Introduction to Networks

NETWORK ANALYSIS IN PYTHON (PART 1)

Eric Ma

Data Carpentry instructor and author of nxviz package

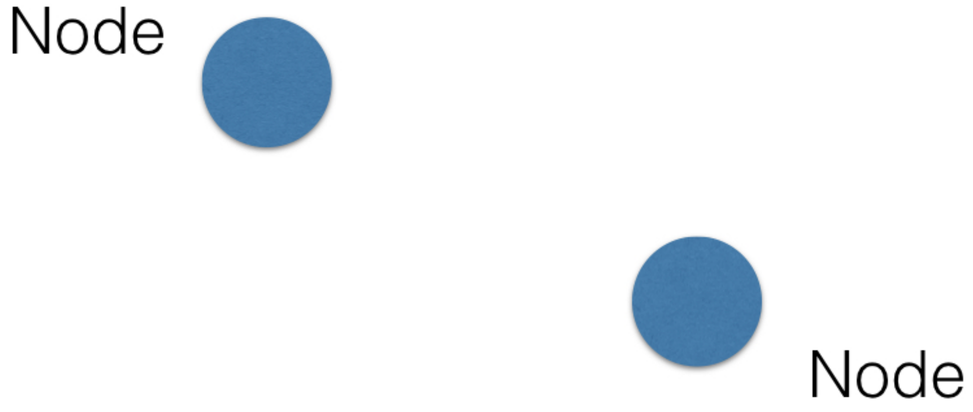
Networks!

- Examples:
  - Social
  - Transportation
- Model relationships between entities

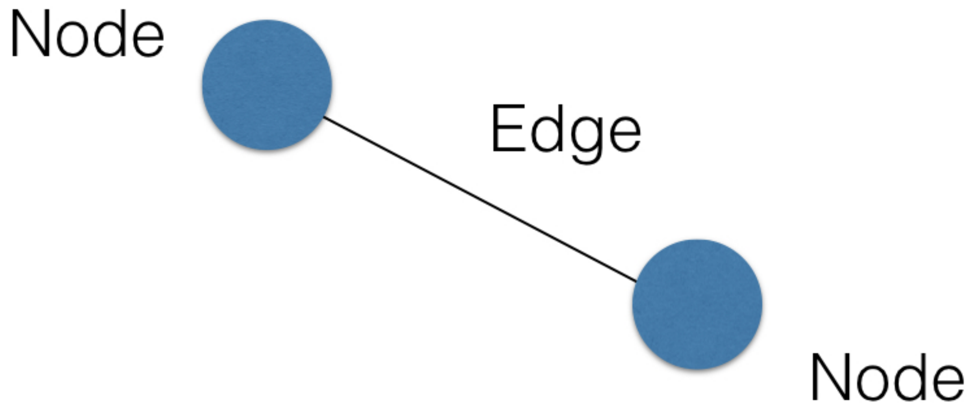
Networks!

- Insights:
- Important entities: influencers in social network
- Pathfinding: most efficient transport path
- Clustering: finding communities

