Big Data Technology

Paul Rad, Ph.D.

Associate Professor
Information Systems and Cyber Security, College of Business School
Electrical and Computer Engineering, College of Engineering

Outline

- **Logistics**
- **Review Graph Theory**
- **Graph Analytics**
  1. **Path Analytics**
  2. **Connectivity Analytics**
  3. **Community Analytics**
  4. **Centrality Analytics**

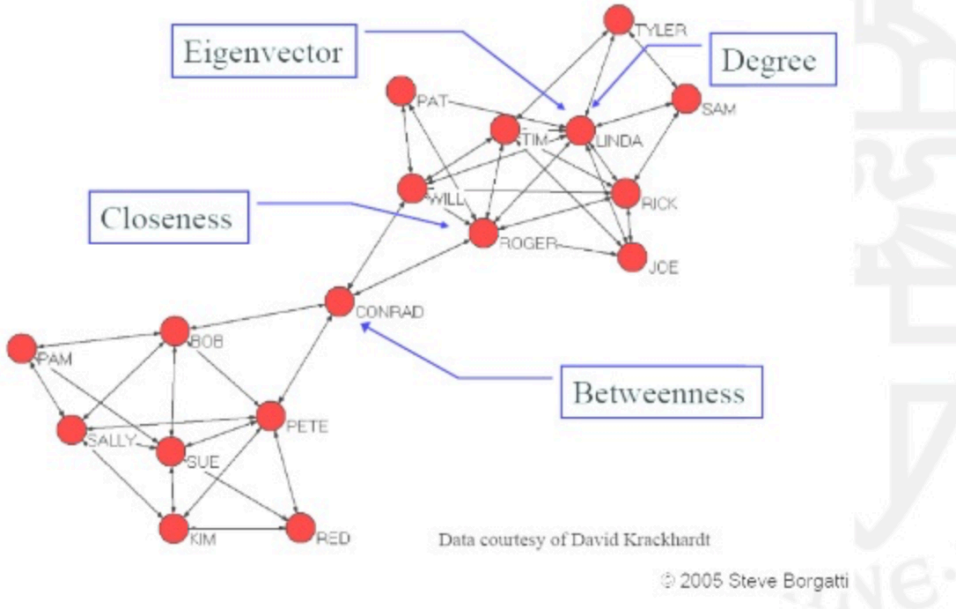
Centrality Measure = who is most important?

❖**Degree Centrality**: The number of edges connected to a node

❖ **Closeness Centrality**: The average of the shortest distance to all other nodes in the graph

❖**Betweenness Centrality**: Extent to which a particular node lies on the shortest path between other nodes

❖**Eigenvalue Centrality**: A measure of the extend to which a node is connected to influential other nodes



who is most important?

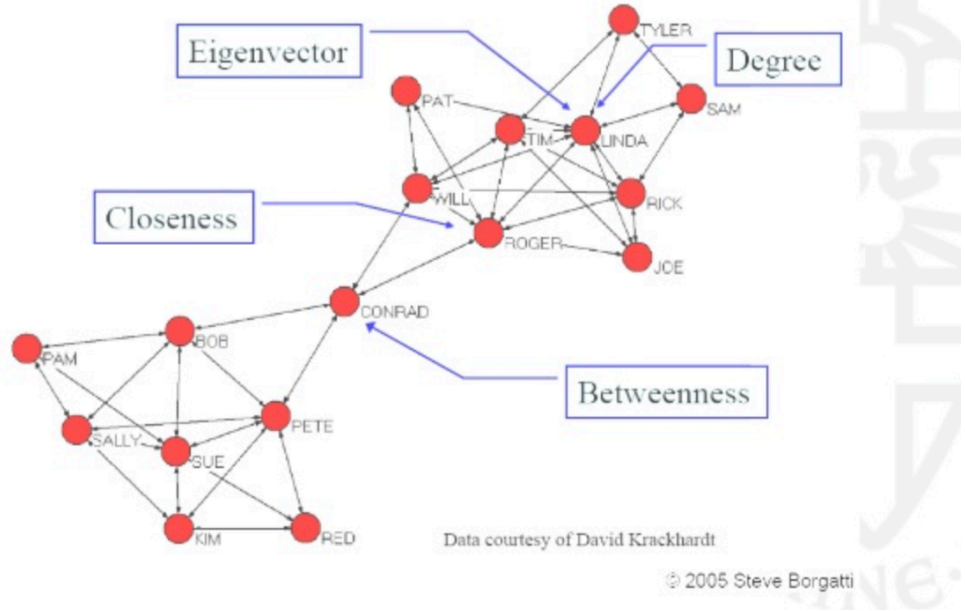
What do Centralities Tell Us?

❖**Degree Centrality**: exposure to the network, opportunity to directly influence.

❖ **Closeness Centrality**: estimates time to hear info; indirect influence; point of rapid diffusion.

❖**Betweenness Centrality**: informal power; gate keeping, brokering controls flow of info; liaison between sub-components.

