Statistics
(for “daily” data analysis in physics)

Sara Bolognesi (IRFU, CEA)

This course

I'm not an expert in statistics, still I use it daily since many years to perform data analysis in CMS (Higgs search) and then in T2K (neutrino oscillations)

You can find beautiful courses on-line on the theory and technical usage of statistics (and very good, high-level, dedicated book)

One of my favorite on-line resource, very good as "entry" point:
Lent Term 2015 by Prof. Mark Thomson (*)

Here I will focus on how we (mis! -)use the statistics in daily data-analysis work in "real life":
- histograms handling
- efficiency (→ purity and significance)
- resolution
- how to correct for such detector effects: unfolding (or 'forward fitting')
- likelihood fit
- practical use cases: neutrino oscillations, Higgs spin/parity determination

An ‘event’ → histograms

1 event = e.g. one neutrino interaction in your detector

Each event has many observables (eg: many outgoing particle and each particle has its 4-momenta)
… And we have many events

→ **How to organize all this information?**
Make histogram for each observable

An ‘event’ → histograms

1 event = e.g. one neutrino interaction in your detector

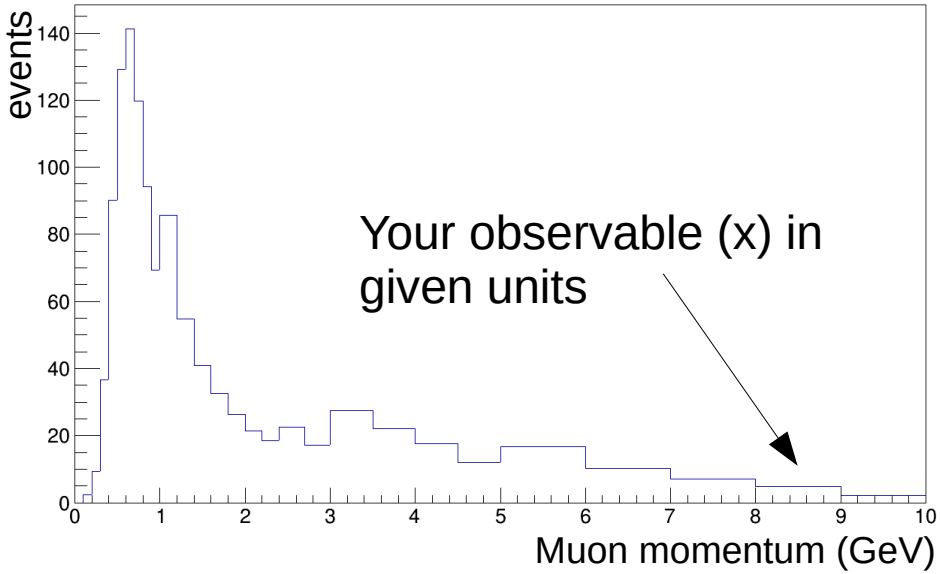
Each event has many observables (eg: many outgoing particle and each particle has its 4-momenta)
… And we have many events

→ **How to organize all this information?**
Make histogram for each observable

Used to represent ‘in principle’ continuous distribution of observable “x”, where **bins are range of possible x values**

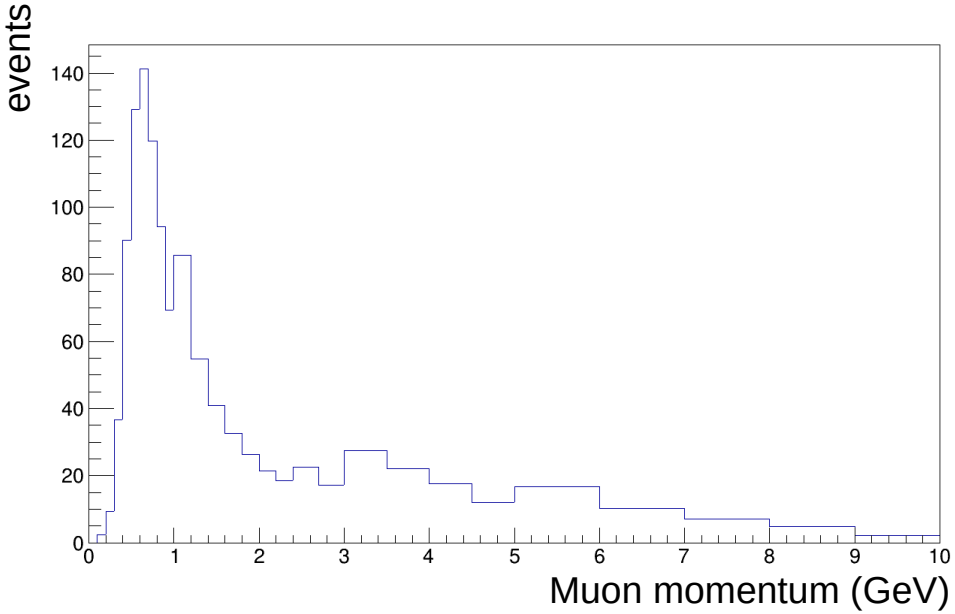
Entry in each bin = how many times the value of your observable x falls inside that range

In practice in physics you never have an infinite number of events (e.g. finite amount data or Monte Carlo statistics) so histograms represent useful numerical approximation to analytical continuous function



Typically, assuming you know the theoretical distribution x, you can recover the continuous analytical distribution which best describe your histogram → fit

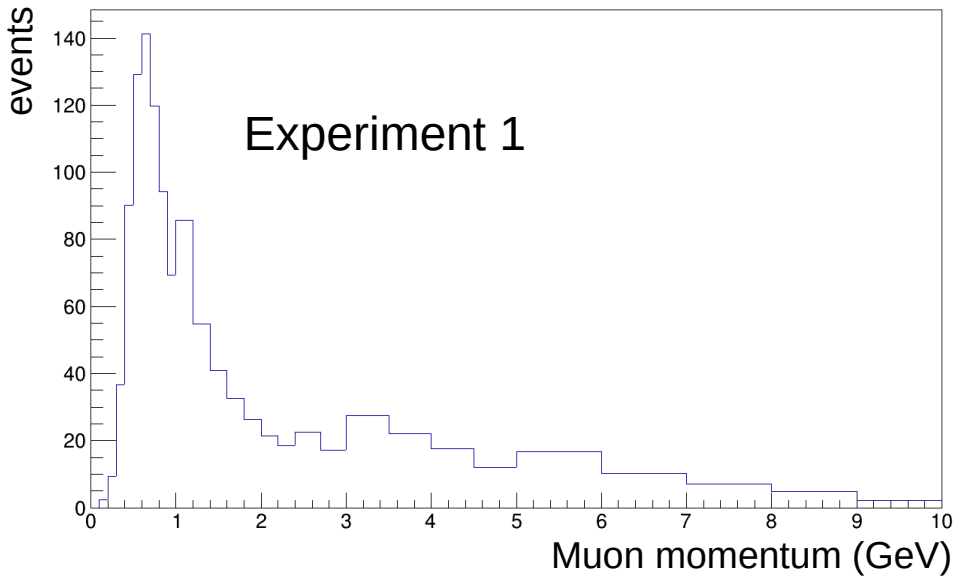
Histogram “normalization”

