

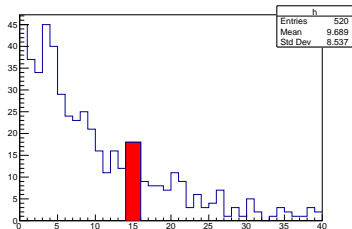
Introduzione Hands-on test d'ipotesi

Laboratorio di Metodi Computazionali e Statistici (2022/2023)

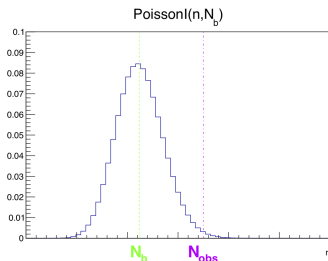
F. Parodi, R. Cardinale

December 19, 2022

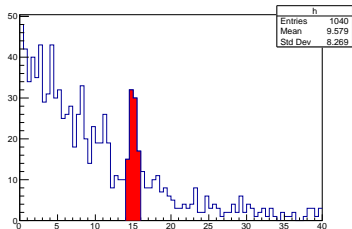
- Un esperimento ha preso dati di una certa variabile x
- Supponiamo che l'esperimento abbia raccolto prima un certo set di dati a bassa statistica (`dati_lowstat.dat`) e poi abbia successivamente raccolto ulteriori dati ad alta statistica (`dati_highstat.dat`)
- Fondo (generato da processi fisicamente noti) con distribuzione esponenziale
- Intervallo di dati interessanti (in cui potrebbe trovarsi del segnale interessante generato da processi mai osservati) individuato da considerazioni/motivazioni teoriche/sperimentali: [14, 16]
- Vogliamo capire se i dati osservati in quell'intervallo sono compatibili con gli eventi di fondo atteso oppure se si discostano da quanto ci aspettiamo



- Si analizzi il file a bassa statistica (dati_lowstat.dat)
- Supponiamo che il numero di eventi attesi (da eventi di fondo noti) sia $N_b = 22.35$
- Si calcoli il numero di eventi osservati nell'intervallo di interesse $[14, 16]$: N_{obs}
- Si calcoli la probabilità che il fondo atteso ($N_b = 22.35$) possa fluttuare fino a raggiungere un valore maggiore o uguale rispetto al numero totale di eventi osservato N_{obs} nell'intervallo $[14, 16]$

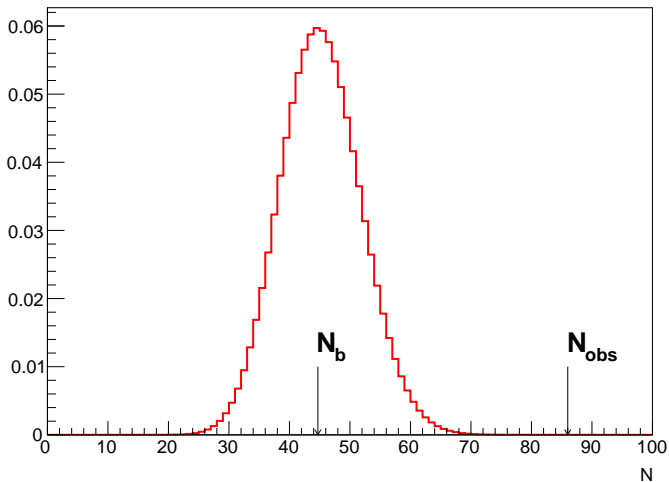


- I dati nell'intervallo sono distribuiti nell'ipotesi H_0 (che prevede solo fondo esponenziale) secondo una Poissoniana centrata su N_b
- Se contate N_{obs} otterrete un numero che è maggiore di N_b (si vede anche dal grafico)
- Dal numero di eventi attesi N_b e dal numero di eventi osservati N_{obs} posso escludere l'ipotesi H_0 ?
- Dovrò calcolare il p-value della Poissoniana: $P(N \geq N_{obs} | N_b)$
- Confrontare il p-value ottenuto con un certo valore di significanza del test (α)

