{\infty} \alpha_r \cos rx + \beta_r \sin rx$$

with gaussian resolution σ as above.

From (IV) we know how to unfold:

$$\alpha_r \cos rx \longrightarrow \exp\left(+\frac{r^2\sigma^2}{2}\right) \alpha_r$$

$$\beta_r \sin rx \longrightarrow \exp\left(\frac{r^2\sigma^2}{2}\right) \beta_r$$

with
- $\alpha_r, \beta_r \to 0$ for $r \to \infty$
- α_r, β_r are measured, have stat. uncertainties

$\Rightarrow$ for $r \to \infty$: α_r, β_r are dominated by stat. uncertainties!

Also $\exp\left(\frac{r^2\sigma^2}{2}\right) \to \infty$ for $r \to \infty$

$\Rightarrow$ Statistical fluctuations are multiplied with exp. factor.

$\Rightarrow$ Unfolded result dominated by stat. fluctuations

II Regularization & Iterative techniques

1. Singular value decomposition

→ Powerful tool to analyze ill-posed problems
→ Decomposes matrix into diagonal and orthogonal part

With $A \in \mathbb{R}^{m \times n}$ $(m > n)$

there exists $U \in \mathbb{R}^{m \times n}$ and $V \in \mathbb{R}^{n \times n}$

such that $U^T U = V^T V = V V^T = \mathbb{1}$

and $\Sigma = \text{diag}(\sigma_1, \ldots, \sigma_n) = U^T A V$

The σ_i are called singular values

with $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_n$

→ $U = (u_1, \ldots, u_n)$, $V = (v_1, \ldots, v_n)$

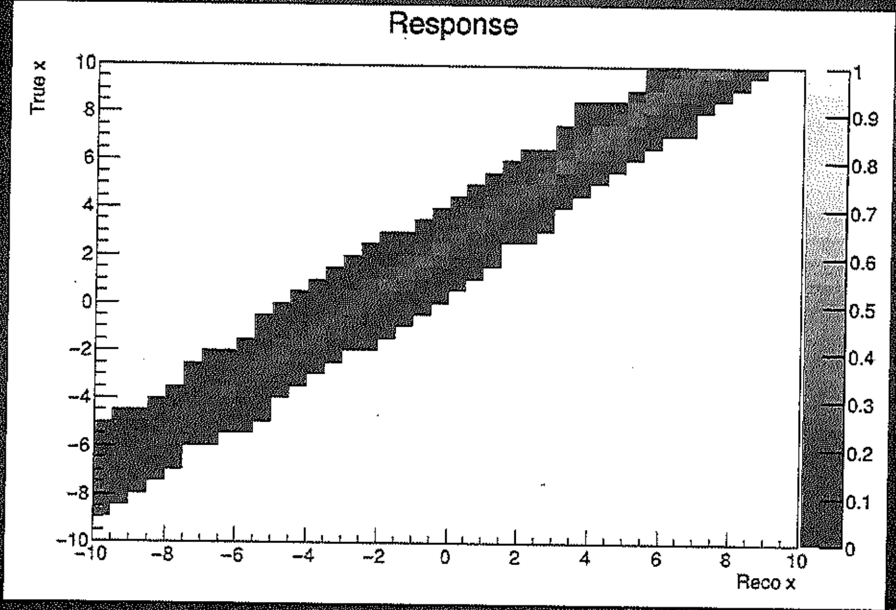
→ The u_i, v_j have increasing sign changes for large i, j.

↳ These correspond to the high frequency modes from the Fourier decomposition in lecture I.

