

# Post P5, pre P6

Bruno Juliá-Díaz (brunojulia@ub.edu)

Dpto. Física Quàntica i Astrofísica

Facultat de Física

**Universitat de Barcelona**

**Curso 2016/2017**

# Sobre la práctica 5

+

- > subgauss correcta, en general
- > histograma correcto

-

- > Problemas con la estructura
  - > Exceso de vectores
  - > Bucles incorrectos
- > Problemas al contar
- > Fallos al extraer información de la simulación
- > Uso de variables de nombres raros

# Exceso de vectores

```
PARAMETER(ncaixes=120)
DOUBLE PRECISION, DIMENSION(ndat) :: xdata
DOUBLE PRECISION, DIMENSION(200) :: t
DOUBLE PRECISION, DIMENSION(400) :: delta
DOUBLE PRECISION, DIMENSION(200) :: deltaxx
DOUBLE PRECISION, DIMENSION(200) :: deltayy
DOUBLE PRECISION, DIMENSION(200) :: deltaxxx
DOUBLE PRECISION, DIMENSION(200) :: deltayyy
DOUBLE PRECISION, DIMENSION(200) :: x
DOUBLE PRECISION, DIMENSION(200) :: y
DOUBLE PRECISION, DIMENSION(200) :: x1
DOUBLE PRECISION, DIMENSION(200) :: y1
DOUBLE PRECISION, DIMENSION(200) :: x2
DOUBLE PRECISION, DIMENSION(200) :: y2
DOUBLE PRECISION, DIMENSION(200) :: x3
DOUBLE PRECISION, DIMENSION(200) :: y3
DOUBLE PRECISION, DIMENSION(200) :: x4
DOUBLE PRECISION, DIMENSION(200) :: y4
DOUBLE PRECISION, DIMENSION(ncaixes) :: xhisto
DOUBLE PRECISION, DIMENSION(ncaixes) :: histo
DOUBLE PRECISION, DIMENSION(ncaixes) :: errhis
DOUBLE PRECISION, DIMENSION(9) :: mom
```

```
PARAMETER (ndat=100000)
PARAMETER (ncaixes=120)
PARAMETER(nmol=250)
DIMENSION xgaus(1:ndat), histo(1:ncaixes),errhisto(1:ncaixes)
DIMENSION xhisto(1:ncaixes),xmol(nmol),ymol(nmol)
DIMENSION xmol1(1:200),xmol2(1:200),xmol4(1:200),xmol3(1:200)
DIMENSION ymol1(1:200),ymol2(1:200),ymol4(1:200),ymol3(1:200)
DIMENSION varianca(1:200),final(1:4)
CREEM NOMBRES GAUSSIANS
```

# Bucles incorrectos / estructura

```
!SEGUIM.
x0=0.d0
y0=0.d0
OPEN(50,file='Posicions.dat')
DO k=0,249 !250 partícules
DO l=1,200
posiciox(l)=x0 !Reinicialitzem
posicioy(l)=y0
ENDDO
WRITE(50,*) 'Partícula',k+1
DO t=1,199 !200 instants
posiciox(t+1)=posiciox(t)+xgauss2(t+200*k+1) !No voler
posicioy(t+1)=posicioy(t)+xgauss2(t+200*k+50000+1)
WRITE(50,*) posiciox(t),posicioy(t) !Escrivim posi
ENDDO
ENDDO
CLOSE(50)
!Ara volem fer un plot amb les 4 primeres.
DO k=0,3
IF (k.eq.0) THEN
OPEN(100+100*k,file='Posicionsparticula1.dat')
ELSEIF (k.eq.1) THEN
OPEN(100+100*k,file='Posicionsparticula2.dat')
ELSEIF (k.eq.2) THEN
OPEN(100+100*k,file='Posicionsparticula3.dat')
ELSEIF (k.eq.3) THEN
OPEN(100+100*k,file='Posicionsparticula4.dat')
ENDIF
DO l=1,200
posiciox(l)=x0 !Reinicialitzem
posicioy(l)=y0
ENDDO
DO t=1,199 !200 instants
posiciox(t+1)=posiciox(t)+xgauss2(t+200*k+1) !No voler
posicioy(t+1)=posicioy(t)+xgauss2(t+200*k+50000+1)
WRITE(100+100*k,*) posiciox(t),posicioy(t) !Escriv
ENDDO
CLOSE(100+100*k)
-----
```

