

# **Learning Objectives**

- Understand what a network graph is and how it is constructed
- Be able to transform raw data into a format suitable for conducting a network analysis
- Know how to create a simple network graph in Python using NetworkX

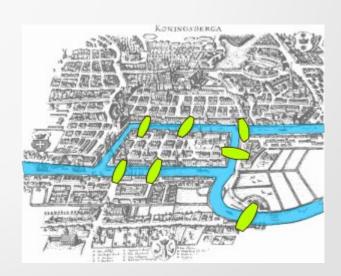
#### What is a network?



Used by Euler in 1930s for logistical problems

Network graphs are used to:

- Study the function of large and/or complex systems
- Visualise relationships and interactions in a intuitive way
- Represent abstract concepts in a concrete way



# Examples

#### **EXAMPLES OF NETWORKS AND THEIR COMPONENTS**

NETWORK	VERTICES	VERTEX ATTRIBUTES	EDGES	EDGE ATTRIBUTES
Airlines Network	Airports	Footfall, Terminals, Staff, City population, International/Domestic, Freight, Hangar capacity	Airplanes / Routes	Frequency, # Passengers, Plane Type, Fuel Usage, Distance covered, Empty seats
Banking Network	Account Holders	Name, demographics, KYC Document, Products, Account status, balance and other details	Transactions	Type, Amount, Authentication (pass/OTP), Time, Location, Device
Social Network	Users	Name, demographics, # connections, likes, circles belong to, subscriptions	Interactions	Medium (like/comment/direct message), time, duration, type of content, topic
Physician Network	Doctors	Demographics, speciality, experience, affiliation (type and size), Weekly patient intake	Patients	Demographics, Diagnosis history, visit frequency, purpose, referred to, insurance
Supply Chain Network	Warehouses	Location, size, capacity, storage type, connectivity, manual/automated	Trucks	Load capacity, # wheels, year of make, geographical permit, miles travelled. Maintenance cost, driver experience

