

Introduction to graph differences

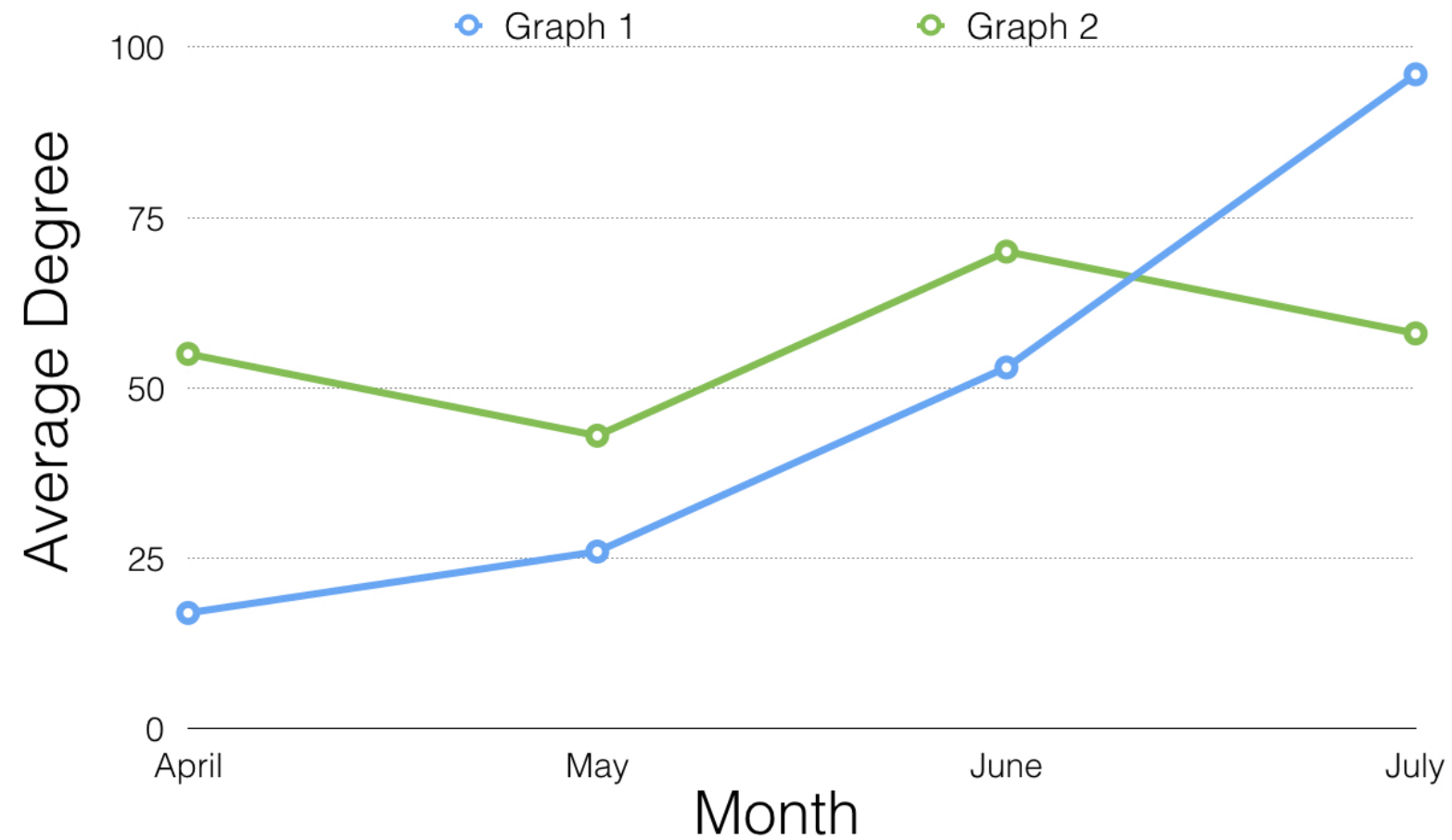
INTERMEDIATE NETWORK ANALYSIS IN PYTHON



Eric Ma

Data Carpentry instructor and author of
nxviz package

Time series analysis



Time series analysis

- How some number changes as a function of time
 - Is there an upward or downward trend?
- Rate of change of things over a sliding window of time
- Examples:
 - Tracking weight over time
 - Tracking stock investment portfolio value over time

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

- Graphs are comprised of:
 - A node set
 - An edge set
- If a node set doesn't change:
 - Changing only the edge set will result in a change in the graph

