Construya una funcion que returne la varianza de un conjunto de datos ( $n \ge 2$ ) cuando se va incorporando dato a dato. Use como primer par de datos los que se generan de la distribucion normal con  $\mu = 3$  y s = 0,3.Use la misma semilla y los mismos parametros de la distribucion para incorporar cada dato. Haga el proceso 50 veces y luego grafique la varianza como funcion de n.

```
1
 2 import numpy as np
 3 import random
 4 import statistics
 5 import matplotlib.pyplot as plt
 6 %matplotlib inline
 7 np.random.seed(1)
 8 Ddn1= np.random.normal(3,0.3, 2)
 9 varm1=statistics.variance(Ddn1)
10 print(f'La varianza muestral de los primeros 2 datos es {varm1}')
11 np.random.seed(2)
12 Ddn2= np.random.normal(3,0.3, 2)
13 varm2=statistics.variance(Ddn2)
14 print(f'La varianza siguientes 2 datos es {varm2}')
15 np.random.seed(3)
16 Ddn3= np.random.normal(3,0.3, 2)
17 varm3=statistics.variance(Ddn3)
18 print(f'La varianza siguientes 2 datos es {varm3}')
19 np.random.seed(4)
20 Ddn4= np.random.normal(3,0.3, 2)
21 varm4=statistics.variance(Ddn4)
22 print(f'La varianza siguientes 2 datos es {varm4}')
23 np.random.seed(5)
24 Ddn5= np.random.normal(3,0.3, 2)
25 varm5=statistics.variance(Ddn5)
26 print(f'La varianza siguientes 2 datos es {varm5}')
27
28 Ddn=[Ddn1,Ddn2,Ddn3,Ddn4,Ddn4]
29 print(f'los datos son {Ddn}')
30
31 varm=[varm1, varm2, varm3, varm4, varm5]
32 print(f'las varianzas son {varm}')
33
34 \text{ Ddnx} = [3.48730361, 2.87497265, 3.53658854, 3.01516851, 3.01516851] \#Tome el primer dato
35 varmy=[0.14182393613078986, 0.11118787361302858, 0.07506739339190276, 0.059302574069518
36 plt.scatter(Ddnx,varmy);plt.xlabel('Datos distribucion normal');plt.ylabel('Varianza')#
37
38
```

```
La varianza muestral de los primeros 2 datos es 0.22500680213266996
La varianza siguientes 2 datos es 0.005847919903340884
La varianza siguientes 2 datos es 0.08227011466991509
La varianza siguientes 2 datos es 0.009087796621876669
La varianza siguientes 2 datos es 0.026826064371373807
los datos son [array([3.48730361, 2.81647308]), array([2.87497265, 2.98311995]), arralas varianzas son [0.22500680213266996, 0.005847919903340884, 0.08227011466991509, 0 Text(0, 0.5, 'Varianza')
```

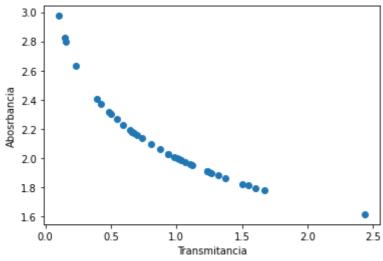


La cantidad de radiacion absorbida puede medirse por la transmitancia. La relacion entre transmitancia (T) y absorbancia (A) esta dada por la ley BeerLambert (ley de Beer). Cree una funcion para la expresion que aparece en <a href="https://www.electrical4u.com/what-is-transmittance/">https://www.electrical4u.com/what-is-transmittance/</a> y genere unos datos de T para determinar A. Grafique la funcion. Finalmente lea la seccion titulada: \*\*Why is Absorbance the Preferred Unit Over Transmittance?\*\*

```
1 import numpy as np
2 import random
3 import math
4 import matplotlib.pyplot as plt
5 np.random.seed(12)
6 T= np.random.normal(1,0.5, 40) #transmitancia
7 A=2-np.log10(T) #absorbancia
8 print(f'% de transmitancia {T}')
9 print(f'Absorbancia {A}')
10
11 plt.scatter(T,A);plt.xlabel('Transmitancia');plt.ylabel('Abosrbancia')#grafica
```

```
% de transmitancia [ 1.23649292  0.65928706  1.12121975  0.14963218  1.37657142  0.23
  1.00256354 0.93988616 0.59650906 2.4359097
                                                 0.70108854
                                                            1.2362285
                                                 1.50625774
  1.54797806 0.3924156
                         1.67117819 0.93892511
                                                            0.54306543
  0.4852349
             1.60489822 1.25093615
                                     1.06942309
                                                 1.32038056
                                                             1.26366633
  0.42281988 -0.10666674 0.15912174 0.10595287 -0.10926747
                                                             0.67628461
  0.73579784 0.98039541
                         1.10748797
                                     0.8078206
                                                 0.87304796
                                                            1.03662604
  0.50139808  0.64307186  1.01770817  0.66102732]
Absorbancia [1.90780837 2.18092545 1.95030926 2.82497499 1.86120125 2.63331686
 1.99888809 2.02692474 2.22438296 1.61333882 2.15422713 1.90790125
 1.8102352 2.40625374 1.77697724 2.02736905 1.82210071 2.26514784
 2.31404797 1.7945525 1.90276486 1.97085044 1.87930088 1.89836759
                  nan 2.79827047 2.97488726
 2.3738446
                                                   nan 2.1698705
 2.13324149 2.00859873 1.95566098 2.09268508 2.0589619 1.98437789
 2.29981733 2.1917405 1.99237674 2.17978059]
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:7: RuntimeWarning: inval
  import sys
```

Text(0, 0.5, 'Abosrbancia')



✓ 0 s completado a las 22:36

X