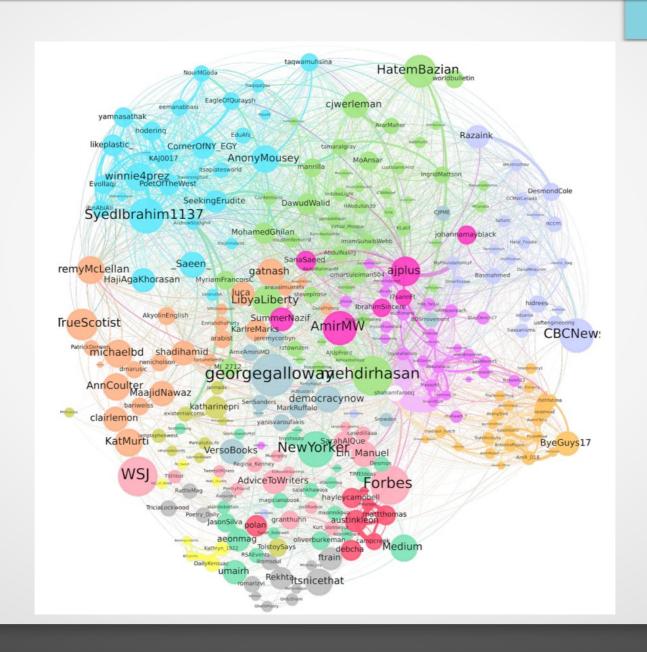


Proprietary and Confidential

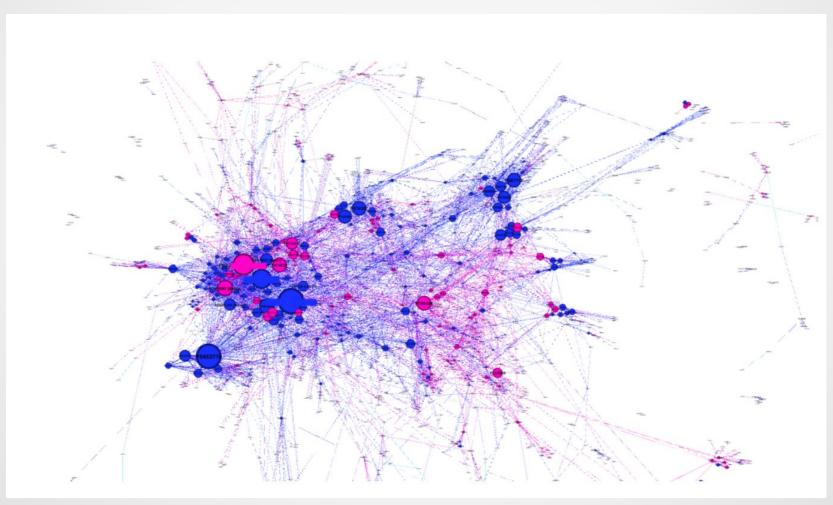
#### Credit:

https://www.analyticsvidhya.com/blog/2018/04/introduction-to-graph-theory-network-analysis-python-codes/

# Example 1 – Twitter Handles

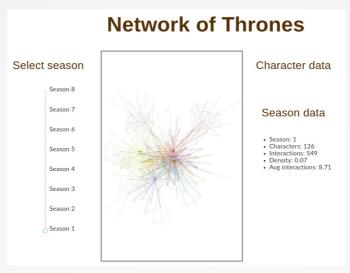


# Example 2 – HSMA: Offenders and Victims

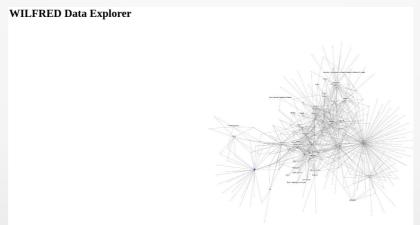


Credit: Jenna Thomas and Charley Bartlett, Devon and Cornwall Police

# Interactive Examples



#### https://got-network-app.herokuapp.com/



https://wilfred-data-explorer.herokuapp.com/

#### Network-based operational modelling for healthcare



Size + colour of nodes is different as we can embed attributes

- Graph as service structure
- Node as service
- Edge as patient movement
- Edge weight as number of patients
- Node attributes as improvement measures

**Arrows** illustrated 'direction'

Operation of healthcare system... Graph is the service structure Priory Cheadle Roval Featherstone Unit Cygnet Beckton Hooper Ward Exeter AMHP Devon Emergency Duty Team Coombehaven Exeter CRHT Personality Disorders Service Devon Liaison and Diversion Team Forensics IPP Management Allied Mental Health Partners

(AMHP)

## What is a network graph?



#### Represented in

Euclidean space, or Cartesian Coordinate System

A network graph is:

A graph based representation of a system

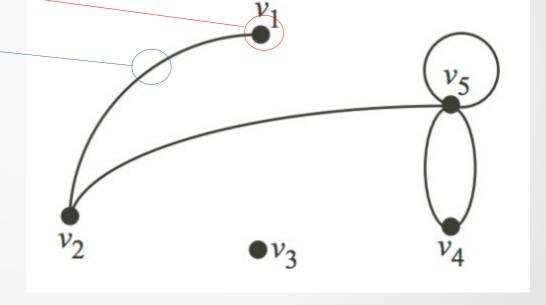
i.e. 2 services

- Based on pairwise relationships between entities
- Quantitatively encodes the properties of the system within the network structure

i.e. LOS, waiting lists, 'centrality of node' within network, # of nearest neighbours and connections, ...

