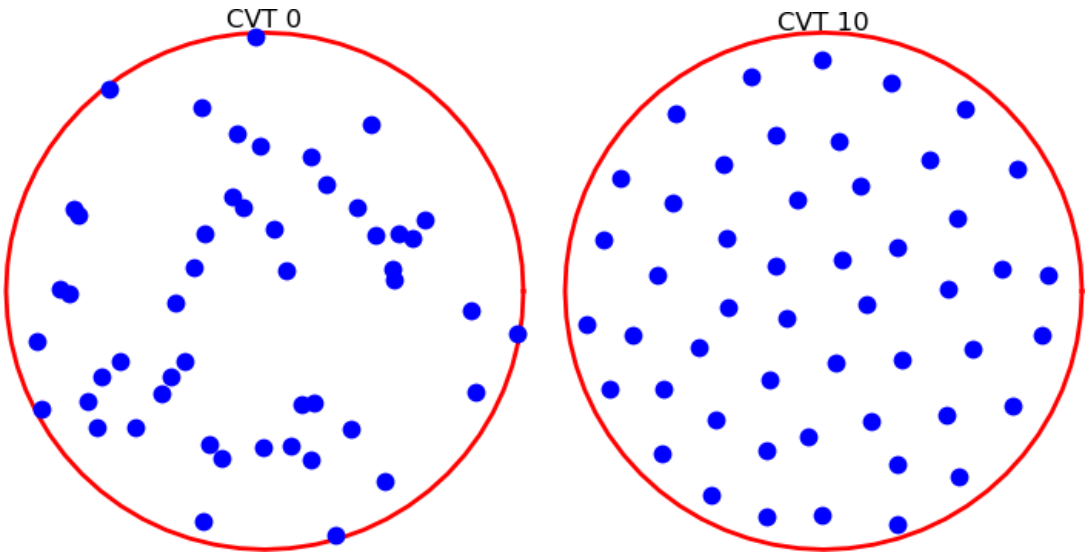
Project: Uniform Point Scattering
Mathematical Programming with Python

MATH 2604: Advanced Scientific Computing 4
Spring 2025
Monday/Wednesday/Friday, 1:00-1:50pm

https://people.sc.fsu.edu/~jburkardt/classes/python_2025/scattering/scattering.pdf

---



Fifty random starting points after improvement.

1 The Scattering problem

Scattering is an important problem in geometry is how to evenly distribute a set of points inside a given shape, which might be a square or circle, a polygon, an ellipse, or an irregular shape such as the outline of a body. A good method for doing this mainly requires the ability to produce as many random sample points from the region as we need. In this project, you will start with a bad arrangement of points inside a circle, and try to transform them into a neat pattern.

2 Lloyd's Algorithm

Suppose we have some region R in the plane, and we want to position a certain number of points within the region, which are "reasonably" separated. For a square, this might seem an easy task, but only if the number of points is itself a square; we think we know what to do with 9 points, but how would we arrange 10 or 11 points evenly? And the question becomes much harder if the region is an irregular polygon or has a curved boundary, or perhaps even internal holes.

One method to handle this problem uses Lloyd's algorithm. To use it, we assume that, at any time, we can easily request a given number of points in the region, which are chosen uniformly at random. We discussed in class how to do that for the circle, and so in this project that is the region we will consider.

To evenly scatter n points using Lloyd's algorthm, we start by placing the points anywhere in the region. The simplest way is simply to ask for n random points. This is step 0.

Now we apply a sequence of smoothing steps. On each step, we request another batch of sample points, usually many more than n. If n is 50, for example, we might ask for 1000 new sample points. Now we replace each of the orinal points by the average of itself and all the new sample points that are closest to it. That's the whole step. It may take a few moments to understand what this means. For instance, in the averaging step, we have to count how many nearby points are included in the average.

By repeating this smoothing step, the clumps and clusters and gaps in the original point arrangement should gradually disappear.

3 Your Project

Your project would be to write a Python program that does this operation for the circle. It would be reasonable to use 50 points, and 1000 sample points on each step, and maybe 10 or 20 smoothing steps. Your report should include before and after pictures.

You may find some more information in the Wikipedia article on Lloyd's algorithm.