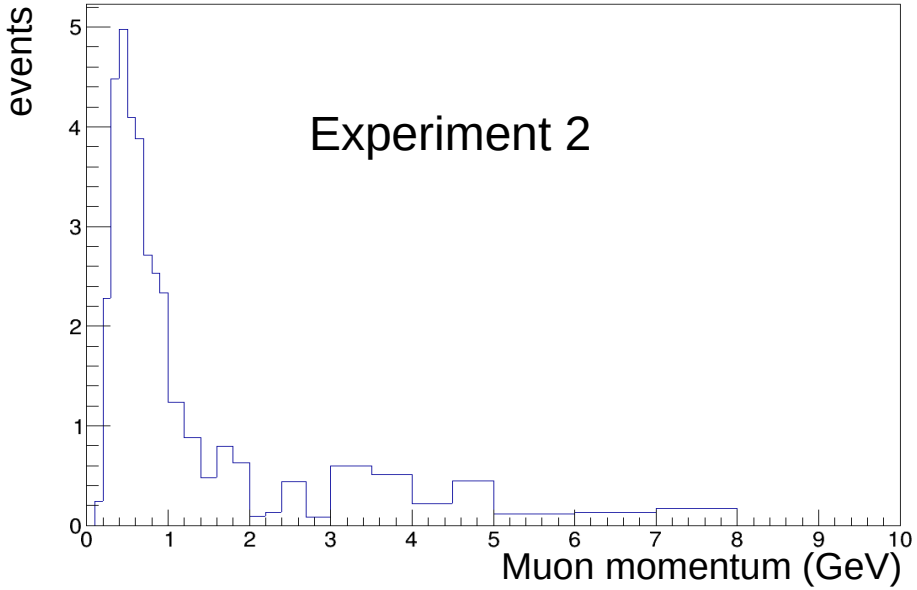


Normally the integral of the histogram (= sum of entries in each bin) gives the total number of events

$$\sum_{i}^{bins} N_i = N_{events}$$

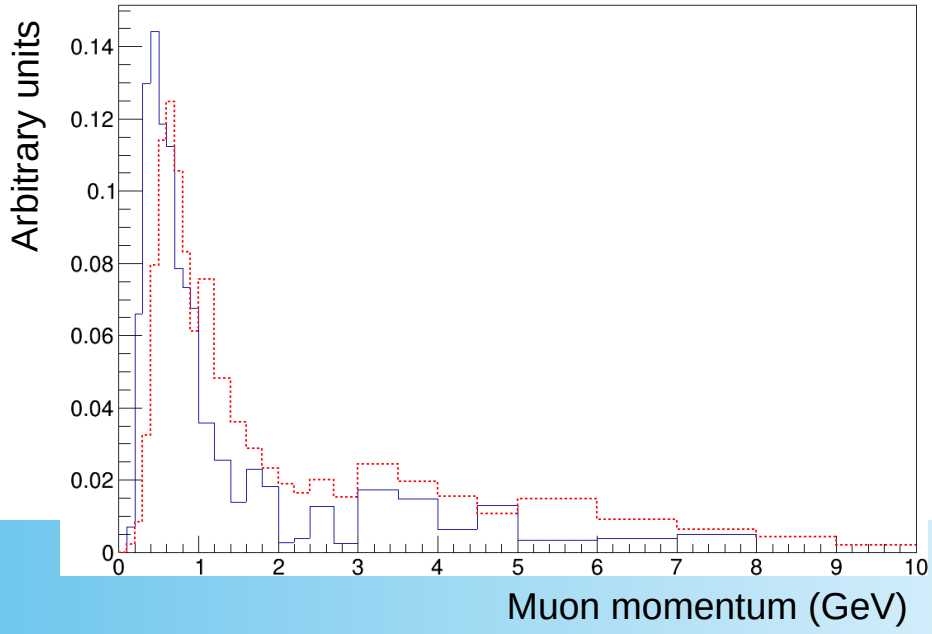
Histogram “normalization”



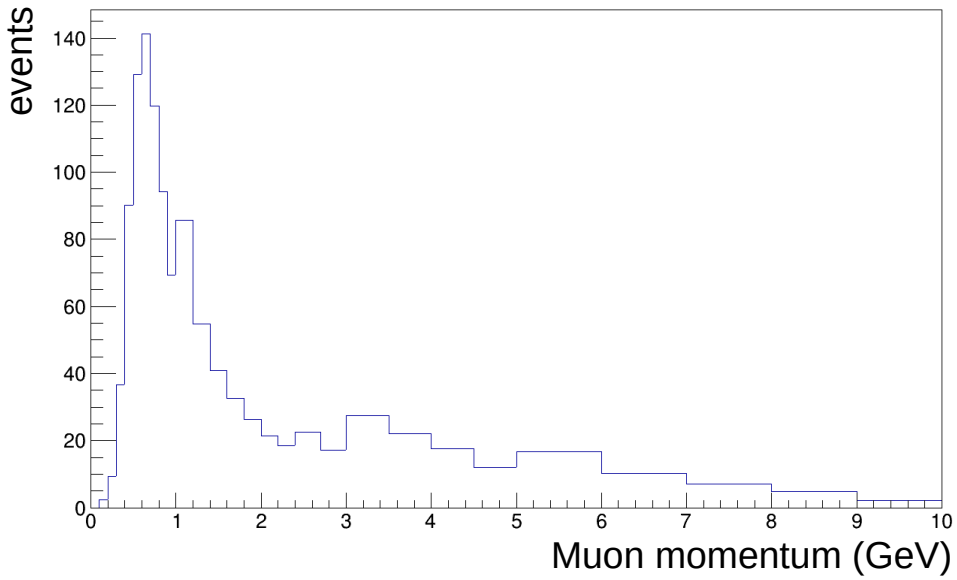


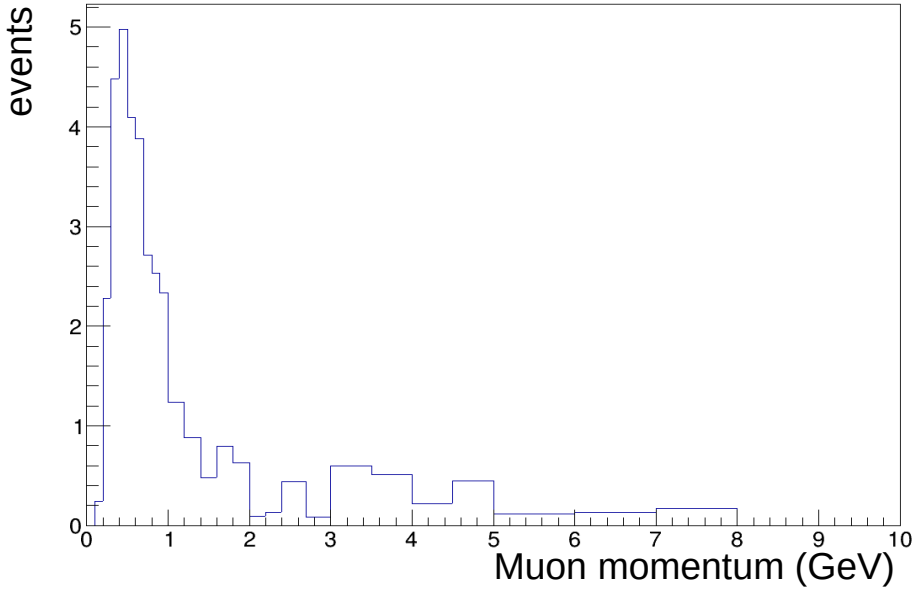
You may be interested only to the **shape of your distribution** (eg to compare two experiments with different number of events measuring the same observable)
→ “normalize” both to an arbitrary number (typically 1)

$$\sum_{i}^{bins} \frac{N_i}{N_{events}} = 1$$



Histogram “normalization”





You may want to renormalize with respect to some given factor: eg **number of events is proportional to cross-section**

