Semplici esempi di ROOT&LaTex

Laboratorio di Interazioni Fondamentali

Dipartimento di Fisica – Università di Pisa

Anno Accademico 2016/2017

Michael J. Morello
Scuola Normale Superiore

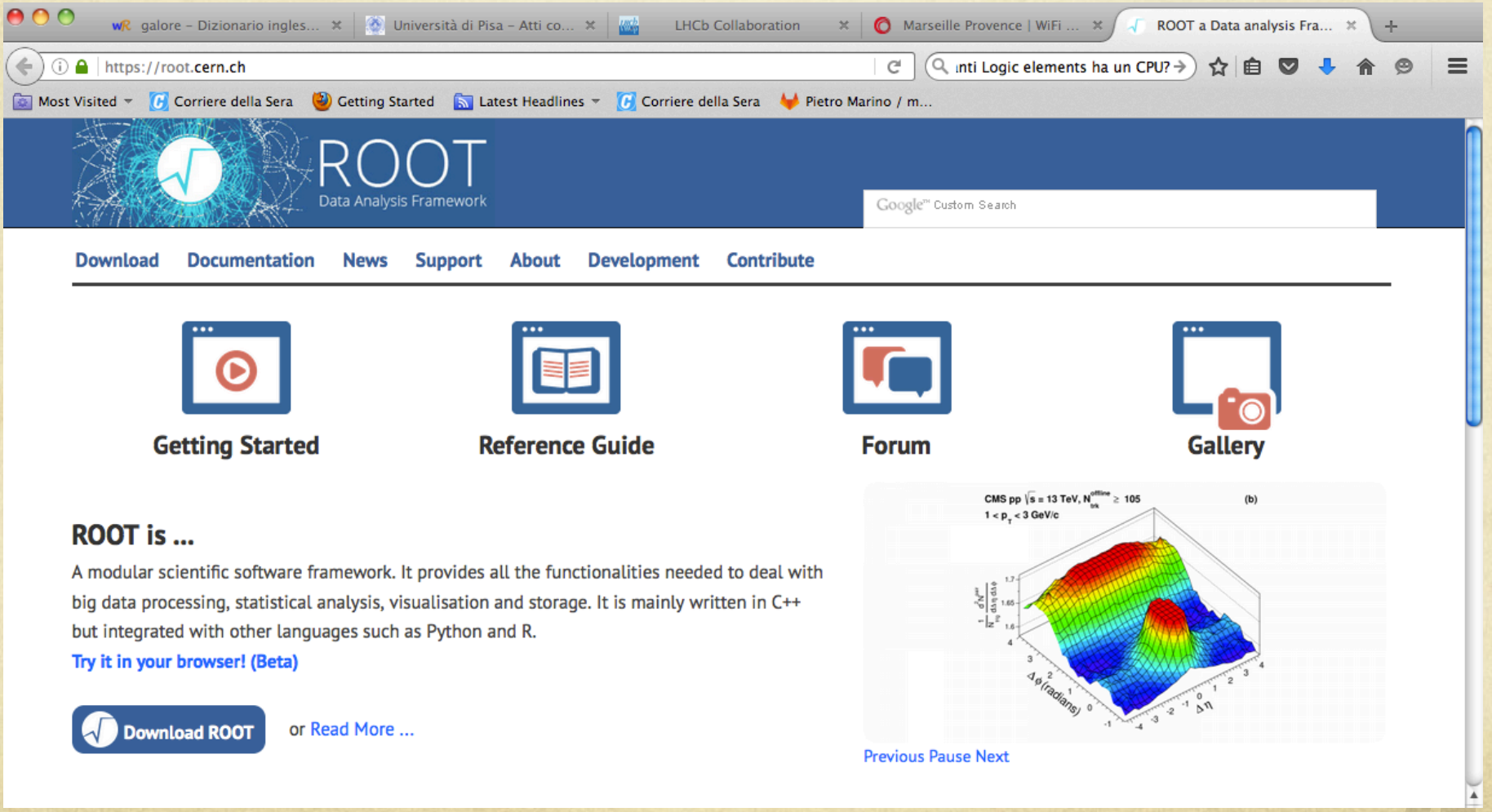
Introduzione

- Le esperienze di Laboratorio sono la parte essenziale del corso, dei veri e propri esperimenti che richiedono di
  - raccogliere i dati
    - setting dell'apparato sperimentale, calibrazione degli strumenti, ecc.
  - analizzare i dati
    - "online" per avere un primo controllo immediato, anche se grossolano
    - "offline" in modo raffinato per estrarre le misure finali
  - scrivere un breve report scientifico (relazione)
    - Il cuore della relazione "dovrebbe" essere scritta durante la presa dati.
- Lo scopo di queste due lezioni è quella di darvi degli strumenti tecnici di base di analisi dati (ROOT) e tipografici (LaTex).

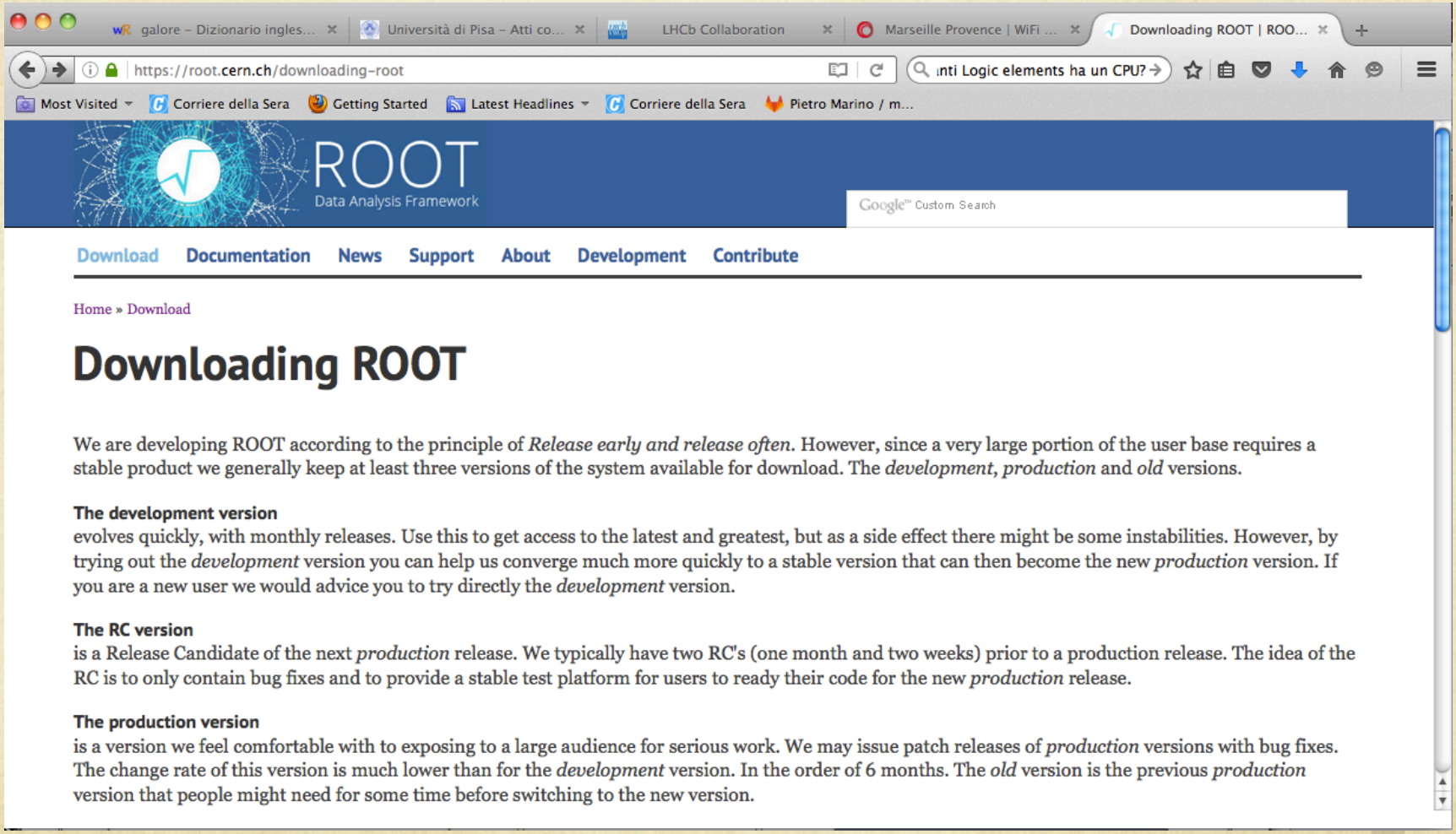
Qualche precisazione

- L'idea è quella di proporre esempi (macro e template) molto semplici e mostrare praticamente il loro funzionamento:
  - leggere e scrivere i dati da un file di testo;
  - riempire e manipolare gli istogrammi (TH1,TH2), i grafici (TGraph, TGraphErrors) e funzioni (TF1,TF2);
  - fare un fit ai dati di minimo chi2;
  - un fit ai dati Binned or Unibinned di Massima Likelihood;
  - creare e leggere un Tree (loop sugli eventi);
  - ....
- Queste (due) lezioni
  - non saranno esaustive, anzi sono il minimo indispensabile
  - non contengono i segreti (e neanche le basi) del C++ e di ROOT
  - toccano solo alcuni piccoli aspetti del linguaggio LaTex.
- Vi mostrerò solo come "utilizzare in pratica" alcuni strumenti.

http://root.cern.ch



Download


galore - Dizionario ingles...
Università di Pisa - Atti co...
LHCb Collaboration
Marseille Provence | WiFi ...
Downloading ROOT | ROO...
https://root.cern.ch/downloading-root
Most Visited
Corriere della Sera
Getting Started
Latest Headlines
Corriere della Sera
Pietro Marino / m...
ROOT
Data Analysis Framework
Google™ Custom Search
Download
Documentation
News
Support
About
Development
Contribute
Home » Download
Downloading ROOT
We are developing ROOT according to the principle of Release early and release often. However, since a very large portion of the user base requires a stable product we generally keep at least three versions of the system available for download. The development, production and old versions.
The development version
evolves quickly, with monthly releases. Use this to get access to the latest and greatest, but as a side effect there might be some instabilities. However, by trying out the development version you can help us converge much more quickly to a stable version that can then become the new production version. If you are a new user we would advice you to try directly the development version.
The RC version
is a Release Candidate of the next production release. We typically have two RC's (one month and two weeks) prior to a production release. The idea of the RC is to only contain bug fixes and to provide a stable test platform for users to ready their code for the new production release.
The production version
is a version we feel comfortable with to exposing to a large audience for serious work. We may issue patch releases of production versions with bug fixes. The change rate of this version is much lower than for the development version. In the order of 6 months. The old version is the previous production version that people might need for some time before switching to the new version.

Download (2)

For each of the three active versions
we provide the full source and pre-compiled binaries for most of the supported platforms. After downloading and unpacking (usually it is enough to double click on the file you downloaded "tar file" or "dmg file") please read the included README/README file on how to set the necessary environment variables.

Basically it is enough to use a special script distributed with ROOT:

- For the sh shell family do: `. <pathname>/root/bin/thisroot.sh`
- For the csh shell family do: `source <pathname>/root/bin/thisroot.csh`

where `<pathname>` is the location where you unpacked the ROOT distribution.

Typically add these lines to your `.profile` or `.login` files.

The following versions are available for download:

Latest ROOT Releases

Pro Release 6.06/08 - 2016-09-01

Old Release 6.04/18 - 2016-06-22

Version 6

Release 6.06/08 - 2016-09-01

Release 6.06/06 - 2016-07-06

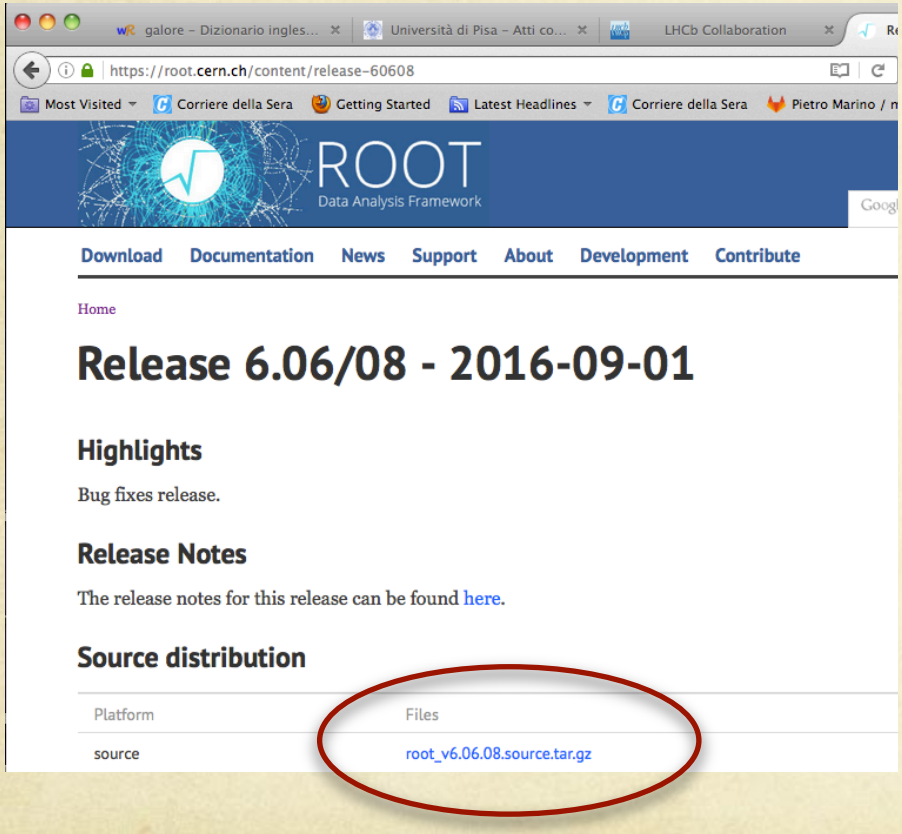
Release 6.04/18 - 2016-06-22

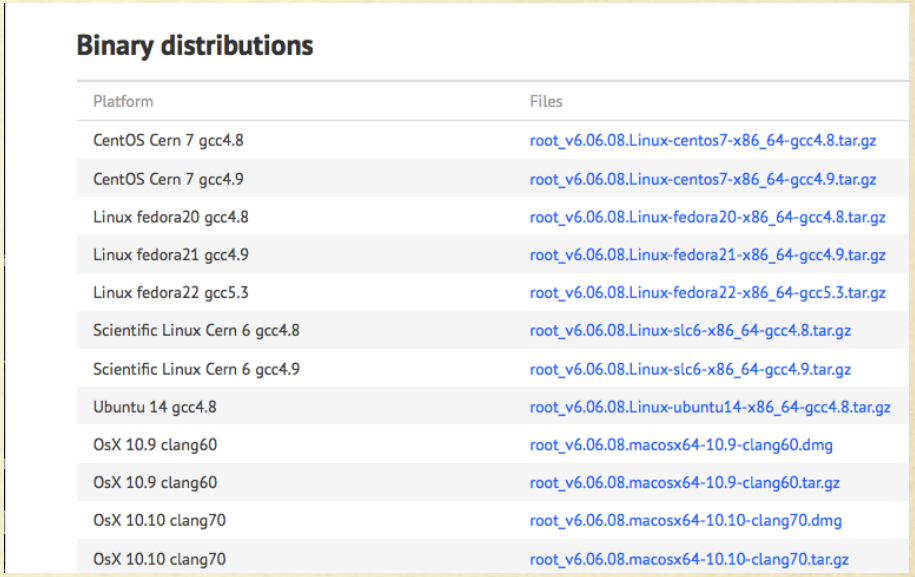
Release 6.06/02 - 2016-03-03

Release 6.04/14 - 2016-02-03

Release 6.06/00 - 2015-12-09

Source code and binaries



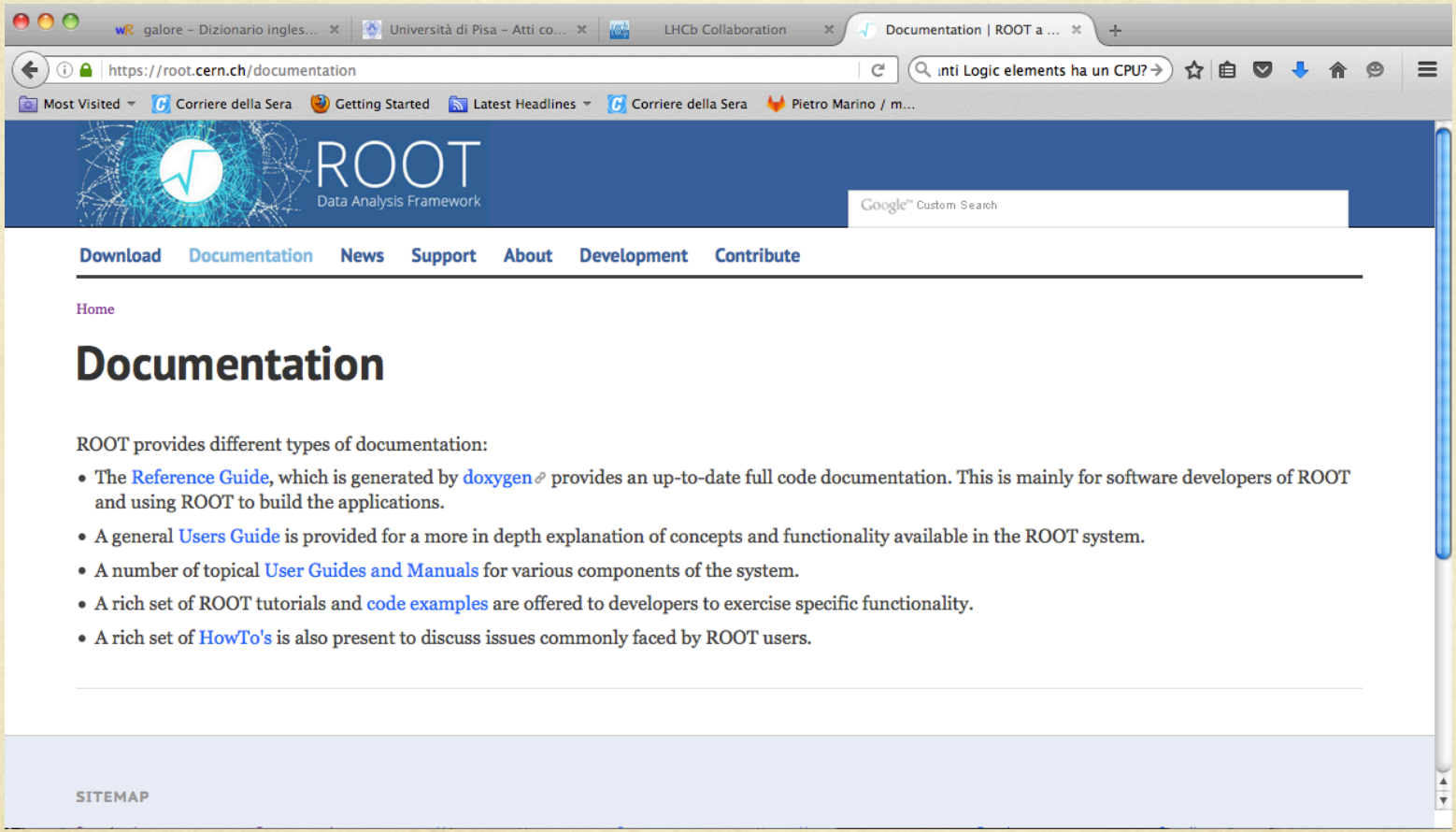


Binary distributions

Platform	Files
CentOS Cern 7 gcc4.8	root_v6.06.08.Linux-centos7-x86_64-gcc4.8.tar.gz
CentOS Cern 7 gcc4.9	root_v6.06.08.Linux-centos7-x86_64-gcc4.9.tar.gz
Linux fedora20 gcc4.8	root_v6.06.08.Linux-fedora20-x86_64-gcc4.8.tar.gz
Linux fedora21 gcc4.9	root_v6.06.08.Linux-fedora21-x86_64-gcc4.9.tar.gz
Linux fedora22 gcc5.3	root_v6.06.08.Linux-fedora22-x86_64-gcc5.3.tar.gz
Scientific Linux Cern 6 gcc4.8	root_v6.06.08.Linux-slc6-x86_64-gcc4.8.tar.gz
Scientific Linux Cern 6 gcc4.9	root_v6.06.08.Linux-slc6-x86_64-gcc4.9.tar.gz
Ubuntu 14 gcc4.8	root_v6.06.08.Linux-ubuntu14-x86_64-gcc4.8.tar.gz
OsX 10.9 clang60	root_v6.06.08.macosx64-10.9-clang60.dmg
OsX 10.9 clang60	root_v6.06.08.macosx64-10.9-clang60.tar.gz
OsX 10.10 clang70	root_v6.06.08.macosx64-10.10-clang70.dmg
OsX 10.10 clang70	root_v6.06.08.macosx64-10.10-clang70.tar.gz

https://root.cern.ch/download/root_v6.06.08.source.tar.gz

Documentation



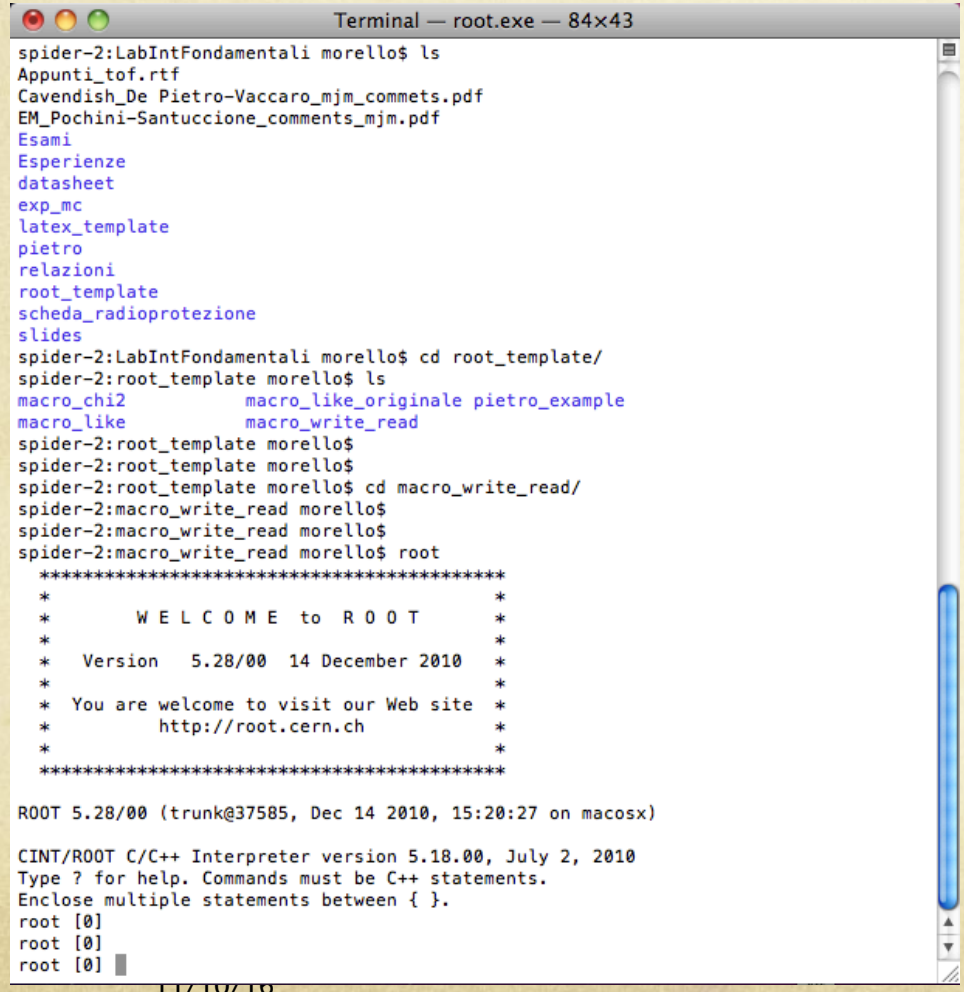
About ROOT

ROOT is a framework for data processing, born at CERN, at the heart of the research on high-energy physics. Every day, thousands of physicists use ROOT applications to analyze their data or to perform simulations. With ROOT you can:

- **Save data** You can save your data (and any C++ object) in a compressed binary form in a ROOT file. The object format is also saved in the same file: the ROOT files are self-descriptive. Even in the case the source files describing the data model are not available, the information contained in a ROOT file is be always readable. ROOT provides a data structure, the *tree*, that is extremely powerful for fast access of huge amounts of data - orders of magnitude faster than accessing a normal file.
- **Access data** Data saved into one or several ROOT files can be accessed from your PC, from the web and from large-scale file delivery systems used e.g. in the GRID. ROOT trees spread over several files can be chained and accessed as a unique object, allowing for loops over huge amounts of data.
- **Mine data** Powerful mathematical and statistical tools are provided to operate on your data. The full power of a C++ application and of parallel processing is available for any kind of data manipulation. Data can also be generated following any statistical distribution and modeled, making it possible to simulate complex systems.
- **Publish results** Results can be displayed with histograms, scatter plots, fitting functions. ROOT graphics may be adjusted real-time by few mouse clicks. Publication-quality figures can be saved in PDF or other formats.
- **Run interactively or build your own application** You can use the Cling C++ interpreter for your interactive sessions and to write macros, or you can compile your program to run at full speed. In both cases, you can also create a graphical user interface.
- **Use ROOT within other languages** ROOT provides a set of bindings in order to seamlessly integrate with existing languages such as Python, R and Mathematica.

Che cosa è ROOT?

"An objected-oriented data analysis framework"



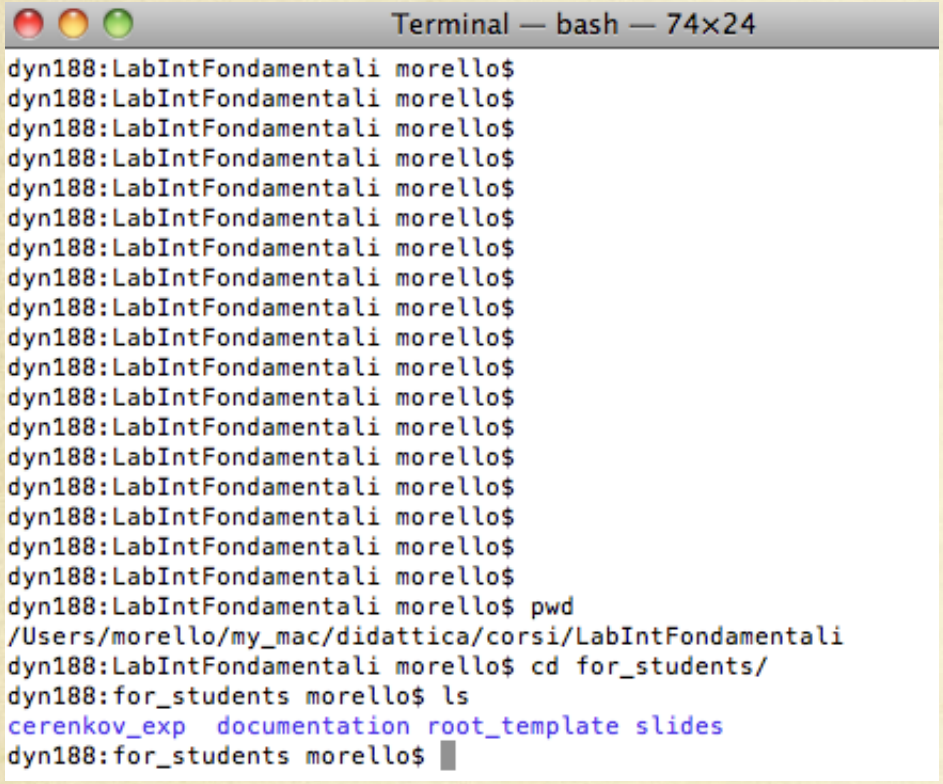
In altre parole ROOT è un insieme di librerie scritte in C++ per manipolare gli istogrammi ed in generale grosse quantità di dati.

E' stato pensato per la Fisica delle Alte Energie e quindi per l'analisi di campioni "enormi" di dati.