## Terminology 1

- •Node/ vertex (v1 → v5)
- Edge
- •Graph unordered pairs of nodes *i.e., order does not matter*
- •Directed Graph (DiGraph) ordered pairs of nodes *i.e., ED* → *Orthopaedics*
- •Edge weight *i.e.*, *line weight*, *or 'traffic'*



Node attributes

## Terminology 2



For directed graphs you need each direction

- Parallel edges
- •Loop i.e., re-referral into a service
- Adjacent edges
- Adjacent nodes

Source

Target

### Required input data



Node ID is unique and (usually) numerical

Data required to build a network graph

If a DiGraph need to describe each direction

- Node list: ID\*, Label\*, n-attributes
- Edge list: Source\* (i.e. from), Target\* (i.e. to), n-attributes (e.g. Weight)

\* Mandatory

Self-loop: Source = Target

### Task 1 – Exploratory Data Analysis (30 mins)

- Open the 01\_data\_analysis.ipyb file from the exercises folder in Jupyter
- This exercise is what you make it! The notebook contains the code to import some data

   it's then up to you and your PSG to interrogate and come up with some interesting
   facts about it. You should also think about how it could be used with respect to network
   analysis.

Data Transformation (Patient level data...)

#### Data transformation process

- 1) Start with the raw data table patient level data **PD\_Data\_HSMA\_2020.csv**
- 2) Clean the data category consistency, negative dates, erroneous white space
- 3) Order the data by patient then by date (i.e. sequential use of services)
- 4) Create an adjacency matrix more on this shortly
- 5) Iterate over the interaction (a.k.a. adjacency) matrix to create edge list source, target, weight
- 6) Use unique instances or services or entities as nodes depends on the resolution of the data and how you want to subset your data

#### Data cleaning

Some things to consider...

- Capitalisation/ case inconsistencies (esp if using names as IDs suggest making all lower case)
- Spelling mistakes (to avoid inadvertent replications i.e., Elliot Vs Elliott)
- Erroneous white space leading, trailing, multiples
- Incorrect date entries reversed, missing
- Missing data

#### Order the Data

Order the data by patient then by date. This allows you to iterate over the data row by row to create the *adjacency matrix*. NA = current user

