TRANSFORMING HEALTHCARE DATA WITH GRAPH-BASED TECHNIQUES USING SAIL DATABANK
PROJECT METHODOLOGY REPORT | JULY - DECEMBER 2022

Jamie Burke
University of Edinburgh
School of Mathematics
College of Science and Engineering
Jamie.Burke@ed.ac.uk

Supervisor: Dan Schofield
NHS England
Transformation Directorate
daniel.schofield1@nhs.net

ABSTRACT

In this project methodology report we explore (directed) hypergraphs as a novel tool for assessing the temporal relationships between coincident diseases, addressing the need for a more accurate representation of multimorbidity. Directed hypergraphs offer a high-order analytical framework that goes beyond the limitations of directed graphs in representing complex relationships. After exploring novel weighting schemes which can capture different aspects of the underlying data, we then turn our attention at the power of these higher-order models through the use of PageRank centrality to detect and classify the temporal nature of conditions. This work was completed as part of an NHS England Data Science PhD Internship project.

Contents

1 Introduction

Multi-morbidity is the health state of having two or more concurrent conditions. This is becoming more common as populations age with improved life expectancy. This is a result of the increased survival rate of both acute and chronic illnesses due to the advancements in healthcare. However, multimorbidity is poorly understood and combated as current healthcare services are a compartmentalised system, originally designed under a single-disease framework, catering for single-condition individuals. This is a model which is known to lead to fragmented, costly and ineffective care [1].

Figure 1 demonstrates the requirement of more comprehensive care models to care for more intensive disease models that include multimorbidity [2]. The expectations of a multidisciplinary teams system model to combat multimorbidity effectively are difficult to meet given the current single-disease model in place in healthcare currently, leading to individuals living with multimorbidity not getting the complex care they need [2].

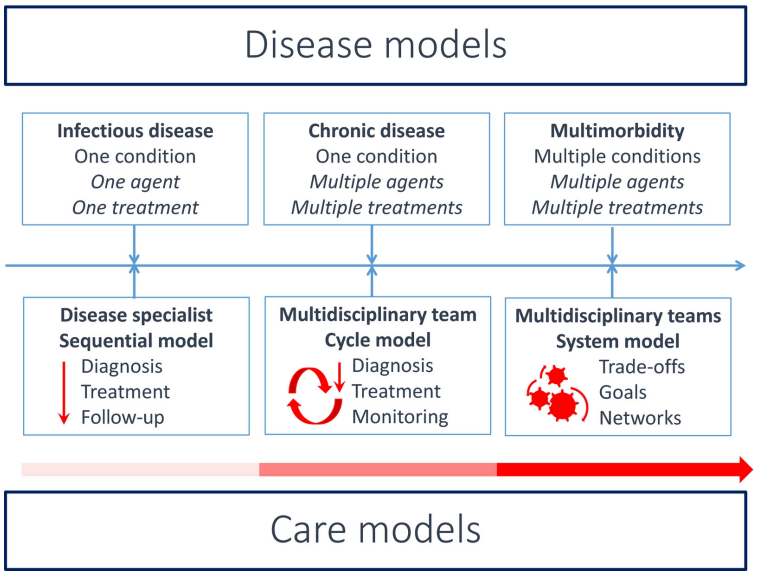

Figure 1: The increase in intensity for care models given more complex disease models [2].

It is expected that 2 of 3 UK adults over 65 will experience multimorbidity by 2035. [3]. In a time where NHS services are struggling to cope with an ageing population set to double in the next 20 years with life expectancy predicted to increase, improving our understanding of multimorbidity may help inform clinical decision to improve patient services and outcomes, as well as help to relieve some the burden on healthcare services and available resources. Moreover, further studies into multimorbidity may lead to improved quality of life and reduced mortality of those individual suffering from multimorbidity.

1.1 SAIL DataBank - Wales multimorbidity study cohort

The collections of linked health datasets into a single research environment creates a number of large opportunities for insight and analysis into multimorbidity investigation. Whilst anonymised extracts from these environments have great value, there is an even larger opportunity to develop methods of deploying modelling to the data remotely. The SAIL Databank - The Secure Anonymised Information Linkage DataBank provides anonymised person-based data for research powered by the Secure e-Research Platform.

The Wales Multimorbidity e-Cohort (WMC) [4] was created to provide an accessible research ready data asset to further the understanding of multimorbidity. Their objectives were to create a platform to support research which would help to understand prevalence, trajectories and determinants in multimorbidity, characterise clusters that lead to the highest burden on individuals and healthcare services, and evaluate and provide new multimorbidity phenotypes and algorithms to the NHS and research communities to support prevention, healthcare planning and the management of individuals with multimorbidity.

The WMC has been created and derived from multi-sourced demographic, administrative and electronic health record data relating to the Welsh population in the Secure Anonymised Information Linkage (SAIL) Databank. The WMC consists of 2.9 million (2,902,101 exactly) people from Wales, collecting longitudinal data from multiple sources from the 1st January 2000 with follow-up until 31st December 2019, Welsh residency break or death. The primary and secondary care data sources also provide rich historical records to track disease progression over the course of an individuals lifetime. Table 1 describes the different datasets combined to create the WMC.

Data Source	Period Covered	WMC Individuals
Critical Care Data Set	01/01/2007 → 31/12/2019	2.7%
Welsh Cancer Incidence Surveillance Unit	01/01/2000 → 31/12/2016	11.3%
Welsh Results Reporting Services	01/01/2015 → 10/12/2018	53.1%
Emergency Department Dataset	01/01/2009 → 31/12/2019	54.4%
Patient Episode Database for Wales	01/01/2000 → 31/12/2019	73.4%
Out Patient Dataset for Wales	01/01/2004 → 31/12/2019‘	75.0%
Welsh Longitudinal General Practice	01/01/2000 → 31/12/2019	82.7%

Table 1: Clinical data sources available for the WMC [4].

The WMC e-Cohort boasts a balanced dataset, equally representing both male and female populations (51% and 49%, respectively) and deprivation quintile demographics based on the Welsh index for multiple deprivation (WIMD). Moreover, the mean age of the population from cohort inception was 39, covering 20 years of an individuals lifetime assuming they survived by December 31st 2019. 19% of the cohort died during the period of analysis and 22% of the cohort moved from Wales due to a residency break or emigration after January 1st 2000.

Among others code lists and processed data tables, two published comorbidity indices were used to measure and conceptualise multimorbidity in the WMC e-cohort using the International Classification of Diseases 10th revision (ICD-10). These include the Charlson and Elixhauser comorbidity indices. Their comorbidities can be found in appendix C. These two standardised comorbidity tables have been used to build two distinct data tables, flagging Date-Time instances where individuals were first observed to have one or more of these conditions found in either table during an interaction with the healthcare system in Wales.

- **Charlson Comorbidity Index**: The Aylin and Bottle Charlson amended ICD-10 code list [5] was used for inpatient diagnoses and Metcalfe et al Charlson Read Code list [6] was utilised for primary care recorded diagnosis. The ICD-10 codes have been taken from the pool of diagnosis codes recorded within hospital admissions data, containing 1024 distinct codes at the four-character level for 16 conditions. The GP data contains 4545 distinct Read Codes at the five-character level. For downstream analyses, these 16 conditions were reduced to 13 to prevent pseudo-clustering in the results. The three related sets of diseases were (i) cancer, lymphoma and leukaemia and metastatic cancer, (ii) mile and severe liver disease and (iii) diabetes with and without complication.
- **Elixhauser Comorbidity Index**: The Elixhauser ICD-10 code list was utilised for inpatient diagnosis [7] and Metcalfe et al Elixhauser Read Code list was utilised for primary care diagnosis [6]. The ICD-10 codes have been taken from the pool of diagnosis codes recorded within hospital admissions data and contains 1423 distinct codes at the four character level for 30 conditions. The GP data contains 6074 distinct Read Codes at the five-character level. Similar to the Charlson comorbidity index, the 30 conditions were reduced to 27 comorbidities to prevent pseudo-clustering in the results. The three related sets of diseases were (i) hypertension with and without complication, (ii) diabetes with and without complication and (iii) lymphoma and metatstatic cancer.

1.2 Project objectives and outcomes

Recent work using graph models to build simpler knowledge discovery systems opens up potential for increased prediction accuracies, reduced pre-processing burden and the application of models of higher complexity to our data. These models have been shown to effectively handle messy data and to learn representation of key factors from the data directly (rather than choosing a set of predictor variables).

A recent paper [8] and accompanying GitHub repository [9] demonstrates applying a hypergraph analysis to a comorbidity investigation by identifying important multi-disease sets according to multimorbidity prevalence. A follow-up paper awaiting peer-review applied the same techniques to the SAIL databank but these important multimorbidity disease sets were determined according to healthcare resource utilisation. This project sought to develop upon this work, investigating different avenues of further complexity such as adding demographics and directionality into the model, investigating different weighting systems and studying the use of mortality within the graph.

In particular, this project has mainly focused on the development of methodology capable of modelling multimorbidity disease progressions using directed hypergraphs. This project has successfully shown that we can understand the observed progression of multimorbidity in a population through the lens of multimorbidity prevalence. Through centrality-based measures such as PageRank, we have also been able to assign directional relationships diseases have with one another through observed patient disease progressions. Moreover, Eigenvector Centrality can also rank

important disease progressions to help towards detecting contributory relationships disease have with one another. The project has also implemented finality into these observed patient disease progressions by considering different avenues of including mortality into the hypergraph models. Finally, we are able to stratify the data accordingly so that we can gain a more comprehensive and detailed understanding of these relationships for different demographics such as age, sex and deprivation status, three major contributors to multimorbidity [10].

It is hoped that the work in this project will be used to provide insight into a powerful and flexible model for complex data sources to help inform policy and strategy decisions. Given this is the first collaboration between NHS England and the SAIL platform, the project also set out to help build suitable skills and ways of working within the SAIL platform for future collaborations. Moreover, from a developmental perspective moving forward from the original work in [8], this project looked to inform the direction of future research using hypergraphs with SAIL data and other similar sources. With a lens toward future endeavours, the project has highlighted some important avenues of further exploration, such as adding demographic factors directly into the graph, or more rigorous modelling of so-called episodic conditions. These areas for future work hope to help inform the direction of future research using hypergraphs with SAIL data and other similar sources.

This report is outlined as follows. Section 2 briefly outlines some related work studying multimorbidity using unsupervised approaches such as simple undirected graphs, simple directed graphs, hierarchical clustering and latent class analysis. Section 3 then describes hypergraphs and the different mathematical constructs and notation used to express these complex models, with section 4 defining our exploration of different weighting systems, an integral part of the model set-up and interpretation of subsequent results. Section 5 defines the directed hypergraph and how we can use it to model multimorbidity disease progressions.

Once our undirected and directed hypergraph models are defined, section 6 outlines the downstream analysis we can perform on these complex structures. Although many other techniques for analysing graph-based techniques exist, the project focused on the implementation of centrality for both the undirected the directed case. Finally, section 7 sets out future extensions, such as how to directly include demographic factors into the graph so they can influence one another, or how to potentially deal with acute, episodic illnesses which are captured by these standard comorbidity indexes and aren't traditionally percieved as *chronic* illnesses.

2 Related work

2.1 Cluster Analysis

Clustering attempts to find distinct and disjoint partitions of the data that best represent the overall structure within it. The typical set up attempts to determine k partitions on data V, $\pi_V = \pi_1, \pi_2, \ldots, \pi_k$, where $\pi_i \subsetneq V$, $\cup_i \pi_i = V$ and $\cap_i \pi_i = \emptyset$. In the context of multimorbidity, V could be a set of diseases or a population.

2.1.1 Hierarchical cluster analysis

Hierarchical algorithms are most appropriate for classification problems where objects are related via some underlying systematic structure. Given the expectation that multimorbidity clusters arise from common underlying factors or as a multiple sequelae of a primary condition, a hierarchical algorithm is more appropriate than nonhierarchical clustering approached for detecting meaningful morbidity clusters. These clustering algorithms follow a process which iteratively tries to improve upon its choice of partitions, or clusters, while attempting to minimise inter-cluster similarity and maximise intra-cluster similarity. The concept of similarity between objects to cluster is intrinsic with cluster analysis and is normally in the form of some distance metric.

In the case of multimorbidity the objects would be diseases, or a population categorised by the diseases each individual suffers from, and the measure of proximity may be some weighted measure of prevalence of that disease within a population [11, 12]. The typical set up here involves a binary flag matrix on which the similarity matrix is defined, which underpins the results of the cluster analysis which hierarchical clustering computes. The similar matrix quantifies the relationship between the objects being clustered, i.e. the individuals of diseases of interest. Popularity similarity indexes used include the Jaccard index [11], or Yule's Q coefficient [13].

Agglomerative clustering is a bottom-up approach where initial clusters represent singular objects and are iterative merged to create one final cluster containing all objects. Unlike other clustering methodologies like k-means clustering, there is no predefined number of clusters, k. Instead, the number of partitions k is usually chosen according to some statistic comparing the intra- and inter-cluster variance for clustering results at different values of k, usually decided using Ward's linkage or Flexible-β linkage [11]. Popular visualisations such as dendrograms are used to inspect the

quality of the cluster results, as well as other more statistically rigorous approaches such as the adjusted rand index and the Calinski-Harabasz pseudo-F statistic, two commonly used in healthcare clustering studies.

2.1.2 Latent class analysis

relates a set of discrete, observed multivariate variables to a set of *latent* (unobserved) variables [14]. It is called a latent class model because the latent variables are also discrete. Covariance structure analysis (such as exploratory factor analysis) provides a popular framework for mapping items onto continuous latent variables. Latent class analysis provides an analogous framework for measuring *categorical* latent variables. Given the input of a binary flag matrix containing multivariate, dichotomous variables (features) representing disease diagnoses for each participant (observation), latent class analysis is a more appropriate probabilistic model than other similar approaches like factor analysis (FA) or principal component analysis (PCA). The latent class model divides a population into mutually exclusive and exhaustive subgroups, and so can be considered to be a probabilistic and qualitative form of cluster analysis that does not rely on similarity metrics.

Like cluster analysis, latent class is aimed at identifying classes of observations that are in some sense 'similar' but does so according to probabilities of the observed values of all variables for each observations. In the context of multimorbidity, latent space models have been used to identify unique classes of individuals which suffer from similar multimorbidity [13, 15, 16].

There have already been studies focusing on analysis with either one methodology or a multi-methodology comparative analysis of various techniques into investigating comorbidity and multimorbidity [8, 13, 16, 17, 18, 19]. These studies have highlighted several different approaches to tackle identifying meaningful multimorbidity clusters using hierarchical cluster analysis, latent class analysis and graph-based techniques.

Other methodologies exist and have been employed to study multimorbidity such as deep learning based approaches or multi-state modelling. Multi-state modelling is an interesting approach to investigating multimorbidity, as it tracks disease states of individuals. [20] used multi-state modelling for deducing hazard ratios for survival analysis. Survival models have also played a role in the study of multimorbidity. This kind of supervised learning would be used to understand which multimorbidity sets are related to those with the poorest outcomes (highest mortality rate) and examine the impact of multimorbidity on mortality [19, 20, 21].

2.2 Network Analysis

2.2.1 Simple graph

Network analysis has been previously used to investigate multimorbidity [22, 23]. Network analysis uses available data to construct a mathematical object called a graph, in which pairwise relationships between elements in one set of objects, called nodes, define elements of a second set of objects called edges. In classical graph theory, the simple graph only allows two nodes to be connected using at most one edge. Therefore, the simple graph is restricted to only modelling pairwise relationships. Figure 2 defines a graph $G(V, E)$ whose node set $V = \{1, \ldots, 5\}$ and edge set $E = \{e_1, \ldots, e_8\}$. Note that each edge only connects to two nodes at most.

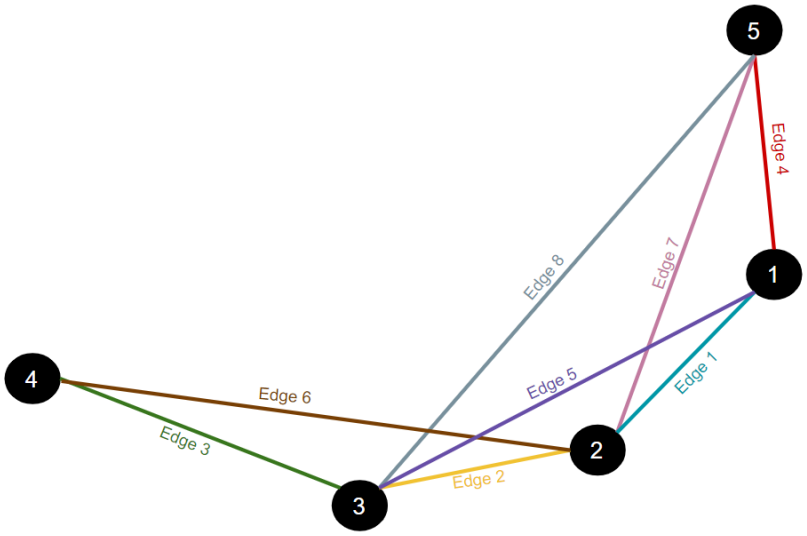

Figure 2: A simple graph G.

In the context of multimorbidity, the nodes here represent diseases and the edges represent disease sets. However, as the edges only model pairwise relationships, information is lost for cases where three or more conditions are observed intra-patient. Figure 3 shows an example of a simple graph applied to a multimorbidity study [23]. However, note that the graph is restricted to only model comorbidity.

In graph theory, the relationships you build between nodes are quantified to represent the strength of connection between any two nodes. These are called the edge weights, and figure 3 visualises these edge weights using the thickness of the edges, where for example the comorbidity hypertension and hyperlipidemia seem to have the strongest connection in the population being studied. The exact way in which these relationships are quantified is discussed in detail in section 4.

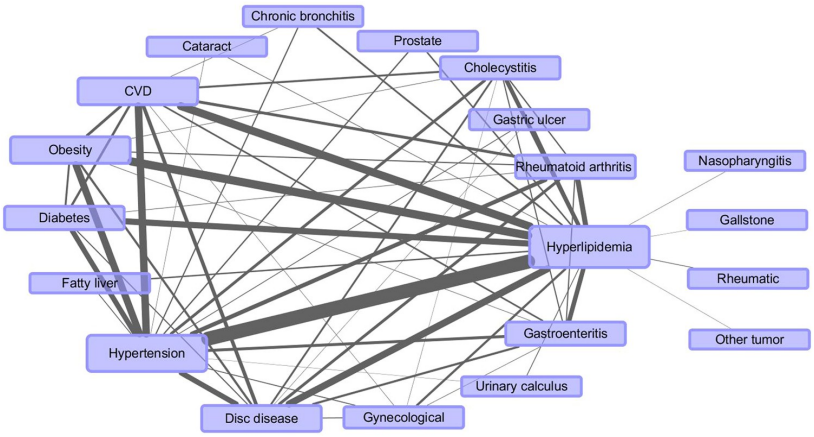

Figure 3: A weighted comorbidity graph G_W [23].

The mathematical representation of graph theory and how it is used for downstream analysis is discussed later in this section once hypergraphs are defined.

2.2.2 Directed graphs

A directed graph is where we impose directionality among the edges and therefore some directionality, or ordering, among the nodes. Let $G_D(V, E_D)$ be a directed graph with the same nodes as G above but with a directed edge set E_D. Figure 4 represents an exemplar directed graph from the same system of nodes and edges we had above from graph G.

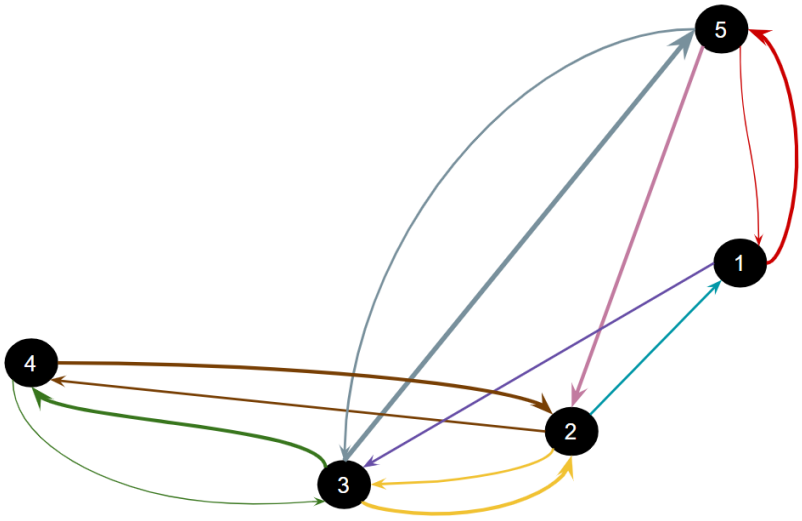

Figure 4: A directed graph G_D using the same system of nodes and edges as simple graph G.

Note that each original edge from E now gives rise to 2 possible directed edges, each with opposite directionality. Therefore, nodes now have incoming and outgoing connections. The visualisation has introduced edge weights using the thickness of each connection. Edge weights in the directed graph are now related to the direction of the edge. For example, node 3 has strong outgoing connections while weak incoming connections. Not every edge in E need give rise to an outgoing and incoming connection in E_D, for example, the original edge e_1 connecting nodes 1 and 2 only gave rise to an outgoing connection from node 1 to 2. The takeaway from this brief description is that each undirected *parent* edge in graph G gives rise to at most 2 **children** directed edges in the directed graph G_D.

Directed graphs are typically used to build flow maps of a system, for example navigating the internet through the use webpages (nodes) and hyperlinks (incoming and outgoing connections). In the context of multimorbidity, a directed graph could represent comorbid disease progressions, where we interpret the outgoing connection of node i to node j to represent the disease progression $i \to j$, i.e disease i was incident before disease j was then coincident with disease i some arbitrary time later. [21] does a comprehensive analysis of UK multimorbidity data using simple network analysis using eigenvector and degree centrality measures and community clustering. They added direction into the network using chronology of disease diagnosis dates and then performed survival analysis. They also stratify the data according to age and socioeconomic status using 10 different age groups (ranging from infancy to elderly) as well as the popular UK wide measure of deprivation, IMD. They included a temporal element into the analysis by constructing the network for each year of data they had available.