ROOT può essere usato in vari modi.

Le librerie possono essere direttamente linkate al vostro programma in C++, esattamente come una vostra libreria personale, oppure tramite il ROOT Prompt.

Step 1



1) Downloading dalla pagina web del corso del file: "**`for_students.tgz`**"

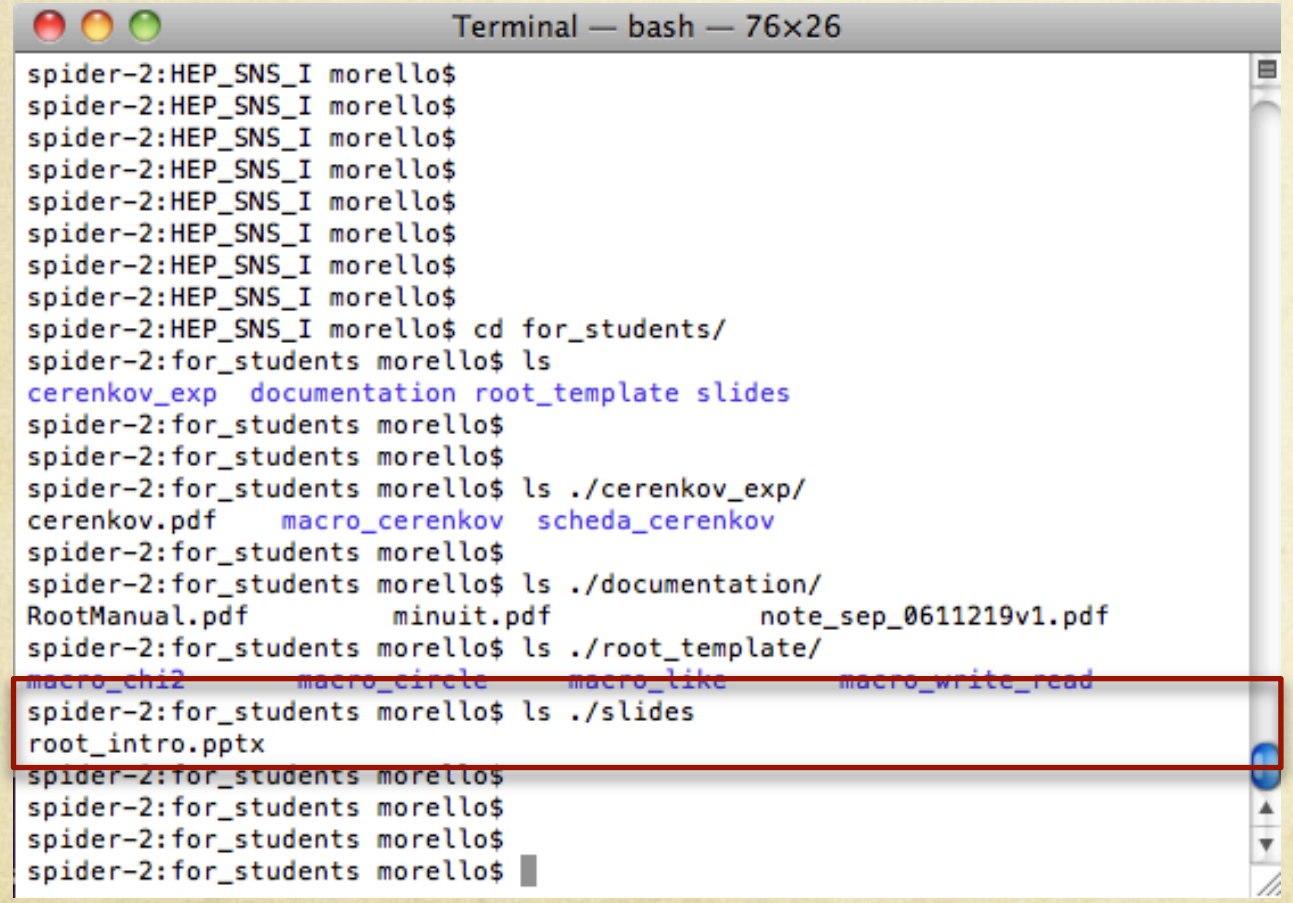
2) Scompattare il file con il comando:

`tar –xvzf for_students.tgz`

3) Ci sono quattro directories

- `cerenkov_exp`
- `documentation`
- `root_template`
- `slides`

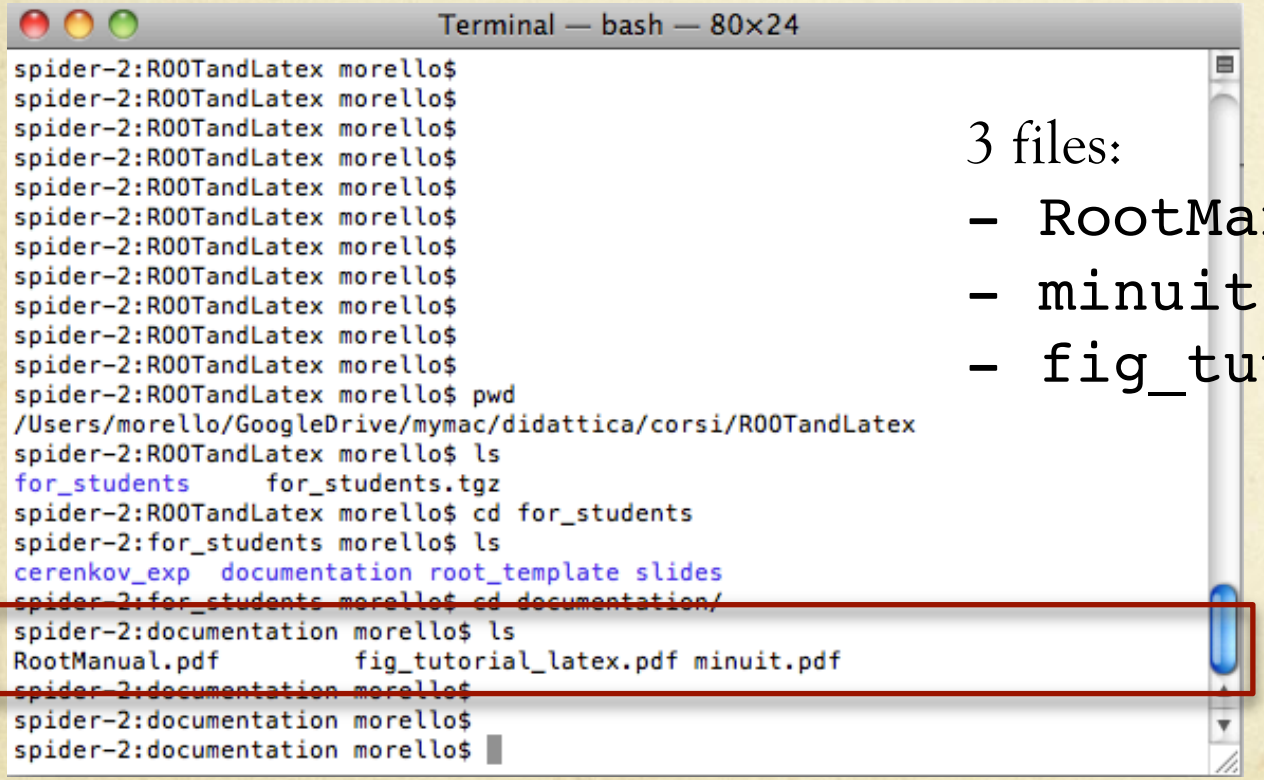
slides



2 file:

- root_intro.pptx
- root_intro.pdf

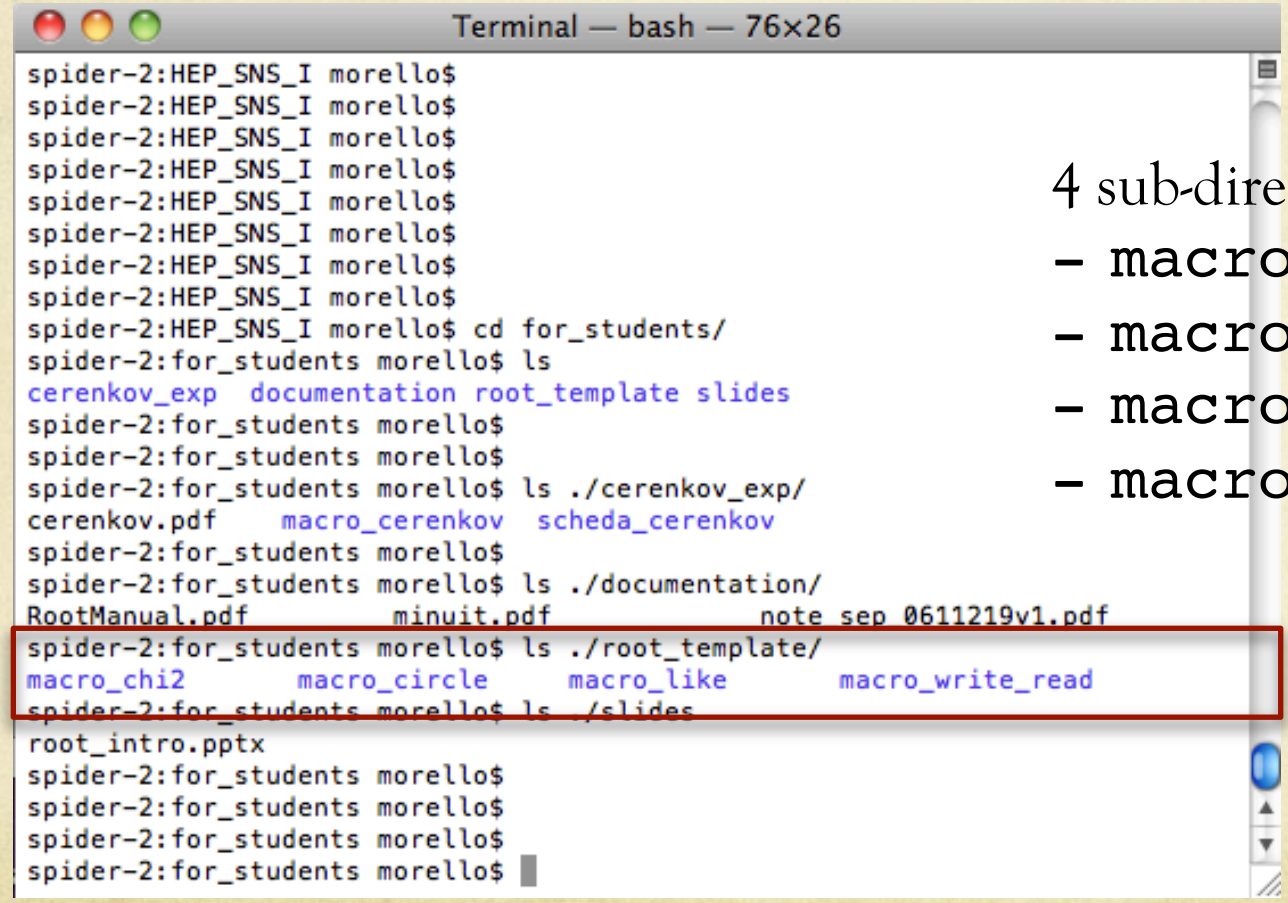
documentation



3 files:

- `RootManual.pdf`
- `minuit.pdf`
- `fig_tutorial_latex.pdf`

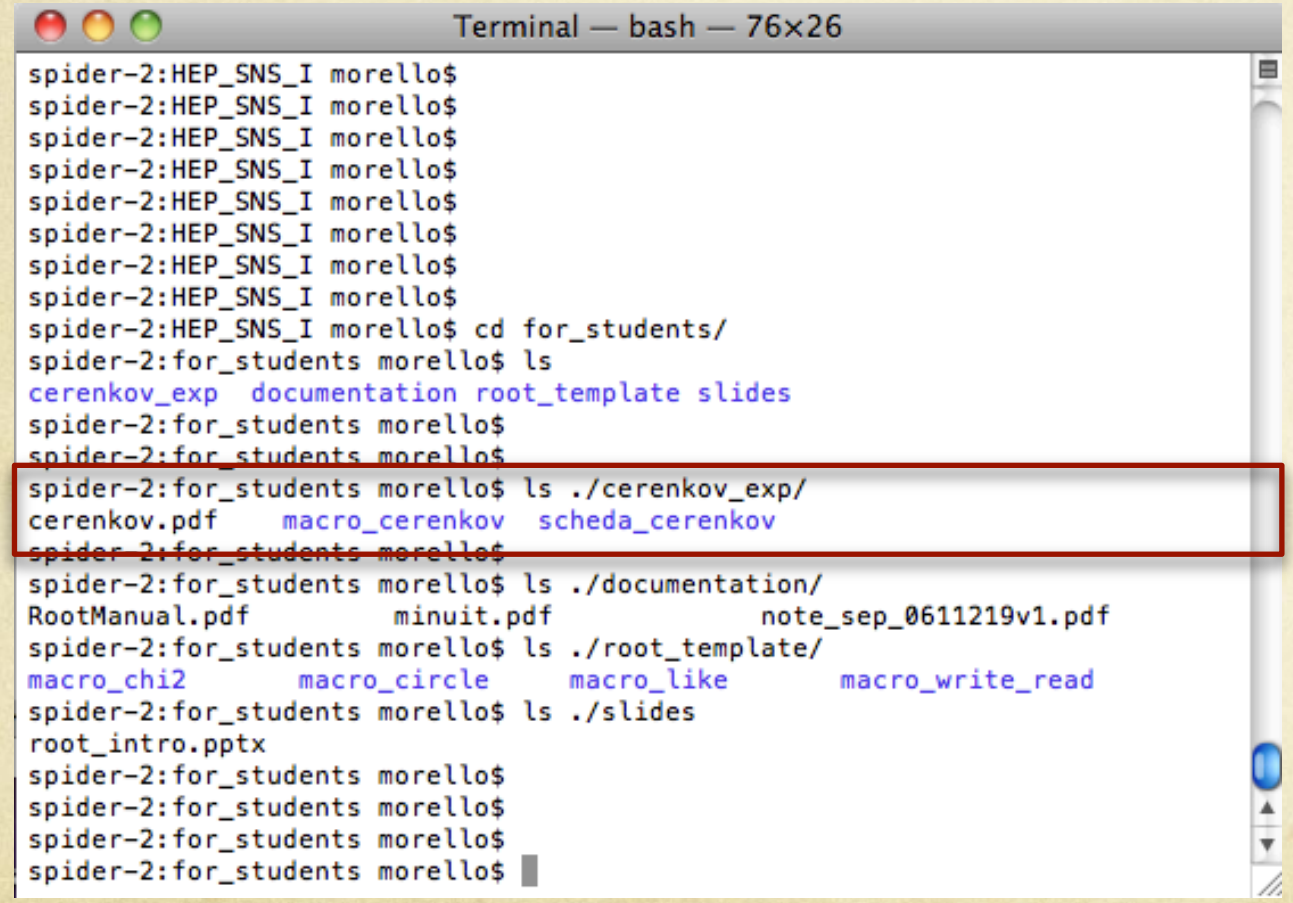
root_template



4 sub-directories:

- `macro_write_read`
- `macro_chi2`
- `macro_circle`
- `macro_like`

cerenkov_exp



2 sub-directories +1 file:

- `macro_cerenkov`
- `scheda_cerenkov`
- `cerenkov.pdf`

root_template

macro_write_read

```
spider-2:test morello$
spider-2:test morello$ cd root_template
spider-2:root_template morello$ ls
macro_chi2       macro_like       macro_write_read
spider-2:root_template morello$
spider-2:root_template morello$
spider-2:root_template morello$ cd macro_write_read/
spider-2:macro_write_read morello$ ls
_fig            _txt            macroRead1.C    macroWrite.C    macroWrite_C.so
_root           labstyle.C      macroRead2.C    macroWrite_C.d
spider-2:macro_write_read morello$
```

- **macroWrite.C (parte I)**
  - Crea un istogramma
  - Genera eventi distribuiti Gauss(0,1)
  - Salva gli eventi su un file testo in _txt/gaussian.txt
  - Riempie l'istogramma, disegna l'istogramma su schermo (o Canvas)
  - Salva l'istogramma su un file .root in _root/histo.root

macro_write_read

```
spider-2:test morello$
spider-2:test morello$ cd root_template
spider-2:root_template morello$ ls
macro_chi2       macro_like       macro_write_read
spider-2:root_template morello$
spider-2:root_template morello$
spider-2:root_template morello$ cd macro_write_read/
spider-2:macro_write_read morello$ ls
_fig             _txt             macroRead1.C     macroWrite.C     macroWrite_C.so
_root            labstyle.C       macroRead2.C     macroWrite_C.d
spider-2:macro_write_read morello$
```

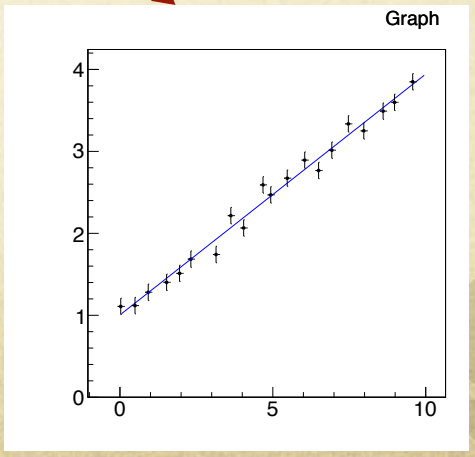
- **macroWrite.C (parte II)**
  - Crea una funzione lineare y=mx+q, disegna la funzione su schermo
  - Prende 20 punti della funzione lineare equi spaziati in (x_i,y_i)
  - Tutti gli (x_i,y_i) sono smearati con una funzione gaussiana sia lungo x che y
  - Salva gli eventi su un file testo in _txt/graph.txt (20 linee con 4 colonne)
  - Crea un TGraphErrors che viene riempito con gli (x_i,y_i) smearati ed i relativi errori.
  - Fa il fit del TGraphErrors con una funzione lineare
  - Salva il risultato del fit sovrapposto ai punti su un file _fig/graph.pdf

macroWrite.C

```
spider-2:macro_write_read morello$
spider-2:macro_write_read morello$
spider-2:macro_write_read morello$ ls
_fig            _txt            macroRead1.C    macroWrite.C    macroWrite_C.so
_root           labstyle.C      macroRead2.C    macroWrite_C.d
spider-2:macro_write_read morello$ root -l
root [0] .L macroWrite.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/corsi/LabIntFondamentali/te
st/root_template/macro_write_read/./macroWrite_C.so
root [1] macroWrite()
 FCN=13.1612 FROM MIGRAD    STATUS=CONVERGED      36 CALLS          37 TOTAL
                     EDM=6.41676e-13    STRATEGY= 1      ERROR MATRIX ACCURATE
  EXT PARAMETER                                   STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  p0           1.00130e+00   4.48183e-02   4.28747e-05   4.40909e-05
   2  p1           2.98180e-01   8.10437e-03   7.75291e-06   2.66697e-04
Info in <TCanvas::Print>: pdf file ./_fig/graph.pdf has been created
Info in <TCanvas::Print>: eps file ./_fig/graph.eps has been created
root [2]
```

- .L macroWrite.C++
- macroWrite()

```
root [2] .q
spider-2:macro_write_read morello$
spider-2:macro_write_read morello$ ls
_fig            _txt            macroRead1.C    macroWrite.C    macroWrite_C.so
_root           labstyle.C      macroRead2.C    macroWrite_C.d
spider-2:macro_write_read morello$ ls _fig/
graph.eps graph.pdf
spider-2:macro_write_read morello$ ls _txt
gaussian.txt graph.txt
spider-2:macro_write_read morello$ ls _root
histo.root
spider-2:macro_write_read morello$
```



TGraphErrors fit:

In case of a **TGraphErrors** object, when x errors are present, the error along x, is projected along the y-direction by calculating the function at the points x-exlow and x+exhigh. The chisquare is then computed as the sum of the quantity below at each point:

$$\frac{(y-f(x))^2}{ey^2+(\frac{1}{2}(exl+exh)f'(x))^2}$$

where x and y are the point coordinates, and f'(x) is the derivative of the function f(x).

In case the function lies below (above) the data point, ey is ey_low (ey_high).

thanks to Andy Haas (haas@yahoo.com) for adding the case with **TGraphAsymmErrors** University of Washington

The approach used to approximate the uncertainty in y because of the errors in x is to make it equal the error in x times the slope of the line. The improvement, compared to the first method (f(x+ exhigh) - f(x-exlow))/2 is of (error of x)**2 order. This approach is called "effective variance method". This improvement has been made in version 4.00/08 by Anna Kreshuk. The implementation is provided in the function FitUtil::EvaluateChi2Effective

NOTE:

1. By using the "effective variance" method a simple linear regression becomes a non-linear case, which takes several iterations instead of 0 as in the linear case.
2. The effective variance technique assumes that there is no correlation between the x and y coordinate.
3. The standard chi2 (least square) method without error in the coordinates (x) can be forced by using option "EX0"
4. The linear fitter doesn't take into account the errors in x. When fitting a **TGraphErrors** with a linear functions the errors in x willnot be considere. If errors in x are important, go through minuit (use option "F" for polynomial fitting).
5. When fitting a **TGraph** (i.e. no errors associated with each point), a correction is applied to the errors on the parameters with the following formula: errorp *= sqrt(chisquare/(ndf-1))

https://root.cern.ch/doc/master/classTGraph.html

Access to the fit result

The function returns a **TFitResultPtr** which can hold a pointer to a **TFitResult** object. By default the **TFitResultPtr** contains only the status of the fit which is return by an automatic conversion of the **TFitResultPtr** to an integer. One can write in this case directly:

```
Int_t fitStatus =  h->Fit(myFunc)
```

If the option "S" is instead used, **TFitResultPtr** contains the **TFitResult** and behaves as a smart pointer to it. For example one can do:

```
TFitResultPtr r = h->Fit(myFunc,"S");
TMatrixDSym cov = r->GetCovarianceMatrix();  //  to access the covariance matrix
Double_t chi2   = r->Chi2(); // to retrieve the fit chi2
Double_t par0   = r->Value(0); // retrieve the value for the parameter 0
Double_t err0   = r->ParError(0); // retrieve the error for the parameter 0
r->Print("V");     // print full information of fit including covariance matrix
r->Write();        // store the result in a file
```

The fit parameters, error and chi2 (but not covariance matrix) can be retrieved also from the fitted function. If the histogram is made persistent, the list of associated functions is also persistent. Given a pointer (see above) to an associated function myfunc, one can retrieve the function/fit parameters with calls such as:

```
Double_t chi2 = myfunc->GetChisquare();
Double_t par0 = myfunc->GetParameter(0); //value of 1st parameter
Double_t err0 = myfunc->GetParError(0);  //error on first parameter
```

https://root.cern.ch/doc/master/classTGraph.html

macro_write_read

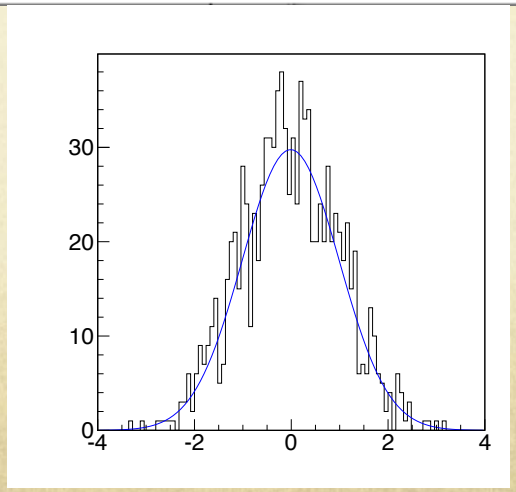
```
spider-2:test morello$
spider-2:test morello$ cd root_template
spider-2:root_template morello$ ls
macro_chi2       macro_like       macro_write_read
spider-2:root_template morello$
spider-2:root_template morello$
spider-2:root_template morello$ cd macro_write_read/
spider-2:macro_write_read morello$ ls
_fig             _txt             macroRead1.C     macroWrite.C     macroWrite_C.so
_root            labstyle.C       macroRead2.C     macroWrite_C.d
spider-2:macro_write_read morello$
```

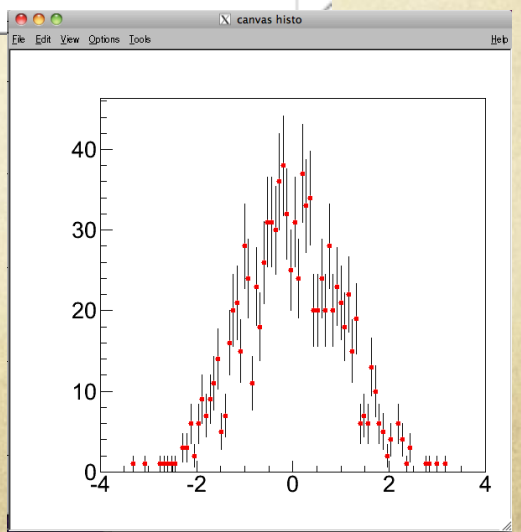
- **macroRead1.C**
  - Legge gli eventi dal dal file _txt/gaussian.txt
  - Crea un istogramma
  - Riempie l'istogramma, disegna su schermo l'istogramma
  - Fa il fit dell'istogramma e lo salva in _fig/histo_gauss.pdf
  - Salva l'istogramma nel file _root/histo.root

macroRead1.C

```
spider-2:macro_write_read morello$ ls
_fig            _txt            macroRead1.C    macroWrite.C    macroWrite_C.so
_root           labstyle.C      macroRead2.C    macroWrite_C.d
spider-2:macro_write_read morello$ root -l
root [0] .L macroRead1.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica
/corsi/LabIntFondamentali/root_template/macro_write_read/./macroRead1_C.so
root [1] macroRead1()
 FCN=78.4938 FROM MIGRAD    STATUS=CONVERGED      71 CALLS          72 TOTAL
                     EDM=8.48465e-10    STRATEGY= 1      ERROR MATRIX ACCURATE
  EXT PARAMETER                                   STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  p0           7.46241e+01   2.46312e+00   1.07021e-02  -8.33034e-06
   2  p1          -1.22650e-02   3.42194e-02   1.48924e-04   9.22230e-04
   3  p2           1.00021e+00   2.83871e-02   1.23319e-04  -5.87359e-04
Info in <TCanvas::Print>: pdf file ./_fig/histo_gauss.pdf has been created
Info in <TCanvas::Print>: eps file ./_fig/histo_gauss.eps has been created
root [2]
```

- .L macroRead1.C++
- macroRead1()





macro_write_read

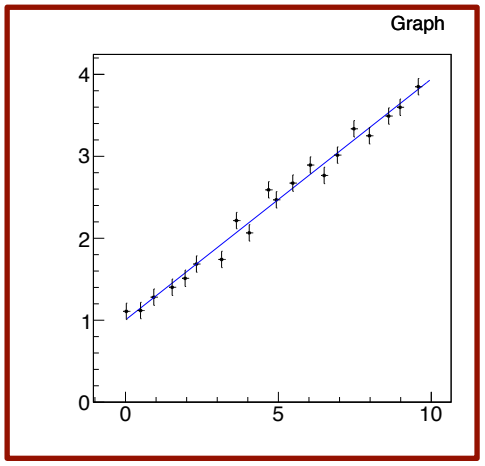
```
spider-2:test morello$
spider-2:test morello$ cd root_template
spider-2:root_template morello$ ls
macro_chi2       macro_like       macro_write_read
spider-2:root_template morello$
spider-2:root_template morello$
spider-2:root_template morello$ cd macro_write_read/
spider-2:macro_write_read morello$ ls
_fig            _txt            macroRead1.C    macroWrite.C    macroWrite_C.so
_root           labstyle.C      macroRead2.C    macroWrite_C.d
spider-2:macro_write_read morello$
```