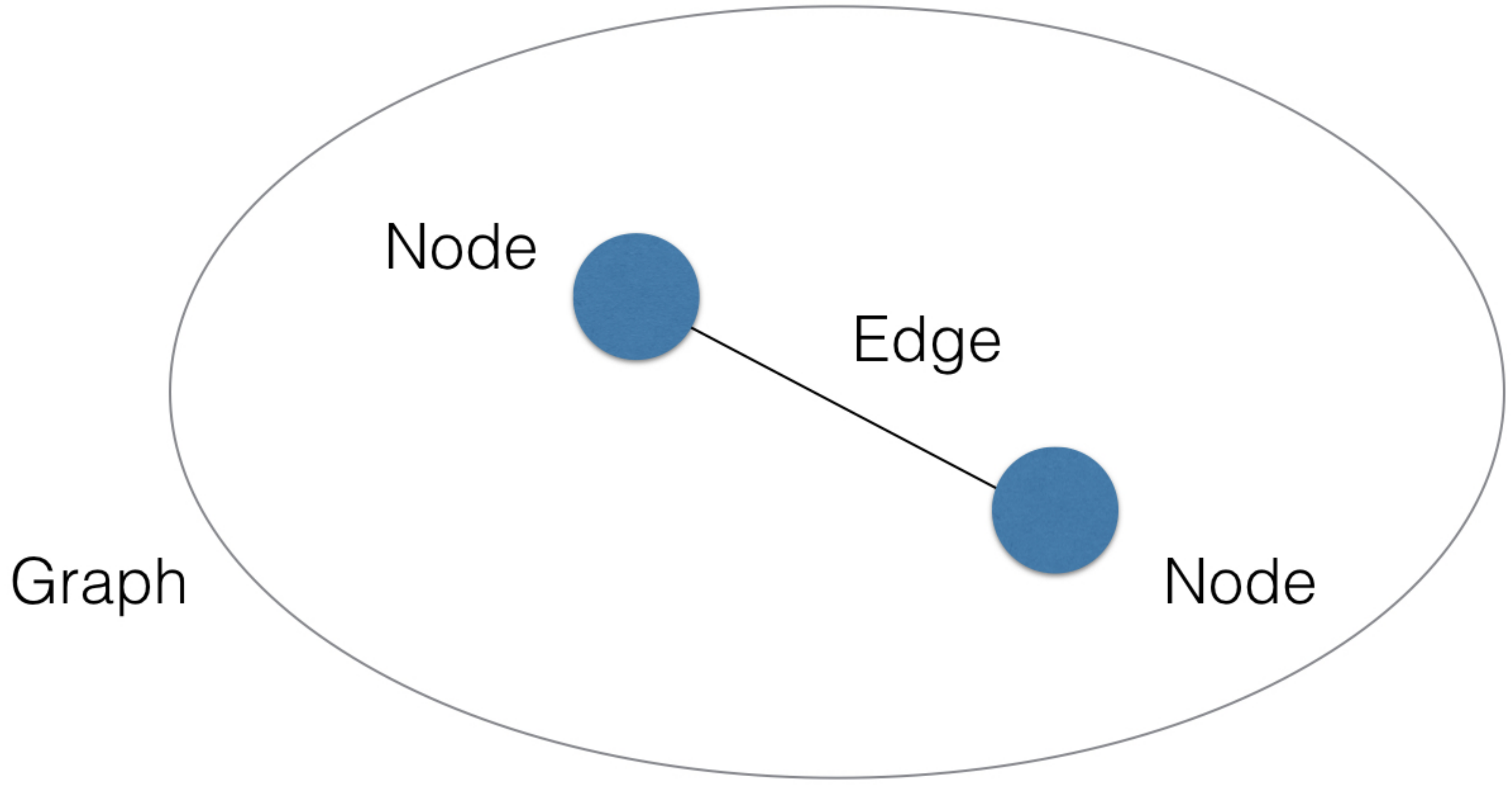
Network Structure



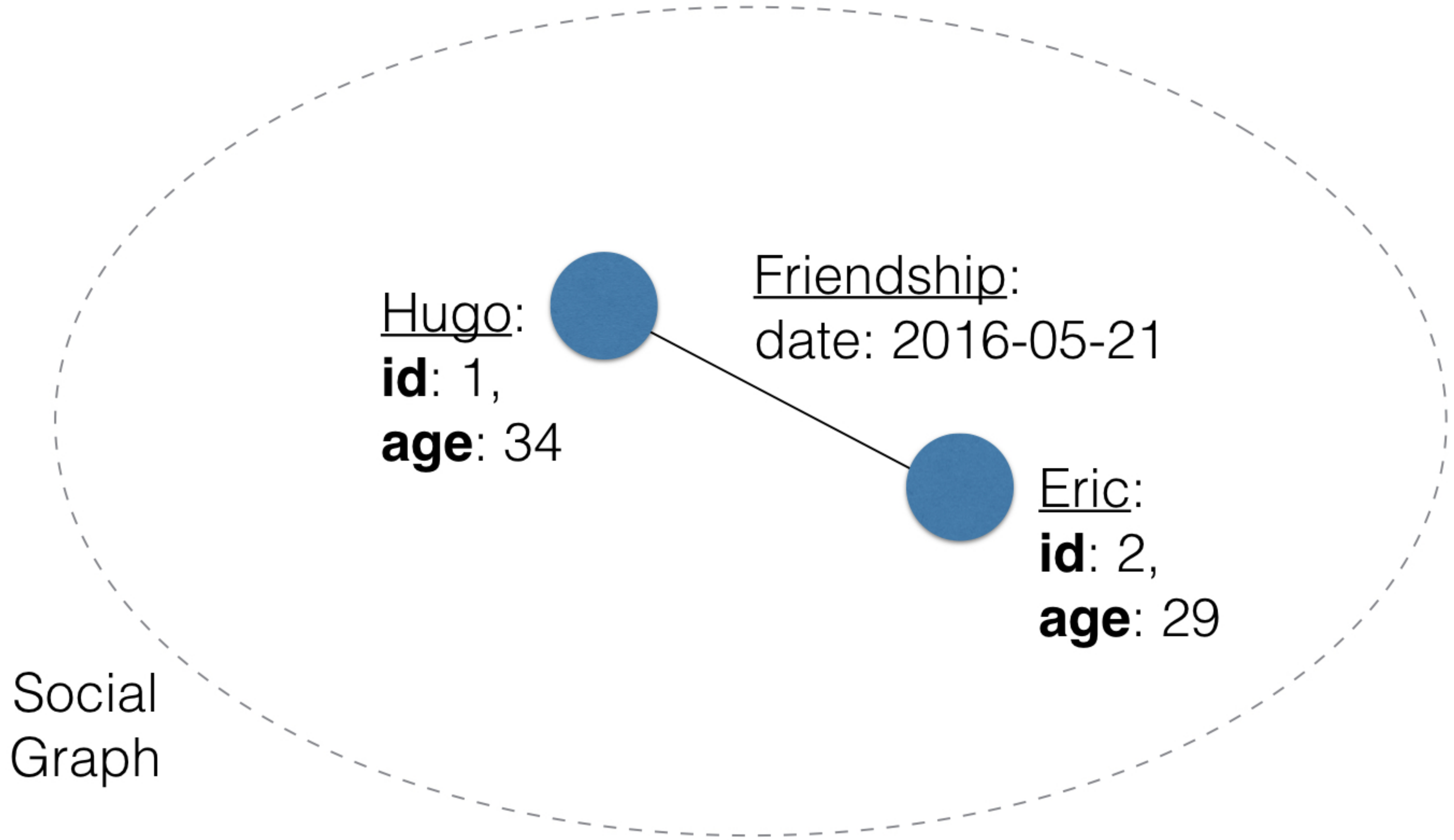
Network Structure



Network Structure



Network Structure



NetworkX API Basics

```
import networkx as nx

G = nx.Graph()

G.add_nodes_from([1, 2, 3])

G.nodes()
```

```
[1, 2, 3]
```

```
G.add_edge(1, 2)

G.edges()
```

```
[(1, 2)]
```