Mucho calculo  
desaprovechado

No se guarda ninguna  
información relevante

Solo se calcula...1 partícula

Intento,  
infructuoso, de  
escribir las  
posiciones

# Problemas al contar

Solamente utiliza 250 de los  
100000 números gaussianos  
...o 200

```
ENDDO  
FEM CANVI VARIABLE DE GAUSS
```

```
DO I=1,250
```

```
XCANVI(I)=XGAUSS(I)*DSQRT(DELTA)
```

```
YCANVI(I)=XGAUSS(I)*DSQRT(DELTA)
```

```
ENDDO
```

```
DO DT=2,200
```

```
DO I=1,250
```

```
X(DT,I)=X(DT-1,I)+XCANVI(I)*0.01D0/DBLE(DT)
```

```
Y(DT,I)=Y(DT-1,I)+YCANVI(I)*0.01D0/DBLE(DT)
```

```
ENDDO
```

```
WRITE(20,*) X(DT,1),Y(DT,1),X(DT,2),Y(DT,2),X(DT,3),Y(DT,3)
```

```
c ,X(DT,4),Y(DT,4)
```

```
ENDDO
```

```
LA GRÀFICA 2 ESTA MALAMENT JA QUE AQUESTA NO POT SER LA TRAJECTO
```

```
DO i=1,250
```

```
igauss=1
```

```
DO ind=1,200
```

```
deltaxx(ind)=delta(igauss)
```

```
deltayy(ind)=delta(igauss+1)
```

```
igauss=igauss+2
```

```
ENDDO
```

```
t0=-0.01
```

```
deltat=0.01
```

```
x0=0
```

```
y0=0
```

```
DO ind=1,200
```

```
t(ind)=t0+deltat*ind
```

```
deltaxxx(ind)=(2.d-7)**(0.5)*deltaxx(ind)
```

```
deltayyy(ind)=(2.d-7)**(0.5)*deltayy(ind)
```

```
x(ind)=x0+deltaxxx(ind)
```

```
y(ind)=y0+deltayyy(ind)
```

```
x0=x(ind)
```

```
y0=y(ind)
```

# Problemas al contar

Excesivamente complicado ...  
probablemente correcto

```
Z = 0
T = 0.D0
VM = 0.D0
VR = 0.D0
NO SÉ PERQUÉ PERÓ ELS VALORS ALEATORIS NO S'ESTAN GENERANT CORRE
OSTRA UNA TRAJECTORIA GENERADA ALEATORIAMENT DE MANERA GAUSSIANA
DO I = 1,200
  T = T + DT
  DO K = 1,NP
    X(K,I) = X(K,I) + VARN*XGAUSS(2*K-1 +Z)
    Y(K,I) = Y(K,I) + VARN*XGAUSS(2*K + Z)
    VM = VM + X(K,I)
  ENDDO
  VALM = VM/NP
  DO R=1,NP
    VR = VR + (X(R,I)-VALM)**2.D0
  ENDDO
  VAR = (VR)/NP
EL CONTADOR Z SIRVE PARA SEGUIR AVANZANDO POR LOS VALORES ALEATO
WRITE(14,*) X(1,I),Y(1,I),X(15,I),Y(15,I),X(150,I),Y(150,
S  X(202,I),Y(202,I)
write(16,*) T, VAR
Z = Z + 2*NP
ENDDO
```

# Problemas al contar

Utiliza el mismo número para las 250 partículas en cada paso de tiempo.

```
viació
bucle per temps
do i=1,tmax,1
    igauss=igauss+i
    Bucle per cada partícula
    do j=1,Nm,1      !Ho fem per 4 molècules
        xgauss(igauss)=xgauss(igauss)*dSQRT(delta*t)
        Vx(j,i)=Vx(j,i)+xgauss(igauss)
        Vy(j,i)=Vy(j,i)+xgauss(igauss+50000)
```

