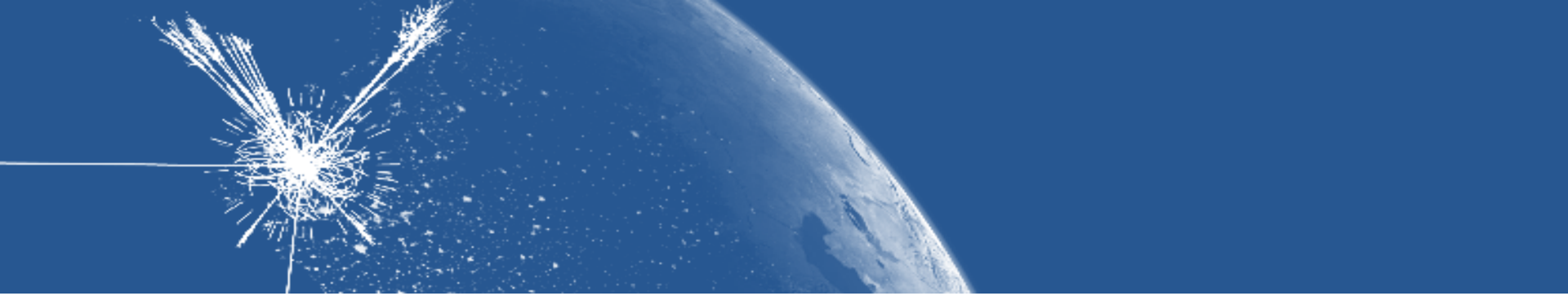
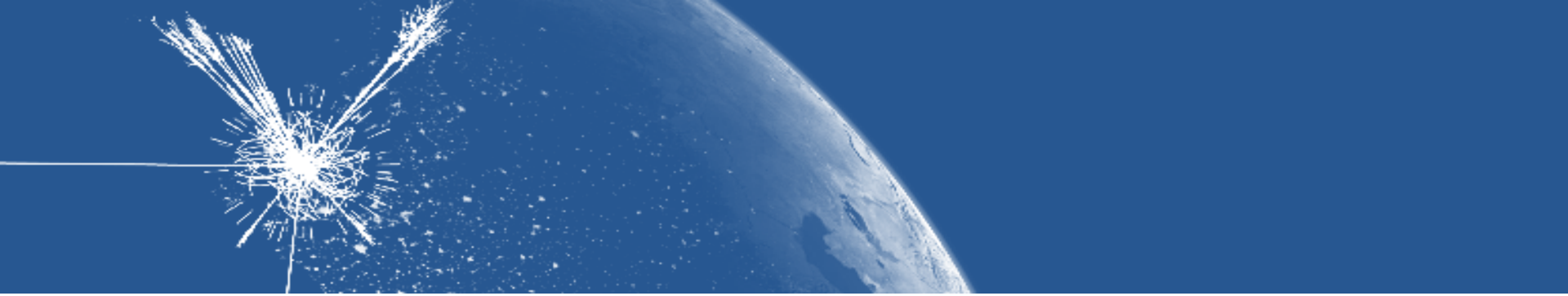


Introduction to probability and statistics (3)

Andreas Hoecker (CERN)

CERN Summer Student Lecture, 18–21 July 2016



Outline (4 lectures)

1st lecture:

- Introduction
- Probability

2nd lecture:

- Probability axioms and hypothesis testing
- Parameter estimation
- Confidence levels

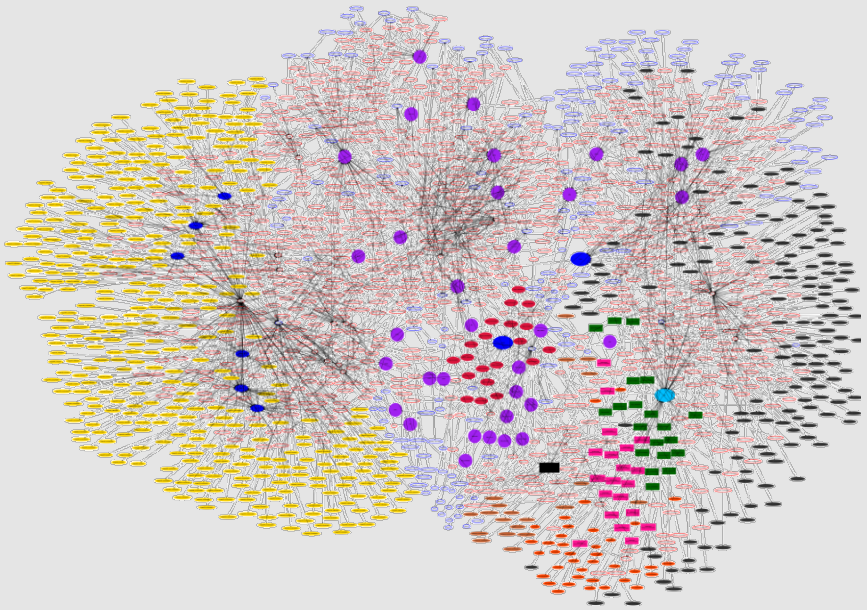
3rd lecture:

- Maximum likelihood fits
- Monte Carlo methods
- Data unfolding

4th lecture:

- Multivariate techniques and machine learning

Maximum likelihood fits



Likelihood functions (reminder)

The *likelihood function* for a simple counting experiment is given by the Poisson PDFs:

$$L(\text{data}(N_{\text{obs}})|\mu) = \frac{(\mu S + B)^{N_{\text{obs}}}}{N_{\text{obs}}!} \cdot e^{-(\mu S+B)}$$, where:

N_{obs}	observed number of events
S	expected number of signal events
B	expected number of background events
μ	"signal strength" modifier

In an unbinned case, the relevant likelihood function for N_{events} events reads:

$$L(\text{data}|\mu) = e^{-(\mu S+B)} \cdot \prod_{i=1}^{N_{\text{events}}} \left(\mu S \cdot p_s(x_i) + B \cdot p_b(x_i)\right)$$

where $p_s(x_i)$ and $p_b(x_i)$ are the values of the signal and background PDFs for the variable x_i

Likelihood functions with nuisance parameters

If the background prediction is subject to an uncertainty, one adds a *nuisance parameter* $\boldsymbol{\theta}$:

$$L(N_{\text{obs}},\mu,\theta) = \frac{(\mu S + \theta B)^{N_{\text{obs}}}}{N_{\text{obs}}!} e^{-(\mu S+\theta B)} \cdot \text{Gauss}(\theta - 1, \sigma_\theta)$$

which is (in this example) constrained to $\boldsymbol{\theta = 1}$ within $\boldsymbol{\sigma_\theta}$ by a Gaussian PDF

The profile likelihood function is maximised with respect to both μ and θ

In realistic use cases, $L(N_{\text{obs}},\mu,\theta)$ can be more complex:

- Both signal and background predictions are subject to multiple uncertainties parametrised by a set of m nuisance parameters $\theta = \{\theta_1, \dots, \theta_m\}$
- There are several distinct signal and background contributions
- Several signal and background *control regions* are simultaneously fit
- The parameter of interests may not only be event abundances, but also signal properties
- Likelihood may be split into categories with different subpopulations of events with common and non-common parameters

One-sided test statistics

To compare the compatibility of the data with the background-only and signal+background hypotheses, where the signal is allowed to be scaled by some factor μ, we construct the following test statistic based on the profile likelihood ratio:

$$\tilde{q}_\mu = -2 \cdot \ln \frac{L\left(\text{data}|\mu, \hat{\theta}_\mu\right)}{L\left(\text{data}|\hat{\mu}, \hat{\theta}\right)}, \qquad 0 < \hat{\mu} < \mu$$

(Condition enforces one-sided confidence intervals for discovery and upper limit tests)

where nominator and denominator are independently maximised.