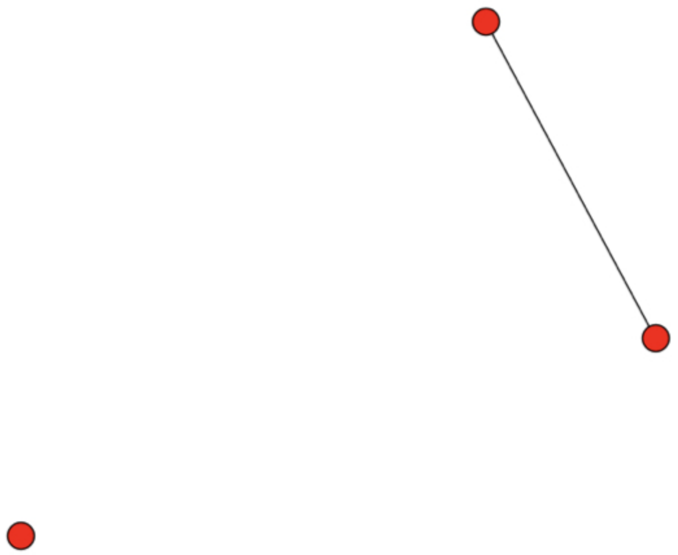
NetworkX API Basics

```
G.node[1]['label'] = 'blue'

G.nodes(data=True)
```

```
[(1, {'label': 'blue'}), (2, {}), (3, {})]
```

NetworkX API Basics

```
nx.draw(G)

import matplotlib.pyplot as plt
plt.show()
```



Let's practice!

NETWORK ANALYSIS IN PYTHON (PART 1)

Types of graphs

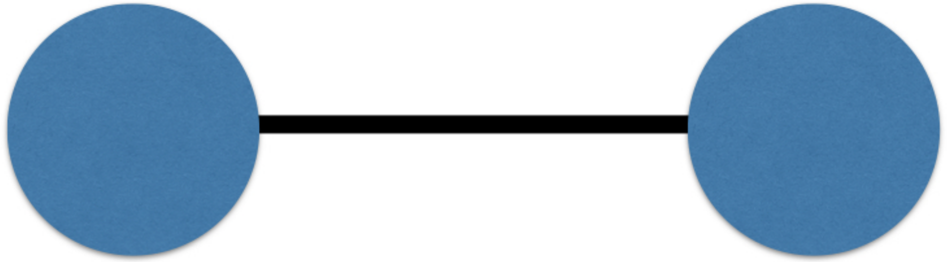
NETWORK ANALYSIS IN PYTHON (PART 1)

Eric Ma

Data Carpentry instructor and author of nxviz package

Undirected graphs

- Facebook social graph



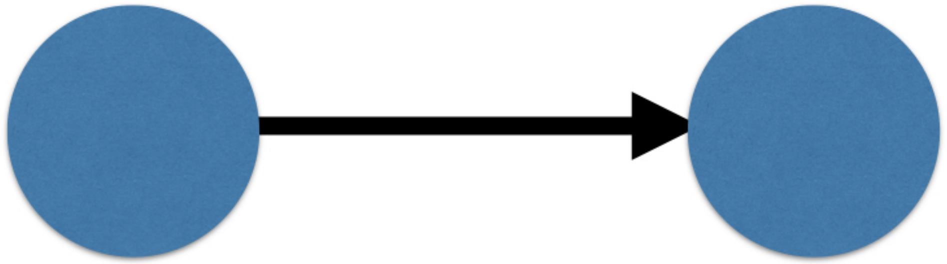
Undirected graphs

```
import networkx as nx
G = nx.Graph()
type(G)
```

```
networkx.classes.graph.Graph
```

Directed graphs

- Directed: Twitter social graph



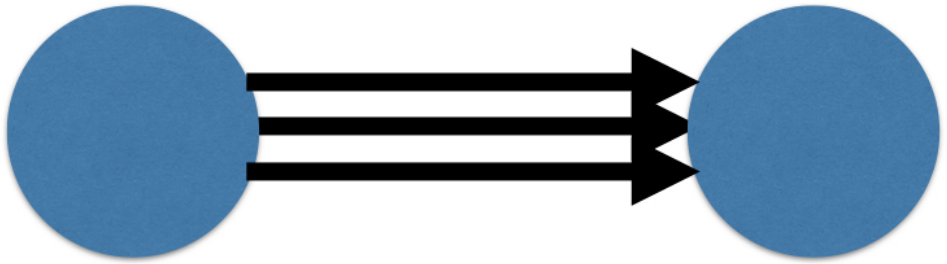
Directed graphs

```
D = nx.DiGraph()
type(D)
```

```
networkx.classes.digraph.DiGraph
```

