## Assumption of K-Means Clustering- Part 2

In the last article, we talked about the case where we can't assume there is a fixed number of clusters in our dataset and also, the complexity of our model to grow with the size of the data.

In this article, we will discuss the cases where clustering doesn't quite represent the hidden patterns that we want to find in our data.

Let's take an example:

Suppose we have a website where people post pictures of their pets.
Now we want to cluster these images. We would like our cluster algorithm to put all of the images of cats in one cluster, all dogs images in another cluster and all the images of lizards in another cluster.Like the figure in below

Figure: 2

One way we could represent this clustering is in a table or a matrix where each row of the table corresponds to one of our images. We will put a 1 in the column that corresponds to the cluster for an image. In the figure, picture 1,2 and 7

belong to the same cluster. Picture 3 and 5 belong to the same cluster and picture 4 is in its own cluster.

But one issue here is,

What if someone posts a picture of both a cat and dog in a single picture.

Also, if someone posts a picture, that doesn't have any animals?

The problem with clustering is that each data point has to belong to only one cluster. There may be uncertainty about which group that is, for some data points.

But ultimately we believe the ground truth, that is there is exactly one true group that the data points belong to.

Now again consider our website and people's pet pictures. There we might want our data points(the pictures) to belong to multiple groups simultaneously or no groups or just one group.
Any of the above options will be allowed.

FEATURES

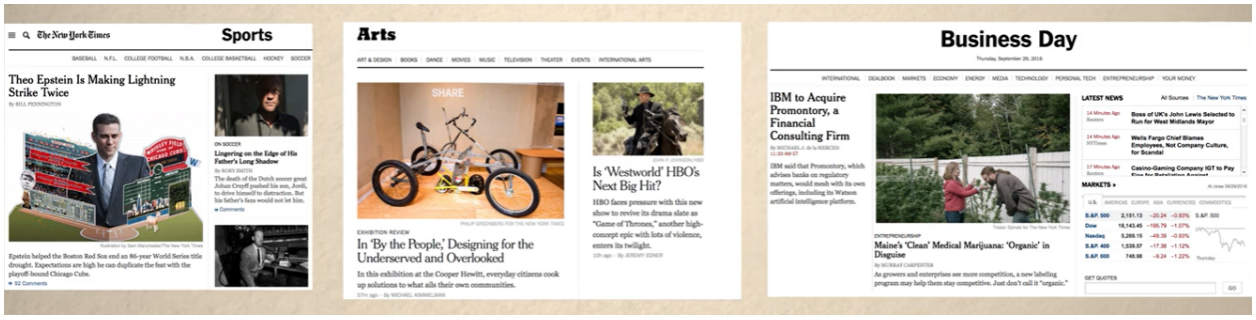
| | CAT | DOG | LIZARD |
|---|---|---|---|
|  | 1 | 0 | 0 |
|  | 1 | 1 | 0 |
|  | 0 | 0 | 0 |

Here, we call the groups **features** instead of clusters and the underlying structure is a **Feature Allocation** instead of a clustering. This **"Feature"** is different from what we have discussed in previous articles.


A similar idea is to say that, the data points exhibit **Mixed Membership**. They can belong to multiple groups at the same time unlike clustering. Sometime clustering is called the **Mixture Model** since each data point comes from one of a bunch of different groups. The bunch of different groups forms the mixture.

Then the case where each data point can belong to multiple groups at the same time is called the **Admixture Model**. Here all the three terms **"Feature Allocation", "Mixed Membership", "Admixture"** capture the idea that **data points** can belong to **multiple groups simultaneously**.

We see this type of example in lots of different data. Suppose we are analyzing a corpus of documents and we want to find the topics or themes that occur, say we have looked at all of the documents and past issues of the New York Times. So, some natural topics that we might find include sports, arts and economics.



Then suppose we read a review of the movie “Moneyball” based on the book by Michael Lewis. Here, the topic can be classified as arts because it’s a movie review. The review is also about sports because the movie is about baseball players. And the review is also about economics, because the movie is about using analytics and statistics to trade players and choose the best players.

So, if we think of a document as a data point, then here we want our data points to be able to belong to multiple groups.

Similarly if we study genomics, we often see that an individual’s DNA may be composed of parts from a number of different ancestral groups.

If we study social networks, we may see that an individual’s interactions on a social network represent various different personal identities such as work identity, family identity, and a social identity separate from the first two identities.
There are numbers of different models and algorithms that let us go beyond clustering and capture this kind of mixed membership structure in data.
The most popular among this kind of algorithm is **Latent Dirichlet Allocation(LDA)** .

LDA was originally designed for text data and it became popular due to the rise of massive amounts of text data available online. But it could be applied much more widely to other types of categorical data including genetics data.

The other K-Means algorithm for the Feature Allocation problem has been derived using MAD-BAYES. For example, DP-Means and BP-Means algorithms have been used to study tumor hydrogenation. In many tumors multiple different types of cancers are present and it’s important to identify all of the different types of cancer, to design effective treatments.