



993SM - Laboratory of Computational Physics week V 25 October 2024

Maria Peressi

Università degli Studi di Trieste - Dipartimento di Fisica
Sede di Miramare (Strada Costiera 11, Trieste)

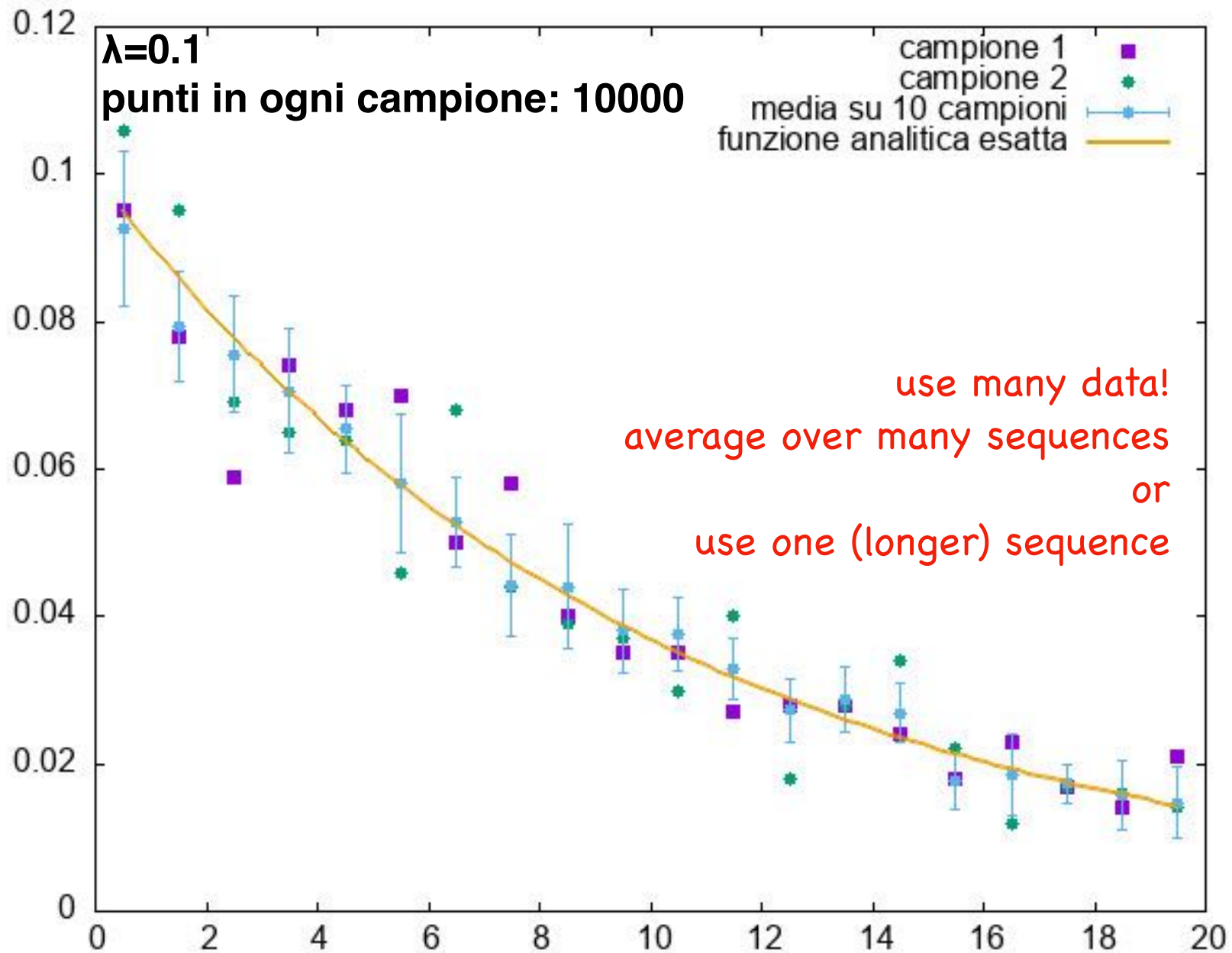
e-mail: peressi@units.it

tel.: +39 040 2240242

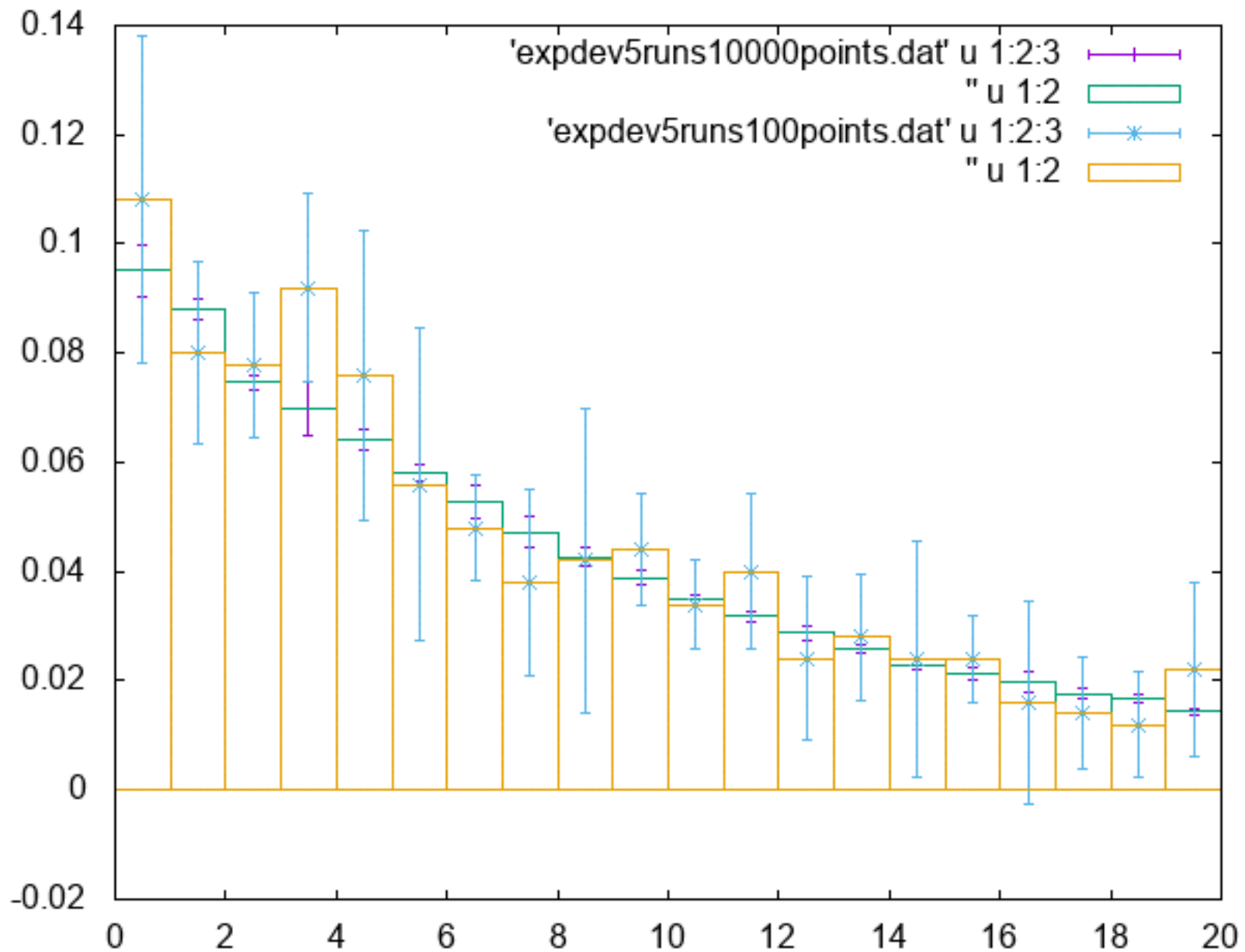
Exponential deviate

Statistical averages

Exponential deviate



Exponential deviate



Exponential deviate

```
allocate (histo(nbin, nruns))
```

```
...
```

```
do h = 1,nruns
```

```
  do i = 1,n
```

```
    ...
```

```
  end do
```

```
end do
```

```
do ibin = 1,nbin
```

```
  avrg(ibin) = sum( histo(ibin, :))
```

Radioactive decay

Statistical averages

&

time discretization

Radioactive decay

$N(t)$ Atoms present at time t

λ Probability for each atom to decay in Δt

$\Delta N(t)$ Atoms which decay between t and $t + \Delta t$

$$\Delta N(t) = -\lambda N(t) \Delta t \Rightarrow N(t) = N(t=0) e^{-\lambda t}$$

Purpose of the exercise:

- fix λ (<1 since it is a probability)
- perform the stochastic simulation
- check whether $N(t)$ has the expected behavior, also quantitatively, **calculating the resulting decay constant from the data fitting and comparing with the given value of λ**

Radioactive decay

$N(t)$ Atoms present at time t

λ Probability for each atom to decay in Δt

$\Delta N(t)$ Atoms which decay between t and $t + \Delta t$

$$\Delta N(t) = -\lambda N(t) \Delta t \Rightarrow N(t) = N(t=0) e^{-\lambda t}$$

Purpose of the exercise:

- fix λ

In our simulation: 1 iteration in the loop $\longleftrightarrow \Delta t$

(the time step is implicitly fixed in a Monte Carlo simulation!

somehow, in this exercise we decide the time discretization by fixing λ)