Graph differences

- Analogy: set differences

```
set(c1, c2, c3).difference(set(c2, c3, c4)) = set(c1)
```

```
set(c2, c3, c4).difference(set(c1, c2, c3)) = set(c4)
```

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

Graph differences in Python

```
G1.edges()
```

```
[('cust1', 'cust2'), ('cust3', 'cust2')]
```

```
G2.edges()
```

```
[('cust1', 'cust3'), ('cust3', 'cust2')]
```

```
G2minusG1 = nx.difference(G2, G1)
```

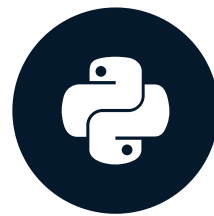
```
G1minusG2 = nx.difference(G1, G2)
```

Let's practice!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON

Evolving graph statistics

INTERMEDIATE NETWORK ANALYSIS IN PYTHON



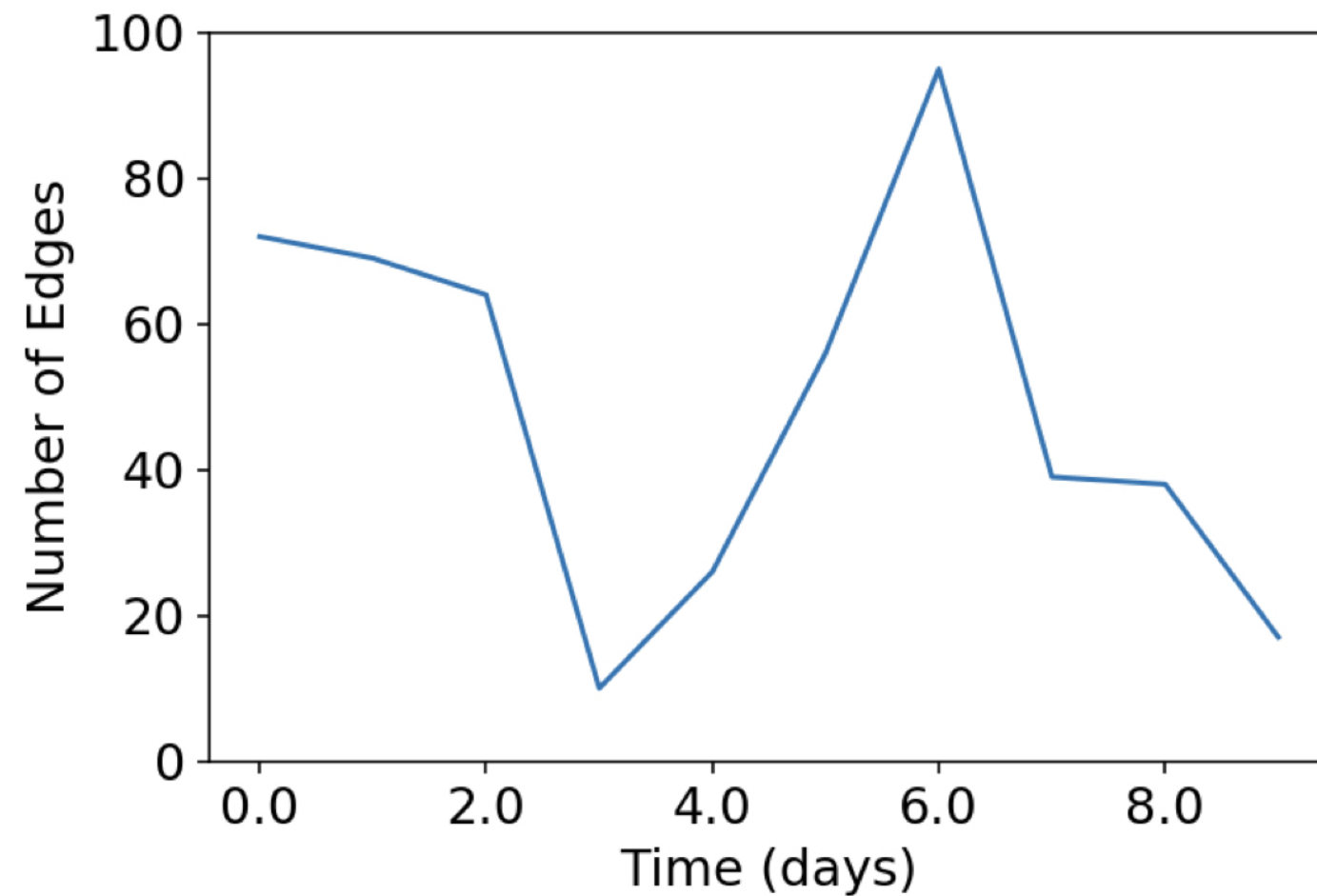
Eric Ma

Data Carpentry instructor and author of
nxviz package

Evolving graph statistics

- Graph summary statistics:
 - Number of nodes
 - Number of edges
 - Degree distribution
 - Centrality distributions

