

CS215 ASSIGNMENT 2

Utkarsh Varun

Question 1

Contents

1	Introduction	2
2	Algorithm for Generating Random Points inside the Ellipse	2
2.1	Code Explanation	2
2.2	Plotting 2D Histogram of Ellipse sample	3
2.2.1	q1_b.py Ellipse 2D histogram plot	3
3	Algorithm for Generating Random Points inside the Triangle	4
3.1	Code Explanation	4
3.2	Plotting 2D Histogram of Triangle sample	4
3.2.1	q1_d.py Triangle 2D histogram plot	4

1 Introduction

I and my partner had agree to use Python programming platform for this question. And for plotting graphs, we use pyplot function from matplotlib package. Also, i used numpy mathematical package for coding.

I had implement the problem statement given in assignment in 4 different files. I had also mention some comments in the code for better understanding.

Those 4 files are as follows:-

- 1) q1_a.py -> For generating a random 2D point inside a ellipse.
- 2) q1_b.py -> For generating N=10⁷ random 2D points inside the ellipse and plotting the 2D histogram of the produced sample of points.
- 3) q1_c.py -> For generating a random 2D point inside a triangle.
- 4) q1_d.py -> For generating N=10⁷ random 2D points inside the triangle and plotting the 2D histogram of the produced sample of points.

When you run first q1_a.py and q1_c.py file, we will get a random point inside given ellipse and triangle respectively in 2D Euclidean plane.

These files are just to show the algorithm for generating random points inside a shape in 2D Euclidean Plane.

Upon running q1_b.py and q1_d.py, we will get the 2D histogram plot of sample of N=10⁷ random points inside the ellipse and triangle.

These files will produce plots in q1_b.png and q1_d.png respectively in the same directory as of code i.e, "code/Question1".

I had also submit all plots in the "results" directory and all 4 python files in "code" directory.

2 Algorithm for Generating Random Points inside the Ellipse

2.1 Code Explanation

Implemented the main logic of algorithm in file q1_a.py.

Output of q1_a.py file :- -0.08037221600046039 -0.7231699188351421

As we know that properties of Ellipse resembles with that of circle.

So, i had used the concept of polar coordinate as Ellipse is also a circular curve.

Firstly, i had generate random variable r(radius) in (0,1) and theta(angular angle) in (0,2*pi) which will give us points inside a unit circle around centre as origin.

Thenafter, i had convert this polar coordinate (r,theta) into cartesian coordinate (x,y).

I had used sqrt(r) as this will give us accurate result as r is generating independently and ($x^2 + y^2 = r$) should be satisfied.

Also, if we didn't do square root, the randomly generated points will concentrated towards centre i.e, we will not get uniform distribution in that case.

After that just multiply x and y coordinate with (b) and (a) value of Ellipse which is respectively ($b = \frac{\text{minor_axislength}}{2}$; $a = \frac{\text{major_axislength}}{2}$) so that we get points inside ellipse $x^2/b^2 + y^2/a^2 = r$.

I assume that Ellipse is vertical Ellipse.

2.2 Plotting 2D Histogram of Ellipse sample

I had write the algorithm for plotting 2D histogram in file q1_b.py.

Firstly, i had generate set of $N=10^7$ sample random 2D points inside the given Ellipse with major axis length as 2 unit along y-axis and minor axis length as 1 unit along x-axis.

And then just use the hist2 function in python to plot the 2D histogram plots of the obtained sample of points.

I had use (bins=500) for approximating the plot to better extents.

2.2.1 q1_b.py Ellipse 2D histogram plot

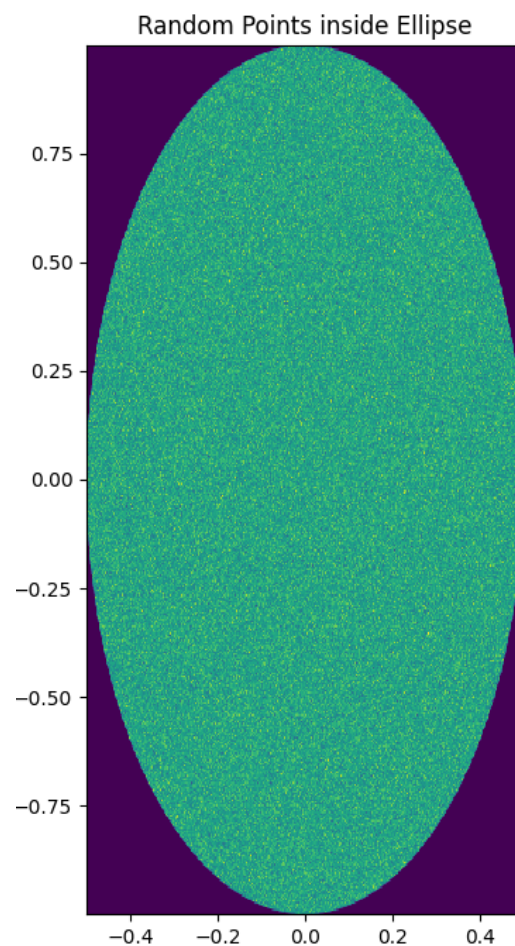


Figure 1: Ellipse 2D histogram plot

3 Algorithm for Generating Random Points inside the Triangle

3.1 Code Explanation

Implemented the main logic of algorithm in file q1_c.py.

Output of q1_a.py file :- 1.1712221535554261 0.7176342049968248

I had use the property linear combination of triangle that is used to get a point inside the triangle.

Logic of my code

let three variable u, v, w and $(x_1, y_1), (x_2, y_2), (x_3, y_3)$ are the vertices of a triangle. Then,

Point (x, y) where x and y given by :-

$$x = u*x_1 + v*x_2 + w*x_3$$

$$y = u*y_1 + v*y_2 + w*y_3$$

will lie inside the given triangle if and only if the following two conditions fulfil :-

- (1) $u > 0, v > 0, w > 0$ and
- (2) $u + v + w < 1$

The prove of this logic is to do by using vector analysis.

The logic stated by me above has been proved in this link.

3.2 Plotting 2D Histogram of Triangle sample

I had write the algorithm for plotting 2D histogram in file q1_d.py.

Firstly, i had generate set of $N=10^7$ sample random 2D points inside the given Triangle with $(0, 0)$, $(\pi, 0)$, $(\pi/3, \exp(1))$ as three vertices of triangle.

And then just use the hist2 function in python to plot the 2D histogram plots of the obtained sample of points.

I had use (bins=500) for approximating the plot to better extents.

3.2.1 q1_d.py Triangle 2D histogram plot

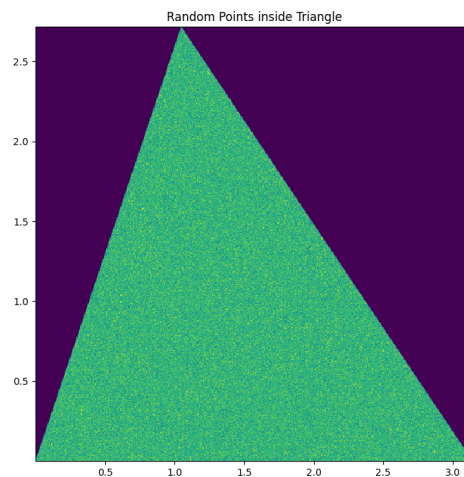


Figure 1: Triangle 2D histogram plot

Thanks