Radioactive decay

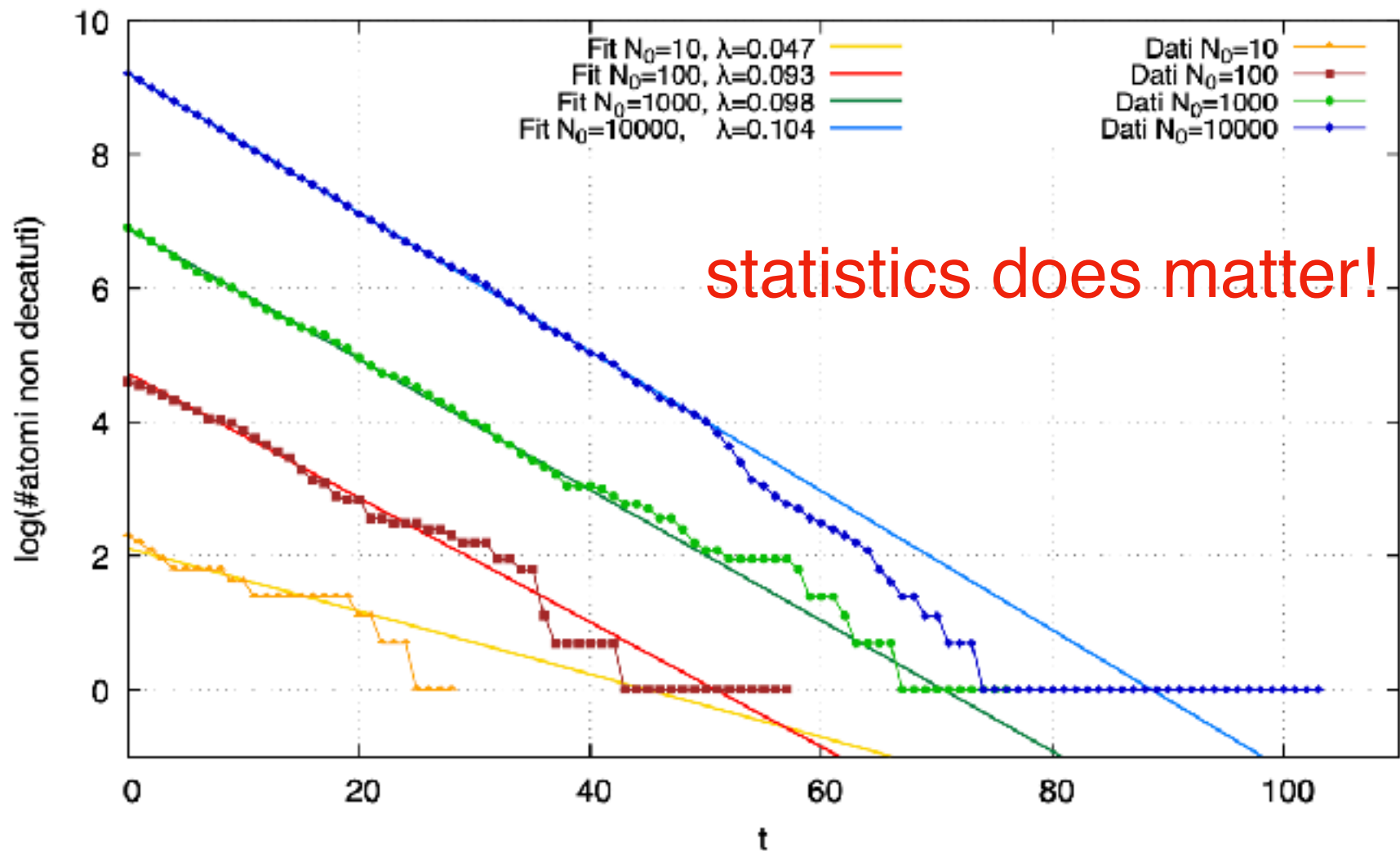


Figura 7: Atomi rimanenti in funzione del tempo, in scala semilogaritmica, con numero variabile di atomi iniziali N_0 e rispettive stime di λ , con dato iniziale $\lambda_{in} = 0.1$