→ renormalize to single factor to show on the y axis the xsec instead of the number of events

$$N \sim L \cdot \sigma \cdot A$$

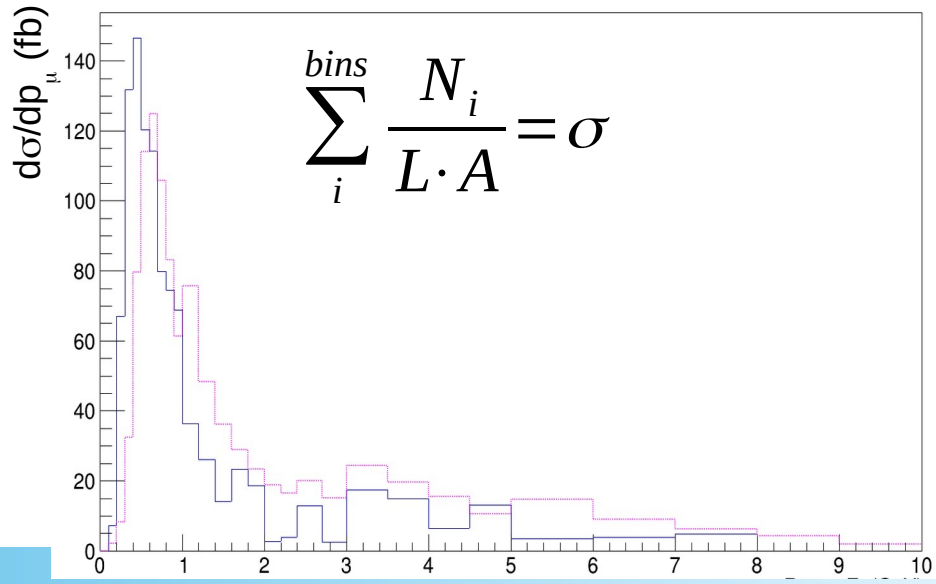
L ~ “luminosity” i.e. how many pp collisions, “flux”, i.e. how many ν produced

A ~ detector term (efficiency, mass for neutrino interactions, ...)