$\hat{\theta}_\mu$ is the *conditional* maximum given the signal strength modifier value μ

$\hat{\mu}, \hat{\theta}$ are the values corresponding to the global maximum of the likelihood

Remarks:

- Large $\tilde{q}_\mu$ values correspond to disagreement between data and hypothesis μ.
- $\tilde{q}_\mu$ behaves as χ^2 for large data samples and Gaussian θ parameters
- Note that the denominator in $\tilde{q}_\mu$ is independent of μ and only a normalisation term

Frequentist limit setting procedure

See: ATLAS & CMS https://cds.cern.ch/record/1375842

1. Construct likelihood function $L(\mu, \theta)$

2. Construct test statistics $\tilde{q}_\mu$

3. Perform fits on data and determine observed $\tilde{q}_{\mu,\mathrm{obs}}$ for hypothesis μ

4. Generate pseudo Monte Carlo events to construct the PDF $p_\mu\left(\tilde{q}_\mu|\mu, \hat{\theta}_{\mu,\mathrm{obs}}\right)$ of $\tilde{q}_\mu$
   (for hypothesis μ, and where $\hat{\theta}_{\mu,\mathrm{obs}}$ is the set of conditional nuisance parameters found in fit to data).
   The nuisance parameters are fixed to $\hat{\theta}_{\mu,\mathrm{obs}}$ for the MC generation, but allowed to float in the fits.
   In the asymptotic limit, $p_\mu(\tilde{q}_\mu|\mu, \theta)$ is independent of θ.

5. Determine the observed p-value for hypothesis μ: $P(\mu) = \int_{\tilde{q}_{\mu,\mathrm{obs}}}^{\infty} p_\mu\left(\tilde{q}_\mu|\mu, \hat{\theta}_{\mu,\mathrm{obs}}\right) d\tilde{q}_\mu$

6. Perform “discovery” test by computing $P(\mu = 0)$

7. Find the 95% upper bound $\mu = \mu_{95,\mathrm{obs}}$ for which: $P(\mu) = 0.05$

In case of complex fits the pseudo-MC procedure can be very CPU intensive.
Fortunately, asymptotic formulas exist that quite accurately reproduce the exact results.

Cowan, Cranmer, Gross, Vitells: https://arxiv.org/abs/1007.1727

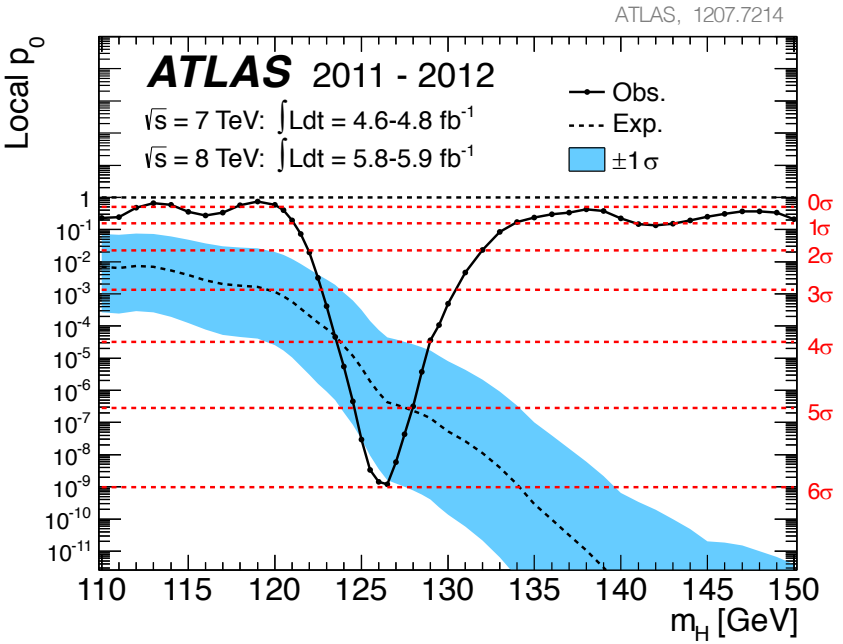
Frequentist limit setting procedure (continued)

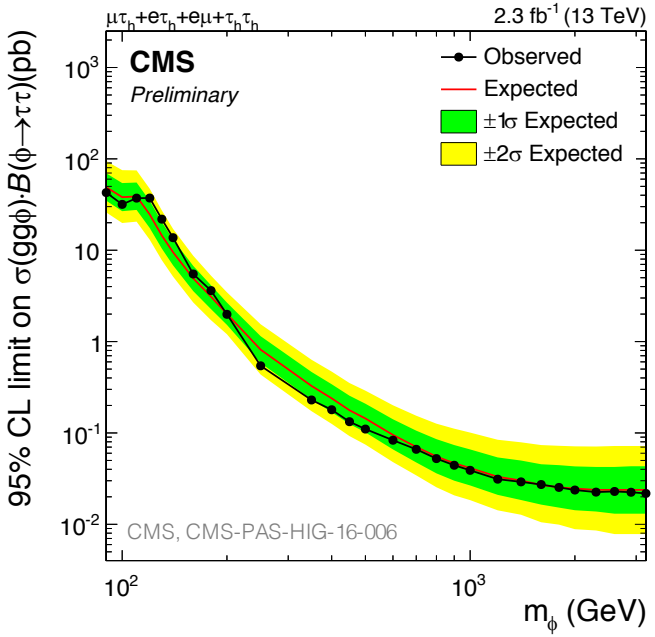
A. Read, https://cds.cern.ch/record/451614

To be more conservative (to avoid that upward fluctuations of background contribute to the p-value), the LHC experiments compute upper limits using: $P_{\mathrm{CL}_s}(\mu) = P(\mu)/P(0) = 0.05$

- CL_s usually *over-covers*, ie, less than 5% of repeated experiments would lie outside the given bound
- A property of CL_s is that in case of $N_{\mathrm{obs}} = 0$, the resulting 95% CL upper limit is $\mu_{95,\mathrm{obs}} S \cong 3$, independent of the background expectation and the nuisance parameters

Let's get back to our earlier discovery and limit plots:





Frequentist limit setting procedure (continued)

The underlying fits are really complex. On the right a graph of *only* the $H \to \gamma\gamma$ likelihood model:

The ATLAS & CMS Run-1 Higgs coupling combination analysis comprises a total of 4200 nuisance parameters ! (Of which a large fraction is of statistical nature)

