Exercise 1 - Introduction

- Schedule:
  - Tuesday - Exercise 1:
    - Random numbers
    - MC method
    - MC integration
  - Wednesday - Exercise 2:
    - Sudakov form factor
    - MC solution of evolution equation
  - Thursday - Exercise 3
    - Calculation & simulation of Higgs production
    - Using MC solution of evolution equation → calculation of pt spectrum of Higgs at LHC

Congruential linear generator

- develop our own simple generator

$$I_j = mod(aI_{j-1} + c, m)$$
$$R_j = \frac{I_j}{m}$$

- with $mod(i_1, i_2) = i_1 - INT(i_1/i_2)i_2$

seed I_0, multiplicative constant a and additive constant c modulus m

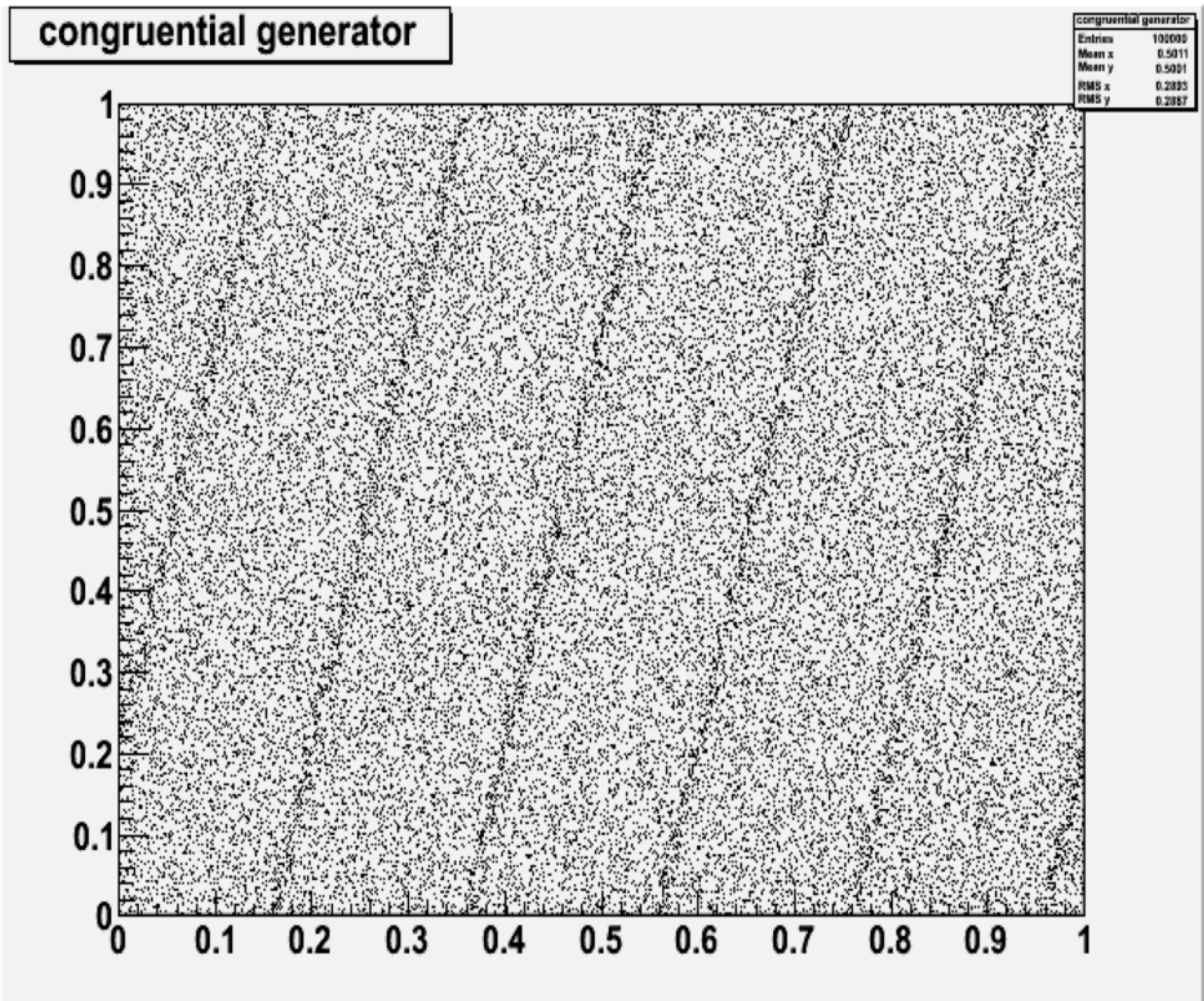
➔ maximal repetition period: $\mathcal{O}(m)$

