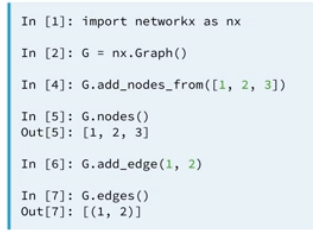
# Social Network Analysis

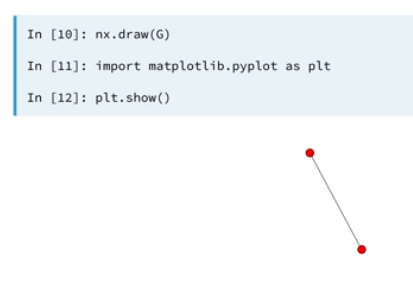
## Introduction to Networks

### Network Structure

* Node
* Edges are what connect the nodes

### NetworkX API basics

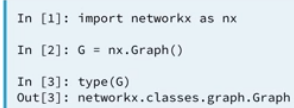


* You can assign metadata to the nodes
  + 
* You can also draw the graph
  + 

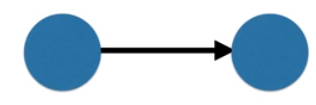
### Types of Graphs

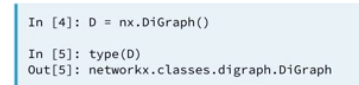
#### Undirected Graphs



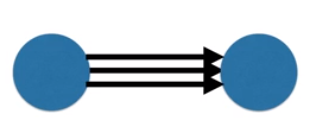


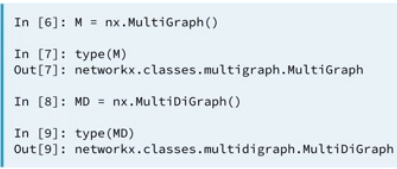
#### Directed graphs





#### Multi(Di)Graph



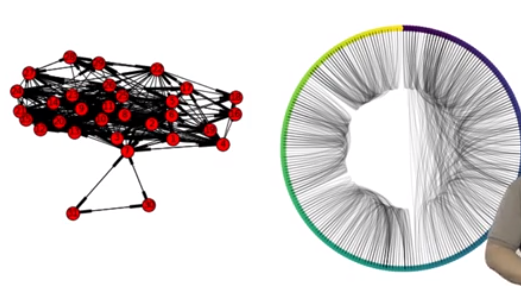


#### Weights on Graphs

* Edges can contain weights
* To set an attribute of an edge:
  + Network\_name.edge[node1][node2][‘attribute’] = value

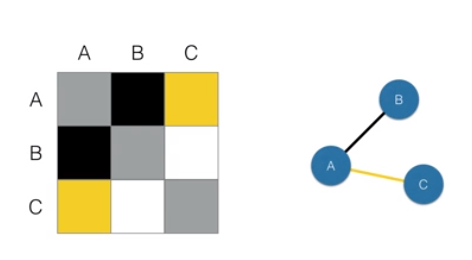
### Network visualization

#### Irrational vs. Rational Visualizations

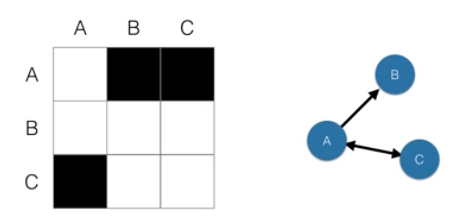


#### Matrix Plots

* If the edges were undirected:

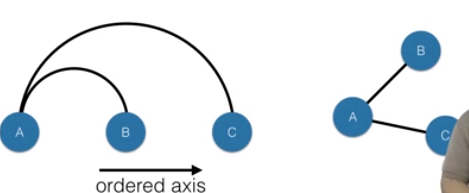


* If the edges were directed:

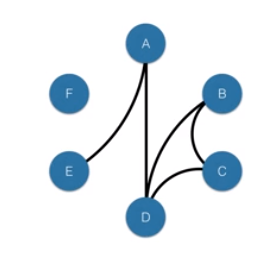


#### Arc plots

* They are a good starting point

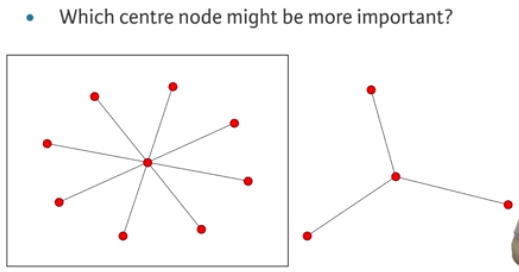


#### Circos plots

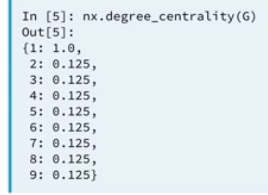


* We can use nxviz API to plot the circos plots
* 

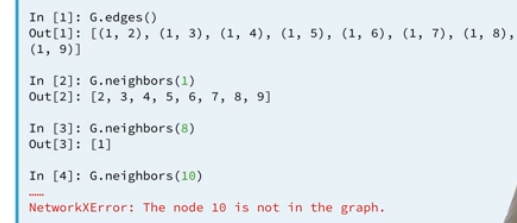
## Important Nodes



### Degree Centrality

* 
* 

#### Number of Neighbors



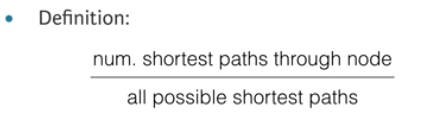
### Graph algorithms

#### Finding Paths

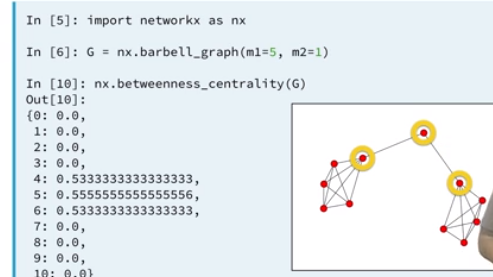
* Pathfinding is important for
  + Optimization: e.g. shortest transport paths
  + Modeling: e.g. disease spread, information passing

#### Breadth-first Search

### Betweenness Centrality

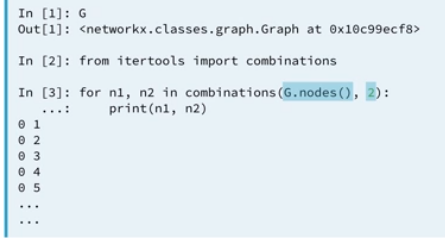


* Application:
  + Bridges between liberal and conservative learning twitter users

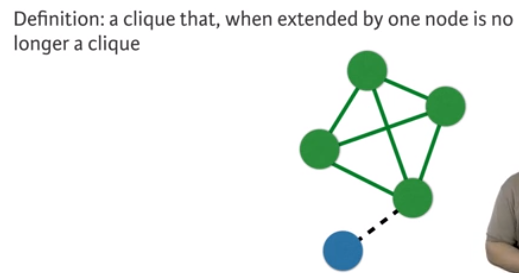


## Structures

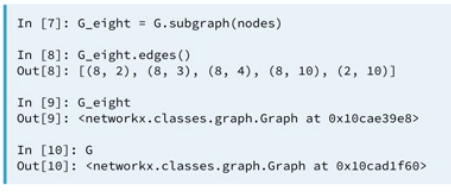
### Cliques & Communities



### Maximal Cliques

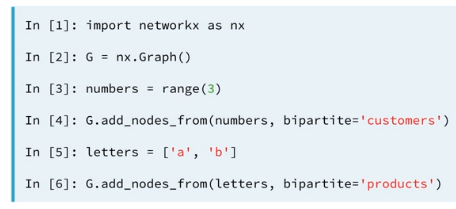
* A clique that, when extended by one node is no longer a clique
* 
* Applications
  + Communities
* Find\_cliques from NetworkX API will find all maximal cliques

### Subgraphs

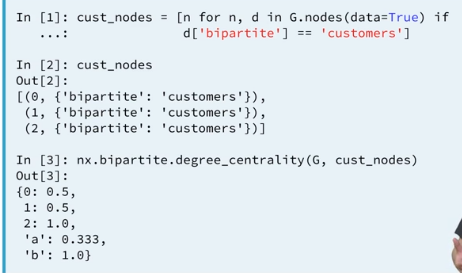


## Bipartite Graphs & Product Recommendation Systems

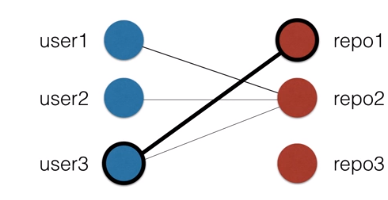
### Bipartite Graphs

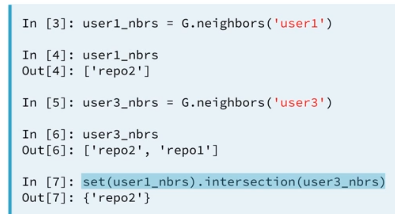
* A graph that is partitioned into two sets
* Node are only connected to nodes in other partitions
* In NetworkX
  + 

#### Bipartite Centrality Metrics

* Denominator: number of nodes in opposite partition, rather than all other nodes
* 

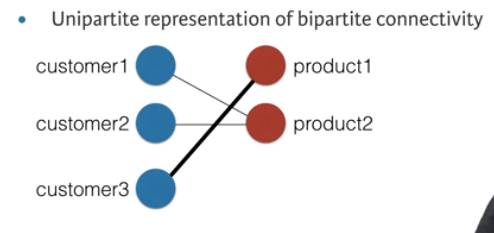
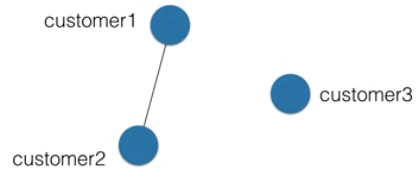
### Bipartite graphs and recommendation systems

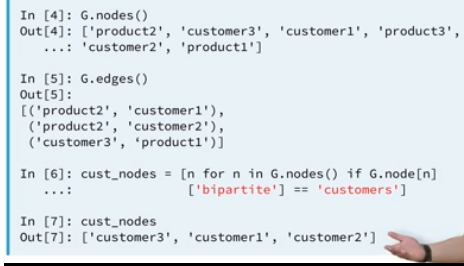


* User1 contributes to repo2, so we see who else contributes to repo2.
* We recommend to user1 to contribute to repo1 because user3 contributes to repo1 and repo2.
* 

## Graph Projections

### Concept of projection

* Projects are the connectivity of customers based on shared purchases.
* It’s useful to investigate the relationships between the nodes on one partition
  + Conditioned on the connections to the nodes in the other partition
* 
  + Results in a customer projection of
    - 

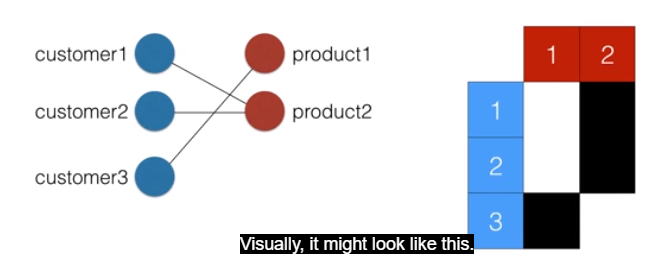


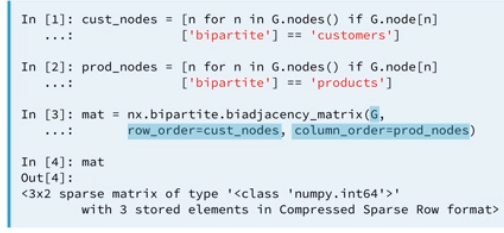


### Bipartite graphs as matrices

#### Matrix Representation

* Rows: nodes on one partition
* Columns: nodes on other partition
* Cells: 1 if edge present, else 0





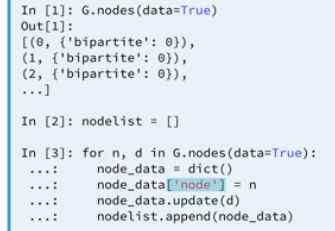
* Project is computed using matrix multiplication.
  + Matrix \* transposed matrix = projection

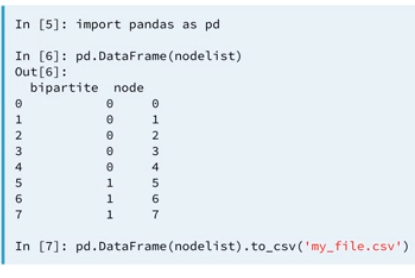
### Representing network data with pandas

#### CSV Files for network data storage

* Advantages
  + Human-readable
  + Do further analysis with pandas
* Disadvantages
  + Repetitive; disk space

#### Graph conversion to CSV files





## Comparing Graphs & Time-Dynamic Graphs

### Introduction to graph differences

#### Evolving Graphs

* Graphs that change over time: communication networks
* Assumptions:
  + Edge changes over time; assume nodes stay constant
  + Both edges and nodes change over time

#### Graph Differences

* In NetworkX: .difference(G2, G2) function
  + Assumes G1 and G2 have equal node sets

### Evolving graph statistics

* Graph summary statistics
  + Number of nodes
  + Number of edges
  + Degree distribution
  + Centrality distributions
* For simple metrics, use edgelist data
* For graph theoretic metrics, use graph object

##### Cumulative Distribution

* Compact way of representing the distribution of values

### Zooming in & zooming out: overall graph summary

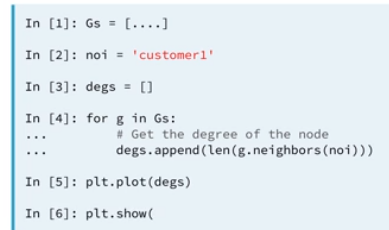
#### Graph Exploration at Scales

* Exploration at global and local scales
* Global: Centrality distributions
* Local: Connectivity and structures

#### Zooming on nodes

* Isolate a given node or set of nodes
* Plot node statistic over time

#### Summarizing Evolving Node Statistics

* Customer-product dataset
  + Investigate how purchasing patterns have changed over time
* ‘customer1’ node of interest
* 

# Hot Topics in Machine Learning

# Advanced Data Visualization