Facultad de Matemática, Astronomía, Física y Computación, U.N.C.

Computación

Aula Virtual: https://famaf.aulavirtual.unc.edu.ar/course/view.php?id=747

Resguardo tutoriales: https://www.famaf.unc.edu.ar/∼moreschi/docencia/Computacion/

Tutorial Problemas 2 de la Guía N° 6

Problema 2: Caminata aleatoria y la distribución binomial:

Un caso típico de un proceso estocástico es la llamada caminata aleatoria. Asumamos que un caminate puede ocupar sitios separados una distancia Δx a lo largo de una línea; de tal manera que podemos determinar la posición con números enteros. Pensemos que el caminante está en la posición $x = 0$ en el tiempo $t = 0$ y que dará pasos a la izquierda o a la derecha en intervalos de tiempo Δt. A la derecha significa valores crecientes y a la izquierda valores decrecientes de la posición. En cada paso el caminante tiene una probabilidad p de dar un paso a la derecha y por lo tanto una probabilidad $q = 1 - p$ de dar un paso a la izquierda.

¿Cuál es la probabilidad $p(m, N)$ de que luego de N pasos esté en la posición m? Supongamos que, sobre un total de N pasos, el caminante hizo $m + l$ pasos a la derecha y l pasos a la izquierda para llegar a m; luego se tiene $N = m + 2l$; por lo que $l = (N - m)/2$ y el caminante hizo: $(N + m)/2$ pasos a la derecha y $(N - m)/2$ pasos a la izquierda. Usando conocimientos de probabilidad uno puede inferir entonces que $p(m, N)$ viene dada por:

$$p(m,N) = \frac{N!}{(\frac{N+m}{2})!(\frac{N-m}{2})!} p^{(\frac{N+m}{2})}(1-p)^{(\frac{N-m}{2})}.$$

Si definimos $r = (\frac{N+m}{2})$ como el número de pasos a la derecha, esta probabilidad, que se conoce como la distribución binomial, se puede expresar por:

$$p(m,N) = p_N(r) = \frac{N!}{r!(N-r)!} p^r (1-p)^{N-r}.$$

a) Usando PYTHON, simule una distribución para este proceso con los comandos:

```
import numpy as np

xmin = 0.
xmax = 1.4

pe = 0.8
ene = 400

# elección de la semilla ---------------------------------------
np.random.seed(11)

# elección de Nrep ---------------------------------------------------
Nrep = 1000
x = np.linspace(xmin, xmax, Nrep)
y = np.random.binomial(ene,pe,Nrep)
```

donde estamos usando la función BINOMIAL del paquete RANDOM de NUMPY y asignaremos la distribución al espacio de las x.

b) Defina un arreglo con los valores de la función binomial con los siguientes comandos:

```
from scipy.stats import binom

pe = 0.8
ene = 400

dist = [binom.pmf(r, ene, pe) for r in range(ene+1) ]
```

c) Genere un gráfico de y vs x, mostrando el resultado en pantalla y guardando el mismo en un archivo con nombre 'graficos/p2-binomial-'+str(ene)+'-'+str(Nrep)+'.png'

d) Genere un gráfico del histograma de este proceso, de la función de Gauss y de la función binomial. Elija los $pro = ene * pe$ y $anc = np.sqrt(ene * pe * (1 - pe))$, haciendo uso de los comandos:

```
n, bins, patches = plt.hist(y, numbines, density=False, facecolor='g',
alpha=0.75, label='histograma')
suma = n.sum()
hbin = (bins[-1]-bins[0])/numbines
integ = suma*hbin
plt.plot(bins , (integ*gau(bins,pro,anc)),'r',
          label='integ * Gausiana mu='+str(pro)+', sigma='+str(anc) )
dist = Nrep*binom.pmf(bins, ene, pe)
plt.plot(bins , dist,'b', label='binomial(r,p,n) p='+str(pe)+',
          n='+str(ene)+', Nrep='+str(Nrep) )
```

muéstrelo en pantalla y guarde el gráfico en un archivo con el nombre 'graficos/p2-histograma-'+str(ene)+'-'+str(Nrep)+'.png'

e) Repita todo lo anterior ahora con la elección $N = 10000$ y compare las simulaciones con la función gaussiana.

Los gráficos deben ser completados con titulo, etiquetas de curvas, etiquetas de ejes, etc.