➔ example:

$$I_0 = 4711$$
$$a = 205$$
$$c = 29573$$
$$m = 139968$$

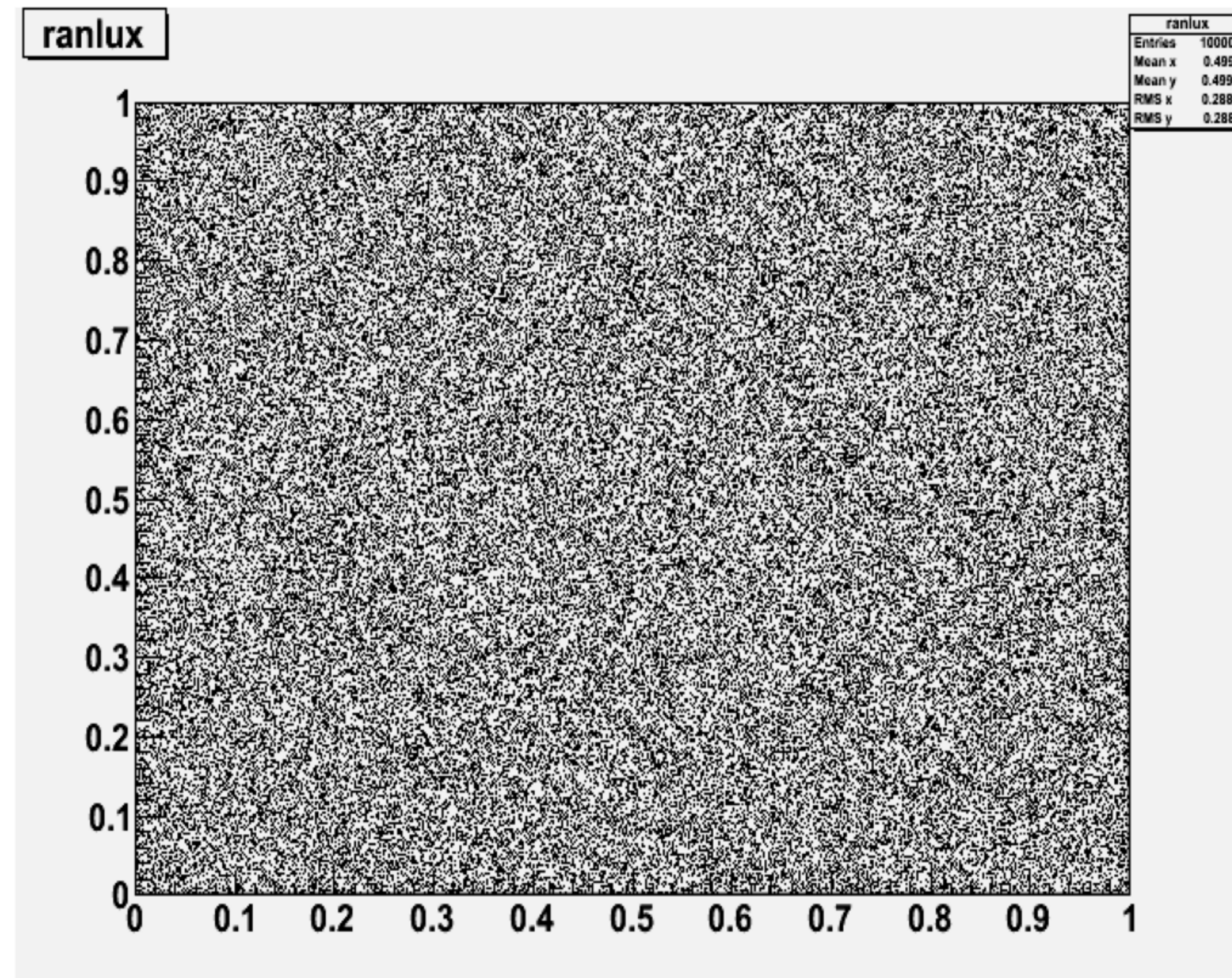
Randomness tests

- Congruential generator



- RANLUX

M. Lüscher, A portable high-quality random number generator for lattice field theory simulations, Computer Physics Communications 79 (1994) 100
http://luscher.web.cern.ch/luscher/ranlux/index.html



➔ RANLUX much more sophisticated. Developed and used for QCD lattice calcs

Generating distributions

- From uniform distributions to distributions for any probability density function
  - use variable transformation
  - linear p.d.f:

$$
\begin{aligned}
f(x) &= 2x \\
u(x) &= \int_0^x 2t\,dt = x^2 \\
x_j &= \sqrt{u_j}
\end{aligned}
$$

  - 1/x distribution

$$
\begin{aligned}
f(x) &= \frac{1}{x} \\
u(x) &= \frac{\int_{x_{min}}^{x} \frac{1}{t} dt}{F_{max} - F_{min}} \\
x_j &= x_{min} \left(\frac{x_{max}}{x_{min}} \right)^{u_j}
\end{aligned}
$$

Generating distributions

- Brute Force or Hit & Miss method
  - use this if there is no easy way to find a analytic integrable function
  - find $c \leq \max f(x)$
  - reject if $f(x_i) \leq u_j \cdot c$
  - accept if $f(x_i) \geq u_j \cdot c$

- Improvements for Hit & Miss method by variable transformation
  - find $c \cdot g(x) \geq f(x)$
  - reject if $f(x) \leq u_j \cdot c \cdot g(x)$
  - accept if $f(x) \geq u_j \cdot c \cdot g(x)$

Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

Law of large numbers

- **Law of large numbers**

x_i independent random variables, having the same mean and variances σ_i^2 :

$$\frac{1}{N}\sum_{i=1}^{N} x_i \to \mu$$

for large enough N the sum converges to the correct answer.

