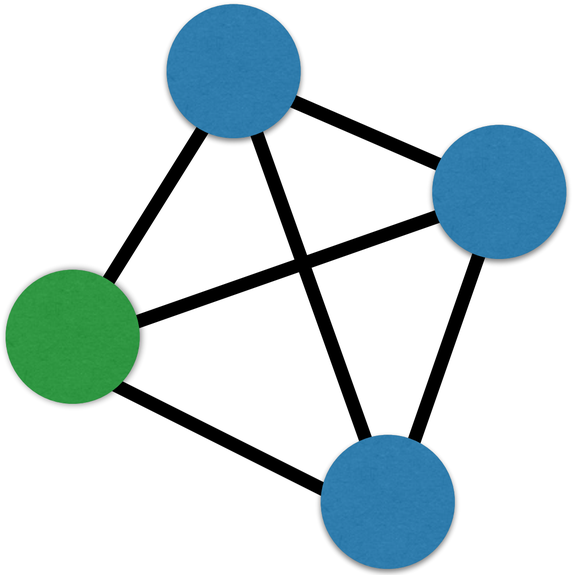
NETWORK ANALYSIS IN PYTHON I

Cliques & communities

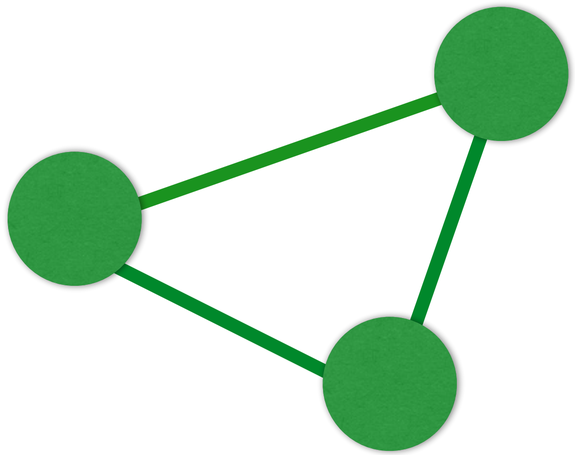
Cliques

- Social cliques: tightly-knit groups
- Network cliques: completely connected graphs



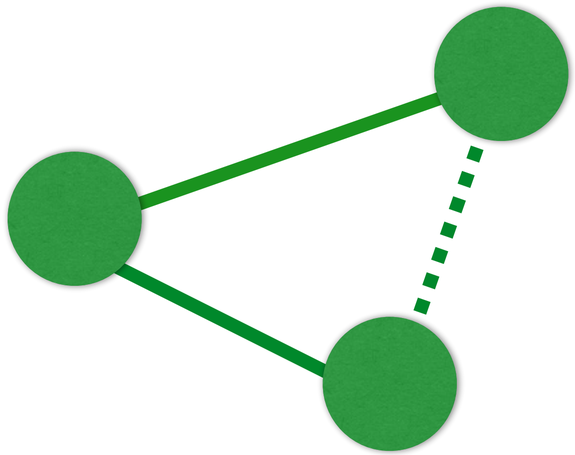
Cliques

- Simplest complex clique: a triangle



Triangle applications

- Friend recommendation systems



Clique code

```
In [1]: G
Out[1]: <networkx.classes.graph.Graph at 0x10c99ecf8>

In [2]: from itertools import combinations

In [3]: for n1, n2 in combinations(G.nodes(), 2):
   ...:     print(n1, n2)
0 1
0 2
0 3
0 4
0 5
...
...
```

NETWORK ANALYSIS IN PYTHON I

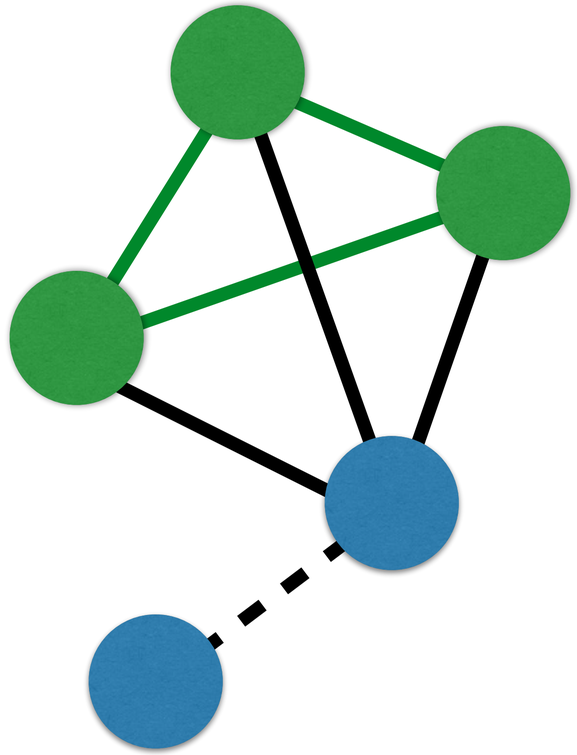
Let’s practice!

