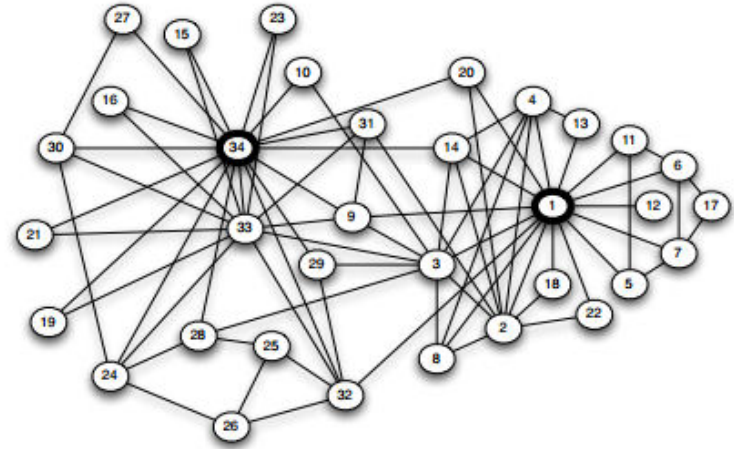


Node Importance

Based on the structure of the network, which are the 5 most important nodes in the Karate Club friendship network?



Friendship network in a 34-person karate club
[Zachary 1977]

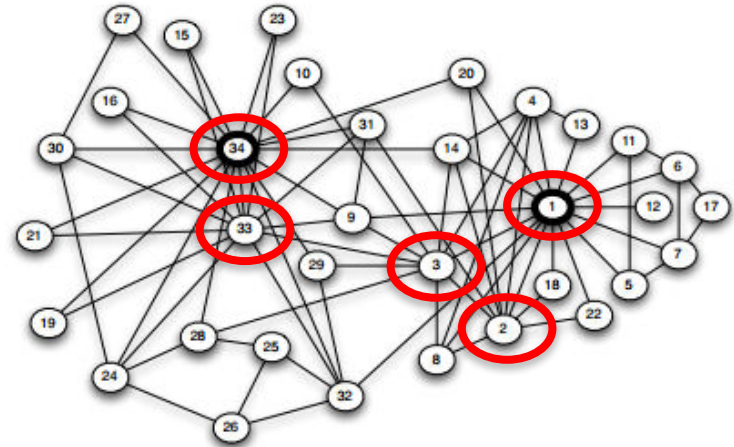
Node Importance

Different ways of thinking about “importance”.

Ex. Degree: number of friends.

5 most important nodes are:

34, 1, 33, 3, 2



Friendship network in a 34-person karate club
[Zachary 1977]

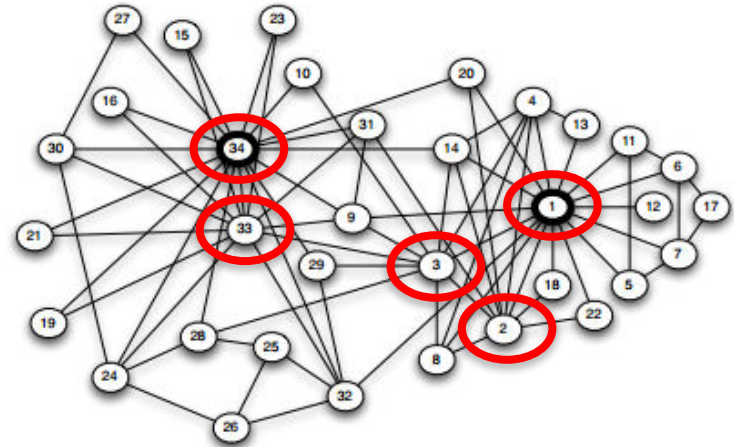
Node Importance

Different ways of thinking about “importance”.

Ex. Average proximity to other nodes.

5 most important nodes are:

1, 3, 34, 32, 9



Friendship network in a 34-person karate club
[Zachary 1977]

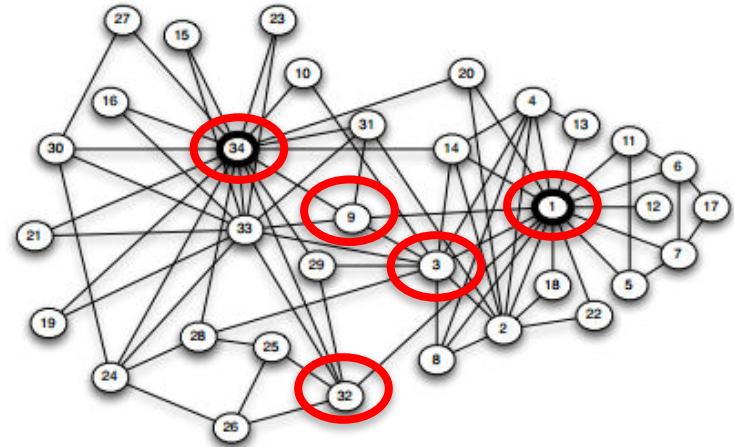
Node Importance

Different ways of thinking about “importance”.

