

Introduction to the dataset

INTERMEDIATE NETWORK ANALYSIS IN PYTHON



Eric Ma

Data Carpentry instructor and author of
nxviz package

Dataset & case study introduction

- College forum posting dataset, 6 months
- Node partitions: students, forums
- Activities in the chapter:
 - Constructing a graph from a pandas DataFrame
 - Computing unipartite projections of a bipartite graph
 - Visualization
 - Time series filtering & analysis
- Recap previously used functions

Graphs from DataFrames

```
df
```

```
  customers  products  
0  customerA  product1  
1  customerB  product2  
...
```

```
G = nx.Graph()  
G.add_nodes_from(df['products'], bipartite='products')  
G.add_nodes_from(df['customers'], bipartite='customers')  
G.nodes()
```

```
['product1', 'customerC', 'product2', 'customerB', 'customerA']
```

```
G.edges()
```

```
[]
```

Graphs from DataFrames

```
G.add_edges_from(zip(df['customers'], df['products']))  
G.edges()
```

```
[('product1', 'customerC'), ('product1', 'customerA'),  
 ('customerC', 'product2'), ('product2', 'customerB')]
```

Bipartite projections

```
cust_nodes = [n for n in G.nodes() if G.node[n]
               ['bipartite'] == 'customers']
prod_nodes = [n for n in G.nodes() if G.node[n]
               ['bipartite'] == 'products']
prodG = nx.bipartite.projected_graph(G, nodes=prod_nodes)
custG = nx.bipartite.projected_graph(G, nodes=cust_nodes)
prodG.nodes()
```

```
['product1', 'product2']
```

```
custG.nodes()
```

```
['customerC', 'customerB', 'customerA']
```

Let's practice!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON

Time based filtering

INTERMEDIATE NETWORK ANALYSIS IN PYTHON



Eric Ma

Data Carpentry instructor and author of
nxviz package

Key concepts

- Filtering graphs
- Datetime
- Visualization

Filtering edges

```
G.edges(data=True)[0:5]
```

```
[(0, 17, {'sale_count': 1}),  
 (0, 18, {'sale_count': 1}),  
 (0, 19, {'sale_count': 2}),  
 (0, 12, {'sale_count': 14}),  
 (0, 13, {'sale_count': 9})]
```

```
[(u, v) for u, v, d in G.edges(data=True) if d['sale_count'] >= 10]
```

```
[(0, 12), (1, 19), (5, 16), (6, 13), (7, 17), (7, 19), (8, 18)]
```

Datetime

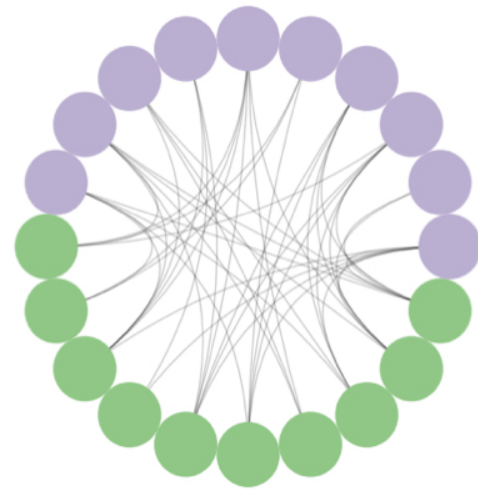
```
from datetime import datetime, timedelta  
year = 2011  
month = 11  
day1 = 10  
day2 = 6  
date1 = datetime(year, month, day1)  
date2 = datetime(year, month, day2)  
date1 > date2
```

True

Graph visualization

```
from nxviz import CircosPlot
c = CircosPlot(G, node_grouping='bipartite',
               node_color='bipartite')

c.draw()
plt.show()
```



Let's practice!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON

Time series analysis

INTERMEDIATE NETWORK ANALYSIS IN PYTHON



Eric Ma

Data Carpentry instructor and author of
nxviz package

Time series

- Global vs. local analysis
- Analyze evolving graph statistics
- Make plots of key evolving stats

Datetime arithmetic

```
date1
```

```
datetime.datetime(2011, 11, 10, 0, 0)
```

```
days = 4  
td = timedelta(days)  
date1 + td
```

```
datetime.datetime(2011, 11, 14, 0, 0)
```

Degree centrality

```
G
```

```
<networkx.classes.graph.Graph at 0x10e7c04a8>
```

```
nx.degree_centrality(G)
```

```
{1: 0.5, 'c': 0.5, 'b': 0.25, 2: 0.5, 'a': 0.25}
```

```
nx.bipartite.degree_centrality(G, [1, 2])
```

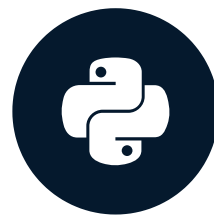
```
{1: 0.6666666666666666, 2: 0.6666666666666666,  
'b': 0.5, 'c': 1.0, 'a': 0.5}
```


Let's practice!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON

Congratulations!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON

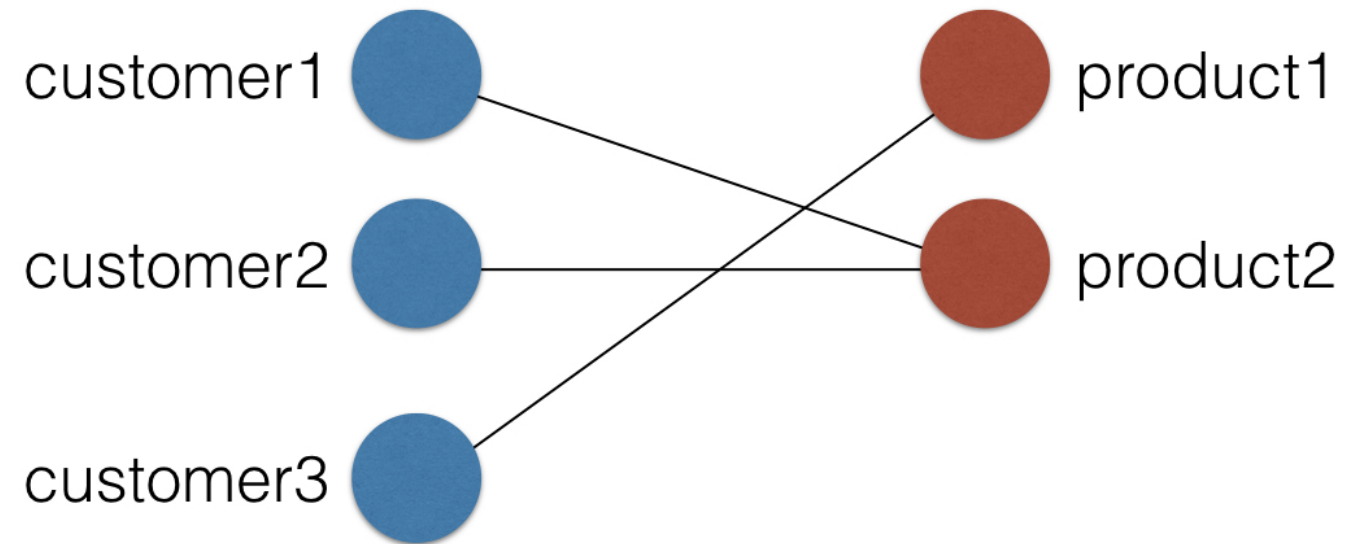


Eric Ma

Data Carpentry instructor and author of
nxviz package

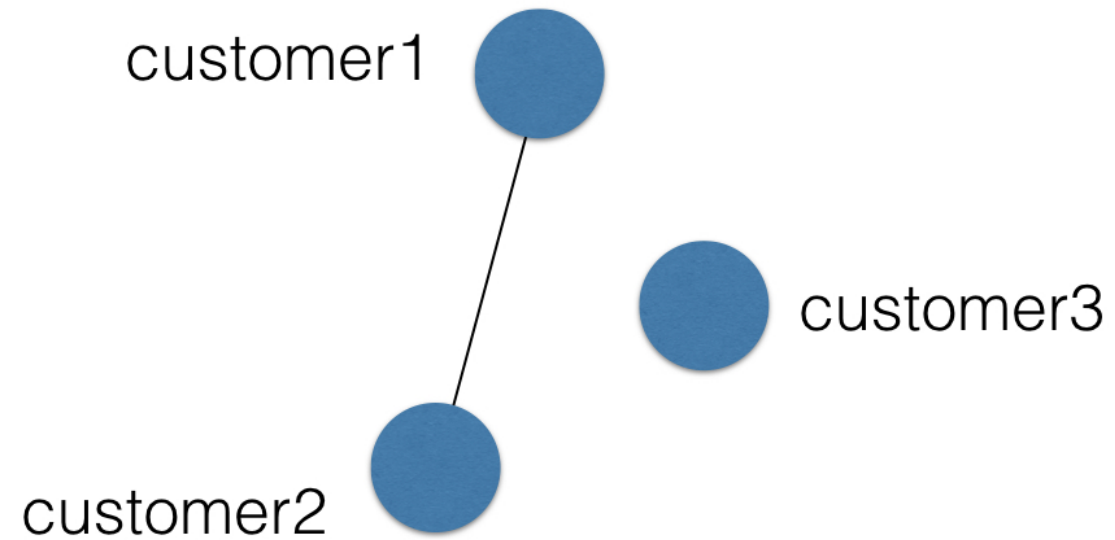
What you've learned

- Bipartite graphs