ATLAS & CMS http://arxiv.org/abs/1606.02266

The tool of choice to perform such complex likelihood fits is **RooFit** (contained in ROOT)

http://roofit.sourceforge.net

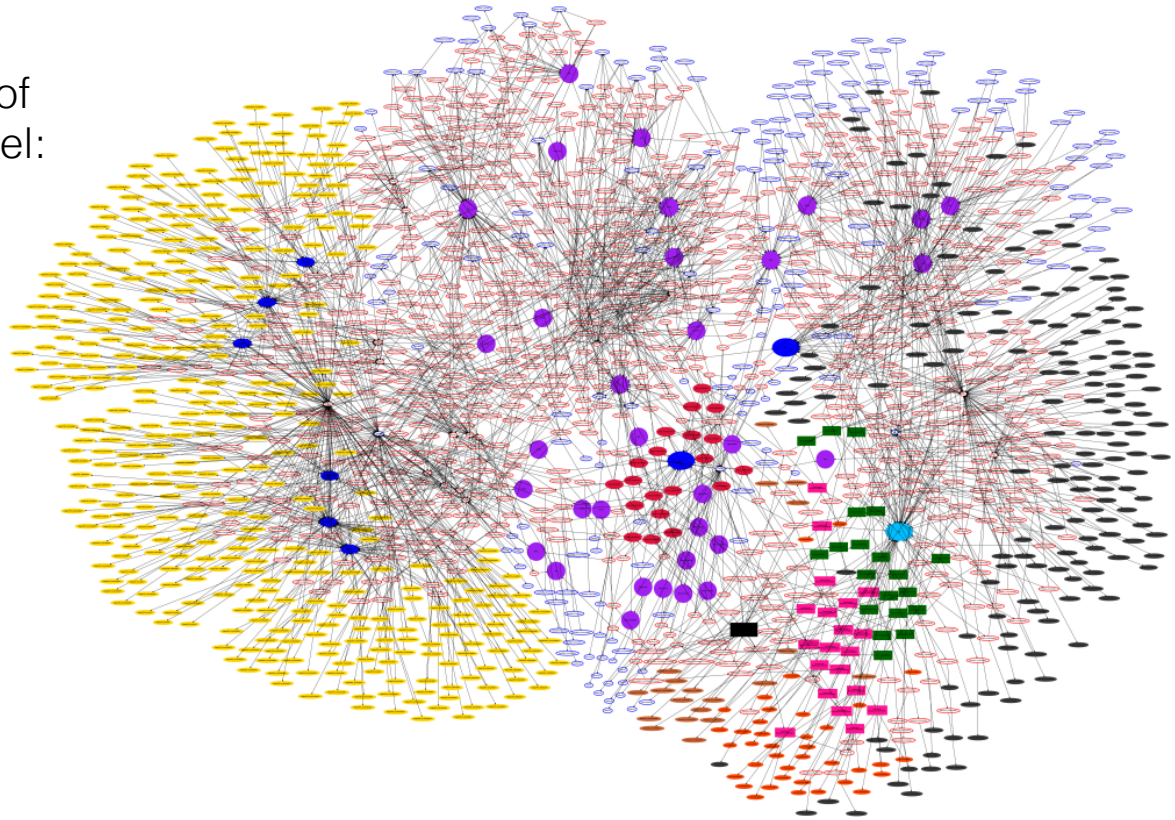


Figure caption: Each node represents either a numerical value, an expression or a PDF. The black box is the top- level PDF, the green boxes are the signal PDFs for each category, the pink boxes are the background PDFs. The bottom part of the graph describes the background: the brown ellipses are the background normalization parameters, while the orange ellipses are the shape parameters. The dark red ellipses are the signal normalization expressions, and the blue ellipse in the center represents the μ parameter. The left part of the graph is devoted to the parameterization of SM signal yields: the gold ellipses are the coefficients of the parameterization, while the blue ellipses are per-mode μ parameters. The right side of the plot describes the signal shape: the dark gray boxes are the signal shape parameters, the blue ellipse represents m_H, and the cyan ellipse is $m_{\gamma\gamma}$. Finally, the purple ellipses represent the nuisance parameters associated with systematic uncertainties, the white boxes with blue outlines are the parameters describing the uncertainties. The red-lined boxes are expressions that bind the model together.

Nicolas Berger, Habilitation thesis, 2016

Why 5σ for a discovery ?

See also G. Cowan, https://arxiv.org/abs/1307.2487

As we have discussed yesterday, it is common practice in particle physics to regard an observed signal a "discovery" when its significance exceeds $Z = 5$, corresponding to a one-sided p-value of the background-only hypothesis of $2.9 \cdot 10^{-7}$

This is in contrast to many other fields (e.g., medicine, psychology) where a p-value of 5% ($Z = 1.64$) may be considered significant

Discoveries of new particles have been relatively frequent during the last ~20 years in the low-energy hadron spectra, but are very rare at high energy

Certainly, from Bayesian reasoning: "*extraordinary claims require extraordinary evidence*"

A discovery (beyond the SM) will be a game changer that we do not want to have to unsay

Another reason for the high Z is the influence of non-statistical systematic uncertainties in some of our particle searches, which alter the properties of the p-value found

Finally, and importantly, the large look-elsewhere-effect (LEE) is a source of fluctuations. While it can be accounted for in a given analysis, the LEE is a global phenomenon that affects the entirety of the searches: the probability of seeing a fluctuation with local $Z = 5$ *anywhere* is much larger than $2.9 \cdot 10^{-7}$!

Why 5σ for a discovery ?

See also G. Cowan, https://arxiv.org/abs/1307.2487

As we have discussed yesterday, it is common practice in particle physics to regard an observed signal a "discovery" when its significance exceeds $Z = 5$, corresponding to a one-sided p-value of the background-only hypothesis of $2.9 \cdot 10^{-7}$

This is in contrast to many other fields (e.g., medicine, psychology) where a p-value of 5%

Note: a discovery requires more than a "5σ" value. It needs the judgement of the scientist that the question asked and the experimental setup used are meaningful, that systematic uncertainties are under control, and that the analysis and interpretation were performed in an unbiased manner.

Another reason for the high Z is the influence of non-statistical systematic uncertainties in some of our particle searches, which alter the properties of the p-value found

Finally, and importantly, the large look-elsewhere-effect (LEE) is a source of fluctuations. While it can be accounted for in a given analysis, the LEE is a global phenomenon that affects the entirety of the searches: the probability of seeing a fluctuation with local $Z = 5$ *anywhere* is much larger than $2.9 \cdot 10^{-7}$!

Monte Carlo techniques

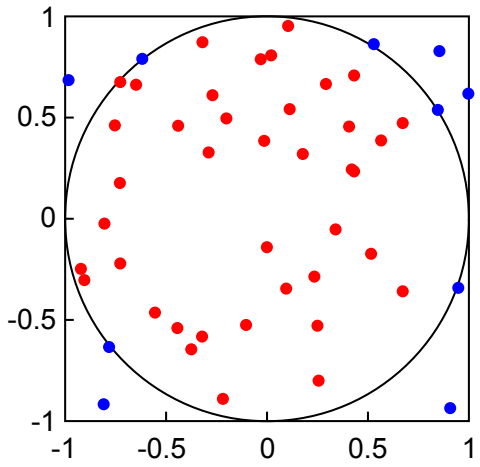
Why “Monte Carlo” techniques ?

Monte Carlo (MC) techniques are computational algorithms that rely on repeated random sampling to obtain numerical results

They are used when analytical solutions are too complex or not even known

Examples:

- Numerical integration of complex, multidimensional integrals (eg phase-space integration of matrix elements describing particle physics processes)
- Simulation of LHC particle collisions (“events”) as measured by the particle detectors. This involves:
  - Matrix element generation of collision
  - Decay of produced particles and propagation of stable particles through detector material
  - Electronic response of active detector layers, and reconstructions of signals
  - Physics analysis
- Simpler: estimation of error on a measured quantity with unknown property (→ next slide)



Numerical estimation of circle area by taking ratio of red to red+blue points times the square’s area