3 Undirected hypergraphs

An extension to the classical graph structure is a hypergraph. This permits each edge, referred to as a hyperedge, to link multiple nodes therefore allowing the incorporation of data on people with any number of conditions. In the context of multimorbidity, using hypergraphs allows for the investigation of detecting multimorbidity clusters using the same analysis techniques as simple graphs, i.e. spectral clustering and centrality. A lot of the information (and, in particular the examples) on hypergraphs, dual hypergraphs and bipartite graphs comes from Jim's original paper [8].

Similar to a simple graph, a hypergraph $\mathcal{H}(V, \mathcal{E})$ which connects nodes from V using the hyperedges in $\mathcal{E}$. The set V contains the nodes $\{v_0, v_1, \ldots, v_n\}$, the set $\mathcal{E} = \{e_1, e_2, \ldots, e_m\}$ contains the hyperedges such that each element $e_i \subseteq V$ to represent which nodes the hyperedge is connecting. As previously mentioned, in contrast to a classic graph, each hyperedge $e_i = \{v_{i_1}, v_{i_2}, \ldots, v_{i_k}\}$ can connect a subset of any size of V. Figure 5 shows an example hypergraph $\mathcal{H}(V, \mathcal{E})$ which is the hypergraph equivalent of the graph $G(V, E)$ introduced in figure 2 from section 2.2.1. Note that the hyperedges $e_1, \ldots, e_4$ are the same undirected edges in $G(V, E)$ but hyperedges $e_5, \ldots, e_8$ now represent connections of more than 2 nodes, which is something that $G(V, E)$ is unable to do.

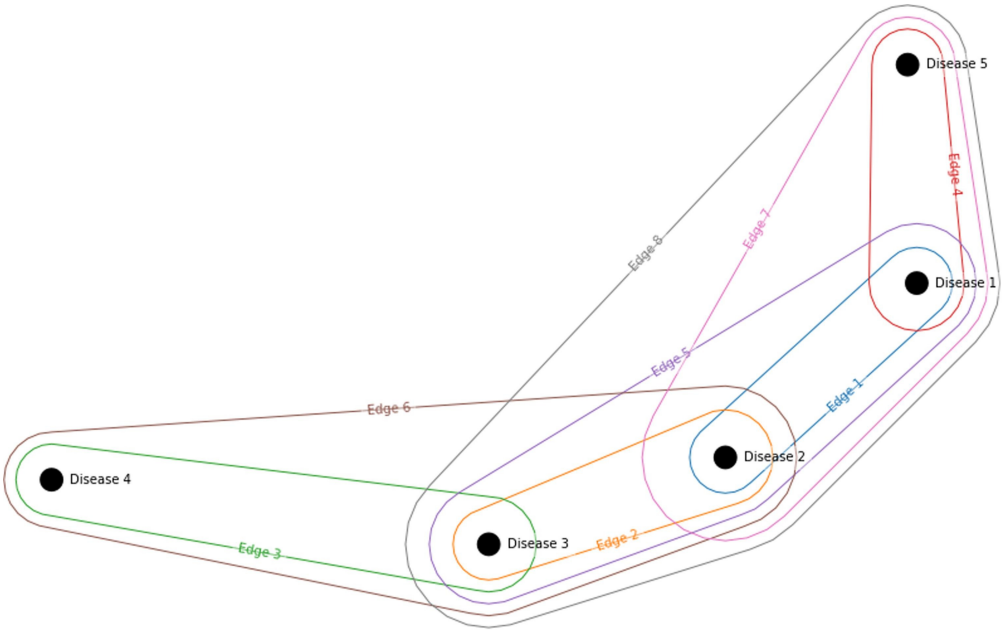

Figure 5: Example hypergraph $\mathcal{H}$ [8] equivalent from graph G in figure 2.

The additional relationships hypergraphs allow directly permit the study of multimorbidity. Hypergraphs are a natural extension to the simple graph, where multi-way relationships, as well as pairwise ones, can be captured. Much like how simple graphs are able to measure pairwise relationships, i.e. comorbidities, the hypergraph is now able to study multimorbidities.

In typical constructions of hypergraphs in network analysis, much like in normal graphs, hyperedges are assigned weights and these weights are dependant on the nodes whose edge they are connecting. In the case for multimorbidity data this is typically some population measure related to some important outcome such as disease prevalence or mortality rate. Here, the hypergraph $\mathcal{H}(V, \mathcal{E}, \mathcal{W}_V, \mathcal{W}_\mathcal{E})$ now has the additional objects called the node weights W_V and hyperedge weights $W_\mathcal{E}$. This information is formed using the diagonal matrices W_E and W_V (note that the individual weights $[W_E]_{ii} = w(e_i) \in \mathcal{W}_\mathcal{E}$ and similarly for W_V and $\mathcal{W}_V$). W_E is a diagonal, $m \times m$ matrix with non-zero entries representing the weight of the corresponding hyperedge, while matrix W_V is a diagonal, $n \times n$ matrix with non-zero entries representing the weight of the corresponding node. Without loss of generality, we will for now assume the node and hyperedge weights are non-negative.

3.1 Notation

Let $\mathcal{H}(V, \mathcal{E})$ denote a hypergraph with node set $V = \{v_1, v_2, \ldots, v_n\}$ and edge set $\mathcal{E} = \{e_1, e_2, \ldots, e_m\}$. The edges are arbitrary subsets of V with weight $w(e)$ associated with hyperedge e. For the remainder of this section we will assume there are $n = |V|$ nodes and $m = |\mathcal{E}|$ edges. Moreover, the terms 'hyperedge' and 'edge' will be used interchangeably throughout this document.

A useful representation of a hypergraph is the *incidence* matrix M, which is an $n \times m$ matrix where each edge is represented by a column of M and each node is represented by a row. The incidence matrix can be considered a fundamental representation of the hypergraph, as there is a one-to-one correspondence between a hypergraph and their corresponding incidence matrix. We use the incidence matrix to understand which nodes are connected and by which edges. The incidence matrix M for the hypergraph in figure 5 is

$$M = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \end{pmatrix} \qquad M(v, e) = \begin{cases} 1 & \text{if edge } e \text{ connects node } v \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Other measures we can compute on the hypergraph are node and edge degrees. The node degree $d(v)$ of a node v and the edge degree $\delta(e)$ for an edge e are defined as

$$d(v) = \sum_{e \in \mathcal{E} | v \in e} w(e) \quad \text{and} \quad \delta(e) = |e|. \tag{2}$$

For the case of simple graphs (graphs whose edges only connect two nodes, without self-connecting edges), these are also known as 2-graphs, and their edge degree is fixed at $\delta(e) = 2$ (since edges can only connect 2 nodes). Given the node-edge incidence matrix M is an $n \times m$ matrix where $M(v, e) = 1$ if $v \in e$ and 0 otherwise, we can define $d(v)$ and $\delta(e)$ using M such that

$$d(v) = \sum_{e \in \mathcal{E}} w(e) M(v, e); \tag{3}$$

$$\delta(e) = \sum_{v \in V} M(v, e). \tag{4}$$

Matrices D_n and D_e are the diagonal matrices of the node and edge degrees, $d(\cdot)$ and $\delta(\cdot)$, respectively. For nodes $u, v \in V$ we have that $D_n(u, v) = d(v)$ for $u = v$ and 0 otherwise. The diagonal matrix D_e is defined similarly such that for edges $e_i, e_j \in \mathcal{E}$, $D_e(e_i, e_j) = \delta(e_i)$ where $e_i = e_j$ and 0 otherwise.

We define the *unweighted* adjacency matrix of simple hypergraph $\mathcal{H}$ as the square matrix whose rows and columns are indexed by the nodes of $\mathcal{H}$ and where. Formally, we define this matrix as

$$[A]_{ij} = |\{e \in \mathcal{E} \ : \ v_i, v_j \in e\}| = |\mathcal{E}_{v_i} \cap \mathcal{E}_{v_j}|, \tag{5}$$

where $\mathcal{E}_{v_i}$ is the subset of edges which connect node v_i. Therefore, for all $u, v \in V$, $u \neq v$: $a_{uv} = |\{e \in \mathcal{E} \ : \ u, v \in e\}|$ and $a_{uu} = 0$ (as we do not allow self-connected edges). This set formulation simply describes the number of hyperedges, or connections, that nodes u and v share together, and is therefore a measure of how connected u and v are in the hypergraph. The matrix formulation for the unweighted adjacency matrix is

$$A = MM^{\mathrm{T}} - D_n. \tag{6}$$

Note that adjacency matrix A is a non-negative symmetric matrix with real elements. One must be careful when interpreting the adjacency matrices of hypergraphs, because the adjacency matrix for nodes cannot be used to distinguish between one very heavily weighted edge and a set of edges with large weights. The setup of incidence, degree and adjacency matrices is identical to a classic graph, but since edges in a normal graph are only allowed to connect two node, the columns of matrix M can only have at most two ones. Moreover, for an unweighted graph, the adjacency matrix A is a *binary*, non-negative and symmetric matrix. This is because in a classic graph, only one edge can contribute to each adjacency matrix element.

For a *weighted* hypergraph, the $m \times m$ edge weight matrix W_E storing all edge weights are taken into account for this adjacency matrix such that the $n \times n$ matrix is defined by

$$A = MW_E M^{\mathrm{T}} - D_n, \tag{7}$$

The set formulation for this matrix now takes into account the edge weights such that for all $u, v \in V$, $u \neq v$:

$$a_{uv} = \sum_{\{e \in \mathcal{E} \, : \, u, v \in e\}} w(e), \tag{8}$$

where of course $a_{uu} = 0 \ \forall u$ as a result of subtracting the node degree matrix D_n from $MW_E M^{\mathrm{T}}$.

For the hypergraph in figure 5, the diagonal elements of the the node degree matrix $\mathrm{diag}(D_n) = [5, 6, 5, 2, 3]$. For example disease node 1 is connected to 5 edges. The diagonal of the edge degree matrix $\mathrm{diag}(D_e) = [2, 2, 2, 2, 3, 3, 3, 4]$ as for example $e_8 \in \mathcal{E}$ connects disease nodes 1, 2, 3 and 5. The unweighted adjacency matrix for the hypergraph in figure 5 is

$$A = \begin{pmatrix} 0 & 4 & 2 & 0 & 3 \\ 4 & 0 & 4 & 1 & 2 \\ 2 & 4 & 0 & 2 & 1 \\ 0 & 1 & 2 & 0 & 0 \\ 3 & 2 & 1 & 0 & 0 \end{pmatrix} \quad [A]_{ij} = \begin{cases} k & \text{if node } i \text{ is connected to node } j \text{ through } k \text{ edges;} \\ 0 & \text{if } i = j \text{ or if zero edges connect nodes } i \text{ and } j. \end{cases} \tag{9}$$

3.2 Graphs vs. Hypergraphs

In the application of multimorbidity, a simple graph only matches pairwise disease sets and so restricts any further analysis to look at sets of 3 or more diseases. We can look at the dual and bipartite representation of simple graphs for studying both single set and/or pairwise sets of disease, but this will still only involve any analyses of disease sets of only 1 or 2 diseases (these two representations are detailed below). This is a major limitation of simple graphs in comparison to hypergraphs. Through computing centrality — a popular measure of node influence and importance across the networks, discussed in more detail in section 6 — on the adjacency matrices of simple graphs and hypergraphs, simple graph centrality will miss key elements from potential hyperedges contributing information to diseases which are part of sets of 3 or more diseases. Moreover, all possible simple graph constructions are but a subfamily of hypergraphs (referred to as 2-uniform hypergraphs), providing even more evidence to the fact that hypergraphs are superior to simple graphs for modelling more complex relationships.

However, a disadvantage to hypergraphs is the computational resource required to build one in comparison to a simple graph. For a simple graph of n nodes, there are a maximum of $\binom{n}{2} = \frac{n!}{2(n-2)!}$ edges, i.e. for the graph shown in figure 2 there are $n = 5$ nodes, therefore meaning there is only $\binom{5}{2} = 10$ possible edges. For a hypergraph, the number of hyperedges grows exponentially with the number of nodes, and in the case of $n = 5$ as seen in figure 5, there are 26 hyperedges. Mathematically, for a simple hypergraph of n nodes, there are $2^n - (n+1)$ hyperedges (assuming we exclude self-connecting hyperedges). This is because for a hypergraph of n nodes, we can have $\binom{n}{k}$ possible combinations for hyperedges of degree $2 \leq k \leq n$, (since we exclude self-connecting hyperedges). This means the upper bound for the number of edges $|\mathcal{E}|$ for a hypergraph $\mathcal{H}$ is

$$|\mathcal{E}| \leq \sum_{k=2}^{n} \binom{n}{k} \tag{10}$$

$$= \sum_{k=0}^{n} \binom{n}{k} - n - 1 \quad \text{since } \binom{n}{0} = 1 \text{ and } \binom{n}{1} = n. \tag{11}$$

$$\implies |\mathcal{E}| \leq 2^n - (n+1) \tag{12}$$

This means that to construct a hypergraph, the computational resource scales exponentially with the number of nodes. For example, for a simple graph of 20 nodes, there are at most 190 edges, while in the hypergraph there are over 1 million possible hyperedges. In practice however, not all maximum edges will be used, but this requires careful consideration on constructing the hypergraph.

3.3 Dual hypergraph

While the typical, weighted hypergraph gives rise to studying the nodes of the graph, i.e. the single disease sets, we can use the dual hypergraph to study the hyperedges. This is an important representation for our downstream analysis using centrality. The dual hypergraph is where the original hypergraph's incidence matrix has been transposed such that the incidence matrix of the dual hypergraph, $\mathcal{H}^*$ is now $M^* = M^{\mathrm{T}}$. This flips the nodes and edges such that $\mathcal{H}^*$ has m nodes and n edges so that the single disease sets (originally pitched as nodes in $\mathcal{H}$) become edges linking between different sets of diseases (originally pitched as edges in $\mathcal{H}$) which are now the nodes of $\mathcal{H}^*$.

Figure 6 shows a simple, 3-node hypergraph $\mathcal{H}_3$, whose nodes and edges are colour coded. The corresponding dual hypergraph $\mathcal{H}_3^*$ is shown to the right, swapping the roles of the nodes and edges. Now the nodes of the original hypergraph are interpreted as the connections linking the hyperedges which contain that node.

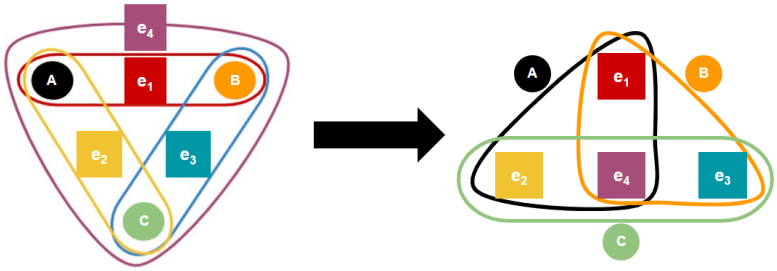

Figure 6: Dual hypergraph $\mathcal{H}_3^*$ of a simple 3-node hypergraph $\mathcal{H}_3$.

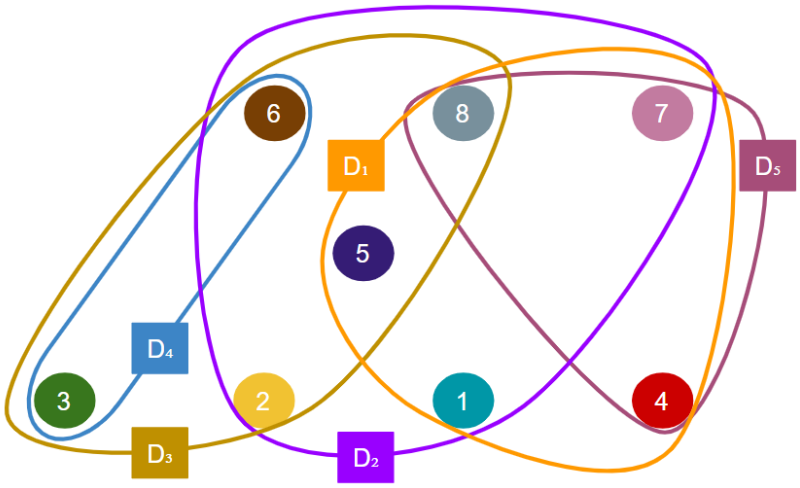

Figure 7: Dual hypergraph $\mathcal{H}^*$ of the original hypergraph $\mathcal{H}$ seen in figure 5. The numerical nodes are the edge indexes.

In the context of multimorbidity, figure 7 shows the dual hypergraph $\mathcal{H}^*$ of the original hypergraph $\mathcal{H}$. Now the nodes are the coincident disease sets while the edges are the individual diseases linking the sets of diseases where each disease is present in. For example, the edge for D_4 only links together the nodes $\{D_4D_3, D_4D_3D_2\}$ as these are the only nodes containing disease 4, D_4. Since the incidence matrix $M^* = M^{\mathrm{T}}$ we define the adjacency matrix similarly to the original hypergraph as

$$A^* = M^{\mathrm{T}} W_V M - D_e, \tag{13}$$

where W_V is the $n \times n$ matrix of dual hyperedge weights, i.e. the node weights and D_e is the $m \times m$ diagonal matrix representing the hyperedge degrees, i.e. the number of diseases within each set/node. In this case the incidence matrix will be

$$M^* = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{pmatrix} \qquad M(e,v) = \begin{cases} 1 & \text{if node } v \text{ connects edge } e \\ 0 & \text{otherwise} \end{cases} \tag{14}$$

The weights stored in W_V are unrelated to the edge weights described above and are instead properties of the single diseases, i.e. the original nodes of the hypergraph $\mathcal{H}$. Note that the dual hypergraph adjacency matrix does not depend on the edge weights of the original hyperedges. To allow the edge weights W_E to influence this adjacency matrix, we define the adjacency matrix of the weighted resultant dual hypergraph as

$$A^*_W = \sqrt{W_E}(M^{\mathrm{T}} W_V M - D_e)\sqrt{W_E}, \tag{15}$$

Where we know that W_E is diagonal, so $W_E^{\mathrm{T}} = W_E$ and $\sqrt{W_E} = \mathrm{diag}(\{\sqrt{[W_E]_{ii}} : \forall i\})$. Any measures computed on this weighted resultant adjacency matrix now takes both the node and edge weights into account. This is important when we discuss eigenvector centrality in section 6. In short, [8] was able to use the hypergraph and eigenvector centrality to not only measure the transitive influence and importance that single set diseases have in the original hypergraph representation, but using the dual hypergraph representation permitted measuring that same influence and importance for multimorbidity disease sets.

4 Weighting systems

4.1 Edge weight dependencies

There is a large degree of flexibility in choosing how to compute the weights of the edges and the nodes. In the context of modelling multimorbidity and using a population cohort such as the WMC E-cohort, there are three questions highlighting the different dependencies that an edge weighting system has:

pat_id	age_inception	sex	depr	disease_A	disease_B	disease_C	death_date
1234	71	0	3	2001-01-01	2004-07-31	2004-07-31	2004-08-12

Figure 9: Hypothetical observation from processed Charlson and Elixhauser population data tables.

1. **Outcome**-dependence: what attribute of the nodes, i.e. diseases, is used to measure the relationship of coincident nodes, i.e. the hyperedges?
2. **Contribution**-dependence: given a population cohort, how does each individual contribute to the graph, i.e. which hyperedges do they contribute to?
3. **Formula**-dependence: Given the above outcome measure and contribution type, how are these combined mathematically to measure the quantitative relationship a hyperedge, or multimorbidity disease set, has relative to other hyperedges?

It should be noted that with the choice of weighting system decides on the interpretation of the analysis. The choice of the edge weighting system for the hypergraph has been decomposed into three components, as seen in figure 8, with different options for each component identified.

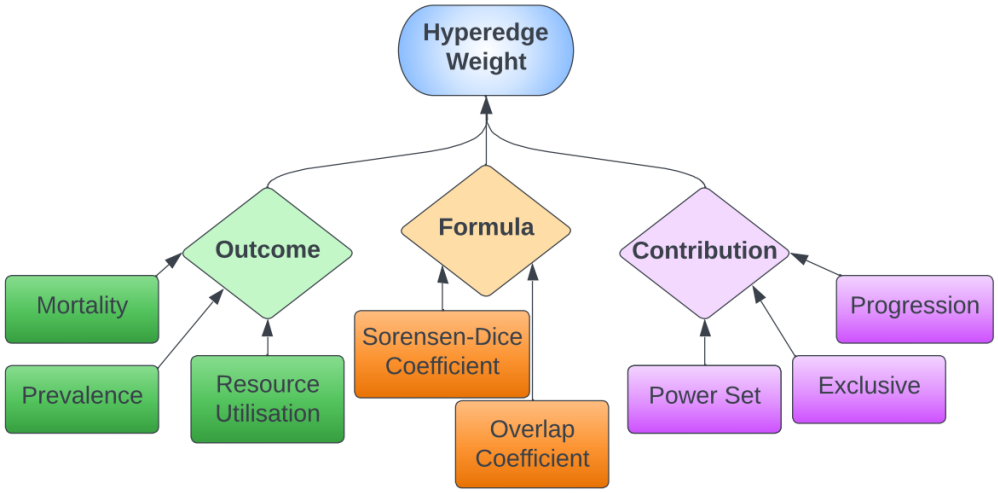

Figure 8: Components of an edge weight system for a multimorbidity hypergraph.

Figure 9 shows an example observation in the processed population data using a hypothetical set up of three diseases A, B and C. This example will be used to describe each of the contribution types below. Note that we have information on the individual's age at inception, their sex, deprivation status (according to WIMD), their date of death (which would otherwise be left blank if they didn't die by the end of the period of analysis) and date-time instances where a disease was first observed during some interaction with the healthcare system. An important observation of this individual is that that the date-time instance diseases B and C were first observed by the healthcare system is the same. Therefore, there is no distinct ordering of disease progression for this individual, as we don't know if disease B came before C or vice versa. This is but one important caveat of the data and needs to be taken into account when constructing the hypergraph model and the hyperedge weighting system.