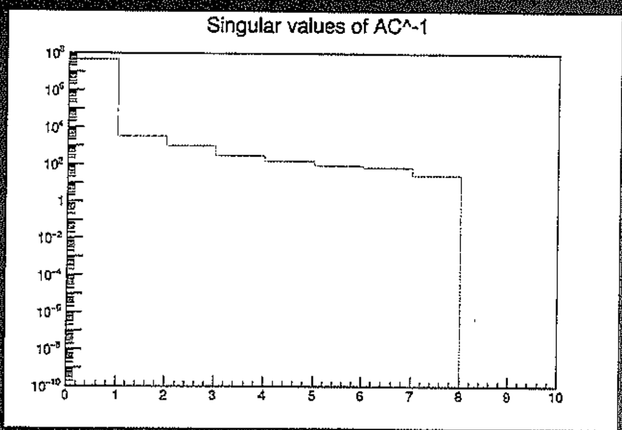
Example from Lecture 1

- Apply detector efficiency, $\epsilon = 0.3 + \frac{0.7}{20}(x + 10)$

  Gaussian smearing $\mu = -2.5, \sigma = 0.5$

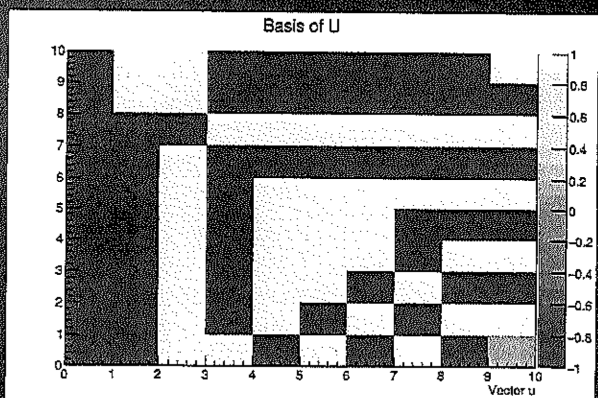


SVD of response matrix



- Reduced binning for visibility
- Singular values decrease
- Singular values span 29 orders

- Basis vectors have more sign changes for larger index!
- Corresponds to higher „frequency"



2.) Unfolding with SVD

To solve (I) $Ax = y$, $A \in \mathbb{R}^{m \times n}$ $(m > n)$

→ Define pseudo-inverse based on SVD

$$A^+ = V \Sigma^{-1} U^T$$

$$\Rightarrow A^+ A = V \Sigma^{-1} U^T U \Sigma V^T = \mathbb{1}$$

→ Apply to I:

$$x = A^+ y = V \Sigma^{-1} U^T y = \sum_{j=1}^{n} \frac{1}{\sigma_j} \underbrace{(u_j^T y)}_{=: c_j} v_j$$

$$= \sum_{j=1}^{n} \frac{c_j}{\sigma_j} v_j =: \sum_{j=1}^{n} d_j v_j$$

→ The σ_j contain information about detector

→ c_j contain information about the data

⇒ This corresponds to the matrix inversion technique from lecture I.

Uncertainties

→ Advantage of inversion technique: Clear prescription for uncertainties

→ For covariance matrix V_y of the data y

$$\Rightarrow \quad V_x = A^+ V_y A^{+T} = V \Sigma^{-2} V^T = \sum_{j=1}^{n} \left(\frac{1}{\sigma_j}\right)^2 v_j v_j^T$$

Effective rank

→ Very small σ_i can lead to large fluctuations in the unfolded spectrum

→ σ_i will not be exactly zero, because rounding errors
↳ may be „effectively" zero within uncertainty

→ Define tolerance δ, $\sigma_p > \delta \geq \sigma_{p+1}$

$$\text{e.g. } \delta = \varepsilon \, \|A\|_\infty$$

where ε is the precision of the MC generator

→ p is called effective rank

The unfolded result can be written as

$$\hat{x} = \underbrace{\sum_{j=1}^{p} d_j v_j}_{\text{meaningful contribution to result}} + \underbrace{\sum_{j=p+1}^{n} d_j v_j}_{\text{„arbitrary" contributions, belongs to null space of } A}$$

→ Simplest solution: Cut off terms with $j > p$

↳ Can lead to new fluctuations (Gibbs-Phenomenon)

→ Better way: regularization

Regularization

→ Goal: Avoid cut-off by slow dampins of insignificant singular values.

→ SVD Unfolding: $\hat{x} = V \Sigma^{-1} U^T y = A^+ y$

introduce regularisation →

$$\hat{x} = V \underbrace{[(\Sigma^2 + \tau \mathbb{1})^{-1} \Sigma^2]}_{=: F} \Sigma^{-1} U^T y =: A^{\#} y$$

filter factor matrix

→ F is diagonal $F = \begin{pmatrix} \varphi_1 & & \\ & \ddots & \\ & & \varphi_n \end{pmatrix}$ $\varphi_j = \frac{\sigma_j^2}{\sigma_j^2 + \tau}$

→ τ is called „regularisation parameter"

→ For $\tau \to 0$: $F \to \mathbb{1}$

The unfolded result becomes:

$$\hat{x} = \sum_{i=1}^{n} \frac{c_i}{\sigma_i} \varphi_i v_i$$

↑ dampening of small σ_i

The variance: $V_x = \sum_{i=1}^{n} \frac{1}{\sigma_i^2} \varphi_i^2 v_i v_i^T$

Choice of τ

→ General statement difficult, case-dependent

→ Typical choice: $\tau = \sigma_k^2$, for particular k

↳ Onset of dampening for σ_j with $j > k$

→ Always need to check χ^2 between y and $\hat{y}$

→ Further checks: curvature, correlation of V_x, ...