Radioactive decay

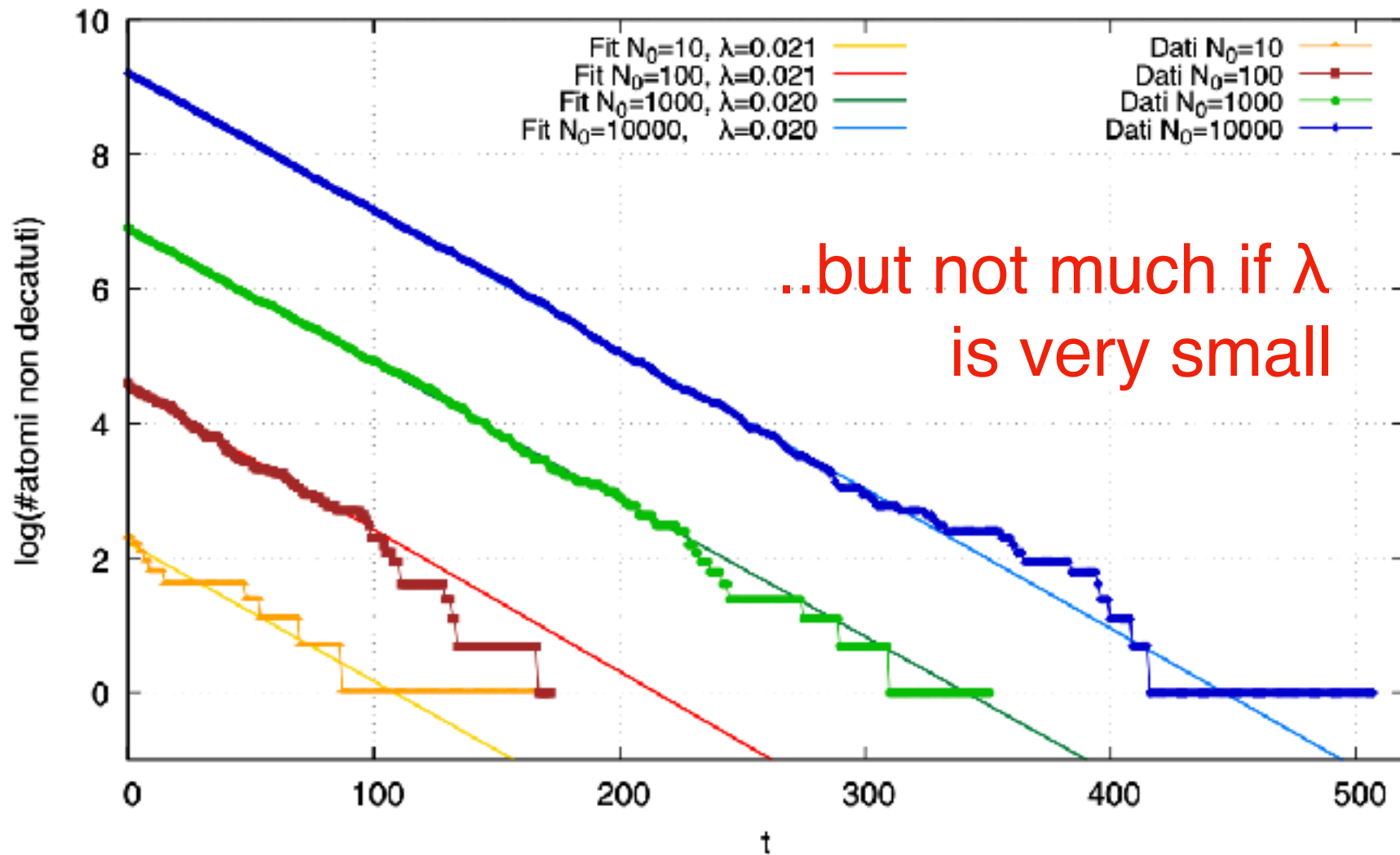
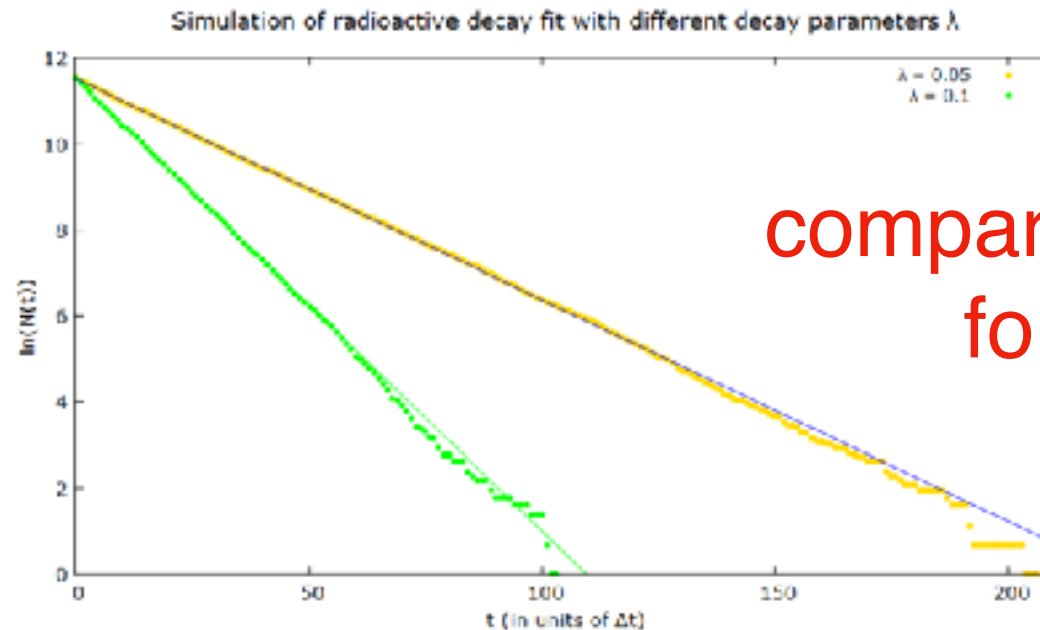