# Problemas al seguir las partículas

Cuenta correctamente los números gaussianos

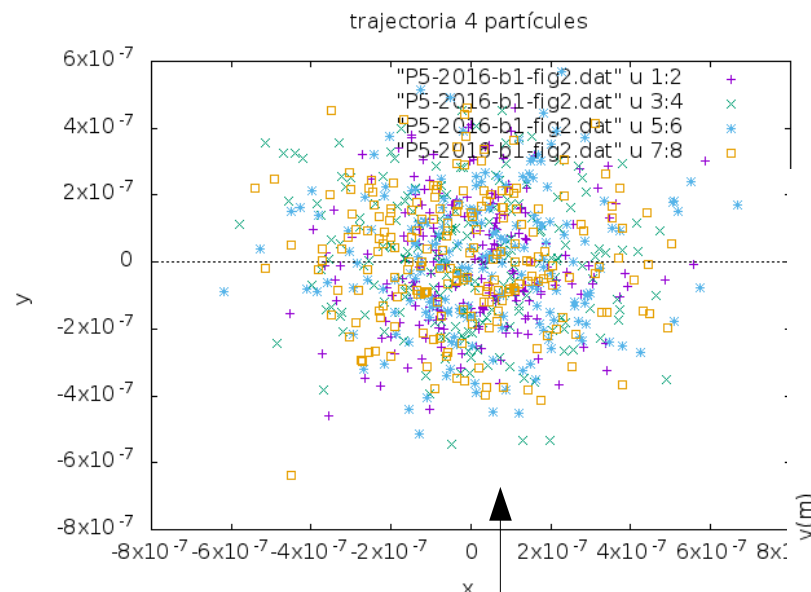
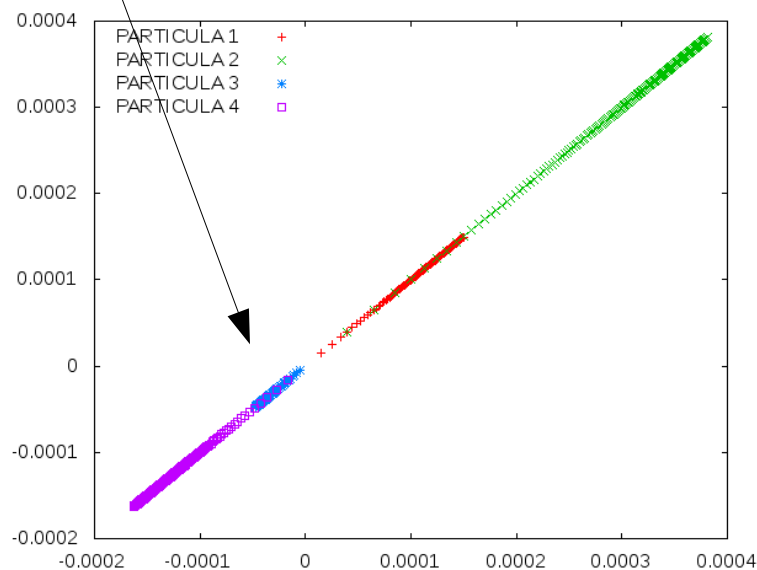
```
igauss=0
DO i=1,200
  DO n=1,250
    igauss=igauss+1
    x(n,i) = x(n,i) + des*xgauss(igauss)
    igauss=igauss+1
    y(n,i) = y(n,i) + des*xgauss(igauss)
    A(i) = (x(n,i))/250.d0 + A(i)
    S(i) = ((x(n,i))**2)/250.d0 + S(i)
  ENDDO
  var1(i)= (S(i)-((A(i))**2))
  t=(i*0.01)-0.01
  write(11,300) t, var(i)
  write(60,300) x(1,i), y(1,i)
  write(70,300) x(2,i), y(2,i)
  write(80,300) x(3,i), y(3,i)
  write(90,300) x(4,i), y(4,i)
Format(e20.12, 1X, e20.12)
ENDDO
```

No hace que la posición actual sea la anterior más una variable aleatoria  
 $x(n,i)=x(n,i-1)+ \text{aleatorio}$