- **Convergence**

is given with a certain probability …

THIS is a mathematically serious and precise statement !!!!

Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

Central Limit Theorem

- Central Limit Theorem

  for large N the sum of independent random variables is always normally (Gaussian) distributed:

$$f(x) = \frac{1}{s\sqrt{2\pi}} \exp\left[-\frac{(x-a)^2}{2s^2}\right]$$

$$\frac{\sum_i x_i - \sum_i \mu_i}{\sqrt{\sum_i \sigma_i^2}} \rightarrow \mathcal{N}(0,1)$$

➔ independent on the original sub-distributions

Central Limit Theorem

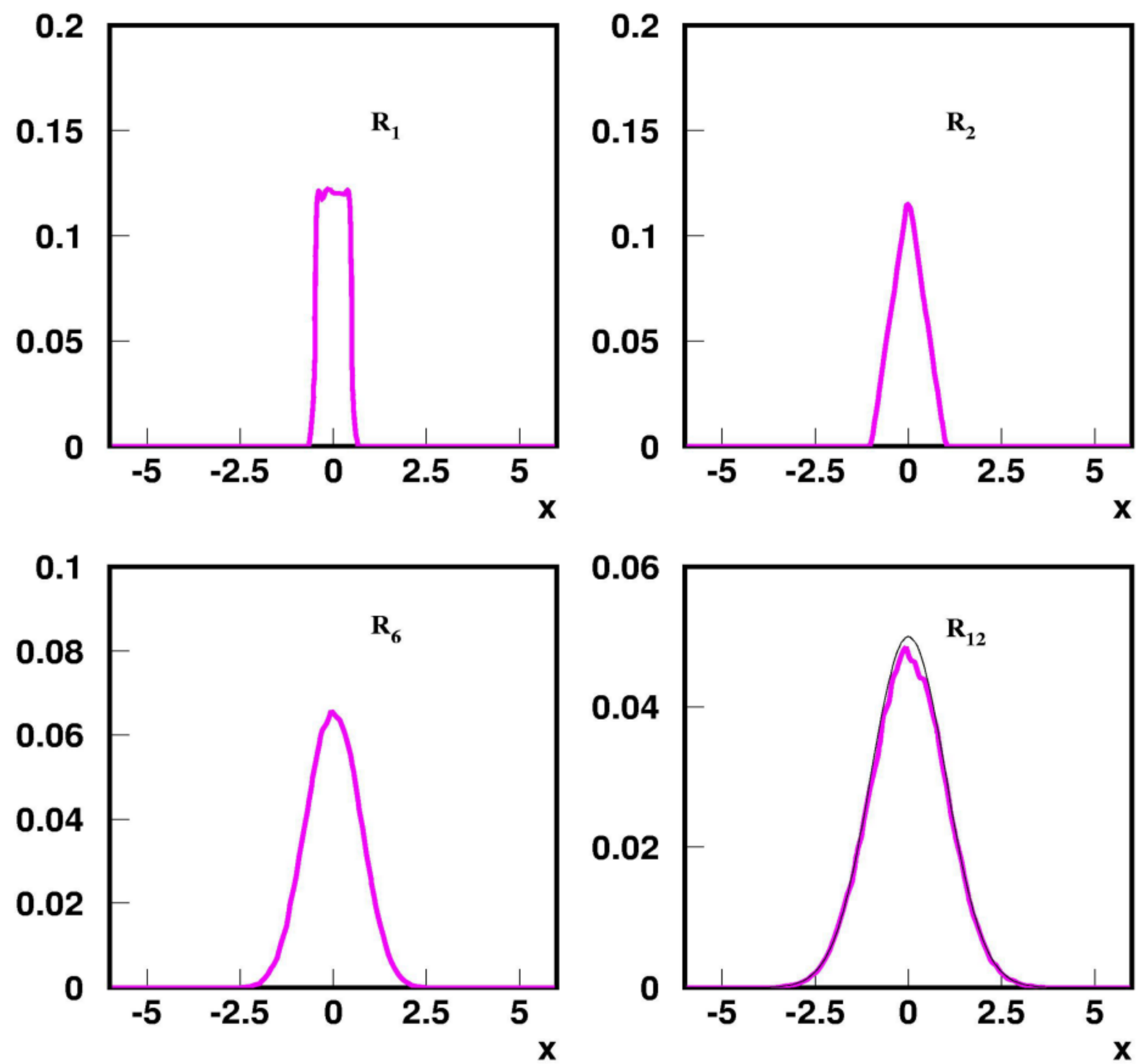
- Central Limit Theorem
  for large N the sum of independent random variables is always normally (Gaussian) distributed:

$$f(x) = \frac{1}{s\sqrt{2\pi}} \exp\left[-\frac{(x-a)^2}{2s^2}\right]$$

- example: take sum of uniformly distributed random numbers:

$$\begin{aligned} R_n &= \sum_{i=1}^{n} R_i \\ E[R_1] &= \int u du = 1/2, \\ V[R_1] &= \int (u-1/2)^2 du = 1/12 \\ E[R_n] &= n/2 \\ V[R_n] &= n/12 \end{aligned}$$