Figura 6: Atomi rimanenti in funzione del tempo, in scala semilogaritmica, con numero variabile di atomi iniziali N_0 e rispettive stime di λ , con dato iniziale $\lambda_{in} = 0.02$

Radioactive decay



comparing results
for different λ

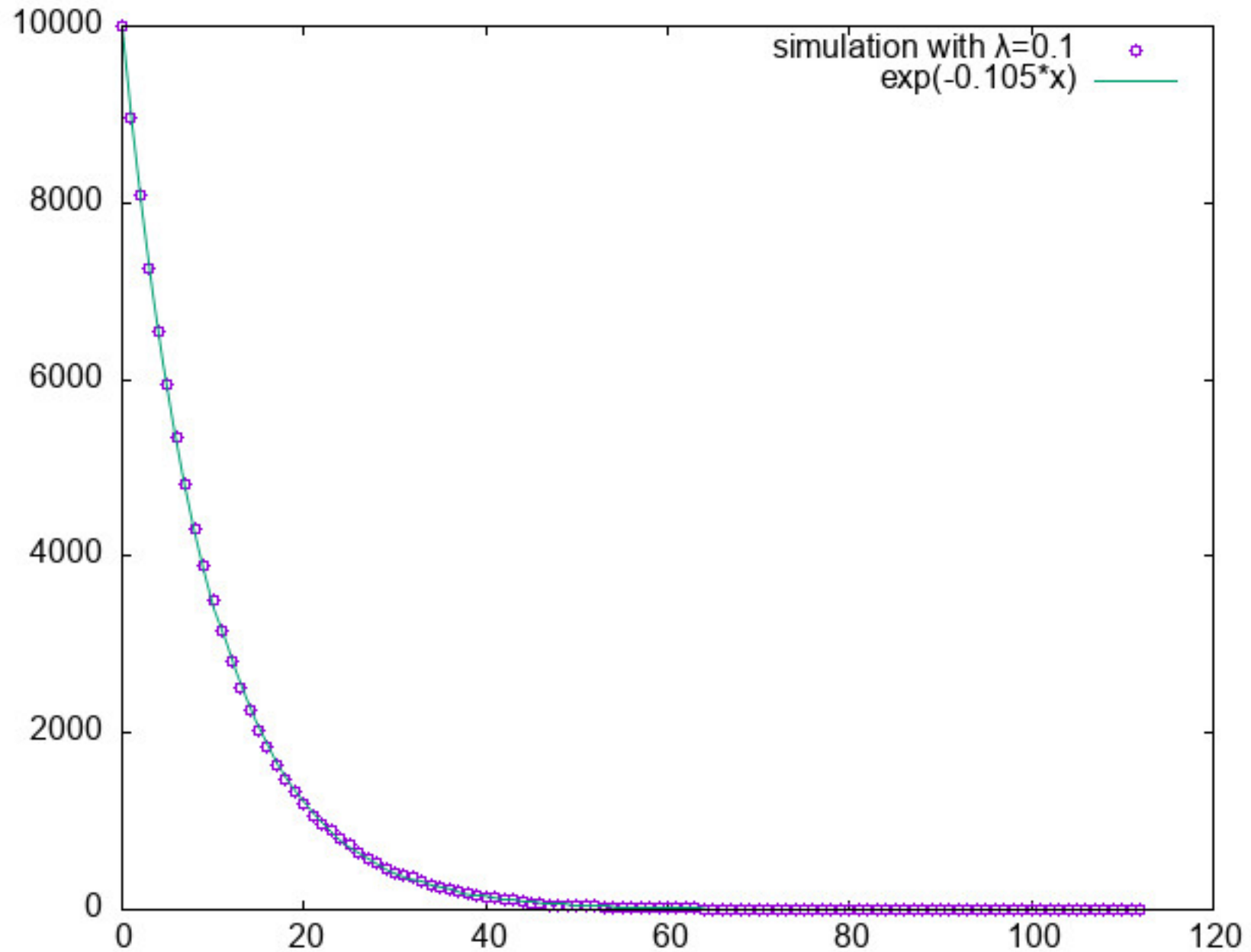
Figura 5: Fit della distribuzione di $N(t)$ in unità di Δt per diversi valori del parametro di decadimento λ .