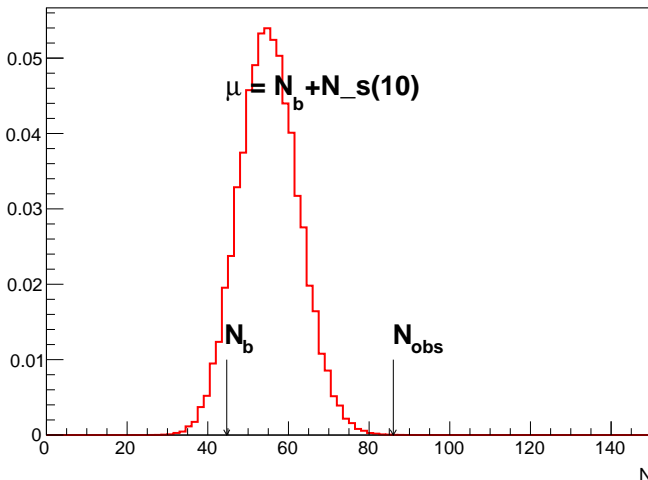


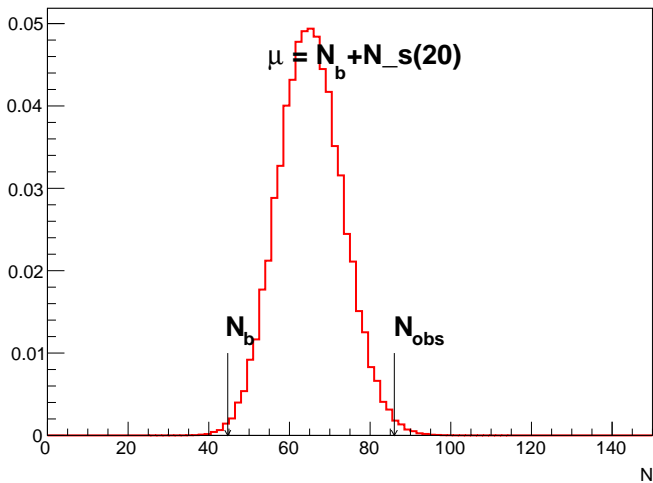
- Ora dovete analizzare il file ad alta statistica: si evidenzia un nuovo processo
- Dovete calcolare il limite superiore (al 95% C.L.) al numero di eventi di segnale (N_s) usando il numero di eventi totali osservati ($N_b + N_s$) nell'intervallo $[14,16]$ e assumendo che il numero atteso di eventi di fondo (N_b) siano 44.7

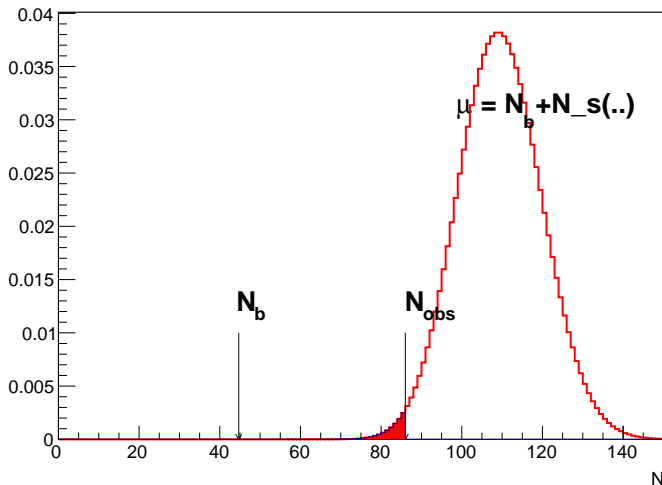
$$N_s = 0$$



- Per calcolare il limite superiore al 95% di CL devo provare ad inserire un numero di eventi di segnale N_s variabile e vedere quale è il valore di N_s massimo che posso avere senza rigettare l'ipotesi $N_s + N_b$ visto il numero di eventi osservati N_{obs}
- Cioè vogliamo testare $N_s \neq 0$ e vedere se possiamo escluderli perchè predicono alti valori di $N_s + N_b$ e quindi avremmo una probabilità bassa di osservare un numero di eventi uguale o più basso di quelli osservati N_{obs}







- $p\text{-value} = P(\mu \leq N_{obs}, N_s, N_b)$
- Cioè la probabilità di osservare un numero uguale o minore di eventi di quelli osservati N_{obs}

Compito a casa: fit della distribuzione

- Eseguire un fit (binned e unbinned) dei dati assumendo come pdf la distribuzione somma di quella di segnale e di quella di fondo
- Il fondo può essere descritto da una funzione esponenziale
- Il segnale può essere descritto da una gaussiana
- È utile definire una funzione di fit che possa essere usata sia per il fit extended che non-extended

$$f(x) = N\Delta w(\alpha \text{Gaus}(x, x_0, \sigma) + (1 - \alpha)\frac{e^{-x/\tau}}{\tau})$$

- Nel caso dell'extended binned likelihood si fissi Δw alla larghezza del bin dellistogramma e N si inizializza al numero totale di eventi
- Nel caso del fit unbinned/binned non extended entrambi i valori vengono fissati a 1 (in modo da ottenere una densità di probabilità)