❖**Eigenvalue Centrality**: connected to influential nodes of high degree, “not what you know but who you know”



who is most important? And Why?

Degree Centrality

A node is important if it has many neighbors, or, in the directed case, if there are many other nodes that link to it, or if it links to many other nodes.

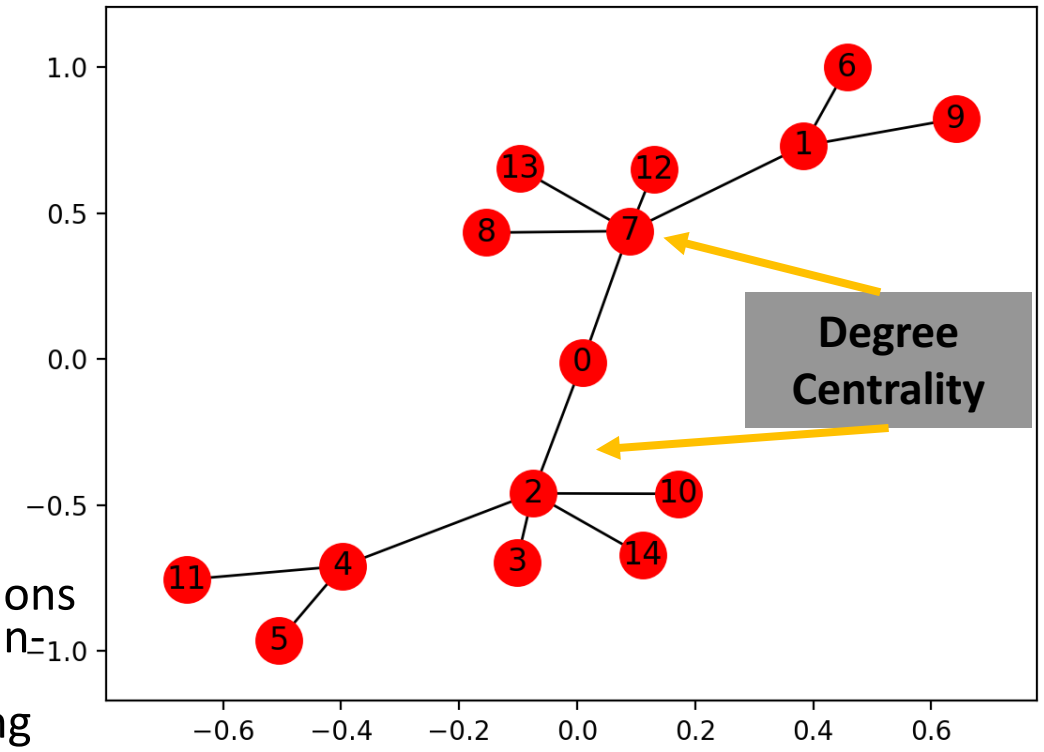
DC(i) $= (\sum_{j=1}^{n} aij)$**/n-1**

i) aij = 0 if i is no connected to j

ii) aij = 1 if i is connected to j

Degree is a simple centrality measure that counts how many neighbors a node has

If the network is directed, we have two versions of the measure: in-degree is the number of in-coming links, or the number of predecessor nodes; out-degree is the number of out-going links, or the number of successor nodes



Node	DC
0:	2/14 = 0.14,
1:	3/14 = 0.21,
2:	5/14 = 0.35,
3:	0.07,
4:	0.21,
5:	0.07,
6:	0.07,
7:	0.35,
8:	0.07,
9:	0.07,
10:	0.07,
11:	0.07,
12:	0.07,
13:	0.07,
14:	0.07

Closeness Centrality

The average of the shortest distance to all other nodes in the graph.

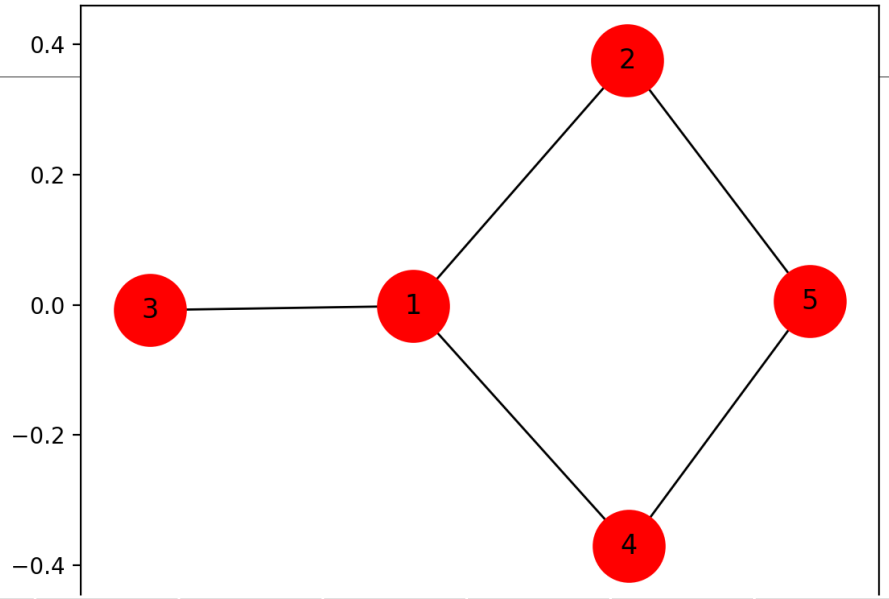
CC(i) =n-1/ $(\sum_{j=1}^{n} dij)$

dij = shortest distance between i and j

What do the Closeness Centrality Tell us?