Tutorial:

- Guarde en el archivo `p2.py` las siguientes instrucciones:

```
"""
definimos una variable con random y
hacemos el histograma correspondiente
"""
import numpy as np
import matplotlib.pyplot as plt
from scipy.stats import binom

def gau(xx,prom,ancho):
    return np.exp(-(xx-prom)**2/(2.*ancho**2))/( np.sqrt(2.*np.pi)*ancho )

numbines = 50
xmin = 0.
xmax = 1.4
```

```python
pro = 0.
anc = 1.

pe = 0.8
ene = 400

# list of pmf values
dist = [binom.pmf(r, ene, pe) for r in range(ene+1) ]
print('dist =',dist)
print('len(dist) =',len(dist))
print('min(dist) =',min(dist))
print('max(dist) =',max(dist))

# elección de la semilla -----------------------------------------
np.random.seed(11)

# elección de Nrep -----------------------------------------------
Nrep = 1000
x = np.linspace(xmin, xmax, Nrep)
y = np.random.binomial(ene,pe,Nrep)

print('x =',x)
print('y =',y)
print('len(x) =',len(x))
print('len(y) =',len(y))
print('min(y) =',min(y))
print('max(y) =',max(y))
print('max(y)-min(y) =',(max(y)-min(y)) )
numbines = (max(y)-min(y))
print('      numbines =',numbines)

pro = ene*pe
anc = np.sqrt(ene*pe*(1-pe) )
print('  pro =',pro)
print('  anc =',anc)


plt.figure( figsize=(10, 7.5) )
plt.title('Distribución binomial con '+str(ene)+' pasos y '+str(Nrep)
          +' puntos')
plt.xlabel('x')
plt.ylabel('y')
plt.grid()
plt.plot(x , y,'o-', label='binomial')
plt.legend(loc="best")
plt.savefig('graficos/p2-binomial-'+str(ene)+'-'+str(Nrep)+'.png', dpi=100)
plt.show()


plt.figure( figsize=(10, 7.5) )
plt.title('Histograma de la distribución binomial con '+str(ene)+' pasos y '
          +str(Nrep)+' puntos')
plt.xlabel('y')
plt.ylabel('dn/dy')
plt.grid()
n, bins, patches = plt.hist(y, numbines, density=False, facecolor='g',
```

```
                              alpha=0.75, label='histograma')
suma = n.sum()
hbin = (bins[-1]-bins[0])/numbines
integ = suma*hbin
plt.plot(bins , (integ*gau(bins,pro,anc)),'r',
         label='integ * Gausiana mu='+str(pro)+', sigma='+str(anc) )
dist = Nrep*binom.pmf(bins, ene, pe)
plt.plot(bins , dist,'b', label='binomial(r,p,n) p='+str(pe)+
         ', n='+str(ene)+', Nrep='+str(Nrep) )
plt.legend(loc="best")
plt.savefig('graficos/p2-histograma-'+str(ene)+'-'+str(Nrep)+'.png', dpi=100)
plt.show()

#print('      suma =',suma)
#print('      hbin =',hbin)
#print(' suma*hbin =',(suma*hbin))
#print('         n =',n)
#print('      bins =',bins)
#print('      dist =',dist)
#print(' len(bins) =',len(bins))
#print('   patches =',patches)
#print('patches[0] =',patches[0])

print('---------------------')

# elección de Nrep -------------------------------------------------------
np.random.seed(11)

Nrep = 10000
x = np.linspace(xmin, xmax, Nrep)
y = np.random.binomial(ene,pe,Nrep)

numbines = (max(y)-min(y))
print('      numbines =',numbines)

plt.figure( figsize=(10, 7.5) )
plt.title('Distribución binomial con '+str(ene)+' pasos y '+str(Nrep)
          +' puntos')
plt.xlabel('x')
plt.ylabel('y')
plt.grid()
plt.plot(x , y,'o-', label='binomial')
plt.legend(loc="best")
plt.savefig('graficos/p2-binomial-'+str(ene)+'-'+str(Nrep)+'.png', dpi=100)
plt.show()


plt.figure( figsize=(10, 7.5) )
plt.title('Histograma de la distribución binomial con '+str(ene)+' pasos y '
          +str(Nrep)+' puntos')
plt.xlabel('y')
plt.ylabel('dn/dy')
plt.grid()
n, bins, patches = plt.hist(y, numbines, density=False, facecolor='g',
                              alpha=0.75, label='histograma')
suma = n.sum()
```

```
hbin = (bins[-1]-bins[0])/numbines
integ = suma*hbin
plt.plot(bins , (integ*gau(bins,pro,anc)),'r', label='integ * Gausiana mu='
         +str(pro)+', sigma='+str(anc) )
dist = Nrep*binom.pmf(bins, ene, pe)
plt.plot(bins , dist,'b', label='binomial(r,p,n) p='+str(pe)+', n='+str(ene)
         +', Nrep='+str(Nrep) )
plt.legend(loc="best")
plt.savefig('graficos/p2-histograma-'+str(ene)+'-'+str(Nrep)+'.png', dpi=100)
plt.show()

#quit()
```

- Desde la terminal ejecute:

  python3 p2.py

  e interprete el resultado.