Figure from: https://en.wikipedia.org/wiki/Monte_Carlo_integration

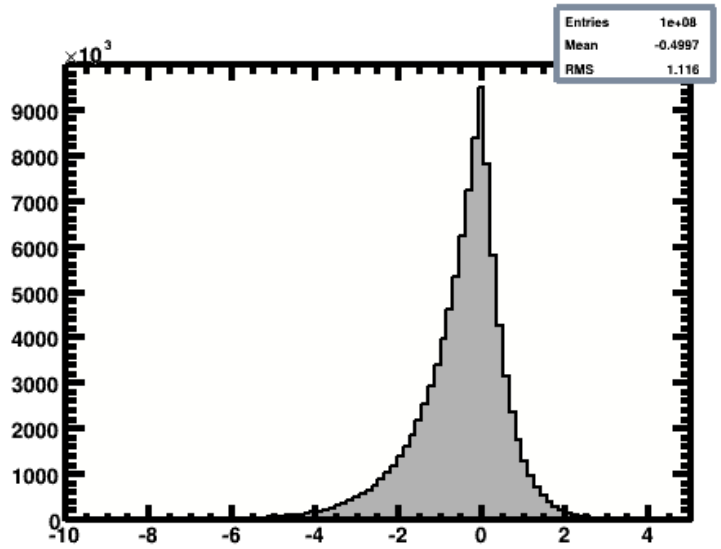
Bootstrap method

Consider the following problem: a quantity $\boldsymbol{x}$ was measured $\boldsymbol{N}$ times: $\boldsymbol{x_i}$ $(\boldsymbol{i = 1 \ldots N})$

One wants to determined a derived quantity $\boldsymbol{y(x_1, \ldots, x_N)}$, and needs an error for it.

→ Error propagation (remember: $\sigma_y = \left.\frac{dy(x)}{dx}\right|_{x=\bar{x}} \cdot \sigma_x$), but it requires to know the PDF of $\boldsymbol{x}$

Assume the distribution of the measured $\boldsymbol{x_i}$ looks like this:



- This ***is*** a PDF, and the best available information
- One can obtain a new set to "simulate" the measurements by applying *resampling with replacement*
- That is: one draws $\boldsymbol{N}$ events from the ensemble allowing to re-draw the same event multiple times
- One does this many times

→ **Bootstrapping**

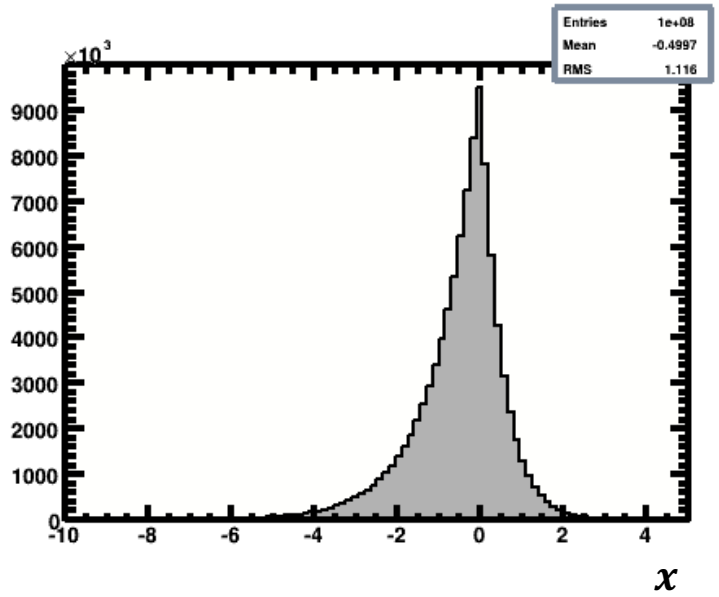
Bootstrap method

Consider the following problem: a quantity $\boldsymbol{x}$ was measured $\boldsymbol{N}$ times: $\boldsymbol{x_i}$ $(\boldsymbol{i = 1 \ldots N})$

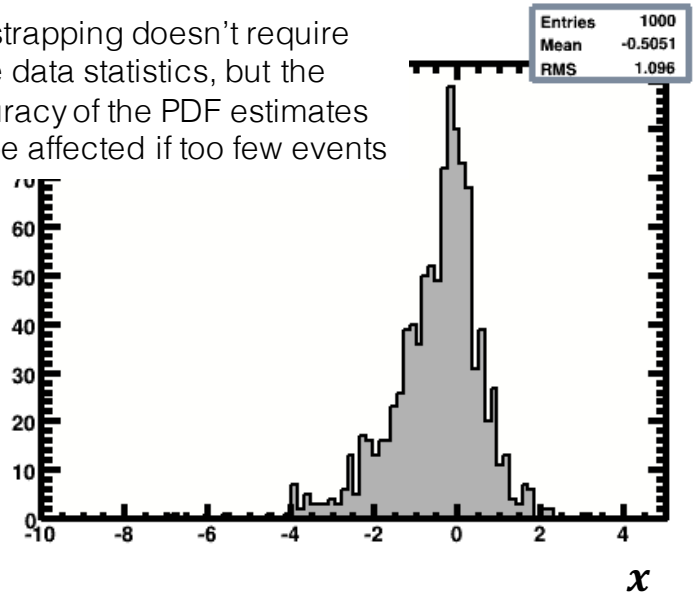
One wants to determined a derived quantity $\boldsymbol{y(x_1, \ldots, x_N)}$, and needs an error for it.

→ Error propagation (remember: $\sigma_y = \left.\frac{dy(x)}{dx}\right|_{x=\bar{x}} \cdot \sigma_x$), but it requires to know the PDF of $\boldsymbol{x}$

Assume the distribution of the measured $\boldsymbol{x_i}$ looks like this:



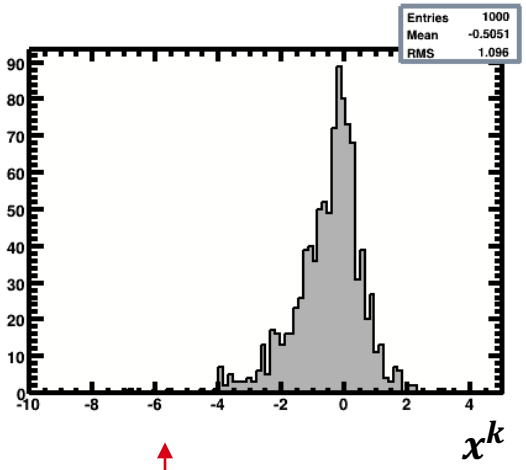
Boostrapping doesn't require large data statistics, but the accuracy of the PDF estimates will be affected if too few events

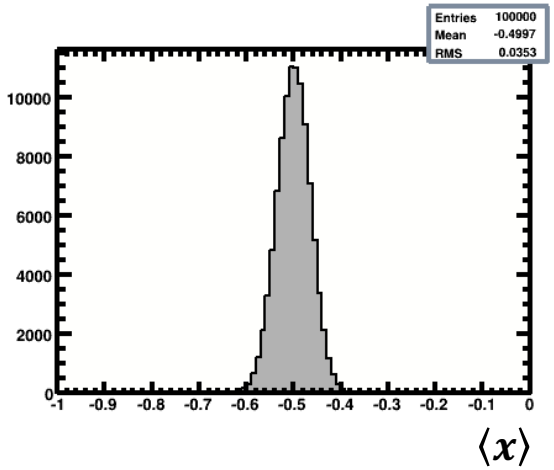


Bootstrap method — does it really work ?