- **Estimates time to hear info**
- **Indirect influence**
- **Point of rapid diffusion.**

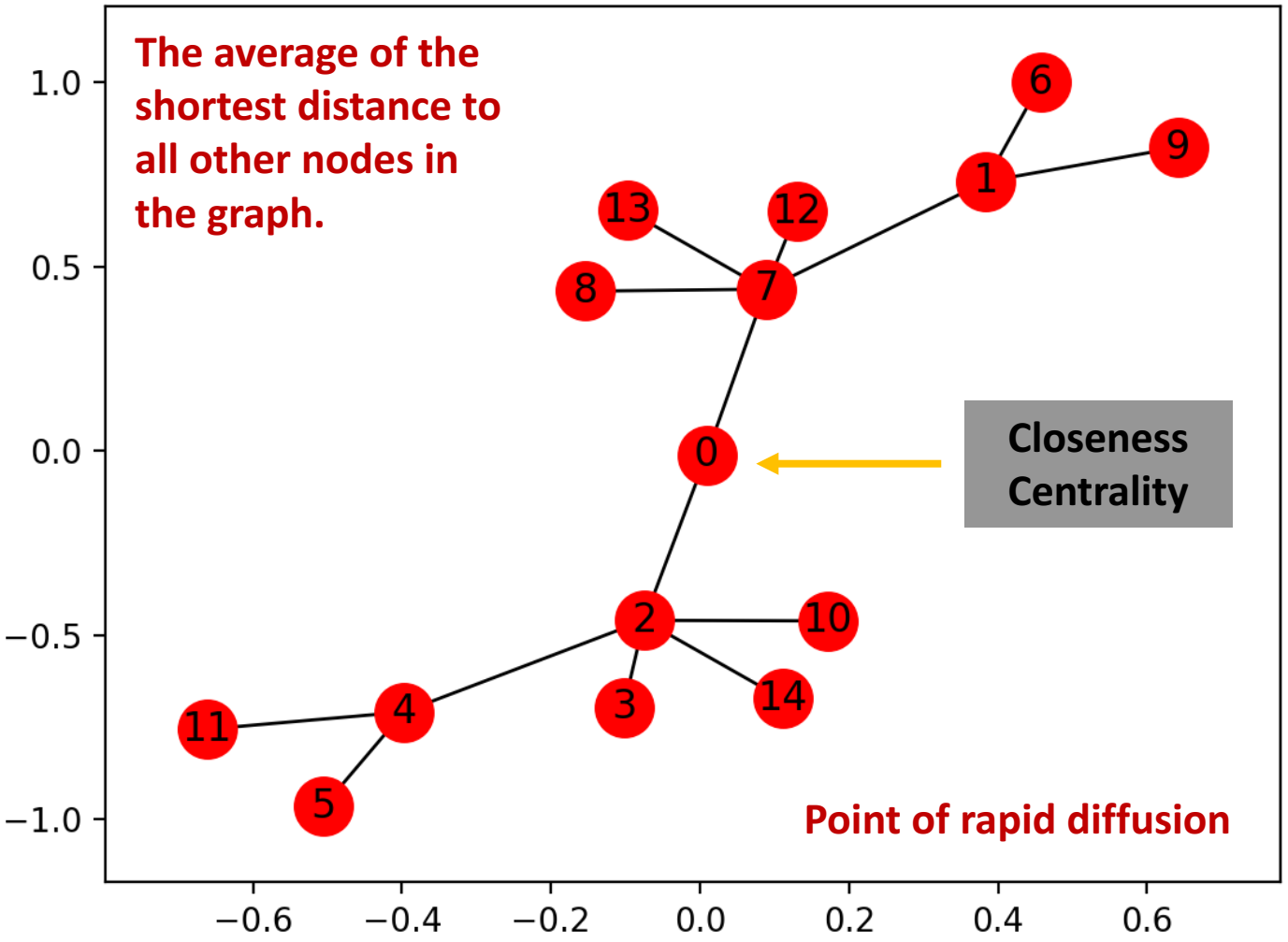
Example: Rumer Spread or Disease Spread



	1	2	3	4	5	$\sum_{j=1}^{n}$	CC(i)
1	0	1	1	1	2	5	4/5
2	1	0	2	2	1	6	4/6
3	1	2	0	2	3	8	4/8
4	1	2	2	0	1	6	4/6
5	2	1	3	1	0	7	4/7