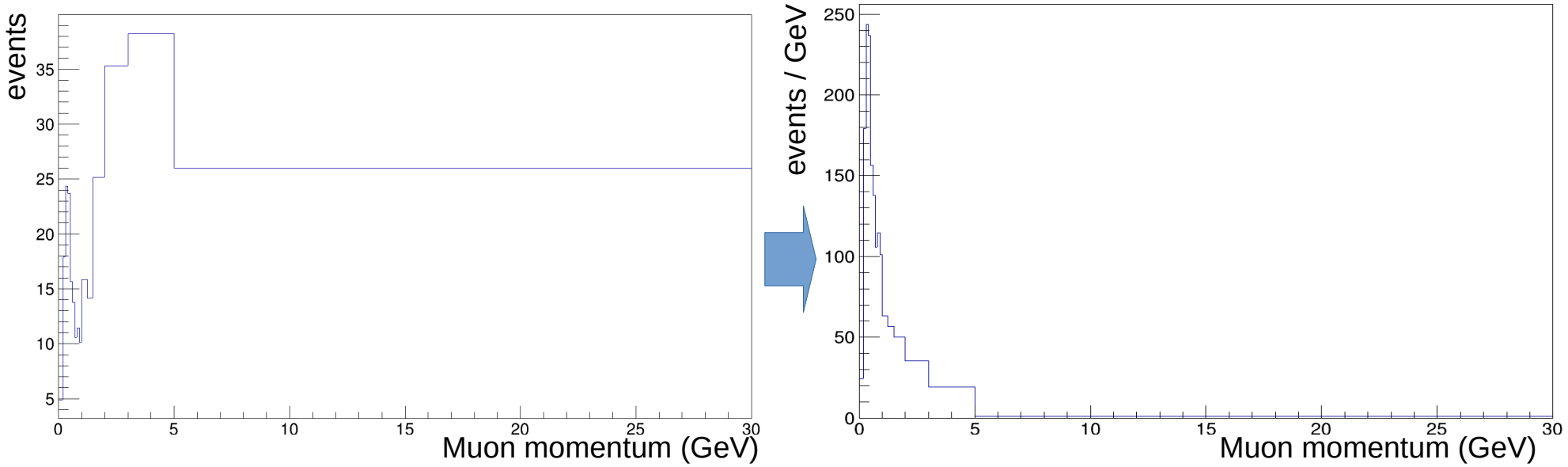
} experiment dependent

σ ~ cross-section, i.e. fundamental physics you want to measure

$$\sum_{i}^{bins} \frac{N_i}{L \cdot A} = \sigma$$



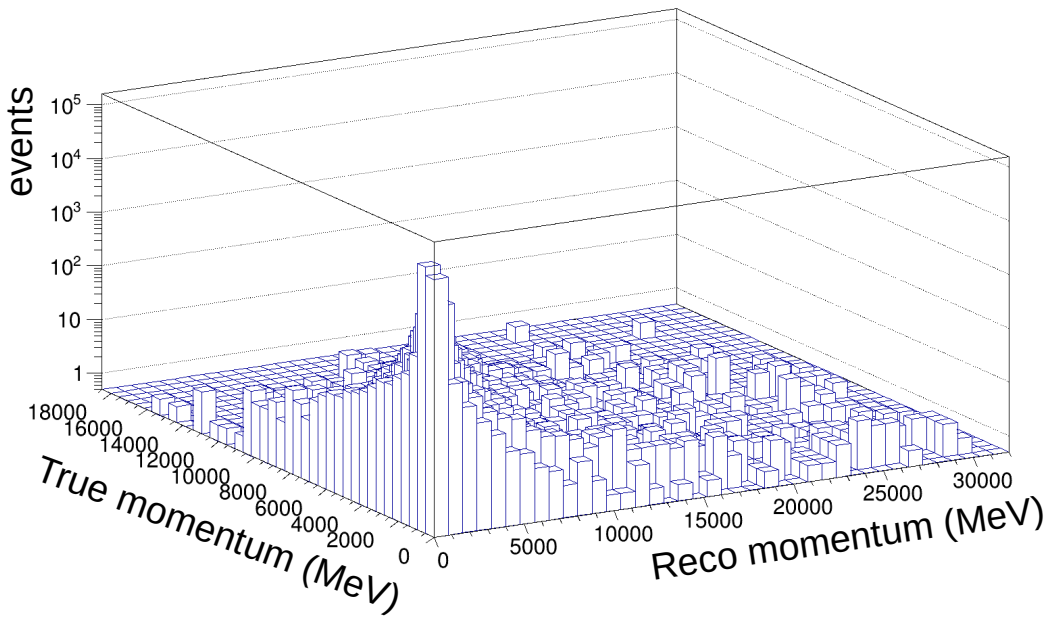
Histograms with variable bin width

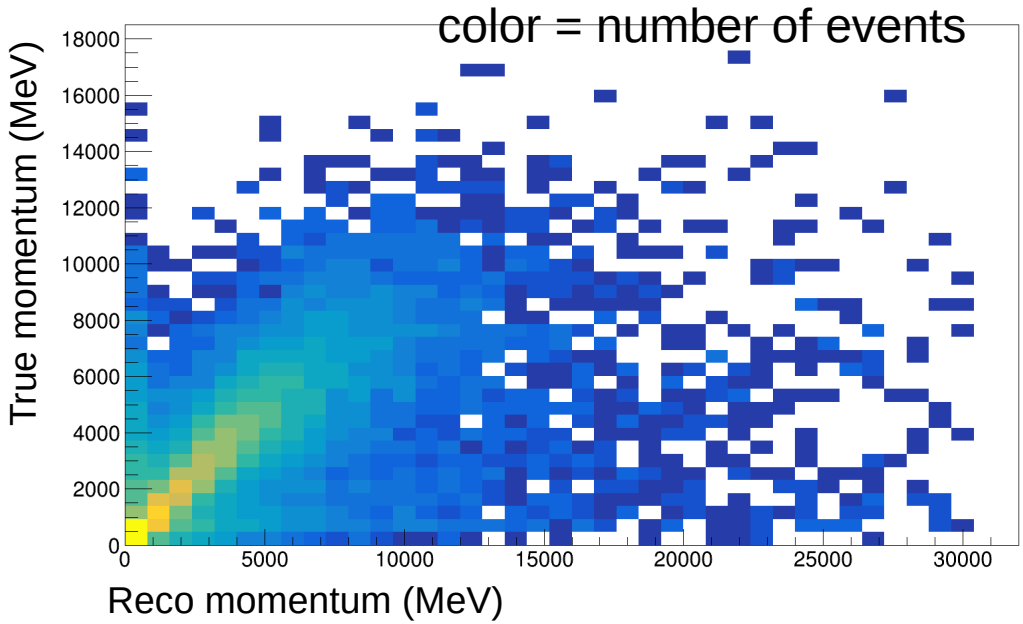


Variable bin:
Each bin can have different width.
This is useful when the histogram has bins with much different population (eg tails with low number of events → large statistical uncertainty)

Bin with larger width will have “by construction” larger number of entries
→ **to have the correct shape you need to divide by the bin width**

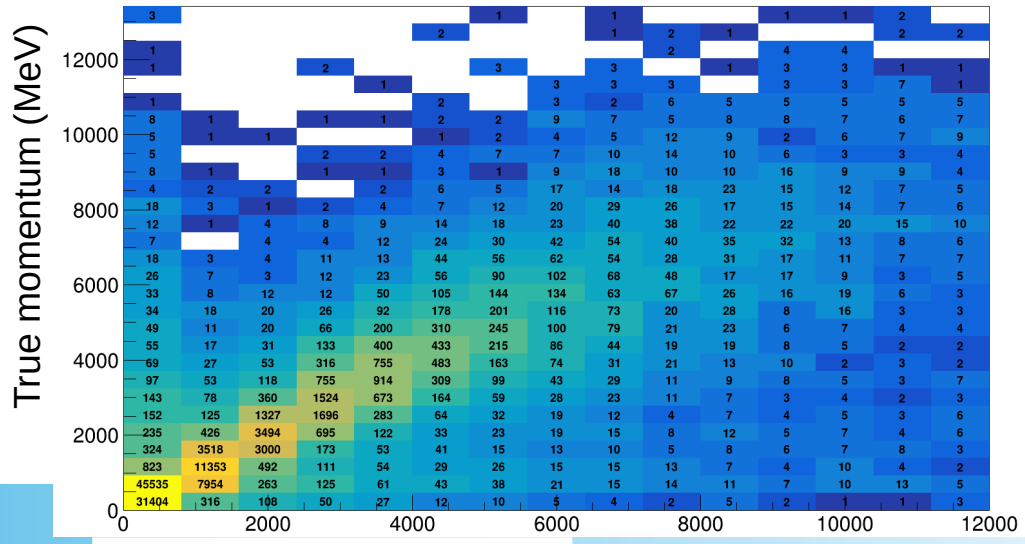
Multidimensional histograms





We considered 1D histograms but you may have 2D or … N-dimensional.
You will not be able anymore to visualize but **conceptually histograms are just "tables" of numbers and can have as many axes as you want**

The number of bins grows fast:
1D 10 bins → 2D 100 bins → 3D 1000 bins
….
Typically limited by available data statistics (you need many events to populate all the histogram with sufficient statistics)



Sufficient statistics?

If the probability for your observable to fall inside a given bin is small, then you will have small number of entries in that bin → large uncertainty

Quantify: what is the uncertainty on the number of events observed in one bin?

Assuming the number of expected entries (events) in one bin (dx) is μ, the probability of N observed entries (event) in that bin is

$$p(n;\mu) = \frac{\mu^n e^{-\mu}}{n!}$$

POISSONIAN DISTIRIBUTION

General description of discrete counts at fixed rate

- μ is the average number of expected events in the bin → our best approximation to μ (for one single experiment) is just the observed number of entries N_i
- the uncertainty (the variance of the distribution) is sqrt(μ) ~ **sqrt(N_i)**
- for (relatively) large statistics (N~10 or more) the Gaussian distribution is a very good approximation

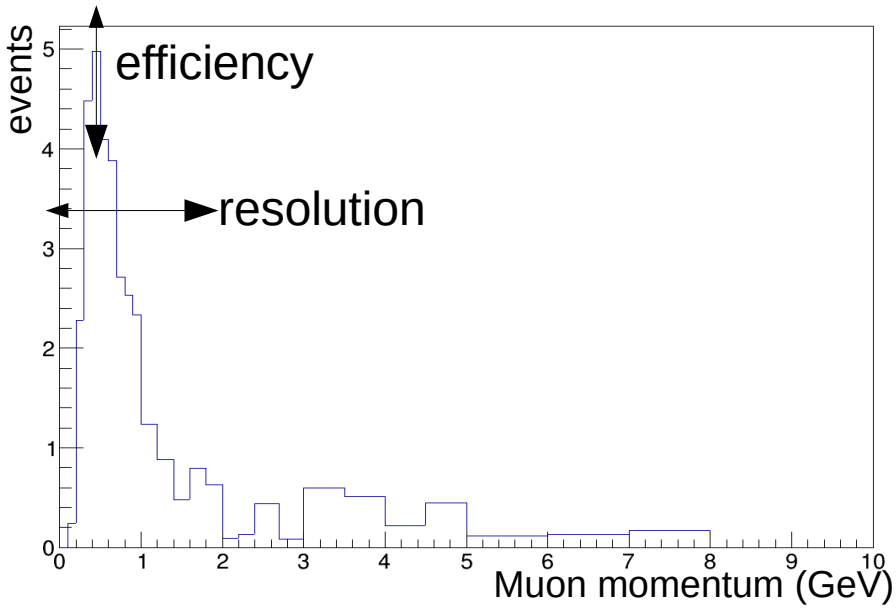
N events in one bin	Uncertainty (%)
1	100%
5	45%
10	32%
100	10%
1000	3%
10000	1%

Why? All derivations here (*)

Correcting for detector effects

Go from observed distribution to “true” distribution

- Efficiency: your detector is not able to register all events.
Typically efficiency is kinematics dependent: low momentum particles may be below the **threshold** of the detector and/or detector has limited **angular coverage**



- Resolution: your detector induce smearing. Also typically kinematics-dependent

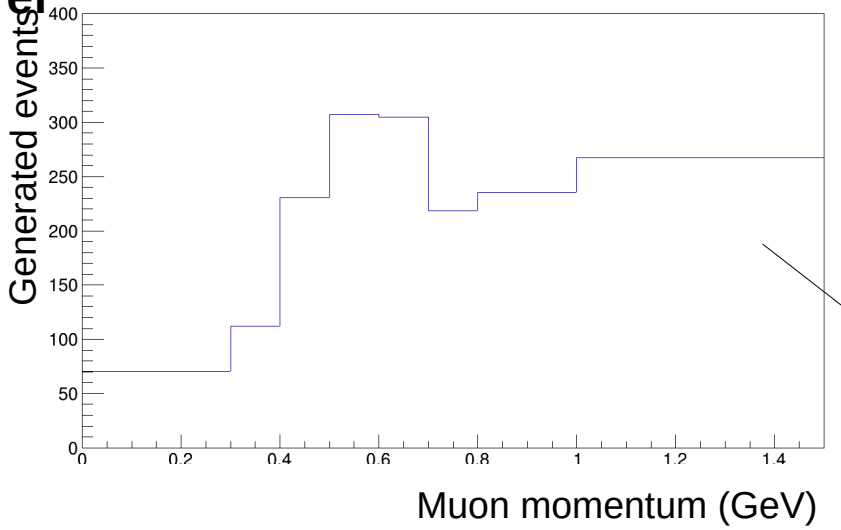
Two methods: Unfolding (from observed to true), Forward folding (from true to observed)