4.2 Outcome

There were three outcome measures considered during the construction of the hypergraph weights:

- Multimorbidity prevalence: the weight for a multimorbidity disease set could be a measure of disease set prevalence, i.e. the proportion of individuals in the population exhibiting the set of diseases the hyperedge represents. [8] used multimorbidity prevalence to show the utility hypergraphs have in modelling multimorbidity. Using centrality here would mean we could detect which single or set of conditions were the most important based on their prevalence among patients in the cohort.

- Mortality rate: the weight for a multimorbidity disease set could be the proportion of people who died with those coincident diseases represented by the hyperedge. Using centrality here would mean we could detect which single or set of conditions were the most important according to those diseases which have the poorest outcomes in patients.
- Healthcare resource utilisation: the weight for a multimorbidity disease set could be the proportion of people who died with those coincident diseases represented by the hyperedge could be some measure of the amount of healthcare resources required to treat individuals exhibiting the set of diseases. [24] constructed hyperedge weights based on inpatient and outpatient utilisation of participants with those single or sets of conditions. Using eigenvector centrality here meant that we could detect which single or set of conditions were the most important in terms of utilising healthcare resources with respect to other single or multiple sets of conditions.

In our model, we selected **multimorbidity prevalence** as our node attribute when computing our hyperedge weights. There was some consideration into using healthcare resource utilisation, and even combining both measures as discussed in section 7, but the work in sections 5 and 6 meant we were time constrained. While **mortality rate** is typically used for survival analysis or multi-state modelling [20, 19], this measure has been considered in this project. Given that we are modelling multimorbidity disease progressions, we can consider introducing finality into these progressions by the inclusion of mortality nodes, as discussed in section 5.5.

4.3 Contribution

The contribution system decides on which hyperedges an individual contributes to in the hypergraph. We will use the example seen in figure 9 to demonstrate what hyperedges each contribution type allows the individual to contribute prevalence to.

4.3.1 Power set

The power set contribution computes the power set of the individuals final disease set. For example, our individual in figure 9 has been observed to have all three conditions $\{A, B, C\}$. Therefore, this particular individual will contribute all elements of the power set of their multimorbidity disease set $\{A, B, C\}$ which is

$$\mathcal{P}(\{A, B, C\}) = \Big\{\{A\}, \{B\}, \{C\}, \{A, B\}, \{A, C\}, \{B, C\}, \{A, B, C\}\Big\}. \tag{16}$$

The motivation behind this is that the individual contributes prevalence to all combinations of their final multimorbidity set, because all conditions are present in the individual, as well as subsets of the condition set. The disadvantage of this contribution type however is that disease influence is ignored, i.e. allowing the individual to contribute prevalence to disease set $\{A, C\}$ ignores the potential influence that disease B has on diseases A and B.

4.3.2 Exclusive

The simplification therefore is to restrict an individual to only contribute prevalence to the hyperedge which connects all of their diseases in their final multimorbidity set. In our example in figure 9, the only hyperedge the individual would contribute to is the one connecting all three diseases, i.e. $\{A, B, C\}$.

The interpretation here is that we've taken a snapshot of the individuals multimorbidity disease state at the end of the period of analysis and built our prevalence count from that snapshot. An immediate disadvantage is the complete ignorance of time. Given we know the date-time instances where diseases A, B and C

4.3.3 Progression

'Progression' contribution was where a middle ground was reached between the 'Power' set and 'Exclusive' set contribution types. Here is where we could take advantage of an individual's date-time interactions with the healthcare system to select the hyperedges the individual should contribute to based on their **observed** disease progression. Moreover, this contribution type is able to model both temporal and inter-disease interactions, while the other two contribution types aren't. There are two types of progression contribution considered:

1. Simple progression: Because graphs are a subfamily of hypergraphs, we could build a hypergraph whose maximum degree hyperedges are fixed at 2. This enables suitable comparison between the results from a hypergraph representation with its inferior and simpler subfamily of representations discussed in section 2.2.

this progression type selects hyperedges of maximum degree 2 which obey the individual's ordered condition set. For example, given the example observation in figure 9, the hyperedges contributed to would be

$$\Big\{\{A\}, \{A, B\}, \{A, C\}, \{B, C\}\Big\}. \tag{17}$$

This is because disease A was the first condition observed by the healthcare system, with diseases $B$and C following disease A.

2. Aggregate progression: When we allow hyperedges of any degree, we can allow the individual to contribute prevalence to hyperedges which form an aggregate perspective on their observed disease progression. Therefore, we select hyperedges which increase in degree with each newly observed condition in their ordered condition set. In this case, the hyperedges contributed to would be

$$\Big\{\{A\}, \{A, B\}, \{A, C\}, \{A, B, C\}\Big\}. \tag{18}$$

Note that if the date for diseases B and C weren't the same, only one of $\{A, B\}, \{A, C\}$ would be contributed to, depending on which disease had been observed first by the healthcare system. The advantage here is that we can model each stage of the individual's multimorbidity path, i.e. beginning with disease A, then moving onto either disease B or C.

4.3.4 Example dataset

The left-hand table in 2 defines 10 example individuals using three hypothetical diseases A, B and C. The right-hand table then describes exactly which hyperedges each individual contributes their unitary prevalence to.

Observed Disease Progression		**Hyperedge Contribution Set**			
Progression 1	**Progression 2**	**Power Set**	**Exclusive**	**Progression (Simple)**	**Progression (Aggregate)**
$A \to B$	$A \wedge B \to C$	Full Power Set, $\mathcal{P}(\{A, B, C\})$	$\{A, B, C\}$	$\{A\}, \{A, B\}$, {A, C},{B, C}	$\{A\}, \{A, B\}$, {A, B, C}
$A \to B$	$A \wedge B \to C$	Full Power Set, $\mathcal{P}(\{A, B, C\})$	$\{A, B, C\}$	$\{A\}, \{A, B\}$, {A, C},{B, C}	$\{A\}, \{A, B\}$, {A, B, C}
$A \to B$	$A \wedge B \to C$	Full Power Set, $\mathcal{P}(\{A, B, C\})$	$\{A, B, C\}$	$\{A\}, \{A, B\}$, {A, C},{B, C}	$\{A\}, \{A, B\}$, {A, B, C}
$C \to A$	$A \wedge C \to B$	Full Power Set, $\mathcal{P}(\{A, B, C\})$	$\{A, B, C\}$	$\{C\}, \{A, C\}$, {B, C},{A, B}	$\{C\}, \{A, C\}$, {A, B, C}
$B \to C$	-	$\{B\}, \{C\}$, {B, C}	$\{B, C\}$	$\{B\}, \{B, C\}$	$\{B\}, \{B, C\}$
$B \to B$	-	$\{B\}$	$\{B\}$	$\{B\}$	$\{B\}$
$C \to C$	-	$\{C\}$	$\{C\}$	$\{C\}$	$\{C\}$
$B \to A$	$A \wedge B \to C$	Full Power Set, $\mathcal{P}(\{A, B, C\})$	$\{A, B, C\}$	$\{B\}, \{A, B\}$, {B, C}, {A, C}	$\{B\}, \{A, B\}$, {A, B, C}
$B \to A$	-	$\{B\}, \{A\}$, {A, B}	$\{A, B\}$	$\{B\}, \{A, B\}$	$\{B\}, \{A, B\}$
$A \to C$	-	$\{A\}, \{C\}$, {A, C}	$\{A, C\}$	$\{A\}, \{A, C\}$	$\{A\}, \{A, C\}$

Table 2: The set of hyperedges each individual contributes prevalence to, given a contribution type.

4.4 Formula

With regards to our application into multimorbidity and coincident diseases, the weighting scheme for disease prevalence is some normalised quantification of the number of people with all the conditions represented by the edge. This is a problem of overlapping sets. Below describes three different formulations to compute multi-set overlap.

For the remainder of this section, we will use the function $C(e_i)$ to represent the prevalence of a particular hyperedge e_i, given a contribution type discussed in the section above. Define this arbitrary hyperedge e_i connecting n disease nodes $i_1, \ldots, i_n$ such that $e_i = \{i_1, \ldots, i_n\}$. $C(e_i)$ outputs a real-valued number representing the prevalence of the disease set which the hyperedge e_i connects.

Note that the prevalence of the hyperedge e_i, $C(e_i)$ comes directly from the prevalence counter which can be calculated from different contribution types, i.e. power set ($C_{po}(e_i)$), exclusive set ($C_{ex}(e_i)$) or progression set ($C_{pr}(e_i)$). There exists an inequality relationship between all three such that

$$C_{ex}(e_i) \leq C_{pr}(e_i) \leq C_{po}(e_i). \tag{19}$$

This is because for progression-based contribution, the intersection counts anyone who had all diseases of interest sequentially as part of their progression. Individuals who had all diseases but whose condition ordering was observed to be non-consecutive are discounted from this prevalence. However, all individuals who had all diseases regardless of observed ordering are counted for the power set-based contribution.

4.4.1 Overlap coefficient

[8] used the overlap coefficient, generalised to apply to any number of sets to weight each edge. It is the prevalence of the disease set, divided by the minimum number of people with one of the diseases in that same set. We define the overlap coefficient of a hyperedge e_i, $O(e_i)$ such that

$$O(e_i) = O(\{i_1, \ldots, i_n\}) = \frac{C(e_i)}{\min\{|P_{i_1}|, \ldots, |P_{i_1}|\}} \tag{20}$$

where P_k is the population of individuals with disease k. The overlap coefficient has an intuitive interpretation for when the degree of the edge is 2 as it constitutes to measuring the extent of overlap of the individuals with the disease of smaller population with the individuals with the disease of the larger population.

However, for edges of higher degree, the overlap coefficient for the corresponding disease set measures how many of the individuals who have the disease with the smallest population also has all other diseases contained in the hyperedge. Because of this denominator term, measuring relative prevalence may have the unexpected behaviour of exaggerating overlap coefficients when the smallest disease population is significantly smaller than all other diseases in the set. As a result, the overlap coefficient may exaggerate the score for the hyperedge.

Figure 10 shows an example where we have three disease sets, but the population of disease C is significantly smaller than diseases A and B. Note that the dissimilarity of A and B is quite stark but yet the overlap coefficient only considers the size of C when measuring the relative prevalence of $\{A, B, C\}$. The coefficient would measure approximately 1/4, but this would be an overestimate given the dissimilarity A and B have with each other and C. Moreover, if the individuals with disease C are entirely subset of A and C, this would set the overlap coefficient to be 1, an egregious overestimate for the weight of the hyperedge by ignoring the dissimilarity of diseases A and B.

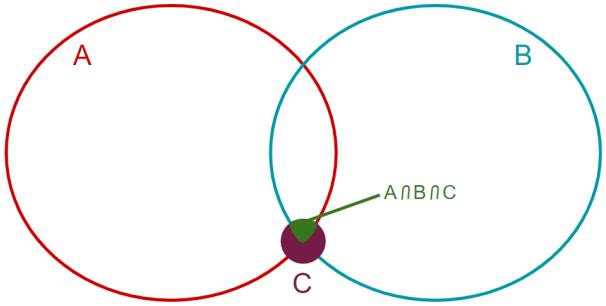

Figure 10: Simple example to demonstrate the issue of using overlap coefficient for hyperedges of degree greater than 2.

4.4.2 Modified Sørensen–Dice coefficient (power set)

Instead of the overlap coefficient, we could consider using a measure of similarity which takes into account the sets of which it measures similarity of, such as the Sørensen–Dice Coefficient. For arbitrary sets P_1 and P_2, this coefficient is

$$D(P_1, P_2) = \frac{2|P_1 \cap P_2|}{|P_1| + |P_2|}. \tag{21}$$

This weight will compute the intersection of P_1 and P_2, relative to the total population of P_1, P_2, or both. An extension to this similarity coefficient multiple sets is to set the denominator as all possible subset-intersections. For arbitrary sets $P_1, \ldots, P_m$ this coefficient would be

$$D(P_1, \ldots, P_m) = \frac{|P_1 \cap \cdots \cap P_m|}{|P_1 \cap \cdots \cap P_m| + \sum_{i=1}^{m-1} w_i |D_i|}, \tag{22}$$

where w_i is an optional weight indexed by the size of the subset and D_i represents all combinations of subset-intersections of i of m diseases. For example, when $m = 3$, $|D_2| = |P_1 \cap P_2 - P_3| + |P_1 \cap P_3 - P_2| + |P_2 \cap P_3 - P_1|$, i.e. individuals of $\{P_1, P_2, P_3\}$. Figure 11 shows three disease sets A, B and C and each portion of intersection has been annotated.

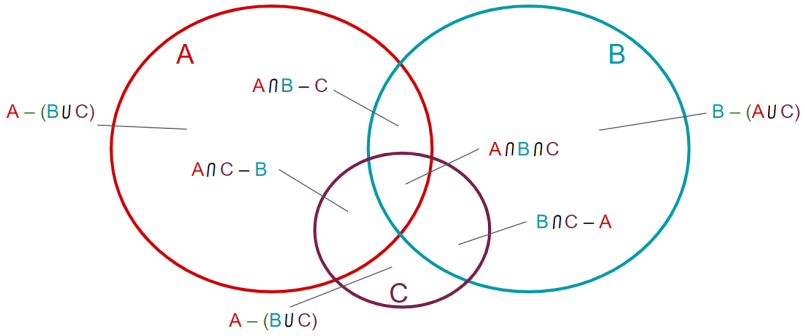

Figure 11: Three disease set scenario with portions of intersections annotated.

Using these annotations, we can define the Sørensen–Dice coefficient for disease set $\{A, B, C\}$ for example as

$$D(A, B, C) = \frac{|A \cap B \cap C|}{|A \cap B \cap C| + \sum_{i=1}^{2} w_i |D_i|}, \tag{23}$$

where

$$|D_1| = |A - (B \cup C)| + |B - (A \cup C)| + |C - (A \cup B)|, \tag{24}$$
$$|D_2| = |(A \cap B) - C| + |(A \cap C) - B| + |(B \cap C) - A|. \tag{25}$$

This formulation allows the intersection of any disease set of interest to be penalised according to all other subset-intersections, i.e. other related portions of intersections. Equation (32) is modelled after the Tversky index but for multiple sets, i.e. for $m = 2$ and $w_1 = 1/2$ we recover the Sørensen–Dice coefficient since

$$D(P_1, P_2) = \frac{|P_1 \cap P_2|}{|P_1 \cap P_2| + \frac{1}{2}|D_1|} \tag{26}$$
$$= \frac{|P_1 \cap P_2|}{|P_1 \cap P_2| + \frac{1}{2}(|P_1 - P_2| + |P_2 - P_1|)} \tag{27}$$
$$= \frac{2|P_1 \cap P_2|}{2|P_1 \cap P_2| + |P_1 - P_2| + |P_2 - P_1|} \tag{28}$$
$$= \frac{2|P_1 \cap P_2|}{|P_1| + |P_2|}, \tag{29}$$

which matches equation (21) above. Note that $|P_1 - P_2|$ is the notation for the number of individuals who have disease 1 but not disease 2, and likewise for $|P_2 - P_1|$.

With respect to measuring relative prevalence where we have a hyperedge $e_i = \{i_1, \ldots, i_n\}$ then we penalise the prevalence $C(e_i)$ by all other prevalence's of similar disease subsets such that

$$D(e_i) = \frac{C(e_i)}{C(e_i) + \sum\limits_{e_j \in \mathcal{P}(e_i)} w_j C(e_j)}, \tag{30}$$

where

$$\mathcal{P}(e_i) = \{e_j \,:\, e_j \subset e_i\}. \tag{31}$$

The denominator constitutes a weighted sum of prevalence's from hyperedges which connect a subset of disease nodes of hyperedge e_i. This has the effect of measuring the prevalence of the hyperedge e_i relative to all subset disease sets. Note that w_i provides a flexible weighting scheme to weight different subset-intersections based on domain knowledge. For simplicity, we have set $w_i = 1$ for all i. This has the effect of weighting the prevalence of the disease set of interest, $\{i_1, \ldots, i_n\}$ by all other prevalence's of disease subsets of disease nodes $\{i_1, \ldots, i_n\}$ *equally*.

We could choose a variable weighting such that w_i reduces as i increases. This means the prevalence of $\{1, \ldots, m\}$ is penalised most by the prevalence of single-set diseases $\{1\}, \{2\}, \ldots, \{m\}$, and least by the prevalence of all $m-1$-subset prevalence's, $\{1, \ldots, m-1\}$, etc. This might make sense since those contributions for disease sets containing $m-1$ of m diseases are from individuals more similar in disease diagnoses than those contributions for single disease sets, i.e. disease sets with only 1 of m diseases. The specific weighting in mind could as simple as $1/i+1$ or $1/2^i$, to ensure that when $m = 2$ we recover the original formula for the Sørensen–Dice Coefficient.

4.4.3 Modified Sørensen–Dice coefficient (complete set)

This is an extension of the formula above for the modified Sørensen-Dice coefficient. This formulation constructs the divisor of each relative prevalence as not only the subset-intersections of the disease set of interest, but also all disease sets which are super sets of the disease set of interest, i.e. all those disease sets of which the disease set in question is a subset of. We compute this as

$$W(e_i) = \frac{C(e_i)}{C(e_i) + \sum\limits_{e_j \in \mathcal{P}(e_i)} w_j C(e_j) + \sum\limits_{e_k \in \mathcal{S}(e_i)} w_k C(e_k)}, \tag{32}$$

where

$$\mathcal{S}(e_i) = \{e_k \,:\, e_i \subset e_k\}. \tag{33}$$

The main difference here is that we now penalise the prevalence of a disease set based on the prevalence of *all* similar disease sets. That is, disease sets which contain the disease set of interest as well as those which are subsets of the disease set of interest.

The rationale behind this version is the following. The former, power set-based version measures multi-set similarity of disease set $\Phi = \{A, B, C\}$ by penalising the prevalence contribution of all three diseases by a weighted sum of the prevalence's of each power-set element $\phi \subset \Phi$. This will penalise the prevalence of Φ by all similar subset disease sets. However, if we know we have many more diseases in our data, say $\Theta = \{D, E, F\}$, then there will exist individuals who not only have all three diseases of interest, but may also have more diseases from Θ, i.e. individual $I \in \Phi \cup \theta$, for $\theta \subset \Theta$. Here, it might make sense to take these individuals into account, noting that they not only have all diseases of interest, but are dissimilar to the prevalence of Φ and should therefore contribute to the penalisation in the denominator.

The Overlap Coefficient penalises the intersection using the smallest possible number of individuals available, while the *complete set* version of the modified Sørensen-Dice coefficient computes the *largest* possible number of individuals as the denominator. This allows every hyperedge's prevalence to be weighted relative to all similar disease set prevalence's that contain the disease set or a subset of it, providing an estimate for how pervasive a disease set it among the population, compared to other, similar disease sets.

The weighting w_j and w_k again provide flexibility in choosing how to weight each similar disease set's penalising prevalence. There are many assumptions that could be made here. For example, we could define w_j, w_k based on the hyperedge degree, promoting symmetry for disease sets in the denominator which are subsets or super sets of the disease set of interest. This would weight those disease sets closer in degree to the disease set of interest less harshly, implying less dissimilarity to the disease set of interest. Here, the dissimilarity is based crudely on the number of diseases in common the disease sets in the denominator have with the number of diseases of interest.

Or, given some domain knowledge we could perhaps conclude that weights w_k for those disease super sets should be smaller as those prevalence's should contribute less in the penalisation term because super set disease sets are more similar to the disease set of interest than any subset due to how the diseases manifest themselves and interact with each other.

The suggestions above ignores any interaction effect that different combinations of diseases can have in the body, which likely doesn't make sense from a clinical perspective. However, there are multiple ways in which the values for w_j, w_k could be chosen. At present, we will assume a neutral, unitary weighting of $w_j = w_k = 1$ so as not to make any naive, or presumptuous conclusions on the similarity disease sets in the denominator have with the disease set in question, especially given how at present we have ignored the specific diseases comprising each set.

This formulation for the complete set variation of the modified Sørensen-Dice coefficient can still recover the Sørensen-Dice coefficient for the two-case setting, i.e. when there are only diseases in the dataset, therefore the hyperedge weight for edge $e_i = \{i_1, i_2\}$ would have no members in the super set and so only consider prevalence's from $\mathcal{P}(e_i)$, which are just the hyperedges $\{i_1\}$ and $\{i_2\}$. Assuming $w_1 = 1/2$ we recover the original Sørensen-Dice coefficient. Of course, when $n > m = 2$ then this variation of the will be different to the setup above where $n = m = 2$. This is because we would now penalise the coefficient further by taking into account prevalence's of disease sets which are the super set of the disease set of interest.

4.5 Model setup

In summary, our weight system is as follows:

- **Outcome**-dependence: we use multimorbidity prevalence as our outcome of interest.
- **Contribution**-dependence: we use progression-based contribution to model observed patient disease progressions. Within this contribution type, we experiment with aggregate progression-, power set progression- and simple progression-based contribution.
- **Formula**-dependence: we use the complete set-based version of the modified Sørensen-Dice coefficient.

5 Directed hypergraphs

5.1 Notation

A directed hypergraph $\mathcal{H}_D(V, \mathcal{E})$ is a collection of nodes stored in $V = \{v_1, \ldots, v_n\}$, and *directed* hyperedges, or *hyperarcs*, $\mathcal{E} = \{e_1, \ldots, e_m\}$. Each hyperarc $e_i = \langle T(e_i), H(e_i) \rangle$ where $T(e_i), H(e_i) \subseteq V$ represents a collection of nodes such that those $v_i \in T(e_i)$ are the tails of the hyperarc and those $v_j \in H(e_i)$ are the heads of the hyperarc.

There are three main kinds of hyperarcs you can consider, but are all built from the assumption that there requires at least one tail node and at least one head node. The three kinds of hyperarcs are B-hyperarcs, F-hyperarcs and BF-hyperarcs. A B-hyperarc is where we restrict $|H(e_i)| = 1$ for all hyperarcs e_i so that the directed hyperedge transitions from a set of *tail* nodes in $T(e_i)$ to a single *head* node in $H(e_i)$. An F-hyperarc is the opposite where $|T(e_i)| = 1$ so that the directed hyperedge transitions from a single *tail* node to all remaining *head* nodes. A BF-hyperarc is where there is no restriction on the size of $T(e_i)$ and $H(e_i)$. Figure 12 shows a 3-node setup where we have the fully connected directed graph on the left and two directed hypergraphs on the right, one only using B-hyperarcs and the other using F-hyperarcs.

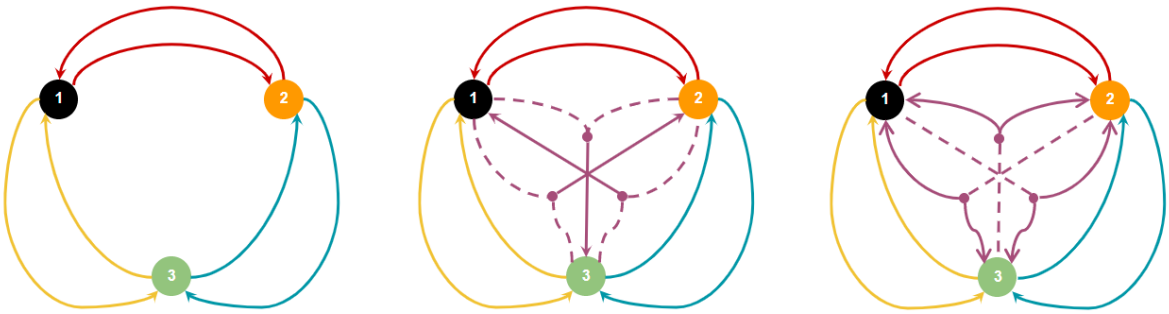

Figure 12: (Left) 3-node directed graph. (Middle) 3-node directed hypergraph of B-hyperarcs. (Right) 3-node directed hypergraph of F-hyperarcs

Note that 2-degree hyperedges are by definition always B-hyperarcs, F-hyperarcs and BF-hyperarcs. Moreover, in a 3-node setup, it is not possible to show an explicit BF-hyperarc as we require $|T(e_i)|, |H(e_i)| \geq 2$. Much like in the directed graph case discussed in 2.2.2, undirected, parent hyperedges give rise to directed, children hyperedges called hyperarcs. Allowing all B-, F- and BF-hyperarcs to be generated from a single undirected k-degree hyperedges means there are

$$\sum_{j=1}^{k-1} \binom{k}{j} \tag{34}$$

possible hyperarcs to consider. This is because there are $\binom{k}{j}$ ways we can build a hyperarc e_i such that $|T(e_i)| = j$ for $j = 1, \ldots, k-1$. Note here we exclude hyperarcs where $|T(e_i)| = k$ or $|H(e_i)| = k$.

5.2 Model assumptions

Although our dataset provides date-time instances where diseases were first observed by healthcare practitioners on distinct dates, these are only the first recorded date where a health practitioner observed the chronic condition within the individual during an interaction with the healthcare system. This is very different to the date at which the individual first developed the condition, which is not known to us. Therefore, when we discuss an individual's disease progression,

this is actually the *observed* disease progression seen by health practitioners and we only use the difference in date, not time, to determine the ordering of conditions an individual has.

When building our multimorbidity directed hypergraph, we will only consider B-hyperarcs, i.e. those seen in the middle diagram of figure 12 where all hyperarcs e_i have a solitary head sets, $|H(e_i)| = 1$. This is because we define progression to be the transition from one multimorbidity disease state to another, requiring an additional n^{th} onset condition to be observed within an individual's current $n-1$ multimorbidity disease state. Therefore, it makes sense that the directionality of hyperedge represents the $n-1$ previously observed disease nodes acting as tail nodes, flowing toward the n^{th} onset condition, acting as the head node. This means an individual will contribute to a hyperarc, given

1. They exhibit all conditions the hyperarc connects at some point in their progression;
2. They have progressed from their previously observed disease state exhibiting $n-1$ of those conditions connected by the hyperarc;
3. The hyperarc's tail set contains those previous $n-1$ conditions, and it's head set contains the n^{th} onset condition.

This means that the number of possible hyperarcs for a directed hypergraph is

$$|\mathcal{E}| \leq n + \sum_{k=2}^{n} \binom{n}{k}\binom{k}{1} \tag{35}$$

$$= n + \sum_{k=2}^{n} k\binom{n}{k} \quad \text{since } \binom{k}{1} = k. \tag{36}$$

Note that we increment the possible number of hyperarcs by the number of nodes n in the directed hypergraph because we must account for self-looped hyperarcs, i.e. $e_i = \langle T(e_i), H(e_i)\rangle$ where $T(e_i) = H(e_i) = \{v_i\}$. In the context of multimorbidity, these hyperarcs represent individuals who only have one disease throughout the entire period of analysis.

To put equation (35) into perspective, the Charlson comorbidity index contains 16 diseases, but grouped into 13 to remove pseudo-clustering. A directed graph with 13 nodes contains a maximum of 156 directed edges, a directed hypergraph with 13 nodes contains a maximum of 53,248 possible hyperarcs, as opposed to 8,178 hyperedges in an undirected hypergraph.

Due to the caveats of the data, it may be that an individual's first recorded date where a health practitioner has observed a particular condition may be the same for other conditions the individual has been noted to have. This reduces our confidence in what their multimorbidity progression was (and what their specific progression set they contribute to) as we have no empirical means to track their disease progression.

As described in [8], there were 2,178,938 people in the cohort, who were diagnosed with 2,918,569 conditions (this includes people who had no record under the Charlson index). 1,313,219 (approx. 60%) individuals had at least one condition and 755,421 (approx. 35%) had more than one conditions diagnosed. Around 30% of these individuals had a 'clean' progression of multimorbidity, i.e. their records when interacting with the healthcare system show that their conditions were first observed on distinct observation dates by healthcare practitioners. Approximately 120,000 (approx. 5%) did not, i.e. these individuals had multiple conditions first observed by healthcare practitioners at the same time, therefore making their observed patient disease progression difficult to assess. This is an important caveat when constructing the weights for the hyperarcs and an individual's corresponding contribution.

5.3 Weighting hyperarcs

Note that this section is for building the weights for a directed hypergraph and therefore we will be using the aggregate-progression contribution type throughout this section.

5.3.1 Contribution

At this point we introduce the notation for a hyperarc as

$$D_1 \wedge \cdots \wedge D_{n-1} \to D_n \quad \text{or} \quad D_1, \ldots, D_{n-1} \to D_n. \tag{37}$$

This notation represents the hyperarc $h_i = \langle T(h_i), H(h_i)\rangle$ such that

$$T(h_i) = \{D_1, \ldots, D_{n-1}\} \quad H(h_i) = \{D_n\}. \tag{38}$$

Note that this hyperarc is one of n possible B-hyperarcs generated from the n-degree, undirected hyperedge $\{D_1, \ldots, D_n\}$. For the majority of our individuals, we will have distinct dates where conditions have been observed by health practitioners. For example, given an individual whose final, recorded multimorbidity disease set was $\{A, B, C, D\}$. Let their disease progression be $A \to A, B \to A, B, C \to A, B, C, D$. Then given a contribution type discussed in section 4.3, they will contribute prevalence to the hyperarcs which belong to their *progression set*. The progression set are the set of hyperarcs of which the individual traverses during their observed disease progression. Given an aggregate progression contribution, this individual's progression set is

$$\Big\{\{A \to B\}, \{A \wedge B \to C\}, \{A \wedge B \wedge C \to D\}\Big\}. \tag{39}$$

This is in contrast to the hyperedges they contribute prevalence to, as each hyperarc above is an explicit child of their corresponding parent hyperedge. Say the individual was observed to have disease B before A, then they would contribute prevalence at the degree 2 stage to hyperarc $B \to A$. In both cases, the individual contributes prevalence to the *hyperedge* $\{A, B\}$, but only one of its children $A \to B$ or $B \to A$. However, we ignore the order of conditions once they are in the tail set, i.e. both cases would still contribute prevalence to the hyperarc $A \wedge B \to C$ as this is the same hyperarc as $B \wedge A \to C$. Note that for an individual with only one condition A, the progression set they contribute prevalence to is simply $\Big\{\{A \to A\}\Big\}$.

Note from (41), each element of the progression set is a single-set each containing a different degree hyperarc, with each consecutive set element containing a hyperarc of incremental degree to its preceding set element. For weighting hyperarcs due to prevalence, an individual contributes itself wholly to each subset of the progression set. The progression set is formed this way because within each subset there could potentially be multiple hyperarcs to consider if the individual's records show multiple conditions first observed at the same time. Any instance of duplication requires the individual to contribute equally to all members of each subset of their progression set.

Within the 5% of individuals who have multiple conditions first observed at the same time, their progression set must include all potential progressions at the point in their progression where these first-observation dates were the same. We define an individual with a single n-duplicate to have n conditions observed for the first time during a single interaction with the healthcare system. We have divided these instances into 1-duplicates, 2-duplicates and $n \geq 3$-duplicates, for $n \geq 2$:

1. 1-duplicates: These are individuals with clean progressions, i.e. for each distinct date-time instance, i.e. interaction with the healthcare system, a single condition was observed for the first time. The progression set is unchanged from above.
2. 2-duplicates: These are individuals whose observed disease progression has a date-time duplication such that 2 conditions were observed at the same time. An individual can have several 2-duplicates, as long as each date-time instance for each duplicate is different. Otherwise, they are $n \geq 3$-duplicates below.
3. $n \geq 3$-duplicates: These are individuals whose observed disease progression contains duplicate date-time instances for multiple conditions, i.e. more than 2 conditions.

The categorisation above is necessary as we will only account for individuals presenting with 1- and/or 2-duplicates in our directed hypergraph model. For individuals with 2-duplicates, only one or two additional hyperarcs are required to model the uncertainty in two conditions being observed at the same time. The number of extra hyperarcs are dependant on where in their disease progression this single duplication occurred.

1. If the duplicate is at the beginning of an individual's disease progression, this only adds one more hyperarc (of degree 2).
2. If the duplicate is found anywhere after the beginning there will be two more hyperarcs, one hyperarc of degree equal to the location in the disease progression where the duplicate was found and another for the degree above.

For case 1, consider the individual at the beginning of this section presenting with four conditions $\{A, B, C, D\}$. If there was a 2-duplicate for diseases A and B then their progression set would be

$$\Big\{\{A \to B,\ B \to A\}, \{A \wedge B \to C\}, \{A \wedge B \wedge C \to D\}\Big\}, \tag{40}$$

to take into account that disease B may have come before A or vice versa. Therefore, this individual would contribute $1/2$ to each hyperarc of degree 2 and 1 to every other hyperarc part of their progression set. For case 2, consider the example individual from figure 9. This individual presents with 3 conditions whose date-time instance show diseases B

and C has having a 2-duplicate. In this case, the progression set would be

$$\Big\{\{A \to B,\ A \to C\}, \{A \wedge B \to C,\ A \wedge C \to B\}\Big\}. \tag{41}$$

When we consider individuals with $n \geq 3$-duplicates, the progression set can become much more complicated, requiring consideration of all permutations of hyperarcs which permute all possible progressions of those diseases first observed on the same date. As a result, many more hyperarcs are required to take into account the uncertainty of ordering for the conditions observed at the same time.

The concern here is that for $n \geq 3$-duplicates, the individual's trajectory becomes less clear (and their contribution becomes less significant) for those hyperarc degrees where we have to consider many different cases. When we start looking at $n \geq 3$ where an individual may have 4 or more conditions first observed at the same time it may not be worth including these individuals in our directional analyses due to the large progression sets they will be contributing to, potentially exaggerating some trajectories which are known in the domain to be unlikely.

Based on the work in this section, it is possible to include those individuals whose observed disease progression comprises of 1- or 2-duplicates. Within the 5% individuals with duplicates, 107,000 individuals (approx. 87%) exhibit these kinds of duplicates. Therefore, of the original 755,521 individuals used for the undirected hypergraph analysis 98% are suitable for building the directed hypergraph. The remaining individuals with $n \geq 3$-duplicates have been excluded.

5.3.2 Formula

For the weighting system of hyperarcs according to prevalence, there should be consideration of both the prevalence of the hyperarc itself, i.e. the progression it represents, as well as the hyperedge to which it is a child of. Therefore, we could consider weighting a hyperarc by weighting its prevalence among other children of its parenting hyperedge, weighted by the relative prevalence of the hyperedge itself.

Consider the directed hypergraph $\mathcal{H}_{\mathcal{D}}(V, \mathcal{E})$ and the undirected hypergraph $\mathcal{H}(V, P(\mathcal{E}))$ with the same node set V, but where $P(\mathcal{E})$ is the set of parent hyperedges to which all hyperarcs in $\mathcal{E}$ are generated from. Let $p(h_i) \in P(\mathcal{E})$ be the parent hyperedge to which the child hyperarc h_i is generated from and

$$\mathcal{K}(h_i) = \{h_j \ :\ p(h_j) = p(h_i)\} \tag{42}$$

be the set of all B-hyperarc children of the parent hyperedge $p(h_i)$. Then we define the weight for hyperarc h_i as

$$w(h_i) = Wpro(p(h_i)) \frac{C(h_i)}{\sum_{h_j \in \mathcal{K}(h_i)} C(h_j)}. \tag{43}$$

Note that $W(\cdot)$ represents a *hyperedge* weight while $w(\cdot)$ represents a *hyperarc* weight. $W(p(h_i))$ is the weight of the parent hyperedge in the undirected hypergraph model. This could be for example any of the weight functions discussed in section 4.4. This formula means that for any hyperarc h_i, we have that

$$W(p(h_i)) = \sum_{h_j \in \mathcal{K}(h_i)} w(h_j). \tag{44}$$

Therefore, each child hyperarc of the same parent hyperedge takes some proportion of the parent hyperedge weight based on the number of contribution each child has.

This weighting scheme measures how prevalent the hyperarc h_i is among other children hyperarcs of the same parent, relative to how prevalent that parent hyperedge is among the population. That is, it measures how prevalent an observed disease progression is among other disease progressions in the same multimorbidity set, relative to the prevalence of the multimorbidity set within the population.

5.3.3 Example dataset

Figure 13 shows the hyperedge and hyperarc weights computed for the example dataset specified in section 4.3.4. The shaded bars represent the hyperedge weights which are indexed on the right-hand y-axis, while the hyperarc weights are the solid bars indexed on the left-hand $y - axis$. Note that each hyperarc's weight is superimposed onto its corresponding parent hyperedge weight to visualise how each hyperarc weight is computed from its parent hyperedge weights. Some observations for the hyperedge weights are:

- $W(\{A, B\}) > W(\{A, B, C\})$ because although there are an equal number of contributions for both hyperedges, $W(\{A, B, C\})$ is now penalised by all other disease sets containing C, unlike $W(\{A, B\})$.

- $W(\{B\}) > W(\{A\}) > W(\{C\})$ because there were only 2 individuals starting their progression with disease C, and 4 starting with either disease A or B. Moreover, $W(\{B\}) > w(\{A\})$ as there are more progressions observed including disease A than there are disease B, i.e. $A \to C$ and $C \to A$.
- $W(\{c\})$ is large even though fewer individuals started their progression at disease C compared to diseases A and B. This is because there are fewer progressions including disease C than there are for diseases A and B.

Some observations for the hyperarc weights are:

- $w(A, B \to C) > w(A, C \to B)$ because there are 4 times more contributions to the progression $A, B \to C$ than there are to $A, C \to B$. In fact, $w(A, B \to C) = 4w(A, C \to B)$ and $W(\{A, B, C\}) = w(A, B \to C) + w(A, C \to B)$.
- $w(A \to C) = w(c \to A)$ as there are an equal number of contributions to both progressions, so they both take half the total hyperedge weight $W(\{A, C\})$.
- $w(B \to C) = W(\{B, C\})$ because nobody has the progression $C \to B$.
- $w(A \to A) = 0$ because nobody *only* had disease A — all individuals who started their disease progression with disease A moved onto to develop other conditions.
- $w(B \to B) = {}^1\!/_4 W(\{B\})$ as only 1 individual who started with disease B did not develop anymore conditions.

Observed Disease Progression

Progression 1	Progression 2
$A \to B$	$A \wedge B \to C$
$A \to B$	$A \wedge B \to C$
$A \to B$	$A \wedge B \to C$
$C \to A$	$A \wedge C \to B$
$B \to C$	-
$B \to B$	-
$C \to C$	-
$B \to A$	$A \wedge B \to C$
$B \to A$	-
$A \to C$	-

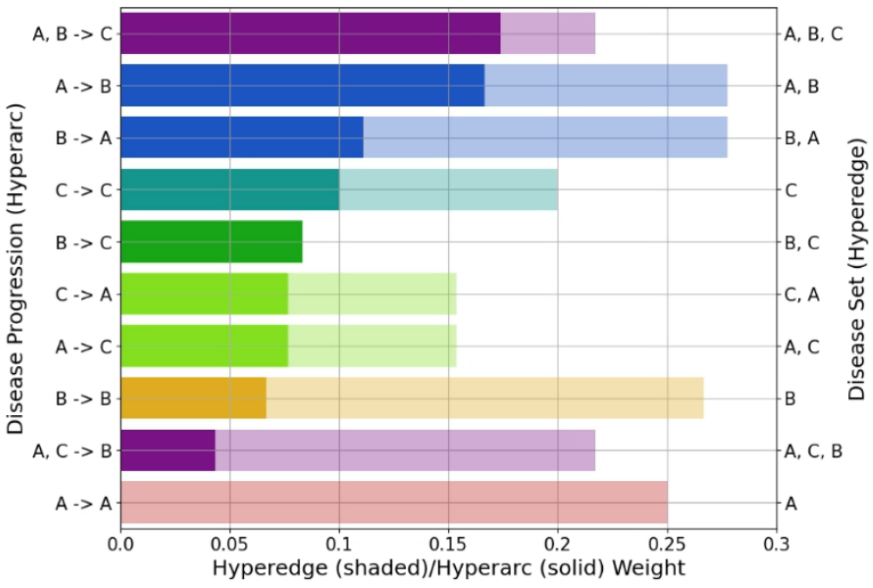

Figure 13: (Left) Progression of each individual from the example dataset from section 4.3.4. (Right) Hyperedge and hyperarc weights computed using multimorbidity prevalence, aggregate progression contribution and the complete set-modified Sørensen–Dice coefficient.

5.4 Undirected representation of the directed hypergraph

5.4.1 Dual directed hypergraph

Much like in the undirected hypergraph, we can represent the directed hypergraph mathematically using it's corresponding incidence matrix. The incidence matrix M_D for all B-hyperarcs for a 3-node directed hypergraph $H_D(V, \mathcal{E})$, $V = \{D_1, D_2, D_3\}$ is defined such that when $M_D(i, j) = -1$, $D_j \in T(h_i)$, while when $M_D(i, j) = 1$, $D_j \in H(h_i)$. The matrix M_D can be seen below. Note that the first three hyperarcs are the self-looping ones, which cannot be explicitly represented using this representation as they are both the tail and head node of a hyperarc. Therefore, we split the incidence matrix into its tail and head components M_D^- and M_D^+.

$$M_D = \begin{array}{l} \\ D_1 \to D_1 \\ D_2 \to D_2 \\ D_3 \to D_3 \\ D_1 \to D_2 \\ D_2 \to D_1 \\ D_1 \to D_3 \\ D_3 \to D_1 \\ D_2 \to D_3 \\ D_3 \to D_2 \\ D_1 \wedge D_2 \to D_3 \\ D_1 \wedge D_3 \to D_2 \\ D_2 \wedge D_3 \to D_1 \end{array} \begin{array}{c} \begin{array}{ccc} D_1 & D_2 & D_3 \end{array} \\ \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \\ -1 & 1 & 0 \\ 1 & -1 & 0 \\ -1 & 0 & 1 \\ 1 & 0 & -1 \\ 0 & -1 & 1 \\ 0 & 1 & -1 \\ -1 & -1 & 1 \\ -1 & 1 & -1 \\ 1 & -1 & -1 \end{pmatrix} \end{array} \implies M_D^- = \begin{array}{c} \begin{array}{ccc} D_1 & D_2 & D_3 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix} \end{array} \quad \text{and} \quad M_D^+ = \begin{array}{c} \begin{array}{ccc} D_1 & D_2 & D_3 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{pmatrix} \end{array} \tag{45}$$

In section 3.3 we considered the dual undirected hypergraph, where the roles of the nodes and hyperedges swapped so that we can analyse the multimorbidity disease sets. This was made possible because taking the transpose of the undirected hypergraph incidence matrix results in another undirected hypergraph. This symmetry is not maintained for our model specification for the directed hypergraph and instead relies on the hyperarcs present in the original directed hypergraph. In general, it is possible that the dual directed hypergraph could exhibit dual hyperarcs h_i where $T(h_i) = \emptyset$, $H(h_i) = \emptyset$ or $|H(h_i)| > 1$.

Because of our assumptive use B-hyperarcs, the rules built for processing the directed hypergraph break down for processing the dual directed hypergraph. For example, the transpose of the matrix M_D above results in dual BF-hyperarcs, with arbitrary tail and head node sets. In the most general case where is no restriction on the nodes that inhabit $T(h_i), H(h_i)$ for the directed hypergraph, the same is true for its corresponding dual directed hypergraph, and so symmetry is upheld. This is not the case here.

However, the ability to analyse multimorbidity disease progressions is one of many advantage of building a directed hypergraph compared to its directed graph model equivalent. As a work-around, we can consider the following set-up as a stepping stone to enable us to utilise the dual representation so that we can perform downstream analysis on the multimorbidity disease progressions. We will consider an undirected hypergraph whose hyperedges are interpreted as the observed disease progressions from a directed hypergraph and whose node set is an expanded version of the original node set of the directed hypergraph.

