CLIQUES

Refer Chapter 7

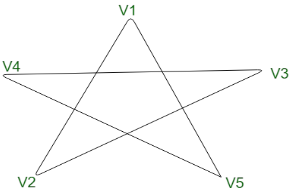
$G : (V, E)$

$G(S) : V', E'$

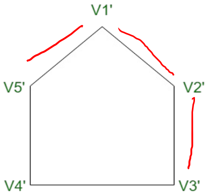
CLIQUE

- A clique in a (an undirected) graph G is a subgraph of G isomorphic to a complete graph
- A Clique is a subset of vertices such that each vertex is connected to every other one. Vertices and edges in a clique makes that subgraph a complete graph.

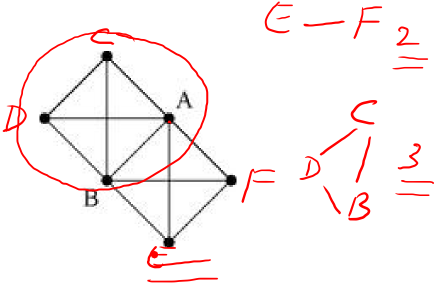
Isomorphism



G



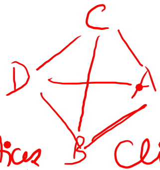
G'



NP - Complete

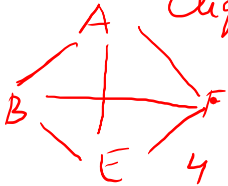
A graph w' n vertices

2^n subsets possible



Clique

K_4



Clique

CLIQUES

CLIQUES ..

- Cliques of order 1 are just single vertices.
- Cliques of order 2 are just pairs of vertices connected by an edge, so for simple undirected graphs the number of cliques of order 2 is just the number of edges.

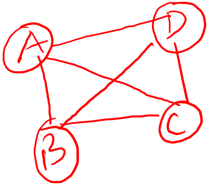
CLIQUES ..

- Cliques of order 3 are triangles in the graph

Example: K_4

- This has one clique of order 4 and 4 of order 3, 6 of order 2 and 4 of order 1

- The clique number of G is the maximum order of a clique in G.
  - For K_n , it is n.
  - For C_n it is 2 unless n = 3 (for $C_3 = 3$).



$\underline{K_4}$ → Complete graph w/ 4 vertices

Cliques of order 1 → A D C B

A subgraph w/ 1 vertex

How many cliques of order 1 = 4

$\underline{G}$

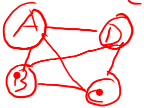
G is subgraph of G

Cliques of order $\underline{2}$ → A — D

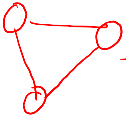
= 6 cliques

• Cliques of order 3 → Triangles = 4 cliques

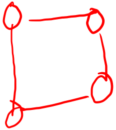
Cliques of order 4 →



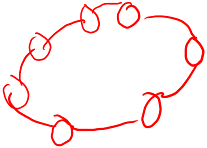
C_n – Circular graph



$\rightarrow 3$ Maximal clique order



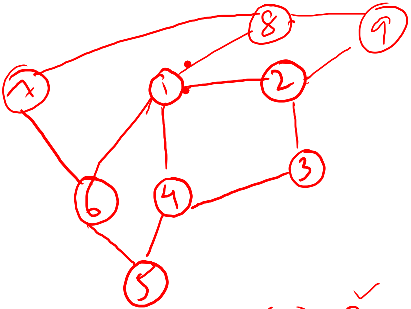
$\rightarrow 2$



$\rightarrow 2$

$C_n = 2$; if $n = 3$, 3

$n = 9$



Algorithm to find clique

1. Find a vertex v of the smallest possible degree in G.
2. If the degree of v is $n-1$, stop; G is a clique, so the largest clique in G has size n.
3. Otherwise, remove v and all of its edges from G. Find the largest clique in the smaller graph. Report that as the largest clique in G.

