Applied Math Methods: K-Means Clustering

Aditya Moger

August 22, 2023

1 Introduction

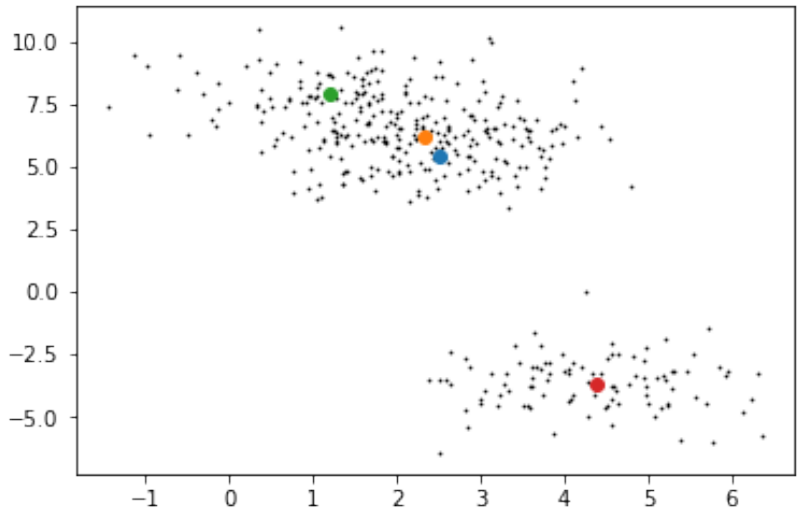
Most of the explanation for thought process and code for the clustering algorithm is written in the notebook. This document contains things that I could not include in the notebook.

2 Limitations of K-Means

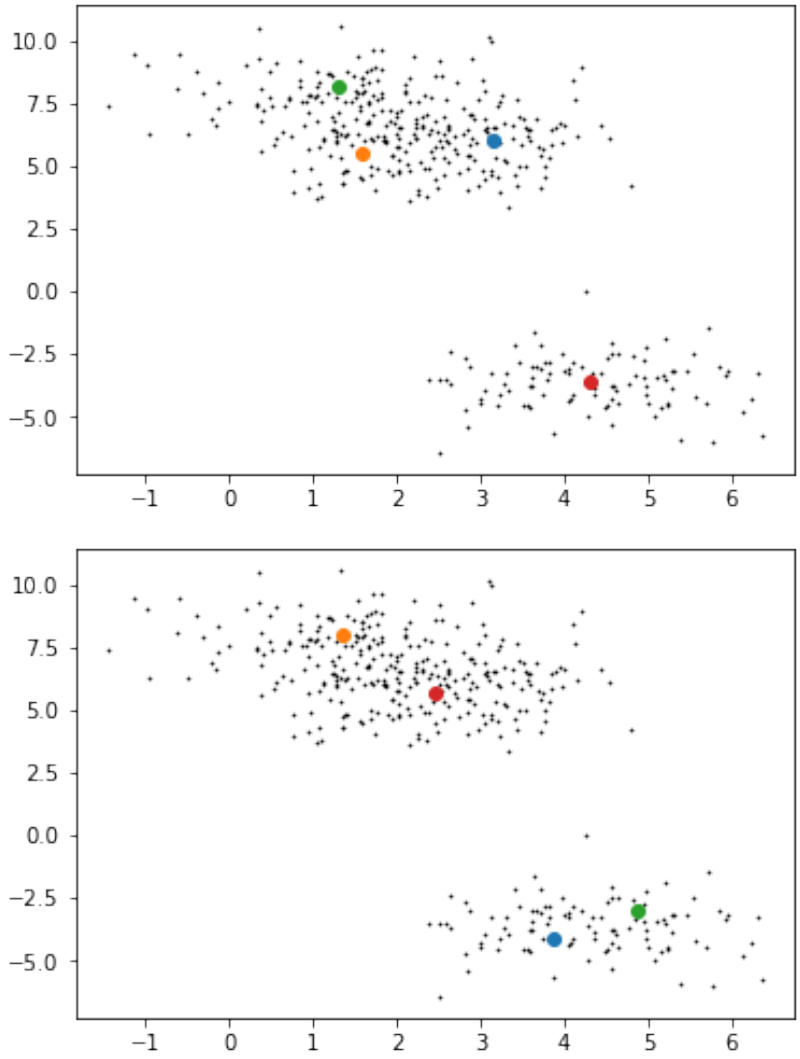
K-Means Clustering heavily relies on two things. One, the choice of initial centers and two, the geometric assumption that clusters are balls (circular in our 2D case). Another drawback is that all points are assumed to be part of clusters, which means that outliers or things that are clearly not identical to a cluster will be assigned to the closest cluster even though it has not physical relevance.