5.4.2 Notation

Let $\mathcal{H}_D(V, \mathcal{E})$ be a directed hypergraph as before, whose node set $V = \{D_1, \ldots, D_n\}$ and whose edge set represents all observed disease progressions, or hyperarcs, $e = \langle T(e), H(e) \rangle \in \mathcal{E}$, where $T(e), H(e) \subseteq \mathcal{E}$. Let the undirected hypergraph $\mathcal{H}(\mathcal{V}, \mathcal{E})$ be the *undirected* representation of $\mathcal{H}_D$ such that it has the same edge set, but $\mathcal{V}$ is defined as

$$\begin{aligned} \mathcal{V} &= V^- \cup V^+ && (46) \\ &= \{D_1^-, D_2^-, \ldots, D_n^-\} \cup \{D_1^+, D_2^+, \ldots, D_n^+, \} && (47) \\ &= \{D_1^-, D_1^+, D_2^-, D_2^+, \ldots, D_n^-, D_n^+, \}, && (48) \end{aligned}$$

where D_i^- is the tail-only node component of disease i and D_i^+ is the head-only node component. Therefore, each original node D_i is now decomposed into its tail- and head-only components. Hence, $|\mathcal{V}| = 2|V|$. This means each hyperarc $e \in \mathcal{E}$ is now decomposed using $\mathcal{V}$ such that $e = \langle T(e), H(e) \rangle$, $T(e) \subset V^-$ and $H(e) \subset V^+$. Note that although each edge is technically an undirected hyperedge, they actually represent the hyperarcs from the directed hypergraph, allowing us to use the same rules for analysing undirected hypergraphs and, in particular, leveraging the dual representation without violating any rules or symmetry like we do in the directed case. Constructing the incidence matrix for this undirected hypergraph is simply concatenating matrices M_D^- and M_D^+ from equation (45) column-wise such that

$$M = \begin{array}{l} \\ D_1 \to D_1 \\ D_2 \to D_2 \\ D_3 \to D_3 \\ D_1 \to D_2 \\ D_2 \to D_1 \\ D_1 \to D_3 \\ D_3 \to D_1 \\ D_2 \to D_3 \\ D_3 \to D_2 \\ D_1 \wedge D_2 \to D_3 \\ D_1 \wedge D_3 \to D_2 \\ D_2 \wedge D_3 \to D_1 \end{array} \begin{array}{c} \begin{array}{cccccc} D_1^- & D_2^- & D_3^- & D_1^+ & D_2^+ & D_3^+ \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 \end{pmatrix} \end{array}, \tag{49}$$

5.4.3 Weights and dual representation

The edge weights here are defined in the same manner as for the directed hypergraph, as seen in equation (43) from section 5.3.2. This will weight a hyperarc not only according to the multimorbidity prevalence of its parent undirected hyperedge, but also the prevalence of the child among its siblings. Therefore, natural suppression can occur for children with low prevalence relative to their siblings.

The dual representation for this undirected hypergraph is $\mathcal{H}^*(\mathcal{E}, \mathcal{V})$, where we have swapped the roles of the nodes and hyperarcs from our original set up with $\mathcal{H}(\mathcal{V}, \mathcal{E})$. What this amounts to is taking the transpose of the incidence matrix and considering M^{T}, where now the nodes become the hyperedges and vice versa. We now have hyperedges which represent the tail- and head-components of each disease, i.e the hyperedge D_i^+ represent all those observed disease progressions where D_i^+ was observed in its head.

In order to utilise the dual representation and construct the adjacency matrix as discussed in section 3.3, much like how we defined edge weights for the standard representation, we now need to define what the node weights in $\mathcal{V}$ are. Let X be a list of each individual's final multimorbidity hyperarc, i.e. the j^{th} element X_j represents the j^{th} individual's final multimorbidity disease progression $X_j = \langle T(X_j), H(X_j) \rangle$ where $T(X_j) = \{D_{j_1}, \ldots, D_{j_{m-1}}\}$ and $H(X_j) = \{D_{j_m}\}$, i.e. individual j has m conditions, of which D_{j_m} is the last one. Define the contribution to node weights $w^*(D_i^-), w^*(D_i^+)$ as $C(D_i^-)$ and $C(D_i^+)$ by

$$C(D_i^-) = \sum_j 1 - I(X_j, D_i) \quad \text{and} \quad C(D_i^+) = \sum_j \big(|X_j| - 1\big) I(X_j, D_i), \tag{50}$$

where $I(X_j, D_i)$ is the indicator functions defined as

$$I(X_j, D_i) = \begin{cases} 1 & D_i \in H(X_j) \\ 0 & \text{otherwise.} \end{cases} \tag{51}$$

Individuals contribute to $C(D_i^+)$ if in their final multimorbidity set, disease D_i was in the head set — otherwise, they contribute unitary prevalence to $C(D_i^-)$. Given a nonzero contribution to $C(D_i^+)$ from an individual, this contribution is scaled by the number of conditions each individual had in their tail set to balance the magnitude of tail- and head-component contributions. This scaling is introduced because $|T(e)| \geq |H(e)| = 1$ for a B-hyperarc, meaning that without this scaling, $C(D_i^-) > C(D_i^+)$ in general.

Note that if an individual only has 1 disease D_i, i.e. they contribute to the weight for hyperarc $D_i \to D_i$, then they contribute a unitary amount each to $C(D_i^-)$ and $C(D_i^+)$. Also, if an individual had 2-duplicate at the end of their progression, i.e. diseases D_i, D_j were first observed on the same date, then the individual contributes $1/2$ to the node weights $C(D_i^-), C(D_i^+)$ and $C(D_j^-), C(D_j^+)$ to take into account the uncertainty on whether disease D_i came before D_j or vice versa.

These contributions are combined to define the node weights $w^*(D_i^-)$ and $w^*(D_i^+)$ as

$$w^*(D_i^-) = \frac{C(D_i^-)}{C(D_i^-) + C(D_i^+)} \quad \text{and} \quad w^*(D_i^+) = 1 - w^*(D_i^-). \tag{52}$$

Therefore, the node weights $w^*(D_i^-)$ and $w^*(D_i^+)$ provide an estimate for how prevalent disease D_i is during an arbitrary disease progression ($w^*(D_i^-)$) or at the end of an arbitrary disease progression ($w^*(D_i^+)$).

Figure 14 shows the node weights for the example dataset from section 4.3.4. Note how $w^*(C^+) > w^*(C^-)$ while $w^*(B^+) < w^*(B^-)$ and $w^*(A^-) < w^*(A^-)$ because most individuals' final multimorbidity disease progression have diseases A or B in the tail or disease C in the head.

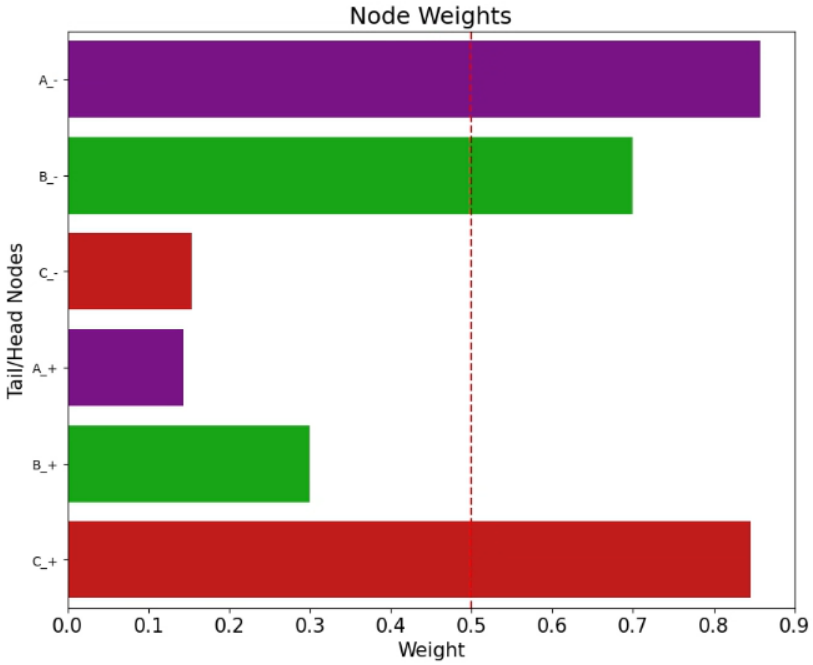

Figure 14: Node weights for tail- and head-nodes for example dataset from section 4.3.4.

5.4.4 Adjacency matrix

The adjacency matrix is such that it stores the strength of connection between pairwise nodes within the hypergraph structure. Given our undirected representation of the directed hypergraph $\mathcal{H}(\mathcal{V}, \mathcal{E})$, where $|\mathcal{V}| = n$ and $|\mathcal{E}| = m$, let us first define the formulas for computing the node and edge degree functions as

$$d(v) = \sum_{e \in \mathcal{E}} w(e) M(v, e) \tag{53}$$

$$\delta(e) = \sum_{v \in \mathcal{V}} w^*(v) M(v, e). \tag{54}$$

Let D_v be the $n \times n$ diagonal matrix whose i^{th} element contains the degree of node $v_i \in \mathcal{V}$, $d(v_i)$. Similarly, let D_e be the $m \times m$ diagonal matrix whose j^{th} element is the degree of edge e_j, $\delta(e_j)$. Note that this is the first instance where $\delta(e)$ is being weighted by the node weights $w^*(v)$. This is because of how we define the adjacency matrix for the dual representation below.

Let W_e define the $m \times m$ diagonal matrix storing the edge weights, i.e. the i^{th} diagonal element of W_e is $w(h_i)$, for $h_i \in \mathcal{E}$. Similarly, let W_v define the $n \times n$ diagonal matrix storing the node weights, i.e. the j^{th} diagonal element of W_v is $w^*(v_j)$.

We can now go ahead and define the adjacency matrix for the standard representation A and also for the dual representation A^*

$$A = M^{\mathrm{T}} W_e M - D_v \tag{55}$$

$$A^* = \sqrt{W_e}\big(M W_v M^{\mathrm{T}} - D_e\big)\sqrt{W_e}, \tag{56}$$

where $\sqrt{W_e} = \text{diag}\big(\sqrt{w(e_1)}, \ldots, \sqrt{w(e_m)}\big)$. Note that A^* is the weighted resultant adjacency for the dual representation so that A^* takes into account the prevalence-weighted observed multimorbidity disease progressions as well as the prevalence-weighted tail- and head-only components of each disease node.

5.5 Directed hypergraphs and mortality

Everything discussed in this section has focused on modelling observed disease progressions. What we have chosen to ignore up until now is that some progressions are more likely to end in death while others are not, or at the very least risk the possibility of accumulating more conditions. This section considers introducing finality into observed disease progression through mortality. Without mortality, there is difficulty deciding which observed disease progressions are stepping stones to others or which lead to death.

Table 3 outlines 6 potential ways of including mortality into the directed hypergraph, using two example individuals I and J. Individuals I has an ordered disease set of $\{A, B, C, D\}$ while individual J has ordered disease set $\{A, B\}$. We also further assume that individual I died before the end of the period of analyses while individual J was still alive. Note that n represents the number of total diseases in the hypergraph.

Mortality Type	Description	Extra Hyperarc		Additional Hyperarcs
		Individual I	Individual J	
0	1 dead nodes M 0 alive nodes	$A \wedge B \wedge C \wedge D \to M$	No extra hyperarc as J is still at risk of continuing progression.	$\sum_{d=1}^{n} \binom{n}{d}$
1	1 dead node M 1 alive node S	$A \wedge B \wedge C \wedge D \to M$	$A \wedge B \to S$	$\sum_{d=1}^{n} 2\binom{n}{d}$
2	n dead nodes $M_i, i = A, B, \ldots$ 0 alive nodes	$A \wedge B \wedge C \wedge D \to M_D$	No extra hyperarc as J is still at risk of continuing progression.	$\sum_{d=1}^{n} d\binom{n}{d}$
3	n dead nodes $M_i, i = A, B, \ldots$ 1 alive node S	$A \wedge B \wedge C \wedge D \to M_D$	$A \wedge B \to S$	$\sum_{d=1}^{n} (d+1)\binom{n}{d}$
4	n dead nodes $M_j, j = 1, \ldots, n$ 0 alive nodes	$A \wedge B \wedge C \wedge D \to M_4$	No extra hyperarc as J is still at risk of continuing progression.	$\sum_{d=1}^{n} d\binom{n}{d}$
5	n dead nodes $M_j, j = 1, \ldots, n$ 1 alive node S	$A \wedge B \wedge C \wedge D \to M_4$	$A \wedge B \to S$	$\sum_{d=1}^{n} (d+1)\binom{n}{d}$

Table 3: 6 different ways to include mortality as nodes into the directed hypergraph.

Wherever a survival node is included on the setup, every individual will have a final multimorbidity disease progression where the head set includes either the mortality node or the survival node. Mortality types 1 and 0 specify a single mortality node with and without an alive node. Mortality types 3 and 2 specify a mortality node for each disease node. Therefore, for an individual who died, this setup generates a final multimorbidity hyperarc to have the head node equal to the mortality node corresponding to the individual's final disease. Mortality types 5 and 4 specify a mortality node for each degree hyperarc, which is the total number of disease nodes. Therefore, for an individual who died, their additional multimorbidity hyperarc will have its head node as the mortality node corresponding the number of conditions the individual died with.

6 Analysis on hypergraphs

6.1 Eigenvector centrality

Eigenvector centrality measures the transitive influence of nodes. That is, each nodes importance is based on its connectivity to other nodes, while also giving consideration to the connections to those same nodes. Consider it as a popularity score, so that a nodes popularity score is based not only on the popularity of itself, i.e. number of connections between other nodes, but also the popularity of these connecting nodes.

In this case, centrality is used for analysing the undirected hypergraph to determine the most central single diseases in the standard representation and sets of diseases in the dual representation. This measure can only be used for analysing the nodes and edges in the undirected representation of the directed hypergraph in section 5.4.

If A is our weighted adjacency matrix of the hypergraph, i.e. $a_{ij} > 0$ if nodes i and j are connected in the hypergraph and $a_{ij} = 0$ otherwise. Let $\mathbf{x}$ be the n-length vector of eigenvector centrality's for all n nodes in the hypergraph

$\mathcal{H}(V, \mathcal{E})$, where $|V| = n$. Then the eigenvector centrality score, x_i of node i is defined by the sum of all eigenvector centrality's of nodes incident to node i such that

$$x_i = \frac{1}{\lambda} \sum_{j \in M_i} x_j = \frac{1}{\lambda} \sum_{j=1}^{n} A_{ij} x_j, \tag{57}$$

where λ is some normalisation constant such that $\lambda > 0$ when the weighted adjacency matrix is non-negative. M_i is the set of nodes incident to node i which can be deduced by summing over the i^{th} row of the weighted adjacency matrix A, since for any node j not connected to node i, $A_{ij} = 0$. By substitution of the weighted adjacency matrix, and multiplication of both sides by λ, we recover the eigenvector equation in matrix notation

$$A\mathbf{x} = \lambda \mathbf{x}. \tag{58}$$

Therefore, to compute the eigenvector centrality, we compute the eigenvectors of the weighted adjacency matrix. In general, there will be many different eigenvalues λ for which a non-zero eigenvector solution exists. Given the weighted adjacency matrix is a symmetric, real and non-negative matrix, the Perron-Frobenius theorem ensures that the eigenvector corresponding the largest eigenvalue (which will be greater than or equal in absolute value to all other eigenvalues) will have all non-negative entries. Therefore, the eigenvalue corresponding to this maximum eigenvalue results in the desired centrality values.

The elements of this eigenvector can be interpreted as a measure of centrality for each of the nodes. The interpretation of eigenvector centrality for hypergraphs is very similar to the interpretation for classic binary graphs. A minor difference is that more than one edge can contribute to each element of the weighted adjacency matrix in a hypergraphs.

Using centrality as a proxy for importance, in the context of multimorbidity this allows us to quantitatively evaluate the most important single diseases using the standard representation of the undirected hypergraph, as well as the most important multimorbidity disease sets using the dual representation of the undirected hypergraph. Also, performing eigenvector centrality on the undirected representation for the directed hypergraph will identify the most central tail- and head-specific single diseases, and doing so on its dual representation will identify the most central disease progressions.

6.2 Random walks on undirected hypergraphs

A random walk is a particular case of a Markov random chain, a random process that consists on visiting a certain number of locations, or states, by taking random steps. In a random walk with n states, each step is taken independently from the previous step, and consequently its behaviour is completely determined by an $n \times n$ *transition* probability matrix P where P_{ij} represents the probability for a random walk at location i to transition to location j. Multiple transitions can be made from an initial probabilistic state through multiple application of the probability transition matrix P.

Following the development of random walks on graphical objects by Zhou [25], we can associate a random walk to an undirected hypergraph $\mathcal{H}(V, E)$. Given a current position node $u \in V$, we first choose a hyperedge $e \in E$ over all the hyperedges incident to u with a probability proportional to $w(e)$. The probability of transitioning from node u to another node at random $v \in V$, $p(u, v)$ is defined as

$$p(u, v) = \sum_{e \in \mathcal{E}} w(e) \frac{h(u, e)}{d(u)} \frac{h(v, e)}{\delta(e)}, \tag{59}$$

where $h(u, e) = 1$ if hyperedge $e \in E$ connects node. d is the node degree function and δ the edge degree functions, as defined in section 3.1.

A nonzero transition probability between u and v only exists if the two nodes are linked by at least one hyperedge. We normalise the probability by $d(u)$ because a node with a large degree (many hyperedges connect to u) has a smaller chance to choose *distinct* hyperedge for the transition. We normalise by $\delta(e)$ so that the transition between u and an incident node v is chosen uniformly at random from all hyperedges incident with nodes u and v.

The transition matrix P of the random walk is defined in matrix form as

$$P = D_v^{-1} H W_e D_e^{-1} H^{\mathrm{T}}, \tag{60}$$

where D_v, D_e are the node and edge degree matrices, W_e the edge weight matrix and H the incidence matrix. Note these symbols represent the undirected case.

6.3 PageRank algorithm

This transition matrix can be represented by a directed graph which can then be directly analysed through the PageRank algorithm. PageRank is the eigenvector centrality of the directed graph representation and is interpreted similarly as the transitive influence for a node in a directed graph according to the number of incoming connections that node had, the strength of those connections, and the strength of the connections connected to those aforementioned connections. Consider the following transition matrix for 3 node directed hypergraph with nodes A, B and C.

$$P = \begin{array}{c} \\ A \\ B \\ C \end{array} \begin{array}{c} \begin{array}{ccc} A & B & C \end{array} \\ \begin{pmatrix} 0.6 & 0.25 & 0.15 \\ 0.5 & 0.4 & 0.1 \\ 0.45 & 0.20 & 0.35 \end{pmatrix} \end{array} \tag{61}$$

Note also that $P_{i,j} \geq 0$ and $\sum_j P_{i,j} = 1$. We can visualise this transition matrix as the directed graph below in figure 15.

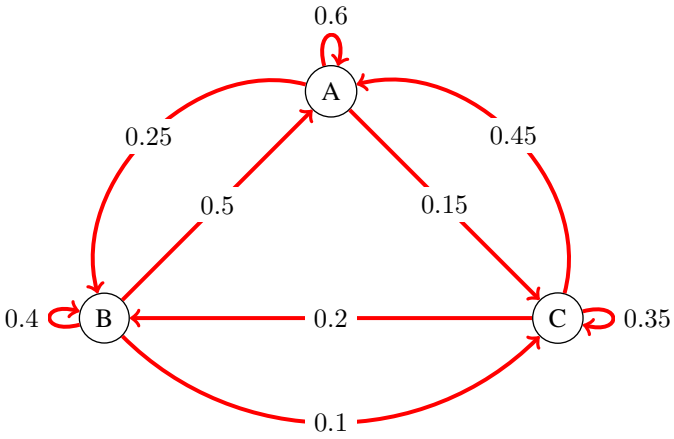

Figure 15: Example weighted, directed graph corresponding to the transition matrix P above.

Note that node A has the strongest incoming connections compared to nodes B and C. This visualisation becomes important to understand the results of computing the PageRank vector down below.

The transition probability matrix P tells us the probability of transitioning between nodes in the directed graph such as the one above. Given some initial probability vector $\boldsymbol{x}_0$, consider looking at long-term probabilities of transition matrix P, i.e. we take t transitions from $\boldsymbol{x}_0$ by applying P^t, but where $t \to \infty$. By the fact P is a Markov matrix this means

$$\lim_{t \to \infty} \boldsymbol{x}_0^{\mathrm{T}} P^t \to \boldsymbol{\pi}. \tag{62}$$

Vector $\boldsymbol{\pi}$ represents the *steady state probabilities* of the nodes in the directed graph. These steady state probabilities represent the long-term probability of reaching a target node, independent from starting at any source node or initial probability vector. As with any probability, this implies how likely the node is to be visited or reached in the network. Moreover, these probabilities do not change under application of the transition matrix such that

$$\boldsymbol{\pi}^{\mathrm{T}} = \boldsymbol{\pi}^{\mathrm{T}} P. \tag{63}$$

Therefore, there is an equivalence between these steady state probabilities and the principal left eigenvector of the stochastic matrix P. This is due to a key property of a stochastic matrix, which is that P has a *principal left eigenvector* whose eigenvalue is 1. Moreover, since these stochastic matrices P are non-negative stochastic matrices so by the Perron-Frobenius theorem, they have all unique and positive eigenvalues which can be ranked, whose largest eigenvalue will be 1.

It follows based on the original work by Brin and Page [26], that the vector π is the *PageRank vector* of the directed graph, ranking the nodes in terms of how accessible, or central, each node is based on its in-degree connectivity and the in-degree connectivity of nodes which have incoming connections to it.

