

Esercitazione ROOT

Matteo Duranti

matteo.duranti@infn.it

Esercitazione

- Riempire un istogramma (*TH1F*) con 100 bin, fra 0 e 10, con 5000 eventi random (*) presi da
 - a) un'Uniforme
 - b) una Gaussiana con $\mu=5$ e $\sigma=1$
 - c) un'Esponenziale con $\tau=1/3$
- Ripetere b) 1000 volte (chiamiamo ciascuna: “prova”) (** attenti al *seed!*)
- Contare gli eventi in $[-\infty, \mu-3\sigma]$ (non $[0, \mu-3\sigma]$: c'è differenza?) e fare:
 - a) un istogramma con il numero ottenuto nelle 1000 prove
 - b) un grafico (*TGraph*) con il numero ottenuto vs il numero sequenziale della prova
- Fittare (***) b) con una “*pol1*” (polinomio di grado 1, i.e. una retta)
- Fittare (***) a) con:
 - una Gaussiana
 - una Landau
 - una Poissoniana (**** non è banale: vedi slide successiva)

* <https://root.cern.ch/doc/master/classTRandom.html>

** ogni volta dovete creare un nuovo oggetto *TRandom* o utilizzare *TRandom::SetSeed(<int>)* per far partire la sequenza da un diverso punto (detto *seed*)

*** <https://root.cern.ch/root/html/doc/guides/users-guide/FittingHistograms.html>

***Fit Poissoniano

- il fit con Poissoniana “a mano” (scrivete voi la *formula*) non funziona:
 - potete utilizzare `TMath::Poisson()` dentro la formula analitica della TF1.
E' una versione “generalizzata” della poissoniana in cui si usa la Gamma di Eulero invece che il Fattoriale (`TMath::Factorial()`) che avrebbe senso solo per valori “interi” della x ;
 - se volete invece provare a scrivere quella “discreta” (sia che la scriviate a mano, sia che utilizzate `TMath::PoissonI()` [c'è una 'I' alla fine: è la vera poissoniana con il fattoriale]) l'unico modo è “castare” (<http://www.cplusplus.com/doc/tutorial/typecasting>, basta che leggete il paragrafetto “Type Casting”) x a intero, i.e. $\text{int}(x)$
- ricordatevi che la poissoniana è una *p.d.f. (probability density function)* cioè ha integrale 1, dove l'integrale è fatto come vi hanno sempre insegnato ad analisi: “somma” dei valori y per la larghezza, in x dei termini che sommate.
Un istogramma, invece, ha, di default, come integrale (`histo->Integral()`) la somma dei “bin content”, fregandosene della larghezza del bin. Quindi o fate uno `Scale()` dell'istogramma per farlo diventare una p.d.f., cioè lo scalate per l'Integrale “corretto” (che tiene conto anche della larghezza del bin):
$$\text{histo->Scale}(1.0/\text{histo->Integral}(\text{“w”}))$$
(dove l'opzione “w” significa “width”)
oppure quando fittate usate DUE parametri liberi: il “lambda” (o “nu”, chiamatelo come vi pare) della Poissoniana + una normalizzazione davanti a tutto (che, se il fit funziona deve venire collegata all'integrale delle istogramma (`histo->Integral()`) e alla larghezza dei bin
- quando fittate “iniziare” con dei valori dei parametri già vicini a quello che vi aspettate, in generale, vi fa riuscire il fit più facilmente:
$$\text{tf1difit->SetParameter}(\text{<numero parametro>}, \text{<valore iniziale>})$$
- se visualizzate la TF1 con un alto numero di punti (`tf1difit->SetNpx(10000)`) vi risulta graficamente meno spigolosa