Statistical Methods in Particle Physics

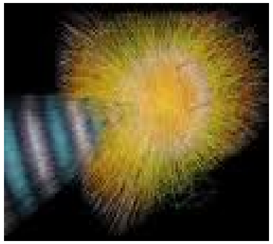
Lecture 4

November 5, 2012

Silvia Masciocchi, GSI Darmstadt
s.masciocchi@gsi.de

Winter Semester 2012 / 13

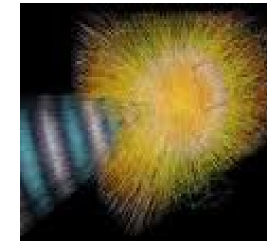
Outline



- Short reminders
- Error propagation
- Correlation between variables

- Monte Carlo methods
  - Transformation method
  - Integration
- Monte Carlo for particle / nuclear physics
  - Event generators
  - Detector simulation

Quick reminders



- **Mean or expectation value**

$$E[x] = \int x f(x)\,dx = \mu$$

- **Variance:**

$$V[x] = E[(x - E[x])^2] = E[x^2] - \mu^2 = \sigma^2$$

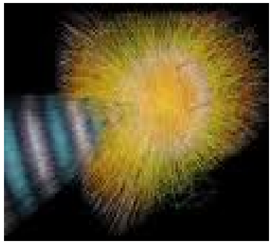
- **Standard deviation:**

$$\sigma = \sqrt{\sigma^2}$$

- **Covariance** $\mathrm{cov}[x, y] = E[(x - \mu_x)(y - \mu_y)]$

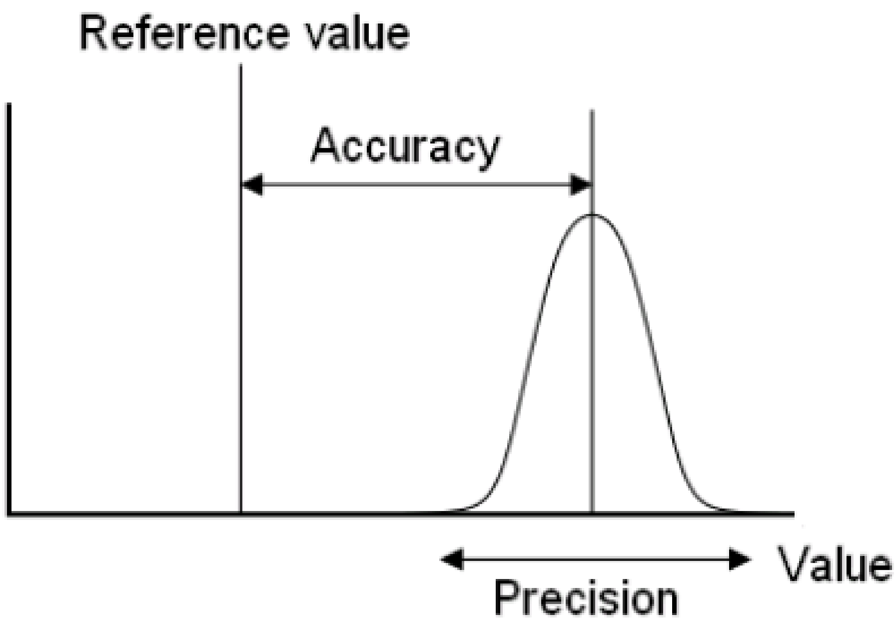
- **Correlation coefficient** $\rho_{xy} = \dfrac{\mathrm{cov}[x, y]}{\sigma_x \sigma_y}, \quad -1 \le \rho_{xy} \le +1$

Accuracy and precision

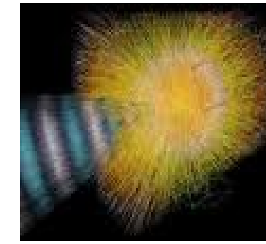


The **accuracy** of a measurement system is the degree of closeness of measurements of a quantity to its true (actual) value.

The **precision** of a measurement system, also called reproducibility or repeatability, is the degree to which repeated measurements under unchanged conditions show the same results.



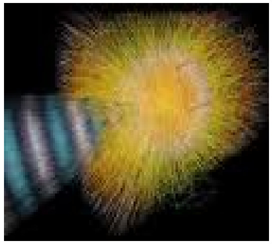
Accuracy and precision



A measurement system can be accurate but not precise, precise but not accurate. See the grouping of arrows on a target:

Fig 3: A target analogy for the comparison of accuracy and precision. Arrows that strike closer to the bullseye are considered more accurate. if a large number of arrows are shot, precision would be the size of the arrow cluster.

Error propagation



Function of ONE variable:

- Variable x, with mean $\bar{x}$, and uncertainty σ_x

- Function y = f(x)

  We want to determine the uncertainty on y

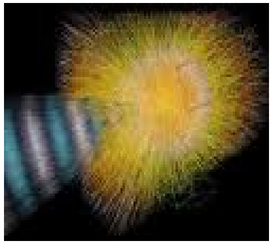
- Use the Taylor expansion:

$$f(x) = f(\bar{x}) + (x - \bar{x}) \left(\frac{df}{dx}\right)_{\bar{x}} + \text{higher order terms}$$

We ignore the higher order terms WHEN the measured values are close to the average values and/or the derivative is constant in the region of interest (see in 3 slides)

Not always the case !!!!

Error propagation: one variable



- It follows that: $f(x) - f(\bar{x}) = y - \bar{y} \approx (x - \bar{x}) \left(\frac{df}{dx}\right)_{\bar{x}}$
- Variance:

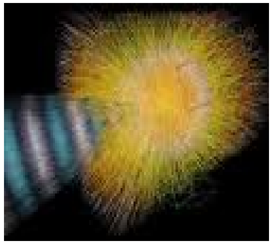
$$V[y] = \langle (y - \bar{y})^2 \rangle = \langle (x - \bar{x})^2 \rangle \left(\frac{df}{dx}\right)_{\bar{x}}^2$$

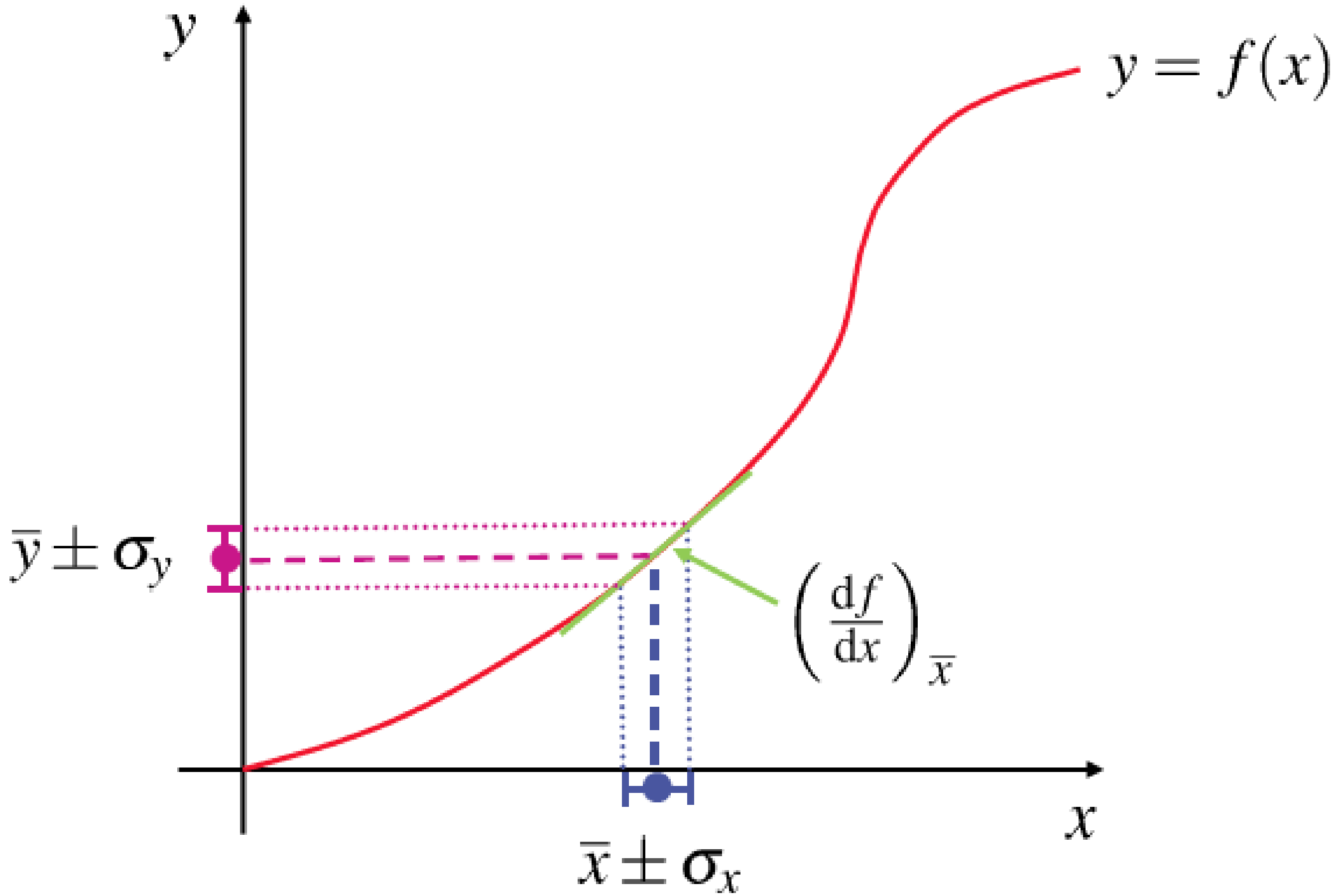
$$\sigma_y^2 = \left(\frac{df}{dx}\right)_{\bar{x}}^2 \sigma_x^2$$

- Standard deviation

$$\sigma_y = \left(\frac{df}{dx}\right)_{\bar{x}} \sigma_x$$

Error propagation: one variable

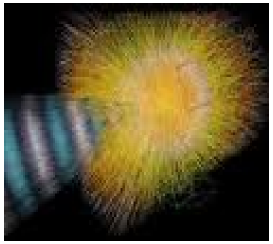




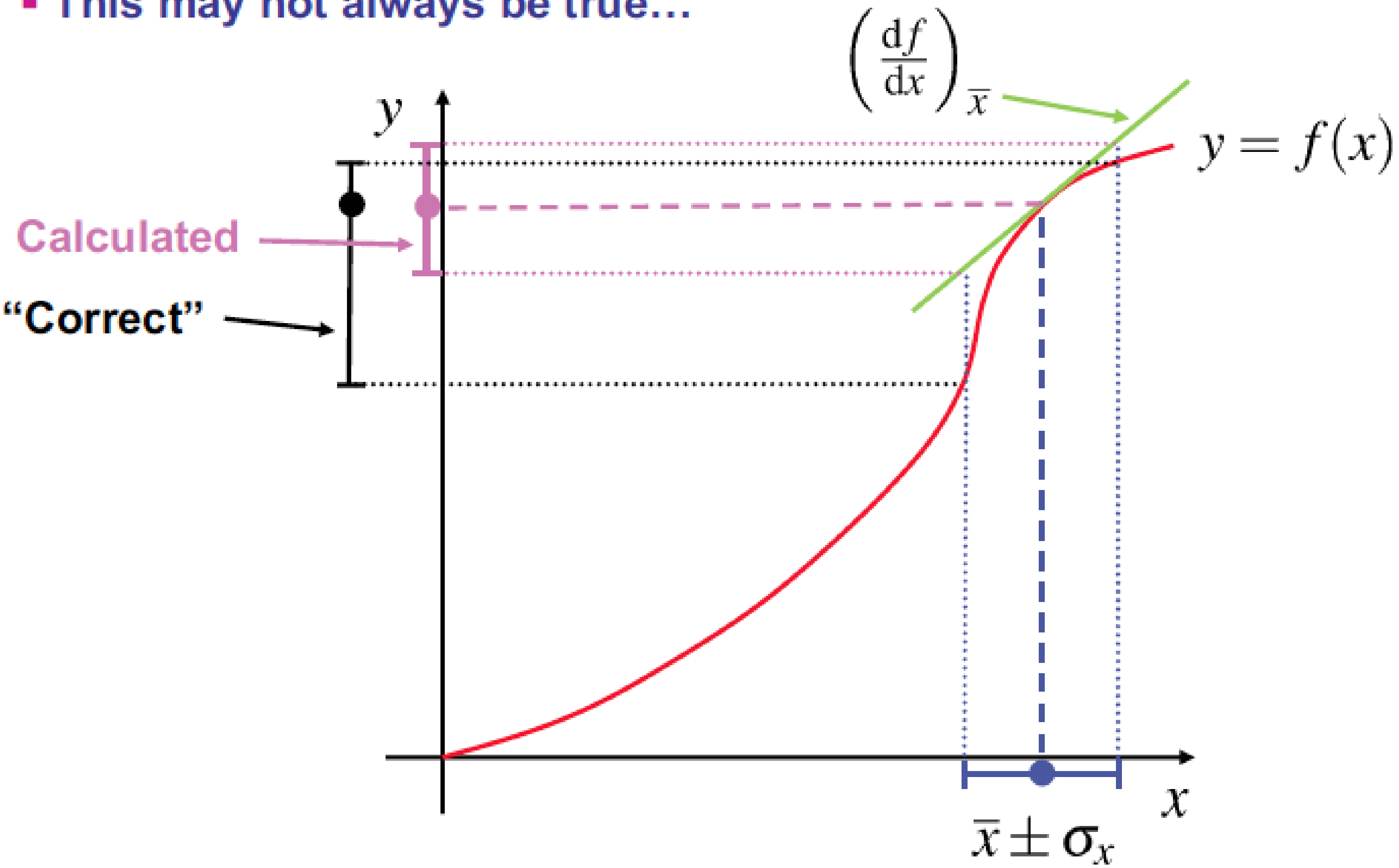
★ **How does a "small" change in x, i.e. σ_x, propagate to a small change in y, σ_y**