λ aspettato ($\cdot 10^{-2}$)	λ stimato ($\cdot 10^{-2}$)	$N(0)$ stimato ($\cdot 10^3$)	σ_λ
5	5.14 ± 0.04	100 ± 2	0.8 %
10	10.5 ± 0.2	100 ± 3	1.9 %
15	16.1 ± 0.3	99 ± 2	1.9 %
30	35.8 ± 0.7	101 ± 3	2.0 %

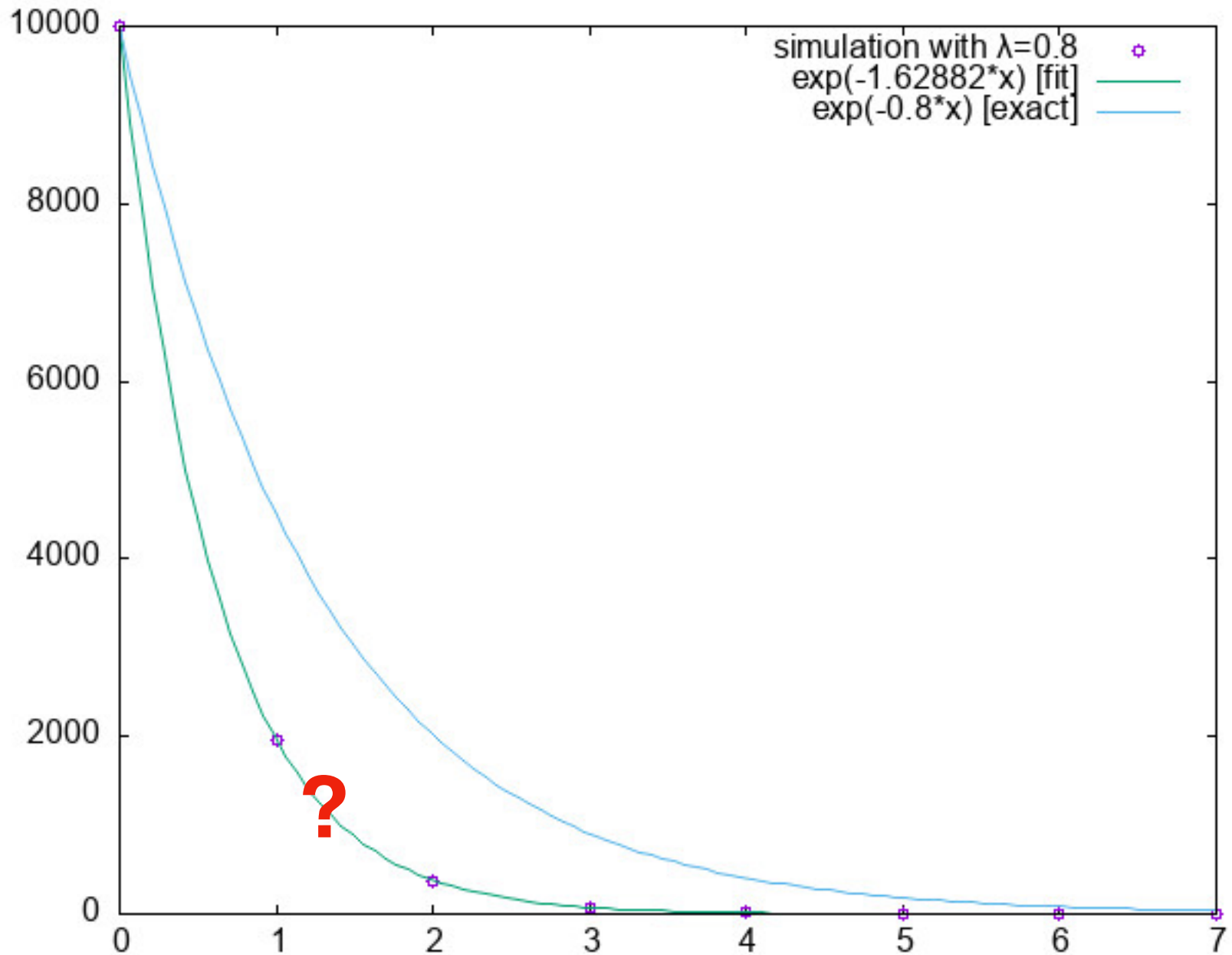
Tabella 3: Stime dei parametri λ e $N(0)$ dal fit della distribuzione di $N(t)$ in scala semilogaritmica.