Evolving graph statistics



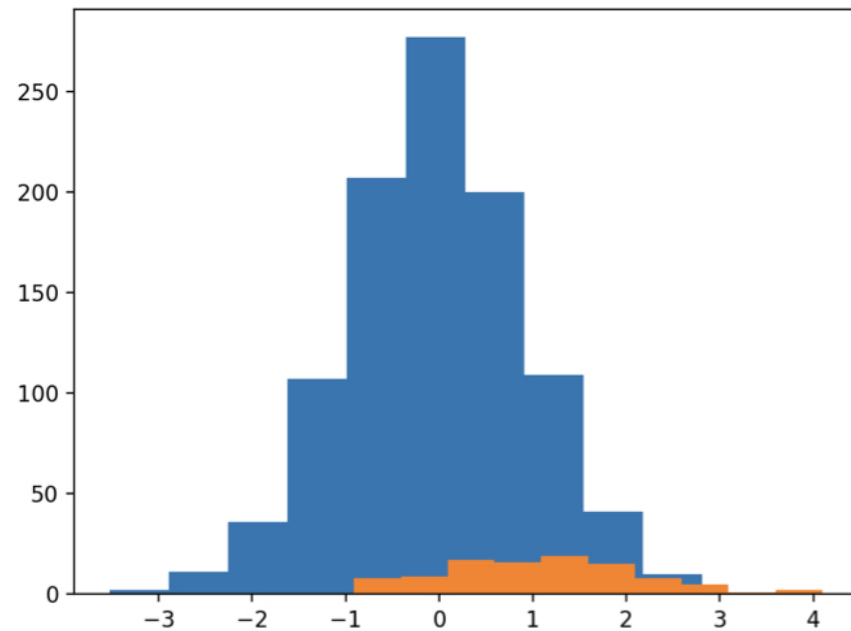
Evolving graph statistics

- For simple metrics, use edgelist data
- For graph theoretic metrics, use graph object

