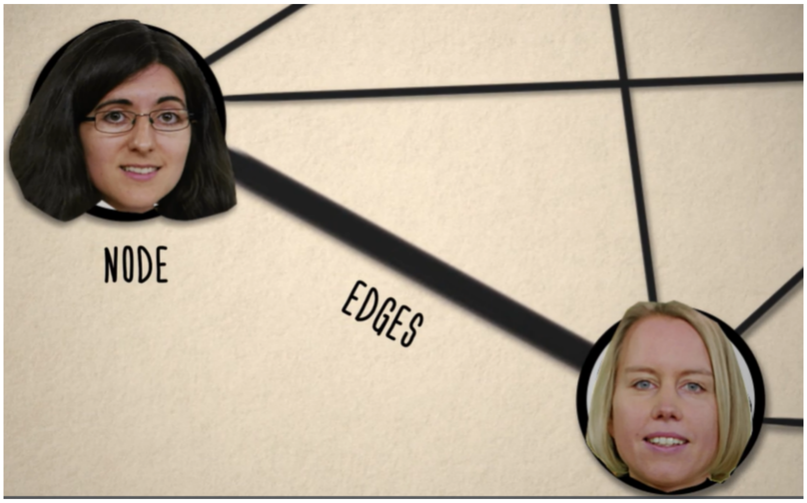
Clustering in Graph and Networks

In the previous module, our data was described by a set of features, but in graphs, we don’t have that flexibility, and each data point is represented by nodes and we don't have features, and we only know who is connected to whom.

There can be different examples of networks like social media networks, Web pages network, biological networks, etc. Each data point in the graph is represented as a **Node** and the connection between any 2 nodes is called an **Edge**.

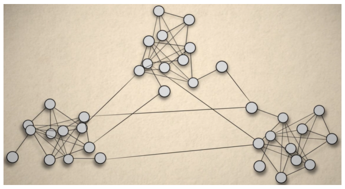


We can have weight on these edges, the larger the weight the stronger the relationship between the 2 nodes.

We can interpret this relationship only by looking at the small graphs. Working and visualizing a large network, such as the biological network is a very difficult task.

The first thing to understand about the network is the community structure of the network, and are there any dense groups/clusters of friends in the network?, etc. Thus we can learn many things using cluster analysis in graphs.

For example - Social networks help us to understand how epidemics spread on the social media network? It can also help us to answer the question of how trends and opinions build in a social network, in biology clusters of proteins indicate the functionality of the body, etc.

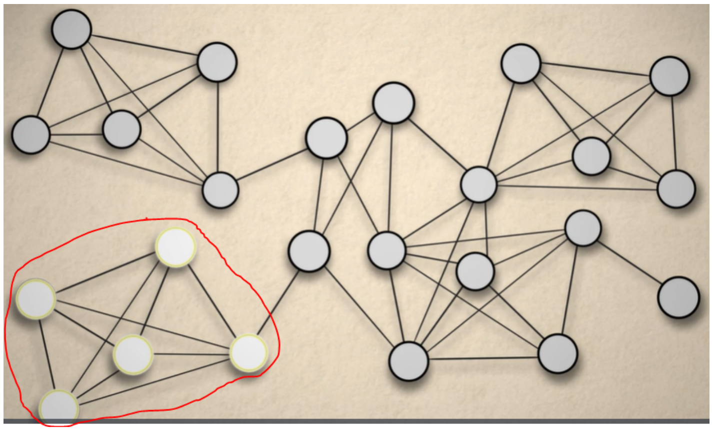


So the formal question now is what is a cluster in a graph?

There are many definitions for clusters in the graph, but they all point to an intuitive idea that *a* cluster consists of points that are well connected with each other, that is, there are a lot of edges between them. The number of edges within a cluster is also called the volume.

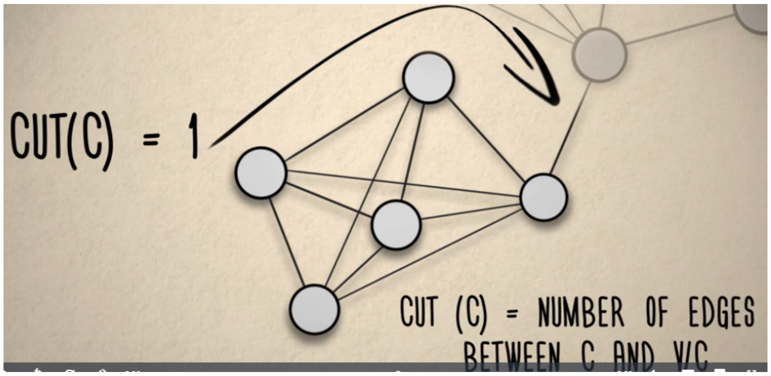
The 1st criteria to define a good cluster is, defined volume per node as density, therefore for a good cluster, we want this density to be large.

The 2nd criterion is that there should be less number of edges between different clusters.



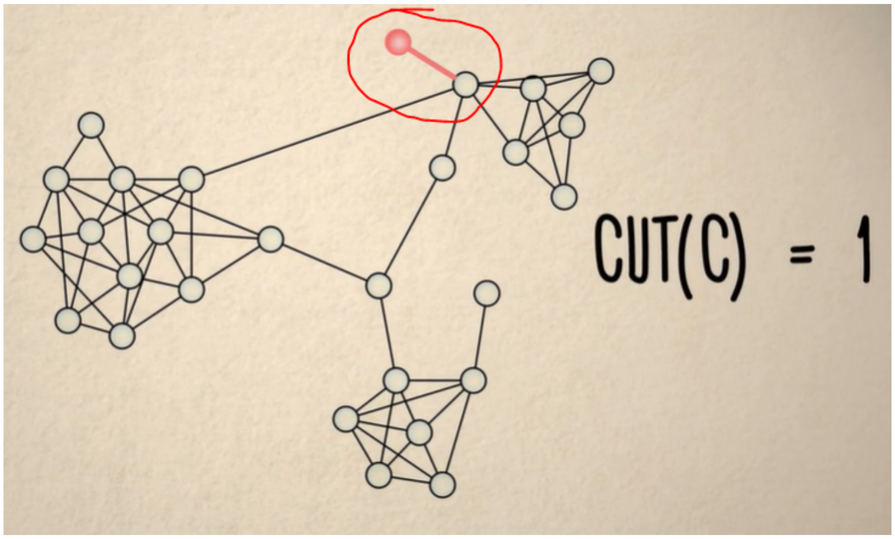
The separation between the clusters can be measured using a cut value. If we want to cut the cluster from the graph we would need to disconnect the edges between the clusters in the graph.

So, we define Cut(C) = number of edges we need to cut to separate the cluster from the result of the graph.



Therefore, logically one criterion could be, having a low cut value, as it separates the cluster from the rest of the graph easily.

But is it the right criteria? Consider an outlier node in the below image, it has a cut value of 1 but is not forming a cluster.

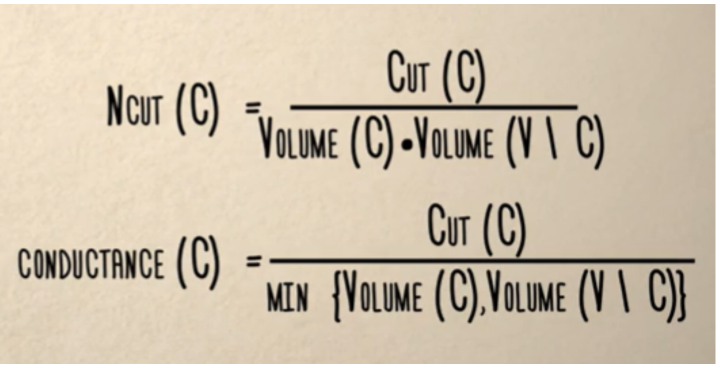


Therefore, choosing a cut value should be a balance of the size of the cluster and separation.

Examples of such, **combined criteria** are:

1. **Conductance**
2. **Normalized cut**

They both divide cut by a measure of volume, and we want to minimize the criteria.
The formula for both the criteria are given below:



$$\mathrm{Ncut}(C) = \frac{\mathrm{Cut}(C)}{\mathrm{Volume}(C) \cdot \mathrm{Volume}(V \setminus C)}$$

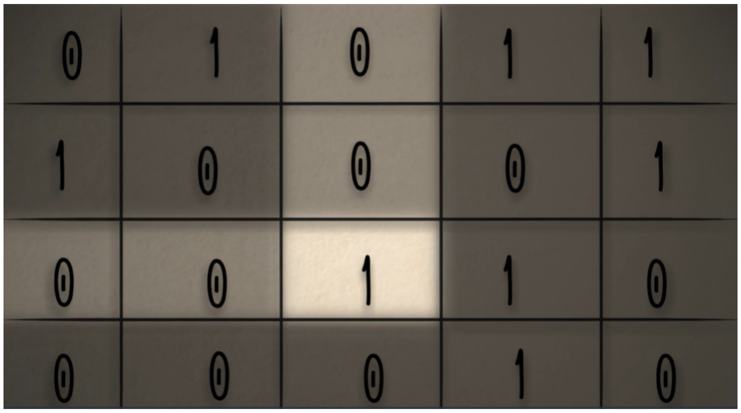
$$\mathrm{Conductance}(C) = \frac{\mathrm{Cut}(C)}{\min\{\mathrm{Volume}(C), \mathrm{Volume}(V \setminus C)\}}$$

In both the formulas, if the volume is small, then the denominator is small and the criterion is large, and vice versa.

