K modes

What if our data is non-numerical?

The basic concept of k-means stands on Euclidian distances. But what if our data is non-numerical or in other words categorical? Imagine to have the ID code and date of birth of the five people of the previous example instead of their heights and weights. We could think of transforming our categorical values in numerical values and eventually apply k-means. But beware k-means uses numerical distances, so it could consider close two really distant objects that merely have been assigned two close numbers.

In the popular K-modes algorithm distance is measured by the number of common categorical attributes shared by the two data points. The Modified k-modes algorithm will replace the means with the modes of the clusters by using a simple matching dissimilarity measure for categorical data. Instead of calculating centroids we calculate cluster mode vectors

The distance metric used for K-modes is instead the Hamming distance from information theory. The Hamming distance (or dissimilarity) between two rows is simply the number of columns where the two rows differ. The k-modes algorithm tries to minimize the sum of within-cluster Hamming distance from the mode of that cluster summed over all clusters. What is the mode of a cluster? if a dataset has m categorical attributes, the mode vector Z consists of m categorical values each being the mode of an attribute. In our skills example, the mode of an attribute is either 1 or 0 whichever is more common in the cluster. In general, each category could take on three or more values and Hamming distance would still apply.

That k-modes has not been more widely adopted at least in Python and the reason is probably related to the lack of a scikit-learn implementation. The Python k-modes library that can be used is called kmodes and it works analogously to scikit-learn's k-means construct.

Steps

1. Number of clusters (k) is chosen

2. Cluster-mode vectors are chosen at random. The popular approaches are Huang and Cao approaches.

3. Observations are allocated to the closest cluster mode by Hamming distance.

4. New cluster modes are calculated and observations are re-allocated to the closest cluster mode

It halts creating and optimizing clusters when either:

- It has stabilized - there is no change in their values because the clustering has been successful

- The defined number of iterations has been achieved.