Types of graphs

- Multi(Di)Graph: Trip records between bike sharing stations



Multi-edge (Directed) graphs

```
M = nx.MultiGraph()

type(M)
```

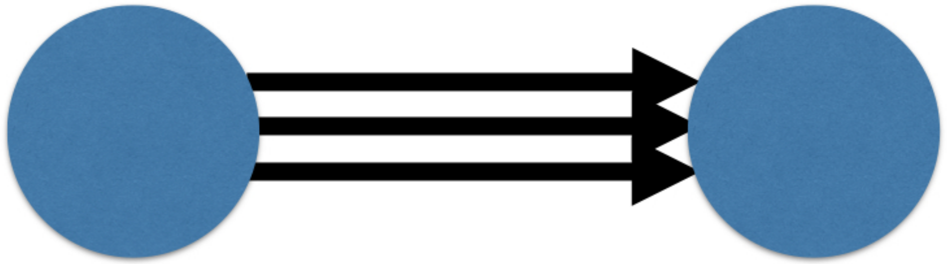
```
networkx.classes.multigraph.MultiGraph
```

```
MD = nx.MultiDiGraph()
type(MD)
```

```
networkx.classes.multidigraph.MultiDiGraph
```

Weights on graphs

- Edges can contain weights

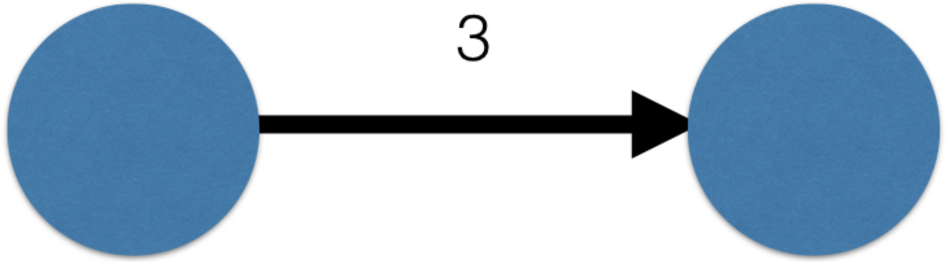


Weights on graphs

- Edges can contain weights

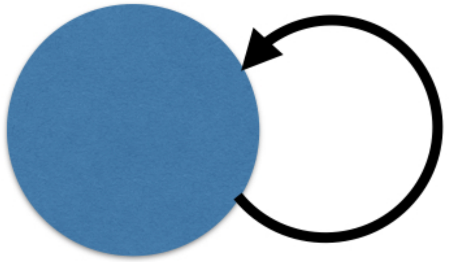
Weights on graphs

- Edges can contain weights



Self-loops

- Nodes that are connected to themselves



Let's practice!

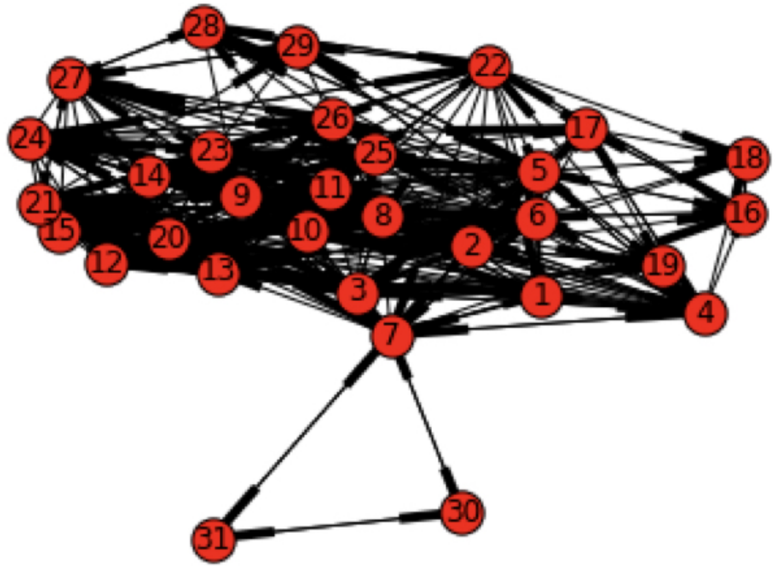
NETWORK ANALYSIS IN PYTHON (PART 1)

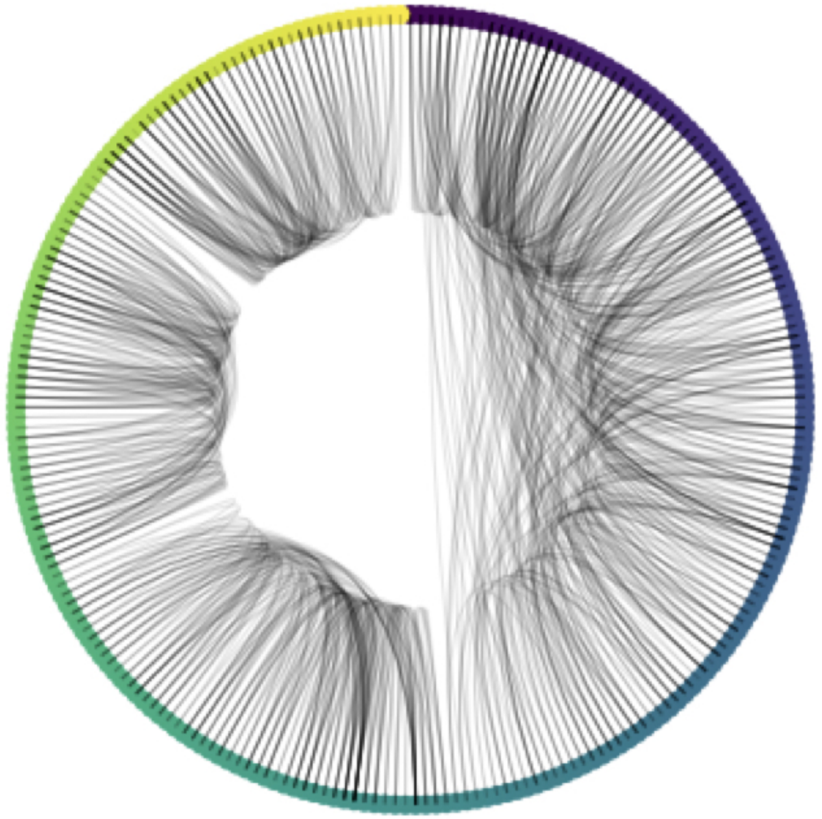
Network visualization

NETWORK ANALYSIS IN PYTHON (PART 1)

