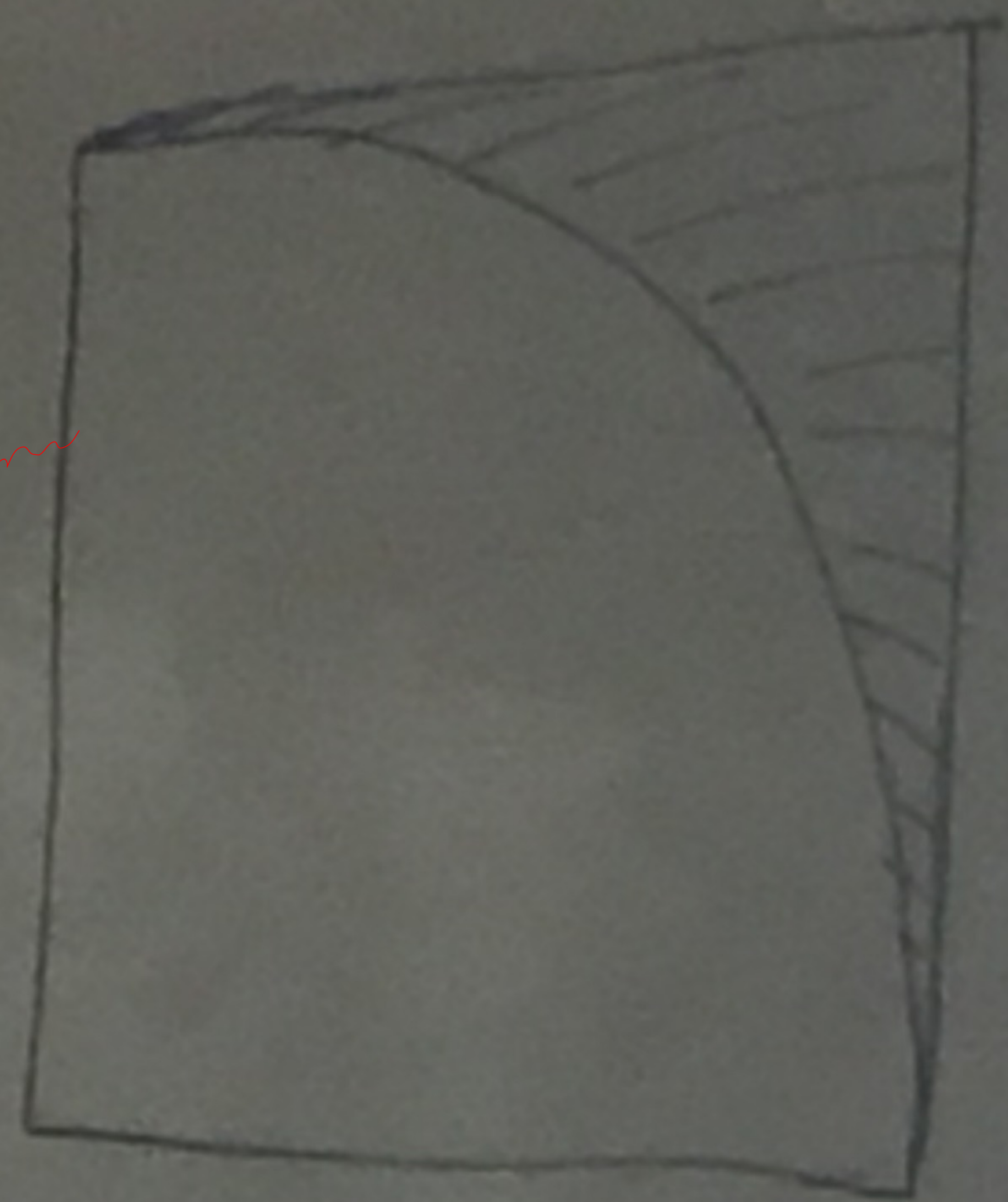


$d\sqrt{x^2 + y^2} = \text{distance}$
 (x, y)



$\sqrt{x^2 + y^2}$

$(0,0)$

\int

1m

1m


```

1 import numpy as np
2 Ndim= 2 (why) pr
3 Npoints= 10000 val 10k, 2
4 Points = np.random.rand(Npoints,Ndim)
5 print("Points")
6 print(Points)
7 dfo= np.zeros((Npoints, 1))
8 print("dfo") [0,0,0,0,0] ..
9 print(dfo)
10 Outside_points= 0
11 for i in range(Npoints) :
12     for j in range(Ndim) :
13         dfo[i] += (Points[i,j])**2 #point of isq , jsq
14         dfo[i] =np.sqrt(dfo[i]) Try if it indented
15     if dfo[i] > 1:
16         Outside_points+= 1
17 print("After FOR LOOP")
18 print("Points")
19 print(Points)
20 print("dfo")
21 print(dfo)
22 print("Outside_points ",Outside_points)
23 print("Npoints ",Npoints)
24 print("Ratio ",Outside_points/Npoints) #ideally 1-pi/4

```

Points (0,0) (0,1)

```

☞ Points
[[0.03991251 0.6317875 ]
 [0.61063513 0.62775559]
 [0.74315733 0.12294848]
 ...
 [0.14168874 0.53854663]
 [0.24351004 0.88195589]
 [0.03590138 0.34880774]]
dfo
[[0.]
 [0.]
 [0.]
 ...
 [0.]
 [0.]
 [0.]]
After FOR LOOP
Points
[[0.03991251 0.6317875 ]
 [0.61063513 0.62775559]
 [0.74315733 0.12294848]
 ...
 [0.14168874 0.53854663]
 [0.24351004 0.88195589]
 [0.03590138 0.34880774]]
dfo
[[0.66262203]
 [1.00235334]
 [0.8707891 ]
 ...
 [0.65705495]
 [1.0106217 ]
 [0.39694863]]
Outside_points 3313
Npoints 10000
Ratio 0.3313

```

Same