(from a homework of a.y. 2022/23)

Radioactive decay



Radioactive decay



recovering and commenting the results of the exercise of past week:

Radioactive decay

$N(t)$ Atoms present at time t

λ Probability for each atom to decay in Δt

$\Delta N(t)$ Atoms which decay between t and $t + \Delta t$

$$\Delta N(t) = -\lambda N(t) \Delta t \Rightarrow N(t) = N(t=0) e^{-\lambda t}$$

Purpose of the exercise:

- fix λ

In our simulation: 1 iteration in the loop $\longleftrightarrow \Delta t$

(the time step is implicitly fixed in a Monte Carlo simulation!

somehow, in this exercise we decide the time discretization by fixing λ :

but in general in the numerical simulation of a dynamical process, the smaller is the time step, the more accurate is the simulation)

Radioactive decay

Stochastic simulations give reliable results when obtained:

- on average and for large numbers
- fine discretisation of time evolution

(in the exercise #1: change λ ; compare the value obtained from the simulation with the one inserted; does the “quality” of the results change with λ ?)

(Commentiamo insieme questa soluzione proposta per l'es. del decadimento radioattivo....)

Si noti che qui λ non corrisponde alla costante di decadimento (che avrebbe le dimensioni di un tempo^{-1}) ma piuttosto alla probabilità di decadimento in Δt , è quindi una quantità adimensionale ottenuta come

$$\lambda = (\text{costante di decadimento}) \times (\text{intervallo di tempo scelto}). \quad (2)$$

Questa scelta permette di definire automaticamente (tramite il valore scelto per λ) l'ordine di grandezza degli intervalli di tempo Δt nel programma, senza che questo venga mai esplicitamente dichiarato. Per chiarire il concetto si considera l'esempio del ^{232}U che ha una costante di decadimento di $3.192 \times 10^{-10} \text{s}^{-1}$. Da questa si possono calcolare le probabilità di decadimento relative a vari intervalli di tempo, per esempio:

- Probabilità di decadimento in *1secondo* = $3.192 \times 10^{-10} = \lambda_s$;
- Probabilità di decadimento in *1giorno* = $2.756 \times 10^{-5} = \lambda_d$;
- Probabilità di decadimento in *1anno* = $1.006 \times 10^{-2} = \lambda_y$.

A seconda del valore di λ che scelgo (ad esempio λ_s , λ_d o λ_y) il singolo Δt andrà interpretato di conseguenza (rispettivamente di 1 secondo, 1 giorno o 1 anno).

Come si vedrà più approfonditamente di seguito, una scelta appropriata per λ è fondamentale per ottenere dei buoni risultati.

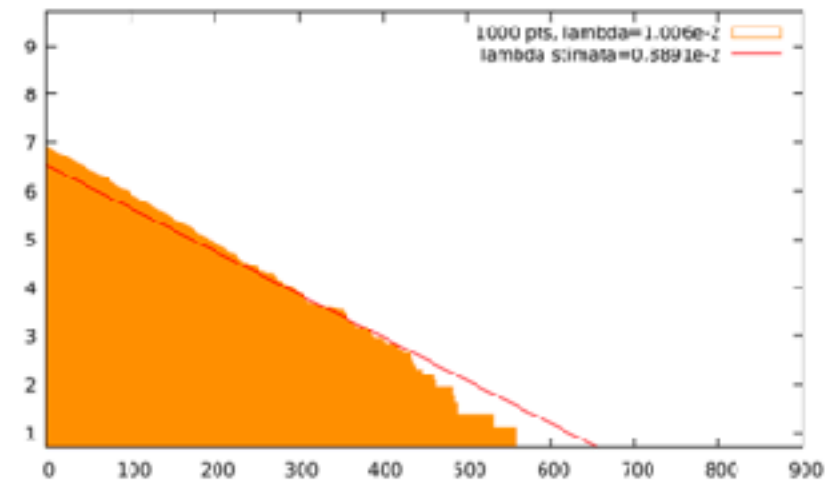
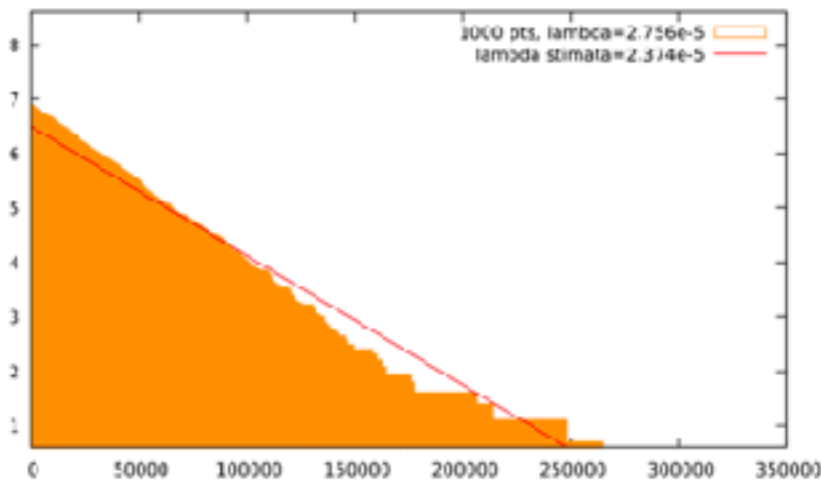
$$t_{1/2} = \frac{\ln(2)}{\lambda^{dec}} \quad (3)$$

che ha permesso un riscontro più immediato fra la simulazione numerica e casi noti. Questo ha in particolare permesso di verificare la relazione fra λ e Δt e il fatto che Δt sia effettivamente determinato dalla scelta di λ .