- for Gaussian with mean=0 and variance=1, take for n=12:



Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

Central Limit Theorem

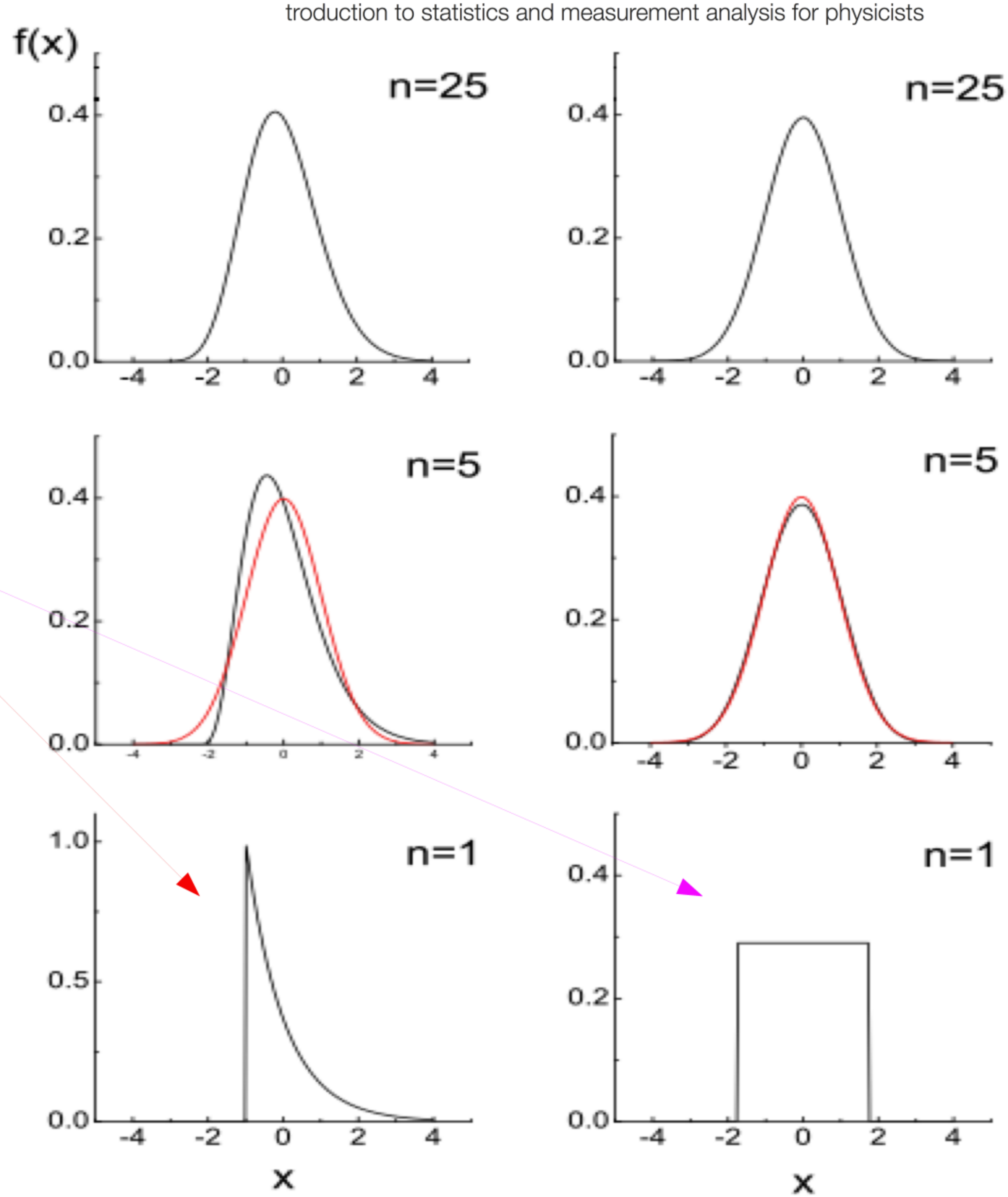
- Central Limit Theorem
  for large N the sum of independent random variables is always normally (Gaussian) distributed

➔ for any starting distribution

➔ for uniform distribution

➔ for exponential distribution

G. Bohm, G. Zechln
troduction to statistics and measurement analysis for physicists



Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

Monte Carlo Integration

- **Law of large numbers**

$$\frac{1}{N}\sum_{i=1}^{N} f(u_i) \rightarrow \frac{1}{b-a}\int_a^b f(u)du$$

MC estimate converges to true integral

- **Central limit theorem**

MC estimate is asymptotically normally distributed

it approaches a Gaussian density

$$\sigma = \frac{\sqrt{V[f]}}{\sqrt{N}} \sim \frac{1}{\sqrt{N}}$$

with effective variance V(f)

➔ to decrease σ, either reduce V(f) or increase N

- advantages for n-dimensional integral ...