Note that for the left dominant eigenvector to be unique, corresponding to the largest eigenvalue which is always 1, the stochastic matrix P must be *irreducible*. A stochastic matrix is irreducible if a pair of states are able to communicate to

one another at some finite time transitions, i.e. state j is accessible from state i if there exists an integer $n > 0$ such that

$$P_{ij}^n = \mathbb{P}(X_n = j \mid X_0 = i) > 0, \tag{64}$$

where X_i is the current state at the n^{th} transition time step. If this is satisfied, then one can get from state i to state j in n steps with probability P_{ij}^n. Irreducibility requires that every state j is accessible from any other state i and vice versa, i.e. states i and j communicate. A reducible stochastic matrix is one where entering a particular state blocks access to other states, resulting in zero probability transitions from that state to certain other states. A popular approach [26] to force an $n \times n$ reducible stochastic matrix $\hat{P}$ into an irreducible stochastic matrix P is to perform the following transformation

$$P = \alpha P + \frac{1-\alpha}{n} \mathbb{1}_n \mathbb{1}_n^{\mathrm{T}}, \tag{65}$$

where $\mathbb{1}_n \mathbb{1}_n^{\mathrm{T}}$ is an $n \times n$ matrix of ones. The resulting matrix P now has strictly positive entries everywhere, meaning that all states communicate, albeit some with only the smallest probability. The resulting matrix P is now an irreducible stochastic matrix with strictly positive entries. Therefore, by the Perron-Frobenius theorem, there will always be a unique left, dominant eigenvector corresponding to the largest eigenvalue which will be 1.

α is the called the damping factor in Brin and Page's original work [26]. The damping factor in webgraphs is used to describe a random webpage surfer's probability in randomly jumping to any webpage. For webpages previously not reachable in the reducible transition matrix, their transition probability is now $\frac{1-\alpha}{n}$. In the case for multimorbidity, if we have a reducible transition matrix we don't want an individual to transition to a random progression, so our damping factor α is very close to 1.

Using the simple 3-node example from figure 15, we can now compute the PageRank vector which is just the left eigenvector of the dominant eigenvalue, which because P is a row-stochastic matrix, will always be 1. The PageRank vector in this case is

$$\pi = \begin{matrix} A \\ B \\ C \end{matrix} \begin{pmatrix} 0.55 \\ 0.28 \\ 0.17 \end{pmatrix}. \tag{66}$$

Note that $\sum_i \pi_i = 1$ and node A has the largest long-term probability. This means that the long-term probability of remaining at node A, independent of the initial starting node, is highest at 0.55. This is what we expected given that the weighted directed graph above shows node A to have the strongest incoming connections. The ranking of all nodes also aligns with our probability transition matrix, where B had stronger incoming connections than node C.

6.4 PageRank for directed hypergraphs

6.4.1 Notation

Let $\mathcal{H}_D(V, \mathcal{E})$ be a directed hypergraph, where V is the set of nodes and $\mathcal{E}$ is the set of hyperarcs. Each hyperarc $e = \langle T(e), H(e) \rangle \in \mathcal{E}$ where $T(e), H(e) \subset V$ represent the set of tail nodes and head nodes, respectively, which are assumed to be nonempty. The directed hypergraph $\mathcal{H}_D$ can be represented using two $n \times k$ incidence matrices M_+ and M_- such that

$$m_-(v,e) = \begin{cases} 1 & \text{if } v \in T(e) \\ 0 & \text{otherwise} \end{cases} \tag{67}$$

$$m_+(v,e) = \begin{cases} 1 & \text{if } v \in H(e) \\ 0 & \text{otherwise} \end{cases} \tag{68}$$

$|V| = n$ and $|\mathcal{E}| = k$. These matrices are indexed such that nodes represent rows and hyperarcs represent columns. Let $w(e)$ be the weight of the hyperarc e. Let W_e be the diagonal matrix containing the weights of the hyperarcs in its diagonal entries. Moreover, let $w(v)$ be the weight for node v and W_V be the diagonal matrix containing the weights of the nodes in its diagonal entries. From the above definitions, we can define the tail and head degrees of the node v and the tail and head degrees of the hyperarc e as

$$d_-(v) = \sum_{e \in \mathcal{E}} w(e) m_-(v,e), \tag{69}$$

$$\delta_-(e) = \sum_{v \in V} m_-(v,e), \tag{71}$$

$$d_+(v) = \sum_{e \in \mathcal{E}} w(e) m_+(v,e), \tag{70}$$

$$\delta_+(e) = \sum_{v \in V} m_+(v,e). \tag{72}$$

Let D_{v^-}, D_{v^+}, D_{e^-} and D_{e^+} be four diagonal matrices containing the tail and head degrees of nodes and the tail and head degrees of hyperarcs in their diagonal entries respectively. Note that D_{v^-} and D_{v^+} are $n \times n$ matrices and D_{e^-} and D_{e^+} are $k \times k$ matrices. As our directed hypergraph uses only B-hyperarcs, this means $\delta(e) = 1$ for all $e \in \mathcal{E}$, and so $D_{e^+} = \mathbb{I}_k$ where I_k is the $k \times k$ identity matrix.

6.4.2 Successor detection

Extending the notion of a random walk on an undirected hypergraph to a directed hypergraph, we consider the following transition rule defined in [27]: given the current position $u \in V$, we choose a hyperarc $e \in \mathcal{E}$ such that $u \in T(e)$ with a probability proportional to $w(e)$. If we consider that a transition can only be made from a tail to a head of a hyperarc, then we choose a node $v \in H(e)$ uniformly at random. The transition probability of the directed random walk from nodes u to v, $p(u, v)$ is defined as

$$p(u,v) = \sum_{e \in \mathcal{E}} w(e) \frac{m_-(u,e)}{d_-(u)} \frac{m_+(v,e)}{\delta_+(e)}. \tag{73}$$

A nonzero transition probability between u and v exists only if there exists a hyperarc e such that $u \in T(e)$ and $v \in H(e)$, following the intuitive idea that a transition can only be done in the direction given by a hyperarc. The probability is normalised by the outer degree of u to represent the probability of choosing a distinct hyperarc between those which contain u in their tails. The inner degree of the hyperarc e is also introduced to take into account that a node v is chosen uniformly at random from $H(e)$. However, since $|H(e)| = 1$, $\delta_+(e) = 1$ for all $e \in \mathcal{E}$ so the transition probability only relies on the choice of a distinct hyperarc where $u \in T(e)$ and $v \in H(e)$. In matrix form, the transition probability matrix P is defined by

$$P = D_{v^-}^{-1} M_- W_e M_+^{\mathrm{T}}. \tag{74}$$

We can show that the rows of the transition probability matrix sum to 1 [28], so that $p(u, v) \geq 0$, for all $u, v \in V$ and $\sum_{v \in V} p(u, v) = 1$, for all $v \in V$. This shows that P satisfies the Markov property and hence is a *stochastic* matrix. This stochastic process models the transition nodes can go following the direction of observed disease progressions. P_{ij} represents the transition probability of progressing from disease i to disease j. In the context of multimorbidity, this transition rule models the transition from a predecessor disease belonging to the tail of a hyperarc to a successor disease belonging in the head of a hyperarc, i.e. the onset disease in an observed disease progression.

This section discussed a transition rule for detecting important successor diseases, i.e. those diseases which are subsequent diagnoses from other conditions. The disease with the highest probability in the PageRank vector will be one which most frequently is flagged by the healthcare system as a subsequent diagnosis to other predecessor diseases. This is computed based on the prevalence-weighted number of observed disease progressions of which have this disease as the head node.

6.4.3 Predecessor detection

This subsection outlines the transition rule for detecting important predecessor diseases. Consider the following transition rule: given the current position $u \in V$, we choose a hyperarc $e \in \mathcal{E}$ such that $u \in H(e)$ with a probability proportional to $w(e)$. If we now consider the *inverse* transition compared to above, i.e. a transition can only be made on a hyperarc from its head to any member of its tail, then we choose a node $v \in T(e)$ uniformly at random. The transition probability of the directed random walk from nodes u to v, $p(u, v)$ is defined as

$$p(u,v) = \sum_{e \in \mathcal{E}} w(e) \frac{m_+(u,e)}{d_+(u)} \frac{m_-(v,e)}{\delta_-(e)}. \tag{75}$$

A nonzero transition probability between u and v exists only if there exists a hyperarc e such that $u \in H(e)$ and $v \in T(e)$. The probability is normalised by $d_+(u)$ to represent the probability of choosing a distinct hyperarc between those which contain u in their head. We normalise by $\delta_-(e)$ so that node $v \in T(e)$ is chosen uniformly at random from all nodes in $T(e)$. In matrix form the transition probability matrix P is defined by

$$P = D_{v^+}^{-1} M_+ W_e D_{e^-}^{-1} M_-^{\mathrm{T}}. \tag{76}$$

The proof to show that $p(u, v)$ is a probability and that P is a row-stochastic matrix is the same as for the successor detection probability transition matrix [28]. According to our transition rule, we would interpret a transition probability P_{ij} to be the probability that disease node j was a predecessor disease of successor disease node i.

Computing the PageRank vector here would help us detect central disease nodes which act as important predecessor conditions for other diseases. The disease with the highest probability in the PageRank vector will be the one which most frequently is flagged by the healthcare system prior to many other subsequent diagnoses. This is computed based on the prevalence-weighted number of observed disease progressions where the disease is within the tail set.

7 Future work and investigation

Note that the work outlined below has not been in implemented in full or tested on the Charlson/Elixhauser population tables. Instead, the sections below outline future endeavours and extensions to this project which were unfortunately not completed in the time frame of the project.

7.1 Demographic hypergraphs

7.1.1 Bipartite representation of a hypergraph

Studying the centrality of the hypergraph $\mathcal{H}$ identifies the most prevalent single diseases, while the dual hypergraph $\mathcal{H}^*$ identifies the most coincident sets of diseases, i.e. sets of 2 or more conditions. There is an exclusivity here whereby the former determines prevalence based on single-disease sets of conditions and the latter determines prevalence based on multimorbidity-disease sets. Therefore, in order to determine centrality of the all single-set and multimorbidity-set conditions, we can use a *bipartite* network.

The bipartite graph, $\mathcal{G}(V \cup E, E_b)$ is a simple graph (each edge can only connect 2 nodes) representation of the original hypergraph $\mathcal{H}(V, E)$ whose bipartite nodes are all nodes and edges of $\mathcal{H}$. There is an additional constraint in $\mathcal{G}$ that all nodes have a binary partition label - the original nodes of V have the label 0 and the edges E have the label 1. This binary partition restricts any edge $e \in E_b$ of the bipartite graph to connect any bipartite node with the same binary label. In this sense, the graph is also known to be *2-colourable*, i.e. if each partition label had a colour assigned, then no edge in the bipartite network would be monochromatic. This is because each edge will connect nodes of a different colouring.

Investigating the centrality of this bipartite graph allows for the quantification of the important of individual diseases and sets of diseases together. Figure 16 visualises the bipartite graph representation of $\mathcal{H}$ shown in figure 5 in section 3.

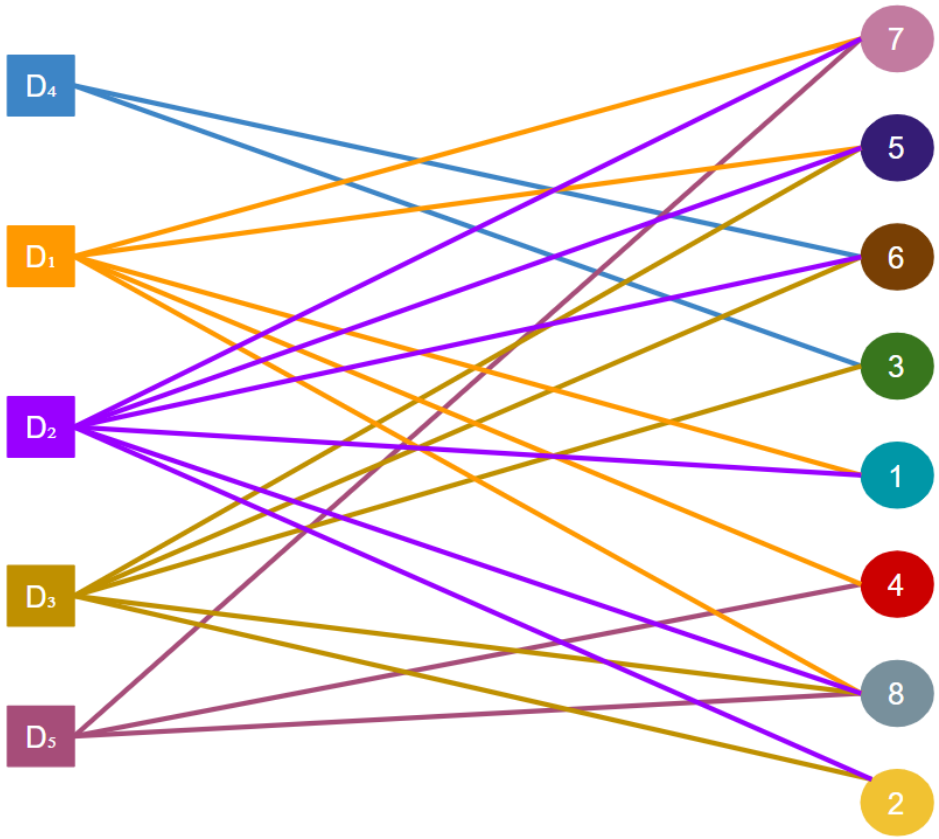

Figure 16: Bipartite graph $\mathcal{G}$ of hypergraph $\mathcal{H}$.

Because there are $|V| = 5$ nodes and $|E| = 8$ edges of $\mathcal{H}(V, E)$, this means the bipartite representation has $|V| + |E| = 13$ total nodes. Moreover, there are a total of 21 individual node-to-hyperedge connections in $\mathcal{H}$, hence $|E_b| = 21$ in $G(V \cup E, E_b)$. This makes the incidence matrix for the bipartite graph M_b a 13×21 binary matrix whose columns only contain a maximum of two ones. This is because the bipartite graph is now a simple graph whose edges are fixed to only connect two nodes at most.

One way to define the edge weights for the bipartite graph is to translate over the edge weights W_E from the original hypergraph $\mathcal{H}$. What this means is that for any bipartite edge connecting a node with binary label 0, i.e. a single disease, to a node with a binary label 1, i.e. a set of diseases, the edge weight will be the edge weight used to quantify the edge connecting those sets of diseases from $\mathcal{H}$. Therefore, the weight of hyperedge $[W_E]_{ii}$ is applied to all new edges of the bipartite graph that connect the i'th hyperedge node of the bipartite graph (with binary label 1).

In the context of multi-morbidity, when considering the centrality of nodes in the dual hypergraph, the set of diseases with the largest centrality is the set of diseases that is most strongly connected to other sets of diseases, and is not necessarily an indication of how strongly connected the diseases are within the set. One must look at centrality of both the hypergraph and the dual hypergraph — or alternatively the bipartite representation of the hypergraph as discussed here — to form a complete picture of the centrality of diseases and sets of diseases.

7.1.2 Demographic bipartite hypergraphs

Much research in multimorbidity include demographics into the analysis by stratifying the population based on demographic factors such as sex, age, socioeconomic status, geographical location or mortality status before conducting analysis and performing inter-model comparisons. This would however prevent allowing these demographic relationships to influence each other if they were instead part of the network itself.

An explicit would be to include demographic variables as nodes into the graph, and there are three ways to do this, and all include the idea of *bipartiteness* and *2-colourability*. Consider the following sets, $X = \{D_1, \ldots, D_n\}$, $Y = \{A, S, D, L, M\}$ and $X^* = \{\mathcal{D}_1, \mathcal{D}_2, \ldots, \mathcal{D}_m\}$. X is the set of diseases, while X^* is the set of all relevant hyperedges for the hypergraph $\mathcal{H}(X, \mathcal{E})$, i.e. $\mathcal{D}_1 = \{D_1, D_2, D_3\}$, $\mathcal{D}_2 = \{D_1, D_3\}$, etc. The set Y contains sets of demographic variables,

- The set A contains age group categories, for example $A = \{[18, 30), [30, 40), \ldots, [90, 100)\}$.
- The set S contains sex nodes, i.e. $S = \{\text{Male}, \text{Female}\}$
- The set D contains nodes to measure socioeconomic status, i.e. deprivation, $D = \{1, 2, \ldots, 5\}$ measured according to WIMD.
- The set L contains location nodes to measure geographic status, i.e. $L = \{\text{Village}, \text{Urban Town}, \ldots, \text{Urban City}\}$.
- The set M contains two nodes identifying whether an individual died during period of analysis or was alive by the end of the period of analysis, i.e. $M = \{\text{Dead}, \text{Alive}\}$.

We could then envisage constructing three potential graphs, a bipartite graph $G_b(X^* \cup Y, E)$, a 2-colourable hypergraph $\mathcal{H}_b(X \cup Y, \mathcal{E}_b)$ and a semi-bipartite hypergraph $\mathcal{H}_b^*(X^* \cup Y, \mathcal{E}^*)$, where we note a binary partition label, or colour, of 0 to members in X, X^* and 1 to members in Y.

1. For $G_b(X^* \cup Y, E)$: This is a *bipartite* graph. Note that in a simple bipartite graph, edges are restricted to only connect nodes which have a different binary partition label or colour, so in the simple bipartite graph construction we would consider simple edges connecting singular members of X^* to Y. Note that members of X^* are in fact hyperedges from $\mathcal{H}(X, \mathcal{E})$ as previously discussed. An edge here would represent a particular population demographic where a set of diseases have been observed (i.e. people who have a deprivation status of 1 with alcohol AND drug abuse).
2. For $\mathcal{H}_b(X \cup Y, \mathcal{E}_b)$: This is a *2-colourable* hypergraph, i.e. each hyperedge in $\mathcal{E}_b$ must contain at least one member of X and Y so that no hyperedge is monochromatic (see section 7.1.1). This representation is not a bipartite hypergraph however, as hyperedges are free to connect nodes with the same binary partition label, *as long* as they also connect to members of the other partition label. Note how the property of bipartiteness is a stronger condition to 2-colourability. Moreover, this is a different construction to $G_b(X^* \cup Y, E)$, since each node in X represents a single disease, while each node in X^* represents a set of diseases already identified by $\mathcal{H}$. This allows us to group together demographic variables with sets of diseases, for example males under the ages of 30, with a deprivation status of 1 living in an urban city, who have alcohol abuse, drug abuse and hypertension.

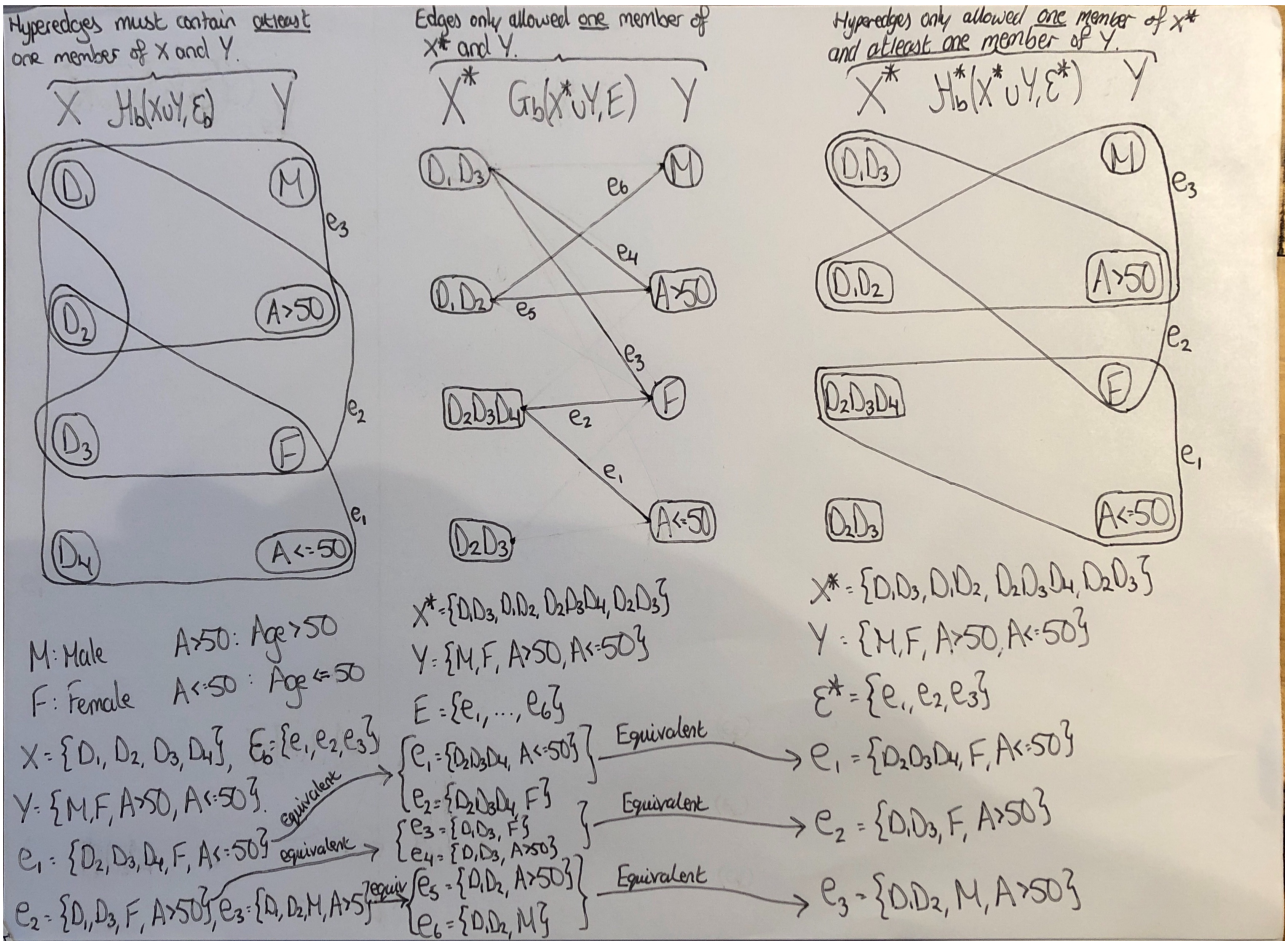

Figure 17: Simple but equivalent examples of $\mathcal{H}_b$, G_b and $\mathcal{H}_b^*$ using 4 diseases and 4 demographic nodes.

