CLINICAL INFORMATICS RESEARCH PROGRAMME

Understanding the fundamentals of graph analytics with emphasis on example EHR data

John Booth
Data Engineer CIRP
UCL Research Student: UCL GOS Institute of Child Health

December 2021

Outline Plan for PhD

Exploring the role of graph databases in the interrogation and visualisation of routinely collected Electronic Patient Record (EPR) data

Three phases:

- What are graphs and graph analytics?
  - Types
  - Databases
  - Analytics
- What are graphs good for in clinical informatics?
  - Define POCs using real EPR data
- How do you communicate with graphs?
  - To a non-data scientist i.e. a clinician

Presentation

- Review the story of phase 1 so far:
  - Graph creation.
  - Graph Analytics.
  - Graph Analytics pipeline.
  - What has been learnt.
  - What are the next stages.

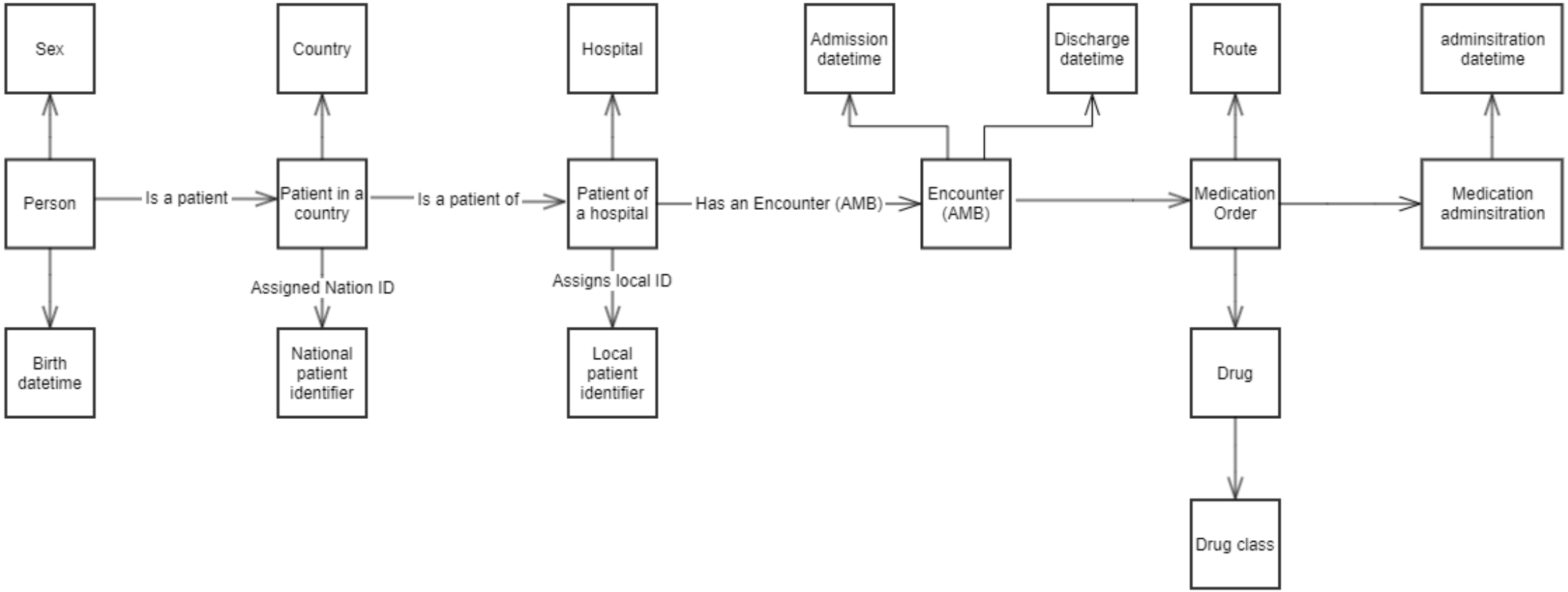
Explore Graph Creation and Visualisation

- Data Extraction
- Use simple data set to explorer the creation and visualisation of a graph
  - Graph Schematics
  - Initial attempts
    - Python, networkX and matplotlib
  - Simplify schematics
  - Explorer graph creation and layout
    - Methods of persistence.
      - YAML, xlsx, GML and graphML
    - Methods of visualisation
      - Gephi, Graphia, plotly

Data Extraction

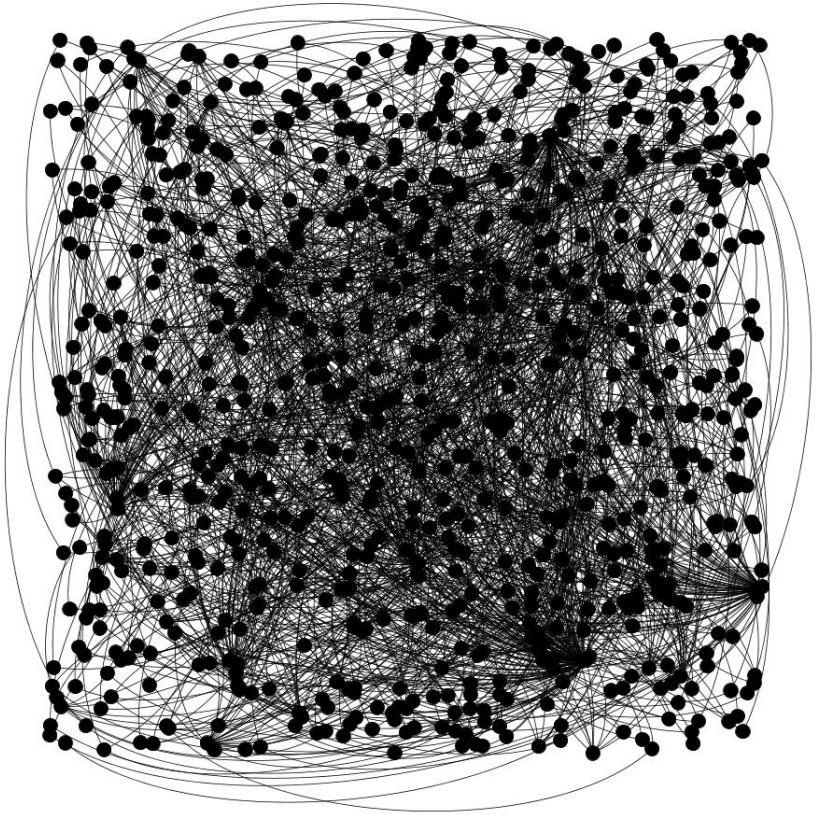
- Extracted all patients with some contact with Nephrology specialty at GOSH and assigned de-id ids.
  - Period: 2019-05-01 to 2021:08:30
  - Age on first contact less than 18 years.
- For this patient cohort I extracted the patient's hospital admissions and selected a given number of patients with a hospital stay between 7 and 14 days.
  - Initially 10 patients were selected.
- For this sample set of patients I extracted the patient's:
  - Demographics
  - Hospital admissions
  - Medication orders