i.e. phase space integrals 2 → n processes

is where other approaches tend to fail

Monte Carlo Integration

- solve

$$I = \int_a^b f(x)dx = (b-a)E[f(x)]$$

- estimate by

$$I \sim I_{MC} = \frac{b-a}{n}\sum_{i=1}^{n} f(x_i)$$

- with variance

$$\begin{aligned} V[I_{MC}] &= \sigma_I^2 = V\left[\frac{b-a}{n}\sum_i f(x_i)\right] \\ &= \frac{(b-a)^2}{n}V[f] \\ &= \frac{(b-a)^2}{n}\left[\frac{\sum_i f(x_i)^2}{n} - \left(\frac{\sum_i f(x_i)}{n}\right)^2\right] \\ &= \frac{1}{n}\left[(b-a)^2\frac{\sum_i f(x_i)^2}{n} - I_{MC}^2\right] \end{aligned}$$

MC method: hit & miss

- Integral in hit & miss

$$I = \int_a^b f(x)dx = I_0 \frac{N}{N_0}$$

- Variance $V[r] = (\delta(N))^2 = \sigma^2$ using binomial statistics with

$E[r] = N_0 P$ and $V[r] = N_0 P(1-P)$ with $P = N/N_0$

giving $V[r] = N(1-P)$

- uncertainties in hit & miss method:

$$\frac{\delta I}{I} = \frac{I_0 \sigma / N_0}{I_0 N / N_0} = \sqrt{\frac{N(1-P)}{N^2}}$$

MC method: do even better ...

- Importance sampling

MC for function f(x)
approximate f(x) ~ g(x)
with g(x) > f(x) simple and integrable
generate x according to g(x):

$$\int_{x_{min}}^{x} g(x')dx' = R1 \int_{x_{min}}^{x_{max}} g(x')dx'$$

example:

$$f(x) = 1/x^{0.7}$$

$$g(x) = 1/x$$

$$x = x_{min} \cdot \left(\frac{x_{max}}{x_{min}} \right)^{R1}$$

reject event if: f(x) < g(x) R2

MC method: do even better ...

- Importance sampling

MC for function f(x)
approximate f(x) ~ g(x)
with g(x) > f(x) simple and integrable
generate x according to g(x):

$$\int_{x_{min}}^{x} g(x')dx' = R1 \int_{x_{min}}^{x_{max}} g(x')dx'$$

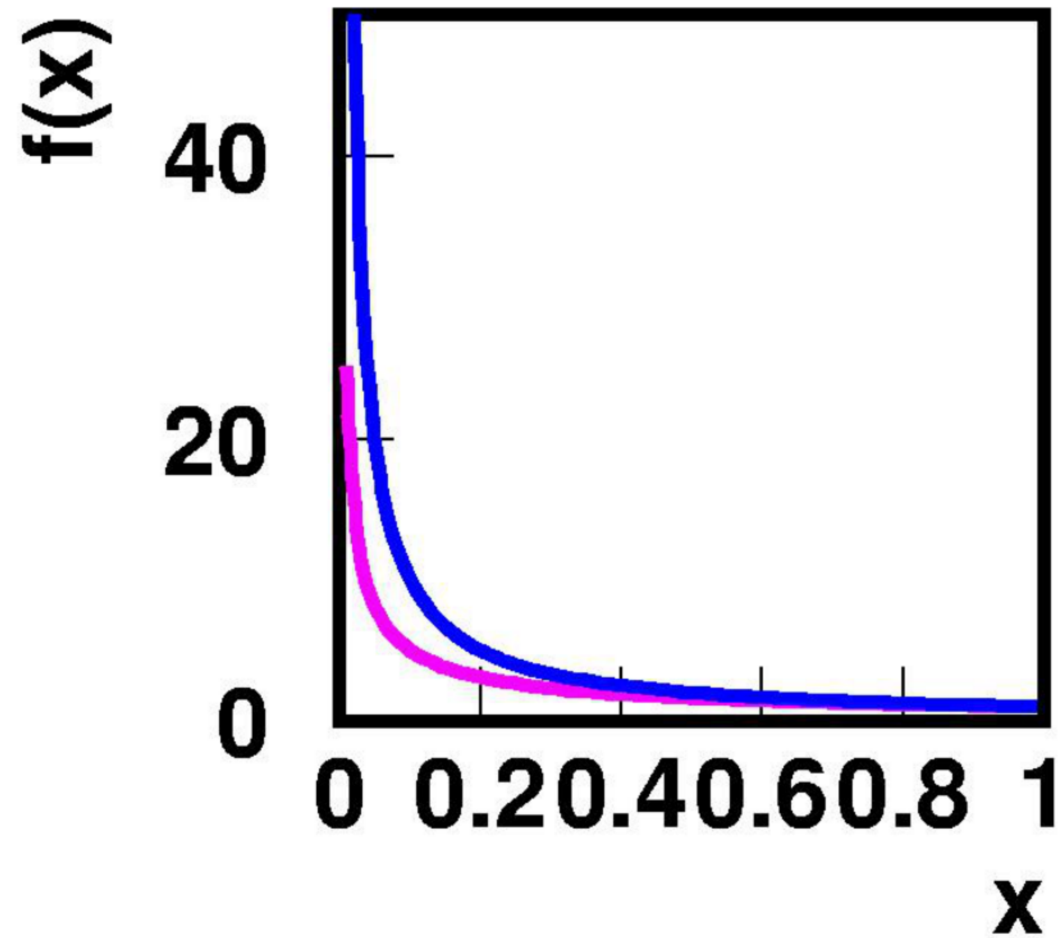
example:

$$f(x) = 1/x^{0.7}$$

$$g(x) = 1/x$$

$$x = x_{min} \cdot \left(\frac{x_{max}}{x_{min}}\right)^{R1}$$

reject event if: f(x) < g(x) R2



Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

MC method: do even better ...