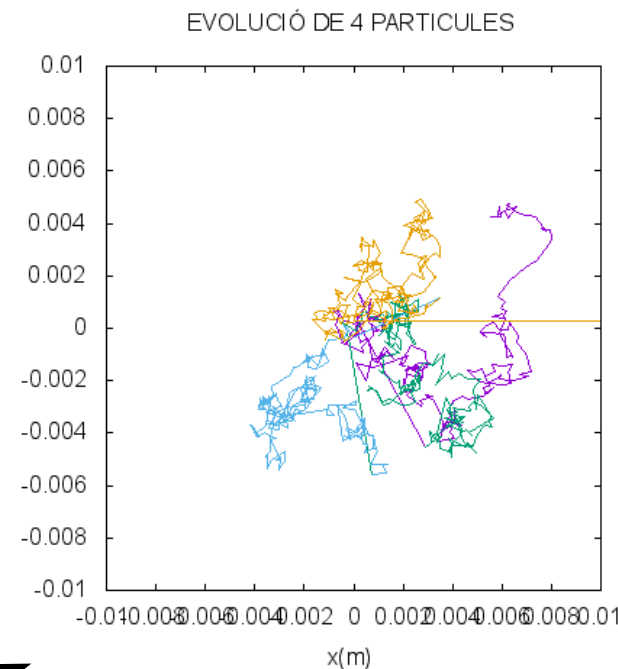


# Figuras que delatan problemas

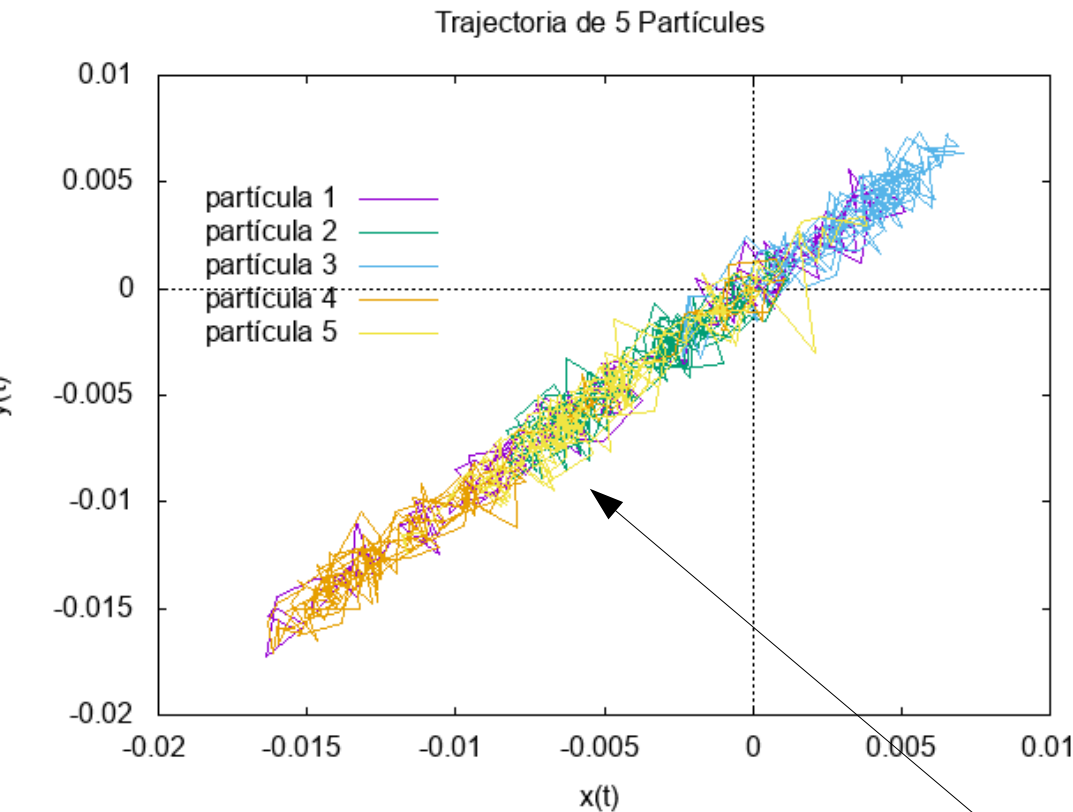
Correlación extrema,  
probablemente utiliza el  
mismo número aleatorio  
todo el rato...



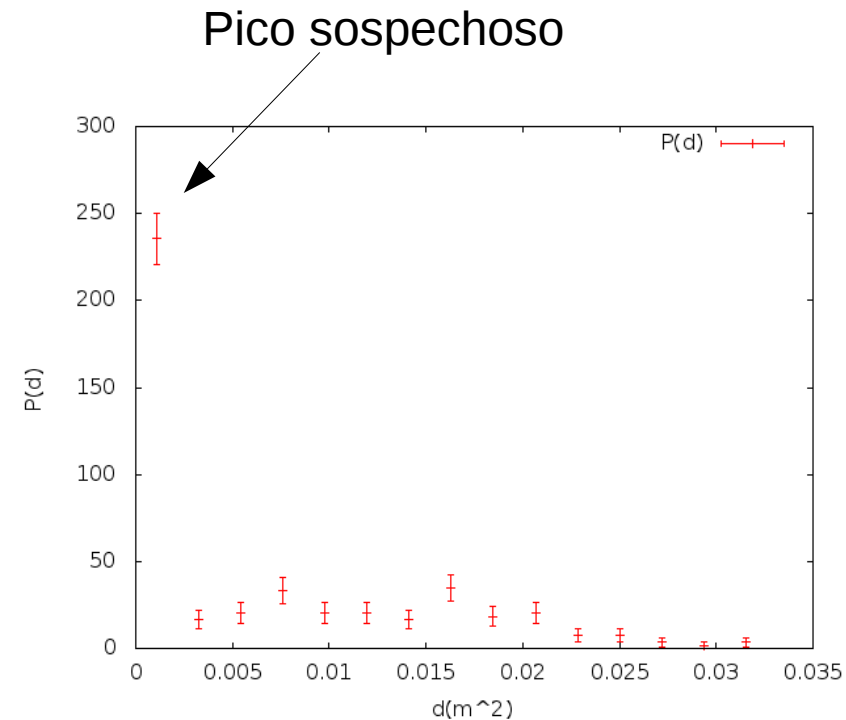
Las  
moleculas no  
se alejan  
Comparad la  
escala:



# Figuras que delatan problemas

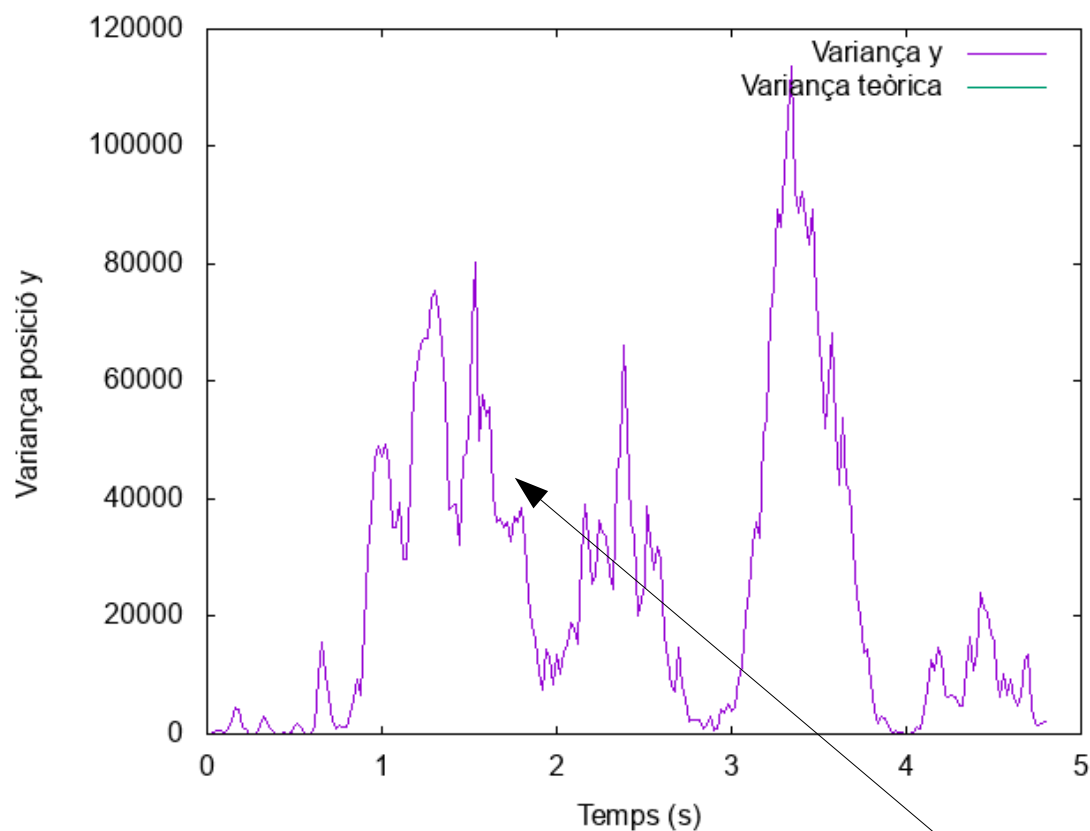


De donde procede esta gran correlación?

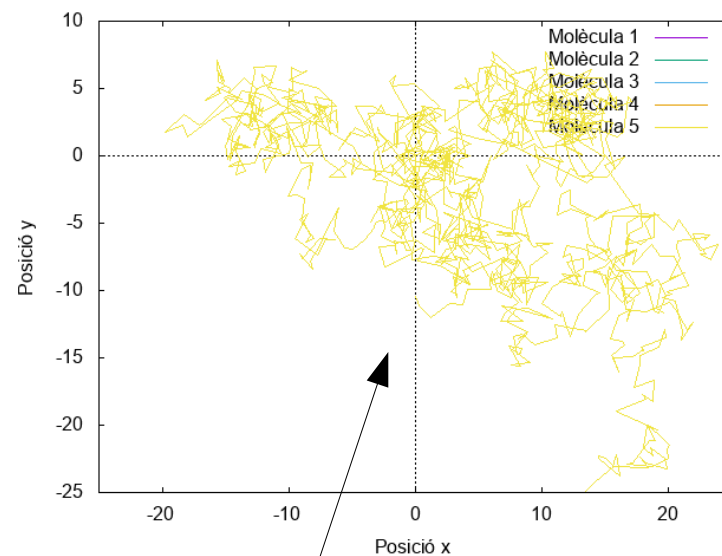


```
k=1
DO t=2,passos
  DO n= 1, numpart
    x(n,t)= x(n,(t-1)) + desvest*xgaus(k)
    k=k+1
    y(n,t)= x(n,(t-1)) + desvest*xgaus(k)
    k=k+1
    print*,k,xgaus(k)
  ENDDO
ENDDO
```