- **macroRead2.C**
  - Legge le colonne del file di testo _txt/graph.txt (x_val,y_val,x_err,y_err)
  - Crea e riempie un TGraphErrors
  - Crea una funzione lineare y=mx+q,
  - Fa il fit del TGraphErrors con una funzione lineare
  - Salva il risultato del fit sovrapposto ai punti su un file _fig/graph.pdf

macroRead2.C

```
spider-2:macro_write_read morello$
spider-2:macro_write_read morello$
spider-2:macro_write_read morello$ root -l
root [0] .L macroRead2.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/corsi/LabIntFondamentali/roo
t_template/macro_write_read/./macroRead2_C.so
root [1] macroRead2()
event 0 :  0.0259935 1.10815 0.1 0.1
event 1 :  0.491726 1.11823 0.1 0.1
event 2 :  0.925336 1.28123 0.1 0.1
event 3 :  1.52565 1.40204 0.1 0.1
event 4 :  1.94923 1.51135 0.1 0.1
event 5 :  2.31936 1.68549 0.1 0.1
event 6 :  3.14253 1.74152 0.1 0.1
event 7 :  3.62687 2.21595 0.1 0.1
event 8 :  4.04578 2.06574 0.1 0.1
event 9 :  4.67795 2.59106 0.1 0.1
event 10 :  4.93149 2.46919 0.1 0.1
event 11 :  5.47031 2.67313 0.1 0.1
event 12 :  6.04227 2.89347 0.1 0.1
event 13 :  6.49627 2.76662 0.1 0.1
event 14 :  6.92894 3.01465 0.1 0.1
event 15 :  7.46943 3.33579 0.1 0.1
event 16 :  7.97459 3.25076 0.1 0.1
event 17 :  8.60595 3.49138 0.1 0.1
event 18 :  8.97883 3.59847 0.1 0.1
event 19 :  9.57428 3.84934 0.1 0.1
 FCN=18.9745 FROM MIGRAD    STATUS=CONVERGED      36 CALLS          37 TOTAL
                     EDM=8.65783e-10    STRATEGY= 1      ERROR MATRIX ACCURATE
  EXT PARAMETER                                   STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  p0           1.00397e+00   4.48915e-02   5.08611e-05   1.64232e-03
   2  p1           2.93942e-01   8.06019e-03   9.13191e-06   9.84273e-03
Info in <TCanvas::Print>: pdf file ./_fig/graph.pdf has been created
Info in <TCanvas::Print>: eps file ./_fig/graph.eps has been created
root [2]
```

- .L macroRead2.C++
- macroRead2()



macro_chi2

```
spider-2:root_template morello$
spider-2:root_template morello$ ls
macro_chi2       macro_like       macro_write_read
spider-2:root_template morello$ cd macro_chi2/
spider-2:macro_chi2 morello$ root -l
root [0] .L fitter.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/corsi/LabIntFondamentali/roo
t_template/macro_chi2/./fitter_C.so
fitter()
root [1] fitter()
```

- **fitter.C**
  - Crea e disegna una funzione TF1
    - p.d.f. = fs*(f1*Gauss1+(1-f1)*Gauss2) + (1-fs)*Exp
  - Genera 100000 eventi distribuiti secondo p.d.f (Von Neumann)
  - Riempie un istogramma con gli eventi generati
  - Fa il fit di minimo chi2 dell'istogramma per stimare mu1, mu2, sigma1, sigma2, f1, fs, e la normalizzazione totale N.
  - Estrae i parametri ritornati dal fit e la matrice di covarianza
  - Salva su un file .pdf il grafico con il fit sovrapposto.

fitter.C

```
spider-2:root_template morello$ ls
macro_chi2       macro_like       macro_write_read
spider-2:root_template morello$ cd macro_chi2/
spider-2:macro_chi2 morello$ ls
_fig        fitter.C    fitter_C.d  fitter_C.so labstyle.C
spider-2:macro_chi2 morello$ root -l
.L root [0] .L fitter.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/corsi/LabIntFondamentali/roo
t_template/macro_chi2/./fitter_C.so
fitter()root [1] fitter()
Info in <TCanvas::Print>: pdf file ./_fig/pdf_gen.pdf has been created
Info in <TCanvas::Print>: eps file ./_fig/pdf_gen.eps has been created
Info in <TCanvas::Print>: pdf file ./_fig/histo_gen.pdf has been created
Info in <TCanvas::Print>: eps file ./_fig/histo_gen.eps has been created
####################  Minimal Fit ##################
 FCN=80.3461 FROM MIGRAD    STATUS=CONVERGED     166 CALLS         167 TOTAL
                     EDM=4.39286e-09    STRATEGY= 1  ERROR MATRIX UNCERTAINTY   1.8 per cent
  EXT PARAMETER                                   STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  N            9.99270e+04   3.17594e+02   4.19501e-02   1.77947e-07
   2  fs           3.00992e-01   1.96364e-03  -6.43839e-07  -1.84262e-02
   3  fA           7.00790e-01   5.35848e-03   8.97058e-08   1.17666e-02
   4  mA           4.00586e+00   3.61409e-03   1.95843e-07   3.09787e-05
   5  sA           2.99822e-01   3.51452e-03   7.77640e-07  -8.24862e-03
   6  mB           5.00537e+00   7.07907e-03  -4.79003e-07  -1.35219e-02
   7  sB           2.98593e-01   6.04152e-03   2.14531e-06  -9.55633e-03
   8  slope       -5.01805e-01   2.55737e-03   1.22475e-08  -2.89605e-02
chi2 = 80.3461
ndof = 92
prob = 0.80196
####################  Fit and Covariance Matrix ##################
 FCN=80.3461 FROM MIGRAD    STATUS=CONVERGED     115 CALLS         116 TOTAL
                     EDM=1.097e-10    STRATEGY= 1      ERROR MATRIX ACCURATE
  EXT PARAMETER                                   STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  N            9.99270e+04   3.16124e+02   1.39011e+00   2.93571e-08
   2  fs           3.00992e-01   1.94698e-03   7.76976e-06   6.80867e-04
   3  fA           7.00790e-01   5.31790e-03   1.44603e-05  -5.04417e-04
   4  mA           4.00586e+00   3.60483e-03   1.25458e-05  -5.97243e-04
   5  sA           2.99822e-01   3.50221e-03   1.10604e-05  -1.66983e-03
   6  mB           5.00537e+00   6.97345e-03   2.09326e-05   2.88188e-04
   7  sB           2.98593e-01   6.00508e-03   1.79762e-05  -9.36678e-04
   8  slope       -5.01805e-01   2.45620e-03   9.97184e-06   3.80871e-03
```

fitter.C

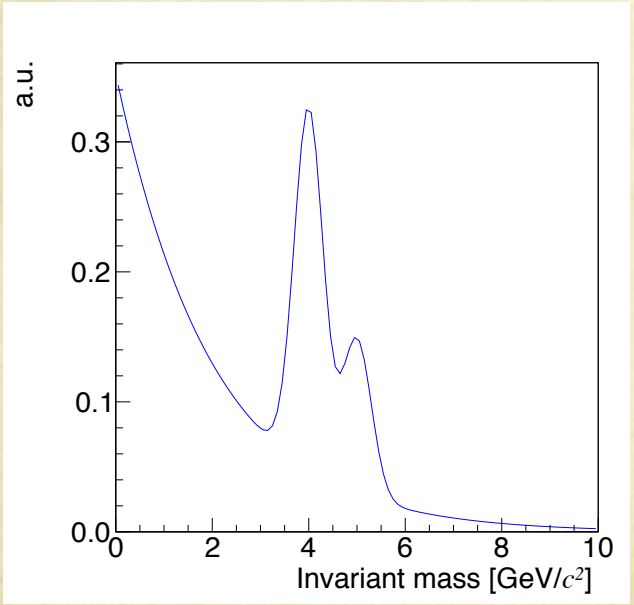
```
*****************************************
Minimizer is Minuit / Migrad
Chi2                      =      80.3461
NDf                       =           92
Edm                       =    1.097e-10
NCalls                    =          116
N                         =        99927   +/-   316.124
fs                        =     0.300992   +/-   0.00194698
fA                        =      0.70079   +/-   0.0053179
mA                        =      4.00586   +/-   0.00360483
sA                        =     0.299822   +/-   0.00350221
mB                        =      5.00537   +/-   0.00697345
sB                        =     0.298593   +/-   0.00600508
slope                     =    -0.501805   +/-   0.0024562

Covariance Matrix:

                        N          fs          fA          mA          sA          mB          sB       slope
N                   99934  4.3443e-07 -0.00010304 -5.3285e-05 -3.9547e-05   -9.06e-05  9.4243e-05 -0.00016452
fs             4.3443e-07  3.7907e-06 -1.4402e-07 -6.4511e-07  1.2408e-06  7.9173e-07  1.1344e-06 -1.6878e-06
fA            -0.00010304 -1.4402e-07   2.828e-05  9.9785e-06   1.168e-05  2.3561e-05 -2.1781e-05   1.051e-06
mA            -5.3285e-05 -6.4511e-07  9.9785e-06  1.2995e-05  4.9059e-06  1.3517e-05  -1.127e-05  6.1692e-07
sA            -3.9547e-05  1.2408e-06   1.168e-05  4.9059e-06  1.2265e-05  1.4288e-05 -1.0088e-05 -7.5954e-07
mB              -9.06e-05  7.9173e-07  2.3561e-05  1.3517e-05  1.4288e-05  4.8629e-05 -2.4711e-05 -5.0162e-07
sB             9.4243e-05  1.1344e-06 -2.1781e-05  -1.127e-05 -1.0088e-05 -2.4711e-05  3.6061e-05 -1.9464e-06
slope         -0.00016452 -1.6878e-06   1.051e-06  6.1692e-07 -7.5954e-07 -5.0162e-07 -1.9464e-06  6.0329e-06

Correlation Matrix:

                        N          fs          fA          mA          sA          mB          sB       slope
N                       1  7.0583e-07 -6.1294e-05 -4.6759e-05  -3.572e-05 -4.1098e-05  4.9645e-05 -0.00021188
fs             7.0583e-07           1    -0.01391   -0.091915     0.18197    0.058313    0.097029    -0.35293
fA            -6.1294e-05    -0.01391           1     0.52053     0.62716     0.63533    -0.68206    0.080465
mA            -4.6759e-05   -0.091915     0.52053           1     0.38859      0.5377    -0.52061    0.069676
sA             -3.572e-05     0.18197     0.62716     0.38859           1     0.58505    -0.47966   -0.088297
mB            -4.1098e-05    0.058313     0.63533      0.5377     0.58505           1     -0.5901   -0.029286
sB             4.9645e-05    0.097029    -0.68206    -0.52061    -0.47966     -0.5901           1    -0.13196
slope         -0.00021188    -0.35293    0.080465    0.069676   -0.088297   -0.029286    -0.13196           1

chi2 = 80.3461
par0 = 99927
err0 = 316.124
cov(2,3)=4.34429e-07
Info in <TCanvas::Print>: pdf file ./_fig/histo_fit.pdf has been created
Info in <TCanvas::Print>: eps file ./_fig/histo_fit.eps has been created
root [2]
```

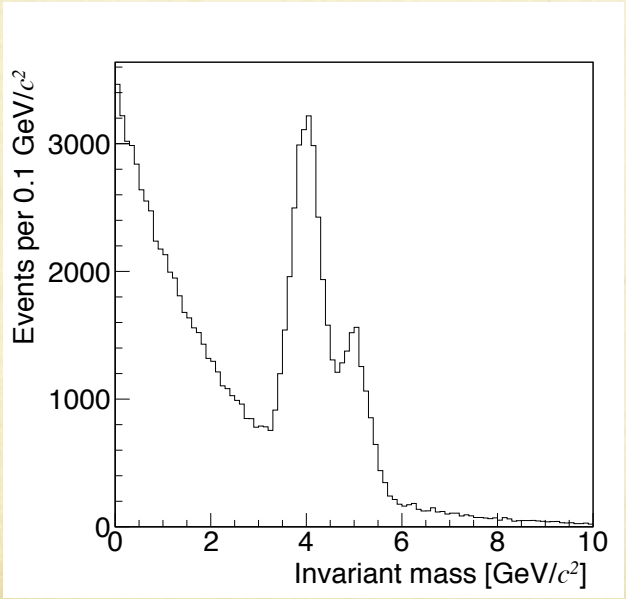
fitter.C: output

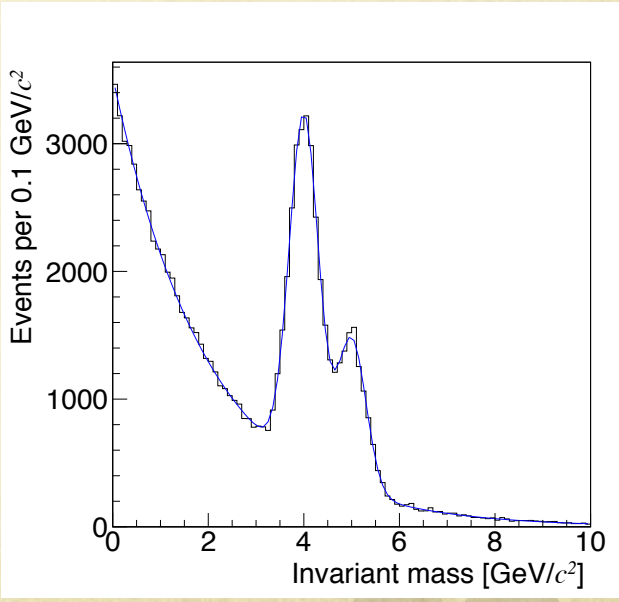
Probability Density Function:
TF1 → una funzione 1-dim, normalizzata a 1. 2 gaussiane + fondo esponenziale

Istogramma:
TH1 → generati eventi distribuiti secondo il modello (p.d.f) e riempito l'oggetto istogramma.

Fit dell'istogramma:
Stima dei parametri del modello: mu e sigma delle gaussiane, slope della exp, e frazioni relative, norm totale.