$$\frac{\sigma_y}{\sigma_x} \approx \left(\frac{dy}{dx}\right)_{\bar{x}}$$

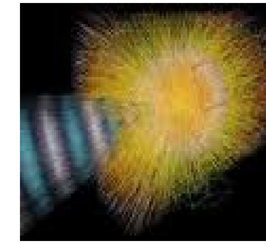
Ignoring higher order terms



- Neglected second order terms in the Taylor expansion
- This is equivalent to saying that the derivative is constant in region of interest
- This may not always be true…



Exercise



- Measurement of transverse momentum of a track from a fit
  - radius of curvature of track helix, ***R***, given by

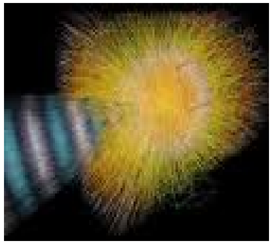
$$R = 0.3B(\mathrm{T})p_{\mathrm{T}}(\mathrm{GeV})$$

  - track fit returns a Gaussian uncertainty in radius of curvature, and hence, the PDF is Gaussian in $1/p_{\mathrm{T}}$

$$\sigma_{1/p_{\mathrm{T}}}$$

  - what is the error in p_{T}

Exercise



- Measurement of transverse momentum of a track from a fit
  - radius of curvature of track helix, ***R***, given by

$$R = 0.3B(\mathrm{T})p_{\mathrm{T}}(\mathrm{GeV})$$

  - track fit returns a Gaussian uncertainty in radius of curvature, and hence, the PDF is Gaussian in $1/p_{\mathrm{T}}$

$$\sigma_{1/p_{\mathrm{T}}}$$

  - what is the error in p_{T}

let $x = 1/p_{\mathrm{T}}$

$$p_{\mathrm{T}} = 1/x$$

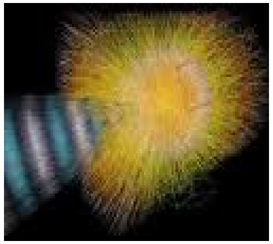
$$\frac{\mathrm{d}p_{\mathrm{T}}}{\mathrm{d}x} = -\frac{1}{x^2} = -p_{\mathrm{T}}^2$$

$$\sigma_{p_{\mathrm{T}}}^2 = \left(\frac{\mathrm{d}p_{\mathrm{T}}}{\mathrm{d}x}\right)^2 \sigma_x^2$$

$$\sigma_{p_{\mathrm{T}}}^2 = \left(p_{\mathrm{T}}^2\right)^2 \sigma_x^2$$

$$\sigma_{p_{\mathrm{T}}} = p_{\mathrm{T}}^2 \sigma_{1/p_{\mathrm{T}}}$$

Error propagation: two variables



$$a = f(x,y)$$

$$a = f(x,y) = f(\bar{x},\bar{y}) + \frac{df}{dx}(x - \bar{x}) + \frac{df}{dy}(y - \bar{y}) + \ldots$$

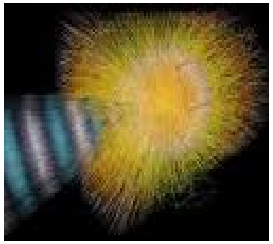
$$\sigma_a^2 = (a - \bar{a})^2 = (f(x,y) - f(\bar{x},\bar{y}))^2 =$$

$$\approx \left(\frac{df}{dx}\right)^2 (x - \bar{x})^2 + \left(\frac{df}{dy}\right)^2 (y - \bar{y})^2 + 2\frac{df}{dx}\frac{df}{dy}(x - \bar{x})(y - \bar{y})$$

$$\sigma_a^2 = \langle (a - \bar{a})^2 \rangle =$$

$$\left(\frac{df}{dx}\right)^2 \langle (x - \bar{x})^2 \rangle + \left(\frac{df}{dy}\right)^2 \langle (y - \bar{y})^2 \rangle + 2\frac{df}{dx}\frac{df}{dy}\langle (x - \bar{x})(y - \bar{y}) \rangle =$$

Error propagation: two variables

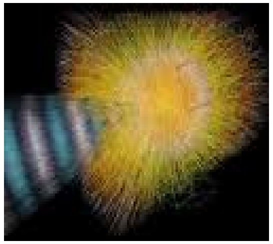


$$\sigma_a^2 = \left(\frac{df}{dx}\right)^2 \sigma_x^2 + \left(\frac{df}{dy}\right)^2 \sigma_y^2 + 2\frac{df}{dx}\frac{df}{dy}\frac{\mathrm{cov}[x,y]}{\sigma_x\sigma_y}\sigma_x\,\sigma_y$$

$$\frac{\mathrm{cov}[x,y]}{\sigma_x\sigma_y} = \rho = \text{correlation coefficient}$$

- $-1 \leq \rho \leq +1$
- $\rho = 0$: variables are INDEPENDENT
- $\rho \neq 0$: variables are CORRELATED
  - $\rho > 0$: correlated
  - $\rho < 0$: anti-correlated

Independent variables

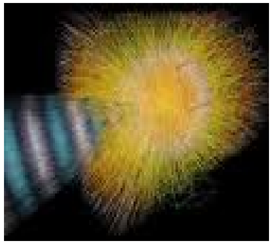


$$\sigma_f^2 = \left(\frac{df}{dx}\right)^2 \sigma_x^2 + \left(\frac{df}{dy}\right)^2 \sigma_y^2$$

Important examples:

Function	Derivative(s)	Variance	Standard deviation
$f = kx$; $k \varepsilon \mathrm{R}$	$\frac{\partial f}{\partial x} = k$	$\sigma_f^2 = k^2 \sigma_x^2$	$\sigma_f = k\sigma_x$
$f = x + y$	$\frac{\partial f}{\partial x} = 1$ and $\frac{\partial f}{\partial y} = 1$	$\sigma_f^2 = \sigma_x^2 + \sigma_y^2$	$\sigma_f = \sqrt{\sigma_x^2 + \sigma_y^2}$
$f = x - y$	$\frac{\partial f}{\partial x} = 1$ and $\frac{\partial f}{\partial y} = -1$	$\sigma_f^2 = \sigma_x^2 + \sigma_y^2$	$\sigma_f = \sqrt{\sigma_x^2 + \sigma_y^2}$
$f = xy$	$\frac{\partial f}{\partial x} = y$ and $\frac{\partial f}{\partial y} = x$	$\left(\frac{\sigma_f}{f}\right)^2 = \left(\frac{\sigma_x}{x}\right)^2 + \left(\frac{\sigma_y}{y}\right)^2$	$\sigma_f = f\sqrt{\left(\frac{\sigma_x}{x}\right)^2 + \left(\frac{\sigma_y}{y}\right)^2}$
$f = x / y$	$\frac{\partial f}{\partial x} = \frac{1}{y}$ and $\frac{\partial f}{\partial y} = -\frac{x}{y^2}$	$\left(\frac{\sigma_f}{f}\right)^2 = \left(\frac{\sigma_x}{x}\right)^2 + \left(\frac{\sigma_y}{y}\right)^2$	$\sigma_f = f\sqrt{\left(\frac{\sigma_x}{x}\right)^2 + \left(\frac{\sigma_y}{y}\right)^2}$

Error propagation: SPECIAL CASES



$$y = x_1 + x_2 \quad \rightarrow \quad \sigma_y^2 = \sigma_1^2 + \sigma_2^2 + 2\mathrm{cov}[x_1, x_2]$$

$$y = x_1 \, x_2 \quad \rightarrow \quad \frac{\sigma_y^2}{y^2} = \frac{\sigma_1^2}{x_1^2} + \frac{\sigma_2^2}{x_2^2} + 2\frac{\mathrm{cov}[x_1, x_2]}{x_1 x_2}$$

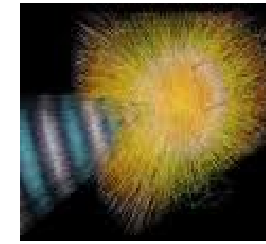
That is, **if the x_i are uncorrelated:**

Add errors quadratically for the sum (or difference),

Add relative errors quadratically for product (or ratio)

correlations can change this completely...

Error propagation – MORE SPECIAL



Consider $y = x_1 - x_2$ with:

$$\mu_1 = \mu_2 = 10, \quad \sigma_1 = \sigma_2 = 1, \quad \rho = \frac{\text{cov}[x_1, x_2]}{\sigma_1 \sigma_2} = 0$$

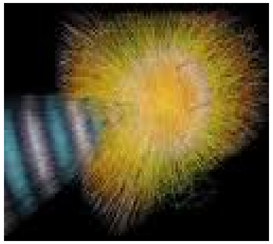
$$V[y] = 1^2 + 1^2 = 2 \quad \rightarrow \quad \sigma_y = 1.4$$

Now suppose ρ=1 (full correlation). Then:

$$V[y] = \text{١}^{2} + \text{١}^{\text{٢}} - \text{٢} = \text{٠} \quad \rightarrow \quad \sigma_y = 0$$

i.e. for 100% correlation, the error in the difference goes to 0 !!

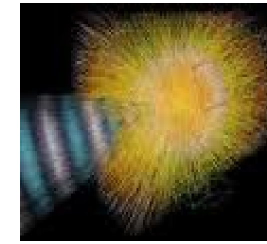
Exercise



We wish to calculate the average speed (displacement/time) of an object. Assume is displacement is measured as $x = 22.2 \pm 0.5$ cm during the time interval $t = 9.0 \pm 0.1$ s.

Calculate speed and error on the speed.

Exercise



We wish to calculate the average speed (displacement/time) of an object. Assume is displacement is measured as x = 22.2 ± 0.5 cm during the time interval t = 9.0 ± 0.1 s.

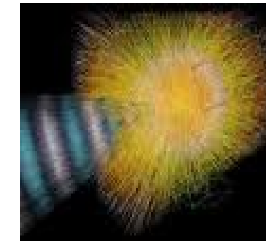
Calculate speed and error on the speed.

$$v = \frac{x}{t} = \frac{22.2}{9.0} = 2.467 \ \text{cm/s}$$

$$\sigma_v = v\sqrt{\left(\frac{\sigma_x}{x}\right)^2 + \left(\frac{\sigma_t}{t}\right)^2} = 2.467\sqrt{\left(\frac{0.5}{22.2}\right)^2 + \left(\frac{0.1}{9.0}\right)^2} = 0.062 \ \text{cm/s}$$

We report the result as: 2.467 ± 0.062 cm/s.

Combination of results



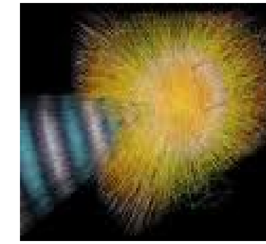
Important case: combine the results of independent experiments

Consider we have n independent experiments with results ai and errors σ_i (i = 1, ..., n). We can combine the results from each experiment to form a more accurate result. For this, a weighted sum is performed where experiments with smaller errors contribute more to the combined result.