Therefore, using this criterion, good clusters are not too small, internally well connected, and are well separated from the rest of the nodes. Clusters can also be seen in the **adjacency matrix.**

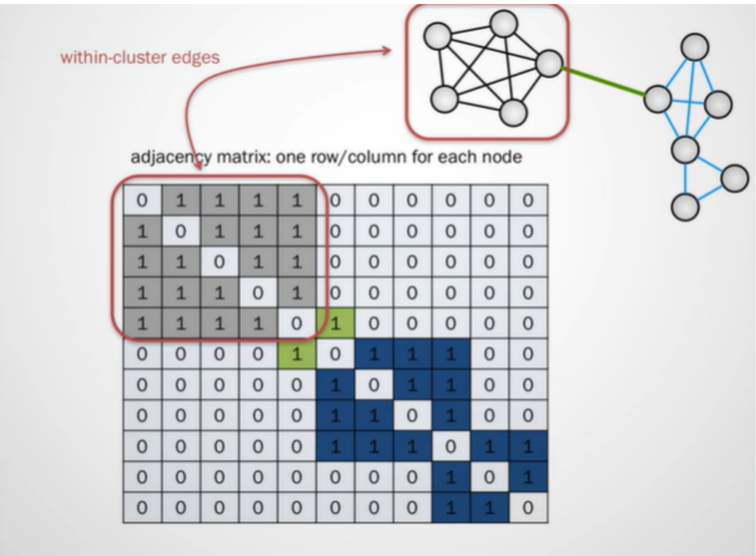
What is an adjacency matrix?

An adjacency matrix is a NxN matrix, where N defines the number of nodes in a graph. If there is an edge between 2 nodes, then the corresponding value in the matrix is 1 else 0.



0	1	0	1	1
1	0	0	0	1
0	0	1	1	0
0	0	0	1	0

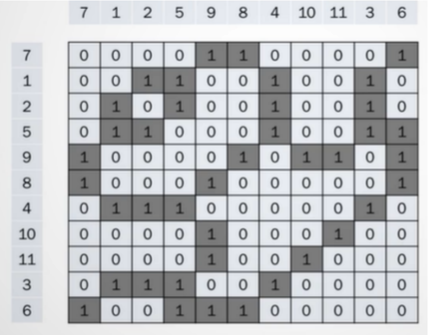
If you know the browns and columns by the cluster, then the matrix is a block-like structure and we can easily see clusters in it.



adjacency matrix: one row/column for each node

0	1	1	1	1	0	0	0	0	0	0
1	0	1	1	1	0	0	0	0	0	0
1	1	0	1	1	0	0	0	0	0	0
1	1	1	0	1	0	0	0	0	0	0
1	1	1	1	0	1	0	0	0	0	0
0	0	0	0	1	0	1	1	1	0	0
0	0	0	0	0	1	0	1	1	0	0
0	0	0	0	0	1	1	0	1	0	0
0	0	0	0	0	1	1	1	0	1	1
0	0	0	0	0	0	0	0	1	0	1
0	0	0	0	0	0	0	0	1	1	0

But typically we don't know the cluster and node structure is shuffled arbitrarily, and the structure looks like this as shown below.



	7	1	2	5	9	8	4	10	11	3	6
7	0	0	0	0	1	1	0	0	0	0	1
1	0	0	1	1	0	0	1	0	0	1	0
2	0	1	0	1	0	0	1	0	0	1	0
5	0	1	1	0	0	0	1	0	0	1	1
9	1	0	0	0	0	1	0	1	1	0	1
8	1	0	0	0	1	0	0	0	0	0	1
4	0	1	1	1	0	0	0	0	0	1	0
10	0	0	0	0	1	0	0	0	1	0	0
11	0	0	0	0	1	0	0	1	0	0	0
3	0	1	1	1	0	0	1	0	0	0	0
6	1	0	0	1	1	1	0	0	0	0	0

Apart from all these criteria, there are many other criteria for clustering, one of them is **Modularity clustering**. It helps us to find how high the density of an edge within clusters is compared to a baseline graph where edges occur randomly.

Also, when the network is very large, we don’t have to look at the entire network, just a part of the network.

Other criteria are correlation clustering, information of the pair of points, if they are similar they should be in the same clusters and we can use different similarity measures to find the similarity between the 2 nodes.