Eric Ma
Data Carpentry instructor and author of nxviz package

Irrational vs. Rational visualizations





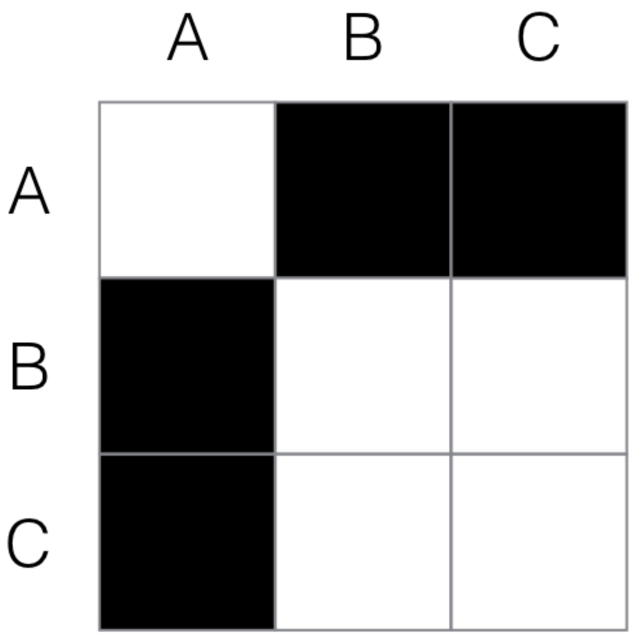
Visualizing networks

- Matrix plots
- Arc plots
- Circos plots

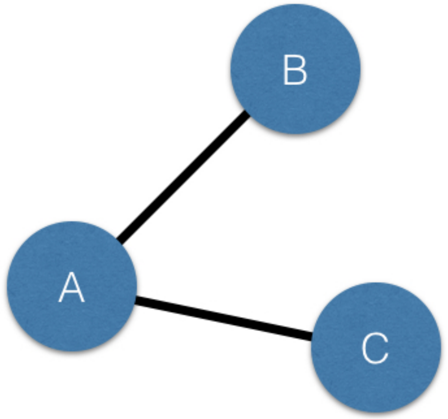
Visualizing networks

- **Matrix plots**
- Arc plots
