

As discussed in class, the cluster assignment is meant to exercise your testing skills, which include your skills in designing classes that can be easily tested.

There are two important ways that you need to test your code. First, you need to write unit tests for the classes. Second, you need to test the solution itself. How can you show that your code is actually working?

Remember my tests of the code for the summer assignment. I tested not only the expected results but also important boundary conditions.

We briefly mentioned the obvious classes - Point, Centroid, Cluster - that would be part of any solution. Your two algorithms should also be separate classes - KMeans and KMeans2 - able to work on any problem set.

KMeans2 is the twist that I talked about. Instead of points seeking the nearest centroids to form a cluster, centroids will seek the nearest N points. Does this even work? What's the strategy for testing something that might not actually work? This is a common problem when writing research code.

To test the solution, you will need to come up with a metric to represent the goodness of a solution. You could call that class Fitness.

Can you find the code for this assignment on the Internet? Yes, but not with the proper object oriented design and a good set of unit tests that matches the approach we discussed.

As well, I am asking you to make some modifications - the “twist” - that will require you to write not just functional code, but code that is well designed and re-usable. Given the small number of classes in this assignment, it will probably be easier to write the code yourself than to find, modify, and test code that you find on the Internet. Nevertheless, as long as you meet the requirements of the assignment, you are welcome to choose either approach.

When you hand in your assignment, name it `CLUSTER_<studentId>_<version>.tar.gz` (or whatever extension your compression software is using).

This assignment is due on Thursday, the 28th of September.