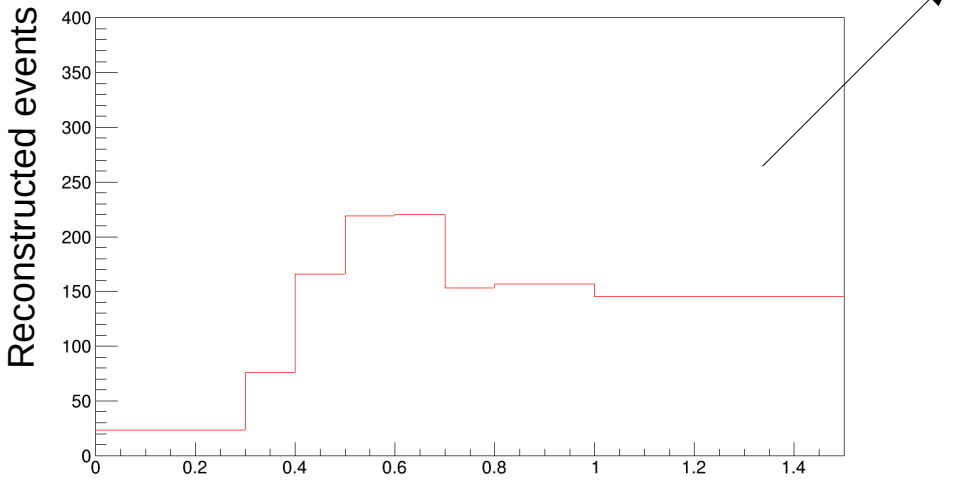
Need true estimation: from Monte Carlo simulation.
“Physics” simulation of particle interactions with “generators” (a.k.a “true” or “generate”) → passed through simulation of detector (GEANT4) (a.k.a “simulated”)

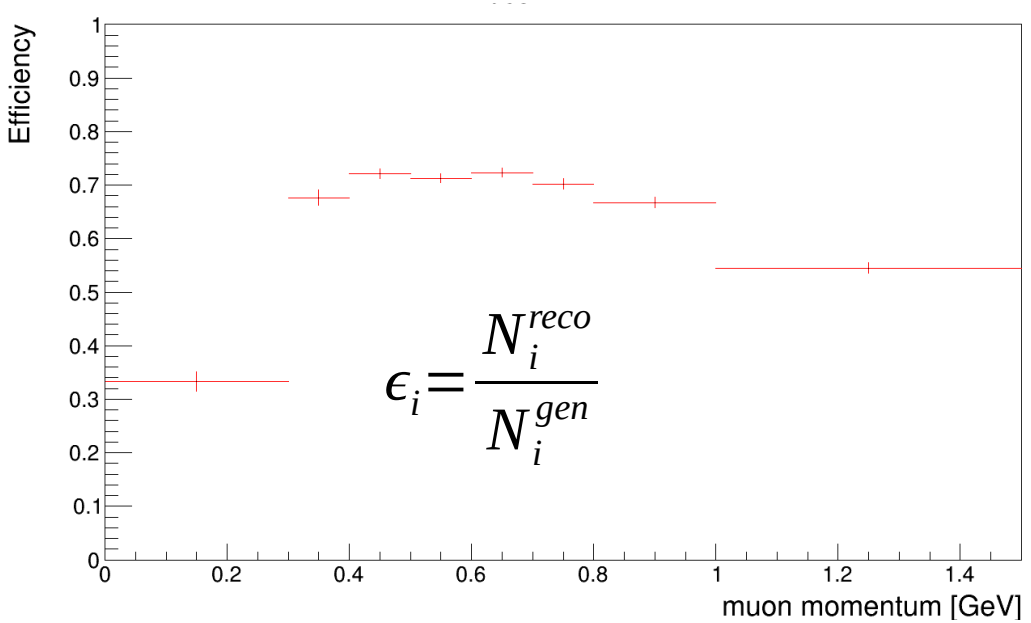
BREAK !

Efficiency

Back to histograms: **efficiency = ratio reco with detector effects / true without detector effects**







Efficiency

Uncertainty on the efficiency?

Take sqrt(N) of each histogram bin and propagate? You may end-up with error bars going above 100% (efficiency above 1)

Having an event entering or not in your detector efficiency it is the same probabilistic event as tossing a coin → BINOMIAL DISTRIBUTION

$$f(k,n,p) = \Pr(k;n,p) = \Pr(X=k) = \binom{n}{k} p^k (1-p)^{n-k}$$

for $k = 0, 1, 2, \ldots, n$, where

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$

- p = probability of k events passing the efficiency over n total events
- efficiency best estimate = k/n
- uncertainty on the efficiency (variance of the distribution)

$$\sigma^2 = \frac{\varepsilon(1-\varepsilon)}{n}$$

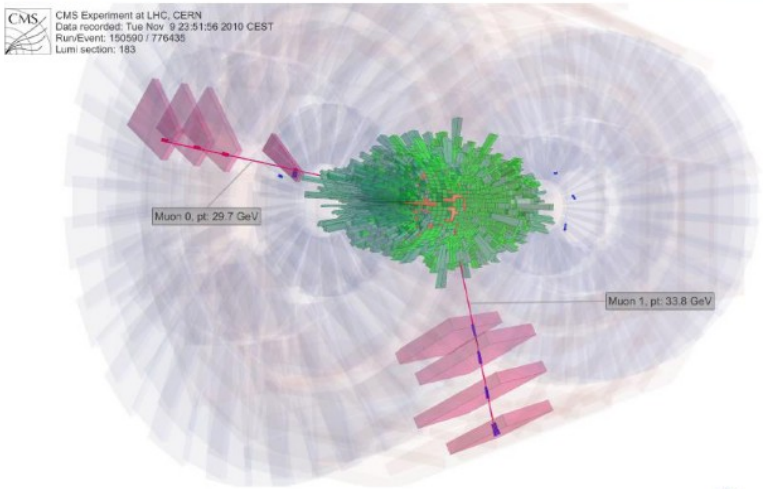
(error bars can never go to beyond 1 or below 0)

Why? All derivations (and even a more correct formula) here (*)

Efficiency: from simulation?

Efficiency = ratio true (without detector simulation) / detector simulated

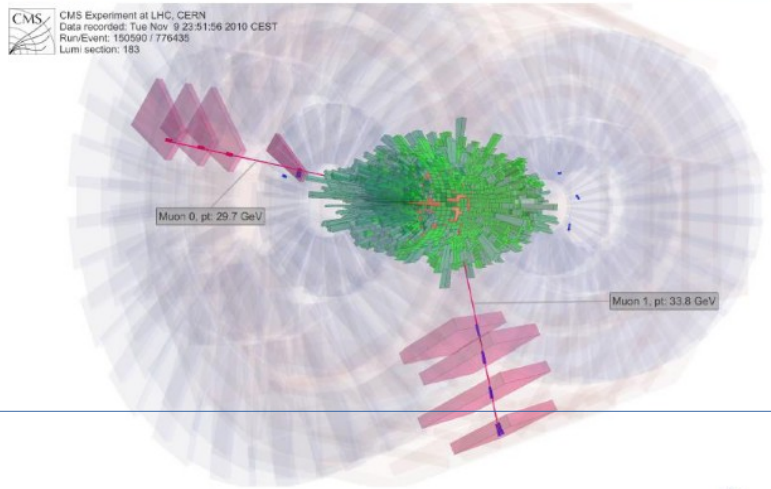
Real experiment: you never trust your simulation!
Use “control samples” to estimate efficiency and resolution (and background)
Control samples = sample of events which you know very well and which does not contain the signal events you are interested to measure
(since, by the definition, you do not know the characteristics of the signal you look for)



Efficiency: from simulation?