- Circos plots

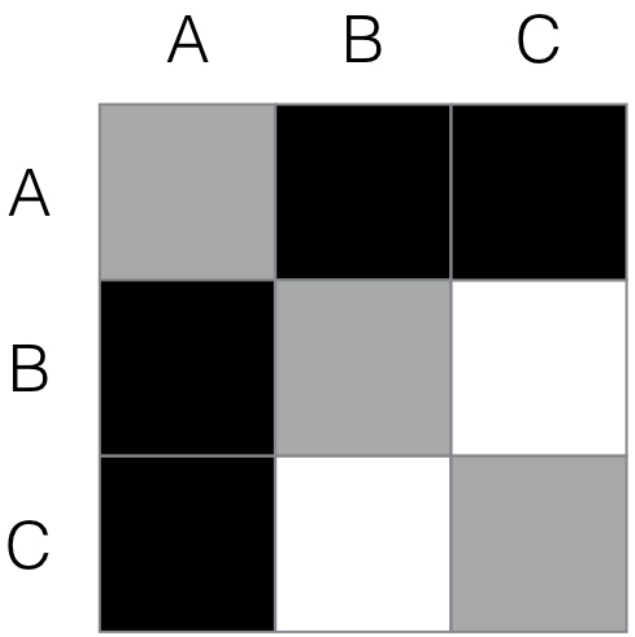
Matrix plot

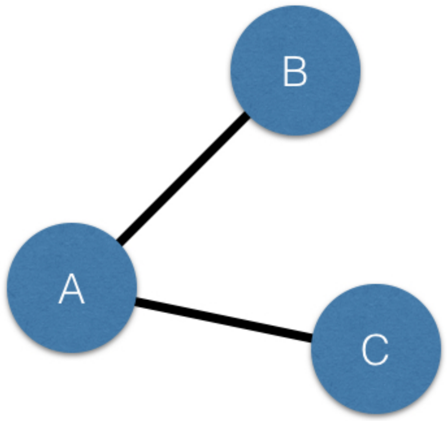


	A	B	C
A	□	■	■
B	■	□	□
C	■	□	□

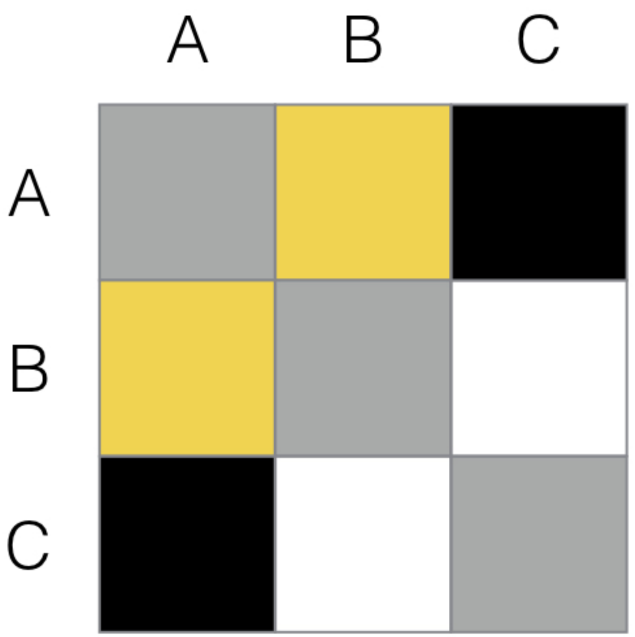


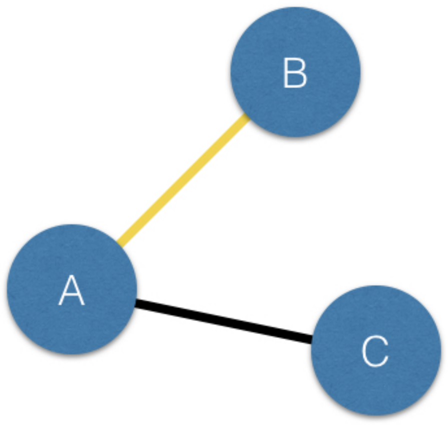
Matrix plot



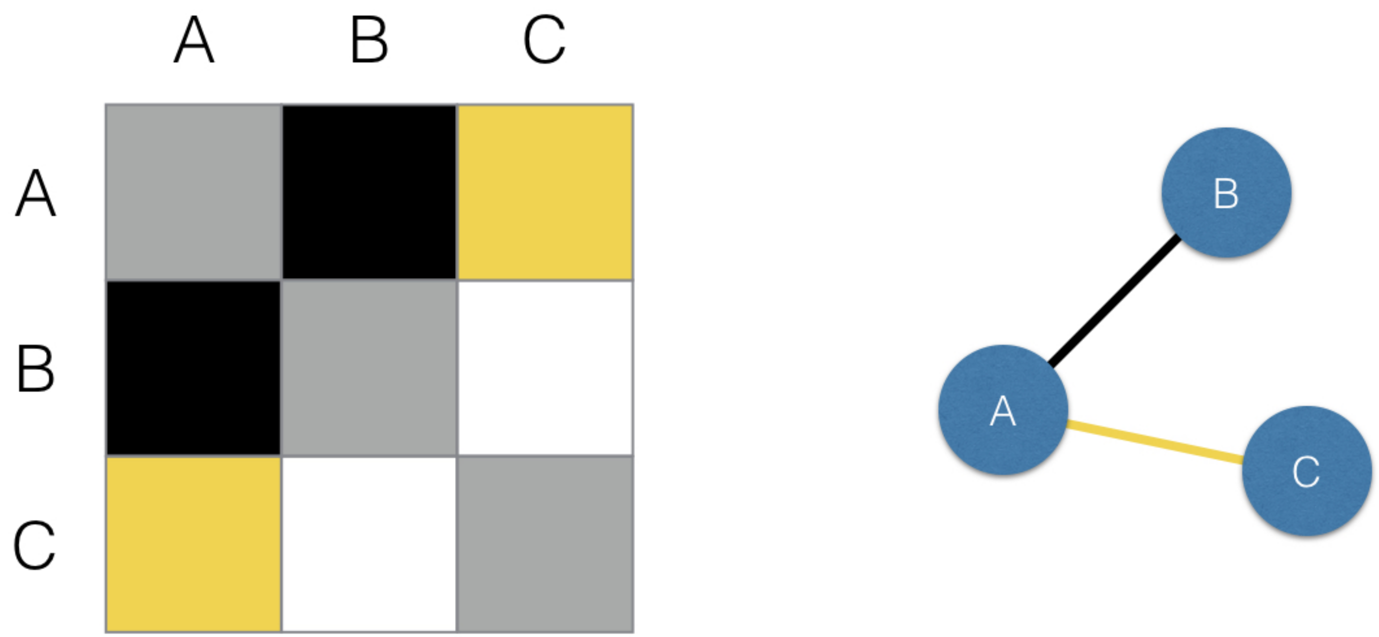


Matrix plot

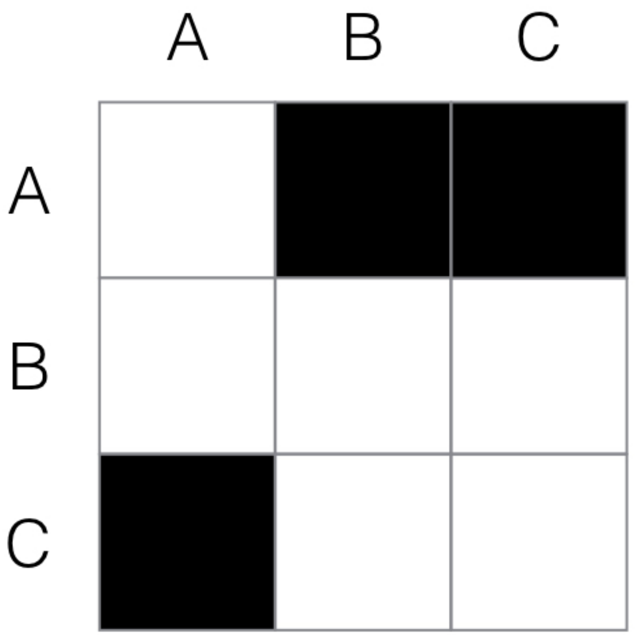




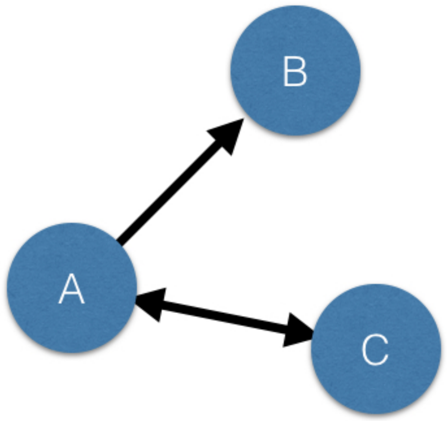
Matrix plot



Directed matrices



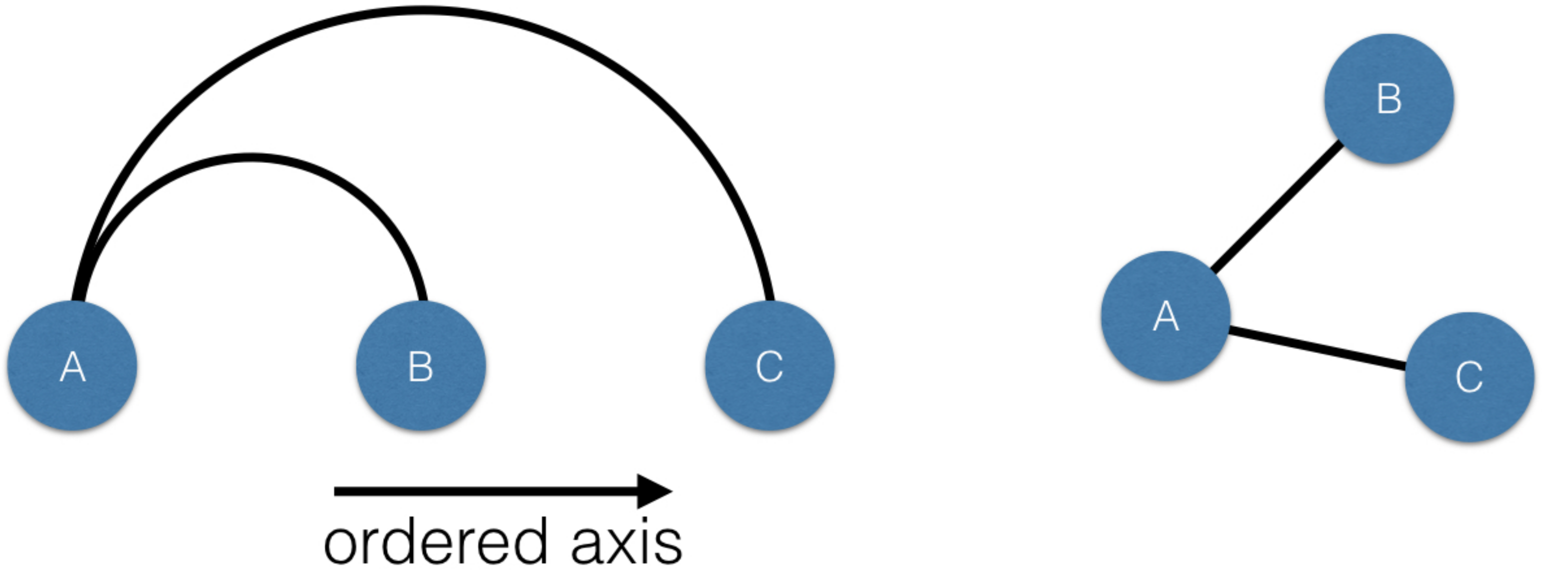
	A	B	C
A		■	■
B			
C	■		



Visualizing networks

- Matrix Plots
- **Arc Plots**
- Circos Plots

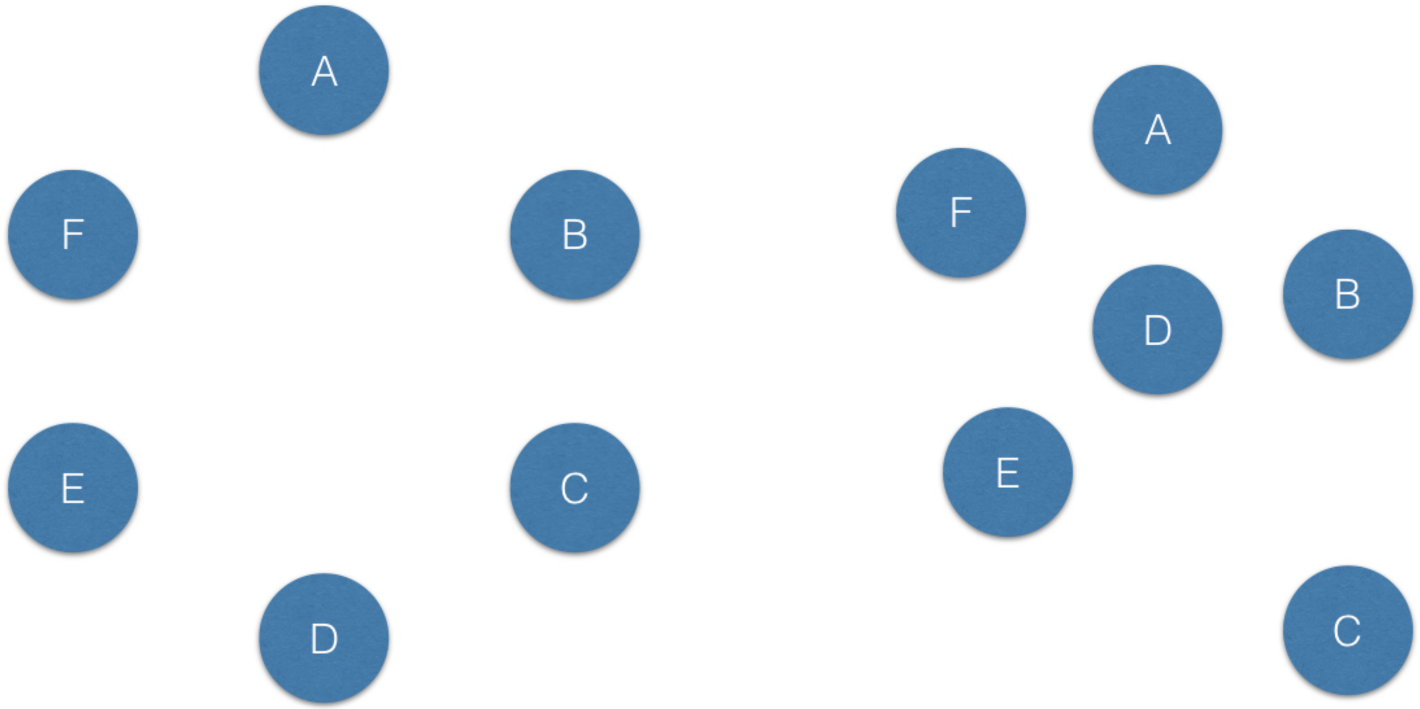
Arc plot