A B	С	D	E	, F	G
ClientID ReferralDate	ReferralDischarge	ReferralSource	WardTeam	GenSpecialty	ICD1
14 27/02/2011	19/05/2016	Internal referral	Link Centre Area 1	ADULT MENTAL ILLNESS	F603
14 13/10/2011	NA	Internal referral	Area 31 Adult Community Sector C	ADULT MENTAL ILLNESS	F603
14 23/01/2015	01/01/2016	Internal - Community Mental Health Team (Adult Mental Health)	Area 23 PPT	PSYCHOTHERAPY	F603
14 07/11/2016	NA	Community Mental Health Team (Adult)	Link Centre Area 1	ADULT MENTAL ILLNESS	F603
64 25/11/2014	12/04/2016	Internal - Community Mental Health Team (Adult Mental Health)	Area 23 PPT	PSYCHOTHERAPY	F606
64 28/02/2015	14/09/2015	Other	Area 23 MHAR	ADULT MENTAL ILLNESS	F606
64 05/07/2016	04/09/2017	Community Mental Health Team (Adult)	Area 23 PPT	PSYCHOTHERAPY	F606
64 11/07/2016	21/08/2016	Choose and Book (GP)	Area 24 MHAT	ADULT MENTAL ILLNESS	F606
64 27/05/2016	NA	Community Mental Health Team (Adult)	Area 12 CMHT	ADULT MENTAL ILLNESS	F606
64 20/08/2017	NA	Community Mental Health Team (Adult)	Link Centre Area 1	ADULT MENTAL ILLNESS	F606
89 05/06/2008	NA	Other	Area 26 CMHT	ADULT MENTAL ILLNESS	F609
89 19/03/2013	27/07/2015	Internal - Inpatient Service (Adult Mental Health)	Personality Disorders Service	PSYCHOTHERAPY	F609
89 06/01/2015	06/01/2015	Other Acute secondary care specialty	Area 30 Liaison Service	LIAISON PSYCHIATRY	F609
89 08/02/2015	08/02/2015	Other Acute secondary care specialty	Area 30 Liaison Service	LIAISON PSYCHIATRY	F609
89 09/05/2015	09/05/2015	Accident And Emergency Department	Area 30 Liaison Service	LIAISON PSYCHIATRY	F60
89 03/03/2015	11/03/2015	Accident And Emergency Department	Area 30 Liaison Service	LIAISON PSYCHIATRY	F60
89 24/02/2015	24/02/2015	Internal - Community Mental Health Team (Adult Mental Health)	Area 30 Liaison Service	LIAISON PSYCHIATRY	F60
89 22/06/2015	23/06/2015	Community Mental Health Team (Adult)	Area 30 AMHP	ADULT MENTAL ILLNESS	F609
89 19/07/2015	20/07/2015	Emergency Medical Unit	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F60
89 29/06/2015	07/07/2015	Inpatient Service (Adult)	Area 30 CRHT	ADULT MENTAL ILLNESS	F60
89 05/08/2015	06/08/2015	DGH Wards	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F609
89 16/09/2015	19/09/2015	Inpatient Service (Adult)	Area 30 CRHT	ADULT MENTAL ILLNESS	F609
89 06/09/2015	07/09/2015	Emergency Medical Unit	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F609
89 31/10/2015	01/11/2015	Emergency Medical Unit	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F609
89 29/09/2015	29/09/2015	Accident And Emergency Department	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F609
89 16/09/2015	16/09/2015	Accident And Emergency Department	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F60
89 20/09/2015	08/04/2016	DGH Wards	Personality Disorders Service	PSYCHOTHERAPY	F609
89 17/11/2015	19/11/2015	Accident And Emergency Department	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F60
89 08/03/2016	08/03/2016	Community Mental Health Team (Adult)	Area 30 AMHP	ADULT MENTAL ILLNESS	F60
89 04/06/2016	NA	General Medical Practitioner	Gender Identity and Sexual Therapy	ADULT MENTAL ILLNESS	F60
89 22/06/2016	23/06/2016	Accident And Emergency Department	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F60
89 22/06/2016	24/06/2016	Community Mental Health Team (Adult)	Area 30 AMHP	ADULT MENTAL ILLNESS	F60
89 03/06/2016	04/06/2016	Emergency Medical Unit	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F60
89 31/07/2016	31/07/2016	Accident And Emergency Department	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F60
89 22/07/2016	23/07/2016	Accident And Emergency Department	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F609
89 11/06/2016	11/06/2016	Accident And Emergency Department	Liaison Psychiatry Area 30	LIAISON PSYCHIATRY	F609

#### Adjacency Matrix (a.k.a. Sparse Matrix)



2D array –
rows are
Source and
cols are Target
1, 1 represent
the same
service

				_	Target					
	200	0	1	2	3	4	5	6	7	8
	0	Θ	0	Θ	Θ	0	Θ	0	Θ	0
t	1	Θ	0	0	Θ	0	0	0	Θ	0
	2	Θ	1	1	1	0	1	0	Θ	0
	3	0	0	1	1	0	0	0	Θ	0
	4	0	0	0	0	1	0	0	Θ	0
	5	Θ	0	1	1	0	9	2	Θ	0
	6	0	0	0	0	0	2	1	Θ	0
	7	Θ	0	Θ	Θ	0	1	0	θ	0
	8	0	0	0	Θ	0	Θ	0	θ	0
	9	0	0	0	0	0	1	0	Θ	0
Sour	C <b>e</b> <sub>0</sub>	0	0	0	0	0	Θ	0	Θ	0
	11	0	0	0	0	0	1	0	Θ	0
7	12	Θ	0	Θ	Θ	0	Θ	0	Θ	0
	13	А	А	A	A	A	5	3	A	Θ

Data underpinning the shape of the network graph itself

- Create a matrix of zeros n-services by n-services in size
- For each patient iterate over their data selecting the first service then the next and increment the location in the matrix referring to those services
- Source in row, target in column, frequency count in cell
- Can require a lot of computational storage (Adjacency List better)