Other possibility: $\varphi_i = \dfrac{1}{\left(1 + \left(\frac{\tau}{\sigma_i^2}\right)^{\alpha}\right)}$, $\alpha > 1$

Regularisation bias

→ We have replaced A^+ by $A^\#$

$\hat{x} = A^\# y$, with $y = A x_{exact} + e$

$\Rightarrow \hat{x} = A^\#(A x_{exact} + e) = x_{exact} + \underbrace{(A^\# A - \mathbb{1}) x_{exact}}_{\text{Systematic bias!}} + \underbrace{A^\# e}_{\text{Stat. error}}$

$\Rightarrow$ Need to check agreement between y and $\hat{y} = A\hat{x} = AA^\# y$

Iterative techniques

Bayesian unfolding

→ SVD and similar techniques based on inversion of matrix A

$$x = A^{-1} y$$

→ Inversion responsible for associated difficulties

→ Different approach: Elements of A are probabilities

$$A = \begin{pmatrix} \ddots & & \\ & \ddots & \\ P_{ij} & & \ddots \end{pmatrix}$$

$$P_{ij} = P(\text{reco in bin } j \mid \text{truth in bin } i) = P(R_j \mid T_i)$$

→ Matrix inversion makes no sense in probability theory

→ Use Bayes theorem

$$P(B \mid A) = \frac{P(A \mid B)\, P(B)}{P(A)}$$

Generalized:
$$P(B_i \mid A) = \frac{P(A \mid B_i)\, P(B_i)}{\sum_n P(A \mid B_n)\, P(B_n)}$$

→ For unfolding: Need P(truth in bin i | reco in bin j)

Response matrix A_{ij} ; Prior

$$\Rightarrow \quad P(T_i | R_j) = \frac{\overbrace{P(R_j | T_i)}^{\text{Response matrix } A_{ij}} \overbrace{P_0(T_i)}^{\text{Prior}}}{\sum_{k=1}^{n} P(R_j | T_k) P_0(T_k)}$$

→ Normalization: $\sum_{i=1}^{n} P_0(T_i) = 1$

$\sum_{i=1}^{n} P(T_i | R_j) = 1$ → for each reco event there is a truth bin

→ Efficiency $0 \leq \varepsilon_i = \sum_{j=1}^{m} P(R_j | T_i)$

→ not each truth event is necessarily reconstructed

→ Prior $P_0(T_i)$: Use best possible knowledge
↳ in worst case: flat prior

How to unfold?

Measured events: $n(R) = \{ n(R_1), n(R_2), \ldots, n(R_m) \}$

Unfolded events: $\hat{n}(T) = \{ \hat{n}(T_1), \hat{n}(T_2), \ldots, \hat{n}(T_n) \}$

$$\Rightarrow \hat{n}(T_i)|_{obs} = \sum_{j=1}^{m} n(R_j)\, P(T_i | R_j)$$

→ Need to accont for inefficiencies

$$\hat{n}(T_i) = \frac{1}{\varepsilon_i} \sum_{j=1}^{m} n(R_j)\, P(T_i | R_j) \quad (\mathrm{I})$$

→ Estimate true total number of events $\hat{N}_{true} = \sum_{i=1}^{n} \hat{n}(T_i)$ (II)

→ Estimate final probability $\hat{P}(T_i) = \frac{\hat{n}(T_i)}{\hat{N}_{true}}$ (III)

What about the influence of the prior?

→ If the prior does not describe the real truth the unfolding agreement will be bad (especially flat prior)

Solution: Iterative approach with update of $P_0(C)$

1.) Initial guess for $P_0(T)$

2.) Use I, II, III to calculate $\hat{n}(T)$, $\hat{P}(T)$

3.) Make χ^2 comparison of $\hat{n}(T)$ and $n_0(T)$

4.) Replace $P_0(T)$ by $\hat{P}(T)$ and $n_0(T)$ by $\hat{n}(T)$ and start again. If χ^2 small enough stop, otherwise → 2.)

