

# 118-PROBA-Paseo-2dim

April 23, 2018

En este ejercicio estudiamos paseos aleatorios en dos dimensiones. El paseante comienza en el origen  $(0,0)$  de  $\mathbb{R}^2$  y se mueve por los puntos con coordenadas enteras. En cada momento de tiempo  $(t = 0, 1, 2, 3, \dots)$  elige aleatoriamente uno de sus cuatro puntos (con coordenadas enteras) vecinos y se desplaza a él. El problema que queremos estudiar es la determinación de la probabilidad de que el paseante vuelva en algún momento al origen. El problema que encontramos es que el tiempo que tarda en volver puede ser astronómicamente grande y usando el ordenador estamos limitados. Para poder tratar este problema debemos fijar un tiempo máximo de espera  $T$  ( $10^5$  en el ejemplo de más abajo), y nos dejaremos convencer de que la probabilidad de retorno es 1 si vemos que va aumentando al dejar crecer  $T$  y parece que se aproxima a 1.

Estudiar el caso 3-dimensional de este mismo problema.

Estudiar la variante en la que consideramos dos paseantes aleatorios que comienzan en  $t=0$  en el origen  $(0,0)$ , y queremos estudiar la probabilidad de que se reencuentren (estén en el mismo momento de tiempo en el mismo lugar).

```
In [1]: def actualizar(L):
        x = randint(1,4)
        if x == 1:
            L[0] = L[0]+1
            return L
        elif x == 2:
            L[1] = L[1]+1
            return L
        elif x == 3:
            L[0] = L[0]-1
            return L
        else:
            L[1] = L[1]-1
            return L

In [2]: def retorno():
        cont = 0
        Pini = actualizar([0,0])
        while (Pini != [0,0] and cont<= 10^5):
            Pini = actualizar(Pini)
            cont += 1
            #if cont%10000 == 0:
            #print cont,Pini
        return cont
```

```
In [3]: def probabilidad(N):  
        contador = 0  
        for muda in xrange(N):  
            cont = retorno()  
            ##print cont  
            if cont != 10^5+1:  
                contador += 1  
            if muda%100 == 0:  
                print muda  
        return (contador/N).n()
```

```
In [4]: probabilidad(10^3)
```

```
0  
100  
200  
300  
400  
500  
600  
700  
800  
900
```

```
Out [4]: 0.7940000000000000
```