$Deg(1) = 4$
$(2) = 3$
$(3) = 2$ ✓
$(4) = 2$ ✓

$Deg(5) = 2$ ✓
$(6) = 3$
$(7) = 2$
$(8) = 3 \to 2$

$Deg(9) = 2$ ✓ — X

n = 8

n = 7

n = 6

n = 5

n = 4

n : 3

n = 2

Clique of order 2

Deg(1): 4 − 3 / 2 − 1

(2) = 3 − 2 − 1 X

(3) : 2 ✓ − X

(4) = 3 − 2

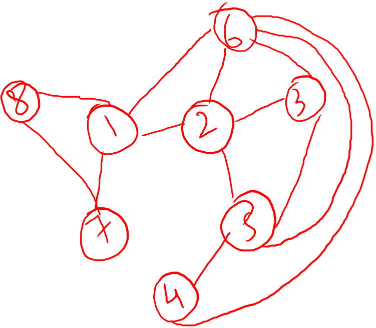
Deg(5): 2 − 1

(6) = 3 → 2 − X

(7) = 2 − X

(8) = 3 → 2 → 1 X

~~Deg(9): 2~~ − X



Deg(1) = 4 → 3 → 2 — X

Deg(2) = 4 → 3 ✓

Deg(3) = 3 ✓

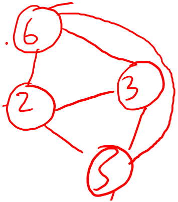
(4) = 2 → X

(5) = 4 → 3 ✓

(6) = 5 → 4 → 3 ✓

(7) = 2 → 1 → X

(8) = 2 → X



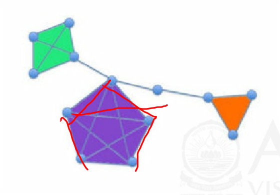
Clique of order 4

Find maximal clique.

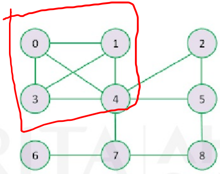
(1,2), (2,3), (3,1), (4,3), (4,1), (4,2) ✓

(1,2), (2,3), (3,1), (4,3), (4,5), (5,3) ✓

Cliques ..

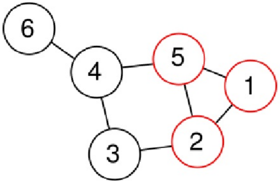


- Cliques → 2, 3, 4 and ~~6~~ 5
- Clique Number → ~~6~~ 5



- Cliques → 2, 3 and 4
- Clique Number → 4

Maximal order



- Cliques → 2 and 3
- Clique Number → 3

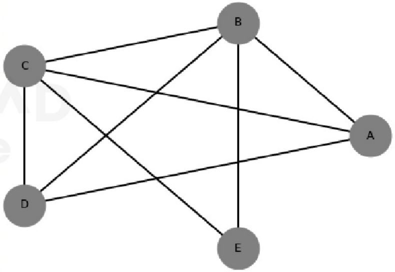
Finding Cliques ..

"Cliques" in NetworkX

- `find_cliques(G[, nodes])`: Returns all maximal cliques in an undirected graph
- `graph_clique_number(G[, cliques])`: Returns the clique number of the graph
- `cliques_containing_node(G[, nodes, cliques])`: Returns a list of cliques containing the given node
- `node_clique_number(G[, nodes, cliques, ...])`: Returns the size of the largest maximal clique containing each given node

CLIQUES USING NETWORKX: EXAMPLES

```
list(nx.find_cliques(G))
[['B', 'C', 'E'], ['B', 'C', 'A'],
 ['B', 'D', 'E']]
nx.graph_clique_number(G)
4
nx.cliques_containing_node(G)
{'A': [['C', 'B', 'A', 'D']],
 'B': [['C', 'B', 'E'],
       ['C', 'B', 'A', 'D']],
 'C': [['C', 'B', 'E'],
       ['C', 'B', 'A', 'D']],
 'D': [['C', 'B', 'A', 'D']],
 'E': [['C', 'B', 'E']]}
 nx.node_clique_number(G)
 {'A': 4, 'B': 4, 'C': 4, 'D': 4, 'E': 3}
```



K-plex

A k-plex of size n is a maximal subset of n vertices within a
network such that each vertex is connected to at least
n - k of the others.

If k = 1, -> ordinary clique

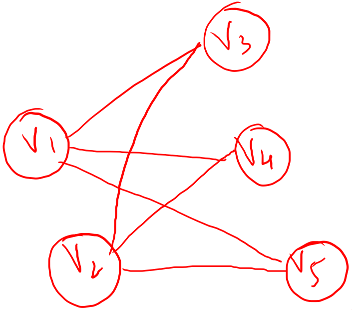
n-k = n-1 of the vertices

n-3

Useful concept for discovering groups within networks: in real life many groups
in social and other networks form k-plexes. There is no solid rule about what value k should take.

K-core

atleast k vertices

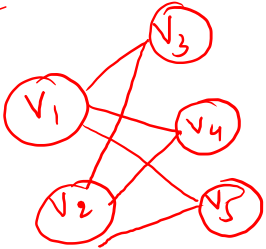


1-plex

X

k-plex ?

k = ?



3-plex.

5 vertices

3 →

$n - k$

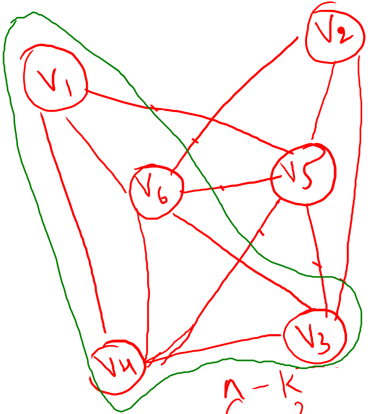
$k = 2$

3 vertices.

k-plex

$n - k = 2$

3-plex



n - K
6 - 3

k-plex?

Min deg(G) = ?

deg(v_1) = 3
(v_2) = 3
(v_3) = 4
(v_4) = 4
v_5 = 5
v_6 = 5

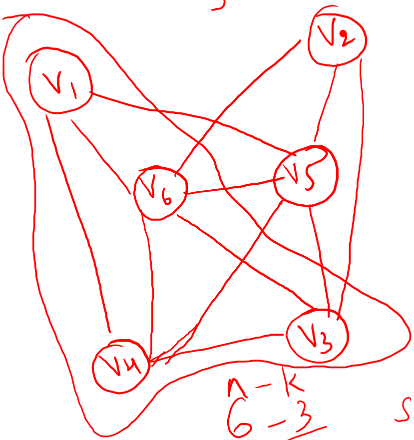
G'

→ 2-plex.

n = 5

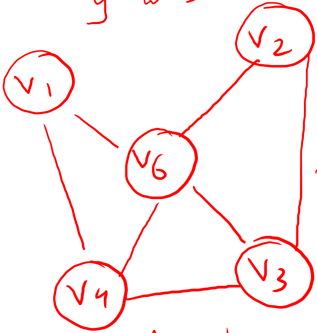
For G' to be 2-plex
deg(v') = n - 2 = 3

G



$n-k$
$6-3$

G' w/ 5 vertices.



X
2-plex

Is it 2-plex?

$n = 5$. Each vertex should be $n-K \Rightarrow n-2$ vertices connected to $\rightarrow$ 3 vertices

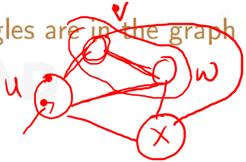
Objectives

Clustering in Networks

- Clustering Coefficient
- Transitivity

CLUSTERING (1/3)

- Cliques are unstable under minor changes, and hence this measure is not considered as robust.
  - *cohesiveness*: tightly-knit-groups
  - identifying cohesive subgraphs: Clustering
    - ∗ Visually this is the notion of how many triangles are in the graph and how much they bunch together.



DEFINITION (CLUSTERING COEFFICIENT)

Let $G = (V, E)$ be a simple undirected graph. The (individual) clustering coefficient of a node $u \in V$ is the probability $C\ell(u)$ that two neighbors of u are themselves neighbors

Clustering Coefficient

$$C\ell(u) = \frac{|\{(v,w) : v,w \in V; v \neq w; (u,v),(u,w),(v,w) \in E\}|}{|\{(v,w) : v,w \in V; v \neq w; (u,v),(u,w) \in E\}|}$$

Clustering (2/3)

Clustering Coefficient ..

$$C\ell(u) = \frac{|\{(v,w) : v,w \in V; v \neq w; (u,v),(u,w),(v,w) \in E\}|}{deg(u)\,(deg(u)-1)}$$

Clustering Coefficient: Simplified Form

$$C\ell(i) = \frac{2\,L_i}{k_i\,(k_i-1)}$$

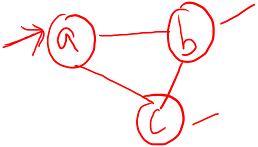
where, L_i is the number of edges between k_i neighbors, and k_i is the degree of node i.

Example

Friendship networks: measures how many of your friends are friends with each other

- The **average clustering coefficient** is the average of $Cl(u)$ over all $u \in V$

Clustering (3/3)



Overall Clustering Coefficient: Transitivity

$$C\ell(G) = \frac{\sum_{u \in V} |\{(v,w) : v,w \in V; v \neq w; (u,v),(u,w),(v,w) \in E\}|}{\sum_{u \in V} |\{(v,w) : v,w \in V; v \neq w; (u,v),(u,w) \in E\}|}$$

Transitivity Simplified

$$C\ell(G) = \frac{\textit{Number of Triangles X 3}}{\textit{Number of Connected Triplets}}$$

CLUSTERING COEFFICIENT: EXAMPLES (1/2)

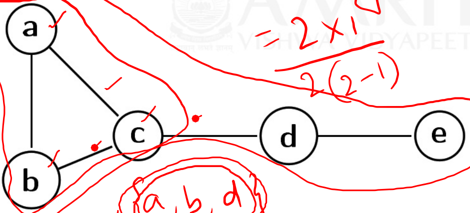
$$C\ell(i) = \frac{2\,L_i}{k_i\,(k_i-1)}$$

deg(i)

$$C\ell(G) = \frac{\textit{Number of Triangles X 3}}{\textit{Number of Connected Triplets}}$$

Cl(a) = 2 + no. of edges bet^n nbrs of a

$= \frac{2 \times 1}{2(2-1)}$



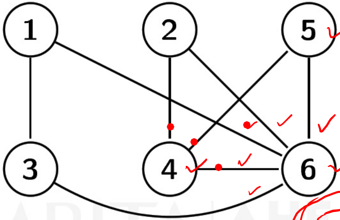
{a, b, d}

a-b

$C\ell(i)$ & $C\ell(G)$

- $C\ell(a) = \frac{2\times 1}{2(2-1)} = 1$
- $C\ell(b) = \frac{2\times 1}{2(2-1)} = 1$
- $C\ell(c) = \frac{2\times 1}{3(3-1)} = 1/3$
- $C\ell(d) = \frac{2\times 0}{1(1-1)} = 0$
- $C\ell(e) = \frac{2\times 0}{2(2-1)} = 0$
- $C\ell(G) = \frac{1\times 3}{6} = 1/2$

Clustering Coefficient: Examples (2/2)



nbrs C_2

$cl(6) = \frac{2L_i}{k_i(k_i-1)} = \frac{2*3}{5*4}$

5, 2, 1, 4, 3

(1,3), (5,4), (2,4),

$C\ell(G)$

- No. of Triangles: 3, No. of Triplets: 17
  - Node 1: $\binom{2}{2} = 1$
  - Node 2: $\binom{2}{2} = 1$
  - Node 3: $\binom{2}{2} = 1$
  - Node 4: $\binom{3}{2} = 3$
  - Node 5: $\binom{2}{2} = 1$
  - Node 6: $\binom{5}{2} = 10$
- $C\ell(G) = \frac{3\times 3}{17} = 9/17$

OBJECTIVES

Clustering using NetworkX

- Clustering Coefficient
- Transitivity
- Triangles

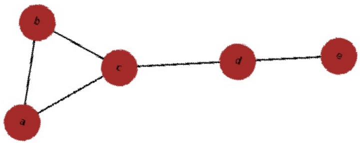
Functions for Clustering

In NetworkX

- clustering(G[, nodes): Compute the clustering coefficient for nodes
- average_clustering(G[, nodes]): Compute the average clustering coefficient for the graph G
- transitivity(G): Compute graph transitivity, the fraction of all possible triangles present in G.
- triangles(G[, nodes]): Compute the number of triangles.

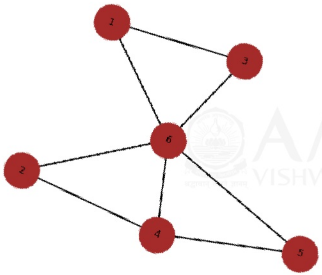
EXAMPLE 1

```python
import networkx as nx
import matplotlib . pyplot as plt
edge_widths = [2]
G2 = nx.Graph([('a', 'b'),('a', 'c'),('b', 'c'), ('c', 'd'), ('d', 'e')
nx . draw (G2 , with_labels = True, node_size=2000, node_color="brown",
width=edge_widths)
```




```python
print(nx.clustering(G2))
{'a': 1.0, 'b': 1.0,
 'c': 0.33, 'd': 0, 'e': 0}
nx.transitivity(G2)
0.5
```

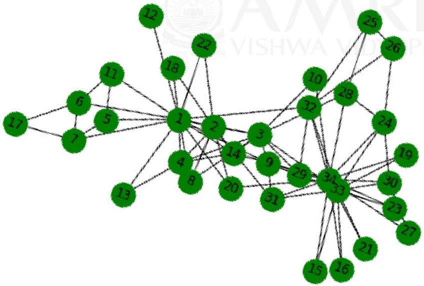
EXAMPLE 2




```
nx.transitivity(G2)
0.5294
print(nx.clustering(G2))
{1: 1.0, 3: 1.0, 6: 0.3,
 2: 1.0, 4: 0.66, 5: 1.0}
nx.average_clustering(G2)
0.8277
nx.triangles(G2)
{1: 1, 3: 1, 6: 3,
 2: 1, 4: 2, 5: 1}
```

Example 3

```
G2 = nx.karate_club_graph()
G2 = nx.convert_node_labels_to_integers(G,first_label=1)
```



```
nx.transitivity(G2)
0.2556818181818182
print(nx.clustering(G2))
{1: 0.15, 2: 0.33, 3: 0.24,..,
17: 1.0,.. 33: 0.196, 34: 0.110}
nx.average_clustering(G2)
0.5706384782076823
nx.triangles(G2,1)
18
```

Summary

NetworkX

- Clustering Coefficient
- Transitivity
- Triangles

References

Hagberg, A., Swart, P., and S Chult, D.
Exploring network structure, dynamics, and function using networkx.
Tech. rep., Los Alamos National Lab.(LANL), Los Alamos, NM (United States), 2008.

Newman, M.
Networks.
Oxford university press, 2018.

Zinoviev, D.
Complex network analysis in Python: Recognize-construct-visualize-analyze-interpret.
Pragmatic Bookshelf, 2018.