Efficiency = ratio true (without detector simulation) / detector simulated

Real experiment: you never trust your simulation!
Use “control samples” to estimate efficiency and resolution (and background)
Control samples = sample of events which you know very well and which does not contain the signal events you are interested to measure
(since, by the definition, you do not know the characteristics of the signal you look for)

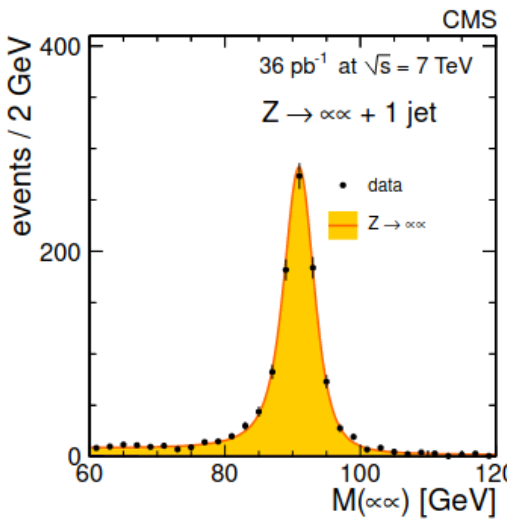


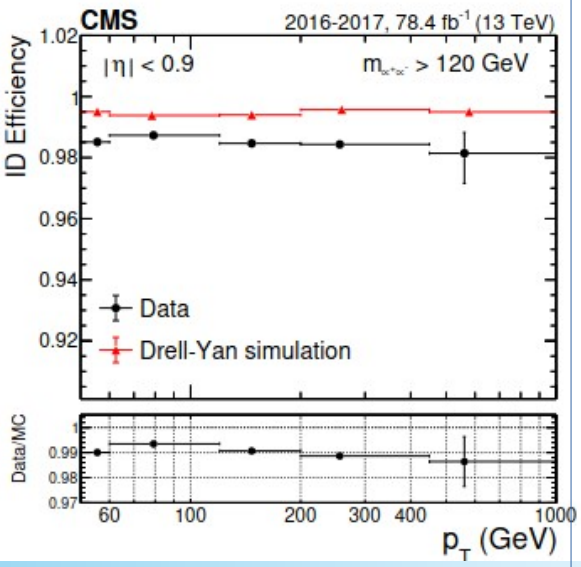
Example: “tag and probe”

You want to know the efficiency of your algorithm of muon selection

- Take a sample of 1 muon selected + look for other tracks with loose selection and reconstruct their invariant mass.
Under the Z peak you are sure that the other track is a muon ($Z \rightarrow \mu\mu$).
How many times the track is also selected as a muon? = Efficiency of muon selection

- Repeat the same exercise in simulated Monte Carlo





Histogram “reweighting”

Once you measured the efficiency from data you typically want to correct your MC simulated distribution to better match reality

“Reweighting” = take your simulated events and reweigth each of them by the difference between data and MC as a function of a given observable

$$\frac{\epsilon_{data}(y)}{\epsilon_{MC}(y)} = w(y)$$

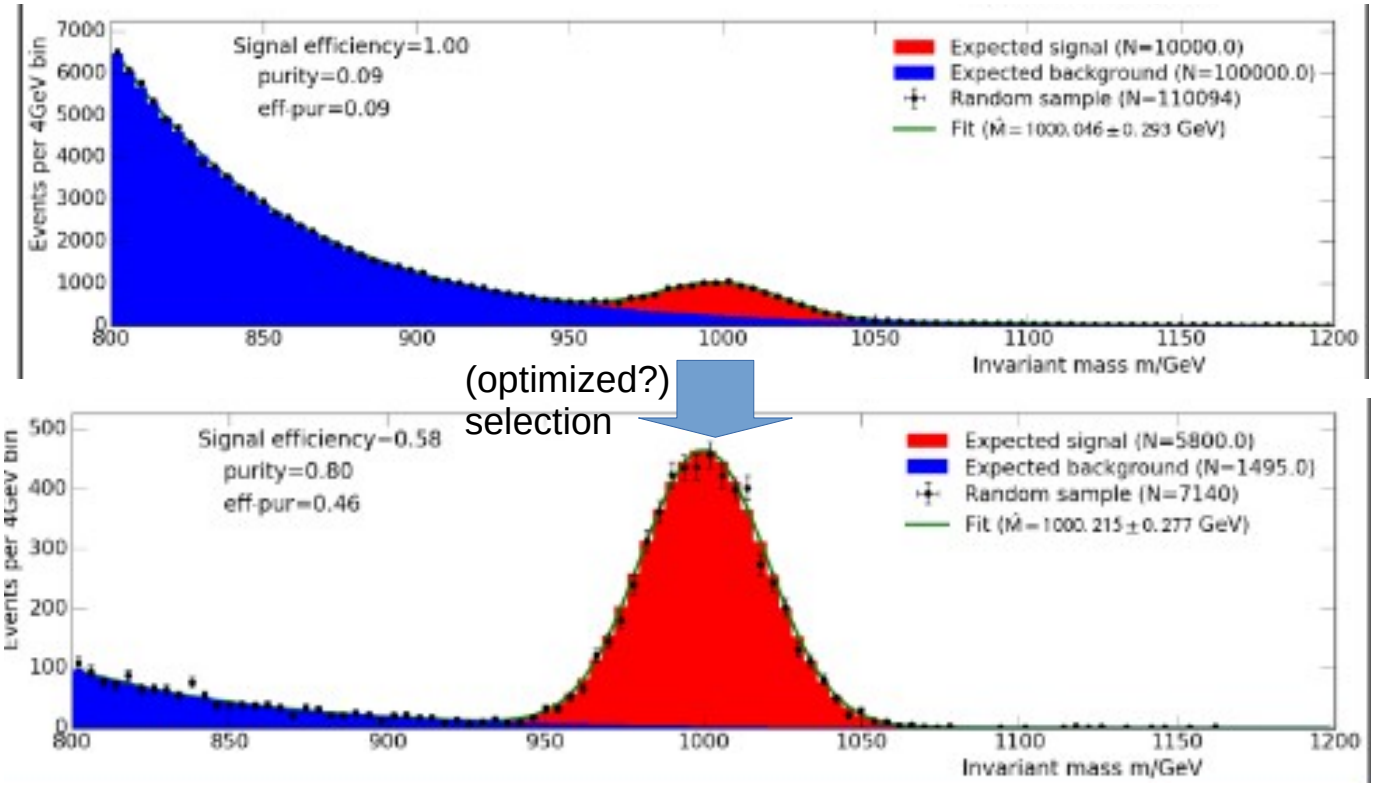
$$\sum_i N_i(x)\, w(y) = N_{events} \frac{\epsilon_{data}}{\epsilon_{MC}}$$

The previous “renormalization” examples were “reweighting” all the entries of your histogram with same weight → often you may have events which have different weights

Search for a signal

When you have your data collected, typically you need to search for your signal (e.g. resonance peak into the continuum background)

You use what you know of your signal kinematics to select signal-enriched sample: selection with cuts on observables
→ you reject (most of) background and you keep (most of signal)



How do you optimize your selection to have best signal “significance”?