Ex. Fraction of shortest paths that pass through node.

5 most important nodes are:

1, 34, 33, 3, 32



Friendship network in a 34-person karate club
[Zachary 1977]

Network Centrality

Centrality measures identify the most important nodes in a network:

- Influential nodes in a social network.
- Nodes that disseminate information to many nodes or prevent epidemics.
- Hubs in a transportation network.
- Important pages on the Web.
- Nodes that prevent the network from breaking up.

Centrality Measures

- **Degree centrality**
- **Closeness centrality**
- Betweenness centrality
- Load centrality
- Page Rank
- Katz centrality
- Percolation centrality

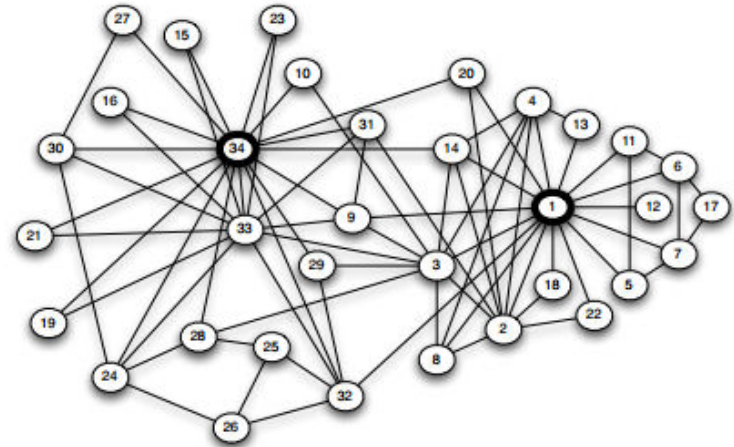
Degree Centrality

Assumption: important nodes have many connections.

The most basic measure of centrality: number of neighbors.

Undirected networks: use degree

Directed networks: use in-degree or out-degree



Friendship network in a 34-person karate club
[Zachary 1977]

Degree Centrality – Undirected Networks

$C_{deg}(v) = \frac{d_v}{|N|-1}$, where N is the set of nodes in the network and d_v is the degree of node v .

```
In: G = nx.karate_club_graph()
```

```
In: G =
```

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

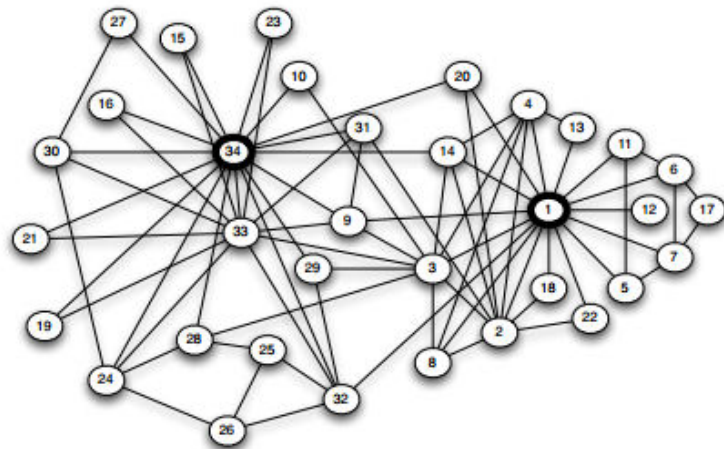
```
In: degCent = nx.degree_centrality(G)
```

```
In: degCent[34]
```

```
Out: 0.515 # 17/33
```

```
In: degCent[33]
```

```
Out: 0.182 # 6/33
```



Friendship network in a 34-person karate club
[Zachary 1977]



Degree Centrality – Directed Networks

$$C_{outdeg}(v) = \frac{d_v^{out}}{|N|-1}, \text{ where}$$

N = set of nodes in the network,

d_v^{out} = the out-degree of node v .