  Alternativamente ejecute:

  python3

  y vaya agregando uno a uno los bloques del programa.

- Estudie cada paso del programa y agrege comentarios explicativos.

- Altere el programa para probar distintas cosas.

Se deberían generar los siguientes gráficos:

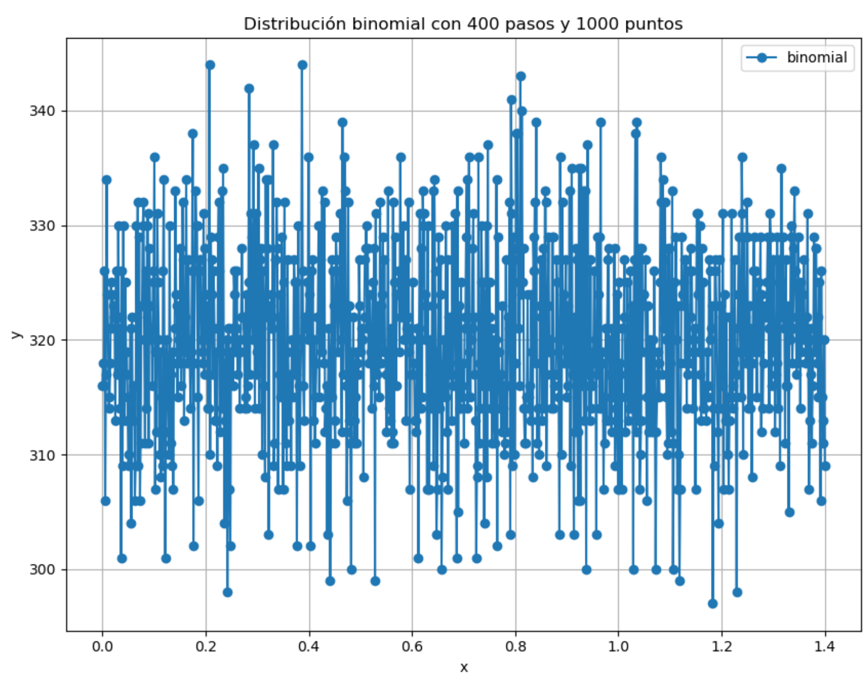

Figura 1: Distribución binomial de 400 pasos con 1000 puntos.

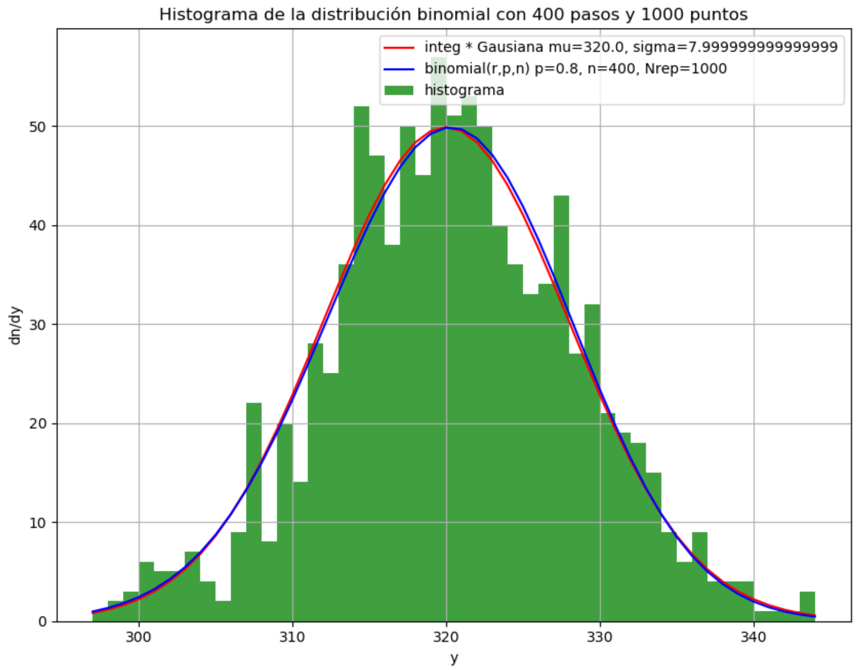

Figura 2: Histograma de la distribución binomial de 400 pasos con 1000 puntos.