3. For $\mathcal{H}_b^*(X^* \cup Y, \mathcal{E}^*)$: This is a *semi-bipartite* hypergraph. We extend the notion of bipartiteness to hypergraphs by restricting hyperedges to only connect *exactly one* member of X^* to *exactly one* member of Y — this would of course recover G_b given nodes of label 0 are members of X^*. Instead, a semi-bipartite hypergraph $\mathcal{H}_b^*$ in this context restricts a hyperedge to connect *exactly one* member of X^* to *at least* one member of Y. This allows the potential to have hyperedges which connect a set of diseases to multiple demographics, for example males under the ages of 30, with a deprivation status of 1 living in an urban city, who have alcohol abuse, drug abuse and hypertension. The advantage of this construction over G_b is that we can group together multiple demographics with a set of diseases, while in G_b we can only specify a single demographic.

Figure 17 shows an example construction for graphs $\mathcal{H}_b$, G_b and $\mathcal{H}_b^*$ using the same edge and node equivalences.

Edge weights in the first instance could be a measure of disease-demographic prevalence among the population using the rules outlined in section 4. In particular, we would want individuals to contribute to the prevalence of hyperedges which connect their disease set and the power set of all their demographic factors so that all disease-demographic hyperedges are constructed evenly from the population.

Note that depending on the number of demographic and disease nodes, constructing $\mathcal{H}_b$ or $\mathcal{H}_b^*$ may be far more computationally involved than constructing G_b (see section 3.2), and in G_b and $\mathcal{H}_b^*$ we already know the members of X^* from $\mathcal{H}$. However, a downside to G_b is that each edge is restricted to only connecting one type of demographic to the sets of diseases, in that each edge only allows the relationship between a set of diseases and a single demographic. A potential solution would be to convert G_b to a bipartite hypergraph $\mathcal{H}_b^*$ as described above. However, this could bear the burden of large computational resource depending on the number of nodes.

Note that the example in $\mathcal{H}_b$ and $\mathcal{H}_b^*$ are the same example, but the different construction means that we explicitly only consider sets of multimorbidity in $\mathcal{H}_b^*$ instead of individual diseases so that hyperedges are restricted to only allowing

exactly one member of X^* and many members of Y, contrary to the hyperedges allowed in $\mathcal{H}_b$. This has the benefit of reducing the possible hyperedges available in $\mathcal{H}_b^*$, unlike in $\mathcal{H}_b$. Moreover, the members of X^* could be reduced according to the analysis of the original hypergraph $\mathcal{H}$, i.e. only consider the most central sets of diseases as members of X^*, to try and reduce the number of nodes in the graph.

Note that in the semi-bipartite and 2-colourable hypergraphs, we need an additional rule that hyperedges cannot connect two demographic nodes which represent different categories of the the same demographic outcome, i.e. $e_i = \{D_{i_1}, \ldots, D_{i_n}, \ldots, M, F\}$ would connect a series of single diseases to males and females. Although contextually this might make sense since this edge considers *any sex* that are diagnosed with the same set of diseases $D_{i_1}, \ldots, D_{i_n}$, logically this wouldn't make sense as we consider edges, and hyperedges, to be logical AND operators, in that an edge e_i would study the prevalence of individuals where diseases $D_{i_1}, \ldots, D_{i_n}$ are present, and who are male *and* female which is impossible.

A way to circumvent the above problem is to have a $k+1$-partition, for $k = |Y|$ above, i.e. have a partition label for each type of demographic, as well as a label for the disease node set X or X^*. This would permit $\mathcal{H}_b^*$ to become fully $k+1$-partite since each hyperedge would link *exactly one* node from each partition label. Hypergraph $\mathcal{H}_b$ would still be semi-$k+1$-partite as we allow hyperedges to connect multiple members from node set X. Following the examples in figure 17, figure 18 would be an equivalent 3-partite hypergraph.

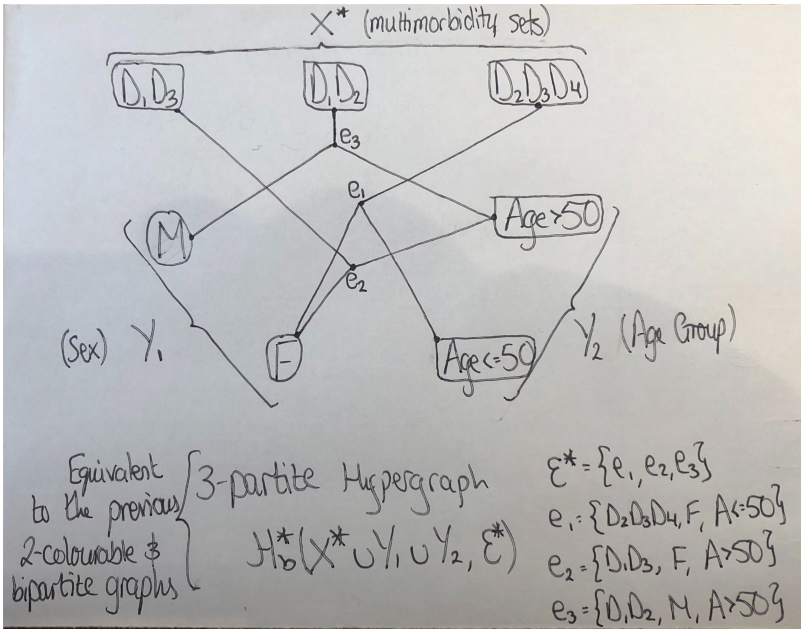

Figure 18: 3-partite hypergraph $\mathcal{H}_3^*(X^* \cup Y_1 \cup Y_2, \mathcal{E}^*)$ equivalent to the examples shown in figure 17.

7.2 Acute episodic events

The definition of multimorbidity at its most basic level is the co-occurrence of two or more illnesses in the same individual at the same time. The exact definition can cause controversy because of the specifics of the illnesses classed as a co-morbid disease [29]. In a population-level analysis of multimorbidity, models are typically built assuming that once an individual has a disease, it remains within them throughout the period of analysis — this condition would be considered a chronic one, such as diabetes or paraplegia. However, many consider the term multimorbidity to encapsulate both *acute* and *chronic* conditions. While an acute event can be justifiably argued to have long-lasting impacts to the body such as a myocardial infarction or an acute stroke, these conditions can be considered *episodic* events in nature. Moreover, these events can happen multiple time in ones life time, for example it is far more likely for another stroke to occur if an individual has already one one before.

During our analysis we identified disease nodes within the Charlson comorbidity index and Elixhauser comorbidity index which are indeed acute, episodic events. For example, in the Charlson comorbidity index, diseases such as cancer, myocardial infarction and stroke can all be considered episodic as they pathways which could lead to remission (cancer) or are acute (myocardial infarction and stroke). The presence of these episodic conditions are exaggerated in the Elixhauser comorbidity index where conditions like alcohol abuse, drug abuse, obesity, weight loss, depression,

psychoses, lymphoma, metastatic cancer can all be considered episodic events. Some of these conditions could even be considered entirely curable, leaving the individual to live a healthy life.

A possible consideration to model these episodic events would be have a single node representing each *chronic* condition while a series of nodes to index the possibly numerous episodic events for each acute illness. This would allow a more comprehensive construction of an individual's disease progression. This additional information would open doorways into understanding how several episodic events of the same acute illness may lead to increased mortality rate as opposed to singular episodic events, etc. It may also reveal a distinct pattern in the healthcare system in that someone with multiple acute events are more likely to take up more healthcare resources so that individual can be checked for other comorbidities helping lead to the second or third acute event. For example, an individual may go through 2 strokes and 2 acute myocardial infarctions, among other conditions, before dying. It would be incredibly useful to model this progression, i.e.

$$\text{Diabetes} \to \text{Renal} \to \text{MI}_1 \to \text{CHF} \to \text{MI}_2 \to \text{Stroke}_1 \to \text{Paraplegia} \to \text{Stroke}_2 \to \text{Death}. \tag{77}$$

In this case the individual died after their second stroke, but it would have been useful to know that the congestive heart failure helped contribute to the second cardiac event the individual suffered. The current model would have only modelled the first stance of MI and stroke.

The reason why this interesting endeavour was not fully investigated is because of the expected lead time to extract all relevant patient information on these episodic events. The prep-processed tables only go so far as to outline the first and last recorded date which patients were observed to have each disease during interactions with the healthcare system. We would need to do a deep dive into the raw data and look at any patient's health records who suffered an acute illnesses and understand how many times it might have happened.

Difficulties could then arise such as the reliability of each record specifying that the acute event had happened, i.e. it is possible clinicians will specify an acute event in their notes, referencing the event in relation to another matter. In this case, the event has already happened and this second reference does not imply the individual is having the same event. These caveats would need to be explored and investigated to make sure issues like these don't crop up in the creation of the model.

Another potential issue is the growth in computational complexity with the more acute, episodic nodes we add. This might not be such an issue with Charlson comorbidity index as we could consider 3 episodic nodes for MI, stroke and cancer and we still would be able to construct the model without issue. The concern is implementing episodic nodes into the Elixhauser comorbidity index because we already consider a 27-node directed hypergraph which although runs in a very reasonable time, using binary representations to store prevalence arrays (see section B) means that the more nodes present in the graph, the memory requirements grows exponentially. There is of course a workaround, which is to improve the memory storage of these prevalence arrays. But that is another implementation-focused future endeavour.

7.3 Other centrality measures

This methodology report has focused solely on eigenvector-based centrality, detecting the transitive influence of disease nodes, allowing the detection of important successor and predecessor conditions in multimorbidity disease progressions. There are a number of other centrality-based measures which have different interpretations that could be useful to implement to better understand the temporal, directional or temporal roles each disease node with other ones. This may be a worthwhile extension — note that many of these centrality measures are for the undirected case, but as observed how eigenvector centrality's extension to the directed case is PageRank, there is likely directed versions of other centrality measures. Below is a list of some other centrality measures and how they might be used in the study of multimorbidity research and disease progressions:

1. **Degree** centrality: This assigns an important score based simply on the number of prevalence-weighted links held by each node in the undirected case. In the directed case, we can measure out-degree in-degree and centrality of the nodes in the directed hypergraph which measures the relative score a node has as belonging in the tail set of a hyperarc (out-degree) or in the head set of a hyperarc (in-degree), i.e. the positioning of the disease in a multimorbidity disease progression.
2. **Closeness** centrality: This centrality scores each node based on their 'closeness' to all other nodes in the network. For each node, shortest paths (based on the problem-dependant edge weights) between all other nodes are calculated, assigning that node a score based on its sum of shorted paths. This helps find nodes who are best placed to influence the entire network most quickly. This seems like more of an undirected approach to understand through the prevalence-weighted multimorbidity disease sets, which disease nodes are most similar to each other based on the population.

3. **Betweenness** centrality: This measures the number of times a node lies on the shortest path between other nodes. This shows which nodes are 'bridges' between nodes in a network. It does this by identifying all the shortest paths and then counting how many times each node falls on one. This could be especially helpful for understanding which diseases mostly occur in the middle of multimorbidity progressions, rather than at the beginning or at the end.

7.4 Hyperedge weight exploration

Stability analysis.

Performing a stability analysis on the choice of weighting formula from using the same type of outcome/measure, i.e. given there are an infinite family of formulae for deducing the weight of edges according to disease prevalence, what do different formulae from the same family do to the construction of the hypergraph and their corresponding hyperedge weights?

Prevalence & healthcare resource utilisation.

In Jim's followup paper [24], he computed three hypergraphs; $\mathcal{H}_1, \mathcal{H}_2$ and $\mathcal{H}_3$. $\mathcal{H}_1$ constructed a hypergraph by weighting hyperedges according to disease prevalence using the overlap coefficient (see section 4.4). $\mathcal{H}_2$ and $\mathcal{H}_3$ were constructed by weighting hyperedges according to inpatient and outpatient healthcare resource utilisation. These two qualitative measures used GP visits, length of inpatient hospital stay, surgery times, etc. to construct a weighting for how much a burden a particular set of diseases had on the healthcare service. These three hypergraph's were individually assessed using hypergraph eigencentrality and then were combined by computing the 3-dimensional Euclidean norm of each hypergraph's eigencentrality for each multimorbidity set. This produced a single number for each multimorbidity, providing a ranking system for multimorbidity sets.

An alternative would be the following: let W_1, W_2 and W_3 correspond to the hyperedge weight matrices for hypergraphs $\mathcal{H}_1, \mathcal{H}_2$ and $\mathcal{H}_3$. This extension would look at combining these weight matrices to use as the hyperedge weight matrix for a hypergraph which would take into account all of these measures together such that for a new hypergraph $\mathcal{H}_*$, the corresponding weight matrix would compute a weighted response $W_* = \sum_{i=1}^{3} \alpha_i W_i$ where α_i are individual weightings to determine how importance each attribute of disease prevalence or healthcare resource utilisation is toward the weighting for W_*.

Alternatively, this weighted scheme could consider health economics, whereby particular utilisation's of healthcare service have a cost or some monetary value assigned to them. Then, we can combine the 'cost' of a multimorbidity's cumulative inpatient and outpatient healthcare utilisation and sum this over the number of participants which have the set of conditions. The hope would be to equally weight high costing rare conditions with low costing, highly prevalent multimorbidity so that common sets of conditions are not over-exaggerated and uncommon sets are under-exaggerated.

A paper published in 2017 [30] discussed and outlined formula for computing inpatient and outpatient healthcare costs to NHS England tabulated by sex, age and IMD deprivation quintile. An accompanying Github repository [31] is available to download these population healthcare cost tables and may prove useful in helping define hyperedge weights for graphs which will include demographics as nodes (see section 7.1). A challenge will be combining these healthcare cost weightings with multimorbidity disease prevalence, as well as understanding how a subset of the population with a set of morbidities interacts with their overall population healthcare cost.

References

[1] L Robinson. Present and future configuration of health and social care services to enhance robustness in older age., 2015.

[2] JM Valderas, J Gangannagaripalli, E Nolte, CM Boyd, Martin Roland, A Sarria-Santamera, E Jones, and M Rijken. Quality of care assessment for people with multimorbidity, 2019.

[3] Andrew Kingston, Louise Robinson, Heather Booth, Martin Knapp, Carol Jagger, and MODEM project. Projections of multi-morbidity in the older population in england to 2035: estimates from the population ageing and care simulation (pacsim) model. *Age and ageing*, 47(3):374–380, 2018.

[4] Jane Lyons, Ashley Akbari, Utkarsh Agrawal, Gill Harper, Amaya Azcoaga-Lorenzo, Rowena Bailey, James Rafferty, Alan Watkins, Richard Fry, Colin McCowan, et al. Protocol for the development of the wales multimorbidity e-cohort (wmc): data sources and methods to construct a population-based research platform to investigate multimorbidity. *BMJ open*, 11(1):e047101, 2021.

[5] Mansour TA Sharabiani, Paul Aylin, and Alex Bottle. Systematic review of comorbidity indices for administrative data. *Medical care*, pages 1109–1118, 2012.

[6] David Metcalfe, James Masters, Antonella Delmestri, Andrew Judge, Daniel Perry, Cheryl Zogg, Belinda Gabbe, and Matthew Costa. Coding algorithms for defining charlson and elixhauser co-morbidities in read-coded databases. *BMC medical research methodology*, 19(1):1–9, 2019.

[7] Anne Elixhauser, Claudia Steiner, D Robert Harris, and Rosanna M Coffey. Comorbidity measures for use with administrative data. *Medical care*, pages 8–27, 1998.

[8] James Rafferty, Alan Watkins, Jane Lyons, Ronan A Lyons, Ashley Akbari, Niels Peek, Farideh Jalali-Najafabadi, Thamer Ba Dhafari, Alexander Pate, Glen P Martin, et al. Ranking sets of morbidities using hypergraph centrality. *Journal of Biomedical Informatics*, 122:103916, 2021.

[9] Rafferty Jim. Hypergraphs for multimorbidity research. `https://github.com/SwanseaUniversityMedical/multimorbidity_hypergraphs`, 2022.

[10] Sanne Pagh Møller, Bjarne Laursen, Caroline Klint Johannesen, Janne S Tolstrup, and Stine Schramm. Patterns of multimorbidity and demographic profile of latent classes in a danish population—a register-based study. *PloS one*, 15(8):e0237375, 2020.

[11] John E Cornell, Jacqueline A Pugh, John W Williams Jr, Lewis Kazis, Austin FS Lee, Michael L Parchman, John Zeber, Thomas Pederson, Kelly A Montgomery, and Polly Hitchcock Noël. Multimorbidity clusters: clustering binary data from multimorbidity clusters: clustering binary data from a large administrative medical database. *Applied multivariate research*, 12(3):163–182, 2008.

[12] Sabine Landau, Morven Leese, Daniel Stahl, and Brian S Everitt. *Cluster analysis*. John Wiley & Sons, 2011.

[13] M Mofizul Islam, Jose M Valderas, Laurann Yen, Paresh Dawda, Tanisha Jowsey, and Ian S McRae. Multimorbidity and comorbidity of chronic diseases among the senior australians: prevalence and patterns. *PloS one*, 9(1):e83783, 2014.

[14] Stephanie T Lanza, Linda M Collins, David R Lemmon, and Joseph L Schafer. Proc lca: A sas procedure for latent class analysis. *Structural equation modeling: a multidisciplinary journal*, 14(4):671–694, 2007.

[15] Zsolt Zador, Alexander Landry, Michael D Cusimano, and Nophar Geifman. Multimorbidity states with high sepsis-related deaths: a data-driven analysis in critical care. *bioRxiv*, page 491712, 2018.

[16] Beatriz Olaya, Maria Victoria Moneta, Francisco Félix Caballero, Stefanos Tyrovolas, Ivet Bayes, José Luis Ayuso-Mateos, and Josep Maria Haro. Latent class analysis of multimorbidity patterns and associated outcomes in spanish older adults: a prospective cohort study. *BMC geriatrics*, 17(1):1–10, 2017.

[17] Albert Roso-Llorach, Concepción Violán, Quintí Foguet-Boreu, Teresa Rodriguez-Blanco, Mariona Pons-Vigués, Enriqueta Pujol-Ribera, and Jose Maria Valderas. Comparative analysis of methods for identifying multimorbidity patterns: a study of 'real-world'data. *BMJ open*, 8(3):e018986, 2018.

[18] Shu Kay Ng, Richard Tawiah, Michael Sawyer, and Paul Scuffham. Patterns of multimorbid health conditions: a systematic review of analytical methods and comparison analysis. *International journal of epidemiology*, 47(5):1687–1704, 2018.

[19] Marlous Hall, Tatendashe B Dondo, Andrew T Yan, Mamas A Mamas, Adam D Timmis, John E Deanfield, Tomas Jernberg, Harry Hemingway, Keith AA Fox, and Chris P Gale. Multimorbidity and survival for patients with acute myocardial infarction in england and wales: Latent class analysis of a nationwide population-based cohort. *PLoS medicine*, 15(3):e1002501, 2018.

[20] Heinz Freisling, Vivian Viallon, Hannah Lennon, Vincenzo Bagnardi, Cristian Ricci, Adam S Butterworth, Michael Sweeting, David Muller, Isabelle Romieu, Pauline Bazelle, et al. Lifestyle factors and risk of multimorbidity of cancer and cardiometabolic diseases: a multinational cohort study. *BMC medicine*, 18(1):1–11, 2020.

[21] Kien Wei Siah, Chi Heem Wong, Jerry Gupta, and Andrew W. Lo. Patterns of multimorbidity. *medRxiv*, 2021.

[22] Fabian P Held, Fiona Blyth, Danijela Gnjidic, Vasant Hirani, Vasikaran Naganathan, Louise M Waite, Markus J Seibel, Jennifer Rollo, David J Handelsman, Robert G Cumming, et al. Association rules analysis of comorbidity and multimorbidity: the concord health and aging in men project. *Journals of Gerontology Series A: Biomedical Sciences and Medical Sciences*, 71(5):625–631, 2016.

[23] Lina Jin, Xin Guo, Jing Dou, Binghui Liu, Jiangzhou Wang, Jiagen Li, Mengzi Sun, Chong Sun, Yaqin Yu, and Yan Yao. Multimorbidity analysis according to sex and age towards cardiovascular diseases of adults in northeast china. *Scientific reports*, 8(1):1–9, 2018.

[24] James Rafferty et al. Using hypergraphs to quantify importance of sets of diseases by healthcare resource utilisation: A retrospective cohort study. *unpublished*, N.D.

[25] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning with hypergraphs: Clustering, classification, and embedding. *Advances in neural information processing systems*, 19, 2006.

[26] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[27] Aurélien Ducournau and Alain Bretto. Random walks in directed hypergraphs and application to semi-supervised image segmentation. *Computer Vision and Image Understanding*, 120:91–102, 2014.

[28] Loc Tran, Tho Quan, and An Mai. Pagerank algorithm for directed hypergraph. *arXiv preprint arXiv:1909.01132*, 2019.

[29] Marjorie C Johnston, Michael Crilly, Corri Black, Gordon J Prescott, and Stewart W Mercer. Defining and measuring multimorbidity: a systematic review of systematic reviews. *European journal of public health*, 29(1):182–189, 2019.

[30] Miqdad Asaria. Health care costs in the english nhs: reference tables for average annual nhs spend by age, sex and deprivation group. 2017.

[31] Miqdad Asaria. English nhs costs by age, sex and deprivation). `https://github.com/miqdadasaria/hospital_costs`, 2017.

A Representing hyperedges

During my investigations I discovered an efficient approach to representing individuals and hyperedges in the dataset which aided the computational efficiency of computing hyperedge and hyperarc weights significantly. In short, we can represent each hyperedge as a unique integer based on the disease node indexes the hyperedge connects being interpreted as a binary string. Figure 19 shows an example 4-node dataset visualising 13 hyperedges shown by the intersections of each disease indexed- circle and their corresponding unique integer mapping.