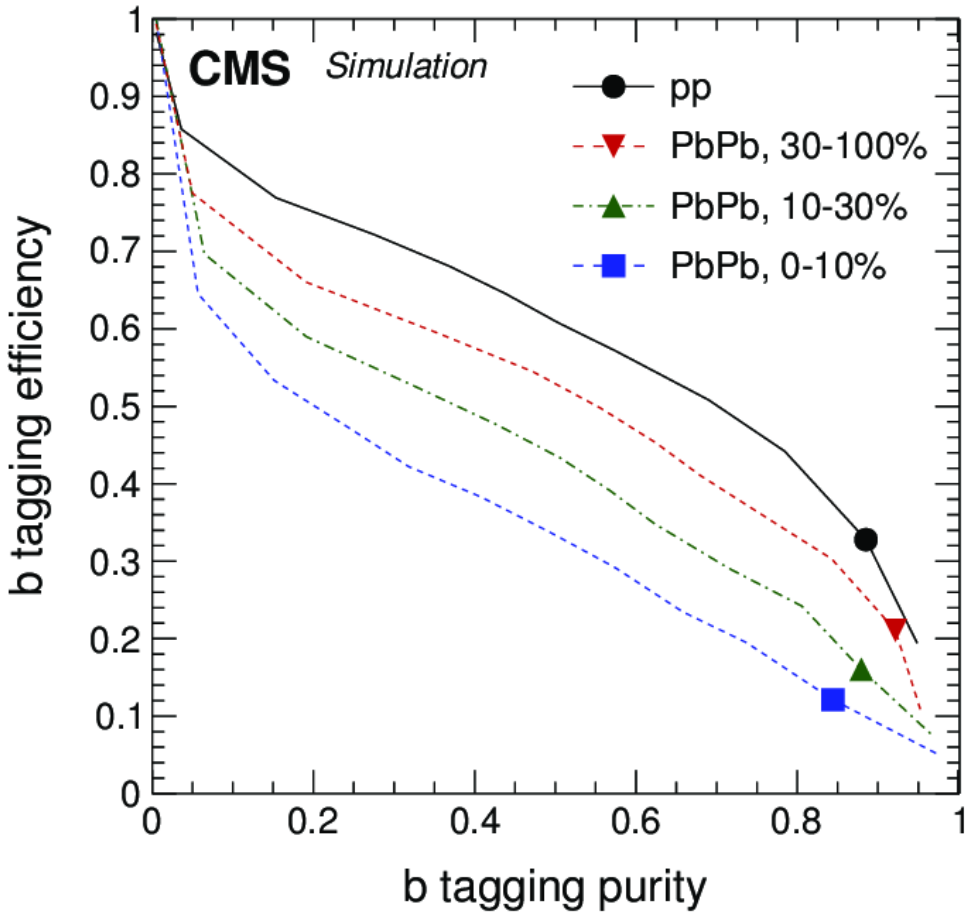
Many possible metrics, we will consider only the most simple ones

Efficiency vs purity

Balancing between signal efficiency (loose cuts to avoid loosing signal) and sample purity (strong cuts to reject as much as possible the background)

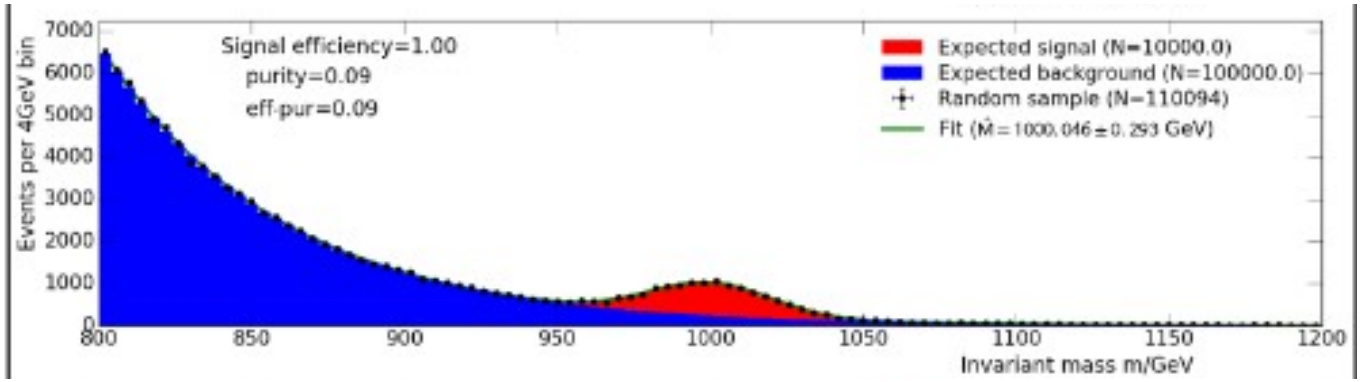
$$\epsilon = \frac{N^{selected}_{signal}}{N_{signal}}$$

$$p = \frac{N^{selected}_{signal}}{N^{selected}}$$



Significance

Given a number of signal events S and a number of background event B and assuming Poissonian uncertainty on background sqrt(B)



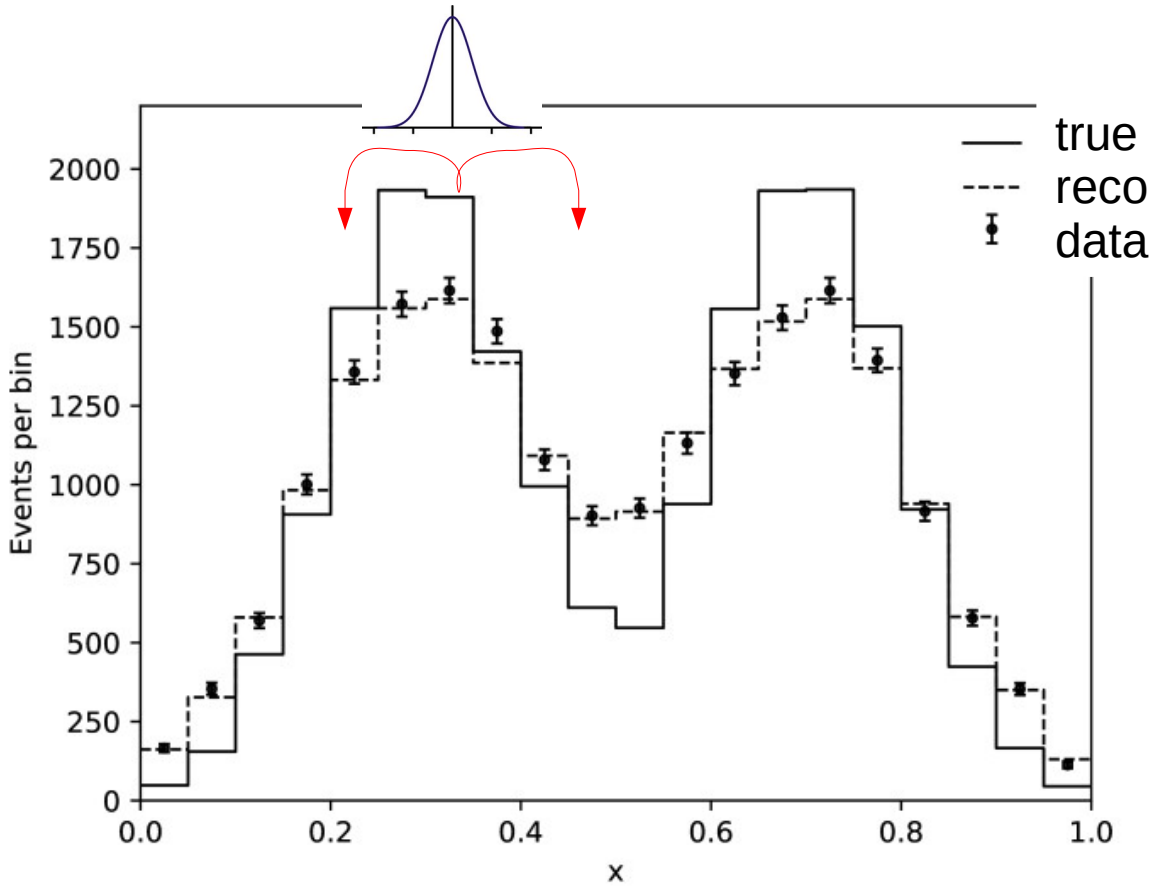
The significance of your expected signal over background is $\frac{S}{\sqrt{B}}$