- Importance sampling

MC for function f(x)
approximate f(x) ~ g(x)
with g(x) > f(x) simple and integrable
generate x according to g(x):

$$\int_{x_{min}}^{x} g(x')dx' = R1 \int_{x_{min}}^{x_{max}} g(x')dx'$$

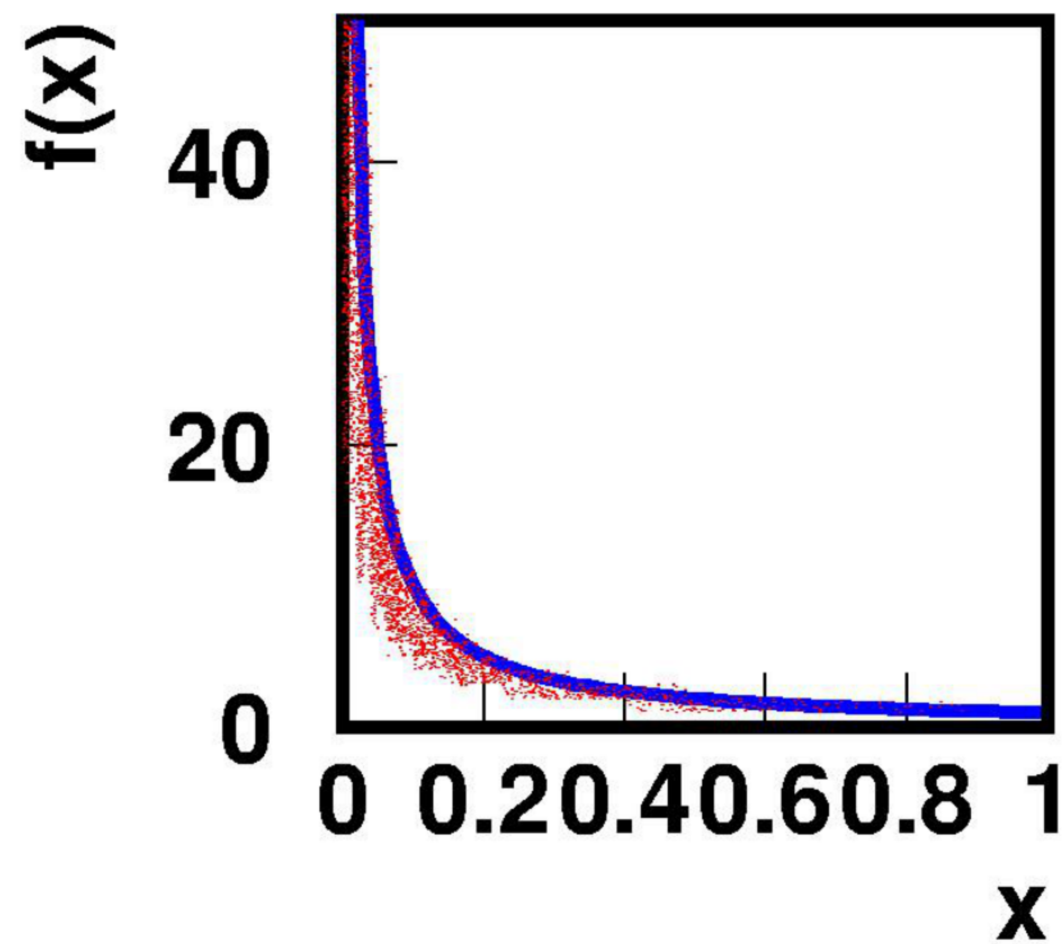
example:

$$f(x) = 1/x^{0.7}$$

$$g(x) = 1/x$$

$$x = x_{min} \cdot \left(\frac{x_{max}}{x_{min}}\right)^{R1}$$

reject event if: f(x) < g(x) R2



Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

MC method: do even better ...