Cumulative distribution

- Compact way of representing the distribution of values

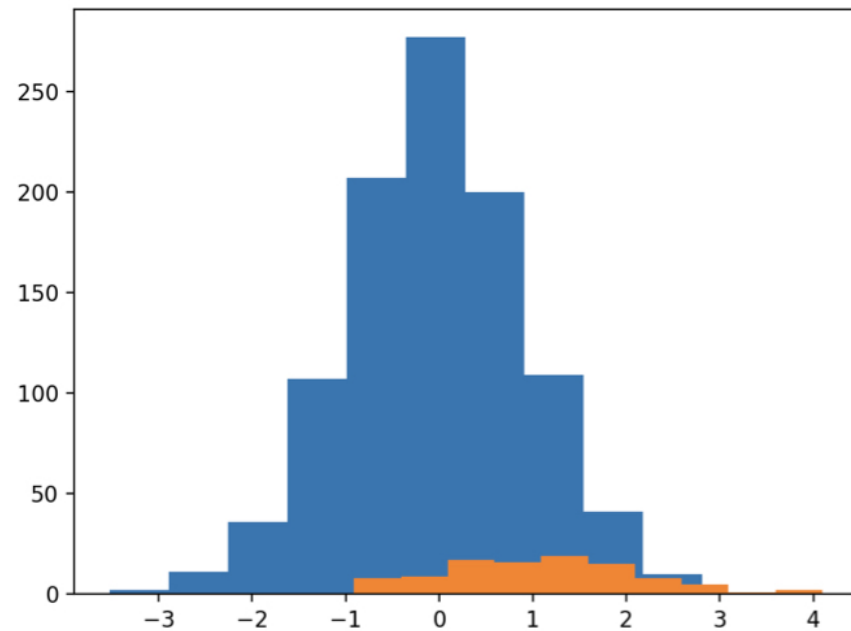
Histogram



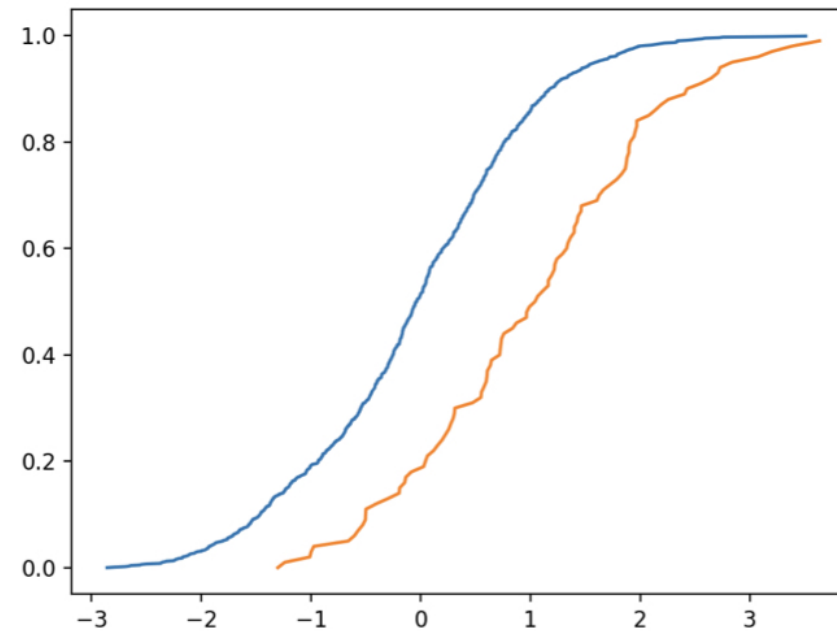
Cumulative distribution

- Compact way of representing the distribution of values

Histogram



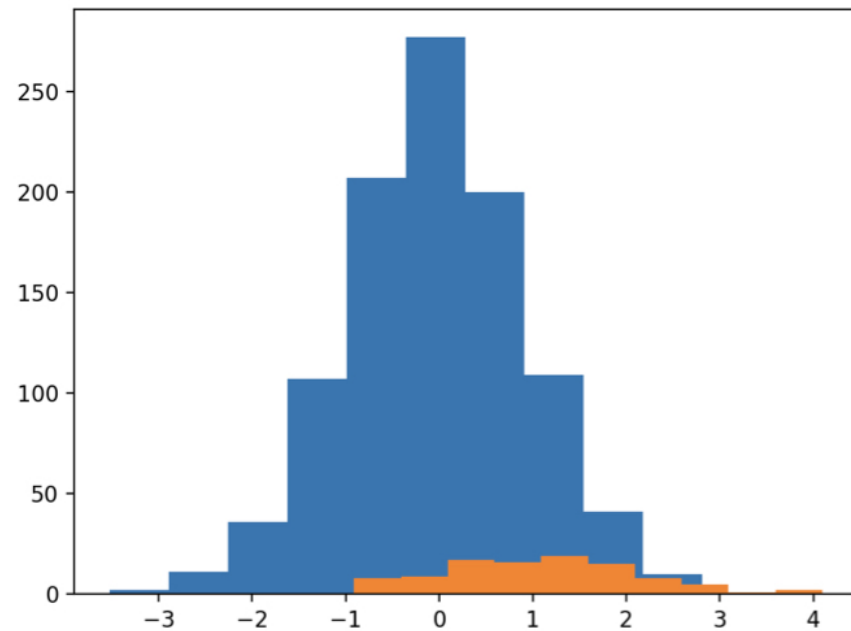
Cumulative Dist.



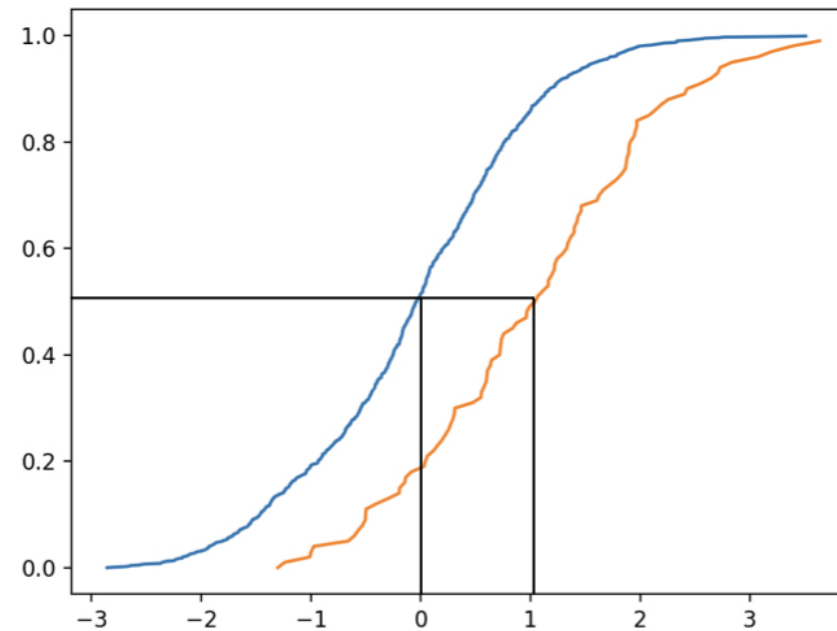
Cumulative distribution

- Compact way of representing the distribution of values

Histogram



Cumulative Dist.