The statistically correct way to combine independent results is:

$$a = \frac{\sum a_i/\sigma_i^2}{\sum 1/\sigma_i^2} \quad \text{and} \quad \sigma = \frac{1}{\sqrt{\sum 1/\sigma_i^2}}$$

Exercise



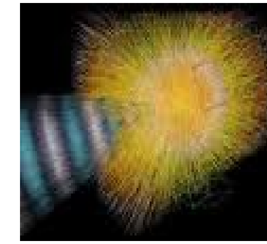
- For the measured gravitational acceleration data:
  9.77±0.14, 9.82±0.10 and 9.86±0.20 m/s^2
  the combined result is:

- Example: Power in an electric circuit.

  $$P = I^2R$$

  - Let $I = 1.0 \pm 0.1$ amp and $R = 10 \pm 1\ \Omega$
    - $P = 10$ watts
  - calculate the variance in the power using propagation of errors

Exercise



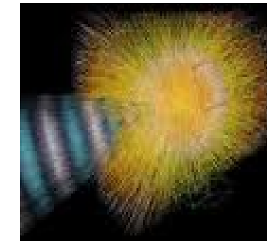
- For the measured gravitational acceleration data:
  9.77±0.14, 9.82±0.10 and 9.86±0.20 m/s^2
  the combined result is: 9.811 ± 0.075 m/s^2

- Example: Power in an electric circuit.
  $$P = I^2 R$$
  - Let $I = 1.0 \pm 0.1$ amp and $R = 10 \pm 1\ \Omega$
    - $P = 10$ watts
  - calculate the variance in the power using propagation of errors

$$\sigma_P^2 = \sigma_I^2 \left(\frac{\partial P}{\partial I}\right)^2_{I=1} + \sigma_R^2 \left(\frac{\partial P}{\partial R}\right)^2_{R=10} = \sigma_I^2 (2IR)^2 + \sigma_R^2 (I^2)^2 = (0.1)^2 (2 \cdot 1 \cdot 10)^2 + (1)^2 (1^2)^2 = 5 \text{ watts}^2$$

- $P = 10 \pm 2$ watts

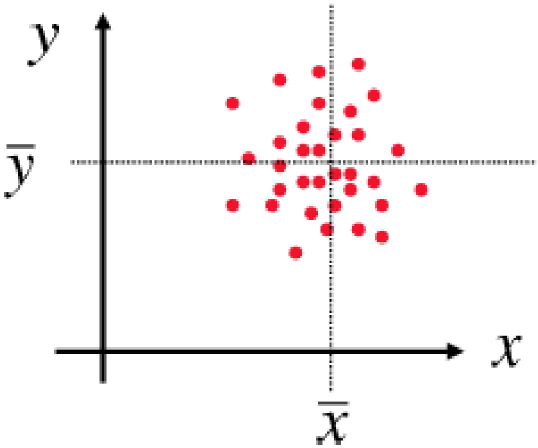
Correlated variables



The covariance matrix and the correlation coefficient express to what extent 2 or more variables "co-vary" randomly, or whether, when one has a given variation, the second one varies by a corresponding quantity / way of behaviour.

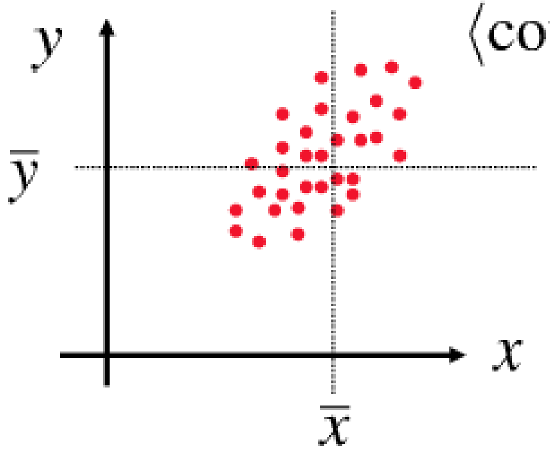
x and y are uncorrelated, i.e. INDEPENDENT

$$\langle \mathrm{cov}(x,y) \rangle = 0$$



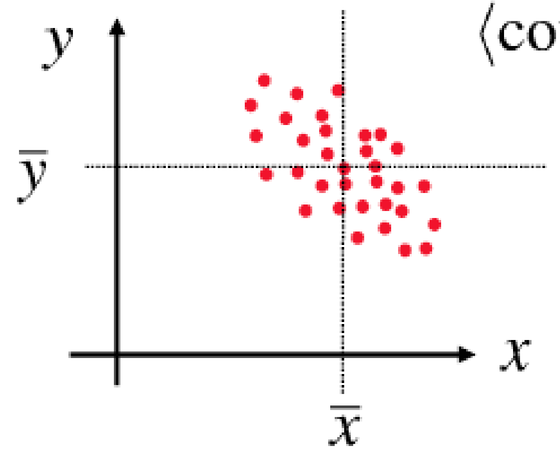
- If x and y are correlated

$$\langle \mathrm{cov}(x,y) \rangle > 0$$

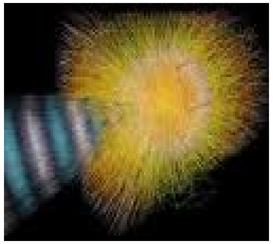


- If x and y are anti-correlated

$$\langle \mathrm{cov}(x,y) \rangle < 0$$



More correlation



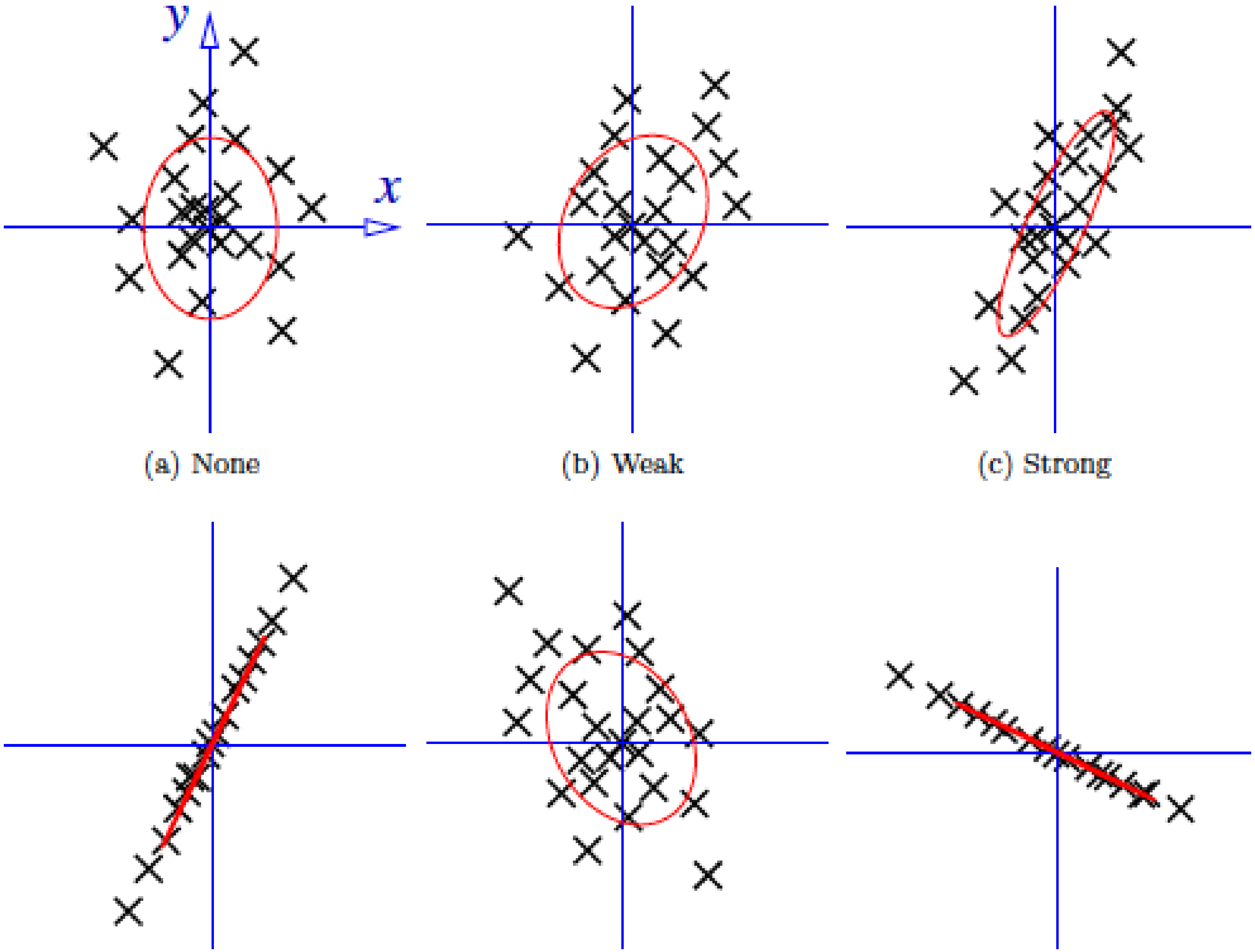
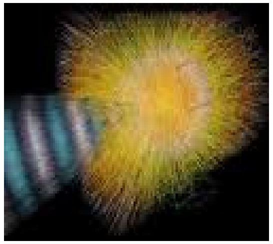


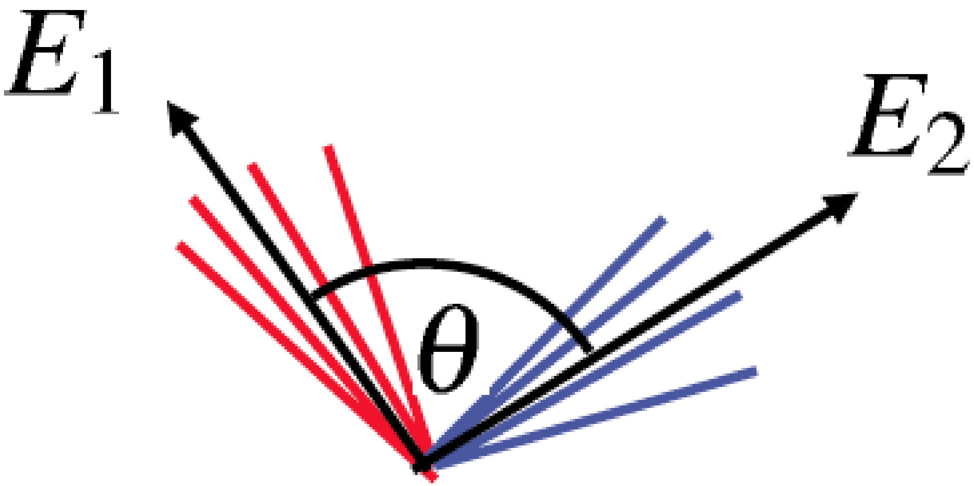
Figure 2: Examples of correlations and anticorrelations

Example



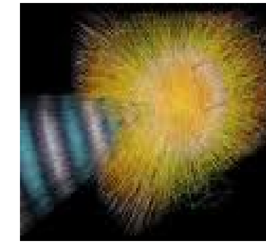
Consider pairs of jets in one event:

From the total energy of
each jet, the di-jet
invariant mass can be calculated



$$m^2 = E_1 E_2 (1 - \cos\theta)$$

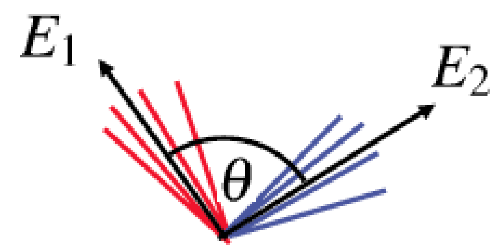
Origin of correlation

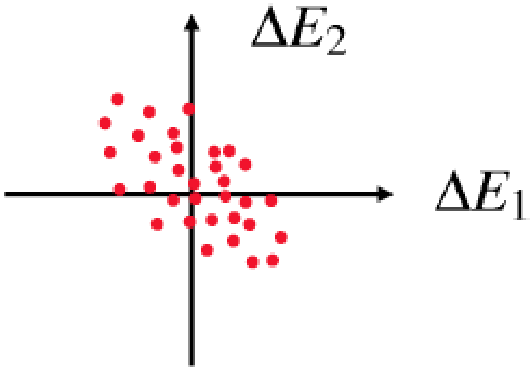


★ Correlations can arise from physical effects, e.g.