# Figuras que delatan problemas



Excesiva estructura



Excesiva casualidad

# Unidades

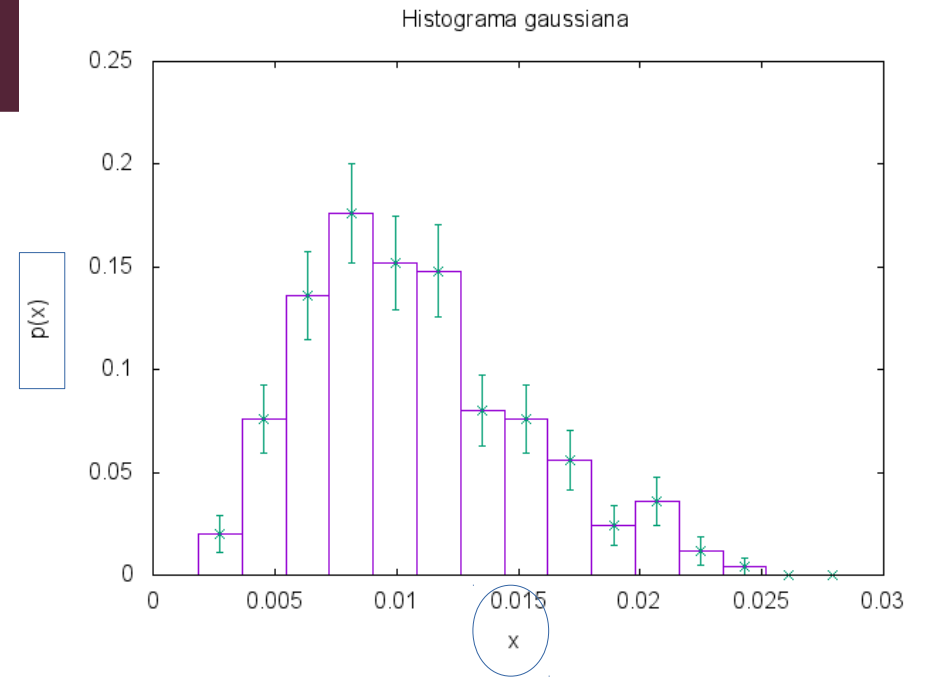
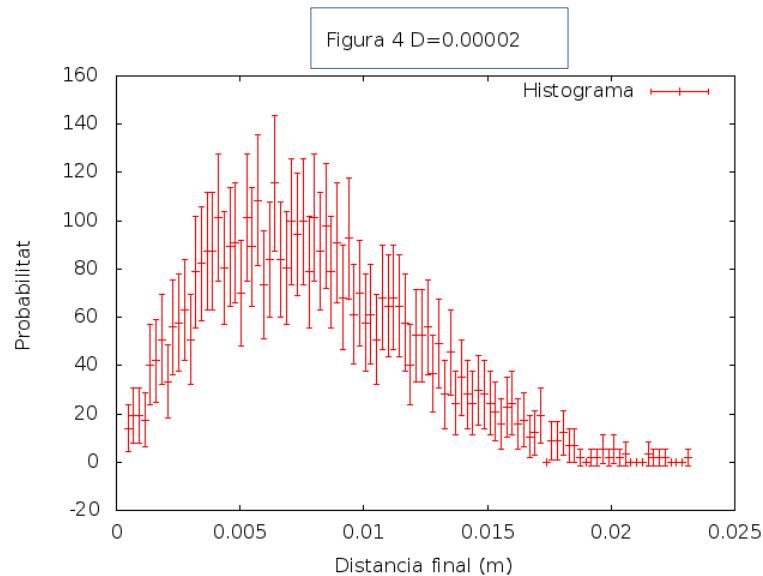
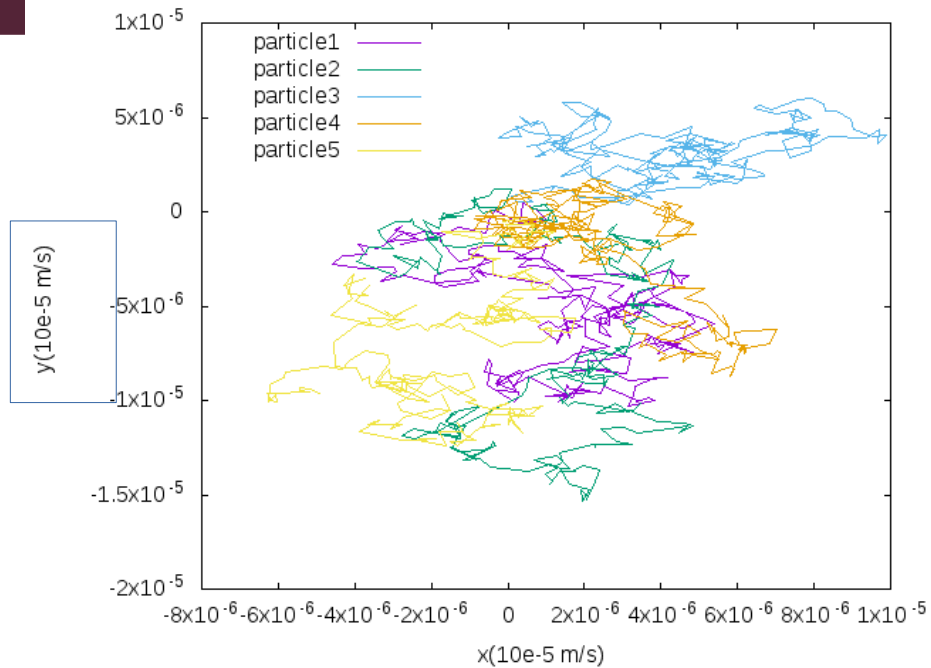
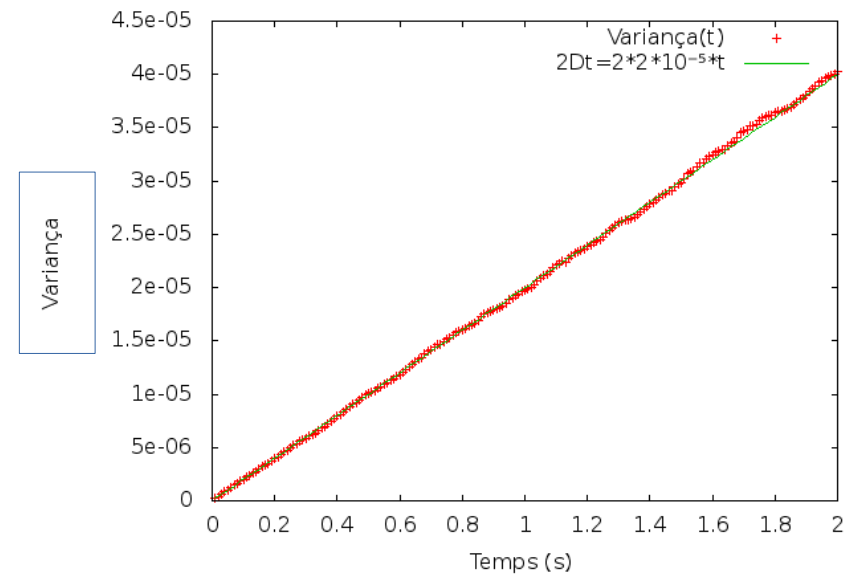