Likelihood fit in a nutshell

Consider a random variable x distributed according to a p.d.f. $f(x;\theta)$. Suppose that the functional form of $f(x;\theta)$ is known, but the value of at least of one parameter (or parameters $\theta = (\theta_1, ..., \theta_m)$) are not known. Suppose a measurement of the random variable x has been repeated n times, yielding the vaules $x_1, ..., x_n$. Here, x could also represent a multidimensional random vector, i.e. the outcome of each individual observation could be characterized by several quantities.

$$\text{probability that } x_i \text{ in } [x_i, x_i + dx_i] \text{ for all } i \; = \prod_{i=1}^{n} f(x_i, \theta) dx_i$$

If the hypothesized p.d.f. and paremeter value are correct, one expects a high probability for tht data that were actually measured. Conversely a parameter value far away from the true value should yield a low probability for the measurements obtained. Since dhe dx_i do not depend on the parameters, the same reasoning applies to the following function L,

$$L(\theta) = \prod_{i=1}^{n} f(x_i, \theta)$$

called **likelihood function**. With this motivation one defines the maximum likelihood (ML) estimators for theparameters to be those which maximize the likelihood function.

$$\max \log L \quad \Longrightarrow \quad \min -2 \log L \quad \Longrightarrow \quad \hat{\theta}.$$

In the case of a sufficiently large data sample, one can estimate V^{-1} by evaluating the second derivative with measured data and the ML estimators $\hat{\theta}$

$$\widehat{(V^{-1})}_{ij} = - \left[\frac{\partial^2 \log L}{\partial \theta_i \partial \theta_j} \right]_{\theta = \hat{\theta}}$$

for a single parameter θ this reduces to

$$\widehat{\sigma^2_{\hat{\theta}}} = \left[- \frac{1}{\frac{\partial^2 \log L}{\partial \theta^2}} \right]_{\theta = \hat{\theta}}$$

macro_like

```
Terminal — root.exe — 92×25
spider-2:macro_like morello$ ls
_fig              fitterLike2.C     fitterLike3.C     fitterLike_C.d    mvb.C
_txt              fitterLike2_C.d   fitterLike3_C.d   fitterLike_C.so   mvb_C.d
fitterLike.C      fitterLike2_C.so  fitterLike3_C.so  labstyle.C        mvb_C.so
spider-2:macro_like morello$ root -l
root [0] .L fitterLike.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/corsi/
HEP_SNS_I/for_students/root_template/macro_like/./fitterLike_C.so
root [1] fitterLike()
 PARAMETER DEFINITIONS:
    NO.   NAME         VALUE      STEP SIZE      LIMITS
     1 fA           3.00000e-01  1.00000e-04    0.00000e+00  1.00000e+00
```

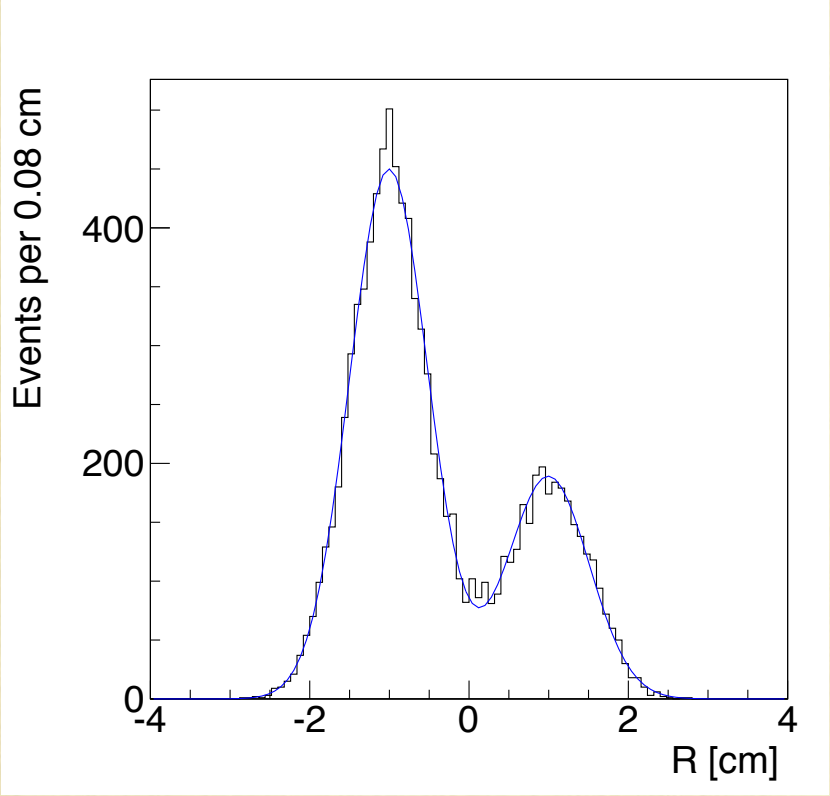
- **fitterLike.C**
  - Genera 10000 eventi distribuiti come la somma di due gaussiane
  - Fa il fit Unbinned di Maximum Likelihood degli eventi generati
  - Usa “TMinuit” in ROOT
  - Accede “direttamente” a quasi tutte le routine di MINUIT che è il software (Fortran) che minimizza numericamente una funzione.

macro_like

```
Terminal — root.exe — 110×18
spider-2:macro_like morello$
spider-2:macro_like morello$
spider-2:macro_like morello$ ls
_fig              fitterLike2.C     fitterLike3.C     fitterLike_C.d    mvb.C
_txt              fitterLike2_C.d   fitterLike3_C.d   fitterLike_C.so   mvb_C.d
fitterLike.C      fitterLike2_C.so  fitterLike3_C.so  labstyle.C        mvb_C.so
spider-2:macro_like morello$ root -l
root [0] .L fitterLike2.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/corsi/HEP_SNS_I/for_stud
ents/root_template/macro_like/./fitterLike2_C.so
fitterLiroot [1] fitterLike2()
 PARAMETER DEFINITIONS:
    NO.   NAME         VALUE      STEP SIZE      LIMITS
     1 fA           3.00000e-01  1.00000e-04    0.00000e+00  1.00000e+00
     2 mA          -1.00000e+00  1.00000e-04     no limits
     3 mB           1.00000e+00  1.00000e-04     no limits
     4 sA           5.00000e-01  1.00000e-04     no limits
     5 sB           5.00000e-01  1.00000e-04     no limits
```

- **fitterLike2.C**
  - Genera 10000 eventi distribuiti come la somma di due gaussiane
  - Fa il fit Unbinned di Maximum Likelihood degli eventi generati
  - Usa “TFitter” in ROOT
  - Accede solo alle routine “principali” di MINUIT
  - TFitter può in generale usare un minimizzatore diverso da MINUIT.

fitterLike2.C




```
Terminal — bash — 86×48
spider-2:macro_like morello$ root -l
root [0] .L fitterLike2.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/
corsi/HEP_SNS_I/for_students/root_template/macro_like/./fitterLike2_C.so
root [1] fitterLike2()
 PARAMETER DEFINITIONS:
    NO.   NAME         VALUE      STEP SIZE      LIMITS
     1 fA           3.00000e-01  1.00000e-04    0.00000e+00  1.00000e+00
     2 mA          -1.00000e+00  1.00000e-04     no limits
     3 mB           1.00000e+00  1.00000e-04     no limits
     4 sA           5.00000e-01  1.00000e-04     no limits
     5 sB           5.00000e-01  1.00000e-04     no limits
 **********
 **    1 **MIGRAD           0         0.1
 **********
 FIRST CALL TO USER FUNCTION AT NEW START POINT, WITH IFLAG=4.
 START MIGRAD MINIMIZATION.  STRATEGY  1.  CONVERGENCE WHEN EDM .LT. 1.00e-04
 FCN=31882.6 FROM MIGRAD    STATUS=INITIATE       20 CALLS          21 TOTAL
                     EDM= unknown      STRATEGY= 1      NO ERROR MATRIX
  EXT PARAMETER                CURRENT GUESS       STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  fA           3.00000e-01   1.00000e-04   2.18218e-04  -1.61205e+04
   2  mA          -1.00000e+00   1.00000e-04   1.00000e-04   2.34039e+03
   3  mB           1.00000e+00   1.00000e-04   1.00000e-04   2.46996e+03
   4  sA           5.00000e-01   1.00000e-04   1.00000e-04   3.66457e+03
   5  sB           5.00000e-01   1.00000e-04   1.00000e-04  -4.85981e+03
 MIGRAD MINIMIZATION HAS CONVERGED.
 MIGRAD WILL VERIFY CONVERGENCE AND ERROR MATRIX.
 COVARIANCE MATRIX CALCULATED SUCCESSFULLY
 FCN=25608.2 FROM MIGRAD    STATUS=CONVERGED      139 CALLS         140 TOTAL
                     EDM=1.282e-05    STRATEGY= 1      ERROR MATRIX ACCURATE
  EXT PARAMETER                                        STEP         FIRST
  NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
   1  fA           6.97066e-01   5.29583e-03   8.13905e-04  -3.56226e-01
   2  mA          -1.00502e+00   7.10343e-03   4.91875e-04   1.07155e-01
   3  mB           9.96012e-01   1.26247e-02   8.13277e-04   1.35886e-01
   4  sA           4.94449e-01   5.53829e-03   3.75892e-04   6.79408e-01
   5  sB           5.11342e-01   9.60829e-03   6.23599e-04   7.50335e-02
 EXTERNAL ERROR MATRIX.    NDIM=  25    NPAR=  5    ERR DEF=1
  2.805e-05  1.022e-05  2.224e-05  8.999e-06 -1.721e-05
  1.022e-05  5.046e-05  3.205e-05  1.395e-05 -2.396e-05
  2.224e-05  3.205e-05  1.594e-04  2.734e-05 -5.836e-05
  8.999e-06  1.395e-05  2.734e-05  3.067e-05 -1.954e-05
 -1.721e-05 -2.396e-05 -5.836e-05 -1.954e-05  9.232e-05
 PARAMETER  CORRELATION COEFFICIENTS
       NO.  GLOBAL      1      2      3      4      5
        1  0.42795   1.000  0.272  0.333  0.307 -0.338
        2  0.46348   0.272  1.000  0.357  0.354 -0.351
```

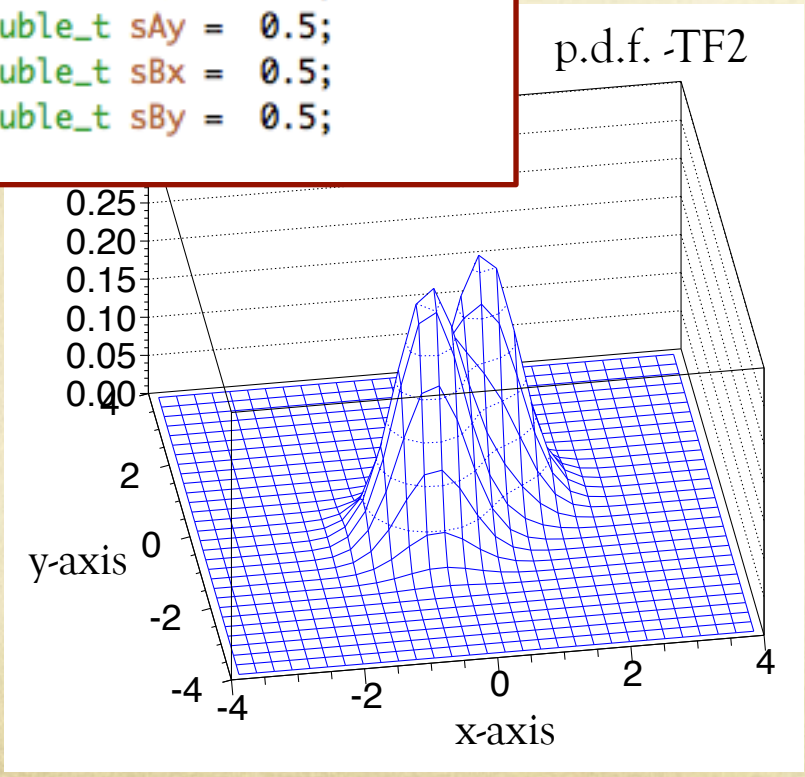
macro_like

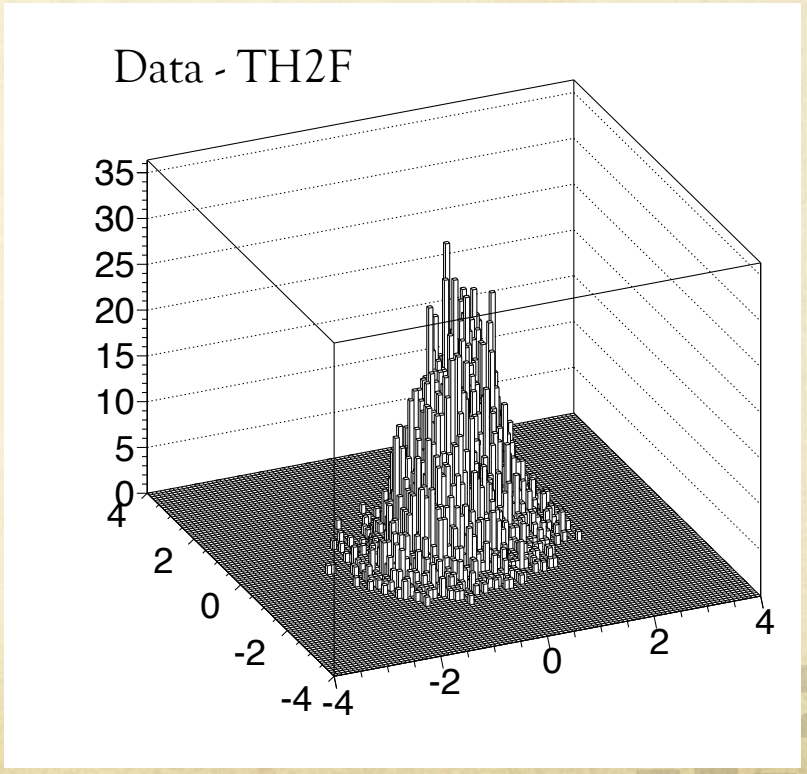
```
Terminal — root.exe — 86×48
spider-2:macro_like morello$ root -l
root [0] .L fitterLike3.C++
Info in <TUnixSystem::ACLiC>: creating shared library /Users/morello/my_mac/didattica/
corsi/HEP_SNS_I/for_students/root_template/macro_like/./fitterLike3_C.so
root [1] fitterLike3()
Integral(fun_tot) = 1
Integral(fun_A) = 1
Integral(fun_B) = 1
Entries histo_tot 10000
Entries histo_A 5000
Entries histo_B 5000
 PARAMETER DEFINITIONS:
    NO.   NAME         VALUE      STEP SIZE      LIMITS
     1 fA           5.00000e-01  1.00000e-04    0.00000e+00  1.00000e+00
     2 mAx         -5.00000e-01  1.00000e-04     no limits
     3 mAy         -5.00000e-01  1.00000e-04     no limits
     4 mBx          5.00000e-01  1.00000e-04     no limits
     5 mBy          5.00000e-01  1.00000e-04     no limits
     6 sAx          5.00000e-01  1.00000e-04     no limits
     7 sAy          5.00000e-01  1.00000e-04     no limits
     8 sBx          5.00000e-01  1.00000e-04     no limits
     9 sBy          5.00000e-01  1.00000e-04     no limits
```