- Would expect E_1 and E_2 to be (slightly) anti-correlated why ?
- Can always check (in MC) by plotting

$\Delta E_1 = E_1 - E_1^{MC}$ against $\Delta E_2 = E_2 - E_2^{MC}$





$$\rho = \frac{\mathrm{cov}(x,y)}{\sigma_x \sigma_y}$$

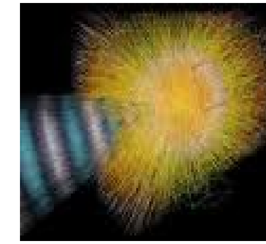
$$\mathrm{cov}(x,y) = \langle (x-\bar{x})(y-\bar{y}) \rangle$$
$$\sigma_x = \langle (x-\bar{x})^2 \rangle^{\frac{1}{2}}$$

NOTE: uncertainty on correlation coefficient $s_\rho \approx \dfrac{(1-\rho^2)}{\sqrt{n-2}}$

★ Correlations also arise when calculating derived quantities from uncorrelated measurements

- e.g. $x = a + b \quad y = a - b$
- this type of correlation can be handled mathematically

Error propagation: general case



Suppose we measure a set of values $\vec{x} = (x_1, ..., x_n)$

which follow some joint pdf $f(\vec{x})$.

$f(\vec{x})$ might be not fully known. But we have the covariances:

$V_{ij} = \mathrm{cov}[x_i, x_j]$, and the means $\vec{\mu} = E[\vec{x}]$ (in practice only estimates)

Now consider a function $y(\vec{x})$.

What is the variance of $y(\vec{x})$?

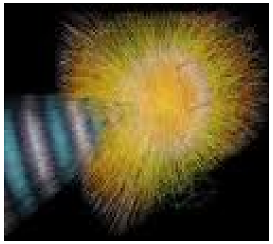
Hard way: use joint pdf $f(\vec{x})$ to find the pdf $g(y)$,

Then from $g(y)$ find

$$V[y] = E[y^2] - (E[y])^2$$

Often NOT practical. $f(\vec{x})$ may not even be fully known ...

Error propagation - 2



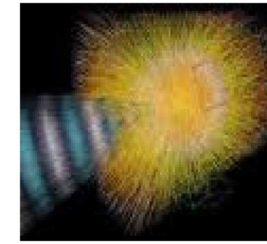
Expand $y(\vec{x})$ to the first order in a Taylor series about $\vec{\mu}$

$$y(\vec{x}) \approx y(\vec{\mu}) + \sum_{i=1}^{n} \left[\frac{\partial y}{\partial x_i}\right]_{\vec{x}=\vec{\mu}} (x_i - \mu_i)$$

To find the variance V[y] we need $E[y^2]$ and E[y]:

$$E[y(\vec{x})] \approx y(\vec{\mu}) \quad \text{since} \quad E[x_i - \mu_i] = 0$$

Error propagation - 3



$$E[y^2(\vec{x})] \approx y^2(\vec{\mu}) + 2y(\vec{\mu}) \sum_{i=1}^{n} \left[\frac{\partial y}{\partial x_i}\right]_{\vec{x}=\vec{\mu}} E[x_i - \mu_i]$$

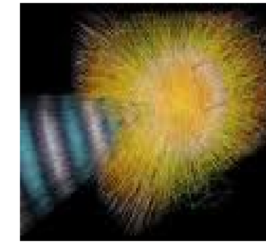
$$+ \; E\left[\left(\sum_{i=1}^{n} \left[\frac{\partial y}{\partial x_i}\right]_{\vec{x}=\vec{\mu}} (x_i - \mu_i) \right) \left(\sum_{j=1}^{n} \left[\frac{\partial y}{\partial x_j}\right]_{\vec{x}=\vec{\mu}} (x_j - \mu_j) \right) \right]$$

$$= \; y^2(\vec{\mu}) + \sum_{i,j=1}^{n} \left[\frac{\partial y}{\partial x_i}\frac{\partial y}{\partial x_j}\right]_{\vec{x}=\vec{\mu}} V_{ij}$$

Putting the ingredients together gives the variance of $y(\vec{x})$

$$\sigma_y^2 \approx \sum_{i,j=1}^{n} \left[\frac{\partial y}{\partial x_i}\frac{\partial y}{\partial x_j}\right]_{\vec{x}=\vec{\mu}} V_{ij}$$

Error propagation - 4



If the x_i are uncorrelated, i.e. $V_{ij} = \sigma^2_i \delta_{ij}$, then this becomes:

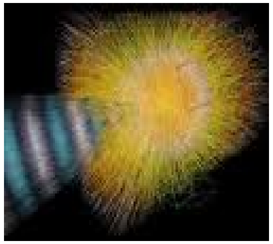
$$\sigma_y^2 \approx \sum_{i=1}^{n} \left[\frac{\partial y}{\partial x_i}\right]^2_{\vec{x}=\vec{\mu}} \sigma_i^2$$

Similar for a set of m functions $\vec{y}(\vec{x}) = (y_1(\vec{x}), \ldots, y_m(\vec{x}))$

$$U_{kl} = \mathrm{cov}[y_k, y_l] \approx \sum_{i,j=1}^{n} \left[\frac{\partial y_k}{\partial x_i} \frac{\partial y_l}{\partial x_j}\right]_{\vec{x}=\vec{\mu}} V_{ij}$$

Or in matrix notation $U = A V A^T$, where $A_{ij} = \left[\frac{\partial y_i}{\partial x_j}\right]_{\vec{x}=\vec{\mu}}$

Monte Carlo

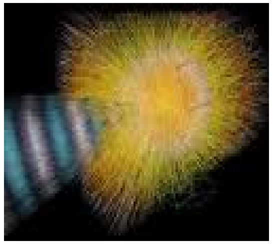


Monte Carlo methods are a class of computational algorithms that rely on repeated random sampling to compute their results.

Monte Carlo methods are often used in computer simulations of physical and mathematical systems.

These methods are most suited to calculation by a computer and tend to be used when it is unfeasible to compute an exact result with a deterministic algorithm

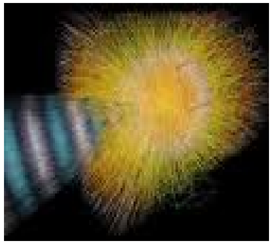
The name



The Monte Carlo method was coined in the **1940s** by John von Neumann, Stanislaw Ulam and Nicholas Metropolis, while they were working on **nuclear weapon projects (Manhattan Project)** in the Los Alamos National Laboratory. It was named in homage to the **Monte Carlo Casino**, a famous casino, where Ulam's uncle would often gamble away his money

Random processes were used extensively for the first time to predict theoretically the interaction of neutrons with matter

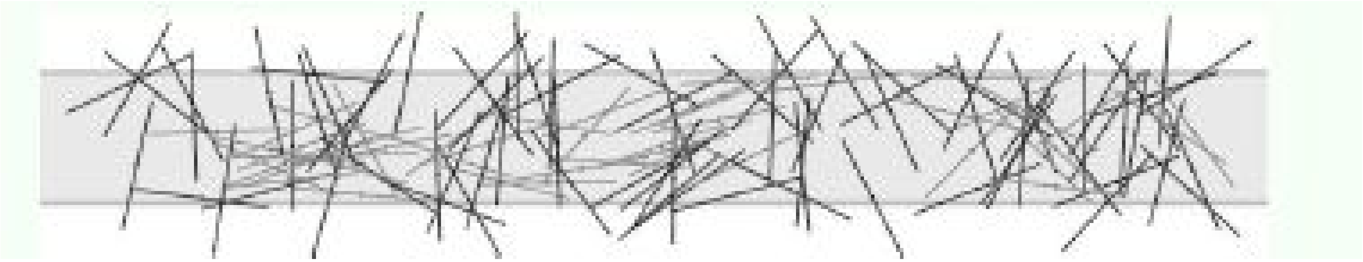
History - 1



An early variant of the Monte Carlo method can be seen in the

Buffon's needle experiment (18th century)

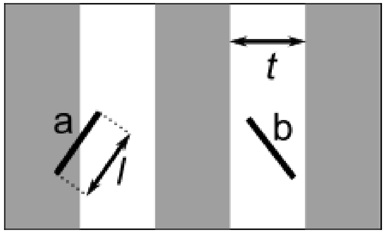
in which π can be estimated by dropping needles on a floor made of parallel strips of wood.



In more mathematical terms:

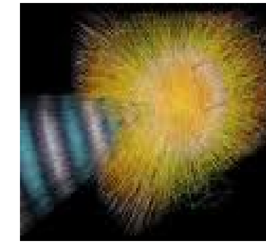
Given a needle of length l dropped on a plane ruled with parallel lines t units apart, what is the probability that the needle will cross a line?

If $l \leq t$ then: $$P = \frac{2l}{t\pi}$$



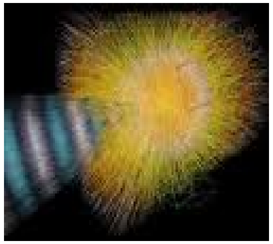
Use this to estimate π !

History - 2



- In the 1930s, Enrico Fermi first experimented with the Monte Carlo method while studying neutron diffusion, but did not publish anything on it
- In 1946, physicists at Los Alamos Scientific Laboratory were investigating radiation shielding and the distance that neutrons would likely travel through various materials. Despite having most of the necessary data, such as the average distance a neutron would travel in a substance before it collided with an atomic nucleus or how much energy the neutron was likely to give off following a collision, the problem **could not be solved with analytical calculations**. Stanisław Ulam had the idea of using random experiments.

Monte Carlo and simulation



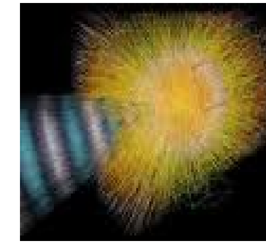
A bit subjective. Sawilowsky:

a simulation is a fictitious representation of reality, a Monte Carlo method is a technique that can be used to solve a mathematical or statistical problem, and a Monte Carlo simulation uses repeated sampling to determine the properties of some phenomenon (or behavior). Examples:

- Simulation: Drawing one pseudo-random uniform variable from the interval [0,1] can be used to simulate the tossing of a coin: If the value is less than or equal to 0.50 designate the outcome as heads, but if the value is greater than 0.50 designate the outcome as tails. This is a simulation, but not a Monte Carlo simulation.
- Monte Carlo method: The area of an irregular figure inscribed in a unit square can be determined by throwing darts at the square and computing the ratio of hits within the irregular figure to the total number of darts thrown. This is a Monte Carlo method of determining area, but not a simulation.
- Monte Carlo simulation: Drawing a large number of pseudo-random uniform variables from the interval [0,1], and assigning values less than or equal to 0.50 as heads and greater than 0.50 as tails, is a Monte Carlo simulation of the behavior of repeatedly tossing a coin.

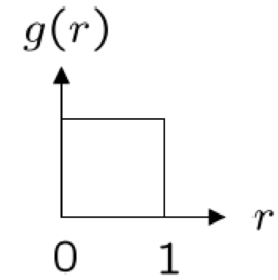
Not always so easy to distinguish

The Monte Carlo method



A numerical technique for calculating probabilities and related quantities using sequences of random numbers

The usual steps:

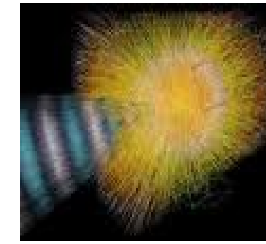


- Generate sequence $r_1, r_2, ..., r_m$ uniform in [0,1]
- Use this to produce another sequence $x_1, x_2, ..., x_m$ distributed according to some pdf f(x) in which we are interested (x can be a vector)
- Use the x values to estimate some property of f(x) e.g., fraction of x values with a < x < b gives $\int_a^b f(x)\,dx$