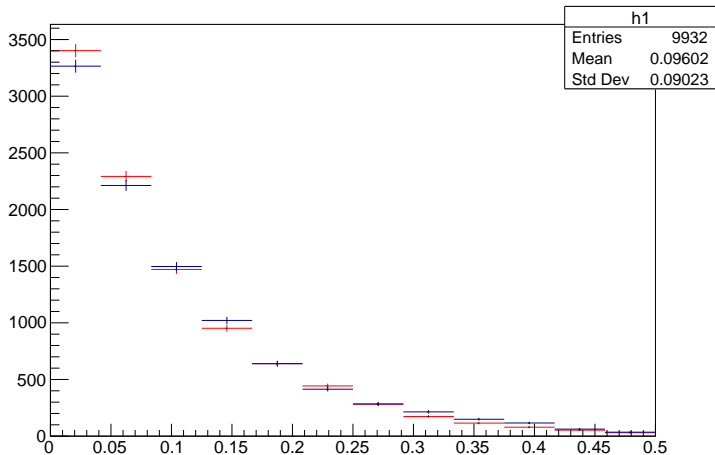
Riassunto metodi ROOT per fit di likelihood

- Utilizzare i metodi di ROOT per il fit alla distribuzione della variabile x

	Binned	Unbinned
Likelihood	Fit + opzione MULTI	TTree + UnbinnedFit
Extended Likelihood	Fit + opzione L	

Test di bontà del fit

- Abbiamo due campioni di dati s1.dat e s2.dat



Test di bontà del fit

- I due campioni di dati seguono la stessa distribuzione?
- Il campione di dati s1.dat segue un'esponenziale (`scipy.stats.expon` con `scale=1/λ`) con $\lambda = 10$?
- Il campione di dati s1.dat segue un'esponenziale?

```
Double_t TH1::Chi2Test ( const TH1 * h2,
                        Option_t * option = "UU",
                        Double_t * res = 0
                        ) const
```

virtual

χ^2 test for comparing weighted and unweighted histograms

Function: Returns p-value. Other return values are specified by the 3rd parameter

Parameters

[in] **h2** the second histogram

[in] **option**

- "UU" = experiment experiment comparison (unweighted-unweighted)
- "UW" = experiment MC comparison (unweighted-weighted). Note that the first histogram should be unweighted
- "WW" = MC MC comparison (weighted-weighted)
- "NORM" = to be used when one or both of the histograms is scaled but the histogram originally was unweighted
- by default underflows and overflows are not included:
 - "OF" = overflows included
 - "UF" = underflows included
- "P" = print chi2, ndf, p_value, igood
- "CHI2" = returns chi2 instead of p-value
- "CHI2/NDF" = returns χ^2/ndf

[in] **res** not empty - computes normalized residuals and returns them in this array

scipy.stats.ks_2samp

scipy.stats.ks_2samp(data1, data2)

[source]

(<http://github.com/scipy/scipy/blob/v0.15.1/scipy/stats/stats.py#L3966>)

Computes the Kolmogorov-Smirnov statistic on 2 samples.

This is a two-sided test for the null hypothesis that 2 independent samples are drawn from the same continuous distribution.

Parameters: **a, b** : *sequence of 1-D ndarrays*

two arrays of sample observations assumed to be drawn from a continuous distribution, sample sizes can be different

Returns: **D** : *float*

KS statistic

p-value : *float*

two-tailed p-value

scipy.stats.kstest

scipy.stats.kstest(*rvs*, *cdf*, *args=()*, *N=20*, *alternative='two-sided'*, *mode='approx'*)
(<http://github.com/scipy/scipy/blob/v0.14.0/scipy/stats/stats.py#L3307>)

[source]

Perform the Kolmogorov-Smirnov test for goodness of fit.

This performs a test of the distribution $G(x)$ of an observed random variable against a given distribution $F(x)$. Under the null hypothesis the two distributions are identical, $G(x)=F(x)$. The alternative hypothesis can be either 'two-sided' (default), 'less' or 'greater'. The KS test is only valid for continuous distributions.

Parameters: *rvs* : *str*, *array* or *callable*

If a string, it should be the name of a distribution in `scipy.stats` (`./stats.html#module-scipy.stats`). If an array, it should be a 1-D array of observations of random variables. If a callable, it should be a function to generate random variables; it is required to have a keyword argument *size*.

cdf : *str* or *callable*

If a string, it should be the name of a distribution in `scipy.stats` (`./stats.html#module-scipy.stats`). If *rvs* is a string then *cdf* can be `False` or the same as *rvs*. If a callable, that callable is used to calculate the *cdf*.

args : *tuple*, *sequence*, *optional*

Distribution parameters, used if *rvs* or *cdf* are strings.

N : *int*, *optional*

Sample size if *rvs* is string or callable. Default is 20.

alternative : *{'two-sided', 'less', 'greater'}*, *optional*

Defines the alternative hypothesis (see explanation above). Default is 'two-sided'.

mode : *'approx'* (default) or *'asympt'*, *optional*

Defines the distribution used for calculating the p-value.

- 'approx' : use approximation to exact distribution of test statistic
- 'asympt' : use asymptotic distribution of test statistic

Returns:

D : *float*

KS test statistic, either D , $D+$ or $D-$.

p-value : *float*

One-tailed or two-tailed p-value.