NETWORK ANALYSIS IN PYTHON I

Maximal cliques

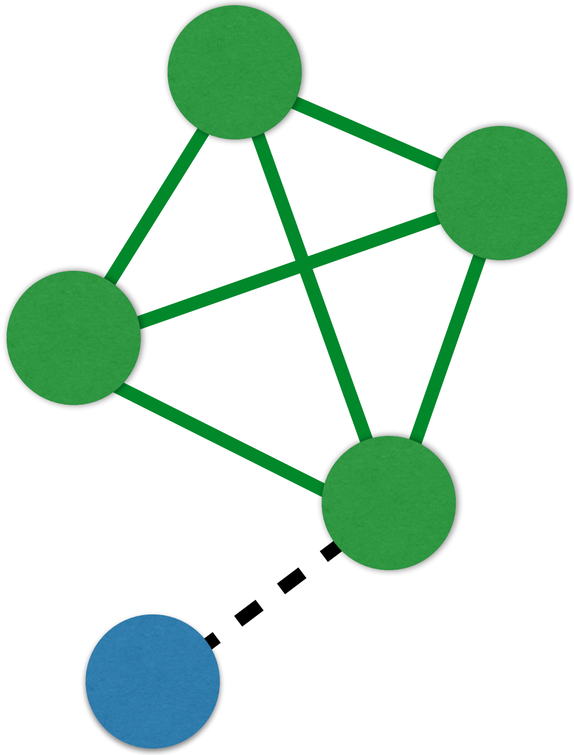
Maximal cliques

- Definition: a clique that, when extended by one node is no longer a clique



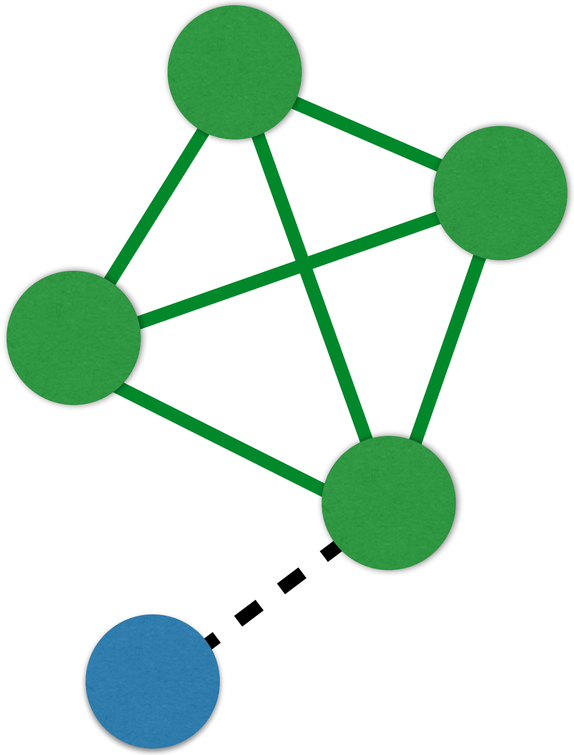
Maximal cliques

- Definition: a clique that, when extended by one node is no longer a clique



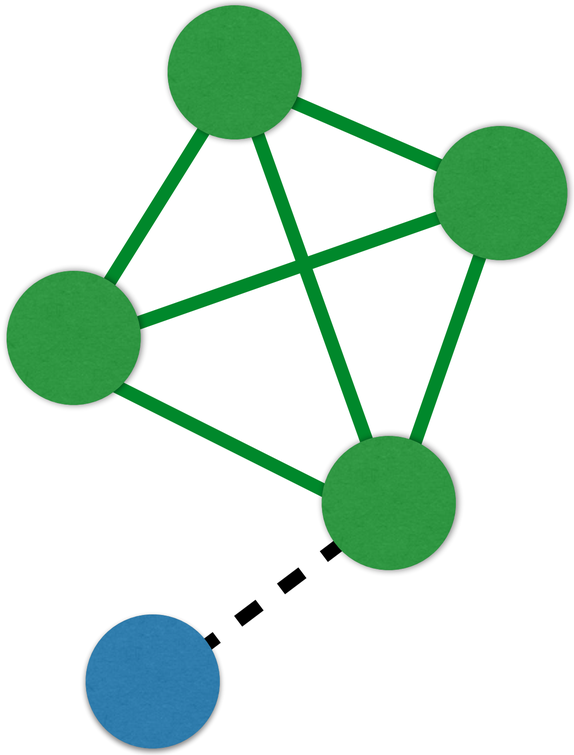
Maximal cliques

- Applications: community finding



Communities

- Find cliques
- Find unions of cliques

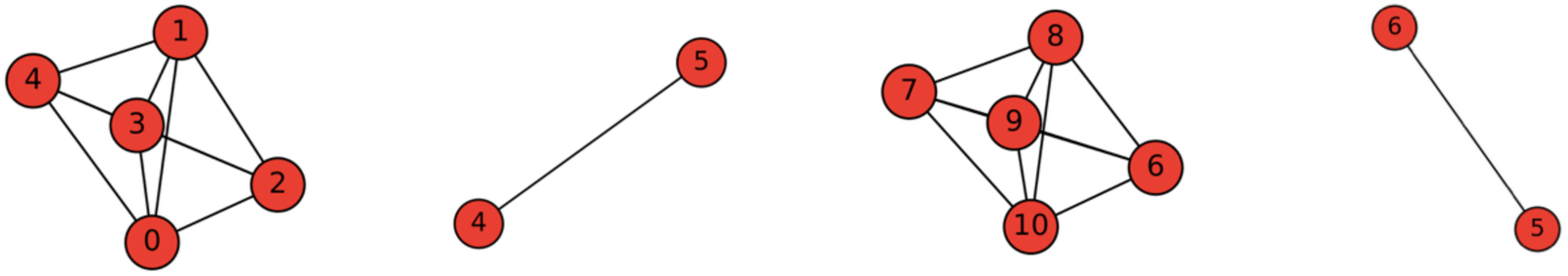


NetworkX API

- find_cliques finds all **maximal** cliques

Maximal cliques

```
In [1]: import networkx as nx

In [2]: G = nx.barbell_graph(m1=5, m2=1)

In [3]: nx.find_cliques(G)
Out[3]: <generator object find_cliques at 0x1043f1f68>

In [4]: list(nx.find_cliques(G))
Out[4]: [[4, 0, 1, 2, 3], [4, 5], [6, 8, 9, 10, 7], [6, 5]]
```



NETWORK ANALYSIS IN PYTHON I