Initial Graph Schematic



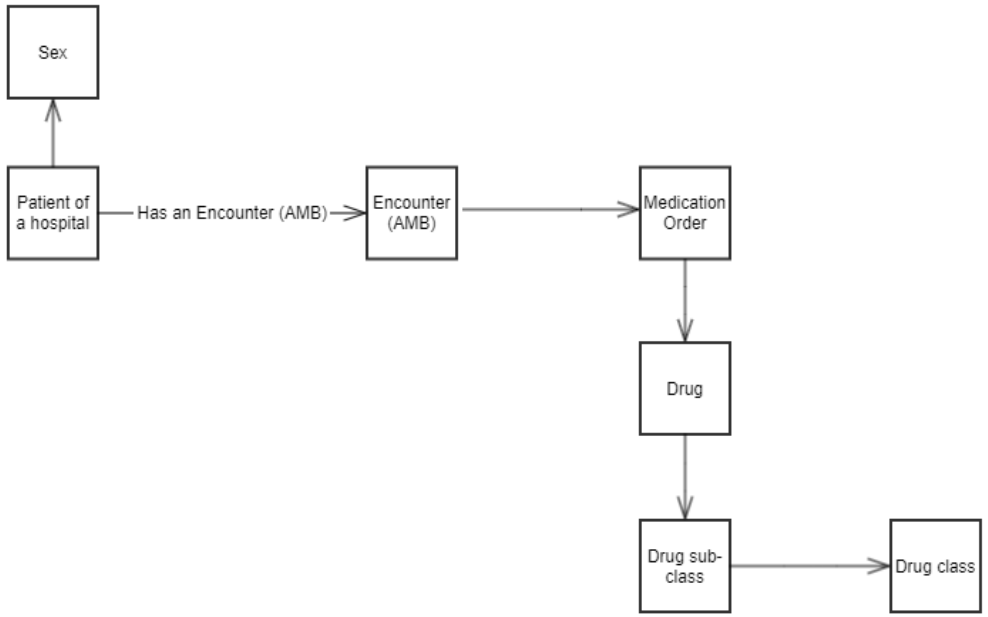
Initial graphs

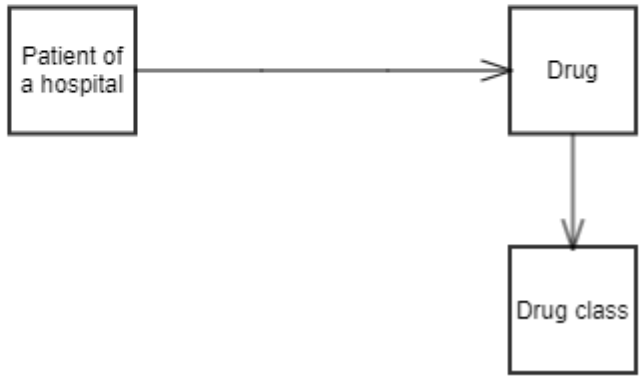
- Initial graph of EPR data as extracted. Graph created in python using networkX and exported to a GML file and loaded into Gephi for visualisation:



Simplifying the graph schematic

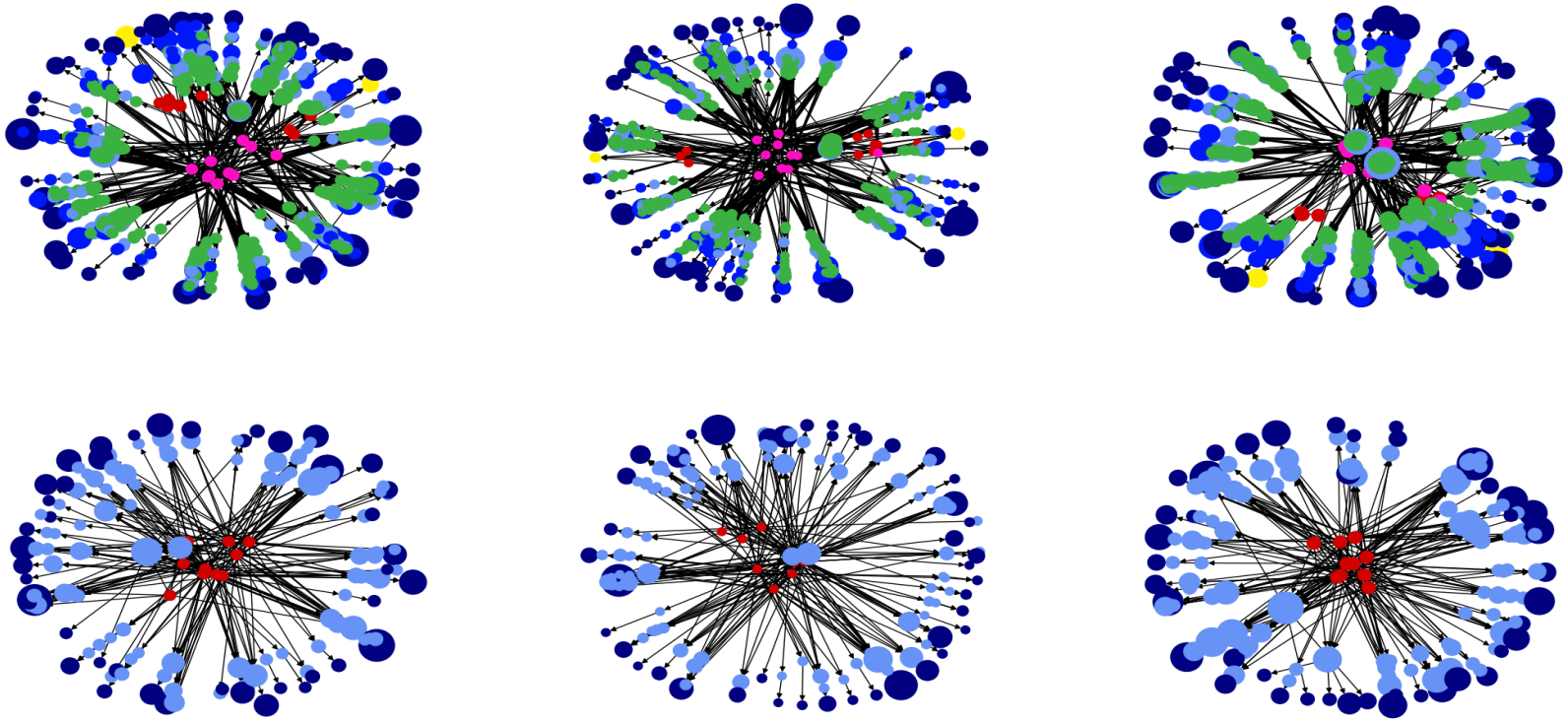
Creating labelled parameter graphs, some of the detail could be moved to parameters to limit the number and variety of nodes:





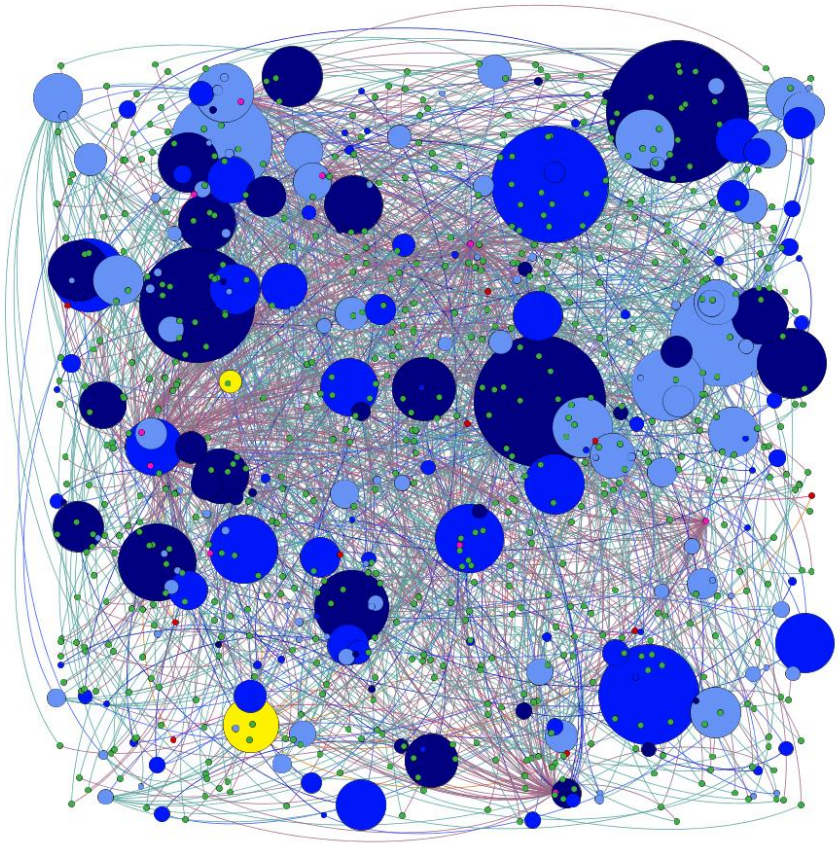
Explorer node colour, size and simplicity