- **fitterLike3.C**
  - Genera 10000 eventi distribuiti come la somma di due gaussiane2D in (x,y)
  - Fa il fit 2D Unbinned di Maximum Likelihood (UML) degli eventi generati
  - 10 parametri liberi (1 frazione relativa + 8 parametri delle Gaussiane)
  - Fa la proiezione sull'asse x e sull'asse y (non usando TF2 di ROOT)
  - Fa il fit 1D UML della in x, dove la y è stata marginalizzata.
  - Fa il confronto del potere di separazione statistico tra il fit 1D vs 2D.

fitterLike3.C

```
Double_t fA  =  0.5;
Double_t mAx = -0.5;//-1. ;
Double_t mAy = -0.5;//-1. ;
Double_t mBx =  0.5;//1. ;
Double_t mBy =  0.5;//1. ;
Double_t sAx =  0.5;//0.5
Double_t sAy =  0.5;
Double_t sBx =  0.5;
Double_t sBy =  0.5;
```

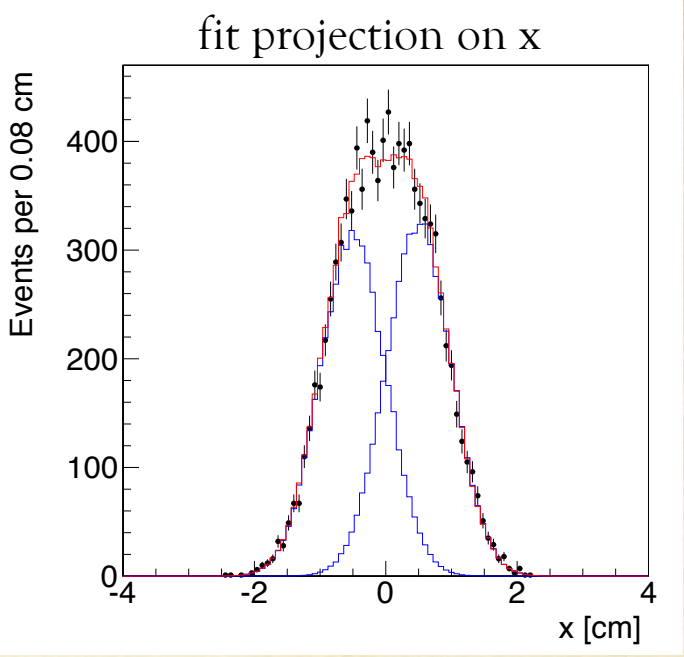


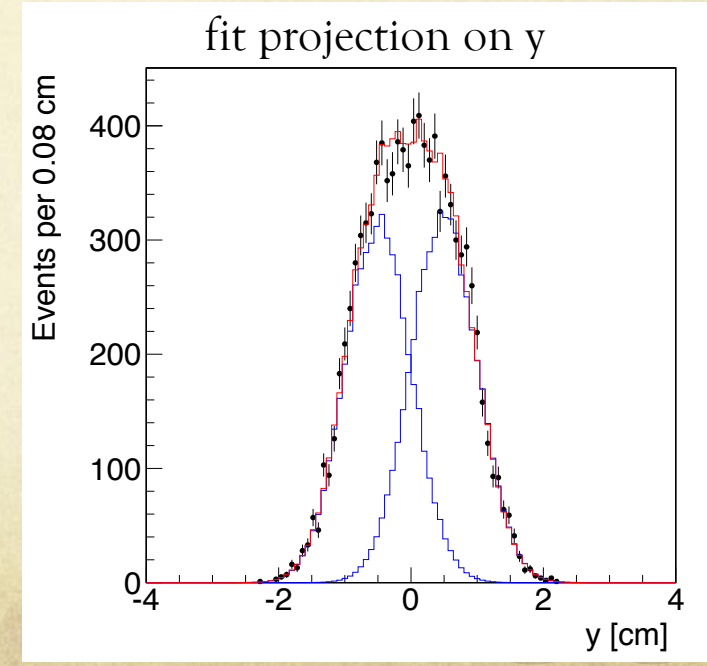


fitterLike3.C

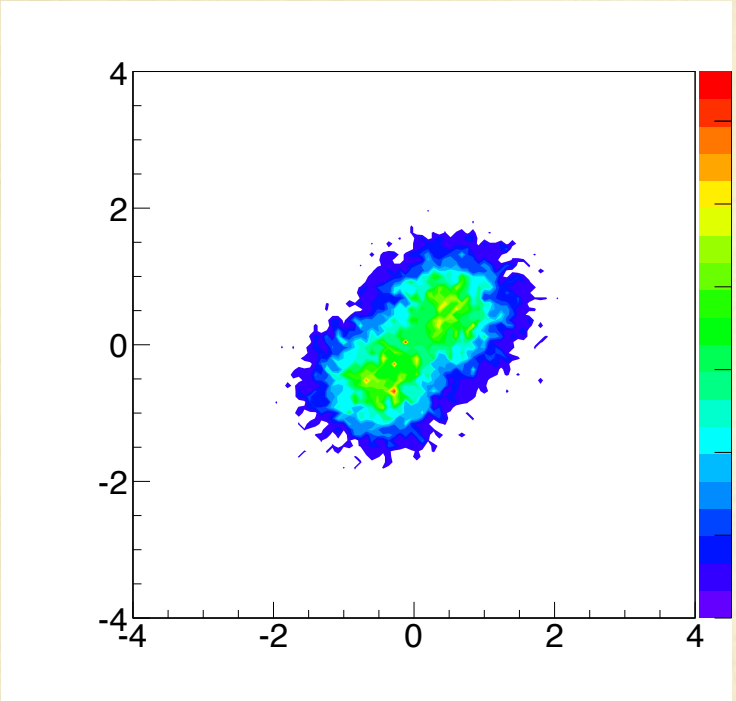
```
MIGRAD MINIMIZATION HAS CONVERGED.
MIGRAD WILL VERIFY CONVERGENCE AND ERROR MATRIX.
COVARIANCE MATRIX CALCULATED SUCCESSFULLY
FCN=39297.9 FROM MIGRAD    STATUS=CONVERGED     239 CALLS         240 TOTAL
                     EDM=1.01183e-06    STRATEGY= 1      ERROR MATRIX ACCURATE
 EXT PARAMETER                                   STEP         FIRST
 NO.   NAME      VALUE            ERROR          SIZE      DERIVATIVE
  1   fA           4.92554e-01   7.80478e-03   1.10601e-03   1.50986e-02
  2   mAx         -4.92715e-01   1.00293e-02   7.82540e-04  -5.44007e-02
  3   mAy         -5.19669e-01   9.84606e-03   7.62493e-04   2.88302e-02
  4   mBx          4.87895e-01   9.84771e-03   7.76987e-04   3.50756e-02
  5   mBy          5.01853e-01   9.91542e-03   7.63307e-04  -1.12006e-01
  6   sAx          5.08492e-01   6.83478e-03   5.78556e-04   4.28902e-02
  7   sAy          4.92110e-01   6.66621e-03   5.61754e-04   8.37214e-02
  8   sBx          5.13777e-01   6.62348e-03   5.68817e-04  -4.59180e-02
  9   sBy          5.01397e-01   6.74006e-03   5.63942e-04  -1.09489e-01
EXTERNAL ERROR MATRIX.    NDIM=  25    NPAR=  9    ERR DEF=1
```

```
PARAMETER  CORRELATION COEFFICIENTS
       NO.  GLOBAL      1      2      3      4      5      6      7      8      9
        1  0.68142  1.000  0.458  0.472  0.450  0.479  0.286  0.300 -0.273 -0.306
        2  0.59170  0.458  1.000  0.289  0.229  0.447  0.398  0.119 -0.042 -0.355
        3  0.59996  0.472  0.289  1.000  0.437  0.273  0.117  0.410 -0.325 -0.073
        4  0.57928  0.450  0.229  0.437  1.000  0.292  0.045  0.341 -0.375 -0.119
        5  0.60625  0.479  0.447  0.273  0.292  1.000  0.340  0.080 -0.119 -0.414
        6  0.48467  0.286  0.398  0.117  0.045  0.340  1.000 -0.009  0.064 -0.327
        7  0.49208  0.300  0.119  0.410  0.341  0.080 -0.009  1.000 -0.304  0.057
        8  0.46134 -0.273 -0.042 -0.325 -0.375 -0.119  0.064 -0.304  1.000 -0.009
        9  0.50287 -0.306 -0.355 -0.073 -0.119 -0.414 -0.327  0.057 -0.009  1.000
```

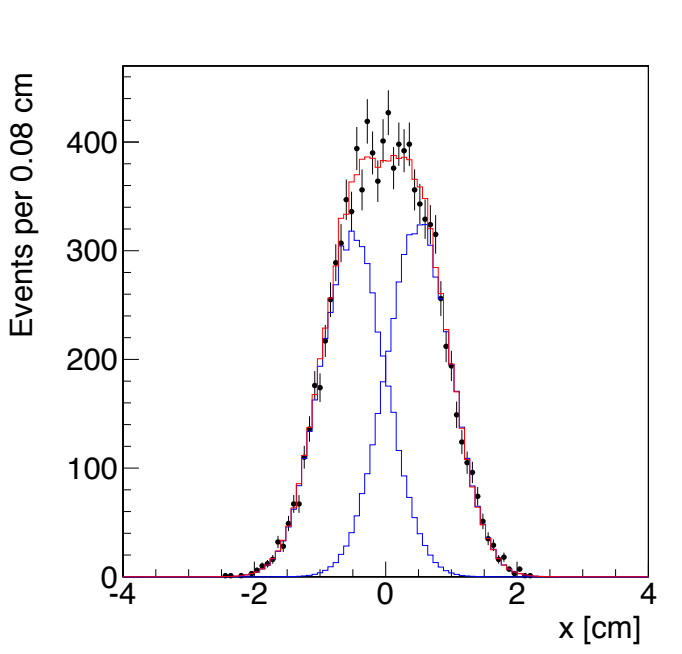




fitterLike3.C



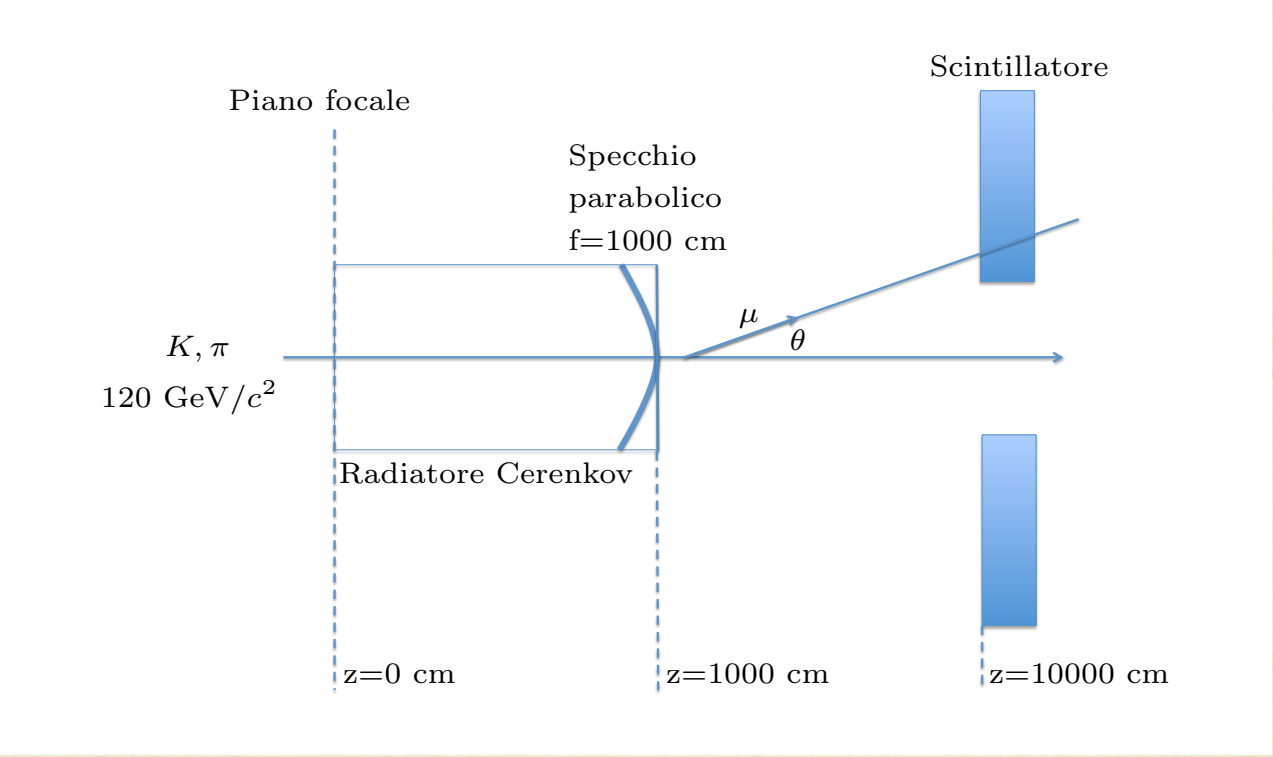
vs




```
5  0.95218  -0.916 -0.871 -0.
 ######## SEPARATION POWER ########
sigma_best = 0.005
sigma_fit2D = 0.00780478
sigma_best/sigma_fit2D = 0.640633
sigma_fit1D = 0.0629794
sigma_best/sigma_fit1D = 0.079391
```