If your background has large uncertainty beyond just statistical (δB) then the significance of your signal signal is $\frac{S}{\sqrt{\delta B^2 + B}}$

Why? Derivation and more complex (correct!) formulas here:
https://www.pp.rhul.ac.uk/~cowan/stat/cowan_munich16.pdf

Resolution

Back to histograms: **observed distribution = true distribution smeared, typically can be described by convolving with a Gaussian.**



Resolution: Gaussian smearing

Observed distribution = true distribution smeared, typically can be described by convolving with a Gaussian. Why?

$$f(x) = \frac{1}{\sigma\sqrt{2\,\pi}}\,\mathrm{e}^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

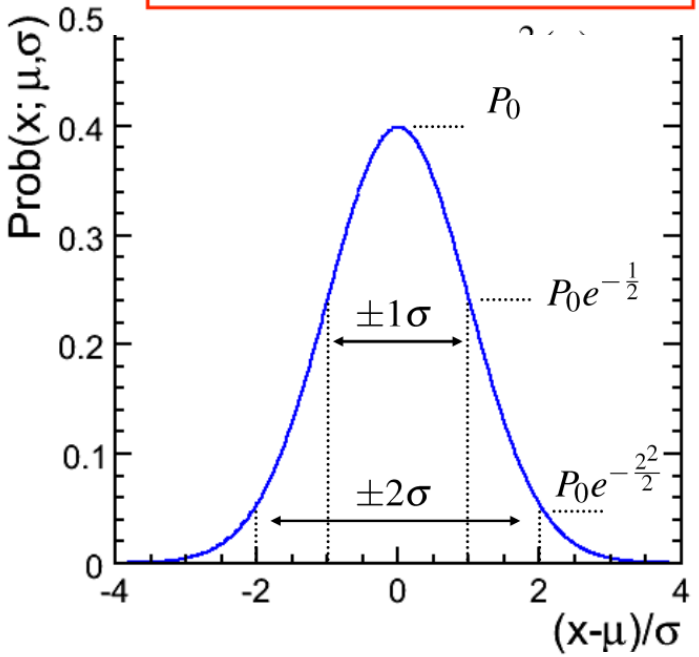
- mean μ (eg μ !=0 bias in momentum scale)
- variance σ = resolution on momentum

- Gaussian distribution describe the distribution of the sum of many *random* observables, whatever distribution of those observables is. (CENTRAL LIMIT THEOREM)

The detector smearing is fundamentally a probabilistic way to describe our limited knowledge of many detector effects so, for large enough number of events, should follow a Gaussian distribution

Gaussian and chi-square

$$G(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$



★ **Natural to introduce** $\chi^2(x)$

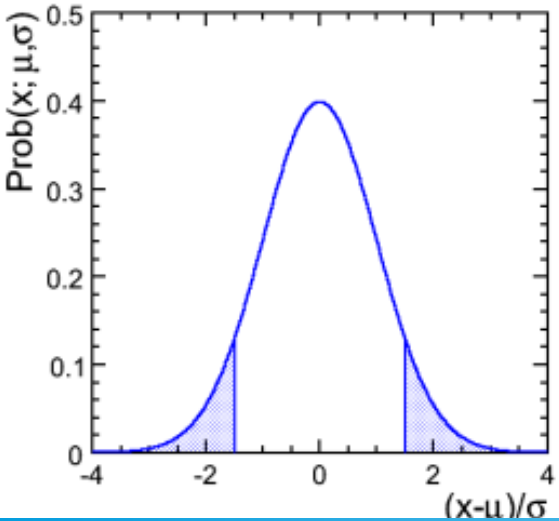
$$\chi^2 = \frac{(x-\mu)^2}{\sigma^2}$$

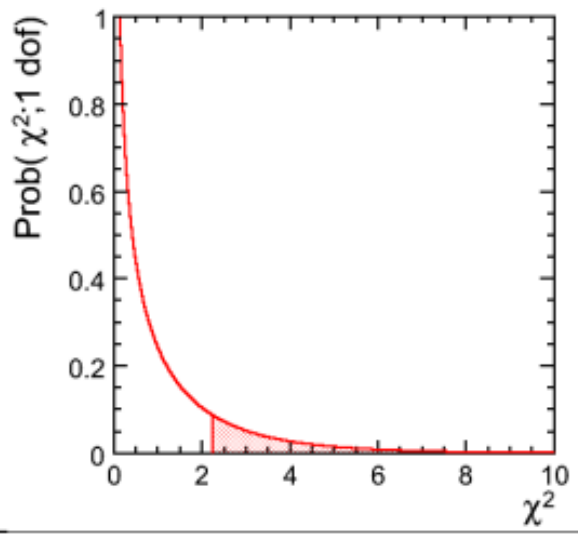
"squared deviation from mean in terms of standard error"

$$G(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{\chi^2}{2}\right)$$

★ **Fractions of events**

$68.3\% : \lvert x-\mu\rvert < 1\sigma$	$(\chi^2 < 1)$
$95.5\% : \lvert x-\mu\rvert < 2\sigma$	$(\chi^2 < 4)$
$99.7\% : \lvert x-\mu\rvert < 3\sigma$	$(\chi^2 < 9)$
$6\times10^{-7} : \lvert x-\mu\rvert > 5\sigma$	$(\chi^2 > 25)$





$$P(\chi^2) = \frac{1}{\sqrt{2\pi}} (\chi^2)^{-\frac{1}{2}} \exp\left\{-\frac{\chi^2}{2}\right\}$$

Why?
All derivations here (*)

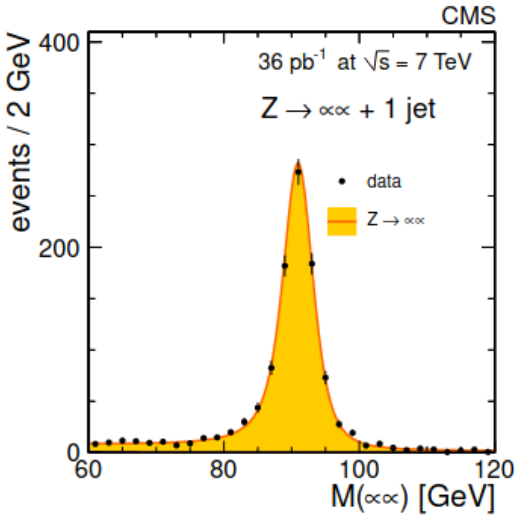
Resolution: unfolding

- How to correct back from observed to "true" → deconvolving detector effects
With histograms is basically an algebric problem

$$N_j^{reco} = N_i^{true} \cdot M_{ij}^{simu}$$ where M_{ji} is a matrix which gives the probability for an event in true bin i to be reconstructed in bin j

Such matrix can be evaluated from MC (typically with cross-check, tuning from control samples)

Eg: Z → μμ mass peak width can be used to tune the MC resolution to match with data



- Unfolding consists in inverting such matrix

$$N_i^{true} = N_j^{reco} \cdot M_{ij}^{-1}$$

See you Tomorrow