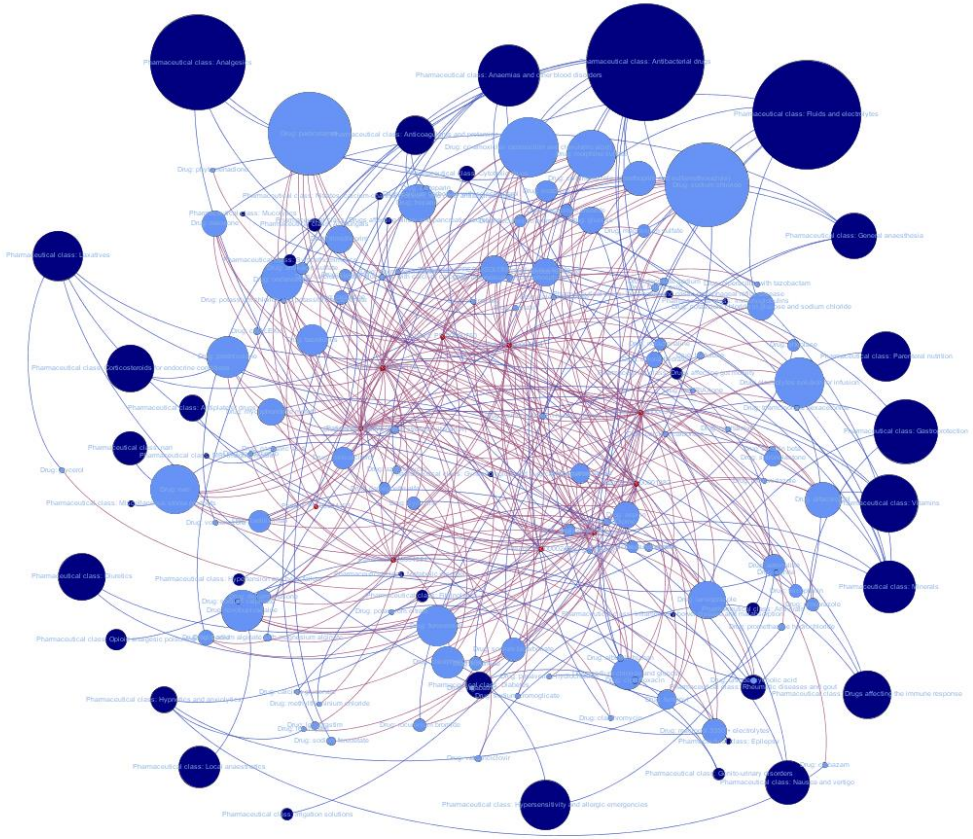
- Graph created in python using networkX, exported as GML file and colours and sizes added for plotting. These plots were produced using matplotlib. The top row has all node types, the bottom row has a reduced set. Size is based on number of events or patients or both :



Developing graphs using Gephi

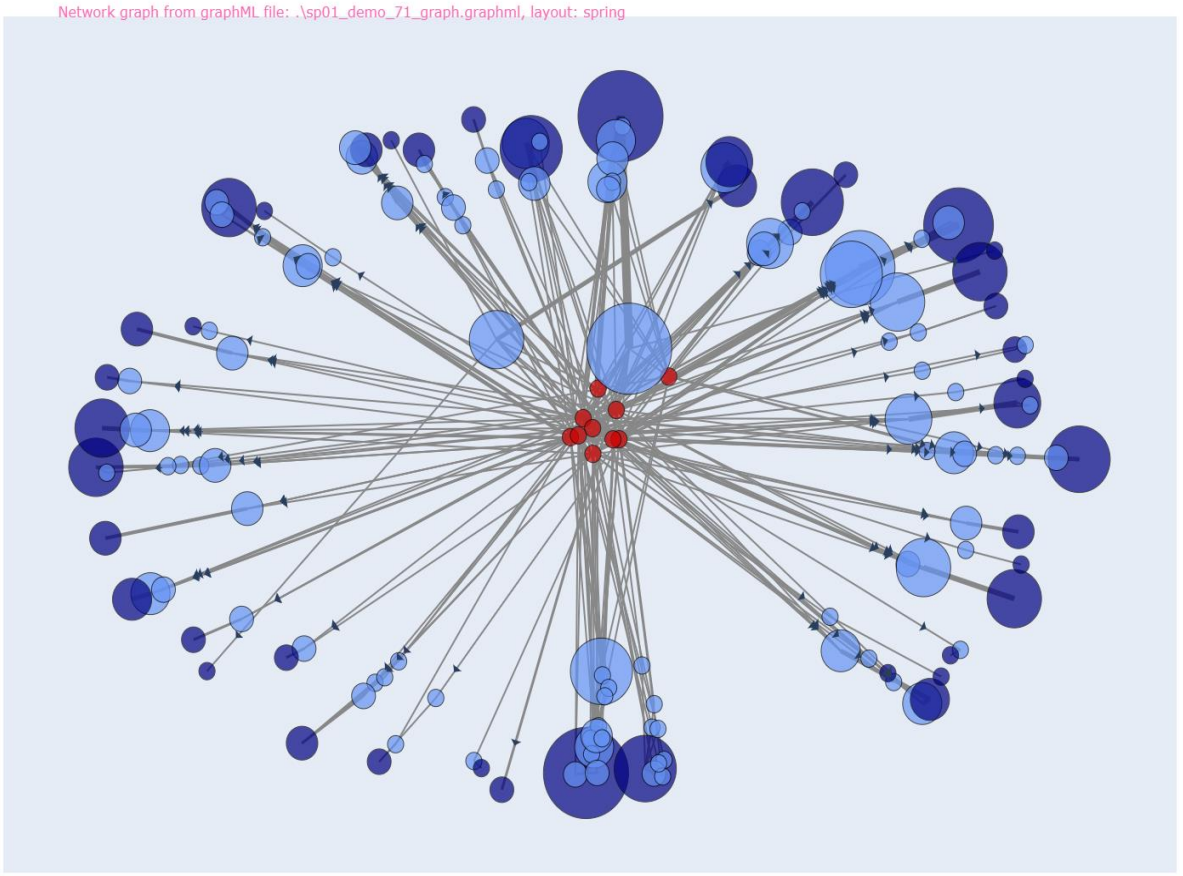
Graphs created in code and exported using graphML which supports extended graph information, the key issue was that to create a finalized image a lot of manipulation had to be done manually:

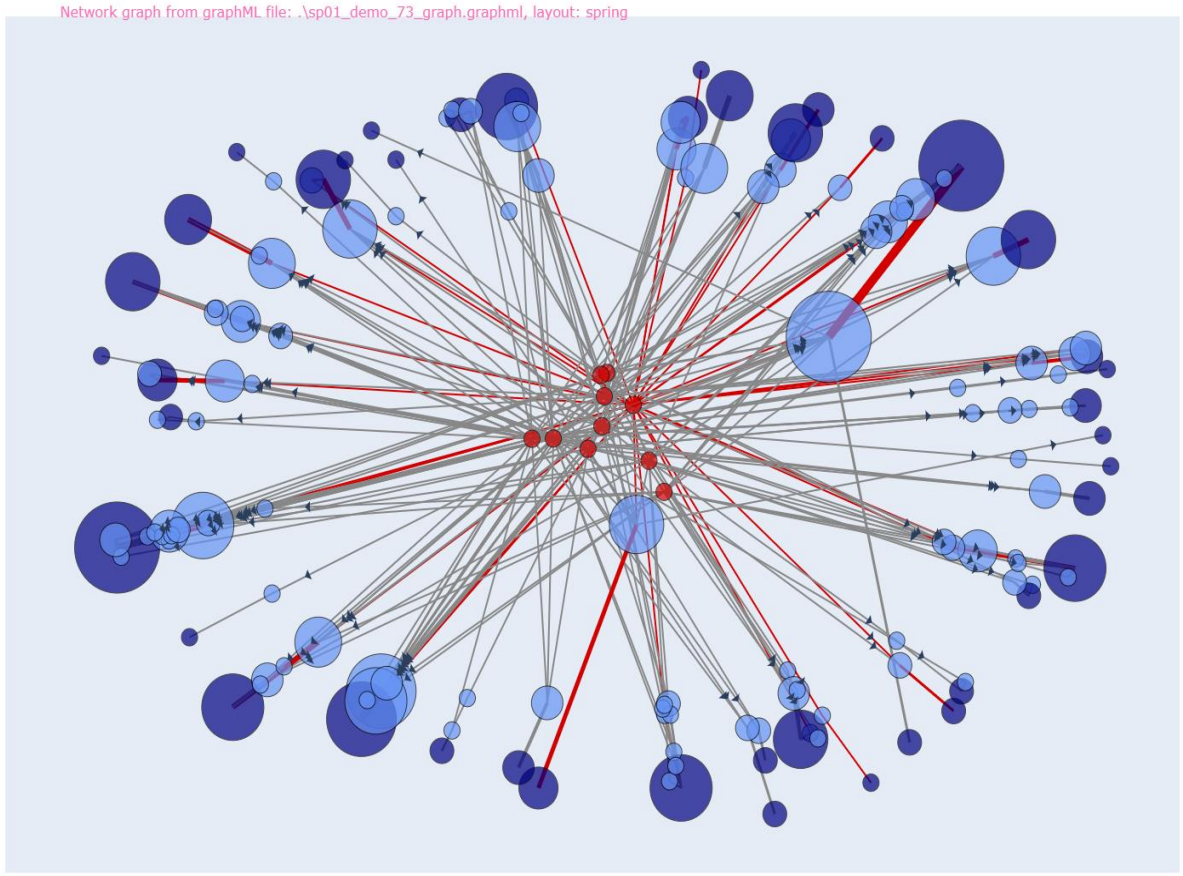




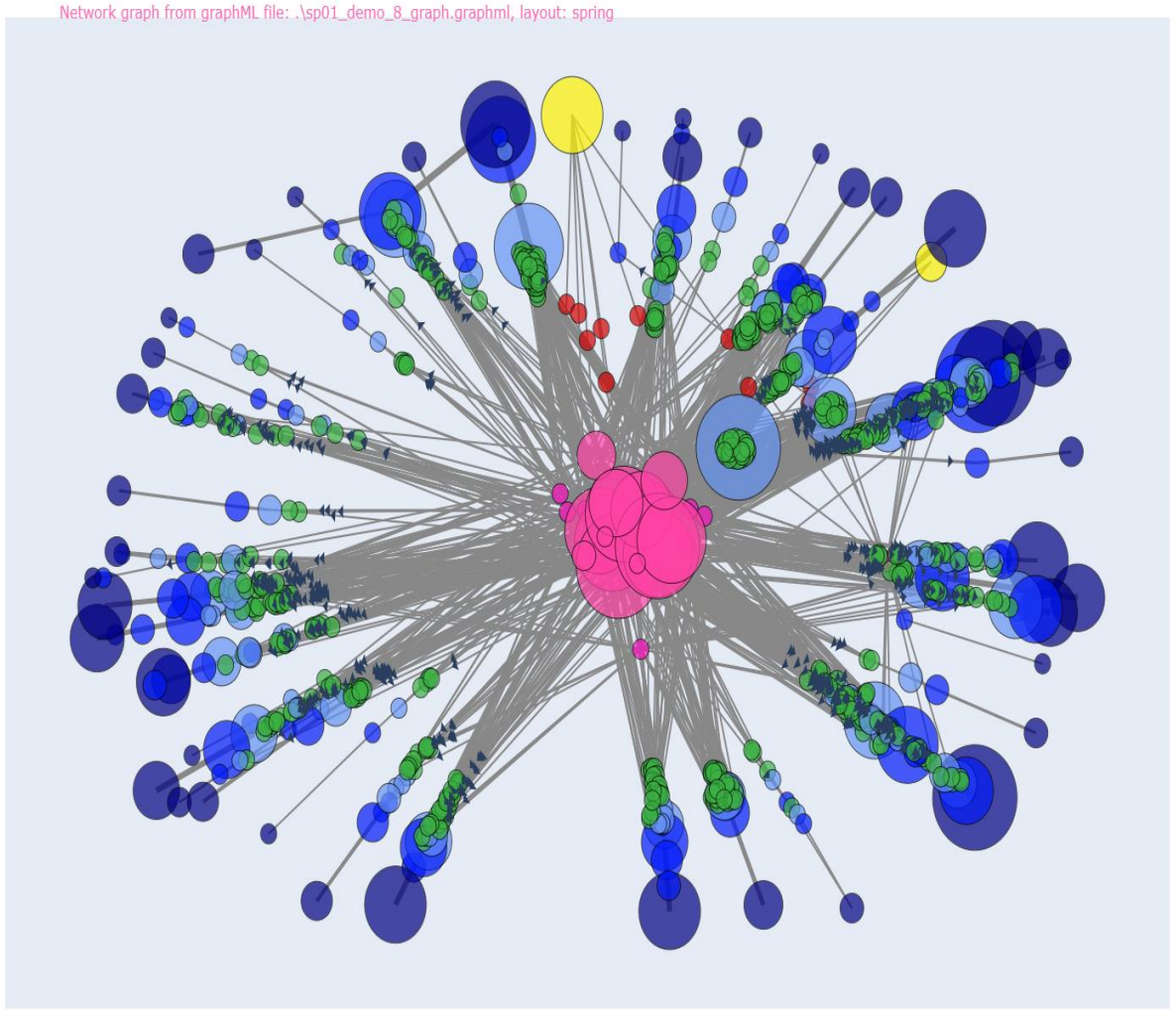
Developing graph visualisation in code.

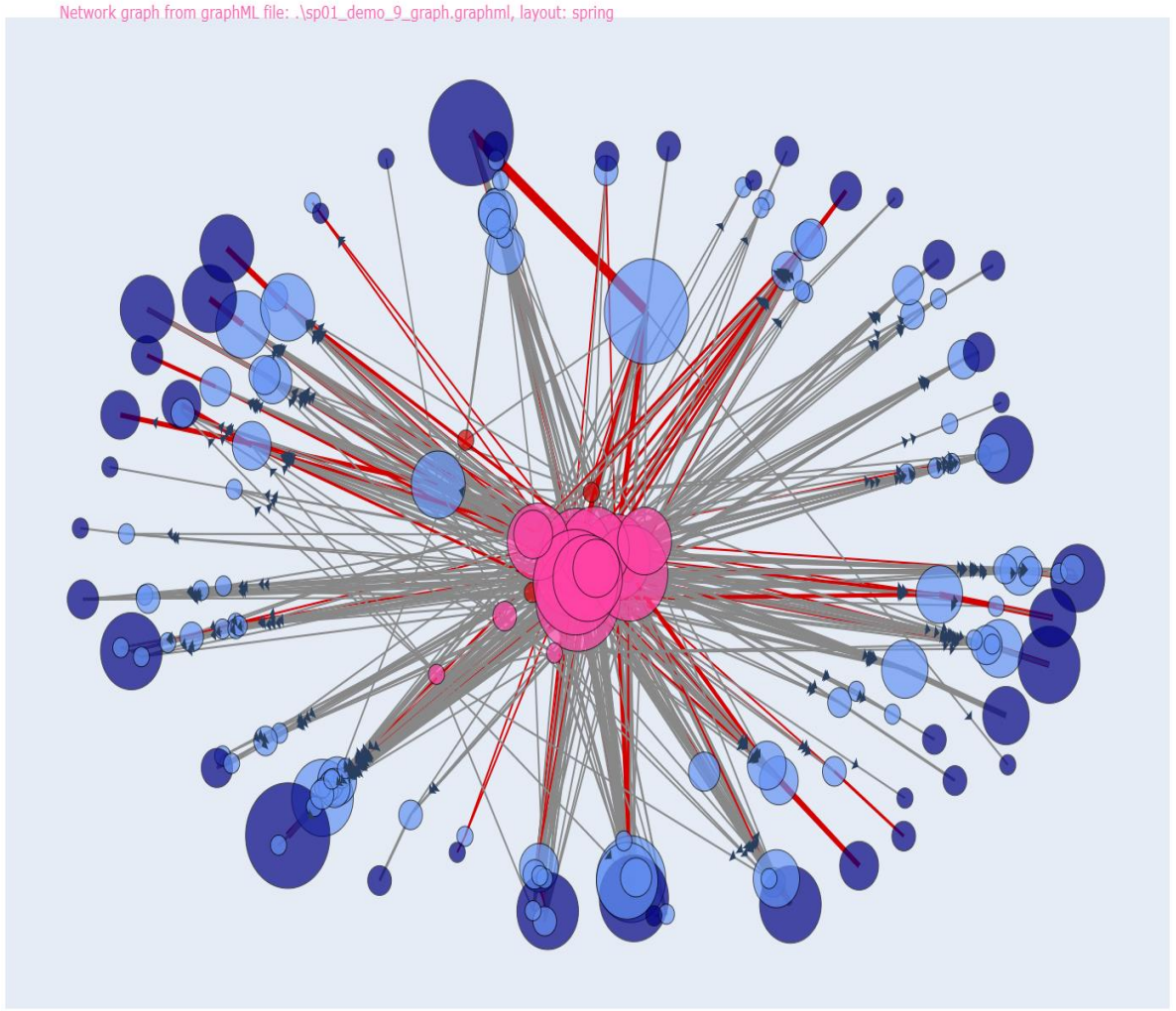
Creating graph in code, storing as a graphML file, read by networkX and plotted using plotly. Adding edge thickness based on patient number, then add highlight of path of an individual patient.