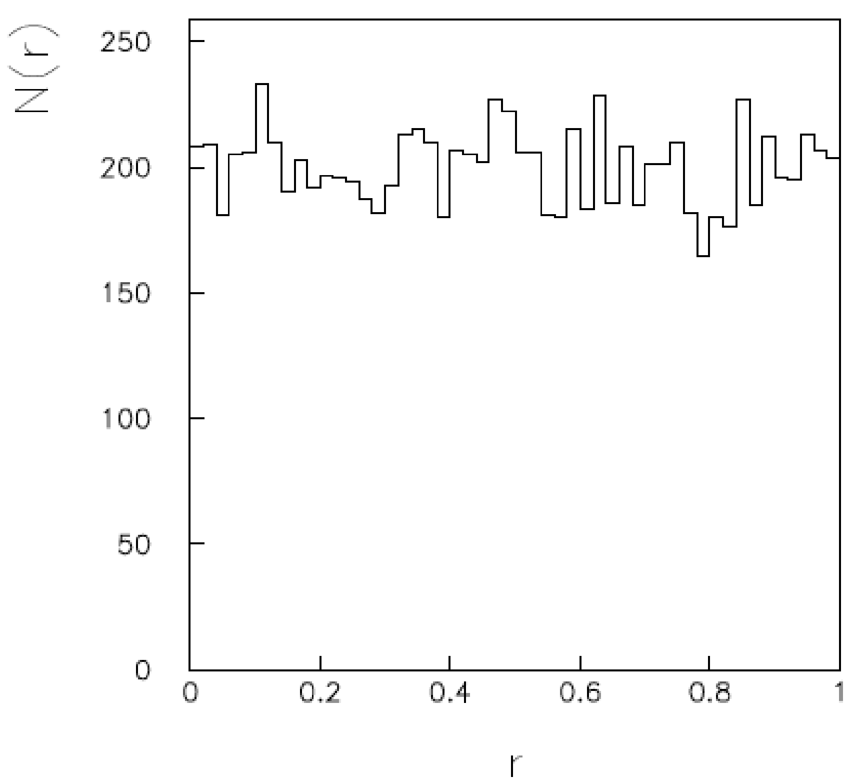
Applications:

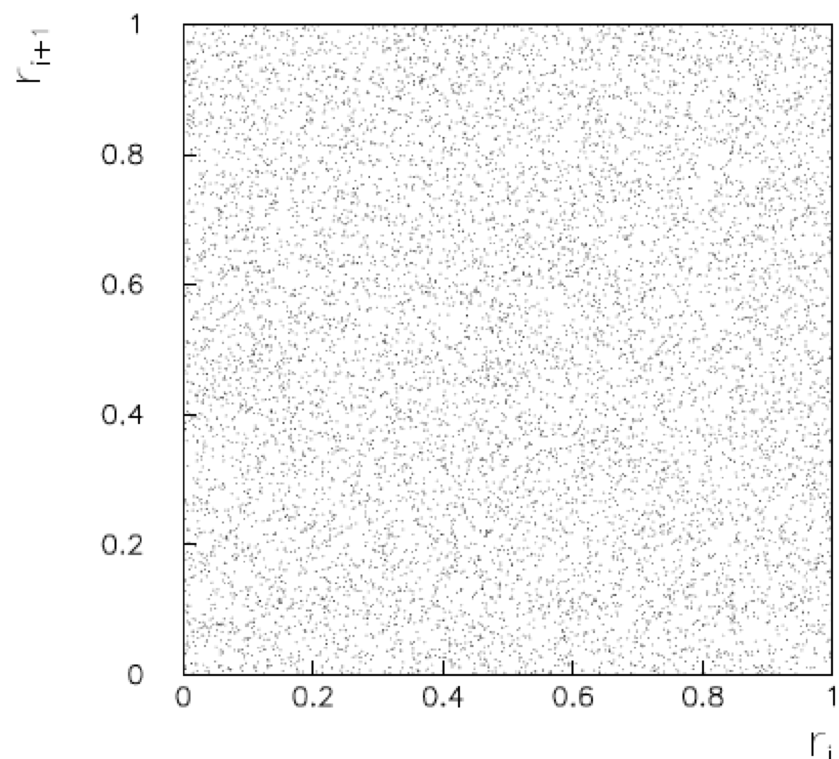
- MC calculation for integration
- Simulation to test statistical procedures

Random numbers

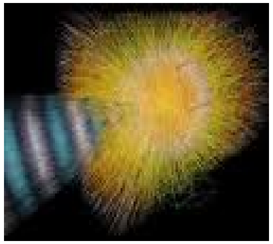


- Extensively appreciated with Nik



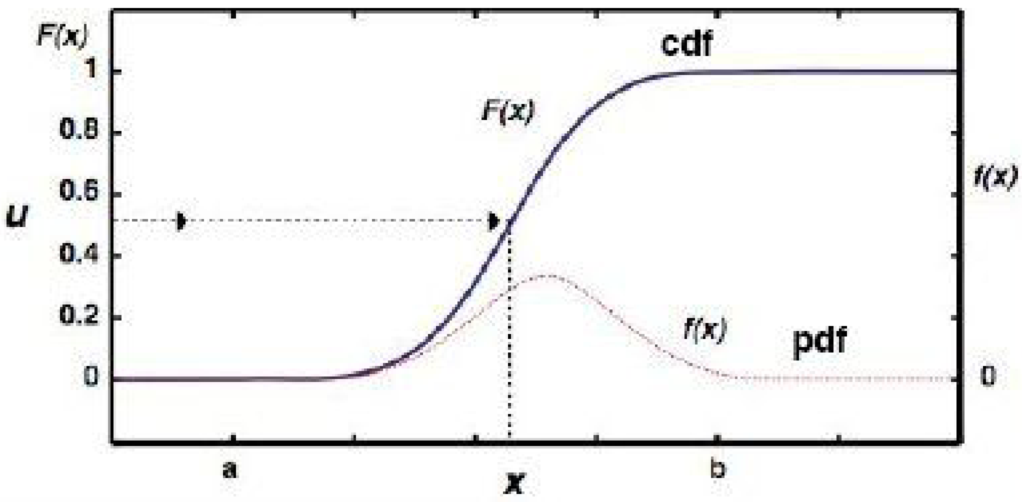


Transformation method



- Random variable x
- Pdf f(x)
- cumulative distribution function

$$F(x) = \int_{-\infty}^{x} f(t)\,dt$$



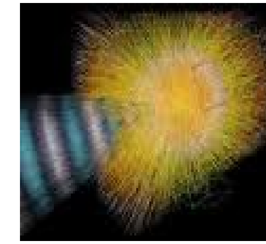
- Case of F(x) analitically invertible:

$$x = F^{-1}(u)$$

- If u_i uniform in [0,1], then $x_i = F^{-1}(u_i)$ follow pdf f(x)

Method is applicable if F(x) and $F^{-1}(u)$ are analytically solvable

Example: exponential distribution



Consider random numbers according to:

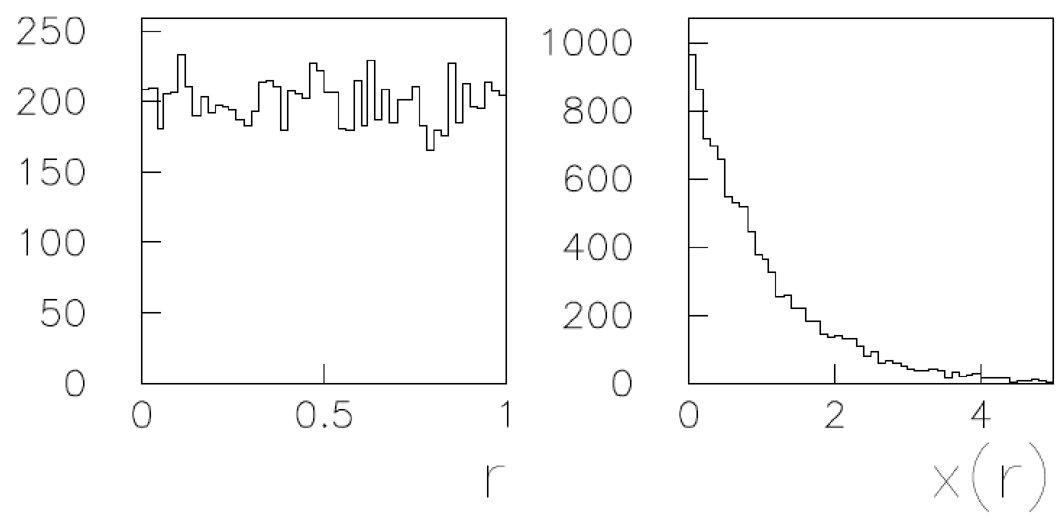
$$f(x,\lambda) = \begin{cases} \lambda e^{-\lambda x}, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

Computation of the inverse pdf:

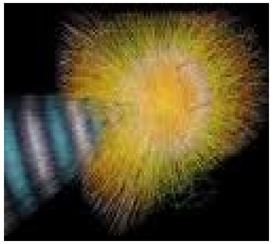
$$\int_{-\infty}^{x} f(t)\,dt = \int_{0}^{x} \lambda\, e^{-\lambda t}\, dt = [e^{-\lambda t}]_0^x = 1 - e^{-\lambda x} = r(x)$$

And solve for x(r).

$\rightarrow \quad x(r) = -1/\lambda \ln(1-r)$

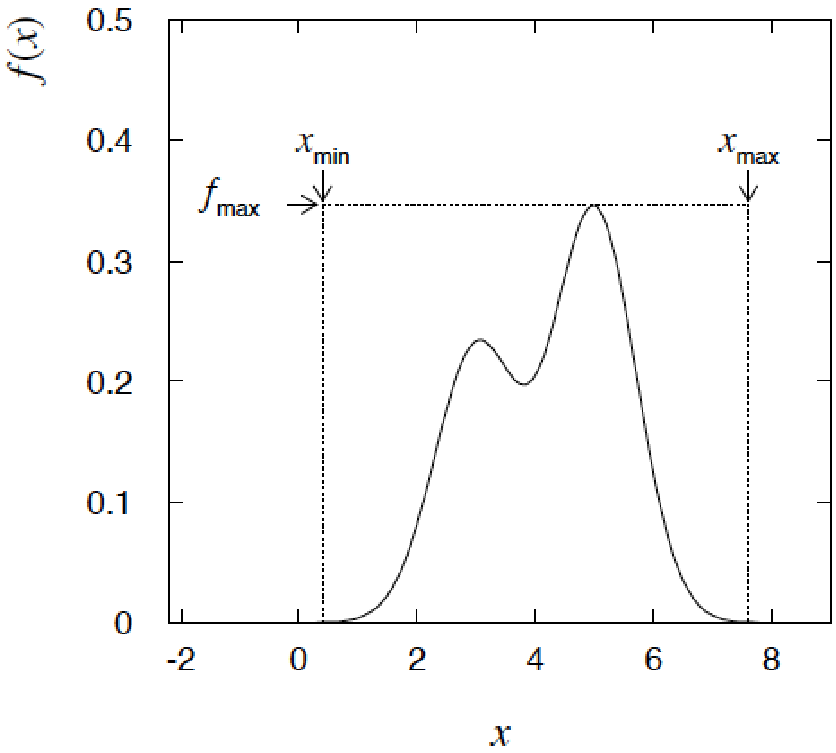


Hit and miss method (von Neumann)



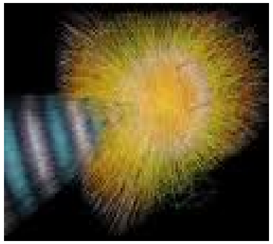
If $F^{-1}(u)$ is not computable, then use “hit and miss” method (also called acceptance-rejection method)

Enclose the pdf in a box:



- Generate a random number x, uniform in $[x_{min}, x_{max}]$, i.e.
  $x=_{xmin}+r_1(x_{max} - x_{min})$
  r_1 uniform in [0,1]
- Generate a second indipendent random number uniformly distributed between 0 and fmax, i.e. $u=r_2\, f_{max}$
- If $u < f(x)$, then accept x
  if not, reject x and repeat