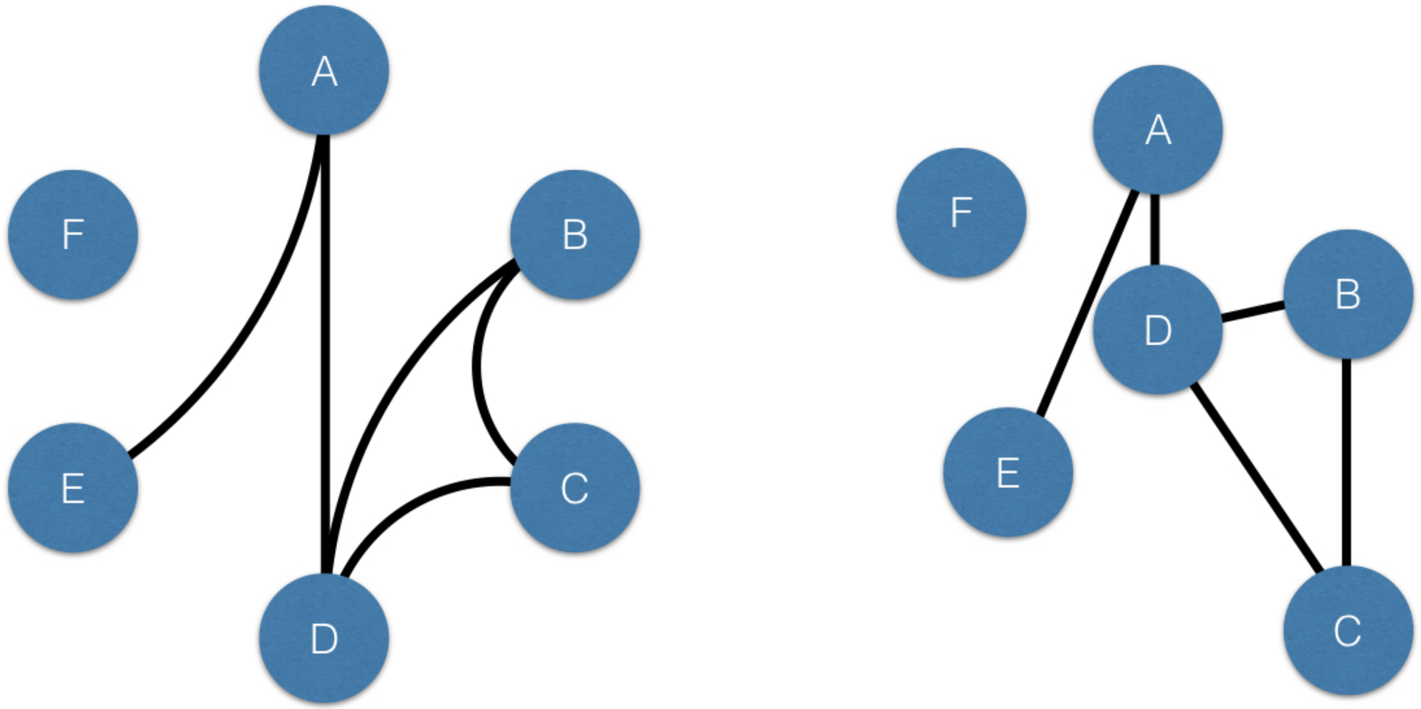
Visualizing networks

- Matrix Plots
- Arc Plots
- **Circos Plots**

Circos plot



Circos plot



nxviz API

```python
import nxviz as nv
import matplotlib.pyplot as plt

ap = nv.ArcPlot(G)

ap.draw()
plt.show()
```

Let's practice!

NETWORK ANALYSIS IN PYTHON (PART 1)