Shortest Path Matrix

Closeness Centrality



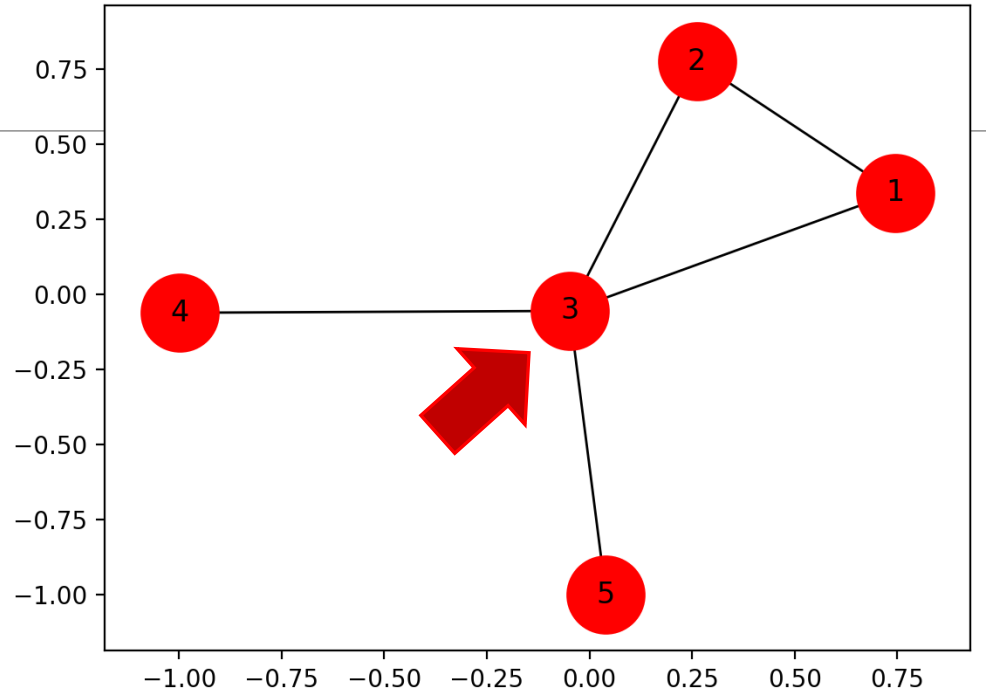
Node	Closeness
0:	**0.46**
1:	0.35
2:	0.45
3:	0.31
4:	0.35
5:	0.26
6:	0.26
7:	0.45
8:	0.31
9:	0.26
10:	0.31
11:	0.26
12:	0.31
13:	0.31
14:	0.31

Betweenness Centrality

The extent to which a particular node lies on the shortest path between other nodes.

What do the Betweenness Centrality Tell us?

- Informal power
- Gate keeping
- Brokering
- Control flow of info
- Liaison between sub-components
- Possible target for disruption of network