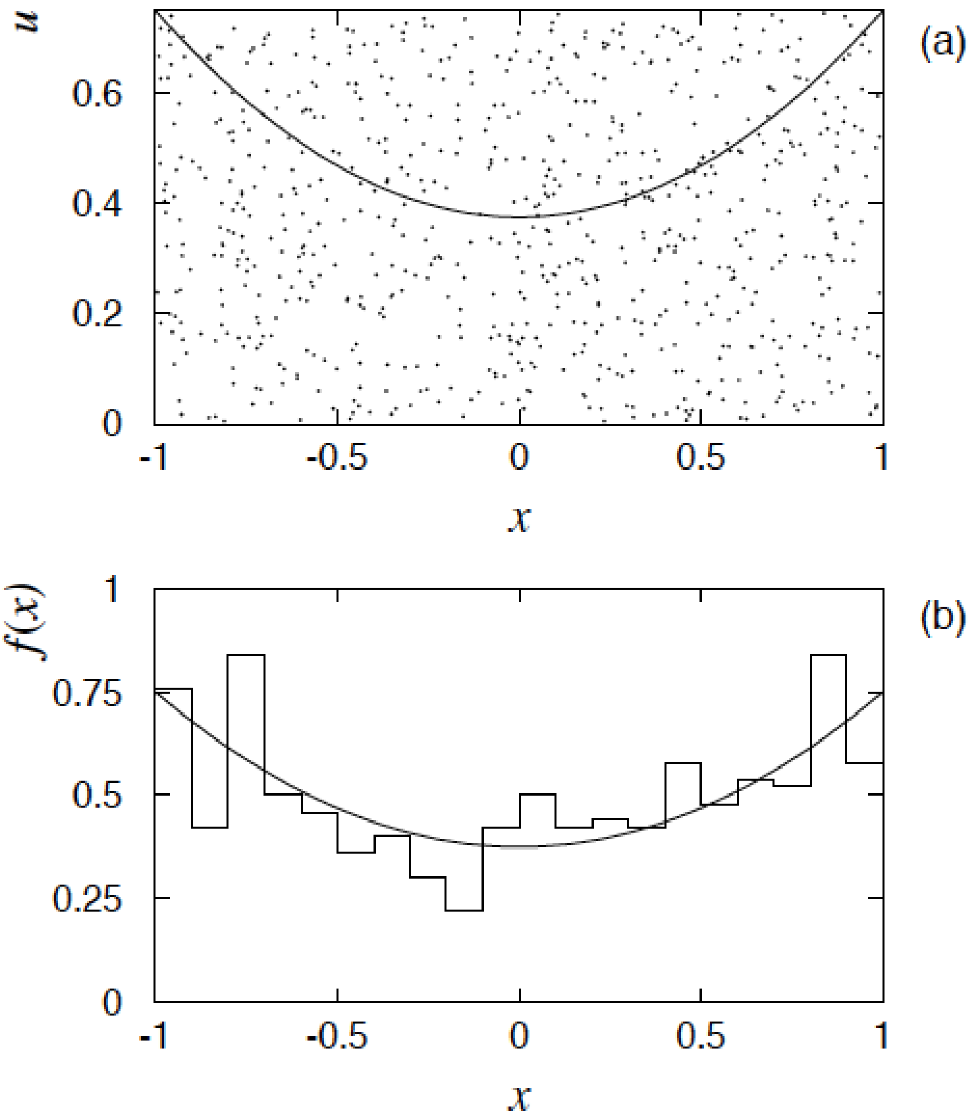
Example



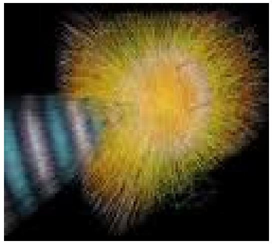
$$f(x) = \frac{3}{8}(1 + x^2)$$

$$(-1 \le x \le 1)$$

If dot below curve, use x value in histogram.



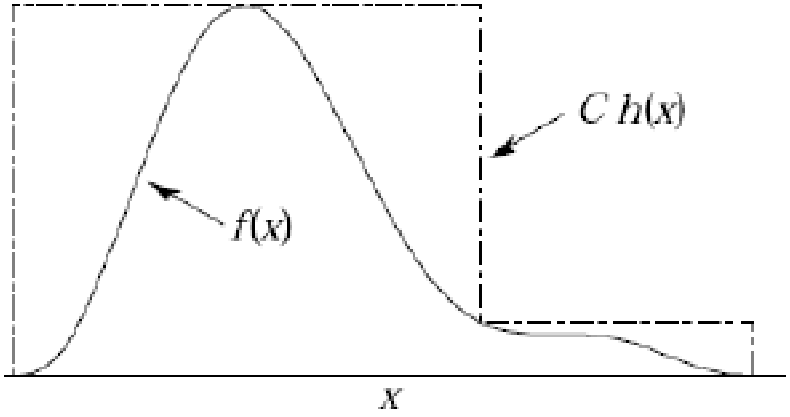
Improve efficiency of hit and miss



The fraction of accepted points is equal to the fraction of the box's area under the curve.

For very peaked distributions, this may be very low and thus the algorithm may be slow.

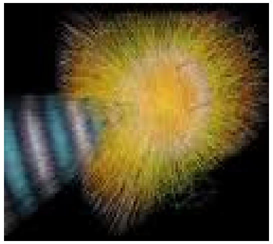
Improve by enclosing the pdf $f(x)$ in a curve $C\,h(x)$ that conforms to $f(x)$ more closely, where $h(x)$ is a pdf from which we can generate random values and C is a constant.



Generate points uniformly over $C\,h(x)$.

If point is below $f(x)$, accept x.

Monte Carlo integration

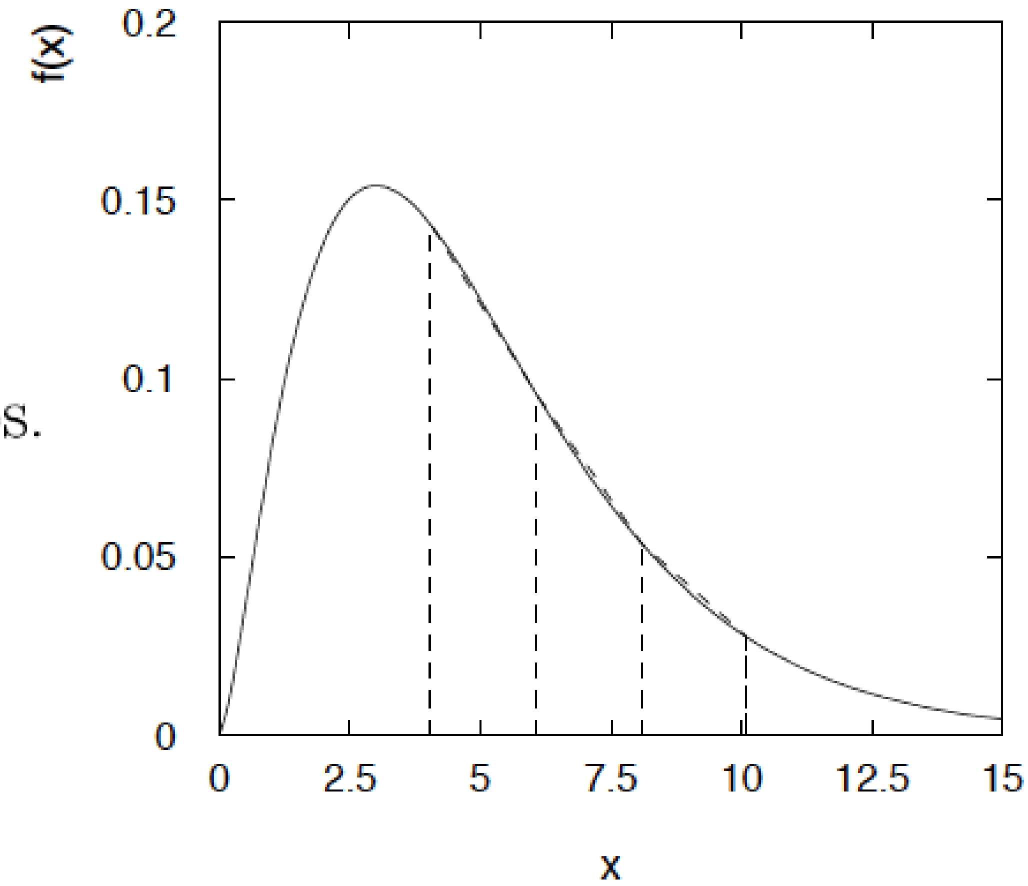


Check accuracy of the method:

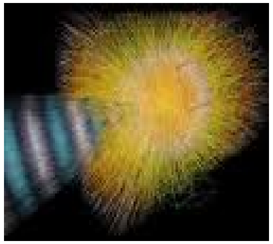
MC calculation = integration.

Compare to trapezoidal rule,

n = number of computing steps.



Accuracy of Monte Carlo integration



For 1-dimensional integral:

MC: $n \propto$ number of random values generated

accuracy $\propto 1/\sqrt{n}$

Trapezoid: $n \propto$ number of subdivisions

accuracy $\propto 1/n^2$

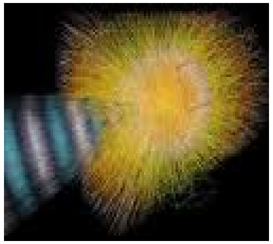
Trapezoid wins! But in d dimensions this becomes

MC: accuracy $\propto 1/\sqrt{n}$ $\leftarrow$ independent of d !

Trapezoid: accuracy $\propto 1/n^{2/d}$

For high enough d (d>4), MC always wins

Monte Carlo: statistical experiments



Often an analytical treatment of physical problems is either difficult or impossible. Therefore we do either an approximation or use a statistical description (via Monte Carlo)

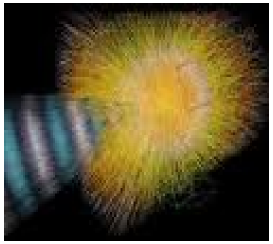
APPLICATIONS:

- High energy and nuclear physics
- Numerical calculations (integration, differentiation)
- Coding/encoding (e.g. secure connections, like ssh)
- Reliability tests
- Investment banking
- Earth sciences

METHODS:

- Find a statistic model
- Produce random numbers properly
- Calculate estimators from random quantities

Monte Carlo: statistical experiments



Qualitatively:

Scattering experiment:

Measure the angular distribution of particles scattered from protons in a fixed target

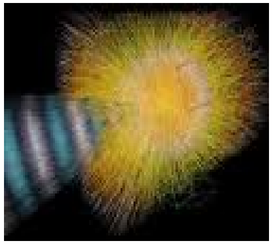
Ingredients:

- Magnitude and direction of the momentum vector of the incident particles
- Probability that a particle will collide with a proton in the target
- Resulting momentum vectors of the scattered particles

All are described in terms of probability distributions !

The final experimental result can be treated in terms of a multiple integration over all these probability distributions !!

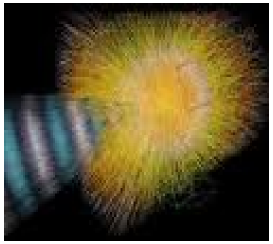
In “our” real life



Several stages:

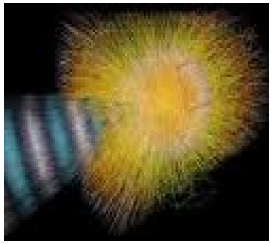
- **Event generator:**
  - Simulation of the PHYSICS process
  - Colliding particles
  - Cross section, processes involved, fragmentation ...
- **Detector simulation:**
  - Interaction of the produced particles with the material
  - Realistic description of the experimental apparatus
    - efficiencies
    - defects (dead channels, etc)
    - misalignment

Event generators



- $e^+e^- \rightarrow$ **hadrons**
  - JETSET (PYTHIA)
  - HERWIG
  - ARIADNE
- $e^+e^- \rightarrow$ **WW**
  - KORALW
  - EXCALIBUR
  - ERATO
- **pp** $\rightarrow$ **hadrons**
  - ISAJET
  - PYTHIA
  - HERWIG
- **PbPb** $\rightarrow$ **hadrons**
  - HIJING

Event generators



The output are so-called “events”, namely for each collision the programs give a list of final state particles, with their momentum vectors, types, angular distribution, etc

Event listing (summary)

I	particle/jet	KS	KF	orig	p_x	p_y	p_z	E	m
1	!p+!	21	2212	0	0,000	0,000	7000,000	7000,000	0,938
2	!p+!	21	2212	0	0,000	0,000	-7000,000	7000,000	0,938
3	!g!	21	21	1	0,853	-0,323	1739,862	1739,862	0,000
4	!ubar!	21	-2	2	-0,621	-0,163	-777,415	777,415	0,000
5	!g!	21	21	3	-2,427	5,486	1497,857	1487.	
6	!g!	21	21	4	-62,910	63,357	-463,274	471.	
7	!~g!	21	1000021	0	314,353	544,843	498,897	979.	
8	!~g!	21	1000021	0	-379,700	-476,000	525,685	980.	
9	!~chi_1-!	21	-1000024	7	130,058	112,247	129,860	263.	
10	!sbar!	21	-3	7	259,400	187,468	83,100	330.	
11	!c!	21	4	7	-79,403	242,409	283,026	381.	
12	!~chi_20!	21	1000023	8	-326,241	-80,971	113,712	385.	
13	!b!	21	5	8	-51,841	-294,077	339,853	491.	
14	!bbar!	21	-5	8	-0,597	-99,577	21,299	101.	
15	!~chi_10!	21	1000022	9	103,352	81,316	83,457	175.	
16	!s!	21	3	9	5,451	33,374	52,302	65.	
17	!cbar!	21	-4	9	20,833	-7,250	-5,938	22.	
18	!~chi_10!	21	1000022	12	-136,266	-72,961	53,246	181.	
19	!nu_mu!	21	14	12	-78,263	-24,757	21,719	84.	
20	!nu_mubar!	21	-14	12	-107,801	16,901	38,226	115.	
21	gamma	1	22	4	2,636	1,357	0,125	2.	
22	(~chi_1-)	11	-1000024	9	129,643	112,440	129,820	262.	
23	(~chi_20)	11	1000023	12	-322,330	-80,817	113,191	382.	
24	~chi_10	1	1000022	15	97,944	77,819	80,917	169.	
25	~chi_10	1	1000022	18	-136,266	-72,961	53,246	181.	
26	nu_mu	1	14	19	-78,263	-24,757	21,719	84.	
27	nu_mubar	1	-14	20	-107,801	16,901	38,226	115.	
28	(Delta++)	11	2224	2	0,222	0,012	-2734,287	2734.	