Figura 3



# Problemas más sutiles

```
NO PUC FER 200 PASSOS JA QUE EL MEU ORDINADOR EXPLOTA  
parameter(TEMPS=20)
```

← Problema grave

El problema en realidad no es pasar de 20 a 200 tiempos sino al pasar de 10000 a 100000 puntos

Lo detectaron en el foro :

**Conclusión: revisad vuestras subrutinas y evitad este caso**



**Mètode de Box Müller quan s'obté  $\text{rand}()=0.0000000000000000$**   
per [Sanchez Gimenez Victor](#) - divendres, 4 novembre 2016, 21:11

M'ha passat una cosa curiosa i és que un dels números aleatoris que demano amb `rand()` dona exactament 0 dins del límit de precisió del double precision (0.0000000000000000). Al fer-ne el logaritme per calcular una distr normal, això dona infinity, causant tota mena de problemes.

Quin criteri creieu millor a seguir per solucionar aquest cas i conservar la màxima aleatorietat?

Possibles solucions són fer un test i posar un 1 a l'última xifra si es dona el cas (0.0000000000000001) , o cridar a `rand()` per obtenir un altre número.

[Edita](#) | [Suprimeix](#) | [Contesta](#)



**Re: Mètode de Box Müller quan s'obté  $\text{rand}()=0.0000000000000000$**   
per [Julià Diaz Bruno](#) - diumenge, 6 novembre 2016, 19:52

Victor,

caram, si que es curiós si. Fixa't que si treus qualsevol "punt" (regió de mida nul·la) el resultat no es veu afectat. Pots demanar-li que només faci coses si  $0 < 0 < 1$  (sense  $\leq$ ),

B

[Mostra el missatge original](#) | [Edita](#) | [Parteix](#) | [Suprimeix](#) | [Contesta](#)

# Prepráctica 6

## Pre-Pràctica 6: Nombres aleatoris 2

Objectius: [Métodes de Montecarlo \(cru, sampleig d'importància\)](#), nombres aleatoris

— Nom del programa principal **P6-2016.f**.

Estructura el programa amb una subrutina per a cada apartat, 1 i 2.

Precisió de reals: **double precision**.

Totes les sortides de dades a **P6-2016-res.dat**.

La práctica tendrá la misma estructura que la prepráctica.

# Prepráctica 6

## 1) Integrals Montecarlo 1D. Subrutina `montecarloP6`.

a) Fes servir el mètode de Montecarlo cru per a calcular les següents integrals definides,

$$I_1 = \int_{-1}^1 \sqrt{1-x^2} dx = \pi/2$$

$$I_2 = \int_{-\pi}^{\pi} x^2 \sin^4(x) dx = \frac{1}{32} \pi (8\pi^2 - 15)$$

Muchos números

Per a cadascuna de les integrals, calcula el valor de la integral i el seu error corresponent utilitzant  $N = 1000, 2000, \dots, 1000000$  sumands. Escriu al fitxer de dades 5 columnes:  $N$ ,  $I_1$ ,  $\sigma_{I_1}$ ,  $I_2$  i  $\sigma_{I_2}$ . Genera una figura, **P6-2016-fig1.png** que mostri la convergència dels càlculs dibuixant l'error real comès comparat amb l'error estimat.

b) Genera 1000000 de nombres gaussians amb valor mitjà igual a zero i variància 1. (fes servir `subgaus` de **P5**).

c) Genera 1000000 de nombres distribuïts segons  $p(x) = (2/\pi^2) \sin(x)^2 |x|$  amb  $x \in [-\pi, \pi]$ . (fes servir `subair` de **P5**).

d) Amb els nombres aleatoris generats a b) i c), calcula, fent servir  $N = 1000, 2000, \dots, 1000000$ , les integrals següents i escriu:  $N$ , els seus valors i errors estimats al fitxer de dades.

$$I_3 = \int_{-\infty}^{\infty} e^{-x^2/2} \frac{1}{\sqrt{2\pi}} \sin^2(x) dx,$$

$$I_4 = \int_{-\infty}^{\infty} e^{-x^2/2} \cos^2(x) dx,$$

$$I_5 = \int_{-\pi}^{\pi} \sin^4(x) x^2 dx.$$

Nota: Per  $I_3$  i  $I_4$  utilitza nombres d'1b), per  $I_5$ , d'1c).