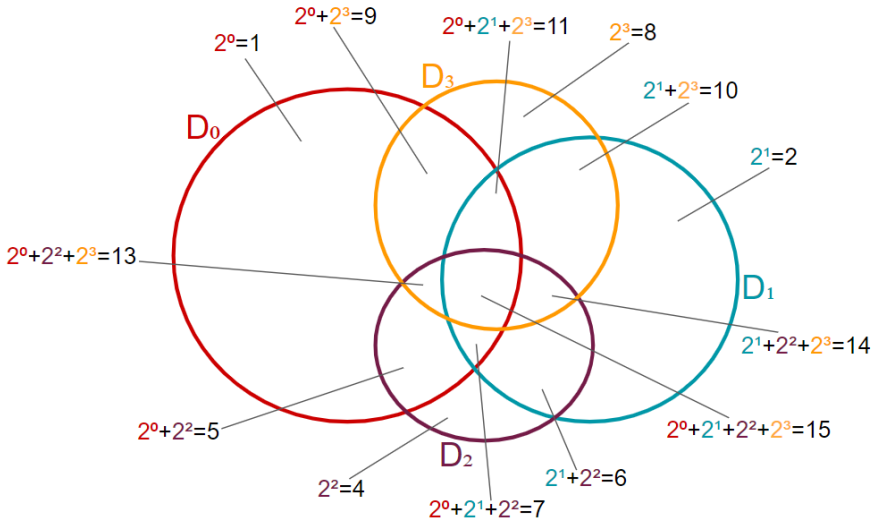

Figure 19: Visualisation representing different intersections by their unique integer representation based on which disease nodes are part of each hyperedge.

We can write this down mathematically using the incidence matrix M of this visualisation, resulting in a one dimensional vector $\boldsymbol{m}$ with unique entries.

$$M = \begin{array}{c} \\ \{D_0\} \\ \{D_1\} \\ \{D_2\} \\ \{D_3\} \\ \{D_0, D_2\} \\ \{D_0, D_3\} \\ \{D_1, D_2\} \\ \{D_1, D_3\} \\ \{D_0, D_1, D_2\} \\ \{D_0, D_2, D_3\} \\ \{D_0, D_1, D_3\} \\ \{D_1, D_2, D_3\} \\ \{D_0, D_1, D_2, D_3\} \end{array} \begin{array}{c} \begin{array}{cccc} 2^0(D_0) & 2^1(D_1) & 2^2(D_2) & 2^3(D_3) \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix} \end{array} \implies \boldsymbol{m} = \begin{pmatrix} 1 \\ 2 \\ 4 \\ 8 \\ 5 \\ 9 \\ 6 \\ 10 \\ 7 \\ 13 \\ 11 \\ 14 \\ 15 \end{pmatrix}. \tag{78}$$

The vector $\mathbf{m}$ provides a unique hyperedge to integer mapping (see section B.1). There are several advantages of this representation. For example, we can store the prevalence of each multimorbidity set (hyperedge) as an array indexed according to each hyperedge's unique integer representation, i.e. for an n-node hypergraph, we can have a 2^n array where the i^th entry corresponds to the prevalence of the hyperedge connecting the disease indexes which make up the binary string for the integer i.

Note also that we can store the hyperarc prevalence arrays in a similar fashion but in two-dimensions, the row indexes the binary-to-integer representation of the tail set of a hyperarc and the column indexes the disease node index of the head set. A benefit of using B-hyperarcs is that since $|H(e)| = 1$ for all edges $e \in \mathcal{E}$, we don't need to convert the head set to its unique integer representation. Therefore, for an n-node directed hypergraph, our hyperarc prevalence array can be a $2^n \times n$ matrix of which the $(i, j)^{\text{th}}$ element corresponds to the hyperarc whose tail set of disease nodes' unique integer representation is i and whose head node is the disease node indexed by j.

Another advantage is that, given a binary flag data matrix X of observations and their disease index flags, we can quickly determine the multimorbidity set(s) each individual contributes prevalence to because we can very quickly convert this two-dimensional data into a one-dimension vector of integers, allowing us to increment each prevalence counter in the hyperedge prevalence array above quickly and efficiently (see section B.2). This is because working with a one-dimensional array is much quicker than working with a two-dimensional array.

An interesting bi-product of this binary representation is is that if we were to remove one of the disease indexes, say D_3, then the hyperedge $\{D_0, D_1\}$ is now explicitly present in the data and indeed in M, since before all instances where D_0 and D_1 were coincident was with other conditions. What this shows is that depending on which diseases we care about including for the hyperedge-to-integer mapping, the code each individual has are subject to change and therefore we can build the prevalent arrays differently.

For example, if we were to convert the original data matrix X (observations indexed by row and disease nodes indexed by columns) using all disease nodes then we would create the most granular encoding of the individuals as possible — computing edge weights using this granular encoding only allows an individual to contribute to their own hyperedge, which corresponds to the exclusive set contribution type (see section 4.3.2.

If instead we were to only consider a subset of the disease noes then we would effectively be ignoring the remaining disease columns during our encoding. Computing the binary-to-integer mapping would change the encoding the individuals all had and therefore the contribution each individual makes to hyperedges changes. This has the effect of allowing an individual to contribute to their hyperedge's power set/progression set of hyperedges (see sections 4.3.1 and 4.3.3).

B Pseudocode implementation schemes

B.1 Representing hyperedges

Algorithm 1 Hyperedges as integer representations

Inputs: $N \times D$ hyperedge binary array `M`.
Outputs: Hyperedge integer representation array `H` of size `N`.

```
H ← zeros(N)                                   ▷ Initialise zeros array.
for i=0:N-1 do
    c ← 0                                      ▷ Initialise counter.
    for j=0:D-1 do
        c ← c + 2^j · M[i, j]                  ▷ Compute hyperedge integer representation.
    end for
    H[i] ← c                                   ▷ Update array.
end for
return H.
```

B.2 Prevalence counter (power & exclusive contribution)

Algorithm 2 Prevalence counter

Inputs: $N \times D$ binary flag matrix `M`. Index array `I` of size $d \leq D$.
Outputs: Prevalence counter array `P` of size 2^D.

```
P ← zeros(2^D)                                 ▷ Initialise zeros array.
for i=0:N-1 do
    c ← 0                                      ▷ Initialise counter.
    for j=0:d-1 do
        c ← c + 2^j · M[i, I[j]]               ▷ Increment counter.
    end for
    P[c] ← P[c] + 1                            ▷ Update prevalence array are hyperedge index c.
end for
return P
```

B.3 Overlap coefficient

Algorithm 3 Overlap Coefficient

Inputs: $N \times D$ data array D, hyperedge disease indices `inds`, hyperedge prevalence arrays P_r of size 2^D, population array P_o of size D and contribution type `cont`.
Outputs: Hyperedge weight `weight`.

```
nconds ← len(inds)
if cont = "power" then                         ▷ If power set contribution, compute prevalence.
    numerator ← PrevalenceCounter(D, inds)[-1]
else                                           ▷ Otherwise, fetch prevalence from prevalence array. binint ← 0
    for i=0:nconds-1 do
        binint ← binint + 2^inds[j]
    end for
    numerator ← P_r[binint]
end if
denominator ← P_d[inds[0]]                     ▷ Denominator is minimum value in population array
for i=1:nconds-1 do
    if P_d[inds[j]] < denominator then
        denominator ← P_d[inds[j]]
    end if
end for
weight ← numerator/denominator
return weight.
```

B.4 Modified Sorensen-Dice coefficient

Algorithm 4 Modified Sorensen-Dice Coefficient

Inputs: $N \times D$ hyperedge worklist array H, hyperedge degree array H_d of size N, hyperedge integer representation array H_r of size N, numerator and denominator prevalence arrays, P_n and P_d, of size 2^D.
Outputs: Hyperedge weights `weight` of size N.

`numweight` $\leftarrow$ `zeros(N)` ▷ Initialise numerator and denominator array for hyperedge weights.
`denomweight` $\leftarrow$ `zeros(N)`
for `i=0:N-1` **do**
    $e_i \leftarrow$ `H[i]`
    `ideg` $\leftarrow$ `H_d[i]`
    `iint` $\leftarrow$ `H_r[i]`
    `inumprev` $\leftarrow$ `P_n[iint]`
    `idenomprev` $\leftarrow$ `P_d[iint]`
    `numweight[i]` $\leftarrow$ `numweight[i]` $+$ `inumprev` ▷ Increment numerator of $w(e_i)$
    `denomweight[i]` $\leftarrow$ `denomweight[i]` $+$ `idenomprev` ▷ Increment denominator of $w(e_i)$
    $\mathcal{S} = \{e_j : e_i \subset e_j\}$ ▷ Calculate hyperedges e_j which are super-sets of hyperedge e_i.
    for j, e_j in $\mathcal{S}$ **do**
        `jdeg` $\leftarrow$ `H_d[j]`
        `jint` $\leftarrow$ `H_r[j]`
        `jdenomprev` $\leftarrow$ `P_d[jint]`
        $\Delta \leftarrow$ abs(`ideg` $-$ `jdeg`) + 1 ▷ Optional weight scheme. Ignored in current implementation.
        `denomweight[j]` $\leftarrow$ `denomweight[j]` $+$ `idenomprev` ▷ Increment denominator for $w(e_j)$.
        if computing CompleteSet Sorensen-Dice coefficient **then**
            `denomweight[i]` $\leftarrow$ `denomweight[i]` $+$ `jdenomprev` ▷ Increment denominator for $w(e_i)$.
        end if
    end for
end for
`weight` $\leftarrow$ `numweight`/`denomweight` ▷ Hyperedge weight $w(e_i) =$ `numweight[i]`/`denomweight[i]`.
return `weight`.

B.5 Constructing aggregated progression set

Algorithm 5 Aggregate progression set

Inputs: Individual's condition set `conds` and 2-duplicate index set `dupls`.
Outputs: Individual's progression set `fullprogset`

```
ndiseases ← len(conds)
maxdeg ← ndiseases − len(conds == -1)        ▷ conds contains ordered condition indexes with remaining entries as -1.
dupls ← dupls[dupls ≠ −1]                    ▷ dupls contains 2-duplicate indexes with remaining entries as -1.
ndupls ← len(dupls)
dummyvec ← −1 ∗ ones(ndiseases)              ▷ dummy vector to help build each progression hyperarc.
progset ← zeros(maxdeg-1, ndiseases)         ▷ Create progressiion set as if individual had no 2-duplicates.
for i=2:maxdeg do
    progset[i] ← concat(conds[:i], dummyvec[:ndiseases - i])
end for
if dupls[0] ≠ −1 then                        ▷ Individual has 2-duplicates.
    nnewhyps ← 2ndupls                       ▷ In general, 2 additional hyperarcs per 2-duplicate.
    if dupls[0] == 0 then                    ▷ If 2-duplicate at start of progression.
        nnewhyps ← nnewhyps − 1              ▷ Decrement number of new hyperarcs.
    end if
    extraprogset ← zeros(nnewhyps, ndiseases)   ▷ Initialise extra progression set array.
    j ← 0                                    ▷ Initialise counter for appending to extraprogset.
    for i=0:ndupls do                        ▷ Loop over 2-duplicates.
        idx ← dupls[i]
        deg0 ← conds[0]                      ▷ Extract condition indexes surrounding 2-duplicate
        deg1 ← conds[idx:idx+2][::-1]
        deg2 ← conds[idx+1]
        if idx == 0 then                     ▷ 2-duplicate at beginning requires 1 additional hyperarc.
            extraprogset[j] ← concat(deg1, dummyvec[:ndiseases - 2])
            j ← j + 1
        else if idx == 1 then
            hyp1 ← concat(deg0, deg2)
            hyp2 ← concat(deg0, deg1)
            extraprogset[mult*j-1] ← concat(hyp1, dummyvec[:ndiseases - len(hyp2)])
            extraprogset[mult*j] ← concat(hyp2, dummyvec[:ndiseases - len(hyp2)])
            j ← j + 1
        else                                 ▷ Any other 2-duplicate location, 2 hyperarcs generated.
            hypprev ← progset[idx-2]         ▷ We need a separate condition for idx == 1 because of this.
            hyp1 ← concat(hypprev, deg1)
            hyp2 ← concat(hypprev, deg2)
            extraprogset[mult*j-1] ← concat(hyp1, dummyvec[:ndiseases - len(hyp2)])
            extraprogset[mult*j] ← concat(hyp2, dummyvec[:ndiseases - len(hyp2)])
            j ← j + 1
        end if
    end for
    fullprogset ← concat(progset, extraprogset)
else                                         ▷ Individual has clean progression.
    fullprogset ← progset
end if
return fullprogset
```

B.6 Building incidence matrix and prevalence arrays

Algorithm 6 Incidence matrix & prevalence arrays

Inputs: $N \times D$ data matrix X, $N \times D$ individuals' condition set matrix C and $N \times {}^{D}/_{2}$ 2-duplicate index I. Progression function `progressionfunc`.
Outputs: Signed incidence matrix M, hyperarc worklist H, hyperarc prevalence array P_d, hyperedge prevalence array P_u, node prevalence array P_n and population prevalence array P_p.

`N, D ← dims(X)`
`Nhyperarcs ← Nhyperarcsfunc(ndiseases)`
`Nhyperedges ← 2^D`
`M ← zeros(Nhyperarcs, D), H ← zeros(Nhyperarcs, D)` ▷ Initialise output arrays
`P_d ← zeros(Nhyperedges, ndiseases), P_u ← zeros(Nhyperedges)`
`P_n ← zeros(2*D), P_p ← zeros(D)`
`n_row ← 0` ▷ Initialise row number for `M` and `H`.
for `i=0:N-1` **do** ▷ Loop over individuals
    `binmat ← X[i], conds ← C[i], dupls ← I[i]`
    `node_weight ← len(conds[conds -1])`
    `P_p[conds] ← P_p[conds] + 1` ▷ Increment population for each condition.
    if `dupls[0]` $\neq -2$ **then** ▷ If individual has more than 1 condition.
        `max_dupl ← max(dupls[0])`.
        if `max_dupl` $\neq$ `N_cond` $- 1$ **then** ▷ Given 2-duplicate not at end.
            `P_n[conds[:N_conds-1]] ← P_n[conds[:N_conds-1]] + node_weight` ▷ Increment tail prevalence.
            `P_n[D+conds[N_conds-1]] ← P_n[D+conds[N_conds-1]] + 1` ▷ Increment head prevalence.
        else ▷ Deal with 2-duplicate at end of progression.
            `P_n[conds[:N_conds-2]] ← P_n[conds[:N_conds-2]] + node_weight`
            `P_n[conds[N_conds-2:N_conds]] ← P_n[conds[N_conds-2:N_conds]] + node_weight/2`
            `P_n[D+conds[N_conds-2:N_conds]] ← P_n[D+conds[N_conds-2:N_conds]] + 1`
        end if
        if `dupls[0] == 0` **then** ▷ If 2-duplicate at start of progression.
            `hyp_cont` $\leftarrow 0.5$
            `hyp_i ← cond[:2]`
        else
            `hyp_cont` $\leftarrow 1.0$
            `hyp_i ← cond[:1]`
        end if
        `P_u[2^hyp_i] ← P_u[2^hyp_i] + hyp_cont` ▷ Increment hyperedge prevalence for first condition.
        `progset ← progressionfunc(conds, dupls)` ▷ Generate progression set.
        for `prog in progset` **do**
            `hyp_int ← 2^prog, tail_int ← 2^prog[:-1], head_node ← prog[-1]`
            `edge_cont` $\leftarrow 1$, `arc_cont` $\leftarrow 1$
            if parent hyperedge already contributed in `progset` **then** ▷ Takes account of hyperedge prevalence contribution.
                `edge_cont` $\leftarrow 0$
            end if
            if same degree hyperarc already contributed in `progset` **then** ▷ Takes account of hyperarc prevalence contribution.
                `arc_cont` $\leftarrow 0.5$
            end if
            if hyperarc `prog` not seen before now **then** ▷ Add to worklist and incidence matrix.
                `H[n_row] ← prog`
                `M[n_row, prog[:-1]]` $\leftarrow -1$, `M[n_row, prog[-1]]` $\leftarrow 1$
                `n_row ← n_row + 1`
            end if
            `P_u[hyp_int] ← P_u[hyp_int] + edge_cont` ▷ Increment prevalence arrays.
            `P_d[tail_int, head_node] ← P_d[tail_int, head_node] + arc_cont`
        end for
    else ▷ If individual has only 1 condition.
        `s ← conds[0]`
        `P_d[0, s] ← P_d[0, s] + 1` ▷ Increment hyperarc and hyperedge prevalence arrays.
        `P_u[2^s] ← P_u[2^s] + 1`
        `P_n[s] ← P_n[s] + 1` ▷ Increment node prevalence to both tail- and head-components.
        `P_n[D+s] ← P_n[D+s] + 1`
    end if
end for
return `M`, `H`, P_d, P_u, P_n, P_p

B.7 Adjacency matrix

Algorithm 7 Adjacency matrix

Inputs: $N \times D$ incidence matrix M_I, $N \times N$ edge weight matrix W_e, $D \times D$ node weight matrix W_n, representation `repr` weight resultant boolean `wres`. **Outputs:** Adjacency matrix `A`.

if `repr` $=$ standard **then** ▷ Representation decides on adjacency matrix of nodes or edges.
  $M \leftarrow M_I$
  $W \leftarrow W_e$
  $W_r \leftarrow W_n$
else
  $M \leftarrow M_I^{\mathrm{T}}$
  $W \leftarrow W_n$
  $W_r \leftarrow W_e$
end if
$N_r, N_c \leftarrow$ `dims(M)`
$M_w \leftarrow MW$
$M_{mwm} \leftarrow M_w M^{\mathrm{T}}$ ▷ Calculate MWM^{T}.
$M_{mwm} \leftarrow M_{mwm} -$ `diag(`M_{mwm}`)` ▷ Subtract diagonal, i.e. D_e or D_n.
if `wres` $=$ `True` **then** ▷ If weighted resultant, multiply both sides by $\sqrt{W_r}$
  $A \leftarrow \sqrt{W_r} M_{mwm} \sqrt{W_r}$
else
  $A \leftarrow M_{mwm}$
end if
return $N_r \times N_r$ matrix A.

B.8 Probability transition matrix

Algorithm 8 Probability transition matrix

Inputs: $D \times N$ tail and head incidence matrices M_- and M_+, $N \times N$ edge weight matrix W_e, $D \times D$ node degree matrix D_n, $N \times N$ edge degree matrix D_e, representation string `repr`, epsilon ϵ.
Outputs: Adjacency matrix `A`.

if `repr` $=$ successor **then** ▷ Successor detection — D_n is tail node degree matrix.
  $M_b \leftarrow M_-$
  $M_f \leftarrow M_+$
  $D_e \leftarrow \mathbb{I}_N$
else ▷ Predecessor detection — D_n is head node degree. D_e is head edge degree matrix.
  $M_b \leftarrow M_+$
  $M_f \leftarrow M_-$
end if
$P \leftarrow (D_n + \epsilon)^{-1} M_b W_e$ ▷ ϵ is used to prevent zero division.
$P \leftarrow P M_f^{\mathrm{T}} D_e^{-1}$
for `i=0:D-1` **do**
  if `sum(P, axis=1)` $= 0$ **then**
    `P[i,i]` $\leftarrow 1$ ▷ Make sure all rows in P sum to 1
  end if
end for
return $D \times D$ matrix P.

B.9 Centrality

Algorithm 9 Eigenvector Centrality (assume matrix is computed)

Inputs: $N \times N$ matrix (adjacency or transition) M, tolerance `tol`, maximum iterations `N_iter` and random seed `seed`, matrix type `type`.
Outputs: Dominant eigenvector $\boldsymbol{e}$ and eigenvalue λ.

```
N, N ← dim(M)
old_e ← random(N, seed=seed)
if type == "adjacency" then                    ▷ If adjacency matrix, normalise using L2 norm.
    old_e ← L2norm(old_e)
else                    ▷ If probability transition matrix, normalise using L1 norm to induce a probability vector.
    old_e ← L1norm(old_e)
end if
for i=0:N_iter-1 do
    new_e ← matrixmult(old_e, M)               ▷ New eigenvector estimate — left-hand multiplication.
    mask ← new_e ≠ 0 and old_e ≠ 0
    iter_e ← new_e[mask]/old_e[mask]
    new_eval ← mean(iter_e)                                ▷ Estimate new eigenvalue and error.
    eval_error ← std(iter_e)
    if eval_error/new_eval < tol then
        e ← new_e
        λ ← new_eval
        Break
    end if
end for
return e, λ
```

C Comorbidity Index Disease Tables

C.1 Charlson Comorbidity Index

Charlson Comorbidity Index	
Chronic Pulmonary Disease	Congestive Heart Failure
Connective Tissue Disease	Dementia
Diabetes (Complication)	Diabetes (w/o Complication)
Cancer, Lymphoma and Leukaemia	Liver Disease (Mild)
Liver Disease (Severe)	Metastatic Cancer
Myocardial Infarction	Paraplegia
Peptic Ulcer Disease	Peripheral Vascular Disease
Renal Failure	Stroke

Table 4: List of comorbidities in the Elixhauser Comorbidity Score [5].

C.2 Elixhauser Comorbidity Index

Elixhauser Comorbidity Index		
Alcohol Abuse	Anaemia Defficiency	Rheumatoid Arthritis
Blood Loss Anaemia	Cardiac Arrythmia	Congestive Heart Failure
Chronic Pulmonary Disease	Coagulopathy	Depression
Diabetes (Complication)	Diabetes (w/o Complication)	Drug Abuse
Hypertension	Hypothyroidism	Liver Disease
Lymphoma	Fluid and Electrolyte Disorders	Metastatic Cancer
Other Neurological Conditions	Obesity	Paralysis
Peripheral Vascular Disease	Psychoses	Pulmonary Circulation Disorder
Renal Failure	Solid Tumor without Metastasis	Peptic Ulcer Disease
Valvular Disease	Weight Loss	AIDS/HIV

Table 5: List of comorbidities in the Elixhauser Comorbidity Score [7].