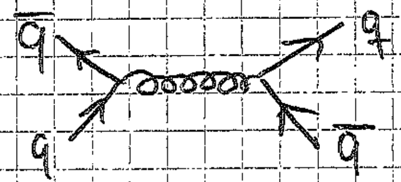
III Unfolding searches

→ So far we have assumed good data – MC agreement (measurement)

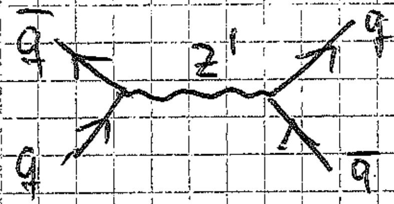
→ Now: Assume additional structures in the data, not described by MC (searches)

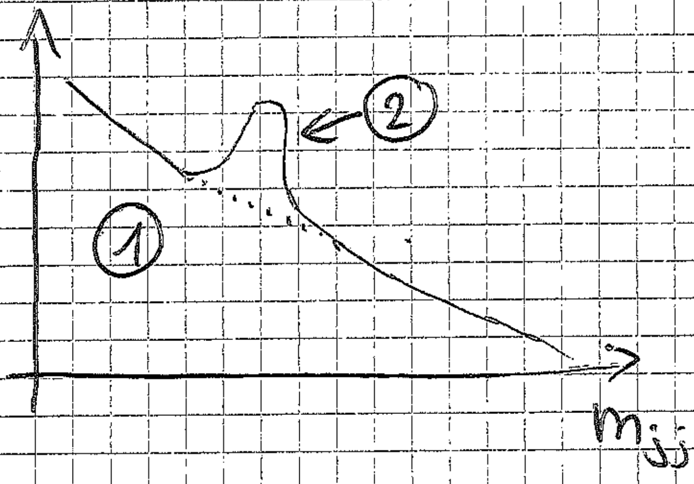
Example: Dijet resonance search

What we know is there: (1)



What might be there: (2)





→ In absence of clear signal $m_{Z'}$ unknown,

→ Only (1) can be used to train detector response matrix A

→ Have to make sure that structures from (2) are preserved in unfolding!

Iterative Dynamically Stabilized (IDS) Unfolding

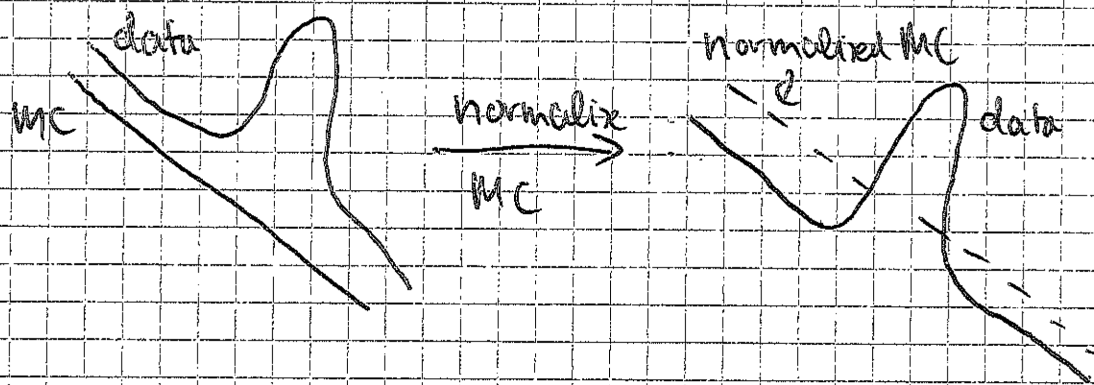
→ „State-of-the-art" unfolding (2009)

→ Designed to deal with structures in data not simulated in MC

→ Takes background fluctuations into account
→ Allows control of correlations

Design goals:

① → Avoid distorted normalization due to additional data



② → Avoid „pollution" of unfolded spectra with events from regions with high uncertainty

↳ e.g. MC background subtraction will introduce uncertainties on data

↳ "Uncertainties can „migrate" to more precise regions of spectrum

③ → Minimize correlations introduced to the data due to the unfolding

⇒ <u>Solution</u>: Do not perform maximum event transfer, but only <u>a part of it.</u>

<u>Regularization function</u>

→ Provides information on the significance of absolute deviations between data and MC

→ Several possibilities

$$\text{e.g. } f_1 = \mathrm{Erf}\left(\frac{\Delta x}{\sqrt{2}\,\lambda\,\sigma}\right)$$

Here λ is called regularization parameter

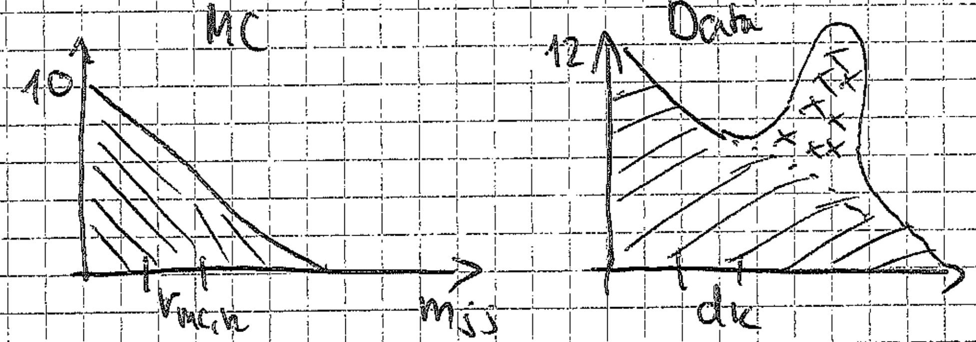
Desired behaviour: $\Delta x \to 0 \quad f \to 0$