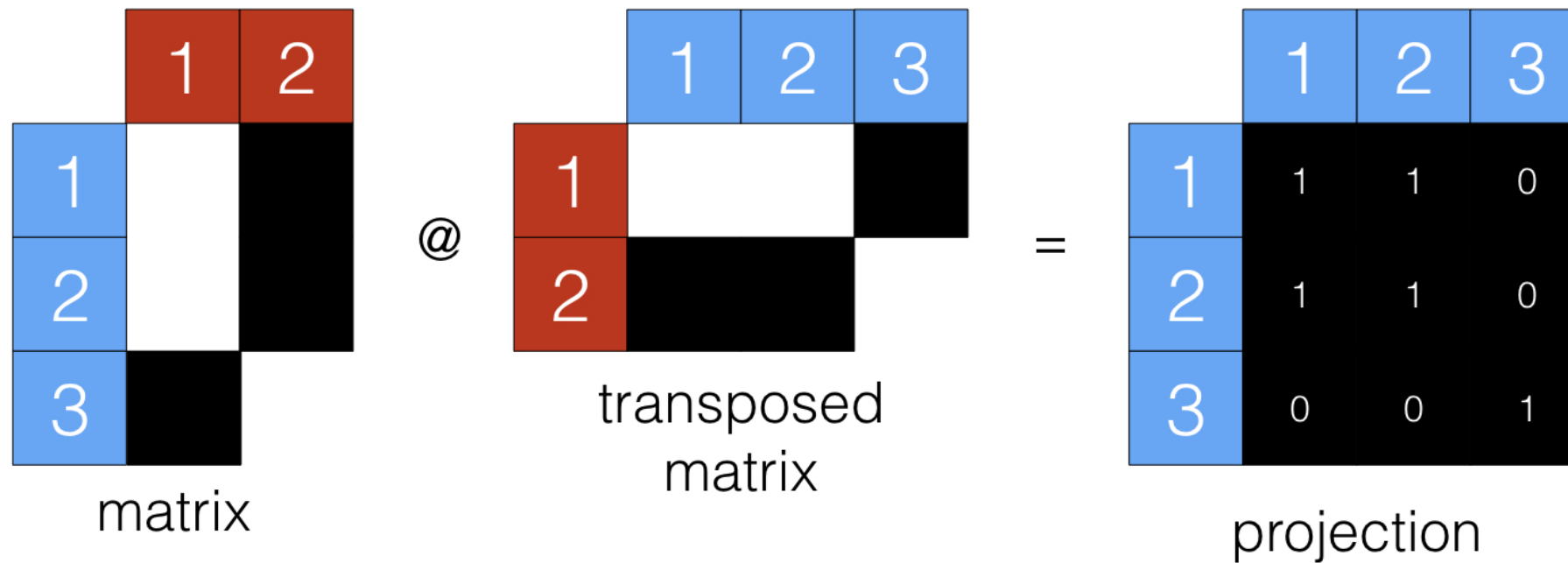
What you've learned

- Bipartite graphs
- Projections



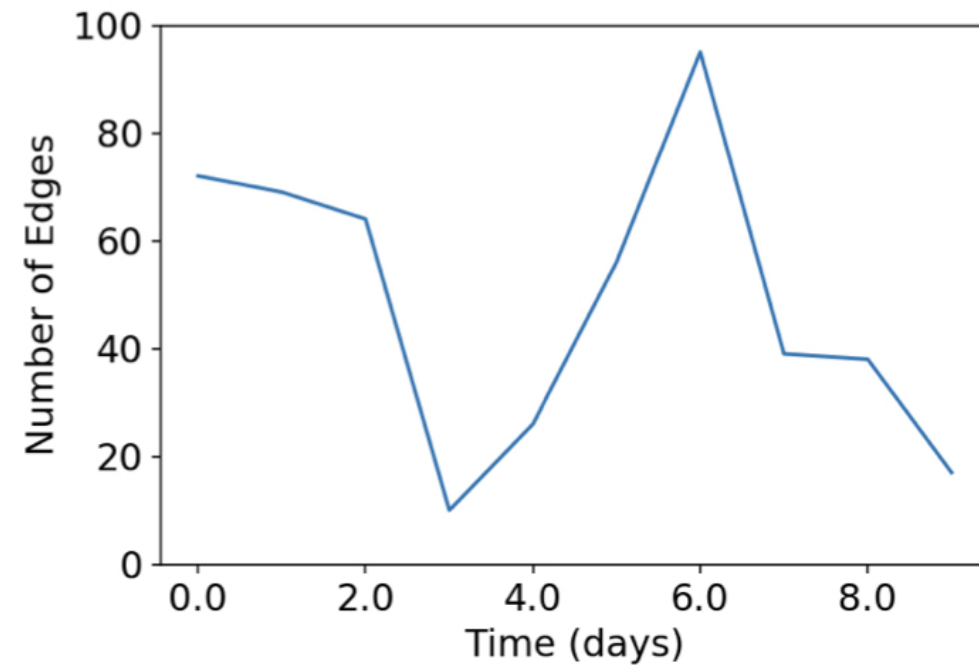
What you've learned

- Bipartite graphs
- Projections
- Matrix representation



What you've learned

- Bipartite graphs
- Projections
- Matrix representation
- Time series



Let's practice!

INTERMEDIATE NETWORK ANALYSIS IN PYTHON