Let's try with our toy example: simulate 100000 experiments with 1000 events each sampled from some analytic PDF (Nature's unknown truth) that we want to approximate

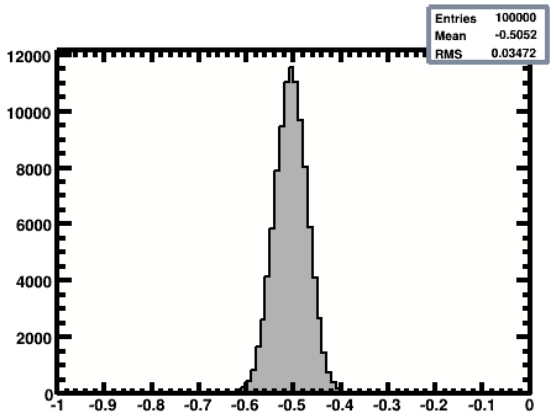
One example experiment k





Distribution of mean values among all experiments

Now, use **this experiment** to sample 100000 bootstrap experiments

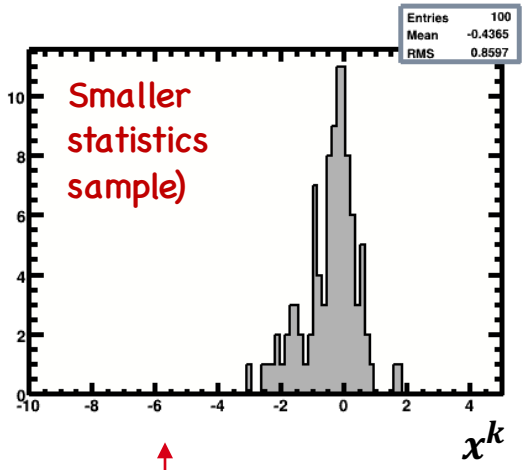


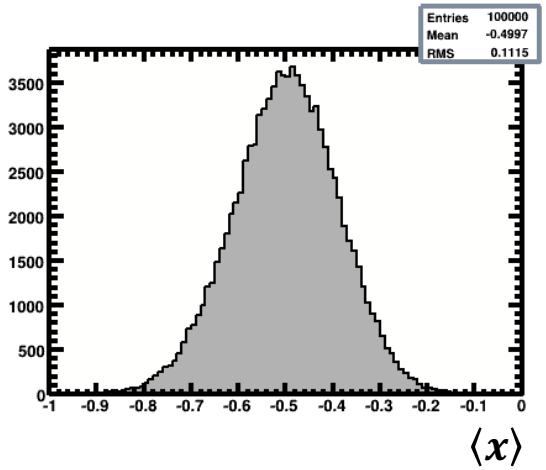
Satisfying result: RMS reproduced within 1.7%

Bootstrap method — does it really work ?

Let's try with our toy example: simulate 100000 experiments with **100** events each sampled from some analytic PDF (Nature's unknown truth) that we want to approximate

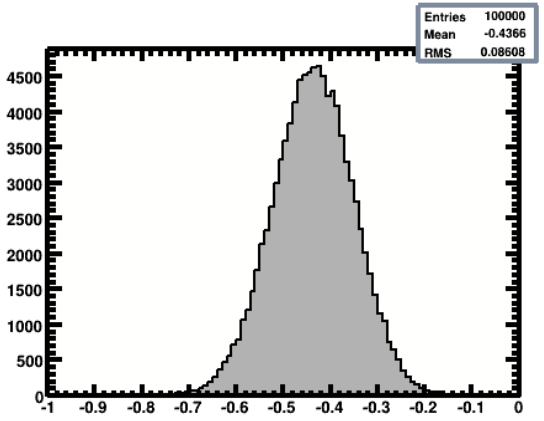
One example experiment k





Distribution of mean values among all experiments

Now, use **this experiment** to sample 100000 bootstrap experiments



Mediocre result: RMS reproduced within 30%

See: http://projecteuclid.org/download/pdf_1/euclid.ss/1177013815 for more information on bootstrap method

Jackknife resampling method (also called: leave-one-out cross validation)

Old method (~1950), basically replaced by bootstrap. Nevertheless instructive to know

Let's again consider: a quantity $\boldsymbol{x}$ was measured $\boldsymbol{N}$ times: $\boldsymbol{x_i}$ $(\boldsymbol{i = 1 \ldots N})$

One wants to determined a derived quantity $\boldsymbol{y = y(x_1, \ldots, x_N)}$, and needs an error for it:

- Study how the $\boldsymbol{y}$ changes when *leaving out one measurement* at the time

  let: $y_i = y_i(x_1, \ldots x_{i-1}, x_{i+1}, \ldots x_N)$,

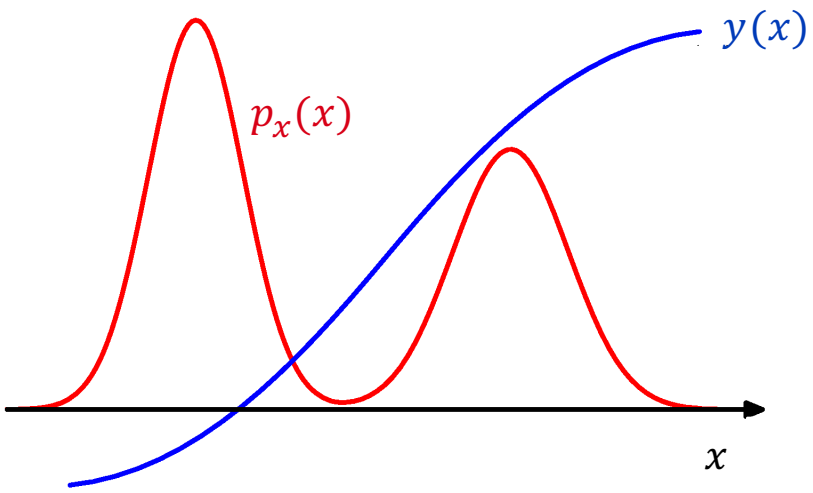
  and compute pseudo-value: $y_i^{\mathrm{Jack}} = Ny - (N-1)y_i = y + (N-1)(y - y_i)$

- Plot y_i^{Jack} for all $i = 1 \ldots N$ and treat them as if they were independent samples of the measured quantity.

- Compute mean or variance from y_i^{Jack} ensemble

Monte Carlo (MC) integration

Want to *numerically* compute an expectation value: $\boldsymbol{E[y] = \int y(x) p_x(x) dx}$



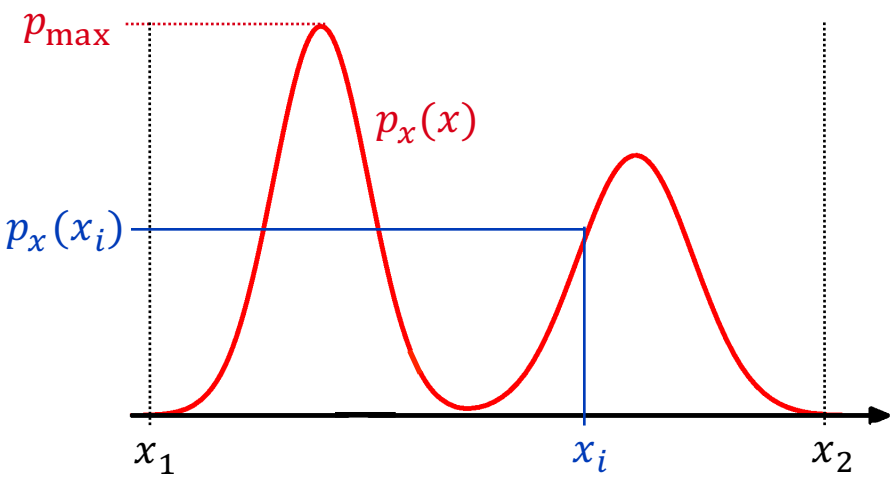
- Simplest solution: n-equidistant stepwise summation
- Works in 1, possibly *few* dimensions D
- Bad course of dimensionality: exponential growth of n with D
- Random MC phase-space sampling converges faster for large D