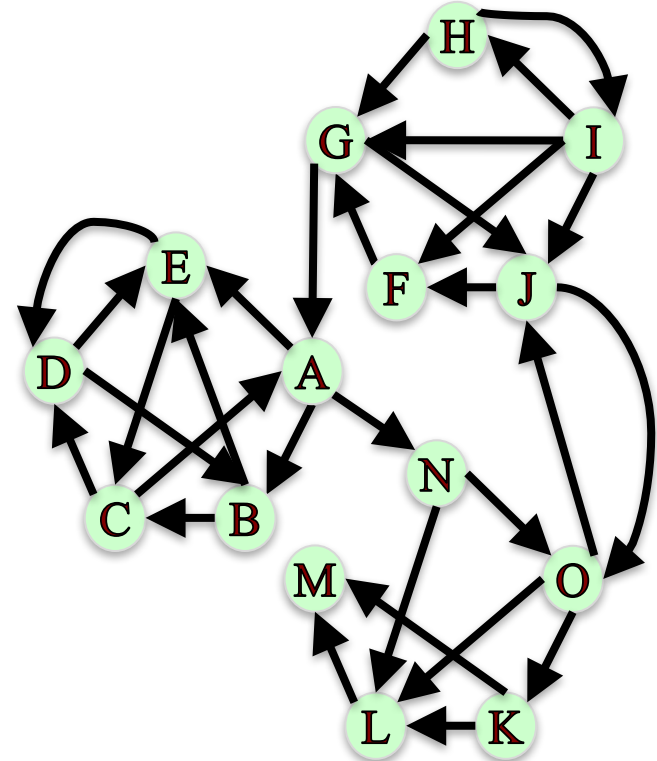
```
In: outdegCent = nx.out_degree_centrality(G)
```

```
In: outdegCent['A']
```

```
Out: 0.214 # 3/14
```

```
In: outdegCent['L']
```

```
Out: 0.071 # 1/14
```



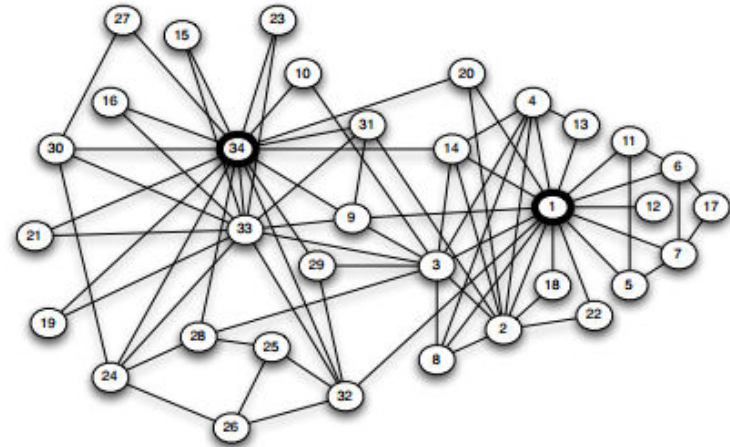
Closeness Centrality

Assumption: important nodes are close to other nodes.

$$C_{close}(v) = \frac{|N|-1}{\sum_{u \in N \setminus \{v\}} d(v,u)}, \text{ where}$$

N = set of nodes in the network,

$d(v,u)$ = length of shortest path from v to u .



Friendship network in a 34-person karate club
[Zachary 1977]

Closeness Centrality

Assumption: important nodes are close to other nodes.

In: `closeCent = nx.closeness centrality(G)`

In: `closeCent[32]`

Out: 0.541

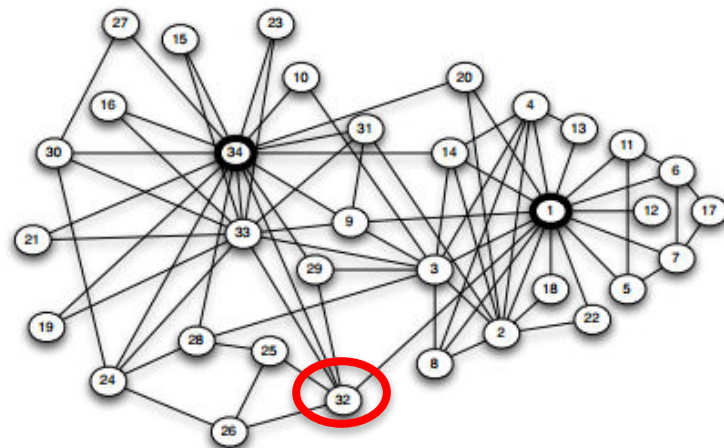
In: `sum(nx.shortest_path_length(G,32).values())`

Out: 61

In: `(len(G.nodes())-1)/61.`

Out: 0.541

$\frac{1}{61} = \frac{1}{61}$ $\frac{1}{61}$ $\frac{1}{61}$
 $\frac{1}{61} = \frac{1}{61}$ $\frac{1}{61}$ $\frac{1}{61}$



Friendship network in a 34-person karate club
[Zachary 1977]

Disconnected Nodes

How to measure the closeness centrality of a node when it cannot reach all other nodes?