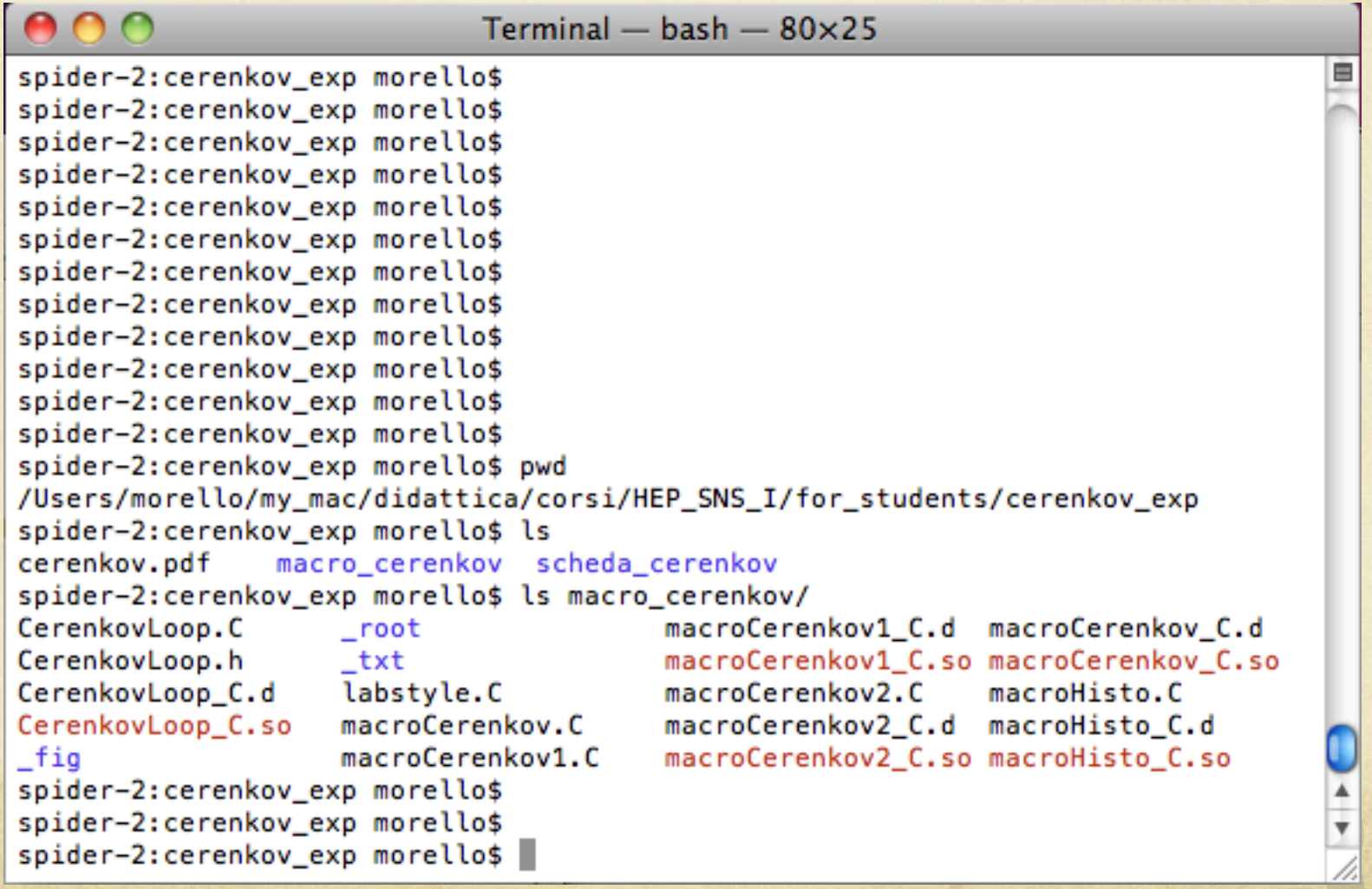
See G. Punzi, "Notes on statistical separation of classes of events" https://arxiv.org/abs/physics/0611219

cerenkov_exp



Nota: Questa esperienza di analisi dati è stata ideata dal Dr Sergio Giudici per il Corso di Laboratorio di Interazioni Fondamentali del Prof. Marco Sozzi del Corso di Laurea Magistrale in Fisica dell'Università di Pisa nell'anno accademico 2013/14, che si trova in http://www.df.unipi.it/~giudici/analisi_dati.html. Mentre nella versiona originale il fit al cerchio, per stimare i parametri R, x_0, y_0, veniva effetuato analiticamente con il metodo dei minimi quadrati, in questo caso si vuole utilizzare il software MINUIT nel framework ROOT per minimizzare l'estimatore.

cerenkov_exp



Cerenkov_exp

- macroCerenkov.C
  - legge i dati dal file "_txt/cerenkov.dat" e li scrive nel file "_txt/mydata.txt" con un differente formato.
  - Passaggio non essenziale, ma ho preferito usare istruzioni C++ basiche ed evitare l'utilizzo di vettori dinamici.
- macroCerenkov1.C
  - Legge i dati da "_txt/mydata.txt";
  - per ogni evento crea il TGraph del cerchio nella matrice dei pixel;
  - Per ogni evento fa un fit di "minimi quadrati" e stima I parametri del cerchio x0,y0 e R;
  - Riempie degli istogrammi con i valori di x0,y0 e R ed altre variabili utili.

Cerenkov_exp

- macroCerenkov2.C
  - Fa la stessa cosa di macroCerenkov1.C
  - In aggiunta crea un Tree con le variabili:

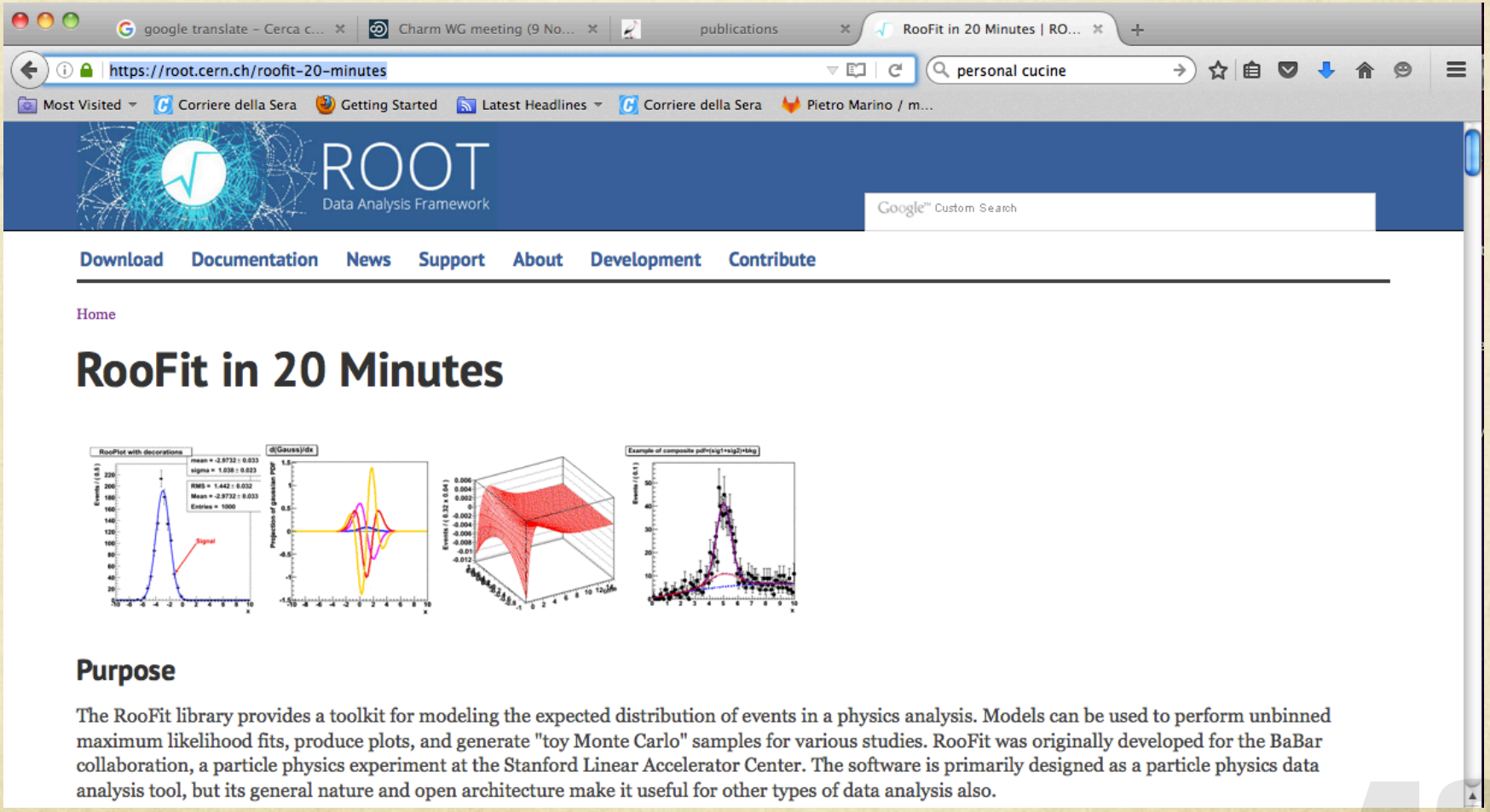
```
//create the file, the Tree and a few branches with
//a subset of Circle
TFile tree_file("./_root/tree_circle.root","recreate");
TTree t2("t2","a Tree with data");
Circle_t ring;
t2.Branch("n_track",&ring.n_track,"n_track/I");
t2.Branch("n_pixel",&ring.n_pixel,"n_pixel/I");
t2.Branch("x_pixel",ring.x_pixel,"x_pixel[n_pixel]/I");
t2.Branch("y_pixel",ring.y_pixel,"y_pixel[n_pixel]/I");
t2.Branch("R",&ring.R,"R/D");
t2.Branch("x0",&ring.x0,"x0/D");
t2.Branch("y0",&ring.y0,"y0/D");
t2.Branch("FCN",&ring.FCN,"FCN/D");
```

  - Salva il Tree nel file "./_root/tree_circle.root"

Cerenkov_exp

- CerenkovLoop.h e CerenkovLoop.C
  - Classe creata con TTree::MakeClass("CerenkovLoop").
  - Legge i dati dal Tree nel file "./_root/tree_circle.root"
  - Fa il loop su tutti gli eventi
  - Riempie gli istogrammi utili all'analisi dei dati
  - Salva tutti gli istrogrammi nel file "./_root/histo_cerenkovLoop.root"
- macroHisto.C
  - Carica gli istogrammi dal file "./_root/histo_cerenkovLoop.root"
  - Fa il fit della distribuzione del raggio del cerchio
  - Fa vari istrogrammi utili all'analisi e li manipola.
  - Stima il potere di separazione.

RooFit



https://root.cern.ch/roofit-20-minutes

Referenze di Statistica

- Glen Cowan, *Statistical Data Analysis*
  - http://ukcatalogue.oup.com/product/9780198501558.do
- Frederick James, *Statistical Methods in Experimental Physics*
  - http://www.worldscientific.com/worldscibooks/10.1142/6096
- T. Del Prete, *Methods of Statistical Data Analysis in High Energy Physics*
  - http://www.pi.infn.it/atlas/documenti/note/statistica.ps.gz

LaTex

- LaTeX is a document preparation system for high-quality typesetting. It is most often used for medium-to-large technical or scientific documents but it can be used for almost any form of publishing.
- LaTeX is not a word processor! Instead, LaTeX encourages authors not to worry too much about the appearance of their documents but to concentrate on getting the right content.
- http://www.latex-project.org/
- http://www.ctan.org/

Version 1.0
13 novembre 2014

Template di un report scientifico

R. Baggio[a], F. Baresi[b,c], A. Del Piero [a], C. Ferrara[b,c], P. Maldini[b]

[a] *Università di Pisa, Pisa*
[b] *Istituto Nazionale di Fisica Nucleare, Sezione di Pisa*
[c] *Scuola Normale Superiore, Pisa*

Sommario

Questo template ha lo scopo di mostrare quali sono le istruzioni ed in generale il framework in ambiente LaTeX per scrivere un report scientifico, come per esempio una relazione di un'esperienza del corso di Laboratorio di Fisica delle Interazioni Fondamentali.

1 Introduzione

L'idea base racchiusa in questo template è quella di fornire allo studente un semplice framework in ambiente LaTeX da poter utilizzare per scrivere le relazioni dell'esperienze di laboratorio. Vengono infatti mostrati, all'interno del file "article.tex", nelle seguenti sezioni dei semplici esempi sull'uso degli strumenti tipografici comuni di base per scrivere un report scientifico: equazioni matematiche, tabelle, referenze, ecc... Si vuole dare allo studente uno strumento agile e veloce da cui inizialmente partire per scrivere un report scientifico.

Vale la pena precisare che l'utilizzo di tale template non è assolutamente obbligatorio. Qualsiasi altro strumento tipografico può essere liberamente utilizzato.

2 Compilazione e creazione del file .pdf

Per prima cosa è necessario fare il download del file `template.tgz` all'interno di una directory personale ed eseguire il seguente comando

```
mbmorello2:test morello$
mbmorello2:test morello$ pwd
/Users/morello/my_mac/didattica/corsi/LabIntFondamentali/lezioni_Nov2014/latex_template/test
mbmorello2:test morello$ ls
article_template.tgz
mbmorello2:test morello$ tar -xvzf article_template.tgz
article_template/
article_template/article.pdf
article_template/article.tex
article_template/fig/
article_template/fig/canvas_fit_AL.eps
article_template/fig/canvas_fit_L.eps
mbmorello2:test morello$
mbmorello2:test morello$ cd article_template
mbmorello2:article_template morello$ ls
article.pdf article.tex fig
```

Per compilare il file è sufficiente entrare dentro la directory `article_template` e digitare i seguenti comandi, nel caso le figure incluse nel testo siano in formato Encapsulated PostScript (.eps):

```
mbmorello2:article_template morello$ ls
article.pdf article.tex fig
mbmorello2:article_template morello$ latex article.tex ; dvipdf article.dvi
```

In caso le figure siano inlcuse in formato Portable Document Format (.pdf) è sufficiente digitare:

```
mbmorello2:article_template morello$ ls
article.pdf article.tex fig
mbmorello2:article_template morello$ pdflatex article.tex
```

Come risultato verrano prodotti alcuni file nella directory `template`, tra cui il file .pdf.

```
bmorello2:article_template morello$ ls
article.aux article.dvi article.log article.pdf article.tex fig
mbmorello2:article_template morello$
```