- Importance sampling

MC for function f(x)
approximate f(x) ~ g(x)
with g(x) > f(x) simple and integrable
generate x according to g(x):

$$\int_{x_{min}}^{x} g(x')dx' = R1 \int_{x_{min}}^{x_{max}} g(x')dx'$$

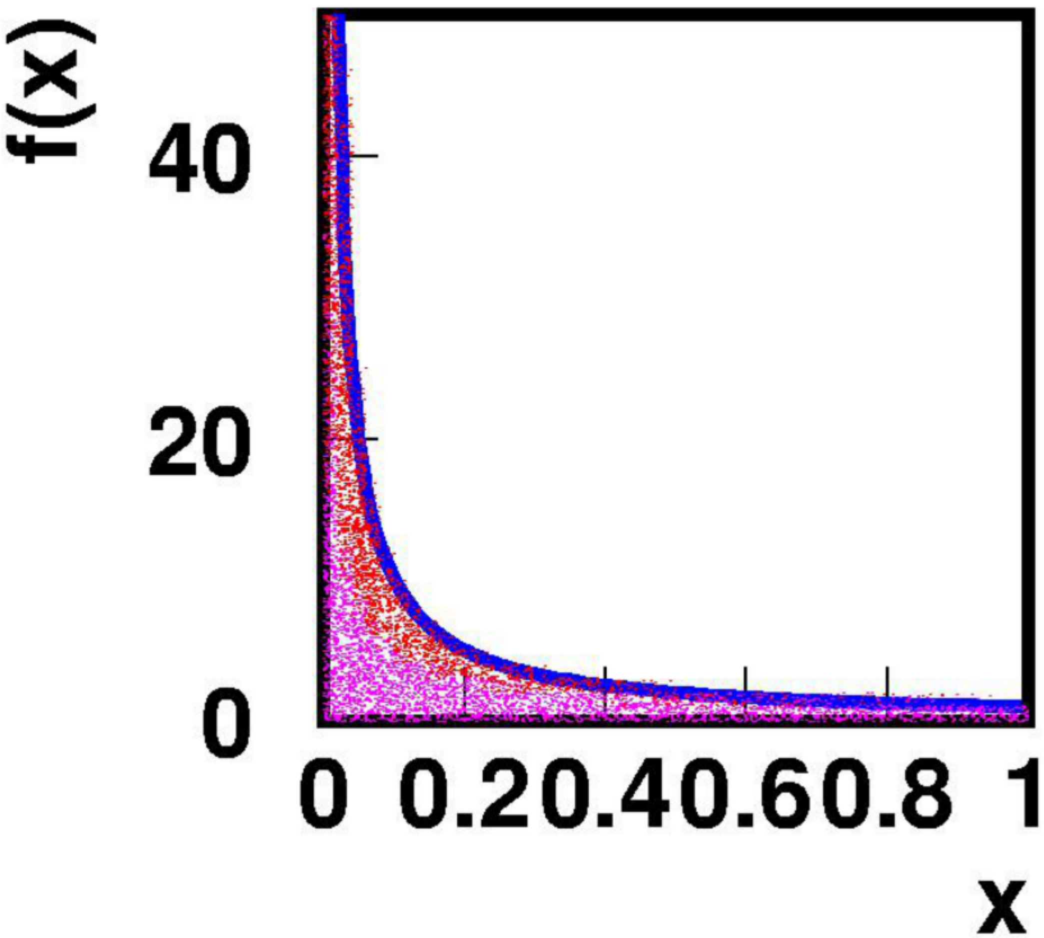
example:

$$f(x) = 1/x^{0.7}$$

$$g(x) = 1/x$$

$$x = x_{min} \cdot \left(\frac{x_{max}}{x_{min}} \right)^{R1}$$

reject event if: f(x) < g(x) R2



A good reference

- PYTHIA 6.4 Physics and Manual

  T. Sjostrand, S. Mrenna and P. Skands

  JHEP 05 (2006) 026 hep-ph/0603175)

4 Monte Carlo Techniques

Quantum mechanics introduces a concept of randomness in the behaviour of physical processes. The virtue of event generators is that this randomness can be simulated by the use of Monte Carlo techniques. In the process, the program authors have to use some ingenuity to find the most efficient way to simulate an assumed probability distribution. A detailed description of possible techniques would carry us too far, but in this section some of the most frequently used approaches are presented, since they will appear in discussions in subsequent sections. Further examples may be found e.g. in [Jam80].

First of all one assumes the existence of a random number generator. This is a (Fortran) function which, each time it is called, returns a number R in the range between 0 and 1, such that the inclusive distribution of numbers R is flat in the range, and such that different numbers R are uncorrelated. The random number generator that comes with PYTHIA is described at the end of this section, and we defer the discussion until then.

4.1 Selection From a Distribution

The situation that is probably most common is that we know a function $f(x)$ which is non-negative in the allowed x range $x_{\mathrm{min}} \leq x \leq x_{\mathrm{max}}$. We want to select an x 'at random' so that the probability in a small interval $\mathrm{d}x$ around a given x is proportional to $f(x)\,\mathrm{d}x$. Here $f(x)$ might be a fragmentation function, a differential cross section, or any of a number of distributions.

One does not have to assume that the integral of $f(x)$ is explicitly normalized to unity: by the Monte Carlo procedure of picking exactly one accepted x value, normalization is implicit in the final result. Sometimes the integral of $f(x)$ does carry a physics content of its own, as part of an overall weight factor we want to keep track of. Consider, for instance, the case when x represents one or several phase-space variables and $f(x)$ a differential cross section; here the integral has a meaning of total cross section for the process studied. The task of a Monte Carlo is then, on the one hand, to generate events one at a time, and, on the other hand, to estimate the total cross section. The discussion of this important example is deferred to section 7.4.