Let’s practice!

NETWORK ANALYSIS IN PYTHON I

Subgraphs

Subgraphs

- Visualize portions of a large graph
  - Paths
  - Communities/cliques
  - Degrees of separation from a node

Subgraphs

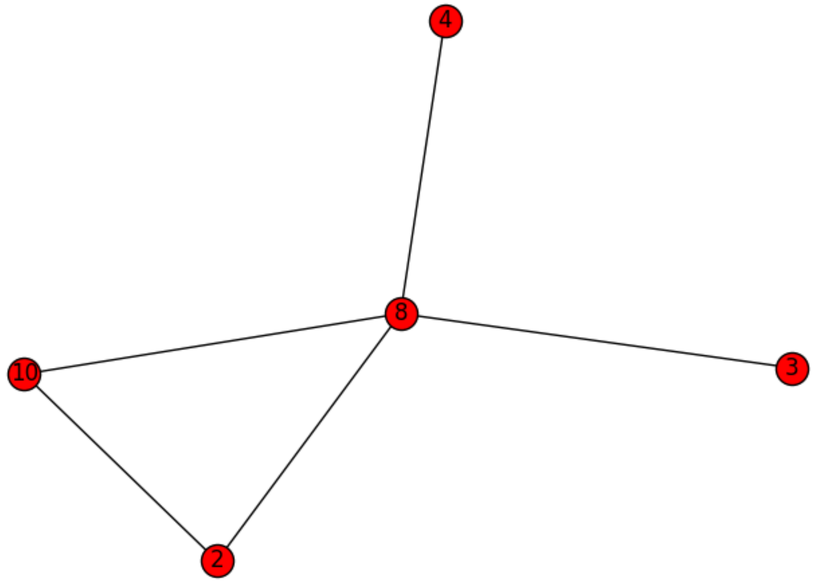
```
In [1]: import networkx as nx

In [2]: G = nx.erdos_renyi_graph(n=20, p=0.2)

In [3]: G.nodes()
Out[3]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16,
17, 18, 19]

In [4]: nodes = G.neighbors(8)

In [5]: nodes
Out[5]: [2, 3, 4, 10]

In [6]: nodes.append(8)
```

Subgraphs

```
In [7]: G_eight = G.subgraph(nodes)

In [8]: G_eight.edges()
Out[8]: [(8, 2), (8, 3), (8, 4), (8, 10), (2, 10)]

In [9]: G_eight
Out[9]: <networkx.classes.graph.Graph at 0x10cae39e8>

In [10]: G
Out[10]: <networkx.classes.graph.Graph at 0x10cad1f60>
```

Subgraphs

```
In [11]: nx.draw(G_eight, with_labels=True)
```



NETWORK ANALYSIS IN PYTHON I

Let’s practice!