MC integration to compute $E[y]$ requires MC sampling according to PDF $p_x(x)$

That given, one finds: $\int \boldsymbol{y(x) p_x(x) dx} \approx \frac{\boldsymbol{1}}{\boldsymbol{N_{samples}}} \sum_{\boldsymbol{i=1}}^{\boldsymbol{N_{samples}}} \boldsymbol{y(x_i)}$

“Hit-or-miss” rejection sampling

Simplest way to generate random numbers (to “sample”) according to PDF $p_x(x)$



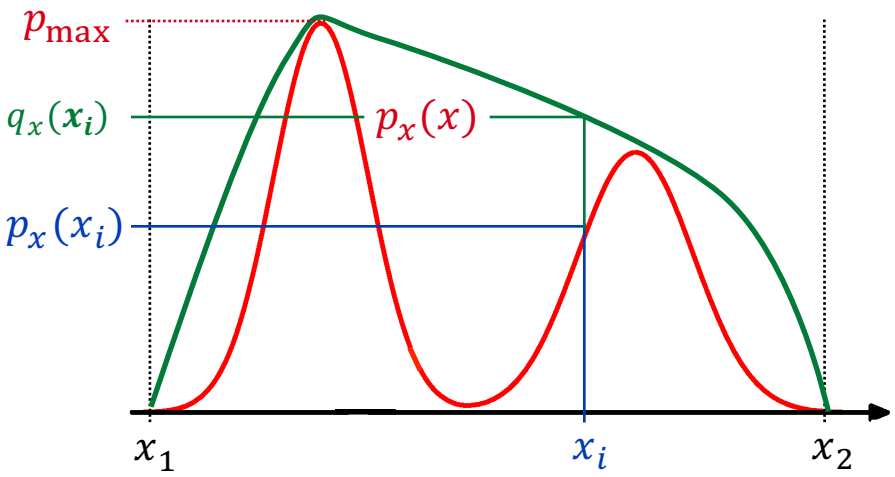
1. Generate uniform random number in interval $[x_1, x_2] \rightarrow \boldsymbol{x_i}$ and $\boldsymbol{p_x(x_i)}$
2. Generate another uniform random number in interval $[0, p_{\max}] \rightarrow \boldsymbol{p_i}$
3. If $\boldsymbol{p_i < p_x(x_i)}$: **accept** $\boldsymbol{x_i}$; **else**: **reject**

$\boldsymbol{q_x(x)}$, the (here uniform) PDF of generated $\boldsymbol{x}$ values defines *proposal distribution*

→ one could be smarter to have a larger “accept” rate (efficiency)

Rejection sampling

One can choose a (known) proposal distribution $q_x(x)$ closer to $p_x(x)$



1. Generate random number according to $q_x(x)$ in interval $[x_1, x_2] \rightarrow \boldsymbol{x_i}$ and $\boldsymbol{p_x(x_i)}$
2. Generate another uniform random number in interval $[0, q_x(x_i)] \rightarrow \boldsymbol{p_i}$
3. If $\boldsymbol{p_i < p_x(x_i)}$: **accept** $\boldsymbol{x_i}$; **else**: **reject**

Fraction of accepted events now larger than before (there are techniques to adapt automatically the proposal distribution during the generation)

→ can be even more clever if only integration needed, no random event generation

Markov chain Monte Carlo (MCMC) method

So far, the accuracy of the sampling depended on how closely $\boldsymbol{q_x(x)}$ follows $\boldsymbol{p_x(x)}$

This is a problem for sparsely known $\boldsymbol{p_x(x)}$ in case of complex multi-*D* structure. Every random point is chosen independently of every other one.

Markov chain: (eg, ”random walk”)

- Consecutive random steps depend on previous ones in random variable space
- Allows to favor stepping into regions where $\boldsymbol{p_x(x)}$ is large

Several algorithms: *Metropolis*, *Gibbs*, … → next pages

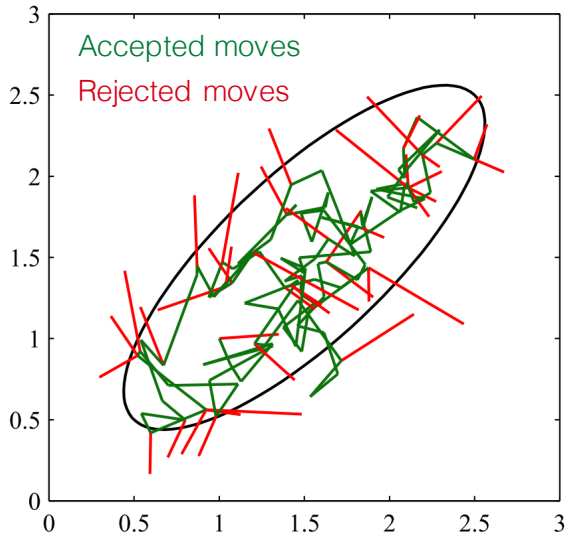
Metropolis sampling algorithm (1953)

Autocorrelation sampling of PDF $\boldsymbol{p_x(x)}$

1. Start anywhere in (multidimensional) $\boldsymbol{x}$ space and sample this point: $\boldsymbol{p_1 = p_x(x_1)}$
2. Provide proposal distribution $\boldsymbol{q_x(x_2|x_1)}$ to move from $\boldsymbol{x_1 \rightarrow x_2}$
   - $\boldsymbol{q_x(x_2|x_1)}$ could be Gaussian with appropriate metric in $\boldsymbol{x}$ space to cover full space
   - Accept $\boldsymbol{x_2}$ if: $\boldsymbol{p_2 > p_1}$ else: with probability $\boldsymbol{p_2/p_1}$
   - Sample either new point $\boldsymbol{x_2}$ (if accepted), or otherwise $\boldsymbol{x_1}$ again
3. Iterate step 2. for $\boldsymbol{x_3}$ vs. $\boldsymbol{x_2}$, etc.

Sample points $\boldsymbol{x}$ will wander closer and closer to the peak of the PDF, still jumping enough from time to time to sample the whole space

(Algorithm requires sufficient iterations. Test by checking stability of derived result, or by comparing several sampling ensembles obtained with different start values)

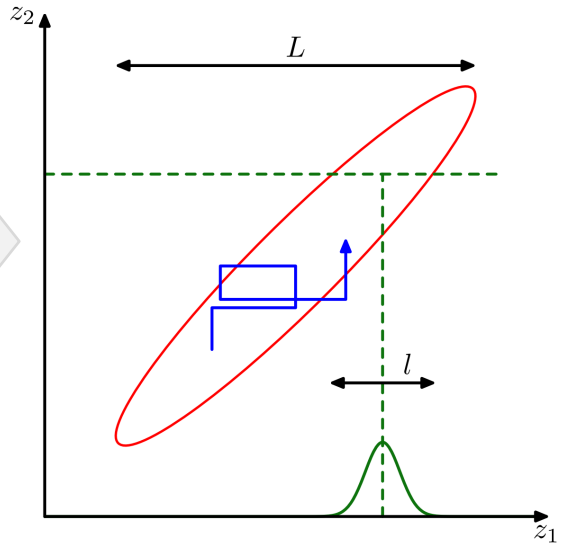


This subfigure from PRML, Bishop (2006)

Gibbs sampling algorithm (1984)

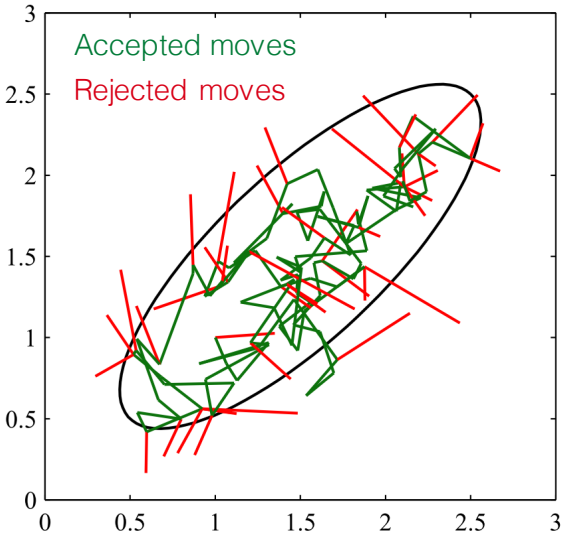
A method with no rejection:

1. Instead of moving along all dimensional components (and reject low-probability moves), the Gibbs sampler moves along 1 component according to the PDF conditioned on all other components.
2. Cycle through all components



Markov chain Monte Carlo usually converge fast and, if metric well chosen, cover the full space

However: care needs to be taken as the sample points $\boldsymbol{x}$ are correlated (with either sampling method): it depends on the application whether or not this is an issue



This subfigure from PRML, Bishop (2006)

Data unfolding

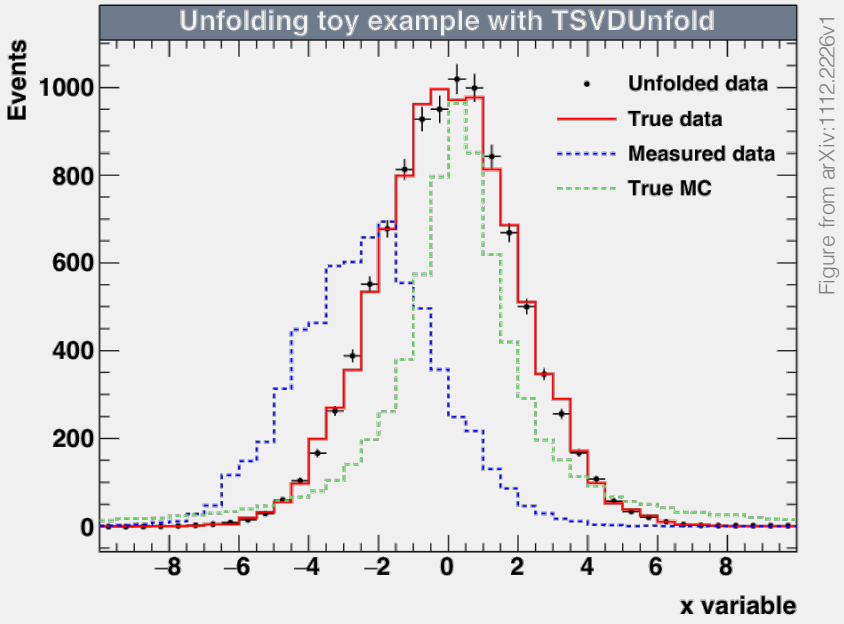


Figure from arXiv:1112.2226v1

Data unfolding — introduction

"Unfolding" means correcting measured data for any effects related to the measurement device. The unfolded data can be directly compared to theory or among experiments

Consider a measured histogram $\boldsymbol{y}^{\mathrm{data}} = \{y_1^{\mathrm{data}}, \ldots, y_m^{\mathrm{data}}\}$, a corresponding Monte Carlo histogram $\boldsymbol{y}^{\mathrm{MC}}$ of the same process as the data that underwent full detector simulation, its *truth* distribution (ie, before detector simulation) $\boldsymbol{x}^{\mathrm{MC}} = \{x_1^{\mathrm{MC}}, \ldots, x_n^{\mathrm{MC}}\}$, and the $m \times n$ matrix $\boldsymbol{A}^{\mathrm{MC}}$ obtained from MC that describes the "smearing" process due to the measurement:

$$\boldsymbol{A}^{\mathrm{MC}} \cdot \boldsymbol{x}^{\mathrm{MC}} = \boldsymbol{y}^{\mathrm{MC}}$$

Note that in general $\boldsymbol{y}^{\mathrm{data}} \neq \boldsymbol{y}^{\mathrm{MC}}$ (the physics leading to $\boldsymbol{y}^{\mathrm{data}}$ is what we want to measure), but we assume $\boldsymbol{A}^{\mathrm{MC}} = \boldsymbol{A}^{\mathrm{data}}$ (we know the detector response).

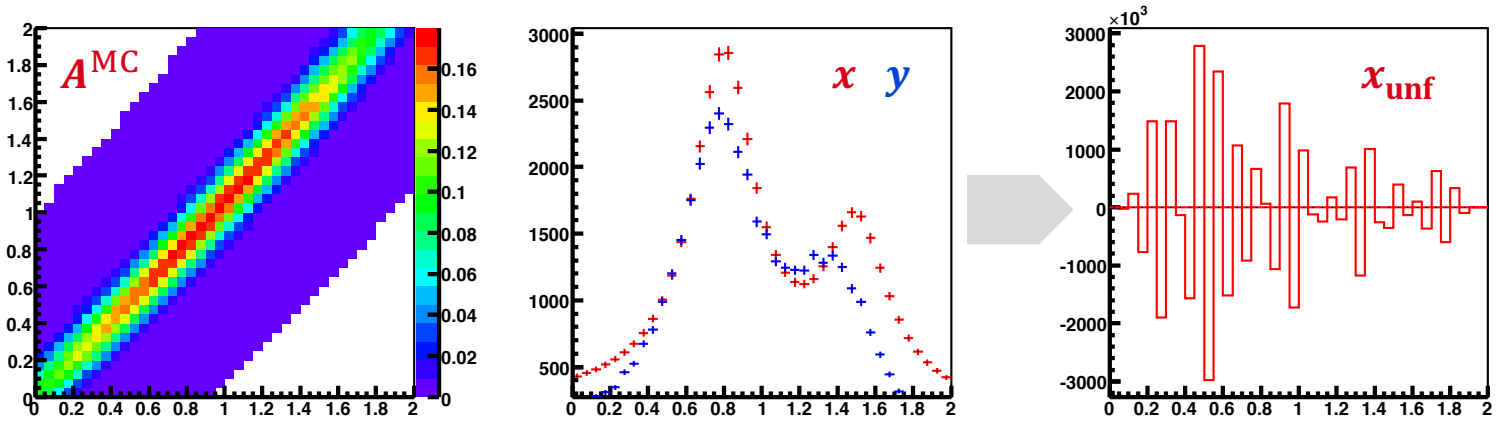
Hence, to obtain the truth information $\boldsymbol{x}^{\mathrm{data}}$, one *just* needs to invert $\boldsymbol{A}$:

$$\boldsymbol{x}^{\mathrm{data}} = \left(\boldsymbol{A}^{\mathrm{MC}}\right)^{-1} \cdot \boldsymbol{y}^{\mathrm{data}}$$

This is where the trouble begins …

Data unfolding — introduction

The distribution $\boldsymbol{y}^{\mathrm{data}}$ and the matrix $\boldsymbol{A}^{\mathrm{MC}}$ have finite statistics. An attempt to solve the problem directly and "exactly" will end up looking like this:



Large bin-to-bin oscillations ("Gibbs phenomenon") in unfolded spectrum

The poor solution, bin-by-bin corrections, $x_i^{\mathrm{data}} = \frac{x_i^{\mathrm{MC}}}{y_i^{\mathrm{MC}}} \cdot y_i^{\mathrm{data}}$, only works if $\boldsymbol{A}^{\mathrm{MC}}$ is square and ~diagonal so that the ratio $x_i^{\mathrm{MC}}/y_i^{\mathrm{MC}}$ corrects for mainly efficiency effects, or if $y_i^{\mathrm{data}} \cong y_i^{\mathrm{MC}}$.