If it is possible to find a primitive function $F(x)$ which has a known inverse $F^{-1}(x)$, an x can be found as follows (method 1):

$$\int_{x_{\mathrm{min}}}^{x} f(x)\,\mathrm{d}x = R \int_{x_{\mathrm{min}}}^{x_{\mathrm{max}}} f(x)\,\mathrm{d}x$$
$$\Longrightarrow \quad x = F^{-1}(F(x_{\mathrm{min}}) + R(F(x_{\mathrm{max}}) - F(x_{\mathrm{min}}))) \,. \qquad (2)$$

The statement of the first line is that a fraction R of the total area under $f(x)$ should be to the left of x. However, seldom are functions of interest so nice that the method above works. It is therefore necessary to use more complicated schemes.

Special tricks can sometimes be found. Consider e.g. the generation of a Gaussian $f(x) = \exp(-x^2)$. This function is not integrable, but if we combine it with the same Gaussian distribution of a second variable y, it is possible to transform to polar coordinates

$$f(x)\,\mathrm{d}x\,f(y)\,\mathrm{d}y = \exp(-x^2 - y^2)\,\mathrm{d}x\,\mathrm{d}y = r\exp(-r^2)\,\mathrm{d}r\,\mathrm{d}\varphi \,, \qquad (3)$$

and now the r and φ distributions may be easily generated and recombined to yield x. At the same time we get a second number y, which can also be used. For the generation of transverse momenta in fragmentation, this is very convenient, since in fact we want to assign two transverse degrees of freedom.

If the maximum of $f(x)$ is known, $f(x) \leq f_{\mathrm{max}}$ in the x range considered, a hit-or-miss method will always yield the correct answer (method 2):

43

Literature for MC Method

- G. Bohm, G. Zech (2010) Online Book
  Introduction to Statistics and Data Analysis for Physicists
- S. Weinzierl Introduction to Monte Carlo method hep-ph/0006269
- G. Cowan. Statistical data analysis, Oxford, UK: Clarendon (1998)
- J. Vermaseren, Lectures on Monte Carlo, Madrid 2008 (http://www.nikhef.nl/~form/maindir/courses/course2/course2.html/)
- History of Monte Carlo Method
  (http://www.geocities.com/CollegePark/Quad/2435/history.html)

Exercise 1

Monte Carlo technique

1. construct a uniform random number generator from the congruential method:
$$I_{i+1} = mod(a \cdot I_i + c, m)$$
$$R_{i+1} = \frac{I_{i+1}}{m}$$
with $I_0 = 4711$, $a = 205$, $c = 29573$ and $m = 139968$
Compare the correlation of 2 random numbers. Compare this with RANLUX.

2. construct a Gaussian random number generator from a uniform random number generator

3. write a small program that integrates (with Monte Carlo method) the function $f(x) = 3x^2$ for $\int_0^1 f(x)dx$, and calculate the uncertainty.

4. write a small program that integrates (with Monte Carlo method) $\int_0^1 \int_0^x dxdy$ with $0 < x, y < 1$.

5. write a small program to integrate a simple function in one dimension:
$\int_{x_{min}}^1 g(x)dx = \int_{x_{min}}^1 (1-x)^5 \frac{dx}{x}$, using Monte Carlo integration, with $x_{min} = 0.0001$
Improve the above integration by using importance sampling.

Exercise 1- if you have time ...

If you have time, you can do some more exercises:

- use the LHAPDF library to calculate the flavor sum rules:

$$\int_0^1 dx\ u_V(x,Q^2) = 2$$

$$\int_0^1 dx\ d_V(x,Q^2) = 1$$

use the LHAPDF library and calculate the momentum sum rule:

$$\int_0^1 dx \sum_{i=-6}^{6} x p_i(x,Q^2)$$

use the MRST(MRST2004nlo) set and the LO* (MRST2007lomod) set. How much is the momentum sum rule violated in the LO* set ? Is the momentum sum rule satisfied (or violated in the same way) for different Q^2 values (use $Q^2 = 5, 10, 100, 1000$ GeV^2).

Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

How to get started

- utilities:

```
courselib.h: include headers
ranlxd.h, ranlxd.cc: random number generator ranlux
```

- initialize ROOT (needed for plotting)

```
module avail
module load root/5.34
```

- copy all the templates (be careful, do not to overwrite …)

  cp -rp /afs/desy.de/user/s/school30/public/Exercises .

- compiling and running:

```
cd exercise-1
make -f makefile-example-1
./example-1
```

- templates are provided which include the general structure – you only have to fill the interesting – important parts … good luck

Monte Carlo School 2015
13-17 April 2015
DESY, Hamburg

Computing setup

- Connect either to eduroam or to the school network:

  Name: terascale

  WPA/WPA2-PSK: XxPWjNH7

- All will get school accounts for naf:
  - for example: ssh -X school30@naf-school01.desy.de
  - create folder:

    `cd public`
  - copy all templates:

    cp -rp /afs/desy.de/user/s/school30/public/Exercises .

- Writeup, Exercise sheets, templates and solutions at:

  `http://www.desy.de/~jung/mcschool2015/`