⋮

A simulated event

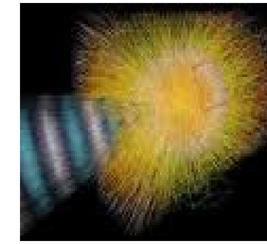
⋮

397	pi+	1	211	209	0,006	0,398	-308,296	308,297	0,140
398	gamma	1	22	211	0,407	0,087	-1695,458	1695,458	0,000
399	gamma	1	22	211	0,113	-0,029	-314,822	314,822	0,000
400	(pi0)	11	111	212	0,021	0,122	-103,709	103,709	0,135
401	(pi0)	11	111	212	0,084	-0,068	-94,276	94,276	0,135
402	(pi0)	11	111	212	0,267	-0,052	-144,673	144,674	0,135
403	gamma	1	22	215	-1,581	2,473	3,306	4,421	0,000
404	gamma	1	22	215	-1,494	2,143	3,051	4,016	0,000
405	pi-	1	-211	216	0,007	0,738	4,015	4,085	0,140
406	pi+	1	211	216	-0,024	0,293	0,486	0,585	0,140
407	K+	1	321	218	4,382	-1,412	-1,799	4,968	0,494
408	pi-	1	-211	218	1,183	-0,894	-0,176	1,500	0,140
409	(pi0)	11	111	218	0,955	-0,459	-0,590	1,221	0,135
410	(pi0)	11	111	218	2,349	-1,105	-1,181	2,855	0,135
411	(Kbar0)	11	-311	219	1,441	-0,247	-0,472	1,615	0,498
412	pi-	1	-211	219	2,232	-0,400	-0,249	2,285	0,140
413	K+	1	321	220	1,380	-0,652	-0,361	1,644	0,494
414	(pi0)	11	111	220	1,078	-0,265	0,175	1,132	0,135
415	(K_S0)	11	310	222	1,841	0,111	0,894	2,109	0,498
416	K+	1	321	223	0,307	0,107	0,252	0,642	0,494
417	pi-	1	-211	223	0,266	0,316	-0,201	0,480	0,140
418	nbar0	1	-2112	226	1,335	1,641	2,078	3,111	0,940
419	(pi0)	11	111	226	0,899	1,046	1,311	1,908	0,135
420	pi+	1	211	227	0,217	1,407	1,356	1,971	0,140
421	(pi0)	11	111	227	1,207	2,336	2,767	3,820	0,135
422	n0	1	2112	228	3,475	5,324	5,702	8,592	0,940
423	pi-	1	-211	228	1,856	2,606	2,808	4,259	0,140
424	gamma	1	22	229	-0,012	0,247	0,421	0,489	0,000
425	gamma	1	22	229	0,025	0,034	0,009	0,043	0,000
426	pi+	1	211	230	2,718	5,229	6,403	8,703	0,140
427	(pi0)	11	111	230	4,109	6,747	7,597	10,961	0,135
428	pi-	1	-211	231	0,551	1,233	1,945	2,372	0,140
429	(pi0)	11	111	231	0,645	1,141	0,922	1,608	0,135
430	gamma	1	22	232	-0,383	1,169	1,208	1,724	0,000
431	gamma	1	22	232	-0,201	0,070	0,060	0,221	0,000

⋮

PYTHIA Monte Carlo
pp → gluino-gluino

Detector simulation

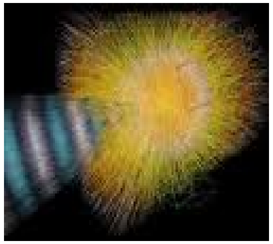


INPUT: particle list (with their species and momenta) from the event generator

Simulate the detector response taking into account:

- Multiple Coulomb scattering (generate scattering angle)
- Particle decays (generate lifetime), nuclear knock-out
- Ionization energy loss (generate Δ, Landau)
- Bremsstrahlung
- Electromagnetic / hadronic showers

Detector simulation



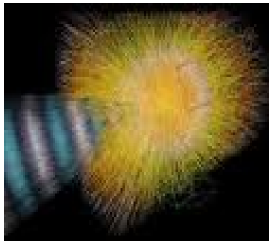
GEANT

- First version ~ 1974
- Till GEANT 3.21, in FORTRAN
- Since ~ 2000, FORTRAN version no longer developed, bug fixes

- Geant4: in C++, with a modern object-oriented design

- Next … Geant 5

Detector simulation

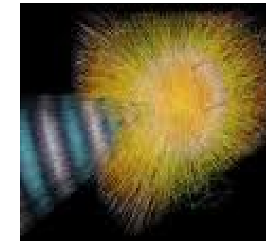


The detector simulation takes into account many more things:

- Production of the signals (electronics response)
- Addition of detector noise
- Description of the detection **efficiency** for each detector component (experiment specific)
- Position and energy **resolution** of each detector component (experiment specific)

The output is a list of digitized signals from all detector components, exactly like real data!! (or data format input for the reconstruction)

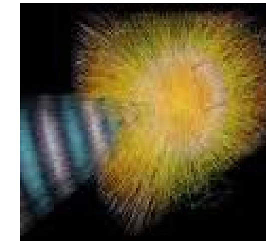
Use of simulations



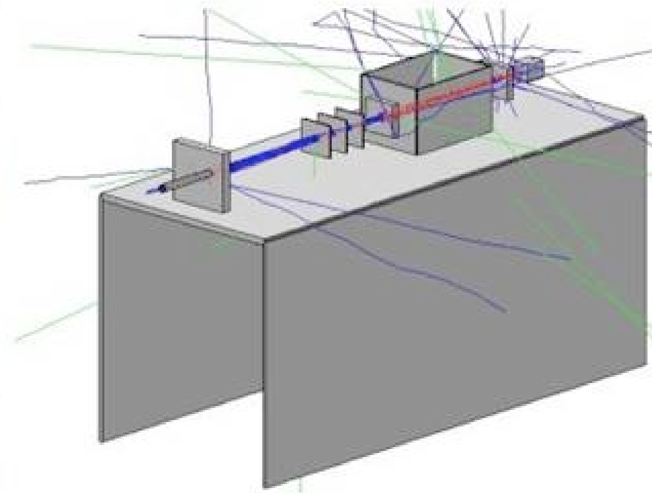
- Develop reconstruction algorithms
  (particle trajectories in the tracking detectors, showers in the calorimeters)
- Optimize trigger selection
- Identify the best signal signature
- Compute efficiency of selections in real data analysis
- During design of an experiment: define the detector acceptance, etc

Simulation is absolutely crucial in the planning phase of experiments, for preparation of data taking, to optmize analyses, to evaluate the significance of the results

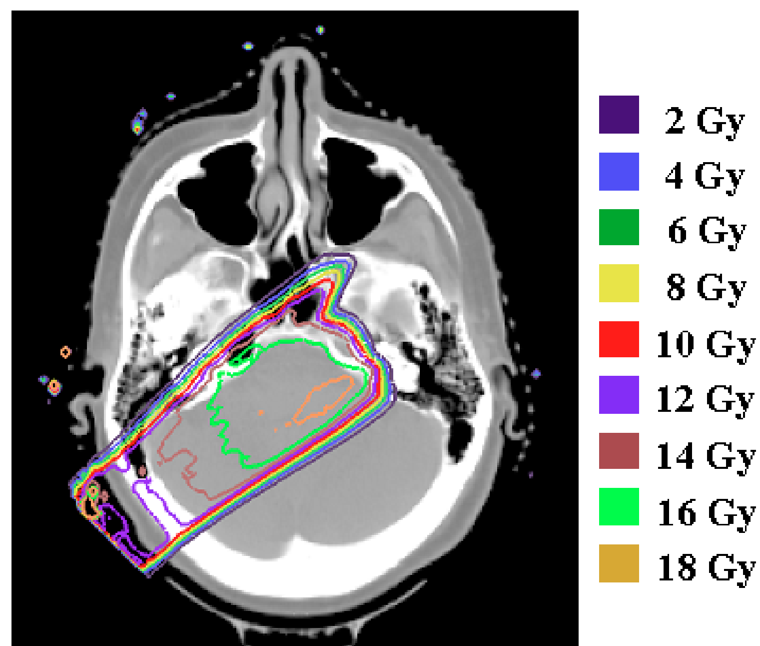
Geant – other applications



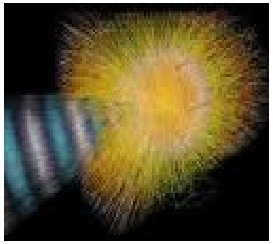
Proton beam eye therapy unit at the Laboratori Nazionali del Sud (INFN) in Catania (left) and a display from the Geant4 advanced example for the simulation of the same beam line (right).

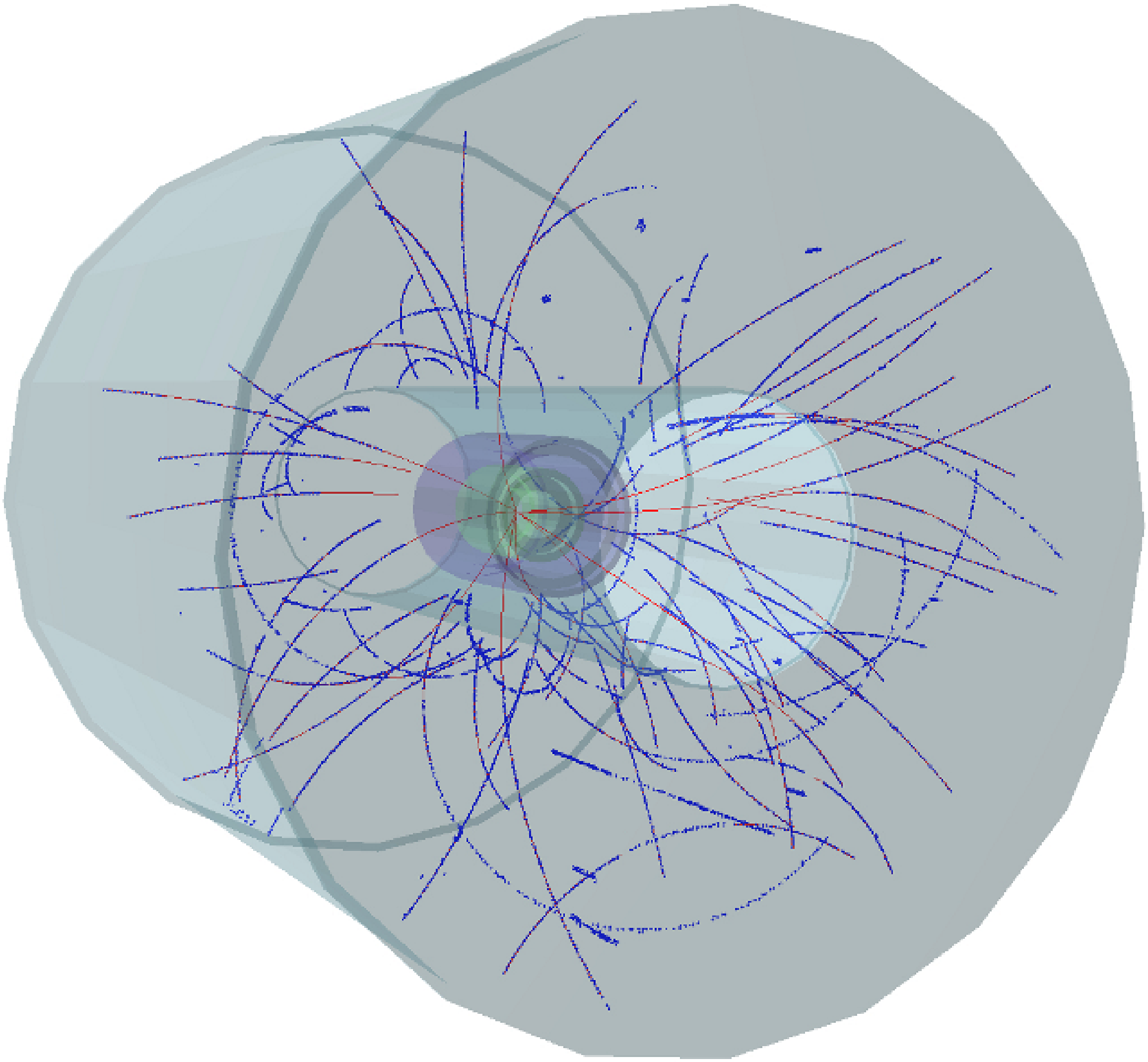


Geant4 simulated dose contours in a human brain with a proton beam.