#### **Node List**

Index	ID	Label	MeanLoS	MedianLoS	Setting
0	Θ	All 00A services	30.4444	26	00A
1	1	Area 10 AMHP	23	23	Community
2	2	Area 10 CRHT	4.625	3	Community
3	3	Area 10 Community Mental Health	1286.5	176.5	Community
4	4	Area 10 OPMH Community	505	234	Community
5	5	Area 11 MHAT	81.1143	58	Community
6	6	Area 12 CMHT	793.818	300	Community
7	7	Area 13 MHAR	1909.33	928	Community
8	8	Area 14 CMHT	550	550	Community
9	9	Area 15 CMHT	227.5	227.5	Community
10	10	Area 15 OPMH Community	173.429	85	Community
11	11	Area 16 Community Mental Health	449.5	234.5	Community
12	12	Area 17 CMHT	764	764	Community
13	13	Area 18 PPT	421	294	Community
14	14	Area 19 CMHT	1090	1090	Community
15	15	Area 20 OPMH Community	241	241	Community

- Required: ID a unique numerical identifier (required to determine source and target)
- Optional: Label a text string descriptor (i.e., service), continuous and discrete attributes of any type (i.e. categorical and numerical)

#### Edge List

Index	Source	Target	Type	Id	Weight
0	37	θ	Directed	0	1
1	49	θ	Directed	1	1
2	57	θ	Directed	2	1
3	58	θ	Directed	3	1
4	59	θ	Directed	4	1
5	69	θ	Directed	5	1
6	73	θ	Directed	6	1
7	83	θ	Directed	7	1
8	2	1	Directed	8	1
9	2	2	Directed	9	1
10	3	2	Directed	10	1
11	5	2	Directed	11	1
12	47	2	Directed	12	1
13	49	2	Directed	13	1
14	56	2	Directed	14	1
15	86	2	Directed	15	1

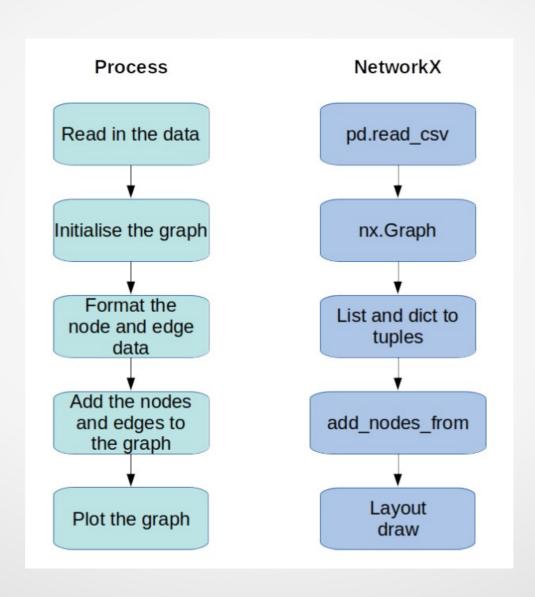
- Required: Source unique node identifier, Target unique node identifier, ID unique edge identifier. Picked out from Adjacency Matrix where cell > 0
- Optional: Type directed or undirected, weight number from adjacency matrix, continuous and discrete attributes

#### Task 2 – Data Transformation (25 mins)

- Open the 02\_data\_transformation.ipynb file from the exercises folder in Jupyter
- "The code below provides an example of how patient data would be transformed to conduct network analysis. Everything you need is below - however the code has been split into EIGHT blocks and then jumbled up."
- "Your task is to re-sort the code blocks so that they run in the correct and sequential order. Note that each of the **FOUR** main steps have further been split in to two substeps. This means main and subsets need to be sorted accordingly."
- As we're nice on the HSMA Programme, the library and data imports have been done for you!

