Breve introduzione a BAT

14 aprile 2011

Mia mail in caso di domande: Massimo.Corradi@bo.infn.it

Homepage di BAT:
http://www.mppmu.mpg.de/bat/?page=home

Trovate 2 articoli utili nella pagina della documentazione:

“Short introduction to BAT” , istruzioni sul programma: user guide, molto utile

“BAT - The Bayesian Analysis Toolkit” == articolo piu' generale

Cosa e' BAT ?

Un set di librerie C++ integrate con Root per risolvere problemi di inferenza Bayesiana

Teorema di Bayes
$f(\bar{x}|data) = 1/N \;\; P(data|\,\bar{x}) \;\; f0(\bar{x})$

Usa un Markov Chain Monte Carlo (MCMC) basato sull'algoritmo Metropolis
Per campionare la “posterior” $f(\bar{x}|data)$.

L'utente deve fornire un modello (un oggetto C++) che implementa due metodi:

```
double LogAPrioriProbability(std::vector <double> parameters);
double LogLikelihood(std::vector <double> parameters);
```

Poi BAT si occupa del resto,
Ma conviene guardare un esempio specifico ...

Esempio 1: Misura di conteggi con background

1) Si misura per un tempo T il numero di conteggi N provenienti da una sorgente e dal fondo

2) Si misura per un tempo T_{BKG} il numero di conteggi N_{BKG} provenienti dal fondo solamente

-> trovare l'intensita' della sorgente, R_s, e del fondo, R_b.

Lo avete gia' visto usando JAGs

E' il tutorial numero 1 di BAT:
http://www.mppmu.mpg.de/bat/?page=tutorials&name=counting_experiment

In breve:
f(R_s,R_b) = C * P(data|R_s,R_b) f_0(R_s) f_0(R_b)

P(data|R_s,R_b)=P(N,R_s,R_b)P(N_b|R_b)

P(N|R_s, R_b) = Poisson (N, λ=T*(R_s+R_b))
P(N_b|R_b) = Poisson (N_{BKG}, λ=T_{BKG}*R_b)

Esempio 2: Fit Lineare di misure con incertezza gaussiana costante.

Ho un set di misure di y_i^M a diversi valori di x_i.

Modello: le misure di y sono distribuite secondo una gaussiana con
Media $y=f(x)=ax+b$ e sigma=costante=σ :

-> cerchiamo di fare una inferenza su a,b e anche di interpolare/estrapolare il risultato:
cioe' trovare Il valore (la pdf) di y per un dato x

-> misure indipendenti: $P(data|a,b) = \Pi_i \, P(y_i^M|a,b)$

$P(y_i|a,b) = 1/(\sigma \sqrt{2\pi}) * \exp(-(y_i^M-f(x))^2 / (2*\sigma^2)) = C \exp(-(y_i^M-f(x_i))^2) = C \exp(-(y_i^M-ax_i-b)^2)$

-> il modello gaussiano e' disponibile pre-confezionato: BCGraphFitter
(vedere l'esempio in BAT-0.4.2/examples/basic/graphFitter)
Altri modelli pre-definiti per fit: binomiale, Poisson, template fitter

-> Sfruttiamo l'esempio anche per vedere come si puo' usare BAT interattivamente in Root
invece che in un programma C++ compilato.

Esempio 3: estensione dell'esempio 2 al confronto tra modelli:

-> confrontare due modelli: Fit lineare o fit quadratico

$P(\text{dati}|\text{mod}) = \int d\bar{x}\, P(\text{dati}|\bar{x})\, f0(\bar{x}) = N$

E' la costante di normalizzazione del teorema di Bayes.

Si puo' assegnare una probabilita' a priori a “ogni possibile modello” e poi calcolare le prob. a posteriori per i diversi modelli:
$P(\text{mod\_1}|\text{dati}) = P(\text{dati}|\text{mod\_1})\, p0(\text{mod\_1}) / [\, \Sigma_{\text{modelli}} P(\text{dati}|\text{mod\_i})\, p0(\text{mod\_i})\,]$

oppure piu' semplicemente guardare il
Bayes factor = P(mod_1|dati) / P(mod_2|dati)

Cose interessanti
-> Implementa automaticamente il Rasoio di Occam
-> dipendenza dal range di integrazione
-> normalizzazione delle prior

Altre cose interessanti di BAT non considerate qui

- Ottimizzazione del MCMC

- calcolo dei p-values

- modelli pre-definiti per fit (Gaussiano, binomiale, Poisson, template fitter)