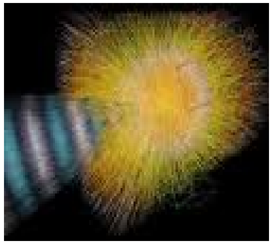


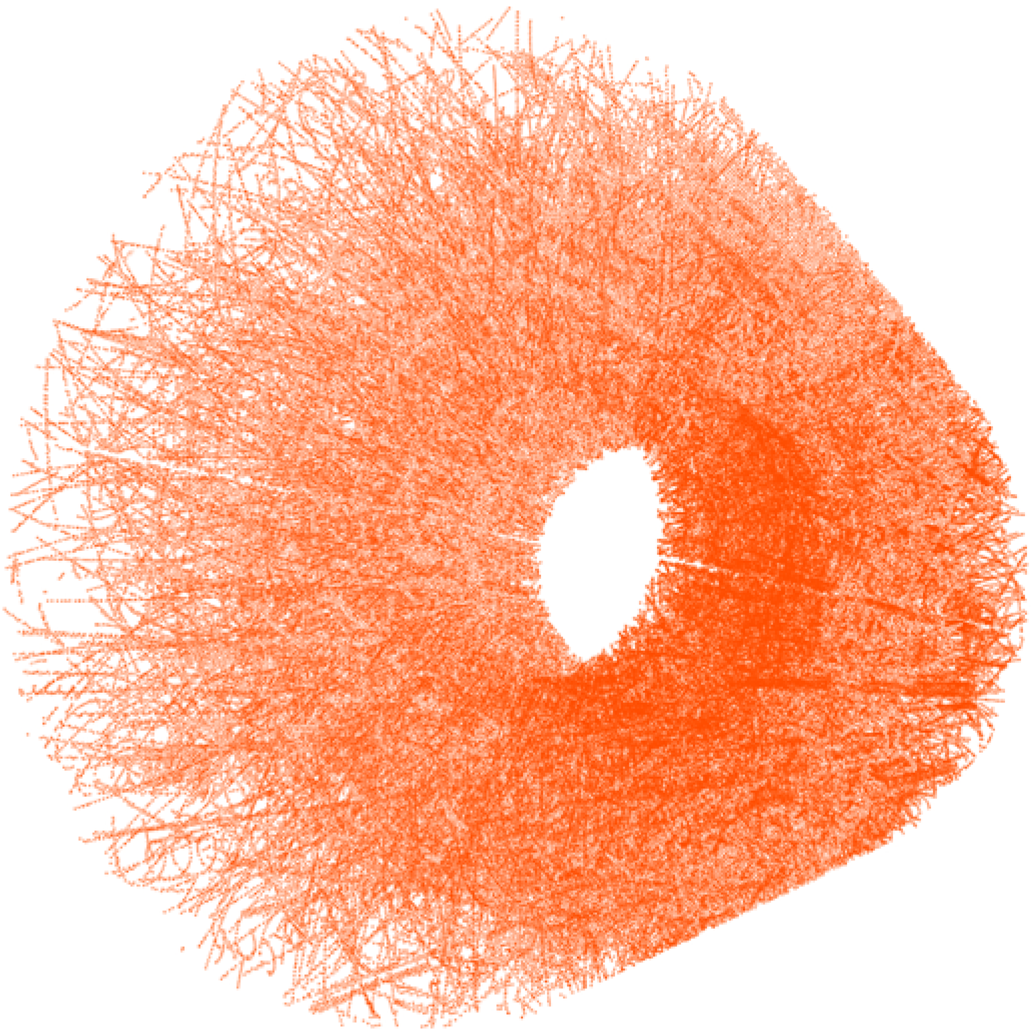
ALICE (TPC) simulation: proton-proton



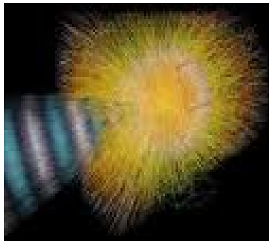


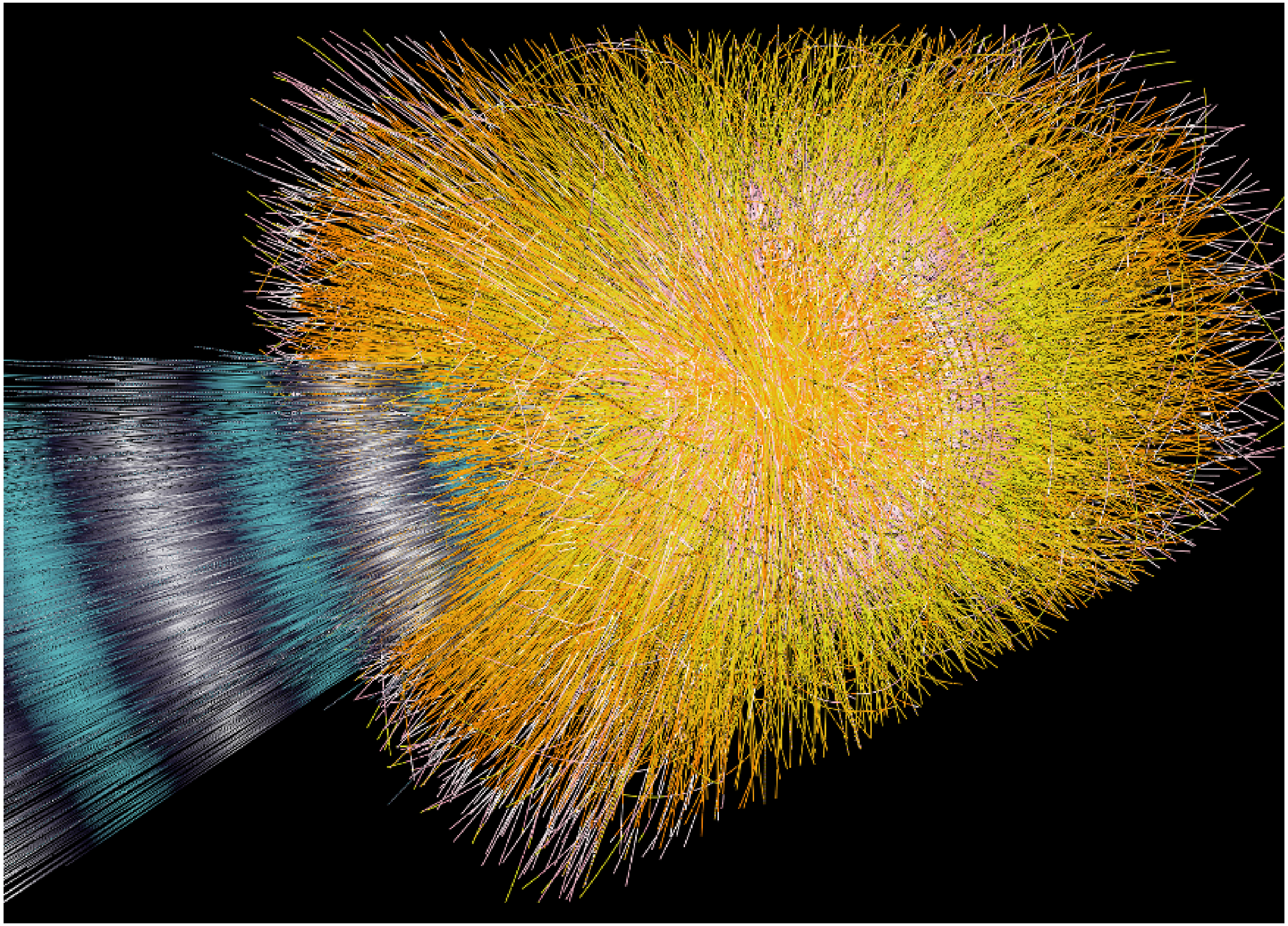
ALICE (TPC) simulation: Pb-Pb





ALICE simulation: Pb-Pb

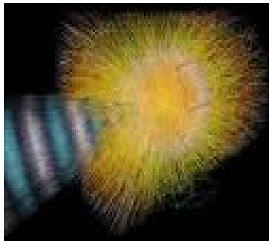


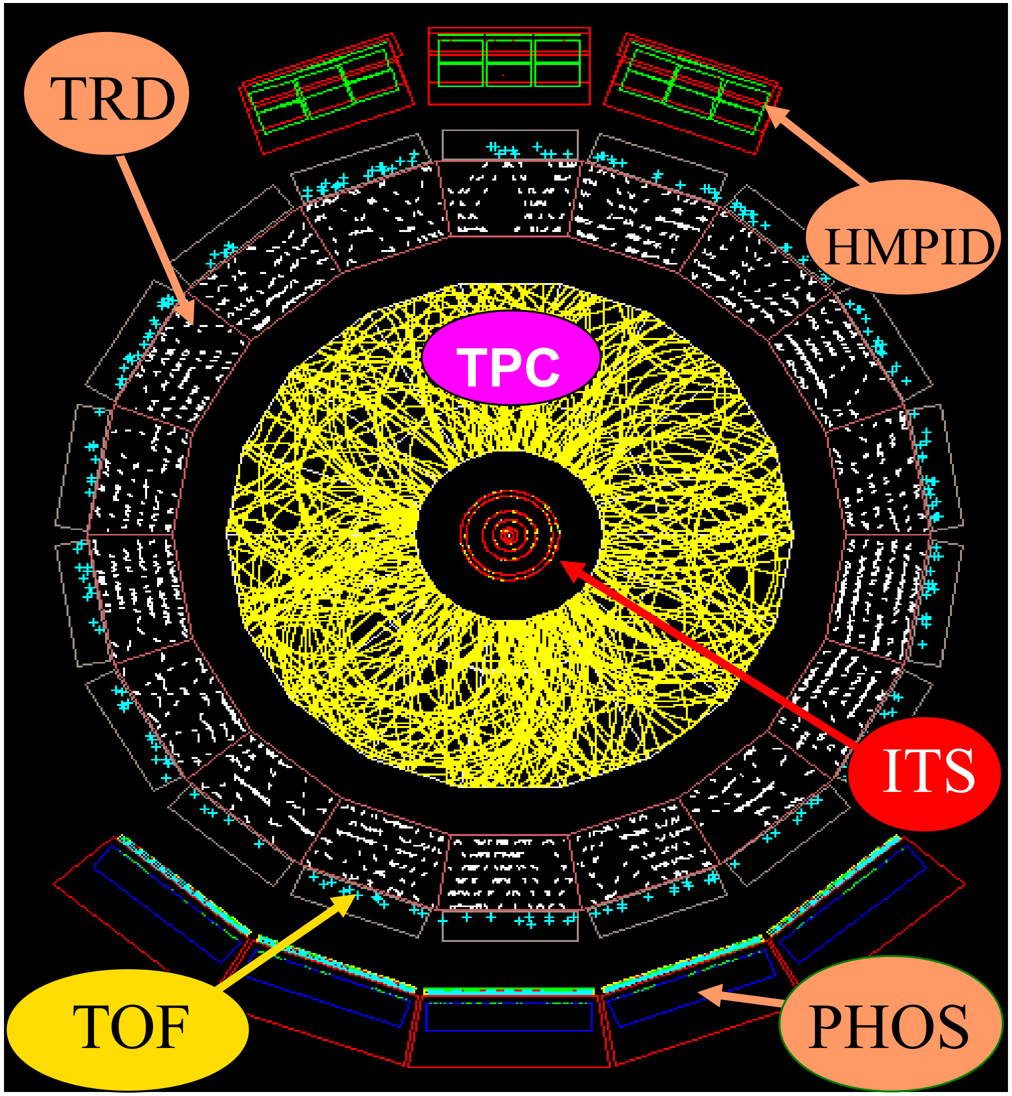


dN/dy ~ 8000

Particles per unit rapidity

ALICE simulation: Pb-Pb





2 degree slice ONLY!!

(~ 500 tracks)

(a bit old ...)