Subrutinas de las P5

$\vec{r}'_1$

# Prepráctica 6, formulas para (0,1)

$$\int_0^1 h(x) dx \simeq \frac{1}{N} \sum_{k=1}^N h(x_k)$$

$$\sigma_{H_N} \simeq \frac{1}{\sqrt{N}} \sqrt{\frac{1}{N} \sum_{k=1}^N h^2(x_k) - \left( \frac{1}{N} \sum_{k=1}^N h(x_k) \right)^2}.$$

A tener en cuenta:

- Es fundamental tener claro cual es la distribución uniforme que utilizaremos y redefinir la “f” como corresponda.

Ejemplo:

$$\int_1^5 x dx = 4 \int_1^5 \frac{x}{4} dx = \langle 4x \rangle_{x \in U(1,5)} = 4 \times 3 = 12$$



# Prepráctica 6

Estructura de un código de montecarlo “crudo”, sencilla:

Inicializo a cero los comandos para sumar

Comienzo un bucle para hacer sumas

    Saco un numero aleatorio uniforme en  $(a,b)$ ,  $x$

    Acumulo el valor de  $f(x)$

    Acumulo el valor de  $f(x)**2$      (necesario para el error)

Acabo el bucle

Calculo el promedio de  $f(x)$

Calculo el promedio de  $f(x)**2$

El promedio de  $f(x)$  es el valor de la integral aproximado  
Para calcular el error necesito el número de sumandos y los dos promedios anteriores

# Prepráctica 6

Modificaciones razonables a esta estructura.

¿Que ocurre si en lugar de un número de sumandos nos piden que hagamos calculos con varios?

Inicializo a cero los comandos para sumar

Comienzo un bucle para hacer sumas

Saco un numero aleatorio uniforme en  $(a,b)$ ,  $x$

Acumulo el valor de  $f(x)$

Acumulo el valor de  $f(x)**2$  (necesario para el error)

Miro si el numero de sumandos parciales es el que quiero

Si es así (con el número de sumandos que tenga):

Calculo el promedio de  $f(x)$

Calculo el promedio de  $f(x)**2$

Acabo el bucle

Calculo el promedio de  $f(x)$

Calculo el promedio de  $f(x)**2$

Calculo el error usando ambos promedios

# Prepráctica 6, integral multidimensional

Fent servir els nombres aleatoris generats a 1b) (via COMMON) calcula la següent integral utilitzant per a cada càlcul  $N = 1000, 2000, \dots, 200000$  sumands. Escriu al fitxer de dades el nombre de sumands,  $N$ , el valor d' $I_6$  i l'error estimat amb el mètode de Montecarlo. Fes una figura mostrant la convergència del resultat, incloent com a títol el resultat final amb el seu error, **P6-2016-fig2.png**.

$$I_6 = \int_{-\infty}^{\infty} dx_1 \int_{-\infty}^{\infty} dx_2 \int_{-\infty}^{\infty} dx_3 \int_{-\infty}^{\infty} dx_4 \int_{-\infty}^{\infty} dx_5 g(x_1, x_2, x_3, x_4, x_5)$$

amb  $g(x_1, x_2, x_3, x_4, x_5) = (x_1^2 x_2^2 \cos(x_4) + x_3^2 (1 + x_1) + \cos(x_4)^2 x_5^2) e^{-(x_1^2 + x_2^2 + x_3^2 + x_4^2 + x_5^2)}$

La estructura es idéntica.

La única diferencia es que el número de sumandos no coincide con el número de números aleatórios

# Inciso sobre gnuplot

¿Cómo hacer varias figuras a partir del mismo fichero de datos?

- Pon los comentarios con un # en la primera columna
- Separa los bloques con 2 líneas en blanco , por ejemplo

Write(14,\*)

Write(14,\*)

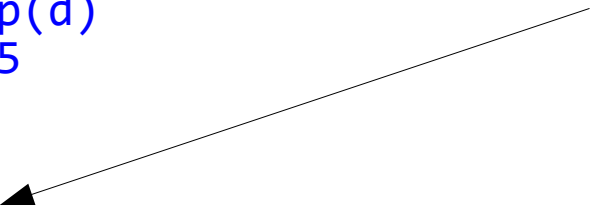
- gnuplot considera el primer bloque como “index 0”, los siguientes, index 1, index 2, etc.

Ejemplo de fichero.dat:

# apartado 1

#	d	p(d)	error	p(d)
1		3	0.5	
2		4.2	0.3	

Dos líneas en blanco



# apartado 2

#	x	y	x1	y1
	0.5	0.2	40	40.34
	0.8	0.3	35	42.2

Gnuplot> plot “fichero.dat” index 0 u 1 : 2 :3 w e

Gnuplot > plot “fichero.dat” index 1 u 3:5

etc

# Inciso sobre nombres de variables

En la medida de lo posible evitad utilizar nombres de variables de 1 solo caracter.

Lo óptimo serian unos 4 caracteres:

Itemps

Kpart

Icont

Xposi(ndat)

Yposi(ndat)