A better solution is to regularise the matrix inversion problem …

Data unfolding — regularisation

Regularisation damps the oscillations, by suppressing statistically insignificant bins in the data distribution and response matrix.

In simplified form, one can write the unfolding problem as a minimisation of

$$\chi^2(\boldsymbol{x}^{\mathrm{data}}) = \left(\boldsymbol{A}^{\mathrm{MC}} \cdot \boldsymbol{x}^{\mathrm{data}} - \boldsymbol{y}^{\mathrm{data}}\right)^T \left(\boldsymbol{A}^{\mathrm{MC}} \cdot \boldsymbol{x}^{\mathrm{data}} - \boldsymbol{y}^{\mathrm{data}}\right) + \tau \cdot \left(C\boldsymbol{x}^{\mathrm{data}}\right)^T \left(C\boldsymbol{x}^{\mathrm{data}}\right)$$

where C is a matrix and $C\boldsymbol{x}^{\mathrm{data}}$ is the sum of squares of the 2nd derivative of $\boldsymbol{x}^{\mathrm{data}}$

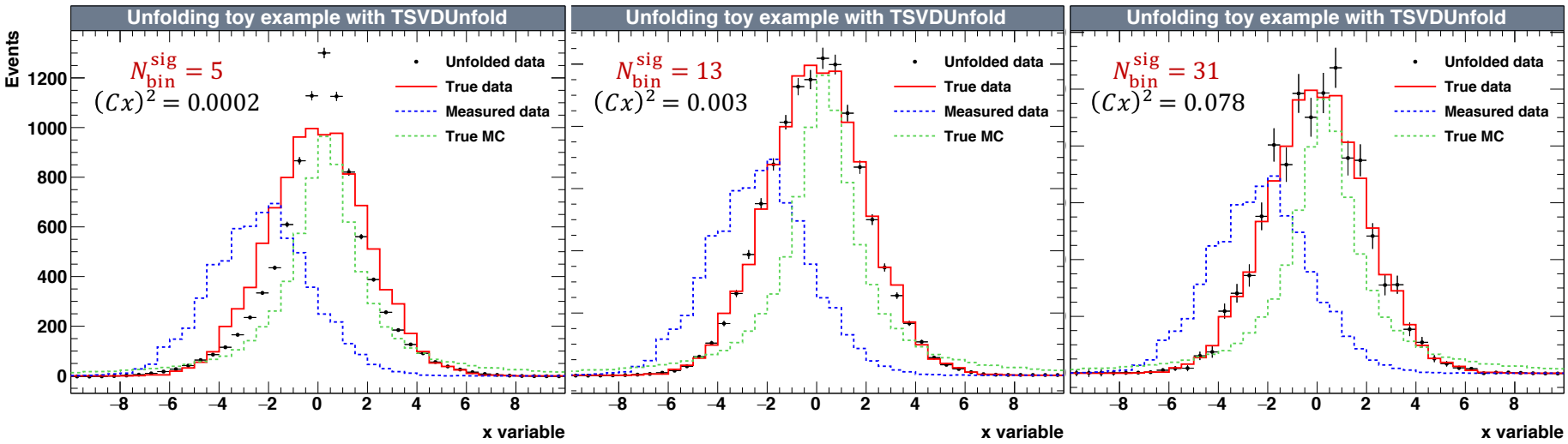
Minimising χ^2 wrt. the first term only corresponds to the (bad) exact inversion solution. The second term regularises the inversion by damping the oscillations.

The parameter τ regulates the strength of the damping:

- If τ too small → oscillations
- If τ too large → information in $\boldsymbol{x}^{\mathrm{data}}$ is suppressed
  ($\boldsymbol{x}^{\mathrm{data}}$ becomes too "smooth" and will be biased towards $\boldsymbol{x}^{\mathrm{MC}}$)
- The right choice captures all significant information and discards the rest

Data unfolding — example

Over-regularised | Best regularisation choice | Under-regularised



The parameter τ regulates the strength of the damping:

- If τ too small → oscillations
- If τ too large → information in $\boldsymbol{x}^{\mathrm{data}}$ is suppressed
  ($\boldsymbol{x}^{\mathrm{data}}$ becomes too "smooth" and will be biased towards $\boldsymbol{x}^{\mathrm{MC}}$)
- The right choice captures all significant information and discards the rest

Folding versus unfolding

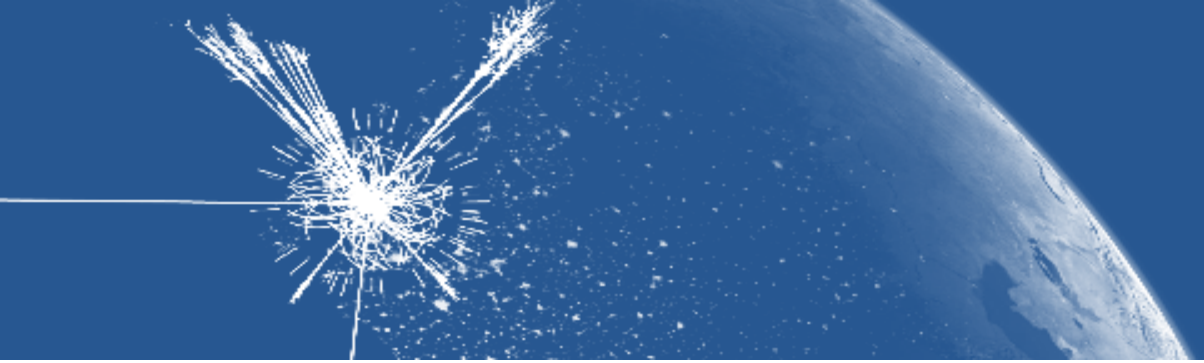
Unfolding is an ill-defined problem which necessarily leads to some obstruction of information in the data and transfer of statistical uncertainty into a systematic one after regularisation (this is similar to a non-parametric fit to data)

Technically simpler and mathematically well defined is the folding of a theoretical prediction $\boldsymbol{x}^{\mathrm{theo}}(\boldsymbol{\theta})$, depending on a set of parameters $\boldsymbol{\theta}$, through the detector response and direct comparison with the measured data. It allows the statistical test:

$$\chi^2(\boldsymbol{x}^{\mathrm{theo}}(\theta)) = \left(\boldsymbol{A}^{\mathrm{MC}} \cdot \boldsymbol{x}^{\mathrm{theo}}(\theta) - \boldsymbol{y}^{\mathrm{data}}\right)^T \left(\boldsymbol{A}^{\mathrm{MC}} \cdot \boldsymbol{x}^{\mathrm{theo}}(\theta) - \boldsymbol{y}^{\mathrm{data}}\right)$$

Folding requires that the experiments either perform the test, or publish $\boldsymbol{A}^{\mathrm{MC}}$ and $\boldsymbol{y}^{\mathrm{data}}$

Folding does not allow a model-independent combination or comparison among experiments. In most case, unfolding is the only viable solution for easy and long-term use of the experimental results.



Summary for today

Maximum likelihood fits are powerful optimisation tools that allow for any required complexity

Bootstrapping methods allow to straightforwardly re-sample measured data for the purpose of error propagation

Brief introduction to Monte Carlo integration and the sampling of random data according to any arbitrary PDF

Markov-Chain Monte Carlo integration is a very effective method that “automatically” samples the important regions (where the PDF is large) more often than tails. Try yourself!

Unfolding is a delicate mathematical operation that requires careful regularisation. Folding can help in some cases.