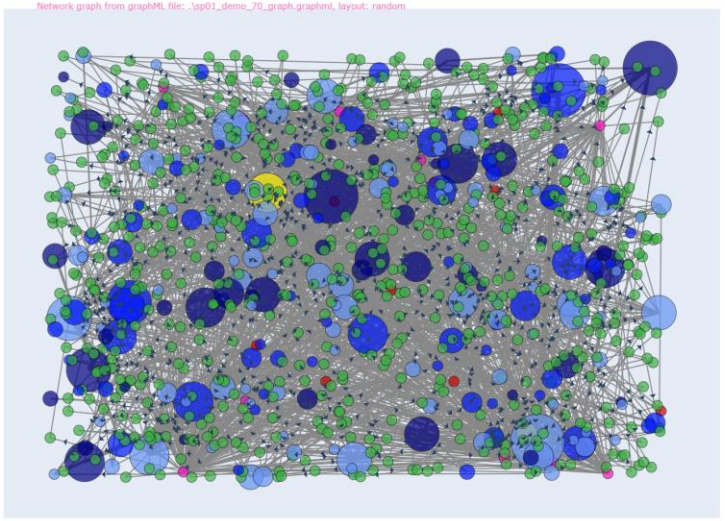


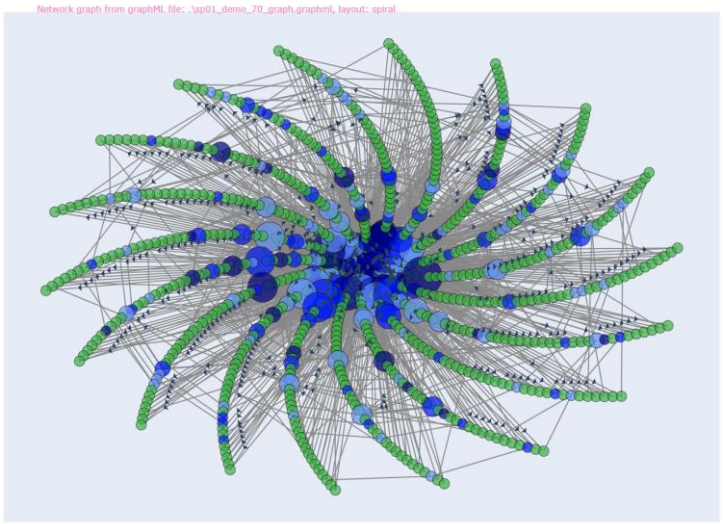
Add in time tree based on day of admission

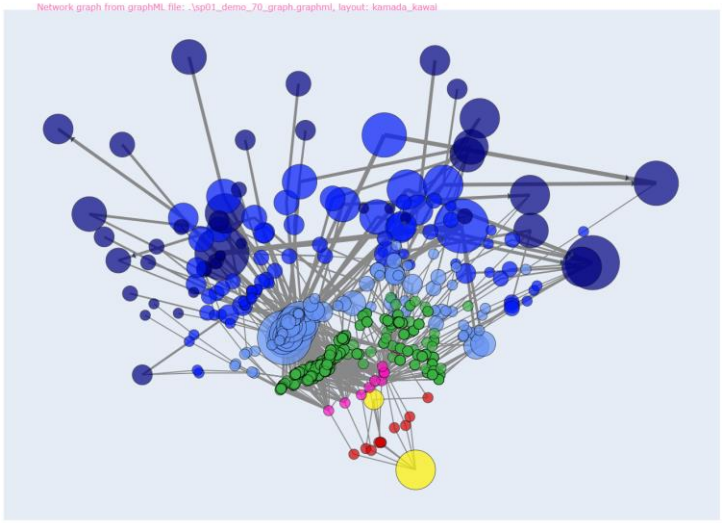


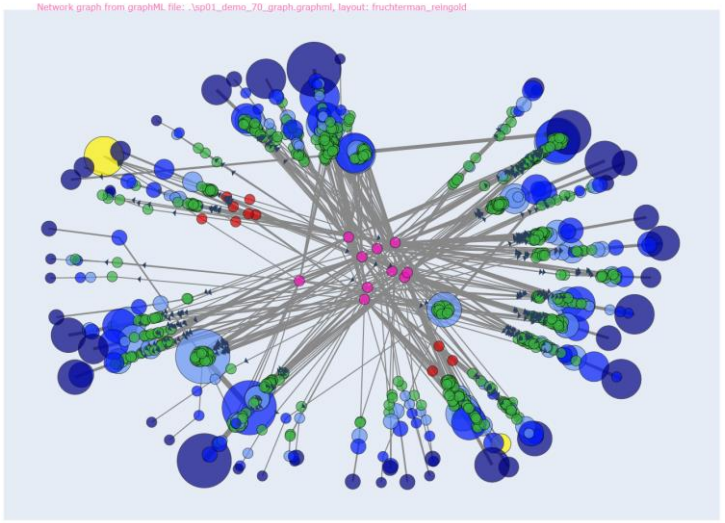


Developed interface to play with graphs and layouts









Interface: dash
Plotting: plotly

Layouts: networkX

- Random
- Spiral
- Kamada kawai
- Fruchterman reingold

Run demo of file and layout selection.

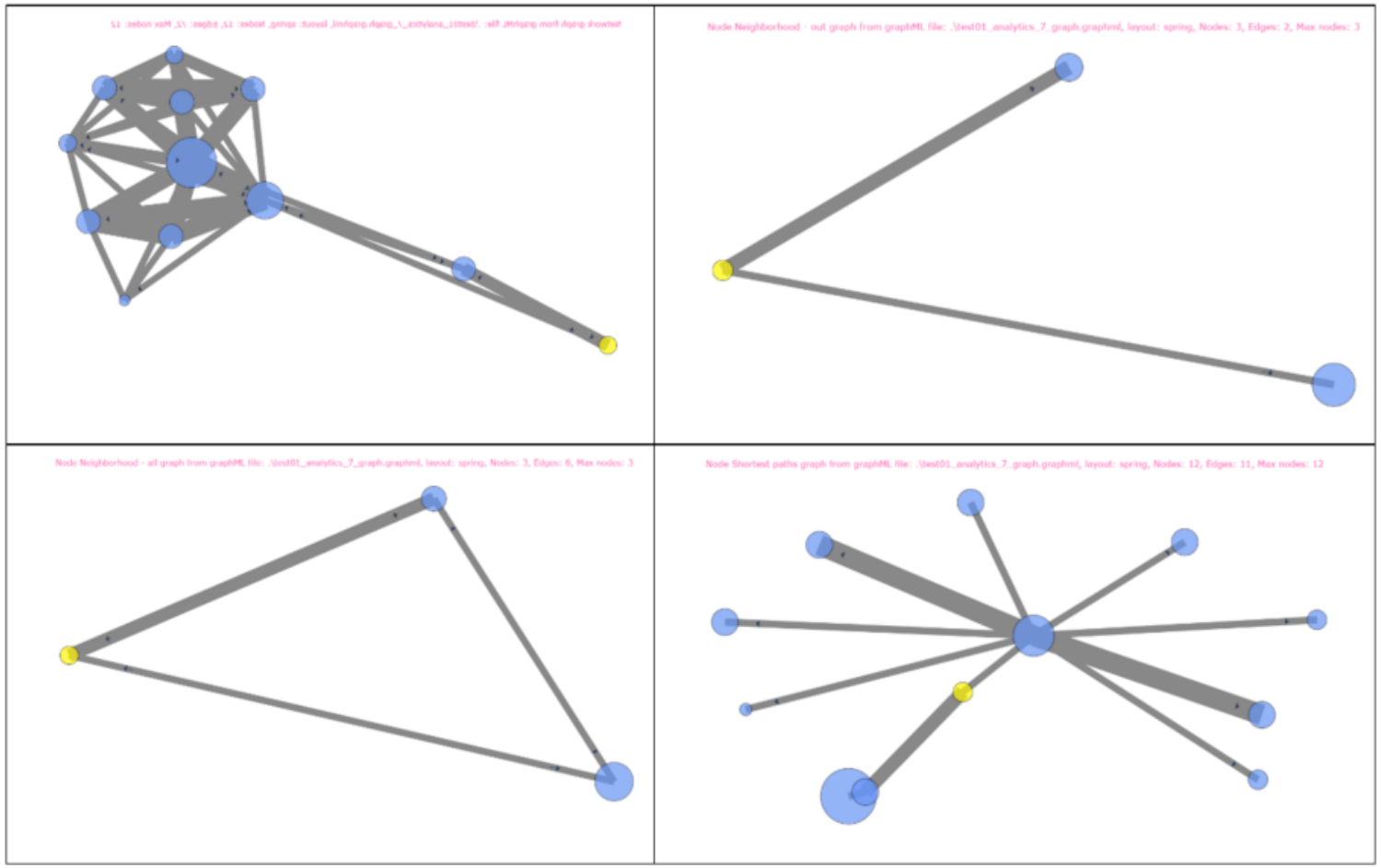
Graph Analytics

- Simplified the data set for plotting:
  - Concentrated on patient daily co-medications.
  - Based on the same data as extracted for the previous work
  - Created a day of therapies (DOTs) RDV
    - Project_id, sex, age_group_at_admission.
    - Day_of_admission
    - Drug_name, pharmaceutical class and sub-class
  - Created a homogeneous directed graph of drugs linked by co-medication on a given period in this case Days.
  - With the defined structure created test data to control how the analytics would be produced.
  - A defined set of analytic values for each graph were then calculated and persisted in a JSON file:

Graph	Node
Density	Neighbourhoods
Transitivity	Shortest path
Average clustering	Centralities

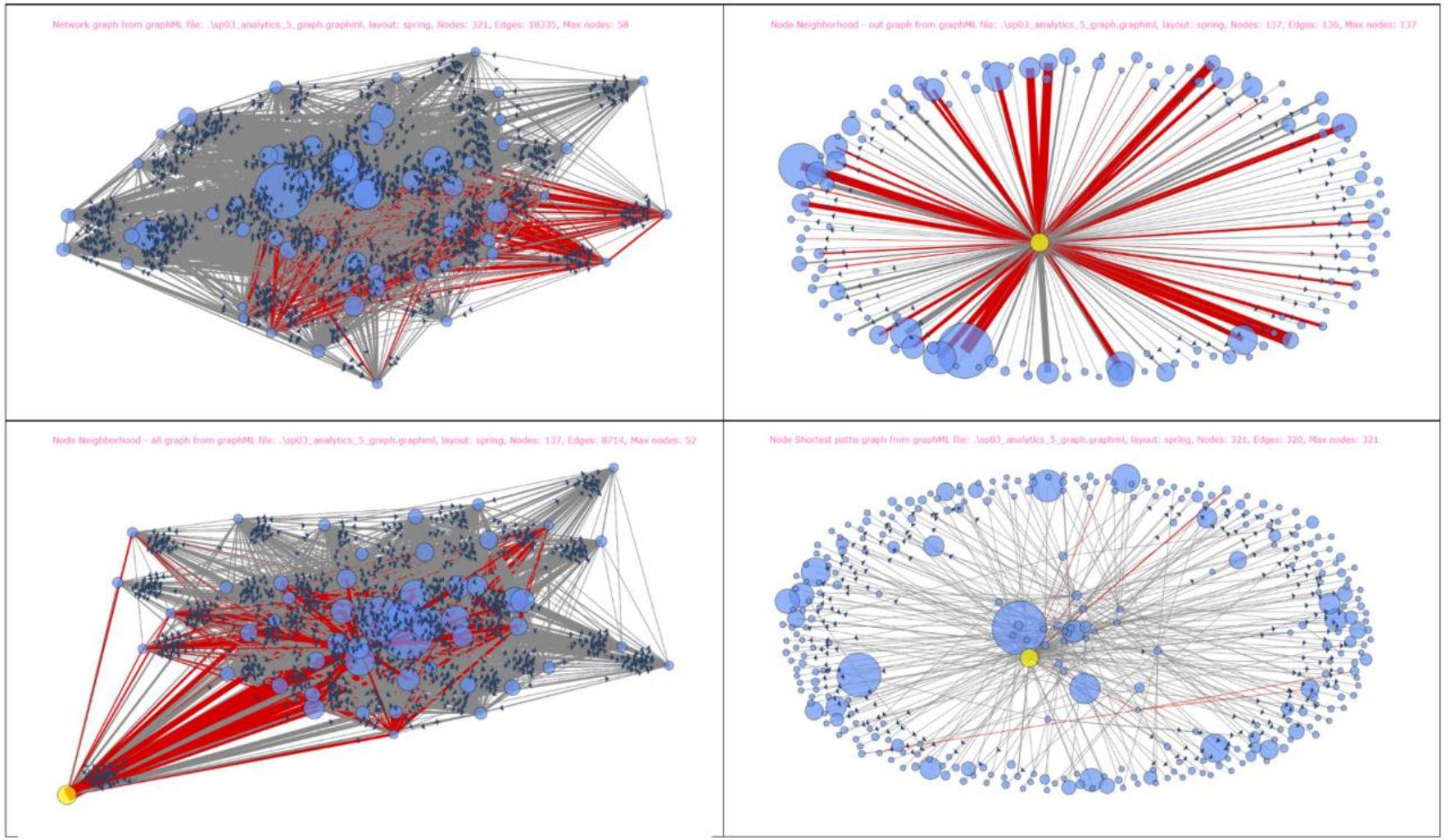
Initial analytics with test data

All nodes, Selected Node: out neighbours, all neighbours and shortest path.



Initial analytics with Patient Data

All nodes, Selected Node: out neighbours, all neighbours and shortest path.

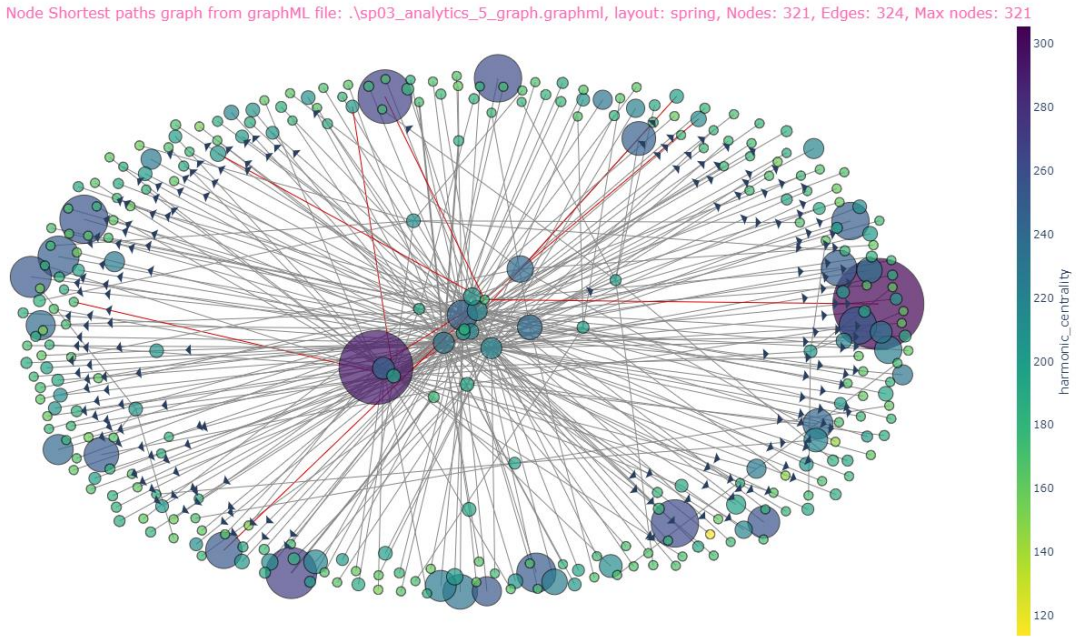


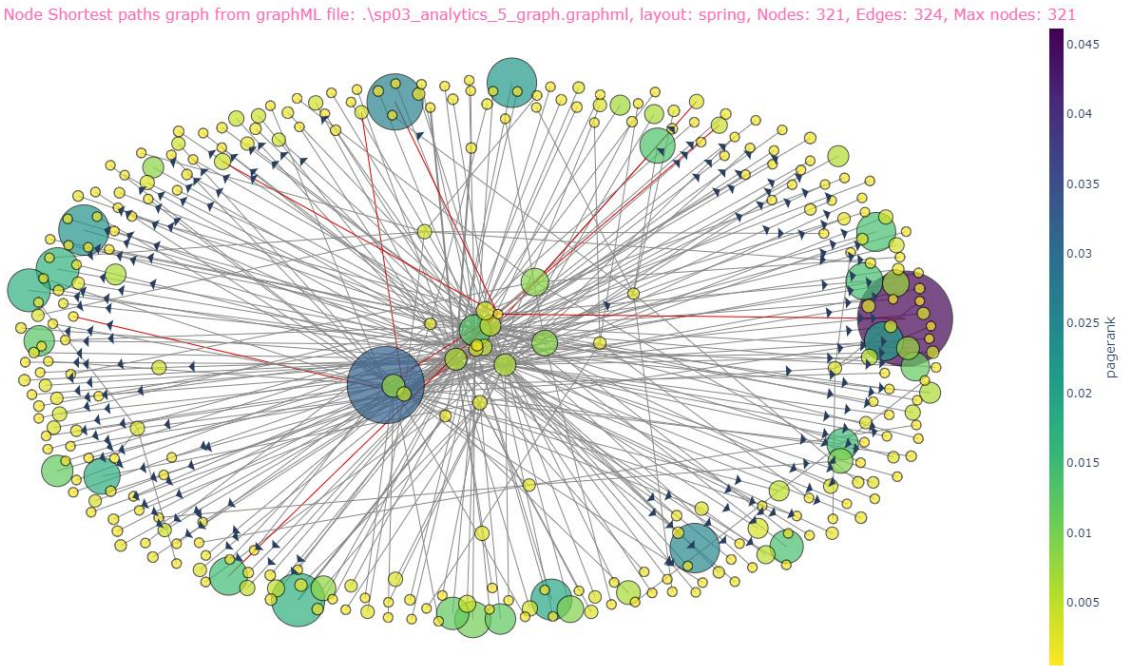
Graph Analytics: Centralities

- Wikipedia:
  - In graph theory and network analysis, indicators of centrality assign numbers or rankings to nodes within a graph corresponding to their network position. Centrality concepts were first developed in social network analysis, and many of the terms used to measure centrality reflect their sociological origin.
  - Centrality indices are explicitly designed to produce a ranking which allows indication of the most important vertices.
  - A number of different algorithms have been developed:
    - Degree Centrality: defined by the number of edges originating or terminating at each node.
    - Closeness Centrality: defined by the length of the shortest path between one node and all other nodes.
    - Betweenness Centrality: The number of times a node acts as a bridge along a shortest path between two other nodes.
    - PageRank: Developed by Larry Page for Google search.

Graph analytics: Shortest Path and Centralities

Added the facility to colour nodes by their different centrality values. The following two plots show the shortest path for the same selected drug comparing harmonic centrality with PageRank.

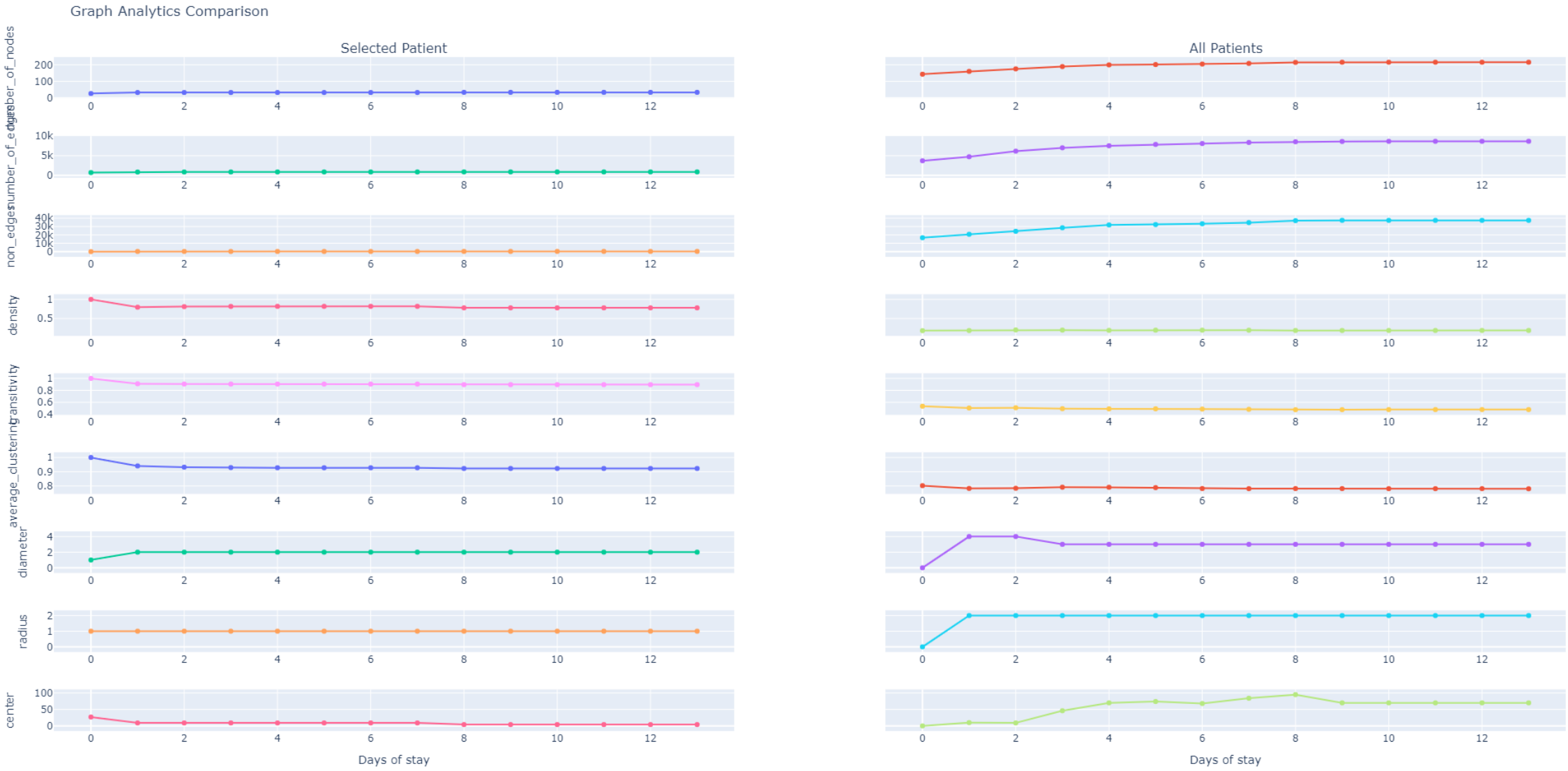




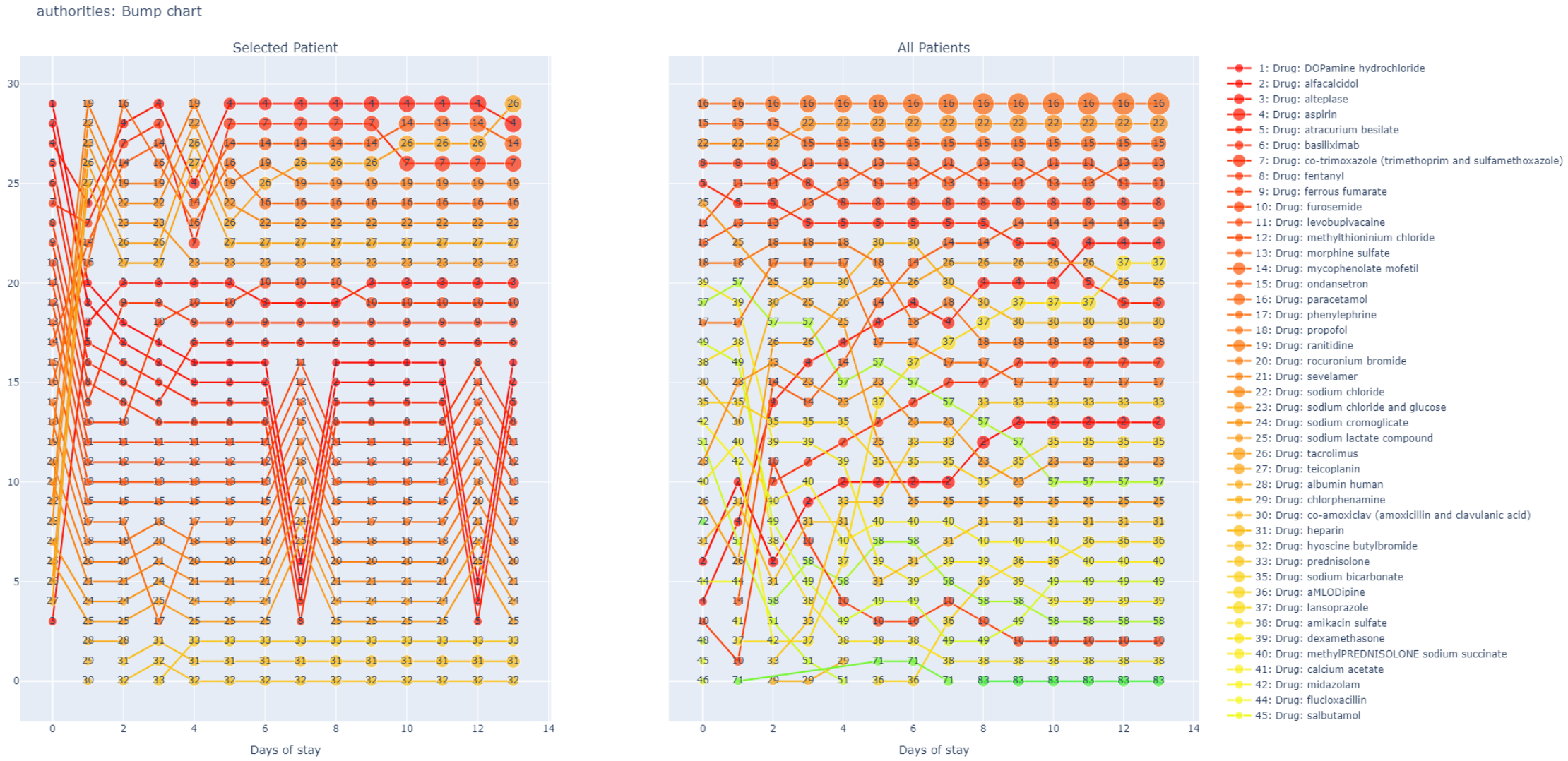
Graph Analytics Pipeline

- Create a simple analytics pipeline do demonstrate the concept of using graph analytics in comparing patient clinical data:
  - Select a patient.
  - Creates a graph and calculate analytics for each cumulative day of admission for co-medications.
  - Create a comparison cohort based on age group at admission (52 patients).
  - Create a graph and calculate analytics for control group for each cumulative day of admission for co-medications.
  - Summarise graph analytics
  - Dash/Plotly page to display summary graph analytics.

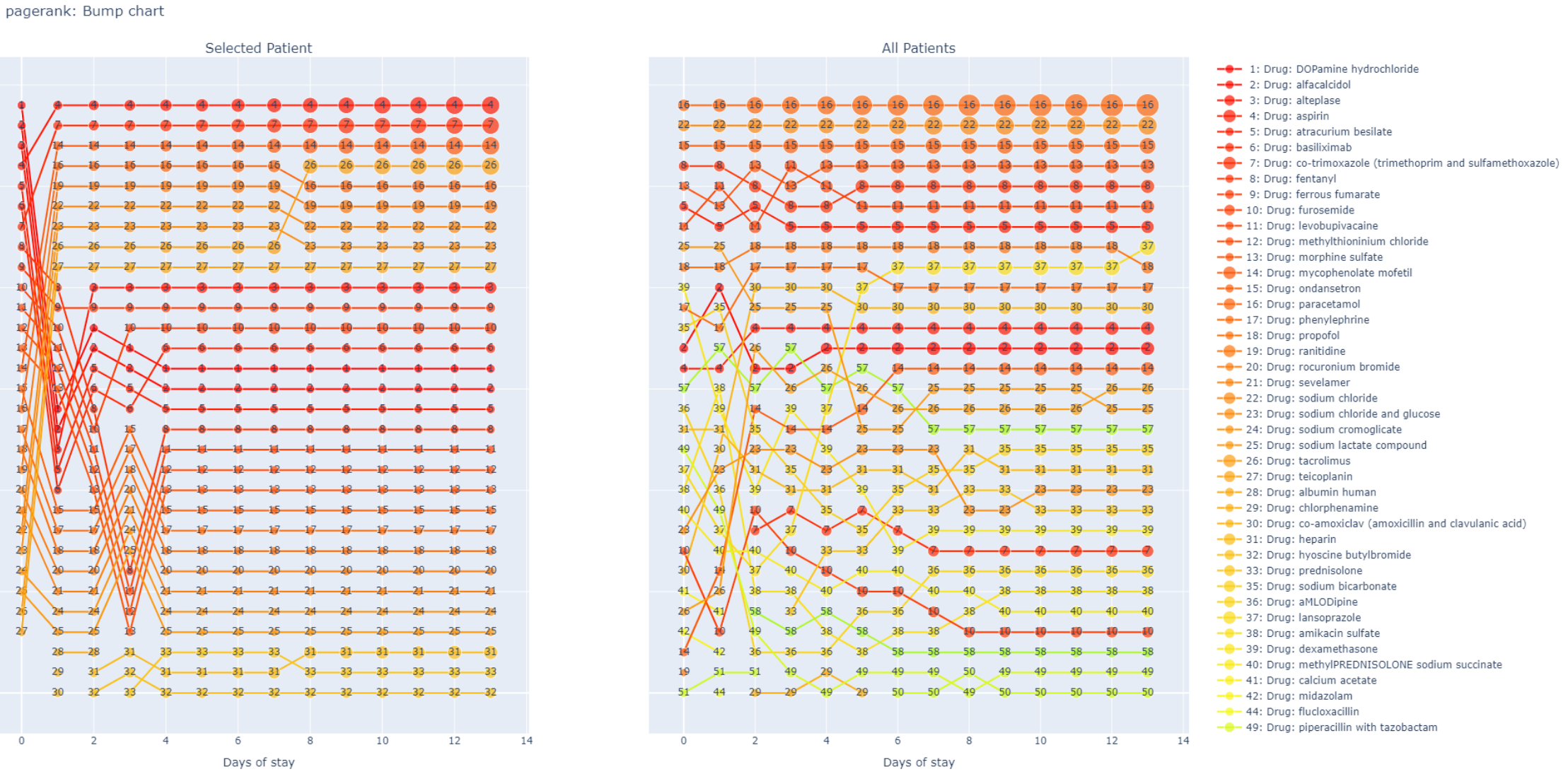
Graph level Analytics



Node Level Analytics



Node Level Analytics



What have I learnt? Graph Creation:

- These terms matter:
  - Undirected and directed
    - Co-medications – undirected
    - Medication sequencing - directed
  - Unweighted and weighted
    - Patients, events, or some combination
  - Unipartite, bipartite and multipartite graphs.
    - Drugs – unipartite
    - Drugs and day of admission – bipartite
    - Patients, medication orders, drugs and day of admission - multipartite
  - Multilayer and multiplex representations
    - Day of admission as a layer and drug co-medication on each layer – Multilayer
    - Patient Drug co-medication and Lab results – Multiplex (different relations for same nodes)
- These terms don’t matter:
  - Graph or Network

What have I learnt? Graph Analytics:

- JSON file format is a lot faster than YAML in both reading and writing dictionaries.
- Having defined a graph producing a range of analytic values is straight forward.
- Interpreting these values is more complex.
  - The more varied the data the more complex it will be to interpret any graph analytics results.
  - Having a constrained test set to demonstrate individual analytic values has proved very valuable.
- These terms are important:
  - Graph Density
  - Graph Transitivity
  - Node Centrality
  - Shortest path

Next Steps – Explore Graph Analytics

- Symantic Graphs
  - To understand the basic process of taking standard RDV's and create a semantic graph and to visualise that graph.
  - Based on Rosie Hamilton's MSc project (A knowledge graph visualisation tool for clinical trials data).
  - Extract clinical notes for Nephrology Cohort as per previous sprints and structure dataset for NLP analysis
  - Extract triplets use the python package scispaCy with en_core_sci_lg a full spaCy pipeline for biomedical data with a larger vocabulary and 600k word vectors.
  - Create graph using NetworkX, visualise graph using plotly.
- Literature Review
  - Use terms built up in previous sprints.
    - (EPR OR EHR)
    - (graph OR network)
    - (directed OR homogeneous OR heterogeneous OR bipartite OR multi-layered OR multiplex)