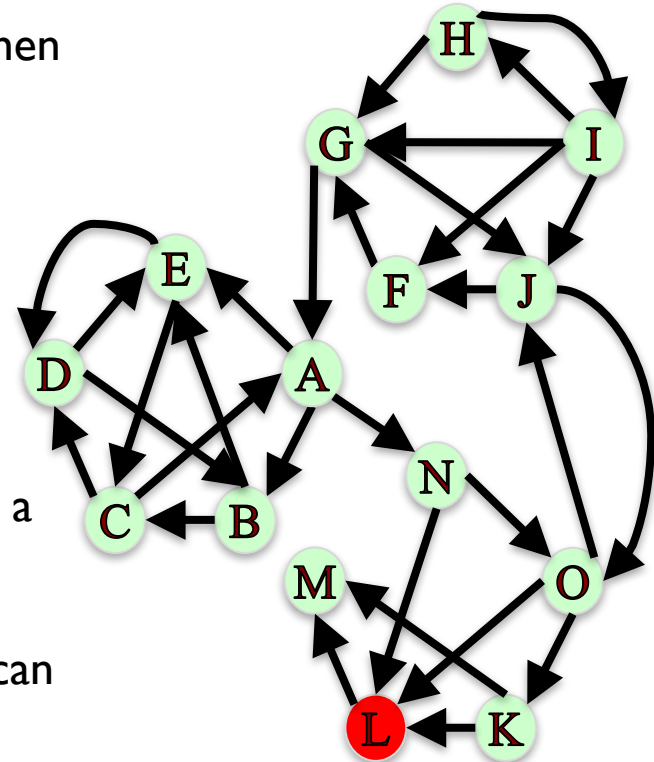
What is the closeness centrality of node L?

Option 1: Consider only nodes that L can reach:

$C_{close}(L) = \frac{|R(L)|}{\sum_{u \in R(L)} d(L, u)}$, where $R(L)$ is the set of nodes L can reach.

$C_{close}(L) = \frac{1}{1} = 1$, since L can only reach M and it has a shortest path of length 1.

Problem: centrality of I is too high for a node than can only reach one other node!



Disconnected Nodes

How to measure the closeness centrality of a node when it cannot reach all other nodes?

What is the closeness centrality of node L?

```
In: closeCent = nx.closeness centrality(G, normalized = False)
```

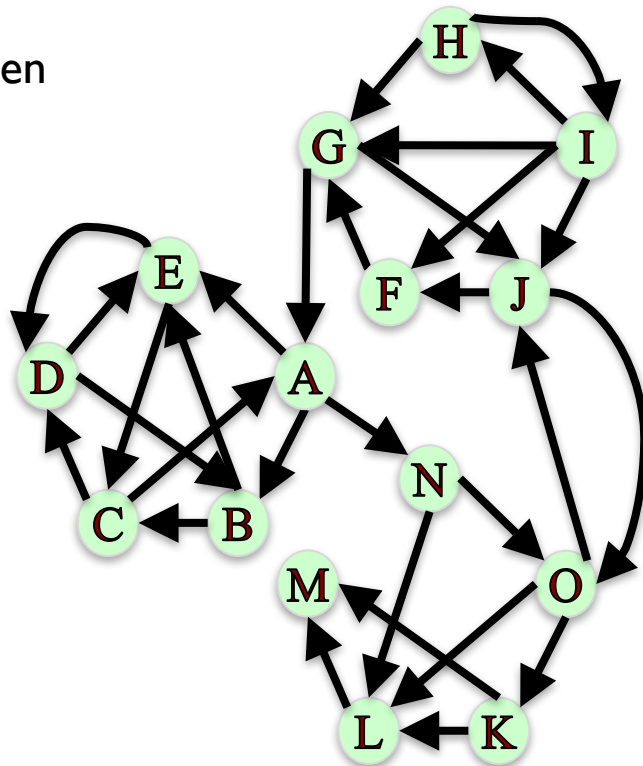
```
In: closeCent['L']
```

```
Out: 1
```

```
In: closeCent = nx.closeness centrality(G, normalized = True)
```

```
In: closeCent['L']
```

```
Out: 0.071
```



Summary

Centrality measures identify the most important nodes in a network:

Degree Centrality

Assumption: important nodes have many connections.

$$C_{deg}(v) = \frac{d_v}{|N| - 1}$$

`nx.degree centrality(G)`

`nx.in_degree centrality(G)`

`nx.out_degree centrality(G)`

Closeness Centrality

Assumption: important nodes are close to other nodes.

$$C_{close}(L) = \frac{\frac{|R(L)|}{|N|-1}}{\sum_{u \in R(L)} d(L,u)}$$

number of nodes I can reach
Σ ↪ steps / distance

`nx.closeness centrality(G, normalized = True)`