Riprendendo l'esempio precedente dell'uranio 232 si è infatti visto che inserendo a turno λ_s , λ_d e λ_y e interpretando rispettivamente Δt come 1 secondo, 1 giorno e 1 anno, si trovano valori di $t_{1/2}$ vicini a quello atteso (pari a 69 anni).

Si nota inoltre che la stima di $t_{1/2}$ (e di λ) migliora lavorando con i valori maggiori possibili per λ .

Di seguito sono riportati i grafici ottenuti con λ_d e λ_y . λ_s risulta troppo piccola e il programma impiega troppo tempo per la simulazione.



Prendendo infatti λ più grande possibile si riducono gli effetti legati al fatto che, essendo λ piccolo rispetto a Δt , ci sono molti intervalli in cui non succede nulla (che danno un effetto simile a quello delle code) e il programma deve eseguire un gran numero iterazioni, che lo rendono inoltre molto inefficiente.

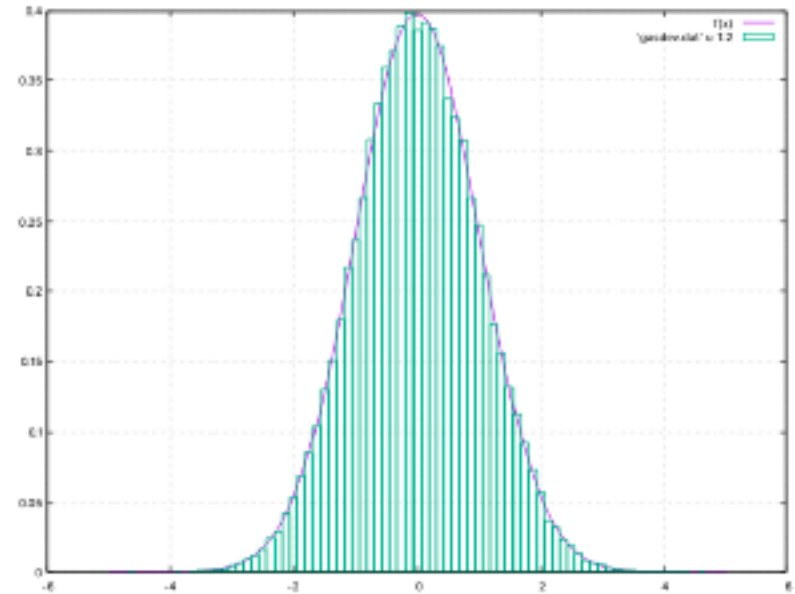
Per la definizione di λ utilizzata, scegliere λ più grandi possibili, vuol dire scegliere λ in modo che Δt sia del giusto ordine di grandezza, se confrontato con i tempi caratteristici (per esempio $t_{1/2}$) del decadimento che si vuole simulare.

Per l'uranio 232, dunque, per cui $t_{1/2} = 69 \text{anni}$, non sarà conveniente utilizzare λ_s o λ_d ma è più opportuno scegliere λ_y o anche una $\lambda_{10y} = 1.006 \times 10^{-1}$ (cioè una probabilità di decadimento in 10 anni).

Cosa ne dite? NB guardare e valutare anche come sono stati fatti i fit....

other remarks

ex. 2



INTEGER PART

nint(x) and the others, similar but different (see Lect. II) =>
ex. II requires histogram for negative and positive data values

Arrays:

possible to label the elements from a negative number or 0:

dimension array(-n:m) (e.g., useful for making histograms)

[default in Fortran: $n=1$; in c and c++: $n=0$]

Array dimension:

default : dimension array([1:]n)

but also using other dimensions e.g.: dimension array(-n:m)

Important to **check dimensions** of the array when compiling or during execution !

If not done, it is difficult to interpret error messages (typically: “segmentation fault”), or even possible to obtain unpredictable results!

Default in gfortran:

boundaries not checked; use **compiler option**:

gfortran -fcheck=bounds myprogram.f90

(obsolete but still active alternative: -fbounds-check)

Typing (Unix line command):

man gfortran

you can scroll the manual pages and see the possible compilation options