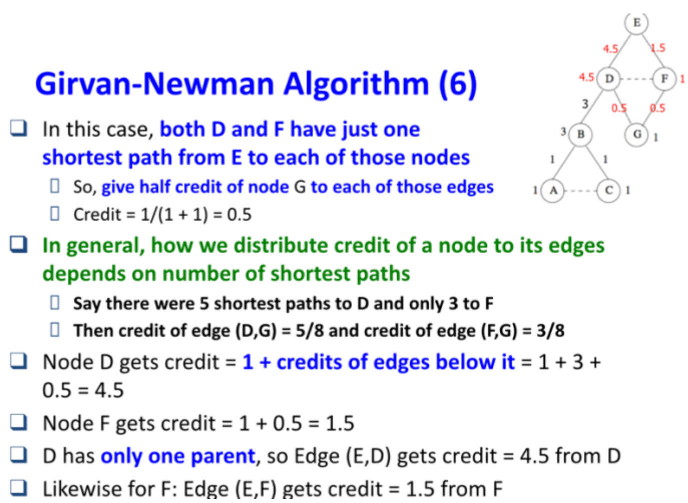
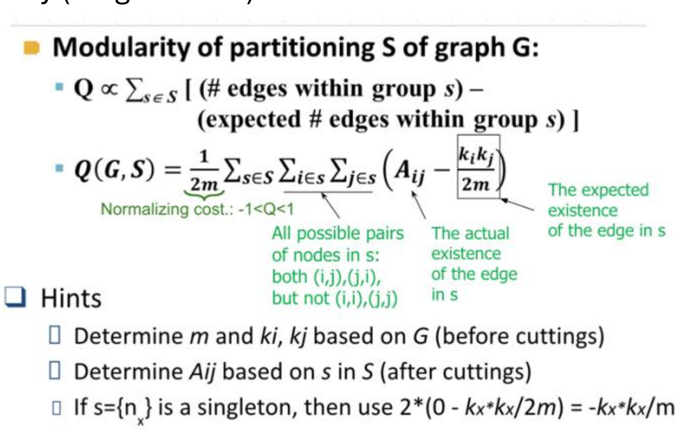
Community detection:

1. Clustering by edge betweenness
    a. Edge betweenness: number of shortest passing over the edge.
        i. High means an edge is important link connect groups
        ii. Low is not important pass
    b. The girvan-newman algorithm:
        i. Computer edge betweenness
        ii. Remove the edge with the highest betweenness
        iii. Recalculate edge betweenness
        iv. Repeat the process until the network is splits into meaningful communities

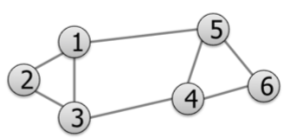


    c. Con for girvan-newman is computationally expensive for large graphs.
2. Define Modularity Q (higher is better)
    a. A measure of how well a network is partitioned into communities
    b. Given a partition of the network into groups s in S
    c. Q = Sum((number edges in groups s)-(expected number edges in s))
3. Modularity (range: -1 to 1):
    a. 

4. Spectral clustering (graph cut)
    a. Partitioning a graph to minimize the number of edges that connect different communities.
    b. Goal: minimize the cut size but smallest group size is not always best
        i. Divide nodes into two sets so that the cut (set of edges that connect nodes in different sets) is minimized
        ii. Want the two sets to be approximately equal in size
        iii. Maximize the number of within-group connections
        iv. Minimize the number of between-group connections
    c. Normalized cut;
        i. $$ncut(A,B) = \frac{cut(A,B)}{vol(A)} + \frac{cut(A,B)}{vol(B)}$$
        ii. Vol(t) = edges (count even after edge cut)
        iii. Good cut= 0 Strongly connected groups with minimal between group edges
        iv. Bad cut = 1. Weak Separation many cross group connections
    d. Matrix:

Adjacency matrix (A):

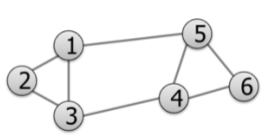
- $n \times n$ matrix
- $A=[a_{ij}]$, $a_{ij}=1$ if edge between node i and j



	1	2	3	4	5	6
1	0	1	1	0	1	0
2	1	0	1	0	0	0
3	1	1	0	1	0	0
4	0	0	1	0	1	1
5	1	0	0	1	0	1
6	0	0	0	1	1	0

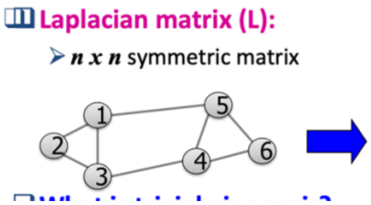
Degree matrix (D):

- $n \times n$ diagonal matrix
- $D=[d_{ii}]$, d_{ii} = degree of node i



	1	2	3	4	5	6
1	3	0	0	0	0	0
2	0	2	0	0	0	0
3	0	0	3	0	0	0
4	0	0	0	3	0	0
5	0	0	0	0	3	0
6	0	0	0	0	0	2

    e. Laplase matrix
        i. row sum = 0
        ii. Symmetric



	1	2	3	4	5	6
1	3	-1	-1	0	-1	0
2	-1	2	-1	0	0	0
3	-1	-1	3	-1	0	0
4	0	0	-1	3	-1	-1
5	-1	0	0	-1	3	-1
6	0	0	0	-1	-1	2

(degree matrix – adjacency matrix)

5. Bypartite graph:
    a. A complete bipartitie graph: Contains all possible edges between a vertex of f and a vertex of c
6. Direct discovery