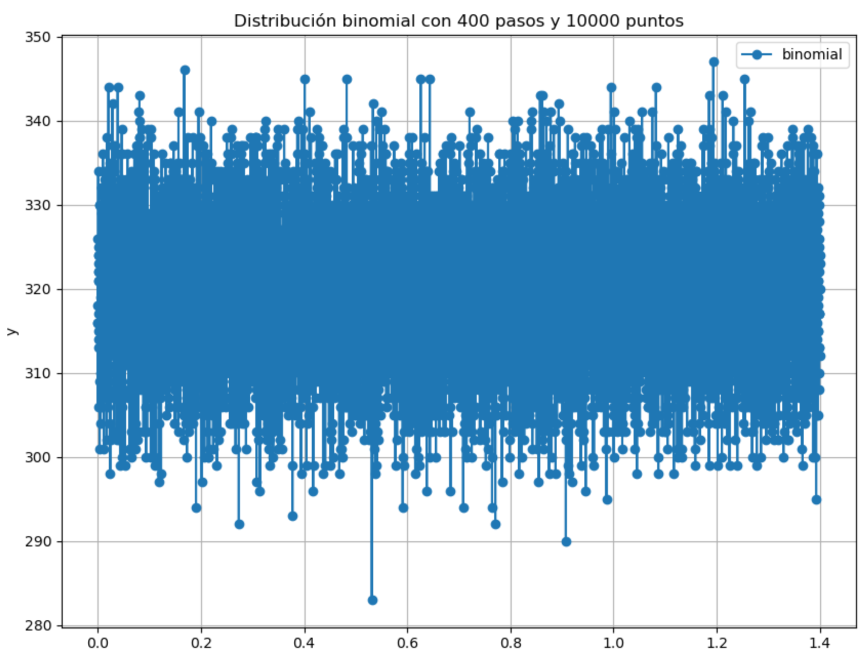

Figura 3: Distribución binomial de 400 pasos con 10000 puntos.

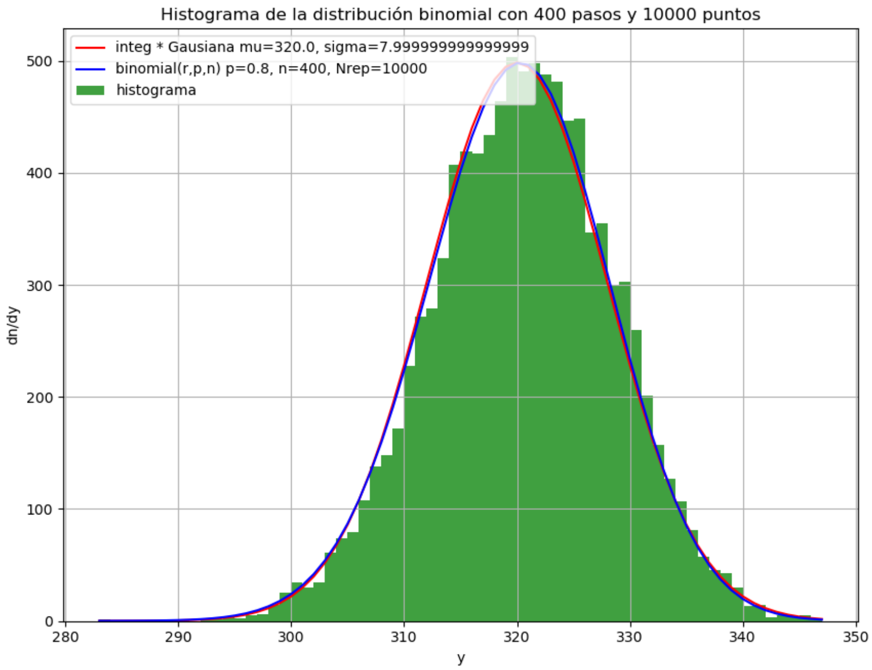

Figura 4: Histograma de la distribución binomial de 400 pasos con 10000 puntos.