Building a Network Graph

#### Building a Network Graph



### Task 3 – Fill in The Gaps (35 mins)

- Open the 03\_build\_a\_network\_graph.ipynb file from the exercises folder in Jupyter
- Update the code in the subsequent slides which contains missing information (identified with \*\*) (3a 3g)
- This task requires the node and edge lists created in Task 2

#### Task 3a – Library & File Imports

#### Import the required libraries and the data we just created

```
import networkx as nx
import pandas as pd

nodes = pd.read_csv('../data/node_list.csv', low_memory=False)

edges = pd.read_csv(**, low_memory=False)
```

## Task 3b – Instantiate Graph Object

Initialise a graph object which we will assign to the variable **G** 

```
** = nx.Graph()
```

#### Task 3c – Prepare Nodes



Node inputs are tuples of the node ID and a dictionary of attributes

Node tuples (list of ID, dict of attributes)

```
idList = nodes[**].tolist()

labels = pd.DataFrame(nodes['Label'])

labelDicts = **.to_dict(orient='records')

nodeTuples = [tuple(r) for r in zip(idList, labelDicts)]

['Label': 'Area 16 Community Mental Health'},
{'Label': 'Area 17 CMHT'},
{'Label': 'Area 18 PPT'},
```

```
1 letters_ = ['a', 'b', 'c', 'd']
2 numbers_ = [1, 2, 3, 4]
3 symbols_ = ['-', '*' , '/', '+']
4
5 [i for i in zip(letters_, numbers_, symbols_)]
[('a', 1, '-'), ('b', 2, '*'), ('c', 3, '/'), ('d', 4, '+')]
```

1 labels.to dict(orient='records')

{'Label': 'Area 10 OPMH Community'},

{'Label': 'Area 15 OPMH Community'},

{'Label': 'Area 10 Community Mental Health'},

[{'Label': 'All 00A services'},

{'Label': 'Area 10 AMHP'}, {'Label': 'Area 10 CRHT'},

{'Label': 'Area 11 MHAT'},
{'Label': 'Area 12 CMHT'},
{'Label': 'Area 13 MHAR'},

{'Label': 'Area 14 CMHT'}, {'Label': 'Area 15 CMHT'},

#### Task 3d – Prepare Edges and Weights

Edge inputs are tuples of source ID, target ID and a dictionary of attributes

Edge tuples (Source, Target, dict of attributes)

```
sourceList = edges[**].tolist()

targetList = edges['Target'].tolist()

weights = pd.DataFrame(edges[**])

weightDicts = **.to_dict(orient='records')

edgeTuples = [tuple(r) for r in zip(sourceList, **, weightDicts)]
```

### Task 3e – Add Nodes & Edges to Graph Object

We add the nodes and edges to the graph object **G** 

```
G.add_nodes_from(**)
```

G.add\_edges\_from(\*\*)

#### Task 3f – Determine Node Coordinates



We pass the graph object to a layout algorithm to determine the node coordinates and hence the edge start and end points.

Various ways to do this... Some layouts are geometric (i.e., cyclical or hierarchical), some fare force based layouts (working on physics engines i.e. proximity and gravity)

```
pos = nx.kamada_kawai_layout(**)
```

In the previous steps we added Nodes and Edges – but we haven't told it how we want to *visualise* it. Quite a complex task given there are so many!

With NetworkX we need to determine a set of coords for the Nodes – this gives us Edge start/ end points

This will assign each Node a position on a Cartesian coord system (i.e. x, y graph) and save into the POS variable

#### Task 3g – Plot the Graph

Finally we draw the graph passing in the graph object G, the position object 'pos' and adjust the node size. Don't forget the labels!

```
plot = nx.draw(**, pos=**, node_size=100,
    with_labels=True, font_size=10)
```

#### Next Time (4C and 4D)

Subsequent sessions will cover...

- Different types of graph
- Graph metrics and their interpretation
- How graph visualisations work
- Graph algorithms and custom layouts
- Interactive graphs

#### Further Info & Acknowledgement

Slides adapted from Sean Manzi

Checkout <a href="https://www.project-nom.com">https://www.project-nom.com</a> for more information and training on the use of network-based operational modelling for whole system modelling in healthcare