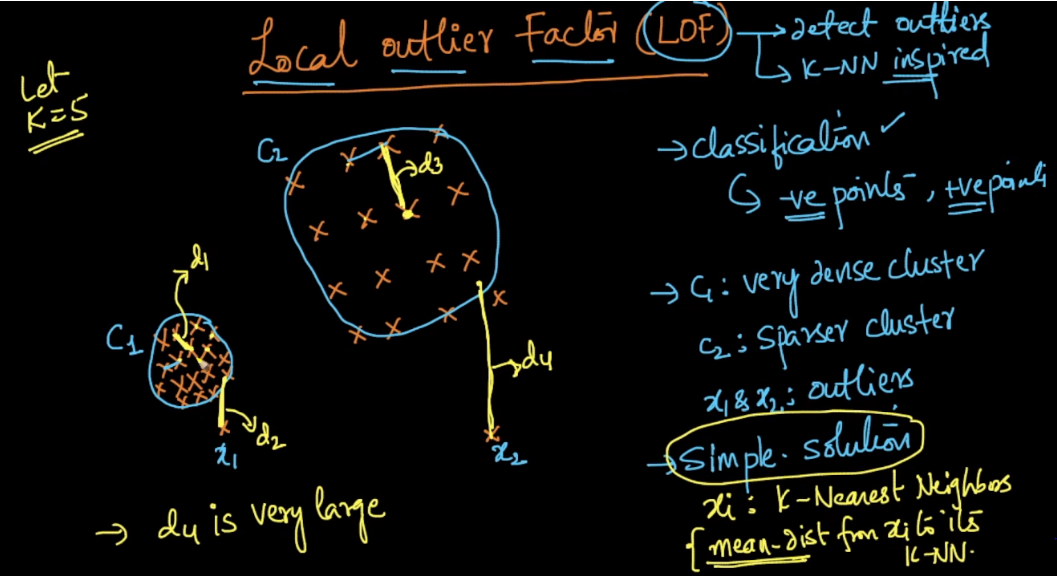
**Local Outlier Factor (LOF):**

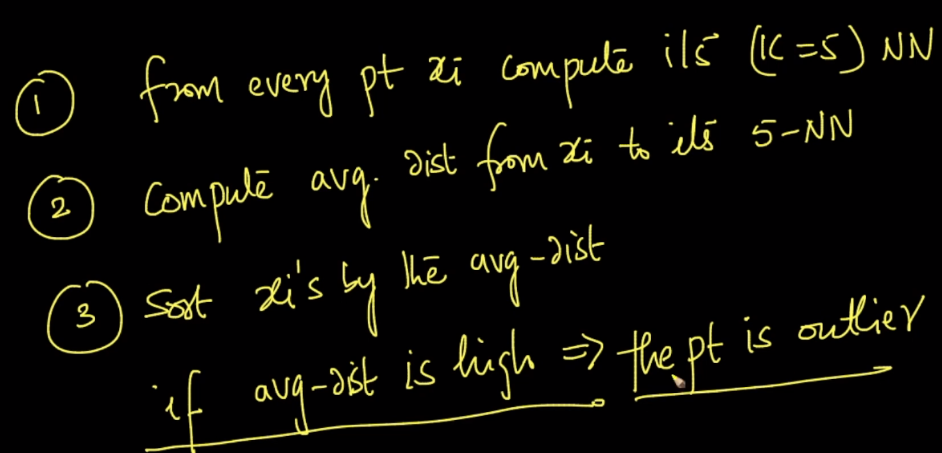
In this technique we use k-NN to detect outliers as:

Let’s say there are two clusters, one is very dense and one is very sparse, and there are 2 outliers.

So procedure to detect these outliers is as follows:

1. From every point xi , compute it’s k nearest neighbors let say k = 5.
2. Compute average distance from xi to it’s 5 nearest neighbors, or find distance of xi from all 5 NN and take mean of it.
3. Then sort all xi’s based on average distance calculated above.
4. If average distance is high then we call that point as outlier.



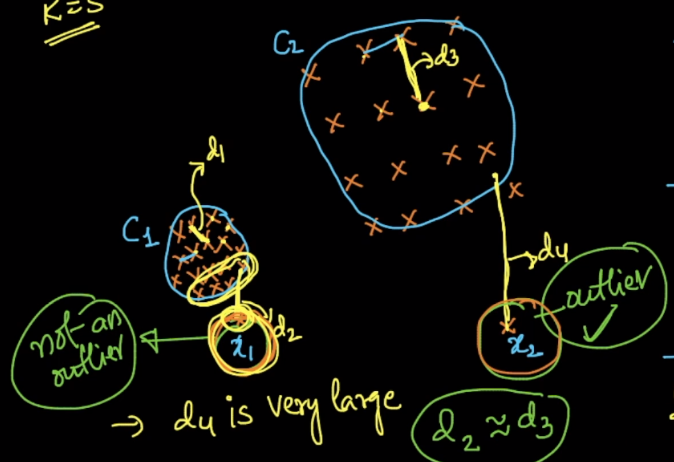


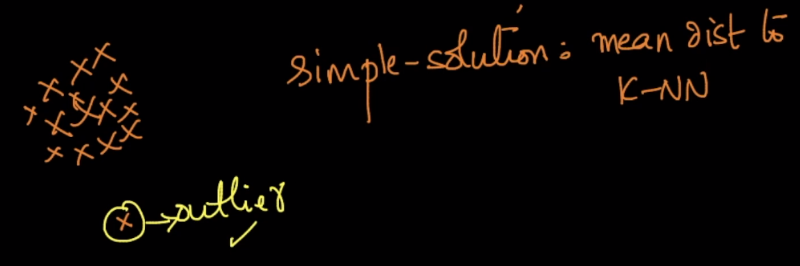
This was the simplest way of detecting outlier, but there is **problem** with this technique as:

It is able to correctly identify x2 as outlier since it’s avg distance d4 is very high,

But since distance of x1 from it’s 5 NN ie d2 is smaller or equal to than d3, therefore now x1 will not be treated as outlier, But it should be treated as outlier as:

If let’s say we had only one cluster c1, then this technique would easily able to detect x1 as outlier, but it’s not working fine for complex problem as we are discussing.





**Comments:**

* Is there any threshold value of avg distance, that any distance greater than this would be treated as outlier.

No threshold. That depends on our data and we have to place a threshold.

* Why don't we separate the clusters using their label names and then find out the avg K NN distance separately for each of those clusters?

That is an option if our dataset contains class labels. But, note that we may have to find outliers in data with no class-labels and with just data-points. What we explained here is a more general case when we have no class-labels. If we do have class labels, we can certainly process points belonging to each class separately as you suggested.