2.1 Geometric limitation

K-Means fails when clusters aren't circular or when two or more clusters overlap.
Following is an example of when two or more clusters overlap:
Initiation clusters (generated by the code given to us)



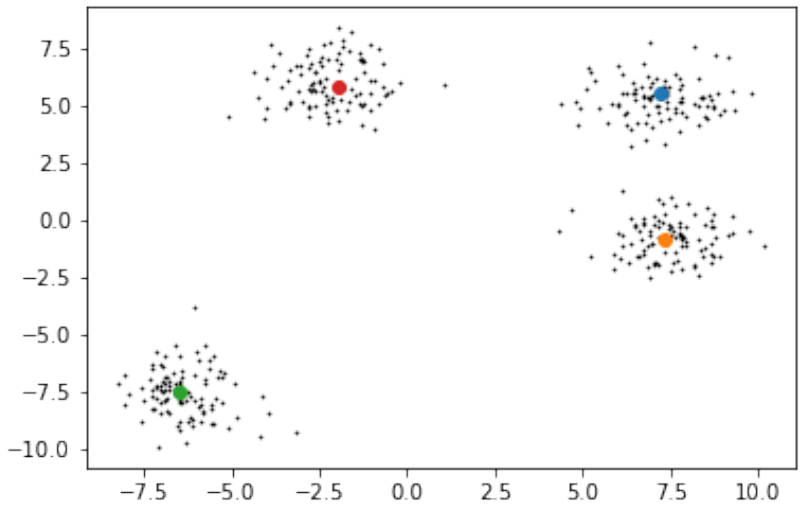
The clustering solution of K-Means algorithm :



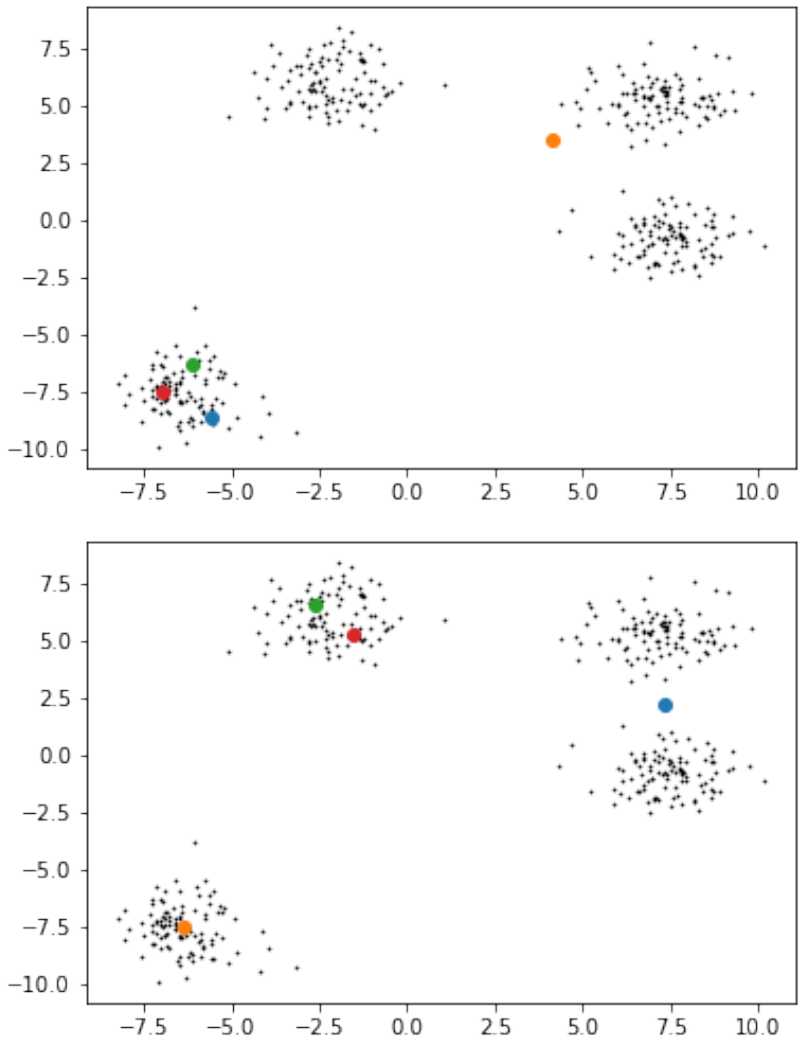
Here we get two potential solutions, which are accurate according to the algorithm but have severely different implications when using these clusters centers to infer things about the given data. In this case, one potential mis-interpretation is that in solution 1 it appears grouped on one side and one outlier cluster and solution 2 is a more even spread.
Another thing is that, the clusters generated by the code aren't circular but K-Means is forcing a circular fit. This is why we see a difference in the "expected" centers from given code and the centers generated by K-Means.

2.2 Issues with initiation of centers

Another major issues lies in initiation of centers. K-Means requires us to know beforehand how many centers are needed and the entire algorithm fails if the initial choice of centers is too close or nuanced, which I will highlight here.
Here's a case:
Initiation clusters (generated by the code given to us):



The clustering solution of K-Means algorithm :



As you can clearly see there are a lot of issues in the centers chosen here. This is due to bad choice of initial points, one due to initial centers being to close another due to initial centers not having enough points around them. I have suggested a solution in the notebook for how to make sure the initial center is not "alone", i.e., is at least surrounded by a threshold number of centers

2.3 Potential solutions to the issues

1. We can devise a scoring system based on number of points around a point. This is a whole new cluster center solving approach. We decide a threshold distance from a point and then score it based on the number of points within that threshold distance from it. We then get some sort of a heat map, where clusters are bordered by low score points and contain high score points inside.

2. Build up on the above, we can now pair K-means with this scoring method. Instead of choosing random points we allow it to choose only between points that have the higher score. this solves issues caused by both bad random centers and number of centers.

3. Some other methods are mentioned in the code.