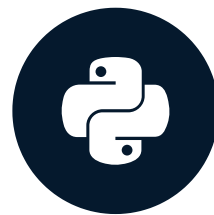


Let's practice!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON

Zooming in & zooming out: Overall graph summary

INTERMEDIATE NETWORK ANALYSIS IN PYTHON



Eric Ma

Data Carpentry instructor and author of
nxviz package

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

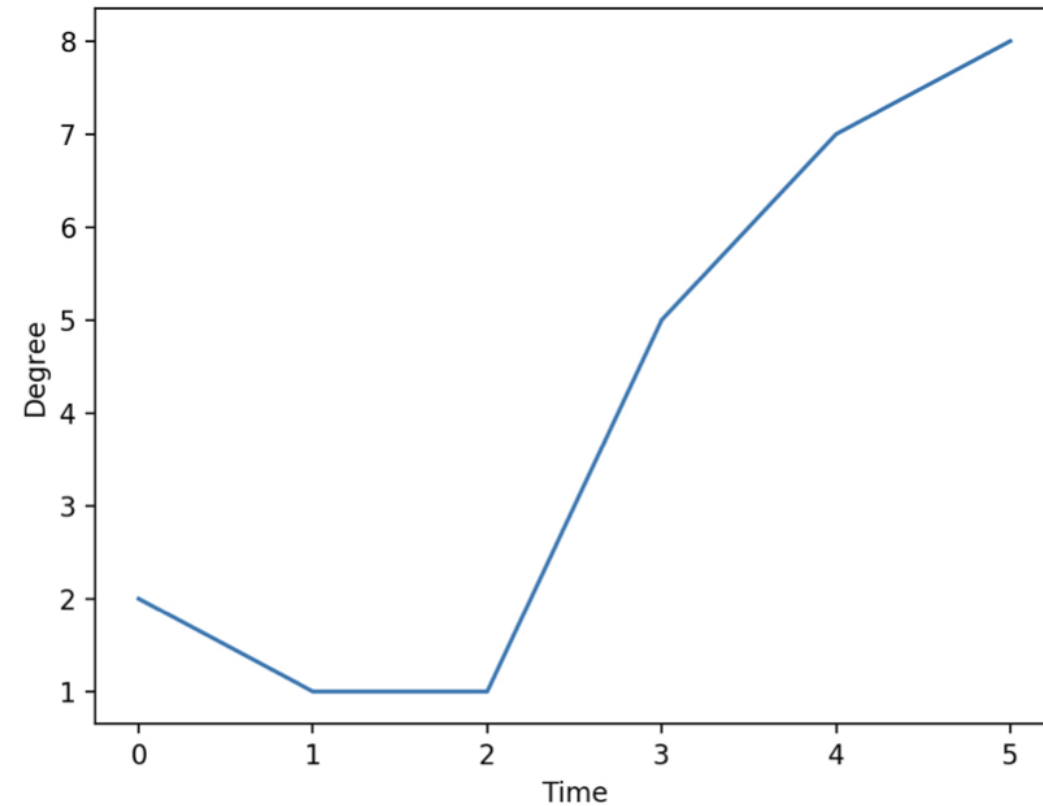
Summarizing evolving node statistics

- Customer-product dataset
 - Investigate how purchasing patterns have changed over time
- `customer1` - node of interest

Summarizing evolving node statistics

```
Gs = [...]\nnoi = 'customer1'\ndegs = []\nfor g in Gs:\n    ...           # Get the degree of the node\n    degs.append(len(g.neighbors(noi)))\nplt.plot(degs)\nplt.show()
```

Summarizing evolving node statistics



Default dictionaries

```
from collections import defaultdict
d = defaultdict(list)
d['heathrow'].append(0.31)
d['heathrow'].append(0.84)
d
```

```
defaultdict(list, {'heathrow': [0.31, 0.84]})
```

Default dictionaries

```
d2 = dict()  
d2['heathrow'].append(0.31)
```

```
KeyError                                Traceback (most recent call last)  
<ipython-input-19-291c74368a8f> in <module>()  
--> 1 d2['heathrow'].append(0.31)  
KeyError: 'heathrow'
```


Let's practice!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON