# Introduction to Networks

INTRODUCTION TO NETWORK ANALYSIS IN PYTHON

**Eric Ma**

Data Carpentry instructor and author of nxviz package

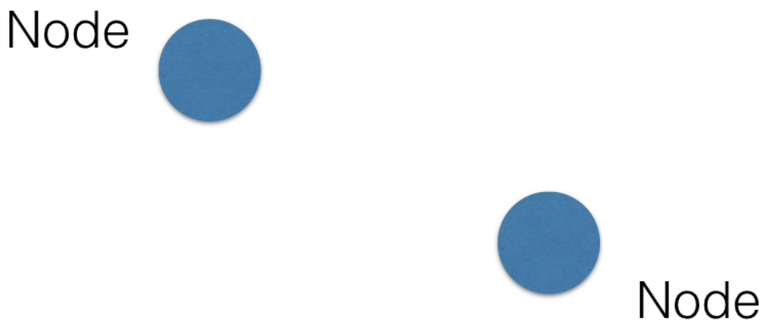
# Networks!

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

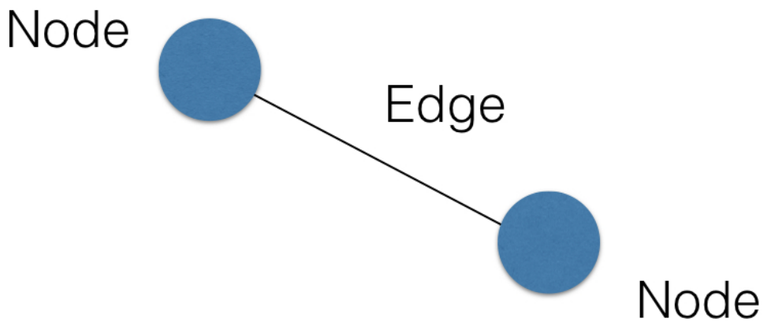
# Networks!

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

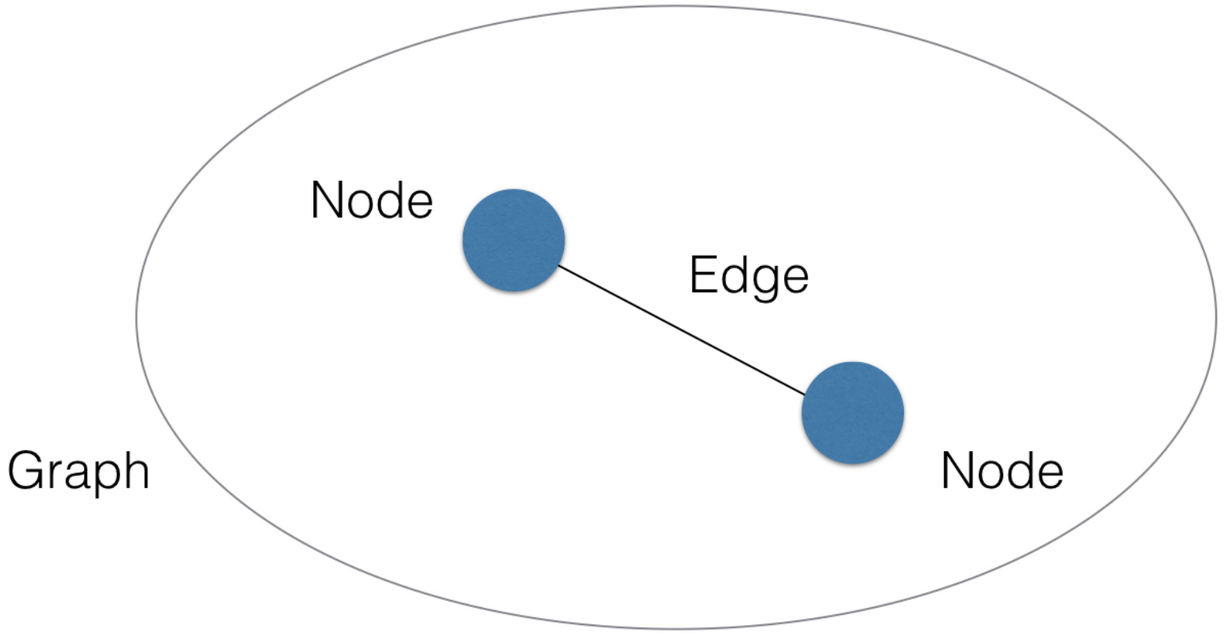
# Network Structure



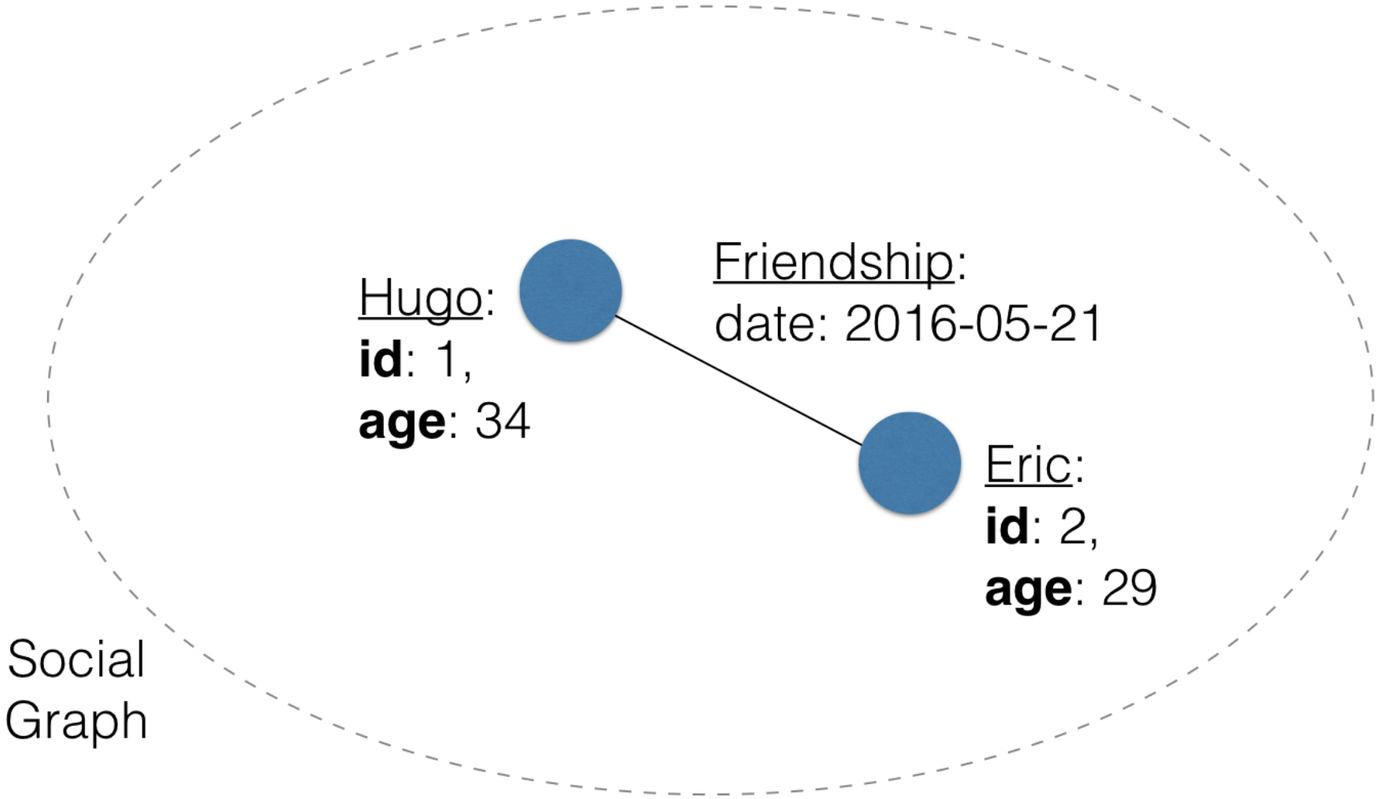
# Network Structure



# Network Structure



# Network Structure



# NetworkX API Basics

```
import networkx as nx
G = nx.Graph()
G.add_nodes_from([1, 2, 3])
G.nodes()
```

```
NodeView([1, 2, 3])
```

```
G.add_edge(1, 2)
G.edges()
```

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

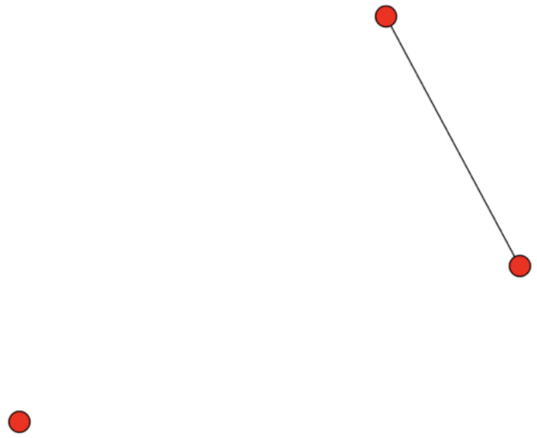
# NetworkX API Basics

```
G.nodes[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!

INTRODUCTION TO NETWORK ANALYSIS IN PYTHON

# Types of graphs

INTRODUCTION TO NETWORK ANALYSIS IN PYTHON

**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

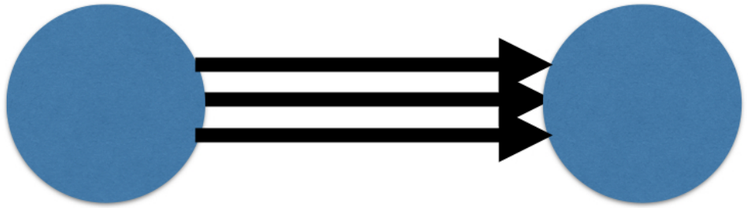
# Directed graphs

```
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)
```

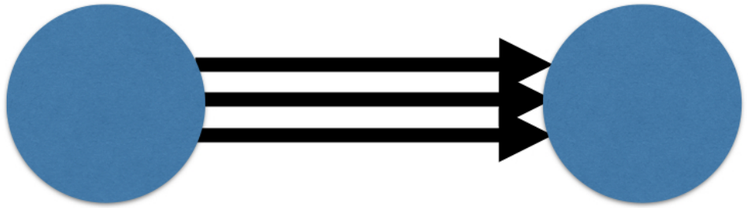
```
networkx.classes.multigraph.MultiGraph
```

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

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

# Weights on graphs

- Edges can contain weights

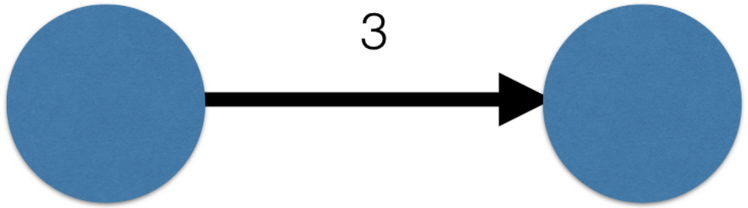


# Weights on graphs

- Edges can contain weights

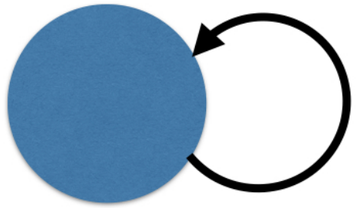
# Weights on graphs

- Edges can contain weights



# Self-loops

- Nodes that are connected to themselves



# Let's practice!

INTRODUCTION TO NETWORK ANALYSIS IN PYTHON

# Network visualization

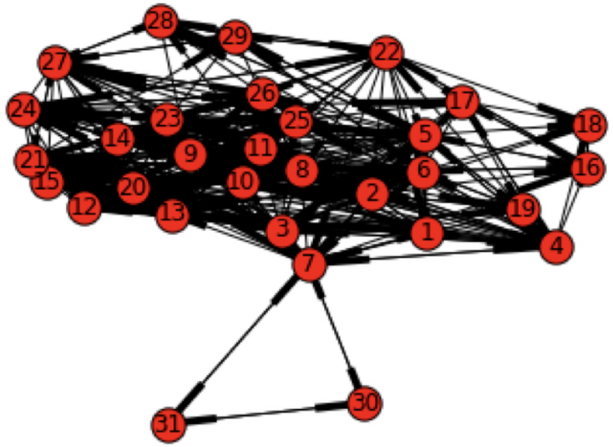
INTRODUCTION TO NETWORK ANALYSIS IN PYTHON

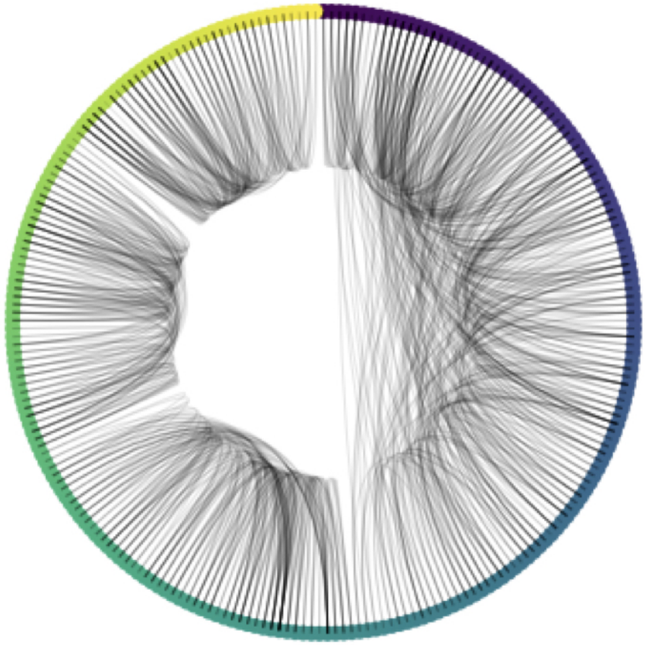
**Eric Ma**

Data Carpentry instructor and author of nxviz package

datacamp

# Irrational vs. Rational visualizations



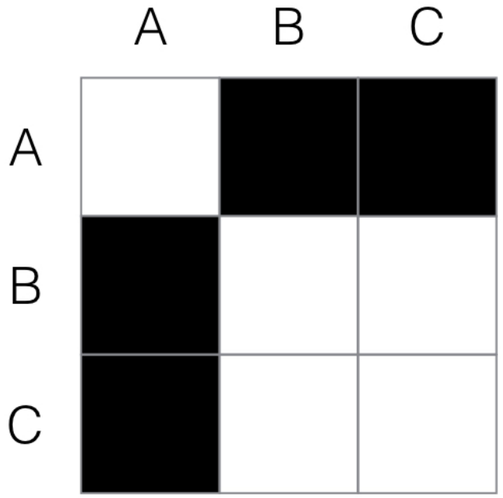
# Visualizing networks

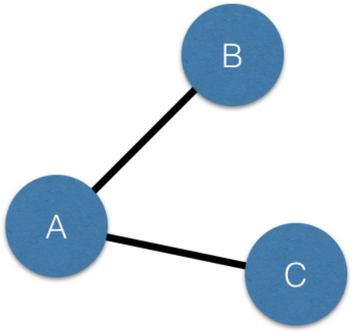
- Matrix plots
- Arc plots
- Circos plots

# Visualizing networks

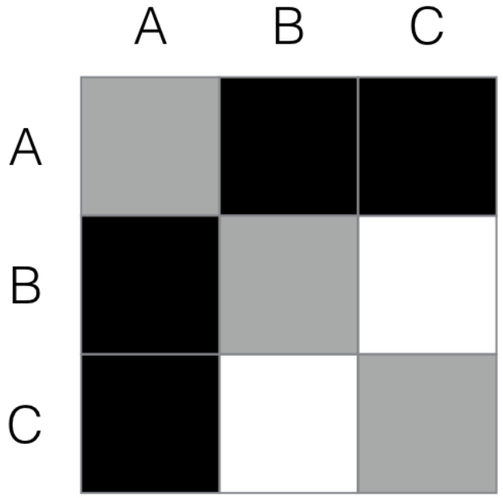
- **Matrix plots**
- Arc plots
- Circos plots

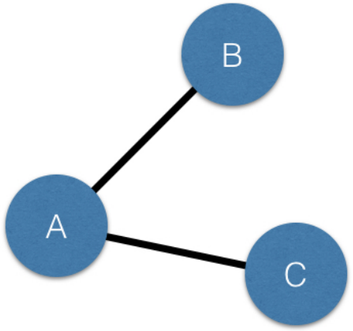
# Matrix plot



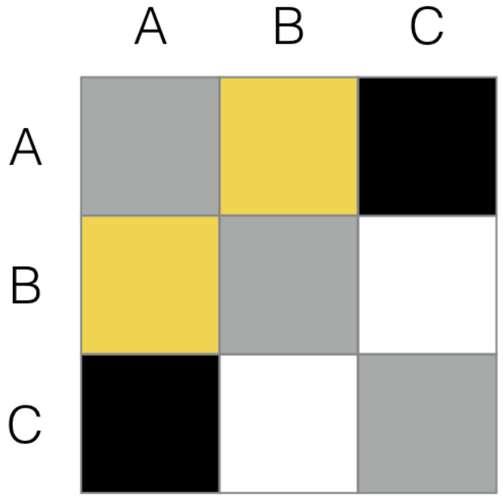



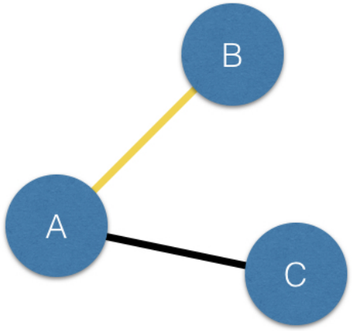
# Matrix plot



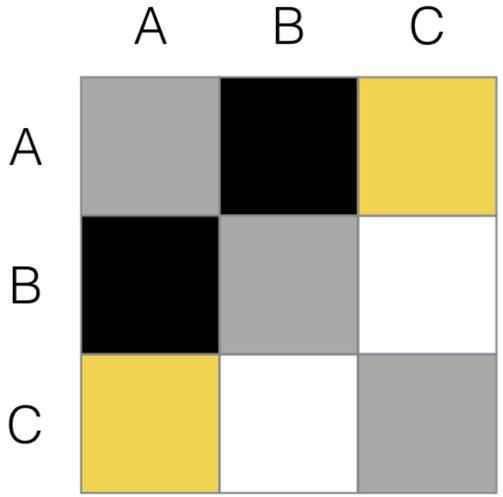



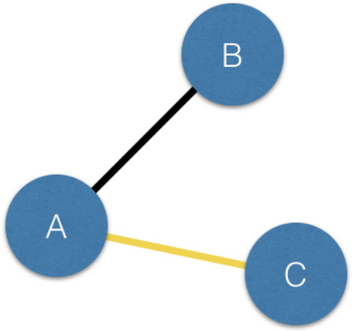
# Matrix plot



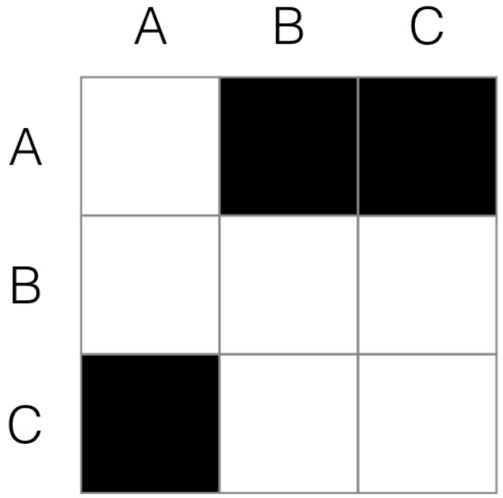



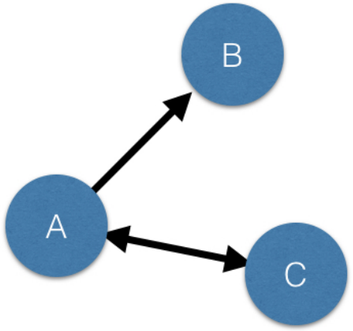
# Matrix plot



# Directed matrices

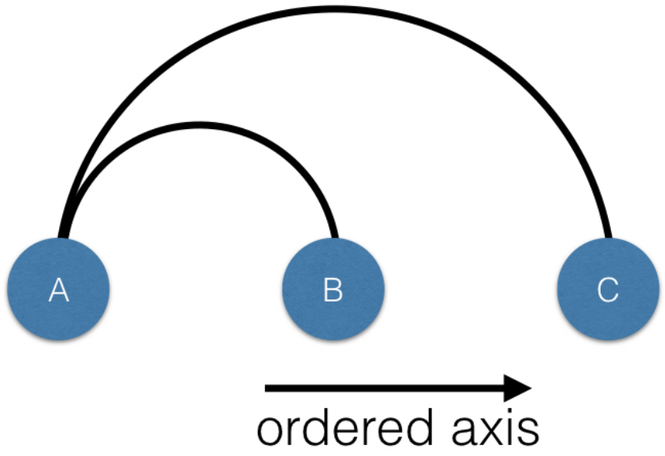


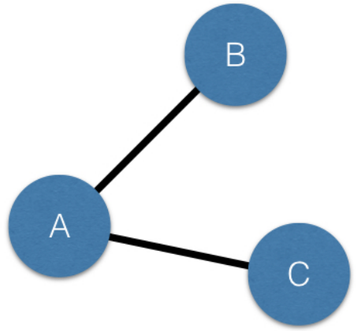



# Visualizing networks

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

# Arc plot

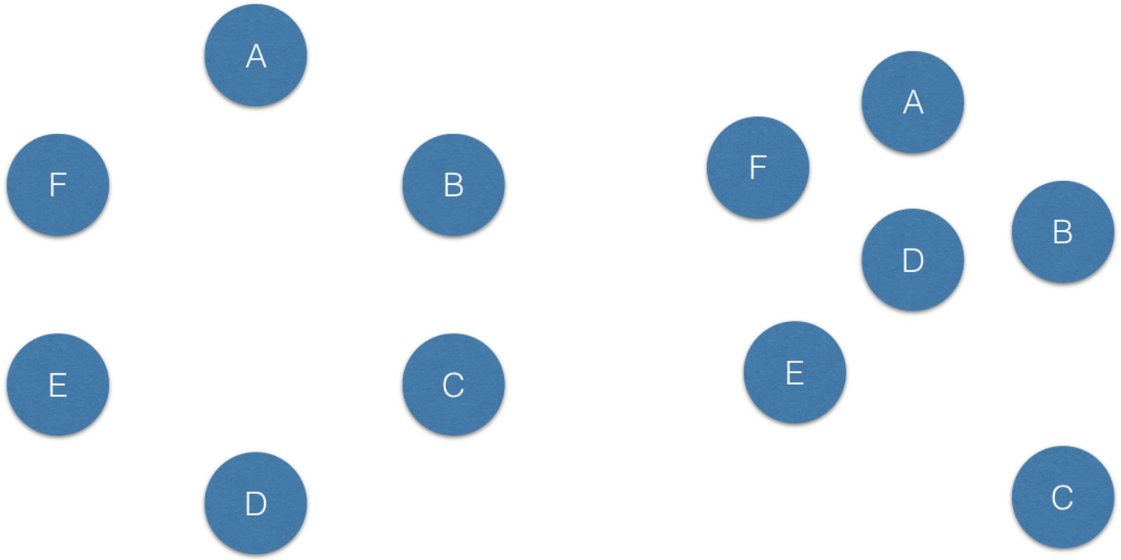


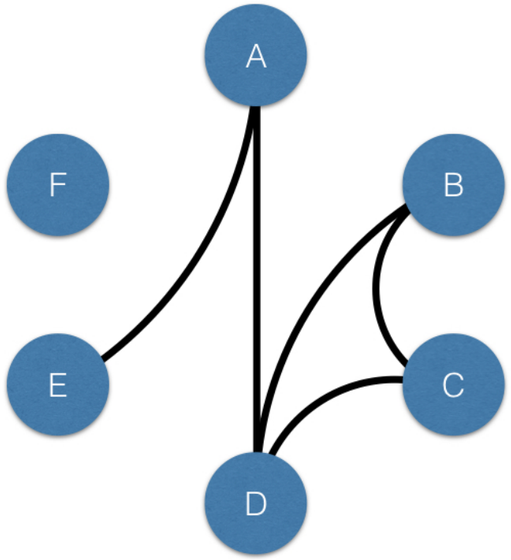
# Visualizing networks

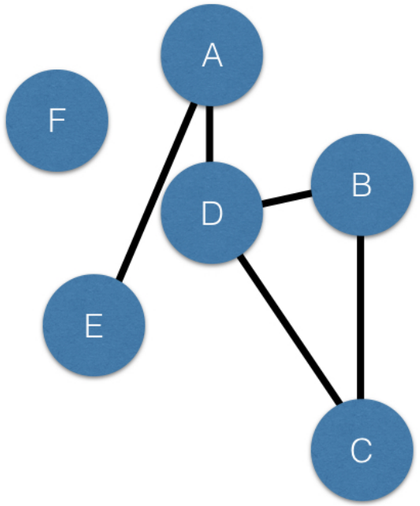
- Matrix plots
- Arc plots
- **Circos plots**

# Circos plot



# Circos plot



# nxviz API

```
import nxviz as nv
import matplotlib.pyplot as plt
ap = nv.arc(G)
plt.show()
```

# Let's practice!

INTRODUCTION TO NETWORK ANALYSIS IN PYTHON