$\Delta x \gg \sigma \quad f \to 1$

Iterative improvements

MC normalization

→ Find # events in data that correspond to simulated structures in MC



(hatched) $= N_{MC}$ (hatched) $= N_{sMC}$ (hatched) + (crosses) $= N_{Data}$

Iterate to find correct N_{sMC}:

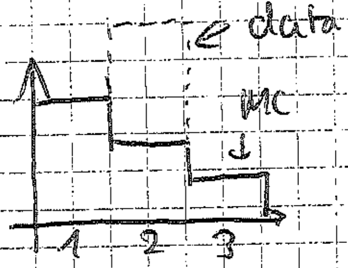
1.) $N_{sMC} = N_{Data}$

2.) $N_{sMC}' = N_{sMC} + \sum_{k=1}^{n} \left(1 - f(\Delta d_k, \sigma_k, \lambda_N)\right) \Delta d_k$

$$\Delta d_k = d_k - \underbrace{\frac{N_{sMC}}{N_{MC}} \cdot r_{MC,k}}_{\text{Best guess for MC normalized to data}}$$

→ $(1 - f(\Delta d_k, \sigma_k, \lambda_N))$ suppresses contributions from significant data–MC differences

Example



$\frac{N_{Data}}{N_{MC}} > 1$

→ perfect agreement in bins 1 & 3 ($d_1 = r_{MC1}$, $d_3 = r_{MC3}$ / $d_2 >> r_{MC2}$)

First iteration

$N_{DMC} = N_{Data}$

$$\Delta d_{1,3} = d_{1,3} - \underbrace{\frac{N_{DMC}}{N_{MC}}}_{>1} r_{MC1,3} < 0$$

$$\Delta d_2 = d_2 - \frac{N_{DMC}}{N_{MC}} r_{MC2} > 0$$

$$\sum \Delta d_i = 0$$

$$\sum (1-f_i) \Delta d_i = \underbrace{(1-f_1)}_{=1} \Delta d_1 + \underbrace{(1-f_2)}_{<1} \Delta d_2 + \underbrace{(1-f_3)}_{=1} \Delta d_3$$

$$< 0$$

$\Rightarrow N_{DMC}' < N_{DMC}$

→ Improved normalization

Update of Matrix

→ Similar to Bayesian matrix update

→ Corresponds to update of prior

Here: A_{ij} is response matrix (not normalized), determined from MC

P_{ij} is probability matrix

$$A_{ij} = \epsilon_j \cdot P_{ij}$$

Iterate:

$$A_{ij}' = A_{ij} + f(|\Delta U_j|, \sigma_j, \lambda_m)\, \Delta U_j\, P_{ij}$$

$$A_{ij}' = \underbrace{(\epsilon_j + f(|\Delta U_j|, \sigma_j, \lambda_m)\, \Delta U_j)}_{\text{corresponds to new truth prior.}}\, P_{ij}$$

$$\Delta U_j = \frac{N_{MC}}{N_{DMC}} \cdot U_j - t_{M,j}$$

(U_j ↑ unfolded result; $t_{M,j}$ ↑ truth MC.)

IDS Unfolding procedure

Unfold via:

$$U_j = t_{MC,j} \cdot \frac{N_{DMC}}{N_{MC}} + \sum_{k=1}^{n} \begin{cases} f(|\Delta d_k|, \sigma_k, \lambda)\, \Delta d_k \cdot \hat{P}_{kj} + (1 - f(|\Delta d_k|, \sigma_k, \lambda))\, \Delta d_k \delta_{kj} & \text{, if } r_{MC,k} \neq 0 \\ \Delta d_k \cdot \delta_{kj} & \text{, if } r_{MC,k} = 0 \end{cases}$$

<u>Here:</u> $\hat{P}_{kj}$ is the unfolding matrix defined in the Bayesian sense (see lecture II)

→ Structures that are well described by MC are just „copied" $\left(\hat{=}\ t_{MC,j} \cdot \frac{N_{DMC}}{N_{MC}} \right)$

→ Only differences between data and reco MC are unfolded (Δd_k)

→ Strength of the unfolding controlled by λ