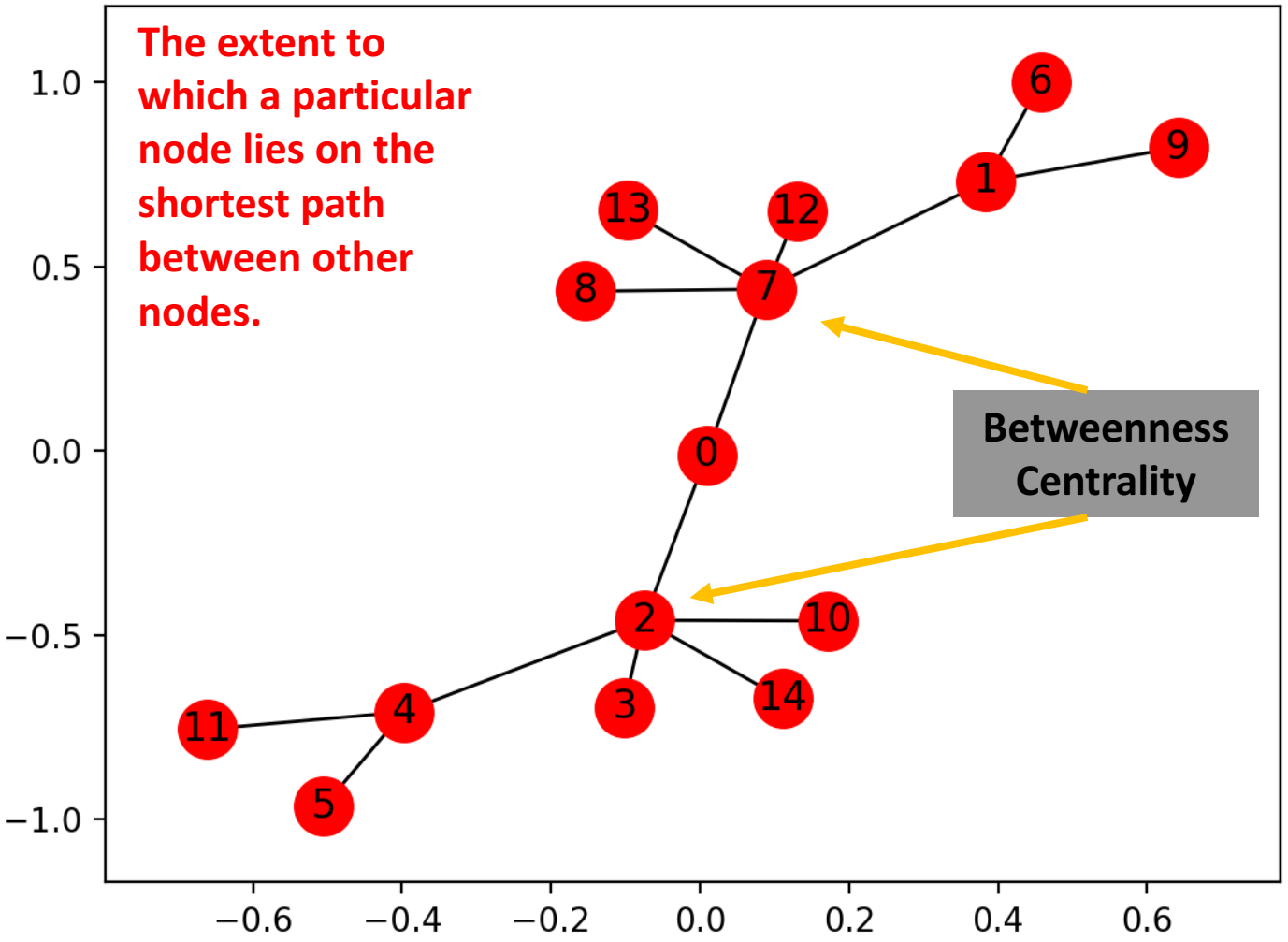


	1	2	3	4	5
1	-	(1,2)	(1,3)	(1,3,4)	(1,3,5)
2	-	-	(2,3)	(2,3,4)	(2,3,5)
3	-	-	-	(3,4)	(3,5)
4	-	-	-	-	(4,3,5)
5	-	-	-	-	-

Edges = (n-1)(n-2)/2 = 4x3/2 = 6

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

Betweenness Centrality



Node	Betweenness
0:	0.53
1:	0.27
2:	**0.65**
3:	0.0
4:	0.27
5:	0.0
6:	0.0
7:	**0.65**
8:	0.0
9:	0.0
10:	0.0
11:	0.0
12:	0.0
13:	0.0
14:	0.0

Eigenvalue Centrality

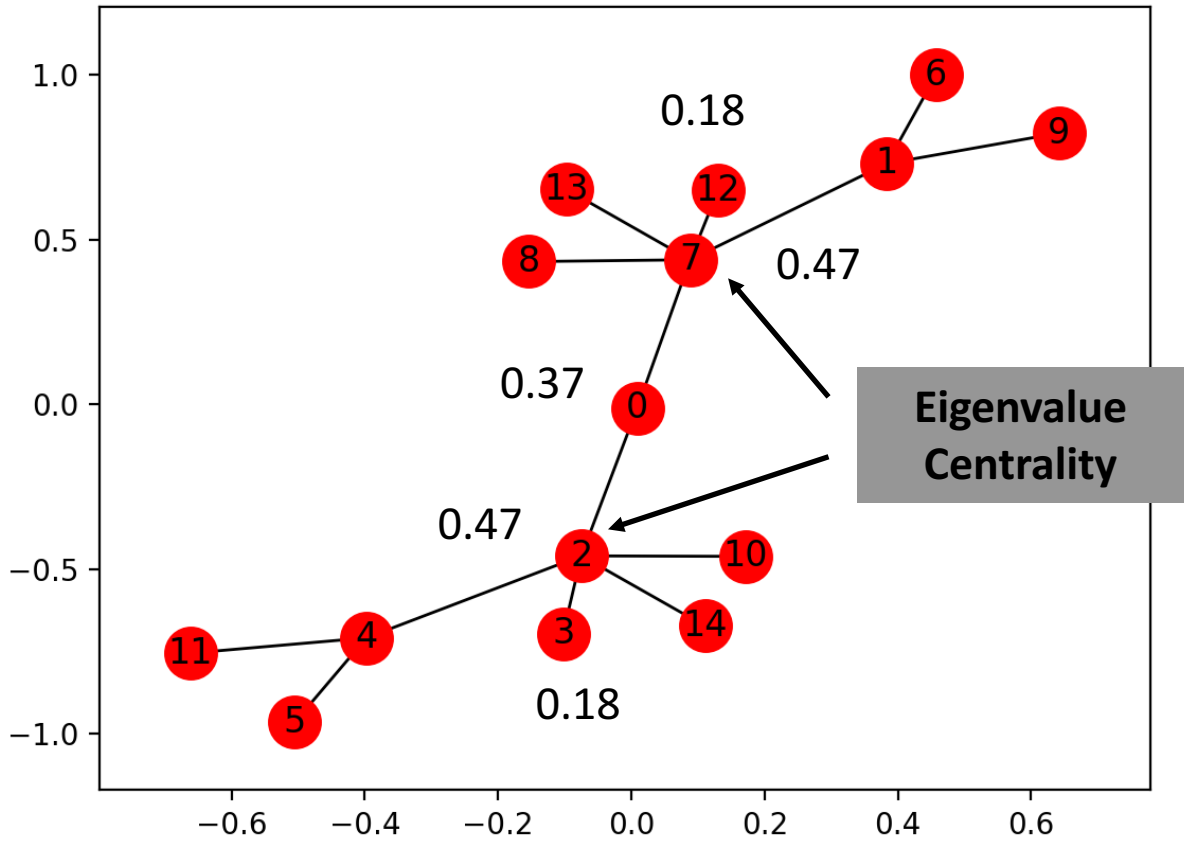
A natural extension of degree centrality is **eigenvector centrality**.

A node is important if it is linked to by other important nodes.

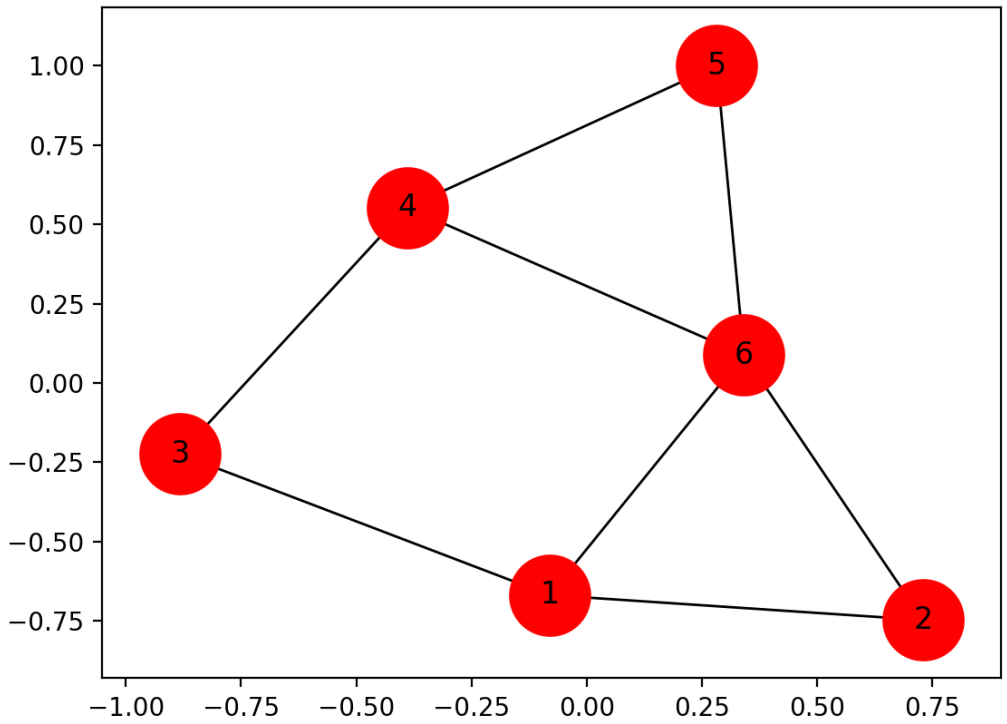
Node has high score if connected to many nodes are themselves well connected
Computed as:

$\boldsymbol{Av = \lambda v}$
where A is adjacency matrix and V is eigenvector

centrality. V is the principal eigenvector of A. Indicator of popularity , “in the know”. Tends to identify centers of large cliques



Eigenvalue

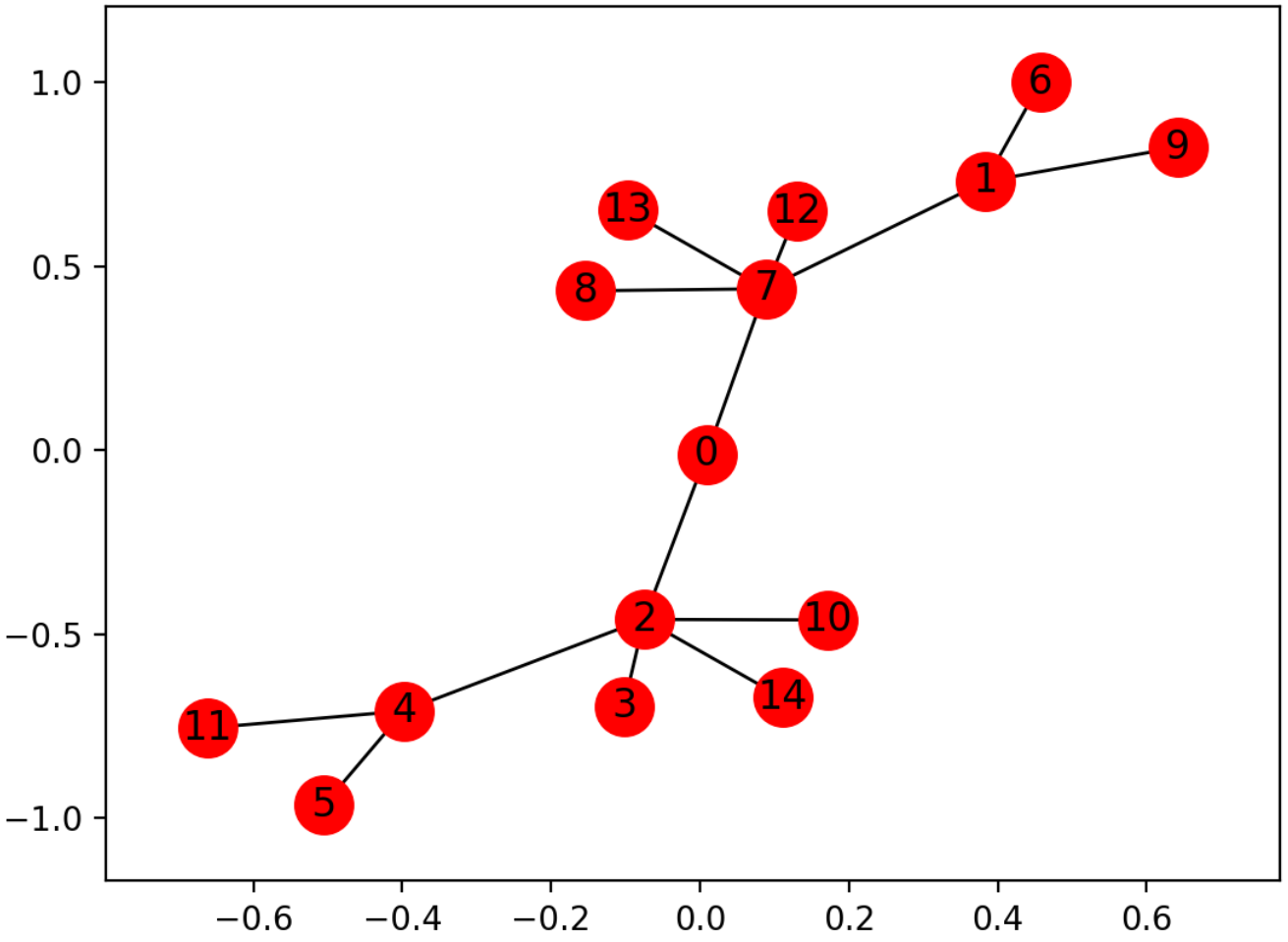


Adjacency Metrix

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

$$Av = \lambda v$$

Eigenvalue Centrality

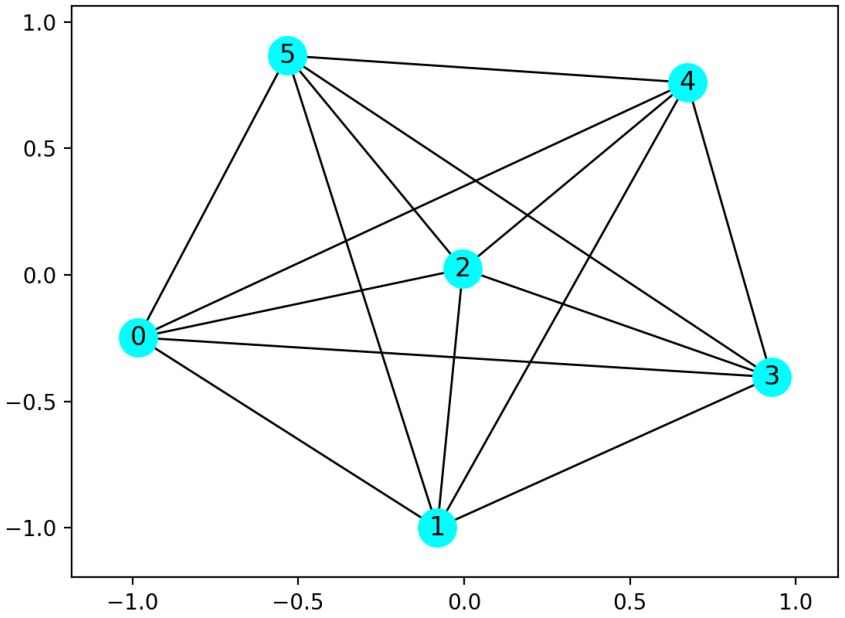


Node	Centrality
0:	0.53
1:	0.27
2:	0.65
3:	0.0
4:	0.27
5:	0.0
6:	0.0
7:	0.65
8:	0.0
9:	0.0
10:	0.0
11:	0.0
12:	0.0
13:	0.0
14:	0.0

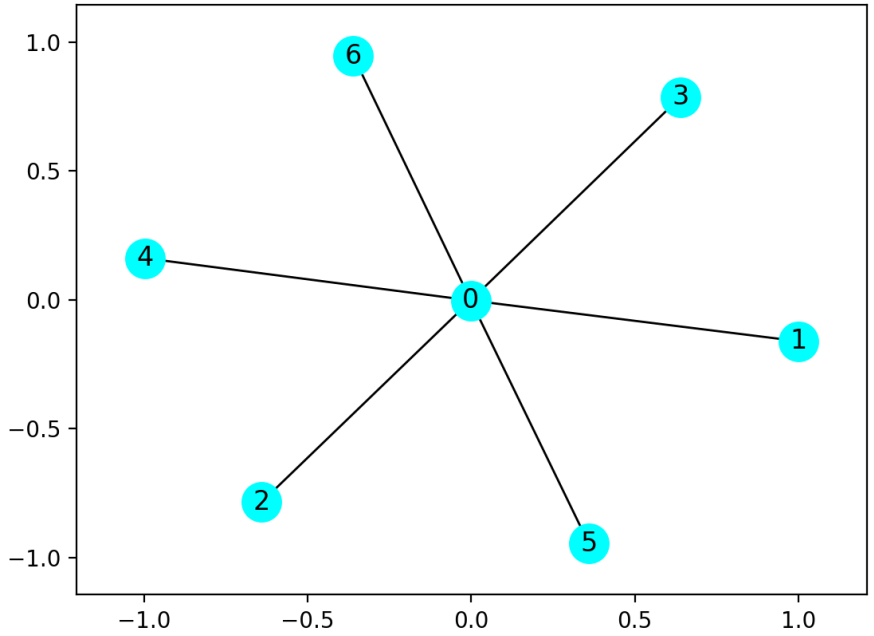
Clustering Coefficient - CC(v)

A clustering coefficient is a measure of the degree to which nodes in a graph tend to cluster together.

How connected are my neighbors?



6-Clique



6-Star

6-Clique topology, for each node CC(v) = 20/20 = 1

6-Star topology, CC(0) = 0

Clustering Coefficient CC(v)

v a node, Degree of node = Dv

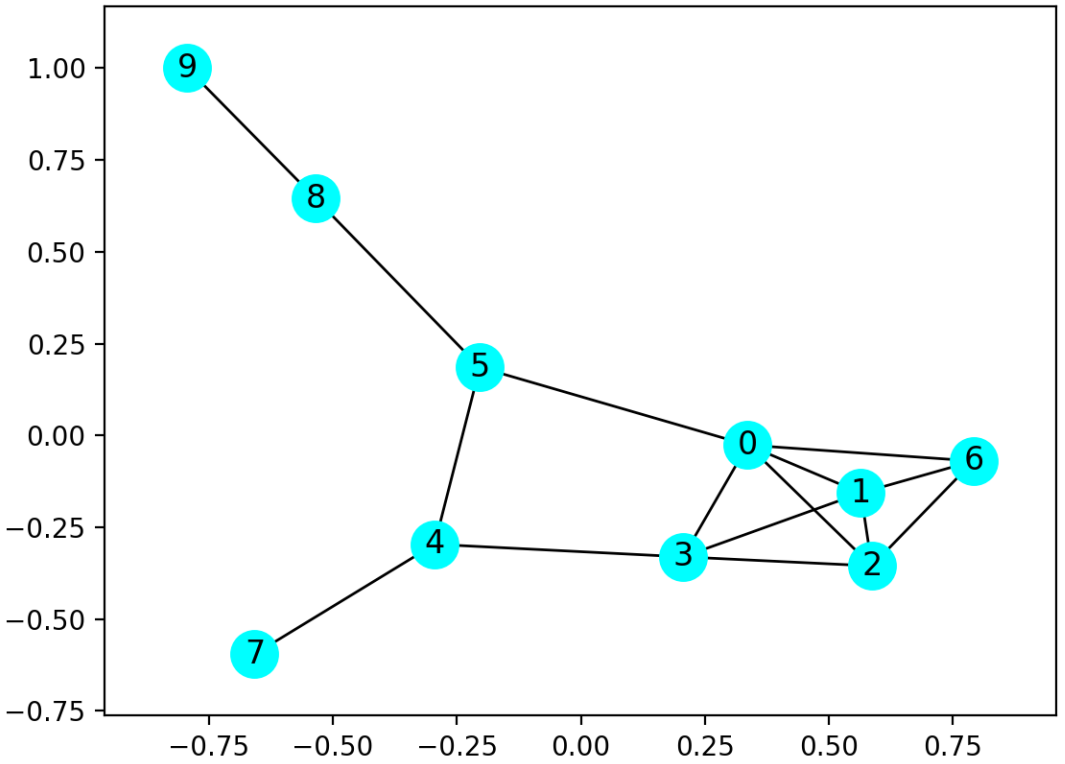
Nv: number of links between Neighbors of V

Number of V's neighbors = Degree of V → Dv

Maximum number of links between V's neighbors
(Dv * (Dv-1))/2

Clustering Coefficient CC(v)

2 x Nv / (Di x (Di-1)) , 0 <= CC(v) <=1



CC(0): 0.5
CC(1): 0.83
CC(2): 0.83
CC(3): 0.5
CC(4): 0
CC(5): 0
CC(6): 1.0
CC(7): 0
CC(8): 0
CC(9): 0